SPECTRA
Computer Code Manuals

Volume 1 – Program Description

Version 24-02

M.M. Stempniewicz

Arnhem, February 2024

K6223/24.277594 MSt-2402

author : M.M. Stempniewicz reviewed : E.A.R. de Geus

approved : J.J. Meulenbrugge

837 page(s)

Spectra-Vol1.doc

Contents

Abstract

SPECTRA (**S**ophisticated **P**lant **E**valuation **C**ode for **T**hermal-hydraulic **R**esponse **A**ssessment) is a fully integrated system analysis code, that models thermal-hydraulic behavior of Nuclear Power Plants, including reactor cooling system, emergency and control systems, containment, reactor building, etc. of various reactor types, like BWR, PWR, HTR. It can also be used to assess thermal-hydraulic response of non-nuclear plants, for example cooling systems of chemical reactors.

The full documentation of SPECTRA consists of the following four volumes:

- Volume 1: Program Description
- Volume 2: User's Guide
- Volume 3: Verification and Validation
- Volume 4: Code Structure, Development, Hardware and Software Requirements

This report presents Volume 1 of the SPECTRA Code Manuals – Program Description. The phenomenological models implemented in the code are described in this volume. The description is organized by packages.

The SPECTRA Manuals are freely available in internet and are also supplied together with the SPECTRA code. The Volume 1 of the Code Manuals is provided in the file **Spectra-Vol1.pdf**.

Explanation of names and abbreviations

CF	Control Function
CV	Control Volume
DIA	Diagnostics file
ICF	Initial Condition File
IT	Isotope Transformation
JN	CV Junction
MP	Material Properties
OUT	Output file
OX	Metal Oxidation
PLT	Plot file
RK	Reactor Kinetics
RT	Radioactive Particle Transport
SC	1-D Solid Heat Conductor
SPECTRA	**S**ophisticated **P**lant **E**valuation **C**ode for **T**hermal-hydraulic **R**esponse **A**ssessment
TC	2-D Solid Heat Conductor
TF	Tabular Function
TFD	Tabular Function Data file
TR	Thermal Radiation

1 Introduction

SPECTRA (**S**ophisticated **P**lant **E**valuation **C**ode for **T**hermal-hydraulic **R**esponse **A**ssessment) is a fully integrated system analysis code, that models the thermal-hydraulic behavior of Nuclear Power Plants, including reactor cooling system, emergency and control systems, containment, reactor building, etc. of various reactor types, like BWR, PWR, HTR. It can also be used to assess thermal-hydraulic response of non-nuclear plants, for example cooling systems of chemical reactors. The full documentation of SPECTRA consists of the following four volumes:

- Volume 1: Program Description
- Volume 2: User's Guide
- Volume 3: Verification and Validation
- Volume 4: Code Structure, Development, Hardware and Software Requirements

This report presents Volume 1 of the SPECTRA Code Manuals – Program Description. The phenomenological models implemented in the code are described in this volume. The description is organized by packages.

The structure of SPECTRA is shown in Figure 1-1. The structure of the program is described in terms of *Packages*, *Models* and *Equations*, defined below.

- ***"Packages"***
  The total program consists of 20 blocks, referred to as "*Packages*". Each *Package* is identified in Figure 1-1 by a two character symbol enclosed by "=" signs. The *Packages* are characterized by:
  - Each *Package* resides in a separate subdirectory of the SPECTRA source code directory.
  - Description of each *Package* is given in a separate chapter of this report.

- ***"Models"***
  Packages consist of physical or mathematical models, referred to as "*Models*". An example of a physical *Model* is the critical flow model. An example of a mathematical *Model* is a matrix solver (i.e. set of procedures to solve a set of linear equations). *Models* available within each *Package* are shown below, in Figure 1-2 through Figure 1-6. The *Models* are characterized by:
  - Each *Model* is written as a separate file (FORTRAN files with extension .FOR).
  - Description of each *Model* is given in this report in a separate section.

- ***"Equations"***
  Models consist of one or more procedures, referred to in this report as "*Equations*". For example the critical flow *Model* consists of *Equations* used for subcooled liquid, superheated steam, perfect gas, etc. The matrix solution *Model* consists of *Equations* to solve full matrices, sparse matrices, tri-diagonal matrices, etc. The *Equations* are characterized by:
  - Each *Equation* is written as a separate code segment (subroutine or function).
  - Description of each *Equation* is given in a separate sub-section.

All *Packages*, except for the Main Program, =SPE=, and the numerical solver, =SL=, are grouped into five "Groups of *Packages*" in Figure 1-1. The Groups of *Packages* are introduced here for descriptive convenience. They have no meaning for the internal code structure. The first two Groups of *Packages* contain physical *Models*. The next Group contains mathematical *Models*.

The last Group of *Packages* contains procedures needed for input processing. It should be noted that this Group contains only the main I/O procedures. Most of the *Packages* from the first four Groups have their own specific routines for input processing and editing data.

As indicated by the connecting lines in Figure 1-1, the SPECTRA Main Program, =SPE=, interacts only with I/O procedures and the Numerical Solver. The task of Main Program is to use the I/O Procedures to read input and to initiate the solution by a call to the Numerical Solver.

The Numerical Solver *Package* is responsible for solving all *Equations*, using a stable, implicit method. As indicated in Figure 1-1, the Numerical Solver interacts with each *Package* from the first four Groups of *Packages*. All *Packages* are included in the main iteration loop to obtain the implicit solution. The Solver *Package* is general enough to find solutions of even complex problems. When the solution does not converge using the attempted time step, the Solver automatically reduces the time step.

Most of the code segments (*Equations*) from the first three *Groups of Packages* reflect the real physical laws. Subroutines or functions, in which the *Equations* are coded, contain extensive comments and references to literature where the equations can be found. This makes verification and eventual future modifications relatively easy.

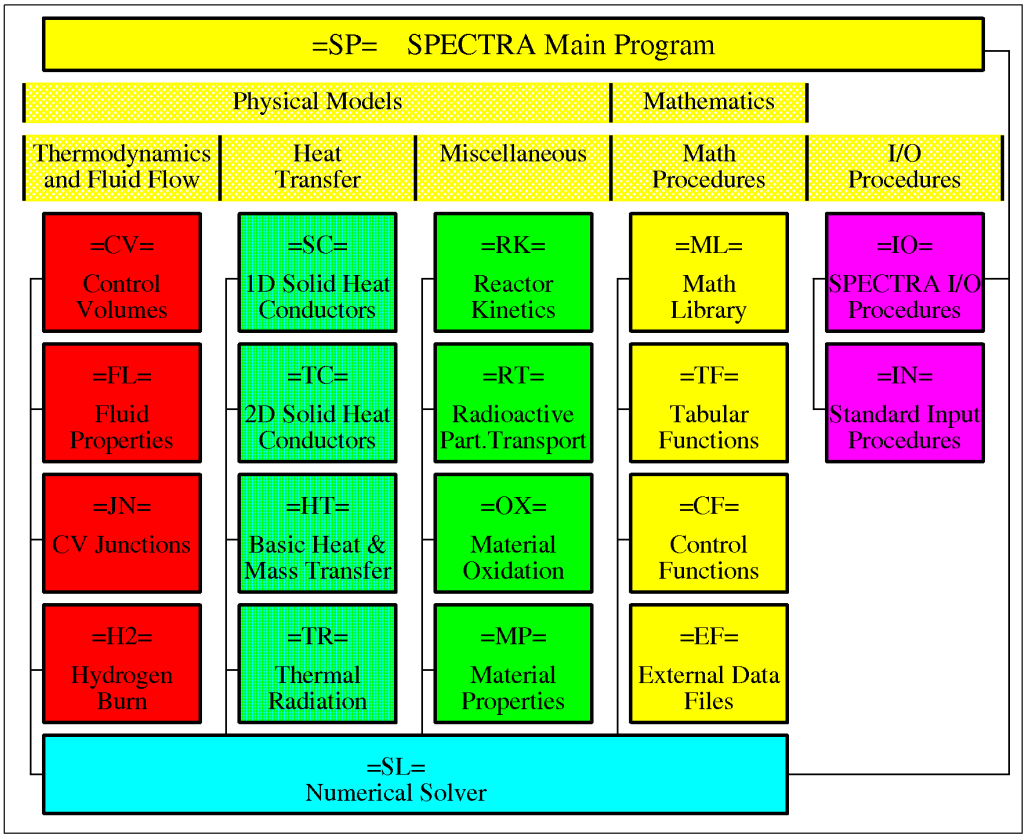


Figure 1-1 SPECTRA code structure.

Figure 1-2 shows *Packages* from the Thermodynamics and Fluid Flow Group. *Models* available in each *Package* are also shown. The *Models* are marked by filled bullets (filled squares). The main assumptions, made to develop the *Models*, are marked by empty bullets (empty squares). This Group consists of the following *Packages*:

=CV= Control Volume *Package*, described in Chapter 2. The main *Models* in this *Package* include: mass end energy balance, atmosphere stratification and pool stratification.

=FL= Fluid Property *Package*, described in Chapter 3. The gas *Model* considers six built-in gases, treated as real gases, and a number of user-defined gases, treated as ideal gases. Phase change is possible only for steam, for which the saturation (liquid/gas) properties are built-in. In addition to the built-in fluids, user-defined gases (e.g. Ne, Ar, Xe) and liquids (e.g.: liquid metals, molten salts) may be used.

=JN= Junction *Package*, described in Chapter 4. The *Models* in this *Package* are: momentum balance, drift flux, vent phenomena, critical flow, wall friction, two-phase pressure drop, pump/compressor and turbine *Models*, and the *Model* to influence gas composition in a junction, for conservative analyses.

=H2= Hydrogen Burn *Package*, described in Chapter 11. The *Models* in this *Package* are: temperature-dependent flammability and ignition limits, hydrogen burns including slow deflagrations, fast turbulent deflagrations and detonations.

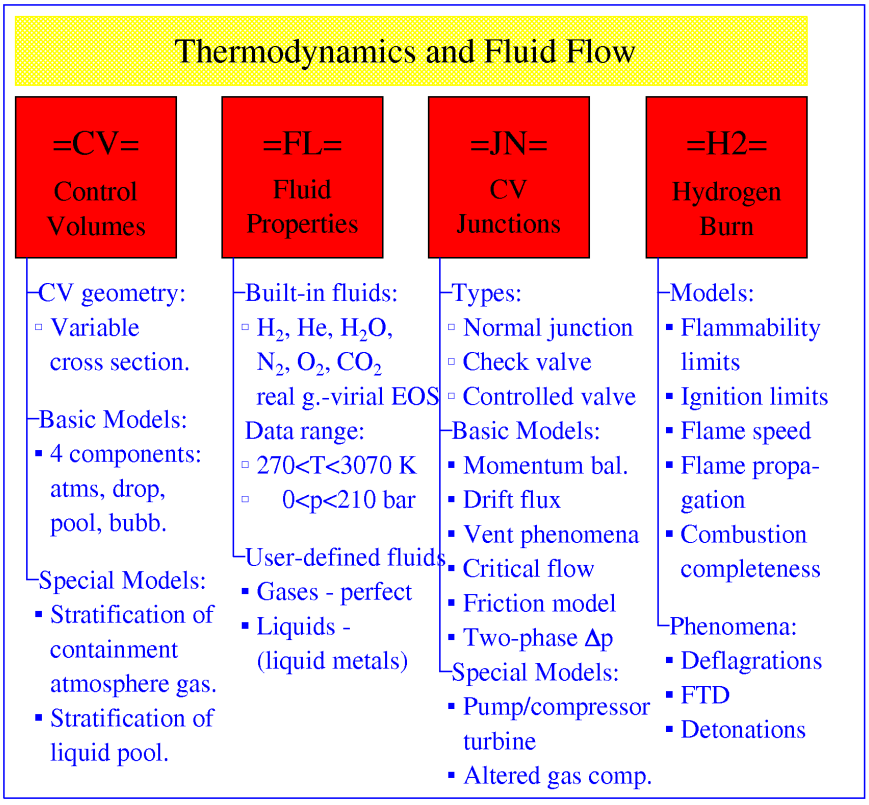


Figure 1-2 SPECTRA code structure - Thermodynamics and Fluid Flow Packages.

Figure 1-3 shows *Packages* from the Heat Transfer Group. This Group consists of the following *Packages*:

=SC= 1-D Solid Heat Conductor *Package*, described in Chapter 5. This *Package* contains one *Model*: 1-D transient heat conduction. Multi-dimensional effects may be simulated be connecting several 1-D structures using the so called direct contact conduction model.

=TC= 2-D Solid Heat Conductor *Package*, described in Chapter 6. This *Package* contains one *Model*: 2-D transient heat conduction. 3-D effects may be simulated be connecting several 2-D structures using the so called direct contact conduction model.

=HT= Basic Heat and Mass Transfer *Package*, described in Chapter 7. This *Package* contains the main *Models* for heat transfer from wall surface and pool surface consisting of wall-to-pool, wall-to-gas and pool-to-gas sub-models. The above *Models* use the following individual heat transfer *Models*: natural convection, forced convection, boiling, including nucleate, transition and film boiling, condensation, heat transfer in two-phase flow. The *Package* includes also the *Model* of non-equilibrium vapor generation rate.

=TR= Thermal Radiation *Package*, described in Chapter 8. This *Package* contains net enclosure, grey gas approximation radiative heat exchange *Models*, *Models* for gas radiation properties and surface radiation properties.

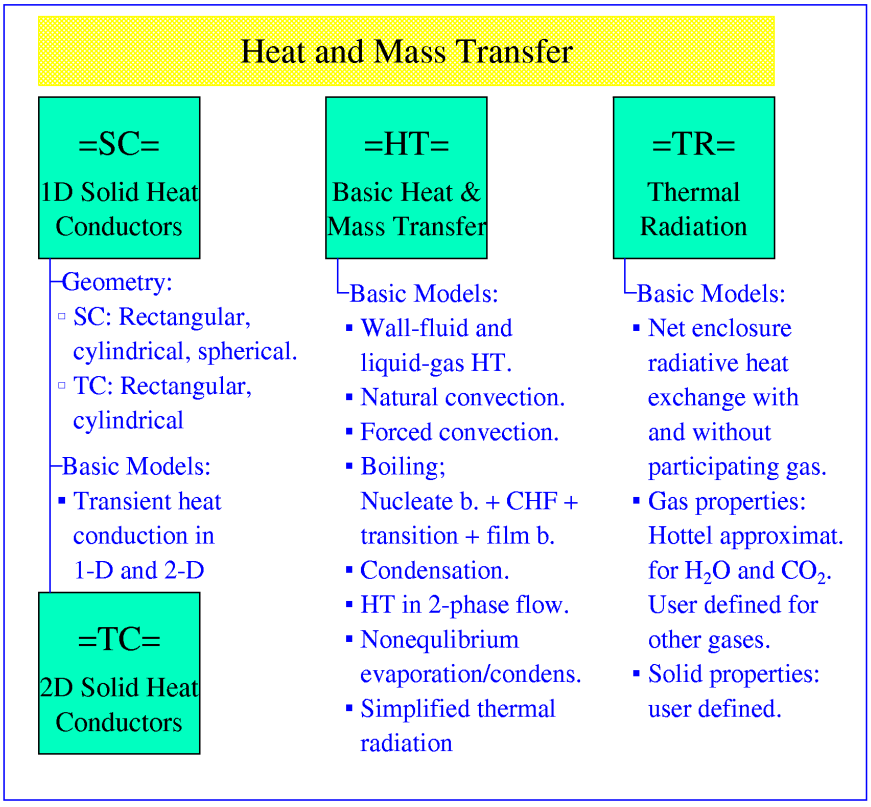


Figure 1-3 SPECTRA code structure - Heat and Mass Transfer Packages.

Figure 1-4 shows *Packages* from the Miscellaneous Group. This Group consists of the following *Packages*:

=RK= Reactor Kinetics *Package*, described in Chapter 9, includes the point kinetics (space-independent), the nodal point kinetics, and the circulating fuel *Models* with reactivity feedbacks from fuel and moderator temperature changes, void fraction changes, and changes of isotope concentrations (for example poisons such as Xe-135). The latter effect is calculated by the Isotope Transformation *Model*.

=RT= Radioactive Particle Transport *Package*, described in Chapter 12, includes aerosol transport, deposition and resuspension, radioactive isotope chains, fission product release, transport sorption of fission product vapors on surfaces.

=OX= Metal Oxidation *Package*, described in Chapter 10, includes *Models* for zircaloy and steel oxidation, as well as a general *Model* for oxidation of any metal. The general *Model* uses parabolic reaction rate. The coefficients for the parabolic reaction rate are defined by the user.

=MP= Material Property *Package*, described in Chapter 14. This *Package* defines tables used to define properties of materials used for 1-D and 2-D solid heat conductors.

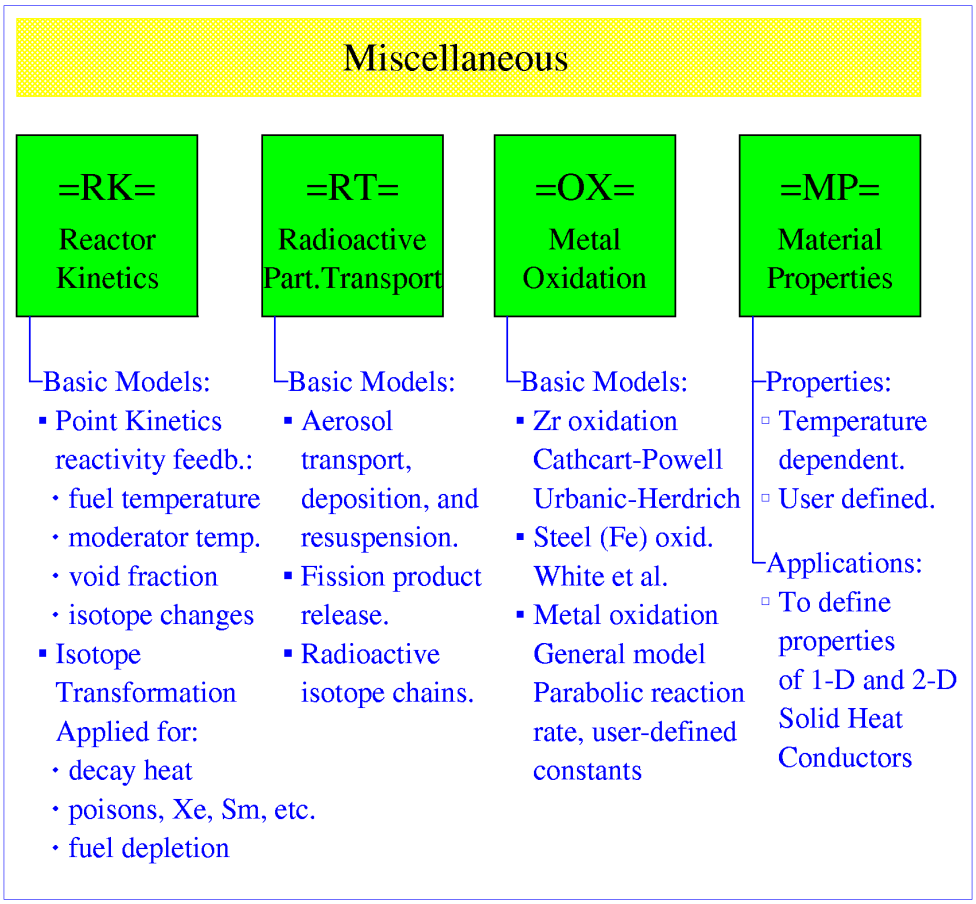


Figure 1-4 SPECTRA code structure - Miscellaneous Packages.

Figure 1-5 shows *Packages* from the Math Procedures Group. This Group consists of the following *Packages*:

=ML= Math Library *Package*, described in Chapter 17. This package contains standard models taken from the Numerical Recipes in FORTRAN [1].

=TF= Tabular Function *Package*, described in Chapter 15. Tabular functions allow the user to define time dependent parameters. All tabular functions are assumed to be functions of time. This allows eliminating Tabular Functions from the iteration procedure to obtain the implicit solution. Tabular Functions may be defined interactively during program execution. With the interactive TF the user or an external code can communicate with SPECTRA during the execution; thus allowing the use of SPECTRA as a background code in a plant simulator.

=CF= Control Function *Package*, described in Chapter 16. These are general functions, dependent on arbitrary arguments. Control Functions are by default included in the main iteration loop to obtain the implicit solution.

=EF= External Data File *Package*, described in Chapter 18. This package contains functions for writing and reading information to and from external files.

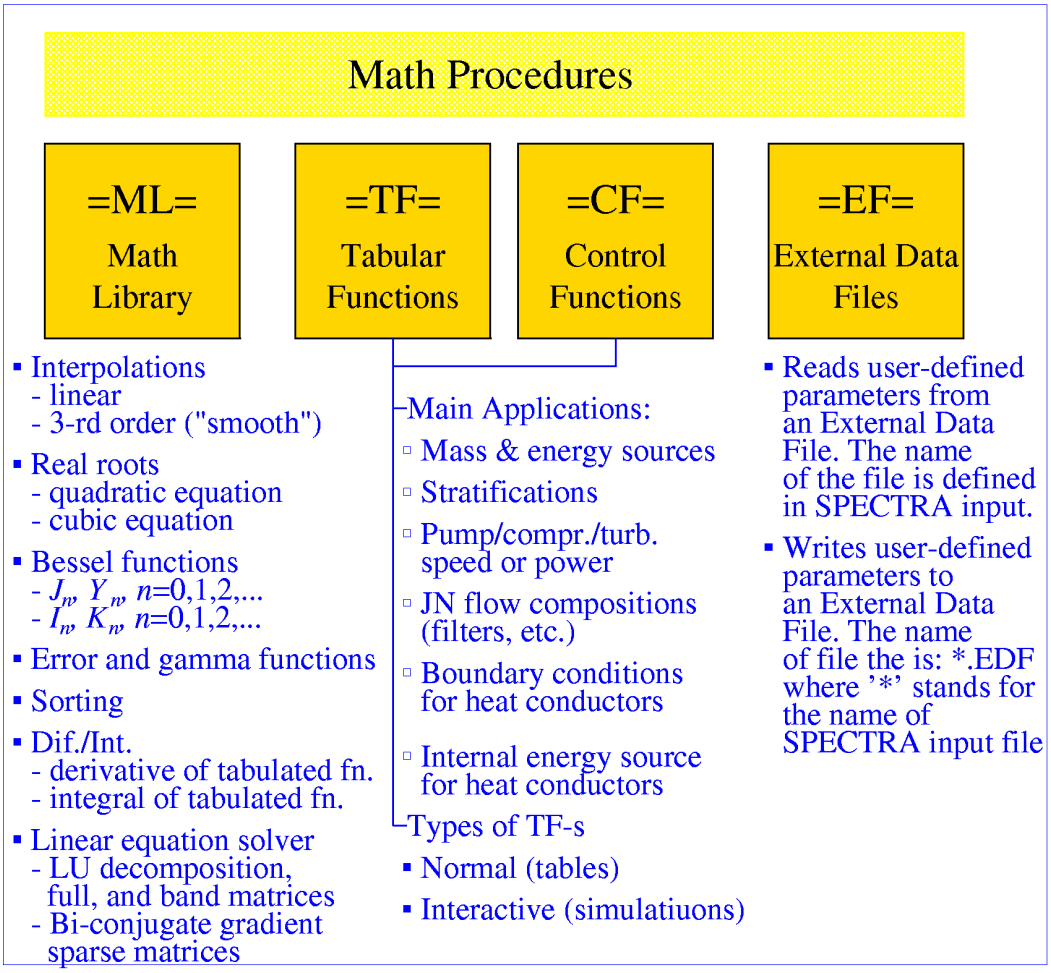


Figure 1-5 SPECTRA code structure - Math Packages.

Figure 1-6 shows *Packages* from the I/O Procedures Group. This Group consists of the following *Packages*:

=IO= SPECTRA I/O Procedures *Package*. The *Package* contains the main procedures to read input files and write output files. This *Package* is only a "coordinator" of input and output processing. The following *Packages*: =CV=, =JN=, =SC=, =TC=, =TR=, =TF=, =CF=, =H2=, =OX=, =MP=, =SL=, have their own procedures, responsible for reading, checking, and writing data belonging to the appropriate *Package*.

=IN= Standard Input Procedures *Package*. The *Package* contains standard procedures to perform preliminary processing of input (remove comments, perform file attachments, etc.).

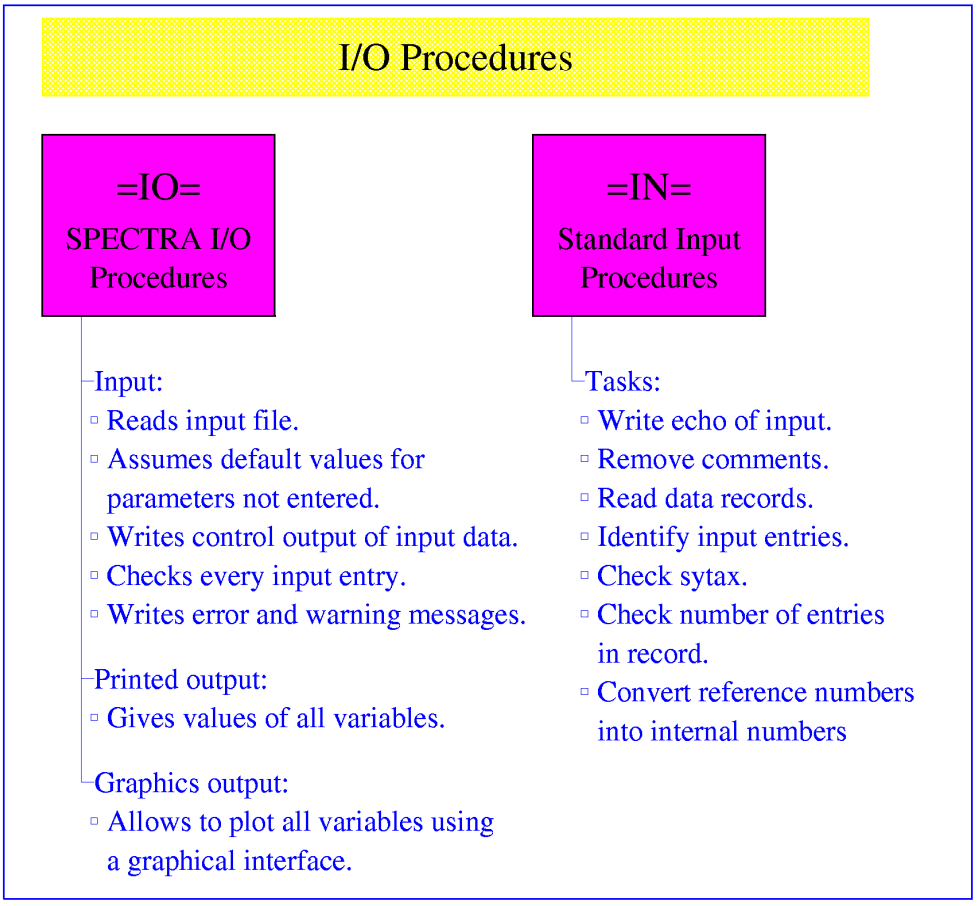


Figure 1-6 SPECTRA code structure - Input/Output Packages.

The last two *Packages*, not belonging to any of the Groups discussed above, are the Numerical Solver *Package* (Figure 1-7) and the SPECTRA Main Program.

=SL= The Numerical Solver *Package*, described in Chapter 19. This *Package* is responsible for solving the problem using the implicit scheme (the Control Functions may be excluded individually or globally from the implicit solution by the user).

=SPE= The SPECTRA Main Program. The tasks of the Main Program are:

- To initiate reading and processing input. This is done by a call to the main subroutine from the =IO= *Package*.
- To initiate problem solution if the input is read properly. This is done by a call to the main subroutine from the =SL= *Package*. If errors are detected by =IO= *Package* then Main Program terminates the execution and writes an appropriate message.

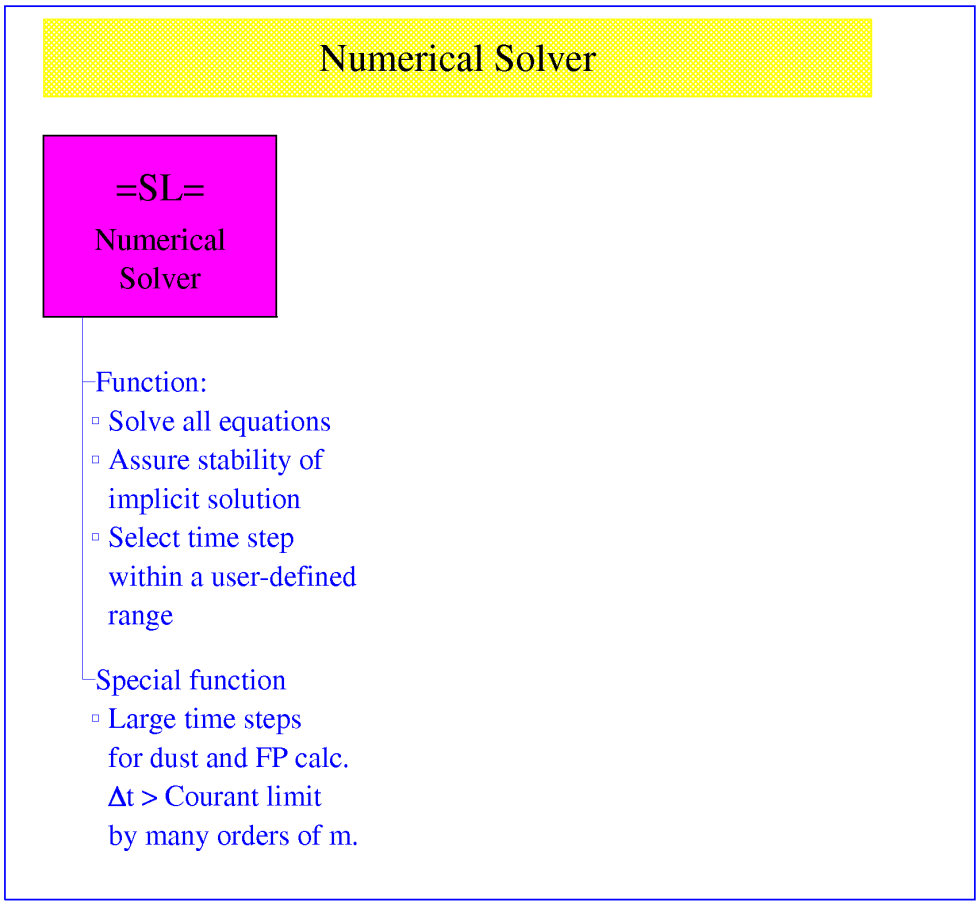


Figure 1-7 SPECTRA code structure - Numerical Solver Package

2 Control Volume Package

2.1 Introduction

This section gives a general description of Control Volumes (CV) and their mathematical treatment. The same models and solution schemes are used for all Control Volumes. Like in the MELCOR code, within the basic Control Volume formulation the treatment is quite general, no specific nodalization is built-in.

The discussion of Control Volumes begins with a description of the CV geometry, presented in section 2.2. Section 2.3 shows what kind of fluids may be present in a Control Volume. Section 2.4 presents the mass and energy conservation equations, used by the CV Package. The next section shows how the velocities inside a Control Volume are calculated. Section 2.6 presents additional equations, which are used for the dispersed components (bubbles and droplets). This include the particle count and the particle position equation, which are used to determine the size of the average particle, and the position of the average particle in a Control Volume. Finally, section 2.7 contains a discussion of the simplified stratification models.

2.2 Control Volume Geometry

Generally a Control Volume is intended to represent a physical "room", or a containment compartment. A Control Volume geometry is characterized by horizontal cross sections and heights. Each Control Volume can be composed of a number of segments, each segment with a different horizontal cross section area and a different height (Figure 2-1).

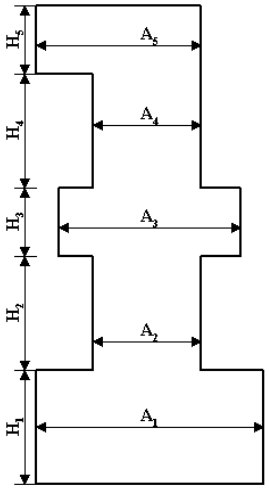


Figure 2-1 Control Volume geometry

The cross section areas may have any values. The only restriction required by the code is that the cross section areas should be positive. The segments are assumed to be stacked above each other. It is assumed that no flow restrictions exist between the segments. No loss factors are associated with segment boundaries.

This approach is similar to that taken in MELCOR [46]. It gives a good way to represent rather closely the geometrical layout of containment compartments, which are often of irregular shapes.

2.3 Control Volume Contents

The materials which can reside in Control Volumes include water, its vapor, and several noncondensable gases. The contents of each Control Volume is divided into two fields, referred to as pool and atmosphere (Figure 2-2). The pool surface, separating those fields, is assumed to be a horizontal plane. Each field is composed of two components: continuous and discontinuous (or dispersed) component.

- Atmosphere of a Control Volume.
  - Continuous component: gas mixture, composed of noncondensables and water vapor.
  - Discontinuous component: water droplets.

- Pool of a Control Volume.
  - Continuous component: water
  - Discontinuous component: bubbles, composed of a gas mixture.

The approach to discontinuous component modelling is based on the assumption that the particles can be well represented by their average values: average size and average position. Those quantities are calculated by the program using the particle count and particle position equations, described in section 2.6. The particle movement is tracked using a drift flux model and empirical correlations for the bubble rise velocity and the droplet fall-down velocity.

Nonequilibrium is assumed between each of the four components in a Control Volume. That means each component is at its own temperature. There are therefore four temperatures in each Control Volume (Table 2-1). Each component has its own velocity. The velocity model calculates vertical and horizontal velocities for each component in a Control Volume. Thus there are eight velocities in each Control Volume (Table 2-1). Note that the assumed positive direction for vertical velocity is always towards the pool surface (see Figure 2-2).

The pressure in a Control Volume is defined at the pool-atmosphere interphase. Therefore the atmosphere and the pool pressures are always the same. Pressures at certain distance above (or below) the pool are obtained, whenever needed, by including the gravity head of the gas (or water) column above (or below) the pool surface. The droplet pressure is assumed to be equal to the atmospheric pressure. The bubble pressure is obtained by including the gravity head of water column between the pool surface and the average bubble position.

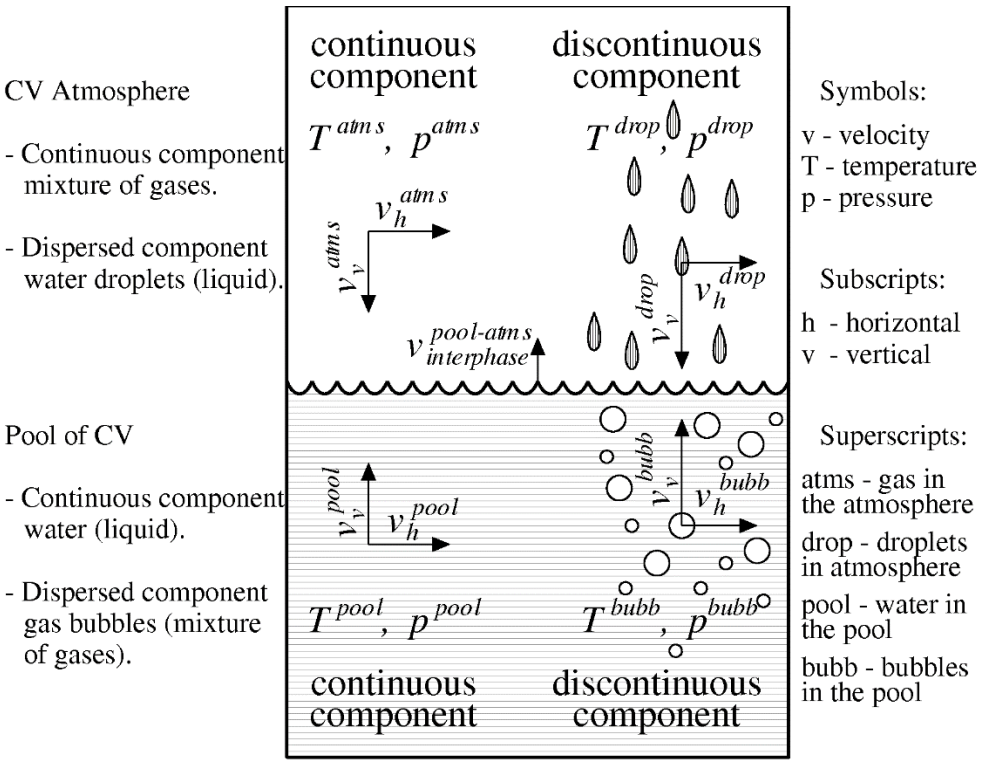


Figure 2-2 Contents of a Control Volume in SPECTRA

Table 2-1 Control Volume thermodynamic parameters

Field	Continuous component		Discontinuous component	
Atmosphere	gas		droplets	
	- temperature	T_{atms}	- temperature	T_{drop}
	- pressure	p_{atms}	- pressure	p_{drop}
	- vertical velocity	$v_{v,atms}$	- vertical velocity	$v_{v,drop}$
	- horizontal velocity	$v_{h,atms}$	- horizontal velocity	$v_{h,drop}$
Pool	water		bubbles	
	- temperature	T_{pool}	- temperature	T_{bubb}
	- pressure	p_{pool}	- pressure	p_{bubb}
	- vertical velocity	$v_{v,pool}$	- vertical velocity	$v_{v,bubb}$
	- horizontal velocity	$v_{h,pool}$	- horizontal velocity	$v_{h,bubb}$

2.4 Mass and Energy Conservation

The thermal-hydraulic behavior in a Control Volume is governed by the equations of conservation of mass and energy. The mass and energy conservation equations take into account several physical mass and energy transfer processes which may occur in a Control Volume, as well as the "user prescribed" mass and energy sources.

The term "Mass Transfer" is used in this section to signify both the mass transfer and the corresponding energy (enthalpy) transfer. For example, a condensation mass flux is accompanied by a certain energy loss for the atmosphere gas, and energy gain for the condensate liquid. The packages responsible for the mass transfer calculations give always both mass transfer and the corresponding enthalpy fluxes. The term "Energy Transfer" is used in this section for those processes which do not involve mass transfer but pure energy (heat) transfer. For example convective heat transfer from the surface of Solid Heat Conductors.

The mechanisms which are taken into account in the mass and energy transfer equations for the atmosphere of a Control Volume are graphically illustrated in Figure 2-3.

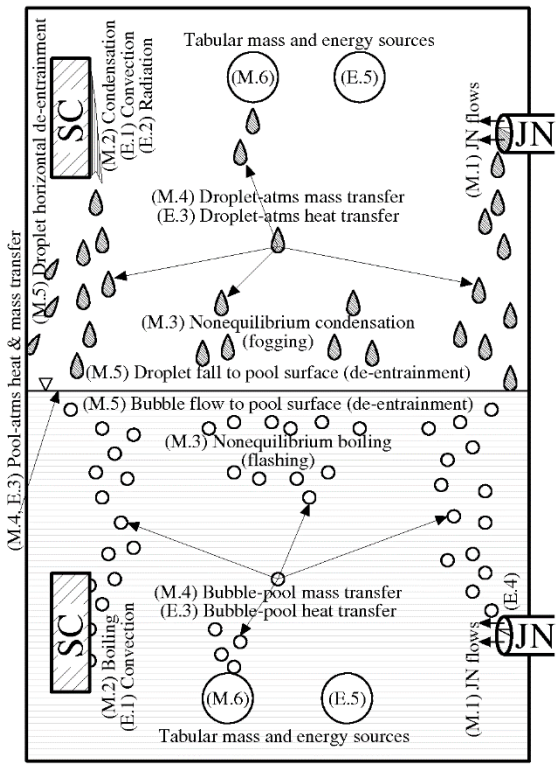


Figure 2-3 Mass and Energy transfer processes in a Control Volume

The following mass and heat transfer mechanisms are taken into account:

Mass Transfer:	(M.1)	Junction flows
	(M.2)	Condensation or boiling on 1-D and 2-D Solid Heat Conductors
	(M.3)	Nonequilibrium mass transfer (flashing or fogging)
	(M.4)	Interphase mass transfer (evaporation or condensation)
	(M.5)	De-entrainment
	(M.6)	Tabular mass sources
	(M.7)	H_2O source, H_2, O_2 sink due to hydrogen burn
	(M.8)	Gas sources and sinks due to oxidation reactions
Energy Transfer:	(E.1)	Convection from 1-D and 2-D Solid Heat Conductors
	(E.2)	Thermal radiation
	(E.3)	Interphase energy transfer (convection)
	(E.4)	Energy source to pool due to bubble collapse
	(E.5)	Tabular energy sources
	(E.6)	Energy source due to hydrogen burn
	(E.7)	Energy sources from pumps compressors and turbines
	(E.8)	Energy source due to radioactive isotopes present in CV

Note that the oxidation heat is considered as a heat source for the structures (1-D or 2-D Solid Heat Conductor) where the reaction takes place and therefore is not on the list of energy sources for a Control Volume.

2.4.1 Equation of Mass Conservation

The mass conservation equations for each of the four components, that may reside in a Control Volume (i.e. atmosphere, pool, droplets, and bubbles), are written in a general form as:

$$\frac{dM_{k,comp}}{dt} = \sum_{m=1}^{8} W_{m,k,comp} \qquad comp = atms, bubb$$

$$\frac{dM_{comp}}{dt} = \sum_{m=1}^{8} W_{m,comp} \qquad comp = pool, drop$$

$M_{k,comp}$ mass of the gas No. *k*, in the component No. *comp*, (kg),
M_{comp} mass of water in the component No. *comp*, (kg),
$W_{m,k,comp}$ mass source of gas No. *k* for the component No. *comp*, due to the mass transfer process No. *m*, (kg/s),
$W_{m,k,comp}$ mass source of water in the component No. *comp*, due to the mass transfer process No. *m*, (kg/s).

The summation is performed over all mass transfer processes (*m* = 1, 2, ..., 8), listed above. The values of the mass source rates are discussed below for the eight mass transfer processes.

- Junction flows (m=1)

The junction flow solution is performed by the CV Junction Package, and is described below, in section 4.2. The values obtained from the CV Junction Package include velocities for all Junctions and all components, as well as the component flow areas. The calculation of the component flow areas depends on the water levels in the connecting Control Volumes, as well as the atmosphere and pool velocities through the counter-current flow limit, as described in section 4.2. Once those parameters are known, the task of the Control Volume Package is to distribute the flows among the components present in a Control Volume. This task is not trivial, and a certain degree of freedom in modelling is given to the user by introducing entrainment indicators. Each of the four components can be assigned a different value of entrainment indicator. The default values of all indicators are zero, and the actual values vary within the limits: 0.0 - 1.0. The influence of the entrainment indicators on the calculations is illustrated in Figure 2-4, and is discussed below.

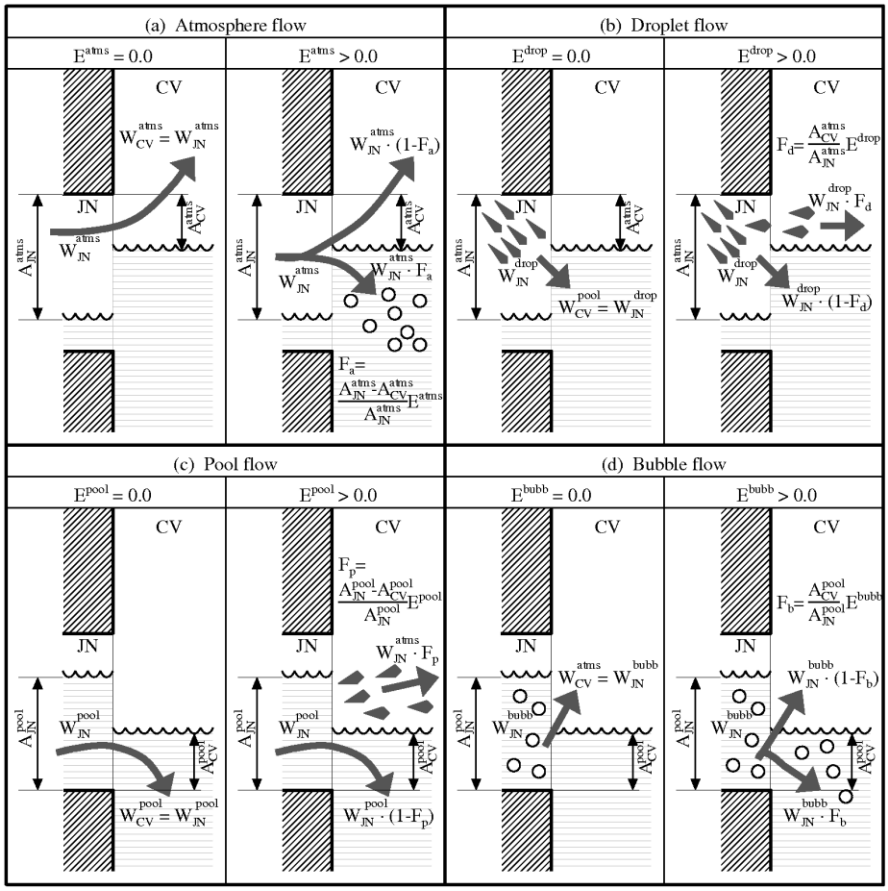


Figure 2-4 Influence of entrainment indicators on distribution of junction flows

In case of atmosphere flow through a junction (Figure 2-4.a) the atmosphere entering the Control Volume is partitioned between atmosphere and bubbles as follows.

- If the water level in the receiving CV is below the water level in the Junction, then all flow is placed in the atmosphere component of the receiving CV.
- If the water level in the receiving CV is above the top of Junction, then all flow is placed in the bubble component of the receiving CV.
- In case of intermediate water level the atmosphere flow is partitioned between the atmosphere and the bubble components as follows:

$$W_{CV,atms} = W_{JN,atms}(1 - F_a)$$
$$W_{CV,bubb} = W_{JN,atms} F_a$$

The factor F_a is the fraction of the atmosphere gas that is entrained into the pool of the receiving CV. The entrained fraction is obtained from the following formula:

$$F_a = \frac{A_{JN,atms} - A_{CV,atms}}{A_{JN,atms}} E_{atms}$$

Here E_{atms} is the entrainment indicator for atmosphere flow, while $A_{JN,atms}$, $A_{CV,atms}$ are the atmosphere areas in the junction and the control volume, as shown in Figure 2-4.

In case of droplet flow through the junction (Figure 2-4.b) the fraction of droplets entrained into the atmosphere of the receiving control volume, F_d, is equal to:

$$F_d = \frac{A_{CV,atms}}{A_{JN,atms}} E_{drop}$$

E_{drop} is the entrainment indicator for droplet flow. The above equation is used for $A_{CV,atms} \leq A_{JN,atms}$. If the water level in the receiving CV is below the water level in the junction, then $F_d = E_{drop}$. Thus F_d remains always within limits: $0 \leq F_d \leq E_{drop}$.

Similar definitions are used in determining the entrainment factors for pool and bubble flow. The entrainment factors for pool flow, F_p, and bubble flow, F_b are defined as:

$$F_p = \frac{A_{JN,pool} - A_{CV,pool}}{A_{JN,pool}} E_{pool}$$
$$F_b = \frac{A_{CV,pool}}{A_{JN,pool}} E_{bubb}$$

$A_{JN,pool}$, $A_{CV,pool}$ are the pool areas in the junction and the control volume, as shown in Figure 2-4 (c) and (d). As above, the limits are imposed, such that: $0 \leq F_p \leq E_{pool}$, $0 \leq F_b \leq E_{bubb}$.

Test calculations showed that the most accurate results are obtained by using the bubble entrainment factor, E_{bubb}, equal to one, while the other entrainment factors have very little influence on overall results, and the values of zero, which lead to minimum entrainment, are sufficient. It has to be remembered that presence of droplets and bubbles in a control volume results in numerically difficult ("stiff") equations, therefore entrainment should be avoided. The bubble entrainment factor, E_{bubb}, has been observed to have an important effect on natural circulation in a pool, consisting of a "chimney" and a "downcomer". In such case flow oscillations may sometimes be observed, driven by dragging of bubbles into the downcomer. Those oscillations were predicted with E_{bubb} equal to one. The correctness of the predicted flow oscillations have been verified by comparison calculations, performed with the TRACG code (see Volume 3).

The overall mass transfer rate due to junction flows into a given Control Volume is obtained by summing the values calculated for all junctions which are connected to this Control Volume. The general form of the equations defining mass sources due to junction flows may be written as follows:

$$W_{1,k,atms} = \sum_{JN \in CV} W_{JN,atms} \cdot (1 - F_a) \cdot c_{k,JN,atms} + W_{JN,bubb} \cdot (1 - F_b) \cdot c_{k,JN,bubb}$$

$$W_{1,drop} = \sum_{JN \in CV} W_{JN,pool} \cdot F_p + W_{JN,drop} \cdot F_d$$

$$W_{1,pool} = \sum_{JN \in CV} W_{JN,pool} \cdot (1 - F_p) + W_{JN,drop} \cdot (1 - F_d)$$

$$W_{1,k,bubb} = \sum_{JN \in CV} W_{JN,atms} \cdot F_a \cdot c_{k,JN,atms} + W_{JN,bubb} \cdot F_b \cdot c_{k,JN,bubb}$$

where $c_{k,JN,comp}$ is the mass fraction of the gas No. *k*, in the flow of the component No. *comp*, through the junction *JN*. The above set of equations looks very regular, because of certain symmetries that may be observed. It is easily seen that a sum of the second and third equations gives the total liquid flow (pool and droplets) for all junctions connected to a given CV. Thus the amount of water that appears in a given CV, in the form of pool or droplets, is exactly the same as the mass of water which is being transported through the junctions (and thus disappearing from another CV). The same can be said about the flows of all gases. The above formulation is therefore conserving mass.

In spite of its regularities, the above set of equations is not the one used by the code. The above equations would be used if the bubble collapse models were disabled for all junctions (which can be done, but is not recommended - see Volume 2). The bubble collapse model, described in section 7.2.5, is used for gas flowing into a pool of a Control Volume. Thus the bubble collapse model is used for both, atmosphere flowing into a pool of a CV, and bubbles flowing into a pool of CV. During the collapse process steam is partially or totally condensed. The uncondensed steam, along with non-condensable gases constitute a mass source for the bubbles, while the condensed steam constitutes a mass source for the pool. The actual set of equations defining the CV mass sources due to junction flows is written as:

$$W_{1,k,atms} = \sum_{JN \in CV} W_{JN,atms} \cdot (1 - F_a) \cdot c_{k,JN,atms} + W_{JN,bubb} \cdot (1 - F_b) \cdot c_{k,JN,bubb}$$

$$W_{1,drop} = \sum_{JN \in CV} W_{JN,pool} \cdot F_p + W_{JN,drop} \cdot F_d$$

$$W_{1,pool} = \sum_{JN \in CV} W_{JN,pool} \cdot (1 - F_p) + W_{JN,drop} \cdot (1 - F_d) + W_{liq,BC-JN,atms} + W_{liq,BC-JN,bubb}$$

$$W_{1,k,bubb} = \sum_{JN \in CV} W_{gas,BC-JN,atms} \cdot c_{k,BC-JN,atms} + W_{gas,BC-JN,bubb} \cdot c_{k,BC-JN,bubb}$$

$W_{liq,BC\text{-}JN,comp}$ is a mass source of liquid, (kg/s), obtained in the process of collapse of bubbles, occurring during the flow of component *comp* (*comp = atms, bubb*), through the junction *JN*, into a pool of the Control Volume *CV*.

$W_{gas,BC\text{-}JN,comp}$ is a mass source of gas (non-condensables + uncondensed steam), (kg/s), obtained in the process of collapse of bubbles, occurring during the flow of component *comp* (*comp = atms, bubb*), through the junction *JN*, into a pool of the Control Volume *CV*.

The following relation is preserved in the bubble collapse model:

$$(W_{JN,comp} F_c) = W_{liq,BC-JN,comp} + W_{gas,BC-JN,comp}$$

This ensures the mass conservation of the final formulation of the equations determining $W_{1,comp}$. In the above equation the term: $W_{JN,comp}F_c$ is used to signify the input to the bubble collapse process, for either atmosphere flow, $W_{JN,atms}F_a$, or bubble flow, $W_{JN,bubb}F_b$.

- Mass transfer on the surface of Solid Heat Conductors (*m*=2)

The mass transfer on 1-D and 2-D Solid Heat Conductors includes condensation and boiling. Boiling occurs when the temperature of the SC/TC surface immersed in water is above the saturation temperature. Condensation occurs when the temperature of the SC/TC surface above water level is below the saturation temperature at the steam partial pressure. Boiling provides a negative mass source for the pool of a control volume, and a positive mass source for the bubbles. Condensation provides a negative source of steam for the atmosphere of a CV, and a positive source for either droplets or pool of a CV, depending on the location of the lower edge of the Solid Heat Conductor relative to the water level. In case of condensation a slight additional complication is introduced by modelling of the condensate film flowing along several Solid Heat Conductors, stacked one above another. Consequently, as the mass source for droplets or pool, only that mass is taken which comes from the lowermost SC/TC stack member.

The general form of the equation defining mass sources due to SC/TC mass transfer is therefore written as follows:

$$W_{2,k,atms} = -\sum_{SC\in CV} A_{SC} X_{atms} m_{cond,SC} \delta_{k,steam}$$

$$W_{2,dropor\ pool} = \sum_{\substack{SC\in atms \\ SC'\notin stack}} \left(\sum_{SC\in SC'} A_{SC} X_{atms} m_{cond,SC} \right)$$

$$W_{2,pool} = \sum_{SC\in pool} \left(- A_{SC} X_{pool} m_{boil,SC} + W_{liq,BC-SC} \right)$$

$$W_{2,k,bubb} = \sum_{SC\in pool} W_{gas,BC-SC} \delta_{k,steam}$$

A_{SC} surface area of Solid Heat Conductor SC/TC, (m^2)
X^{atms} fraction of SC/TC above the pool of control volume, (-)
X^{pool} fraction of SC/TC immersed in the pool of control volume, (-)
$m_{cond,\ SC}$ condensation mass transfer, ($kg/m^2/s$)
$m_{boil,\ SC}$ boiling mass transfer, ($kg/m^2/s$)
$\delta_{k,\ steam}$ Kronecker delta, (-), equal to one if *k* means steam, and zero for all other gases
$W_{liq,\ BC\text{-}SC}$ mass source of liquid obtained in the process of bubble collapse, (kg/s)
$W_{gas,\ BC\text{-}SC}$ mass source of gas obtained in the process of bubble collapse, (kg/s)

In the second equation the superscript "*drop or pool*" means that this term is added either to droplet, or to pool, depending on the relative location of the SC/TC lower edge and the pool level. The first sum is over the all Solid Heat Conductors, SC', which are not stacked above some other heat conductor. The second sum is performed over all SC stacked above SC'. The last two equations contain the Bubble Collapse (BC) parameters: $W_{liq,\ BC\text{-}SC}$, $W_{gas,\ BC\text{-}SC}$. This is a consequence of the modelling approach to boiling. The bubbles produced at the boiling surface first undergo a bubble collapse process, in which they partially or fully condenses. The following relation is preserved in the bubble collapse model:

$$A_{SC} X_{pool} m_{boil,SC} = W_{liq,BC-SC} + W_{gas,BC-SC}$$

This ensures the mass conservation for the SC/TC mass transfer processes. It is seen that with this relation the sum of all four terms defining $W_{2,comp}$ gives zero.

The values of condensation and boiling mass transfer, m_{cond}, m_{boil}, are calculated by using correlations present in the Heat Transfer Package (sections 7.1, 7.2.1, 7.2.2). The bubble collapse parameters are calculated by the bubble collapse model, present in the Heat Transfer Package (section 7.2.5).

- Non-equilibrium mass transport (*m*=3)

Non-equilibrium mass transfer includes non-equilibrium boiling (flashing) and non-equilibrium condensation (fogging). Flashing occurs when the water temperature is above the saturation temperature. Fogging occurs when the gas temperature is below the saturation temperature at steam partial pressure. Nonequilibrium mass transfer may occur in each of the four components (although it is rather unlikely to occur in droplets and bubbles).

Nonequilibrium boiling in a pool results in steam bubbles being suspended in the pool. Nonequilibrium condensation in an atmosphere results in water droplets being suspended in the atmosphere. In case of droplets steam from the nonequilibrium boiling is added to the atmosphere gas. In case of bubbles condensate from the nonequilibrium condensation is added to the water pool.

The equations defining mass sources due to nonequilibrium mass transfer are:

$$W_{3,k,atms} = (V_{atms}\Gamma_{cond,atms} + V_{drop}\Gamma_{boil,drop})\delta_{k,steam}$$
$$W_{3,drop} = -V_{atms}\Gamma_{cond,atms} - V_{drop}\Gamma_{boil,drop}$$
$$W_{3,pool} = -V_{pool}\Gamma_{boil,pool} - V_{bubb}\Gamma_{cond,bubb}$$
$$W_{3,k,bubb} = (V_{pool}\Gamma_{boil,pool} + V_{bubb}\Gamma_{cond,bubb})\delta_{k,steam}$$

V_{comp} component volume, (m^3)
$\Gamma_{cond,comp}$ nonequilibrium condensation mass transfer in the component No. *comp* (*comp = atms, bubb*), ($kg/m^3/s$)
$\Gamma_{boil,comp}$ nonequilibrium boiling mass transfer in the component No. *comp* (*comp = pool, drop*), ($kg/m^3/s$)

As above, $\delta_{k,\ steam}$ is the Kronecker delta, used to distinguish steam from other gases. The convention assumed for the nonequilibrium mass transfer model is that the positive values always mean evaporation. Therefore:

$$\Gamma_{cond,comp} \le 0.0$$
$$\Gamma_{boil,comp} \ge 0.0$$

The sources for the gas components (atmosphere, bubbles) are taken with the positive signs, while the mass sources for the liquid components (pool, droplets) are taken with the negative sign. As it is easily seen, the sum of all four terms defining $W_{3,comp}$ gives zero, which means that the overall mass is conserved in the process.

The values of nonequilibrium mass transfer rates, Γ_{cond}, Γ_{boil}, are calculated by the non-equilibrium mass transfer model, present in the Heat Transfer Package (section 7.1.14).

- Interphase mass transport (*m*=4)

The four component modelling approach, taken for the CV fluid space representation, results in three different interphase boundaries, where mass and energy transfer processes occur:

a) Pool - atmosphere.
b) Droplet - atmosphere.
c) Bubble - pool.

To determine the values of mass and energy transfer at these interphase boundaries one needs to know the heat/mass transfer areas. While the pool-atmosphere area is readily available from the CV geometry and the water level, the calculation of the two other areas is a little more complicated, since it requires knowledge of the particle size. As will be described below, the code does keep track of the particle size (section 2.6) which allows to determine the required heat/mass transfer areas at any time.

The convention taken for the interphase mass transfer model is the same as in case of nonequilibrium mass transfer, that means positive mass transfer means evaporation, while negative means condensation. The equations determining mass sources due to interphase mass transfer processes are therefore written as:

$$W_{4,k,atms} = (A_{drop-atms} m_{in,drop-atms} + A_{pool-atms} m_{in,pool-atms}) \delta_{k,steam}$$
$$W_{4,drop} = -A_{drop-atms} m_{in,drop-atms}$$
$$W_{4,pool} = -A_{bubb-pool} m_{in,bubb-pool} - A_{pool-atms} m_{in,pool-atms})$$
$$W_{4,k,bubb} = A_{bubb-pool} m_{in,bubb-pool} \delta_{k,steam}$$

A means interphase area (m^2), while m_{in} means interphase mass transfer rate, (kg/m^2-s). The interphase areas of the dispersed components are determined as:

$$A_{drop-atms} = N_{drop} \pi D_{drop}^2$$
$$A_{bubb-pool} = N_{bubb} \pi D_{bubb}^2$$

N is the particle number and D is the particle diameter, determined as described in section 2.6. The mass conservation is fulfilled which may easily be checked by summing up all four terms defining $W_{4,comp}$.

The values of mass transfer rates, m_{in}, for each of the three interphase boundaries, are calculated by the appropriate model from the Heat Transfer Package (sections 7.2.3, 7.2.4, 7.2.6).

- De-entrainment of droplets and bubbles (m=5)

The following three processes:
  - bubbles flowing to the pool surface,
  - droplets hitting the pool surface (or dry floor),
  - droplets hitting the walls,

are commonly referred to as de-entrainment processes (Figure 2-5). The mass transfer rate due to bubble de-entrainment is calculated from the formula:

$$W_{v,bubb} = A_{in,p-a} \alpha_{bubb} \rho_{bubb} (v_{v,bubb} - v_{in,p-a})$$

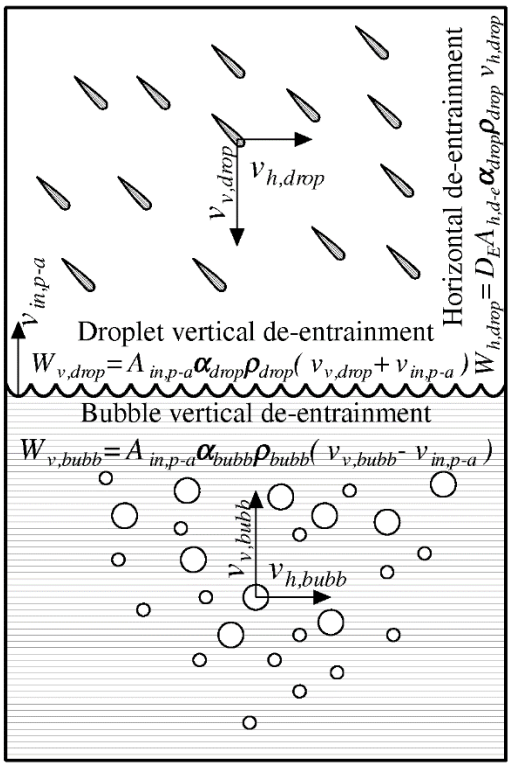


Figure 2-5 De-entrainment of droplets and bubbles

$A_{in,p-a}$ area of pool-atmosphere interphase, (m2)
α_{bubb} local volumetric fraction of bubbles at the pool surface, (-)
ρ_{bubb} density of bubbles, (kg/m3)
$v_{v,bubb}$ vertical velocity of bubbles, (m/s)
$v_{in,p-a}$ velocity (upwards) of pool-atmosphere interphase, (m/s)

A similar equation is used to calculate vertical de-entrainment of droplets. In case of droplets however, the droplet velocity and the pool surface velocities are used:

$$W_{v,drop} = A_{in,p-a} \alpha_{drop} \rho_{drop} (v_{v,drop} - v_{in,p-a})$$

α_{drop} local volumetric fraction of droplets at the pool surface, (-)
ρ_{drop} density of droplets, (kg/m^3)
$v_{v,drop}$ vertical velocity of droplets, (m/s)

Horizontal droplet de-entrainment occurs due to droplets hitting the walls with the horizontal velocity, $v_{h,drop}$. The area for horizontal droplet de-entrainment, $A_{h,\,d\text{-}e}$, is calculated based on the CV flow area for the flow in horizontal direction (input parameter) and the actual water level, as that part of the horizontal flow area which is above the pool surface. The droplet horizontal de-entrainment is equal to:

$$W_{h,drop} = D_E A_{in,p-a} \bar{\alpha}_{drop} \rho_{drop} v_{h,drop}$$

where $\bar{\alpha}_{drop}$ is the average volumetric fraction of droplets in the CV atmosphere, and D_E is a user defined multiplier, introduced to allow more flexibility in modelling horizontal de-entrainment.

Finally, the equations determining mass sources for each of the four CV components due to the de-entrainment processes, are:

$$\begin{aligned}
W_{5,k,atms} &= A_{in,p-a} \alpha_{bubb} \rho_{bubb} (v_{v,bubb} - v_{in,p-a}) c_k \\
W_{5,drop} &= -A_{in,p-a} \alpha_{drop} \rho_{drop} (v_{v,bubb} - v_{in,p-a}) - D_E A_{h,d-e} \bar{\alpha}_{drop} \rho_{drop} v_{h,drop} \\
W_{5,pool} &= A_{in,p-a} \alpha_{drop} \rho_{drop} (v_{v,bubb} - v_{in,p-a}) + D_E A_{h,d-e} \bar{\alpha}_{drop} \rho_{drop} v_{h,drop} \\
W_{5,k,bubb} &= -A_{in,p-a} \alpha_{bubb} \rho_{bubb} (v_{v,bubb} - v_{in,p-a}) c_k
\end{aligned}$$

Here c_k is the mass fraction of the gas No. k in bubbles. Again, the formulation is conserving mass, since the sum of all $W_{5,comp}$ gives zero.

- Tabular mass sources (m=6)

Several mass sources may be present in a Control Volume. The mass source data consists of the composition of source, tables of mass flow rates, temperatures, and pressure, as well as the source geometry data: elevation, flow area, diameter. Tables defining the parameters may be simple functions of time, or (through Control Functions) more complicated functions, depending on any calculated variables.

Depending on the source location relative to the actual water level in a Control Volume, the fluid from mass sources may enter either into the continuous components (atmosphere, pool), or the dispersed components (droplets, bubbles), as shown in Figure 2-6. In case of dispersed component sources, the initial size of droplets and bubbles are determined from correlations shown in section 2.6, using the flow area and diameter of the source.

In absence of the bubble collapse model the equations determining the mass source for each of the four CV components would be written as follows:

$$\begin{aligned}
W_{6,k,atms} &= \sum_{MS \in atms} W_{MS} (1 - x_{liq,MS}) c_{k,MS} \\
W_{6,drop} &= \sum_{MS \in atms} W_{MS} x_{liq,MS} \\
W_{6,pool} &= \sum_{MS \in pool} W_{MS} x_{liq,MS} \\
W_{6,k,bubb} &= \sum_{MS \in pool} W_{MS} (1 - x_{liq,MS}) c_{k,MS}
\end{aligned}$$

W_{MS} mass flow rate from the mass source No.: *MS*, (kg/s)
$x_{liq,\ MS}$ mass fraction of liquid in the mass source *MS*, (-)
$c_{k,\ MS}$ mass fraction of gas *k*, in the mass source *MS*, (-)

As in case of junction flows, the bubble collapse model is introduced for the gas sources which are present in the pool. As a consequence, the actual set of equations defining $W_{6,comp}$ as used by the code is as follows:

$$W_{6,k,atms} = \sum_{MS \in atms} W_{MS}(1 - x_{liq,MS})c_{k,MS}$$

$$W_{6,drop} = \sum_{MS \in atms} W_{MS} x_{liq,MS}$$

$$W_{6,pool} = \sum_{MS \in pool} (W_{MS} x_{liq,MS} + W_{liq,BC-MS})$$

$$W_{6,k,bubb} = \sum_{MS \in pool} W_{gas,BC-MS} c_{k,BC-MS}$$

where $W_{liq,\ BC\text{-}MS}$ is the mass source of liquid, (kg/s), obtained in the process of collapse of bubbles, $W_{gas,\ BC\text{-}MS}$ is the mass source of gas (non-condensables + uncondensed steam), (kg/s), obtained in the bubble collapse process, $c_{k,\ BC\text{-}MS}$ is the gas composition of the bubble at the end of the collapse process, calculated by the bubble collapse model (section 7.2.5).

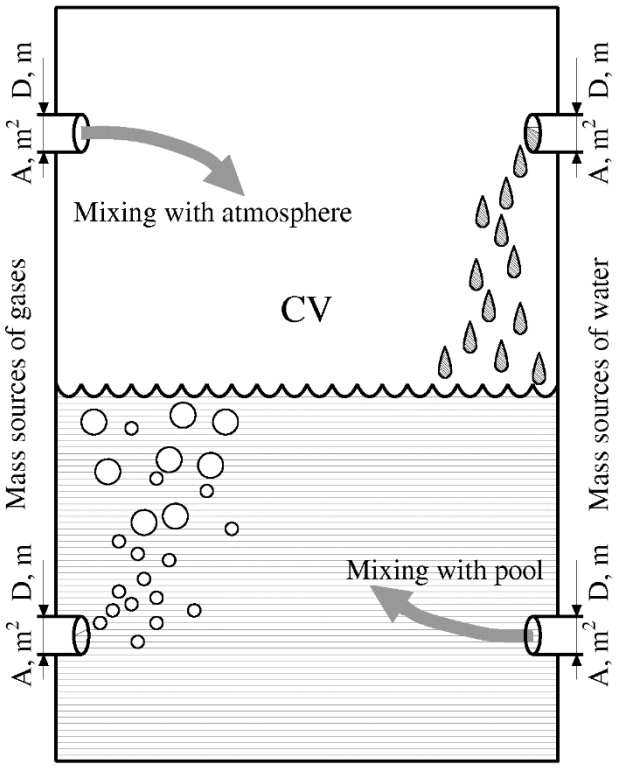


Figure 2-6 Tabular mass sources

With the relation:

$$W_{MS}(1-x_{liq,MS}) = W_{liq,BC-MS} + W_{gas,BC-MS}$$

the mass is conserved in the final formulation of the equations determining $W_{6,comp}$, that means the overall mass, which appears in a control volume in various forms (atmosphere, pool, droplets, bubbles), is exactly the same as the overall mass being supplied with all tabular mass sources. The program keeps track of the supplied masses, by calculating time integrals of all mass source rates, which allows to perform an overall check of the overall mass conservation at any time.

- Hydrogen burn (m=7)

The hydrogen burn reactions are considered as mass sources and sinks for the atmosphere gases. The mass sources are:

$$W_{7,k=H_2,atms} = -\frac{dm_{H_2}}{dt}$$

$$W_{7,k=O_2,atms} = -\frac{dm_{O_2}}{dt}$$

$$W_{7,k=H_2O,atms} = \frac{dm_{H_2O}}{dt}$$

The mass sinks of hydrogen and oxygen, dm_{H2}/dt, dm_{O2}/dt, as well as the mass source of steam, dm_{H2O}/dt, are calculated by the hydrogen burn package, described in section 11.4.

- Oxidation reactions (m=8)

The oxidation reactions are considered as mass sources and sinks for the atmosphere gases. The mass sources are:

$$W_{8,k,atms} = \sum_{SC \in CV} (W_{ox,2}\delta_{k,2} - W_{ox,1}\delta_{k,1}) \cdot A_{SC}$$

$W_{ox,1}$ mass transfer rate of the reacting gas per unit surface area, (kg/m^{2}s)
$W_{ox,2}$ mass transfer rate of the reaction product gas per unit surface area, (kg/m^{2}s)
A_{SC} surface area, (m^2), of the 1-D or a 2-D Solid Heat Conductor, where the reaction occurs; summation is performed over all surfaces present in given CV
δ_{kj} Kronecker delta, equal to 1 when gas j is equal to k, zero otherwise

The mass transfer rates are obtained from:

$$W_{ox,1} = \frac{dm}{dt} n_1 \frac{M_{w,1}}{M_{w,Mt}}$$

$$W_{ox,2} = \frac{dm}{dt} n_2 \frac{M_{w,2}}{M_{w,Mt}}$$

m mass of oxidized material per unit surface area (kg/m^2)
n_1 stoichiometry ratio 1 – number of moles of the oxidant gas per mole of the oxidized material
n_2 stoichiometry ratio 2 – number of moles of the product gas per mole of the oxidized material
$M_{w,1}$ molar weight of the gas 1
$M_{w,1}$ molar weight of the gas 2
$M_{w,Mt}$ molar weight of the oxidized material

The above formulae for the mass transfer rates, $W_{ox,1}$ $W_{ox,2}$, are based on the general definition of the oxidation reaction (section 10.3):

$$Mt + n_1 Gs_1 \rightarrow MtOx + n_2 Gs_2$$

Mt oxidized material
Gs_1 oxidant gas
Gs_2 product gas

2.4.2 Equation of Energy Conservation

The energy conservation equations for each of the four components, that may reside in a Control Volume (i.e. atmosphere, pool, droplets, and bubbles), are written in a general form as:

$$\frac{dU_{comp}}{dt} = \sum_{m=1}^{8} (W_m h_m)_{comp} + \sum_{n=1}^{8} Q_{n,comp} - p_{comp} \frac{dV_{comp}}{dt}$$

U_{comp} internal energy of the component No. *comp,* (J)
$W_{m,comp}$ mass source for the component No. *comp,* from the mass transfer process No. *m,* (kg/s). In case of gas sources the value is equal to the sum of masses of all individual gases in the mixture:

$$W_{m,comp} = \sum_{i=1}^{N_{gas}} W_{m,k,comp}$$

$h_{m,comp}$ specific enthalpy associated with the mass transfer process No. *m,* (J/kg)
$Q_{n,comp}$ energy source for the component No. *comp,* from the energy transfer process No. *n,* (W)
p_{comp} pressure of the component No. *comp,* (Pa)
V_{comp} volume of the component No. *comp,* (m^3)

Eight mass transfer processes (m = 1, 2, ..., 8), and eight energy transfer processes (n = 1, 2, ..., 8), listed above, are taken into account in the energy conservation equation. The term: $p_{comp}dV_{comp}/dt$ represents the work done on the given component.

The formulae defining the enthalpy fluxes associated with mass sources, and the heat fluxes from the energy sources are given below, for the eight mass sources and the eight energy sources. The individual terms of mass transfer processes are very similar to the corresponding terms in the mass conservation equation and are therefore introduced here without further explanations.

- Junction flows (*m*=1)

  The equations determining energy sources for each of the four components that may reside in a Control Volume, are:

$$(W_1 h_1)_{atms} = \sum_{JN \in CV} W_{JN,atms} \cdot (1 - F_a) \cdot h_{JN,atms} + W_{JN,bubb} \cdot (1 - F_b) \cdot h_{JN,bubb}$$

$$(W_1 h_1)_{drop} = \sum_{JN \in CV} W_{JN,pool} \cdot \quad F_p \quad \cdot h_{JN,pool} + W_{JN,drop} \cdot \quad F_d \quad \cdot h_{JN,drop}$$

$$(W_1 h_1)_{pool} = \sum_{JN \in CV} W_{JN,pool} \cdot (1 - F_p) \cdot h_{JN,pool} + W_{JN,drop} \cdot (1 - F_d) \cdot h_{JN,drop} + H_{liq,BC-JN,atms} + H_{liq,BC-JN,bubb}$$

$$(W_1 h_1)_{bubb} = \sum_{JN \in CV} H_{gas,BC-JN,atms} + H_{gas,BC-JN,bubb}$$

  $h_{JN,comp}$ specific enthalpy, (J/kg), of component *comp*, flowing through the junction *JN*

  $H_{liq,BC\text{-}JN,comp}$ enthalpy flux, (J/s), of the condensate obtained in the process of collapse of bubbles, occurring during the flow of component *comp* (*comp = atms, bubb*) into a pool of a Control Volume.

  $H_{gas,BC\text{-}JN,comp}$ enthalpy flux, (J/s), of the gas obtained in the process of collapse of bubbles, occurring during the flow of component *comp* (*comp = atms*, *bubb*) into a pool of a Control Volume.

  The meaning of the other symbols is the same as described above, in the discussion of the mass conservation equation. The values of enthalpy fluxes: $H_{liq,BC\text{-}JN,comp}$ $H_{gas,BC\text{-}JN,comp}$, are obtained directly from the bubble collapse model, present in the Heat Transfer Package (section 7.2.5).

- Mass transfer on Solid Heat Conductors (*m*=2)

  The equations determining energy sources for each of the four components that may reside in a Control Volume, are:

$$(W_2 h_2)_{atms} = \quad - \sum_{SC \in CV} A_{SC} X_{atms} m_{cond,SC} h_{vap}(T_{atms})$$

$$(W_2 h_2)_{dropor\ pool} = \sum_{\substack{SC \in atms \\ SC' \notin stack}} \left(\sum_{SC \in SC'} A_{SC} X_{atms} m_{cond,SC} h_{liq}(T_{cond}) \right)$$

$$(W_2 h_2)_{pool} = \sum_{SC \in pool} \left(- A_{SC} X_{pool} m_{boil,SC} h_{liq}(T_{cond}) + H_{liq,BC-SC} \right)$$

$$(W_2 h_2)_{bubb} = \sum_{SC \in pool} H_{gas,BC-SC}$$

$h_{vap}(T_{atms})$	specific enthalpy of steam in the atmosphere, (J/kg)
$h_{liq}(T_{cond})$	specific enthalpy of condensate, (J/kg), (the condensate is typically close to saturation and therefore h_{liq} is close to the saturated liquid enthalpy at total CV pressure. The condensate is however somewhat subcooled, the degree of subcooling depends on the wall temperature, as described in section 7.1.)
$h_{liq}(T_{pool})$	specific enthalpy of water in the pool, (J/kg)
$H_{liq,\ BC\text{-}SC}$	enthalpy flux of liquid created in the process of bubble collapse, (J/s)
$H_{gas,\ BC\text{-}SC}$	enthalpy flux of gas created in the process of bubble collapse, (J/s)

The above equations do not conserve energy. In case of condensation the difference between the energy gained by the liquid, and lost along with the disappearing steam is equal to: $-m_{cond}(h_{vap}(T_{atms})-h_{liq}(T_{cond}))$. A similar difference is observed in case of boiling. This fact does not cause any violation of the energy balance, since exactly the same energy is simultaneously deposited or removed from the wall of the Solid Heat Conductor. Therefore the overall energy balance, for both Control Volumes and Solid Heat Conductors, is preserved.

- Non-equilibrium mass transfer (m=3)

Before the equations defining enthalpy sources for nonequilibrium processes are shown, a short discussion is necessary to derive the final form of the equations, as used by the code.

Assume that nonequilibrium boiling (flashing) occurs in the pool of a Control Volume. The mass transfer rate is equal to: $(V\Gamma_{boil})_{pool}$ (kg/s). The enthalpy flux of the created steam bubbles is: $(V\Gamma_{boil})_{pool} \cdot h_{vap}$ (J/s), where h_{vap} is the saturated steam enthalpy. The enthalpy flux of the evaporated water is: $-(V\Gamma_{boil})_{pool} \cdot h_{liq}$, where h_{liq} is the saturated liquid enthalpy. The difference, equal to: $(V\Gamma_{boil})^{pool} \cdot (h_{vap}-h_{liq})$, is the energy consumed in the process. This energy is assumed to be taken from the water (thus causing the pool temperature to decrease). The energy effect of the flashing process may be therefore written as follows:

$$flashing: \begin{cases} pool: & -(V\Gamma_{boil})_{pool} h_{liq} - (V\Gamma_{boil})_{pool}(h_{vap} - h_{liq}) \\ bubb: & +(V\Gamma_{boil})_{pool} h_{vap} \end{cases}$$

The term defining the energy effect for the pool is simplified, and the final expression is:

$$flashing: \begin{cases} pool: & -(V\Gamma_{boil})_{pool} h_{vap} \\ bubb: & +(V\Gamma_{boil})_{pool} h_{vap} \end{cases}$$

Similar reasoning may be performed for nonequilibrium mass transfer in other components. In case of fogging (nonequilibrium condensation in atmosphere) the mass transfer rate is equal to: $(V\Gamma_{cond})_{atms}$. The enthalpy flux of the created condensate (fog) is: $(-V\Gamma_{cond})_{atms} \cdot h_{liq}$ (remember that positive Γ means evaporation). The enthalpy flux of the condensed steam is: $(V\Gamma_{cond})_{atms} \cdot h_{vap}$. The difference, $(-V\Gamma_{cond})_{atms} \cdot (h_{vap}-h_{liq})$, is the energy which in this case is "released". It is assumed that this energy is deposited in the atmosphere (thus raising the temperature of the atmosphere). The energy effect of the fogging process may be therefore written as follows:

$$fogging: \begin{cases} atms: & +(V\Gamma_{cond})_{atms} h_{vap} - (V\Gamma_{cond})_{atms}(h_{vap} - h_{liq}) \\ drop: & -(V\Gamma_{cond})_{atms} h_{liq} \end{cases}$$

Again, the term for the atmosphere energy is simplified to give:

$$fogging: \begin{cases} atms: & +(V\Gamma_{cond})_{atms} h_{liq} \\ drop: & -(V\Gamma_{cond})_{atms} h_{liq} \end{cases}$$

The same reasoning leads to similar terms for nonequilibrium mass transfer processes in droplets and bubbles. The final equations defining enthalpy sources due to nonequilibrium mass transfer are:

$$(W_3 h_3)_{atms} = +(V\Gamma_{cond})_{atms} h_{liq} + (V\Gamma_{boil})_{drop} h_{vap}$$
$$(W_3 h_3)_{drop} = -(V\Gamma_{cond})_{atms} h_{liq} - (V\Gamma_{boil})_{drop} h_{vap}$$
$$(W_3 h_3)_{pool} = -(V\Gamma_{boil})_{pool} h_{vap} - (V\Gamma_{boil})_{bubb} h_{liq}$$
$$(W_3 h_3)_{bubb} = +(V\Gamma_{boil})_{pool} h_{vap} + (V\Gamma_{boil})_{bubb} h_{liq}$$

As easily seen, the above equations conserve energy, since the sum of all terms is exactly equal to zero.

- Interphase mass transfer (m=4)

The equations defining enthalpy sources due to interphase mass transfer are:

$$(W_4 h_4)_{atms} = + A_{drop-atms} m_{in,drop-atms} h_{vap} + A_{pool-atms} m_{in,pool-atms} h_{vap}$$
$$(W_4 h_4)_{drop} = -A_{drop-atms} m_{in,drop-atms} h_{liq}$$
$$(W_4 h_4)_{pool} = - A_{bubb-pool} m_{in,bubb-pool} h_{liq} - A_{pool-atms} m_{in,pool-atms} h_{liq}$$
$$(W_4 h_4)_{bubb} = + A_{bubb-pool} m_{in,bubb-pool} h_{vap}$$

where h_{liq} and h_{vap} are the enthalpies of liquid and vapor, consumed or produced during the interphase mass transfer processes. The values of the enthalpies are calculated based on the following assumptions:

Pool-atmosphere mass transfer:
- Evaporation: the appearing vapor and the disappearing water both have the temperature of the pool surface
- Condensation: the appearing condensate and the disappearing steam both have the temperature of the atmosphere gas in the vicinity of the pool surface.

Atmosphere-droplet mass transfer:
- Evaporation: the appearing vapor and the disappearing water both have the temperature of the droplets
- Condensation: the appearing condensate and the disappearing steam both have the local atmosphere temperature at the elevation of the average droplet location.

Pool-bubble mass transfer:

- Evaporation: the appearing vapor and the disappearing water both have the local temperature of the pool, at the elevation of average bubble location
- Condensation: the appearing condensate and the disappearing steam both have the temperature of the bubbles.

The net energy effect of the interphase mass transfer processes is equal to the difference between the internal energy of vapor and water. This energy is consumed during evaporation, or released during condensation. In case of pool-atmosphere mass transfer the energy is assumed to be taken from (in case of evaporation) or deposited in (in case of condensation) the upper layer of the pool. In case of droplet-atmosphere transfer it is assumed to be taken from (or deposited in) the droplets. In case of bubble-pool transfer it is assumed to be taken from (or deposited in) the bubbles. The terms defining those energies are shown below, in the Interphase energy transfer (n=3).

- De-entrainment of droplets and bubbles (m=5)

The equations defining enthalpy sources for each of the four CV components due to the de-entrainment processes, are:

$$(W_5 h_5)_{atms} = A_{in,p-a}\alpha_{bubb}\rho_{bubb}(v_{v,bubb} - v_{in,p-a})h_{bubb}$$
$$(W_5 h_5)_{drop} = -A_{in,p-a}\alpha_{drop}\rho_{drop}(v_{v,bubb} - v_{in,p-a})h_{drop} - D_E A_{h,d-e}\overline{\alpha}_{drop}\rho_{drop}v_{h,drop}h_{drop}$$
$$(W_5 h_5)_{pool} = A_{in,p-a}\alpha_{drop}\rho_{drop}(v_{v,bubb} - v_{in,p-a})h_{drop} + D_E A_{h,d-e}\overline{\alpha}_{drop}\rho_{drop}v_{h,drop}h_{drop}$$
$$(W_5 h_5)_{bubb} = -A_{in,p-a}\alpha_{bubb}\rho_{bubb}(v_{v,bubb} - v_{in,p-a})h_{bubb}$$

where h_{drop} and h_{bubb} are the specific enthalpies of droplets and bubbles respectively. Other symbols are described above, in the discussion of mass conservation equation.

- Tabular mass sources (m=6)

The equations defining enthalpy sources for each of the four CV components due to mass sources, are:

$$(W_6 h_6)_{atms} = \sum_{MS\in atms} W_{MS}(1 - x_{liq,MS})h_{gas,MS}$$
$$(W_6 h_6)_{drop} = \sum_{MS\in atms} W_{MS}x_{liq,MS}h_{liq,MS}$$
$$(W_6 h_6)_{pool} = \sum_{MS\in pool} (W_{MS}x_{liq,MS}h_{liq,MS} + H_{liq,BC-MS})$$
$$(W_6 h_6)_{bubb} = \sum_{MS\in pool} H_{gas,BC-MS}$$

h_{gas} specific enthalpy of gas, (J/kg) (calculated based on user defined tables of source temperature, pressure, and gas composition)

h_{liq} specific enthalpy of liquid, (J/kg) (calculated based on user defined tables of source temperature and pressure)
$H_{liq,\ BC\text{-}MS}$ enthalpy flux of liquid created in the process of bubble collapse, (J/s)
$H_{gas,\ BC\text{-}MS}$ enthalpy flux of gas created in the process of bubble collapse, (J/s)

- Hydrogen burn (m=7)

  The hydrogen burn reactions are considered as mass sources and sinks for the atmosphere gases. The enthalpy source is:

$$(W_7 h_7)_{atms} = -\frac{dm_{H_2}}{dt} h_{H_2}(p,T) - \frac{dm_{O_2}}{dt} h_{O_2}(p,T) + \frac{dm_{H_2O}}{dt} h_{H_2O}(p,T)$$

  The mass sinks of hydrogen and oxygen, dm_{H2}/dt, dm_{O2}/dt, as well as the mass source of steam, dm_{H2O}/dt, are calculated by the hydrogen burn package, described in section 11.4. The enthalpies of hydrogen, oxygen, and steam are calculated for the current pressure and temperature, as shown in section 3.5.2

- Oxidation reactions (m=8)

  The oxidation reactions are considered as mass sources and sinks for the atmosphere gases. The enthalpy sources is:

$$(W_8 h_8)_{atms} = \sum_{SC \in CV} (W_{ox,2} h_{ox,2}(p,T)\delta_{k,2} - W_{ox,1} h_{ox,1}(p,T)\delta_{k,1}) \cdot A_{SC}$$

$W_{ox,1}$ mass transfer rate of the reacting gas per unit surface area, (kg/m^2-s)
$W_{ox,2}$ mass transfer rate of the reaction product gas per unit surface area, (kg/m^2-s)
A_{SC} surface area, (m^2), of the 1-D or a 2-D Solid Heat Conductor, where the reaction occurs; summation is performed over all surfaces present in given CV
δ_{kj} Kronecker delta, equal to 1 when gas j is equal to k, zero otherwise
$h_{ox,1}$ enthalpy of the reacting gas (J/kg)
$h_{ox,2}$ enthalpy of the reaction product gas (J/kg)

The oxidation term concludes the discussion of the terms defining enthalpy sources in the energy conservation equation. The next terms in the energy conservation equation, Q_n^{comp}, are due to the heat transfer processes. Eight such processes are distinguished in the model. These processes are discussed below.

- Convection from 1-D and 2-D Solid Heat Conductors (n=1)

  The energy from the surface of 1-D or 2-D Solid Heat Conductors by convection is deposited always in the continuous component (atmosphere and pool) of a Control Volume. If the SC surface is partly covered with water then there is a convective heat flux to both atmosphere and pool. For each of them a different heat transfer coefficient is calculated, by the appropriate heat transfer model.

The equations defining the convective energy transfer are:

$$Q_{1,atms} = \sum_{SC \in atms} A_{SC} X_{atms} \left\{ q_{SC,atms} - m_{cond,SC} \left[h_{vap}(T_{atms}) - h_{liq}(T_{cond}) \right] \right\}$$

$$Q_{1,drop} = 0.0$$

$$Q_{1,pool} = \sum_{SC \in pool} A_{SC} X_{pool} \left\{ q_{SC,pool} - m_{boil,SC} \left[h_{vap}(T_{atms}) - h_{liq}(T_{cond}) \right] \right\}$$

$$Q_{1,bubb} = 0.0$$

A_{SC} surface area of Solid Heat Conductor SC, (m^2)
X^{atms} fraction of SC above the pool of control volume, (-)
X^{pool} fraction of SC below the pool of control volume, (-)
$m_{cond,\,SC}$ condensation mass transfer, ($kg/m^2/s$)
$m_{boil,\,SC}$ boiling mass transfer, ($kg/m^2/s$)
$q_{SC,comp}$ wall heat flux from the surface of a Solid Heat Conduction No. *SC*, to the component No. *comp*, (*comp = atms, pool*), (W/m^2)

The values of $q_{SC,comp}$ are calculated by the 1-D and 2-D Solid Heat Conductor Packages (chapters 5, 6), using the correlations present in the Heat Transfer Package (section 7.1).

If there is no mass transfer (boiling or condensation) on the surface of a Solid Heat Conductor, then this term specifies the total wall heat flux from all Solid Heat Conductors. If there is a mass transfer process then only a part of the total wall heat flux is entering the continuous component, the rest of the heat is being used to support the mass transfer process. For example, suppose that a boiling occurs on a surface of a Solid Heat Conductor, with the mass flux of $m_{boil,\,SC}$=1.0 (kg/m^2-s). The evaporation enthalpy (the enthalpy of the produced steam bubbles minus the enthalpy of water) is equal to, say $2.0 \cdot 10^6$ (J/kg/s). The overall wall heat flux is equal to $q_{SC,pool}$=$2.1 \cdot 10^6$ (W/m^2). In this case the major part of the wall heat flux, namely $2.0 \cdot 10^6$ (W/m^2), is used to provide energy for the evaporation process, while the rest ($0.1 \cdot 10^6$ (W/m^2)) is entering the pool of a Control Volume, resulting in an increase of the water temperature.

The above formulation of the energy balance allows exact conservation of the overall energy in the system being considered. The energy is being transferred from wall to fluid, and from one CV component to another, but the total energy which leaves the surface of a Solid Heat Conductor is exactly the same as the total energy increase of all fluids in a Control Volume. The word "exactly" means here: within the accuracy of the round-off errors of the computer arithmetics which, for the double precision arithmetics applied throughout the code means a relative error of about 10^{-15}.

- Thermal Radiation (*n*=2)

The radiative heat exchange is calculated by the Thermal Radiation Package (chapter 8), using the grey enclosure model with or without a participating gas. The model includes radiation to a pool surface. When the model is activated, the pool surface is treated as one of the radiating surfaces. The model takes into account aerosols that may be present in the atmosphere.

The model calculates radiative heat fluxes for all surfaces, as well as the net energy absorbed by the participating gas. The contribution of the thermal radiation to the Control Volume energy balance equation is given by:

$$Q_{2,atms} = Q_{gas,TR}$$
$$Q_{2,drop} = 0.0$$
$$Q_{2,pool} = -Q_{pool,TR}$$
$$Q_{2,bubb} = 0.0$$

$Q_{gas,\,TR}$ net radiative heat absorbed by participating gas (and aerosols if present), (W)
$Q_{pool,\,TR}$ radiative heat flux from the surface associated with the pool surface (if present), (W)

The values of $Q_{gas,\,TR}$ and $Q_{pool,\,TR}$ are calculated by the Thermal Radiation Package, as described in chapter 8. The convention in the Thermal Radiation Package is that the heat flux is positive when energy is emitted from the surface, therefore the term for the pool energy source is taken with the negative sign.

- Interphase mass transfer (n=3)

  This term includes two parts:
  - the interphase heat transfer, and
  - the energy effect (evaporation enthalpy) of the interphase mass transfer.

  It should be noted that although the interphase heat and mass transfer processes are grouped here together, they are two different an independent processes and should not be mixed.

  The interphase heat transfer process is an example of a typical heat transfer process, driven by the temperature difference between two different phases. The value of heat transfer rate is determined by the temperature difference and the heat transfer coefficient. The heat transfer coefficient is calculated using correlations appropriate for planes (pool-atmosphere interphase), droplets (droplet-atmosphere), or bubbles in swarms (bubble-pool), as described in chapter 7.

  In contrast to the heat transfer process, the mass transfer process is driven by the difference in steam concentrations. The mass transfer rate is determined by calculating the appropriate mass transfer coefficient, using correlations valid for the process in question (chapter 7).

  The equations defining the interphase heat transfer are:

$$Q_{3,atms} = +A_{drop-atms}q_{in-HT,drop-atms} + A_{pool-atms}q_{in-HT,pool-atms}$$
$$Q_{3,drop} = -A_{drop-atms}q_{in-HT,drop-atms}$$
$$Q_{3,pool} = -A_{bubb-pool}q_{in-HT,bubb-pool} - A_{pool-atms}q_{in-HT,pool-atms}$$
$$Q_{3,bubb} = +A_{bubb-pool}q_{in-HT,bubb-pool}$$

where $q_{int,\ HT}$ are the interphase heat transfer rates, (W/m^2), which are calculated by the appropriate model from the Heat Transfer Package (sections 7.2.3, 7.2.4, 7.2.6).

The equations defining the energy effect of the interphase mass transfer processes are built based on the following assumptions:

- In case of pool-atmosphere mass transfer the evaporation/condensation energy is assumed to be taken from (in case of evaporation) or deposited in (in case of condensation) the upper layer of the pool.
- In case of droplet-atmosphere transfer the evaporation/condensation energy is assumed to be taken from (or deposited in) the droplets.
- In case of bubble-pool transfer the evaporation/condensation energy is assumed to be taken from (or deposited in) the bubbles.

As a consequence of those assumptions the energy effects of the interphase mass transfer processes are given by:

$$Q_{3,atms} = 0.0$$
$$Q_{3,drop} = -A_{drop-atms} q_{in-MT,drop-atms} (h_{vap} - h_{liq})$$
$$Q_{3,pool} = -A_{pool-atms} q_{in-MT,pool-atms} (h_{vap} - h_{liq})$$
$$Q_{3,bubb} = -A_{bubb-pool} q_{in-HT,bubb-pool} (h_{vap} - h_{liq})$$

where $q_{int,\ MT}$ are the enthalpy changes due to interphase mass transfer, (W/m^2), h_{liq} and h_{vap} are the enthalpies of liquid and vapor, consumed or produced during the interphase mass transfer processes. The way the values of those enthalpies are calculated is described above, below the equations determining the interphase mass transfer.

The overall term defining the interphase energy transfer, $Q_{3,comp}$, is a sum of the two terms shown above, and thus takes into account both the interphase heat transfer, and the energy effect of the interphase mass transfer.

$$Q_{3,comp} = Q_{3,HT} + Q_{3,MT}$$

- Heat source for pool due to bubble collapse (n=4)

The bubble collapse model is introduced in four different cases:

- In case of Junction flows, when an atmospheric gas flows into a pool of the receiving Control Volume.
- In case of Junction flows, when bubbles, being transported along with the pool, flow into a pool of the receiving Control Volume.
- In case of boiling on the surface of a Solid Heat Conductor.
- In case of tabular mass sources of gases located below the pool surface.

In each of those cases the initial stream of gas undergoing the collapse process, $W_{ini,\,BC}$, is divided into two streams: condensed water, $W_{liq,\,BC}$, and gas, $W_{gas,\,BC}$, (see bubble collapse model description, section 7.2.5). The initial enthalpy flux, $H_{ini,\,BC} = W_{ini} \cdot h_{ini}$, is replaced by the enthalpy fluxes of liquid, $H_{liq,\,BC} = W_{liq} \cdot h_{liq}$, and gas, $H_{gas,\,BC} = W_{gas} \cdot h_{gas}$. The difference between the initial enthalpy flux and the sum of the final enthalpy fluxes of liquid and gas, constitutes the heat released during the bubble collapse process. This heat is assumed to be deposited fully in the pool, at the location of the bubble source.

The heat sources from the four bubble collapse processes mentioned above, are grouped here together to form the heat source term No. 4 in the energy conservation equation. The equations defining energy sources due to the bubble collapse processes are:

$$\begin{aligned} Q_{4,pool} = & \sum_{JN \in CV} \left[W_{JN,atms} F_a h_{JN,atms} - \left(H_{liq,BC-JN,atms} + H_{gas,BC-JN,atms} \right) \right] + \\ & + \sum_{JN \in CV} \left[W_{JN,bubb} F_a h_{JN,bubb} - \left(H_{liq,BC-JN,bubb} + H_{gas,BC-JN,bubb} \right) \right] + \\ & + \sum_{SC \in CV} \left[A_{SC} X_{pool} m_{boil,SC} h_{vap} - \left(H_{liq,BC-SC} + H_{gas,BC-SC} \right) \right] + \\ & + \sum_{MS \in CV} \left[W_{MS} (1 - x_{liq,MS}) h_{liq,MS} - \left(H_{liq,BC-MS} + H_{gas,BC-MS} \right) \right] \end{aligned}$$

$$Q_{4,atms} = 0.0 \qquad Q_{4,drop} = 0.0 \qquad Q_{4,bubb} = 0.0$$

The above term closes the energy balance for terms involving bubble collapse, and causes the energy to be conserved within the accuracy of round-off errors.

- Tabular energy sources (n=5)

Several energy sources may be present in a Control Volume. The energy source data consists of a table, defining energy source rate versus time, as well as the position of the source relative to the bottom of a Control Volume. The table may define the energy source as a simple function of time, or (through Control Functions) as a more complicated function, depending on any calculated variables.

The energy from the source is deposited always in a continuous component of a Control Volume. The location of the energy source, relative to the actual water level in a Control Volume, determines whether the source power is deposited in the atmosphere or the pool of a Control Volume.

The equations defining the distribution of heat from tabular energy sources, are:

$$Q_{5,atms} = \sum_{ES \in atms} Q_{ES} \qquad Q_{5,drop} = 0.0$$

$$Q_{5,pool} = \sum_{ES \in pool} Q_{ES} \qquad Q_{5,bubb} = 0.0$$

where Q_{ES} is the energy transfer rate, from the tabular energy source No. ES, (W).

- Hydrogen burn (*n*=6)

  Hydrogen burn gives a heat source to the CV atmosphere. Therefore:

$$Q_{5,atms} = Q_{burn}$$
$$Q_{5,drop} = 0.0$$
$$Q_{5,pool} = 0.0$$
$$Q_{5,bubb} = 0.0$$

  The heat of burn is equal to:

$$Q_{burn} = \frac{dm_{H_2}}{dt} \cdot Q_{H_2} = \frac{dm_{H_2}}{dt} \cdot 1.206 \times 10^8$$

  Q_{H2} is the heat of burn, equal to 120.6 (MJ/kg_{H2}). dm_{H2}/dt is the burn rate (kg/s), calculated by the Hydrogen Burn Package, described in section 11.4.

- Heat source due to pumps, compressors, or turbines (*n*=7)

  Pumps, compressors, and turbines provide an energy source (sink) for the control volumes downstream the junction in which such machine is installed. If the flow through the junction reverses, then a different Control Volume receives the power.

  The energy source or sink is given by:

$$Q_{6,atms} = \sum_{JN \in CV} Q_{hydr,atms,JN}$$
$$Q_{6,drop} = 0.0$$
$$Q_{6,pool} = \sum_{JN \in CV} Q_{hydr,pool,JN}$$
$$Q_{6,bubb} = 0.0$$

  The pump or turbine power, $Q_{hydr,atms,JN}$ and $Q_{hydr,pool,JN}$ is calculated as described in sections 4.6.2.5 and 4.6.3.3.

- Decay heat from radioactive isotopes (*n*=8)

  The calculation of decay heat from radioactive isotopes is described in detail in section 12.3.12 and therefore is not discussed here.

2.5 Average Velocities in a Control Volume

Velocities of all components that may reside in a Control Volume are needed to calculate heat transfer to Solid Heat Conductors as well as at the interphase boundaries, and to track the movement of dispersed components (droplets and bubbles).

Several different models to calculate CV average velocities are applied in the existing codes. The model used in RELAP5 [44], defines CV velocities through the following equation:

$$J_{CV,\varphi} = \alpha_{CV,\varphi} v_{CV,\varphi} A_{CV} = \frac{1}{2}\left(\sum_{JN\ to\ CV} J_{JN,\varphi} + \sum_{JN\ from CV} J_{JN,\varphi} \right) \qquad \text{(RELAP5)}$$

$J_{CV,\varphi}$ average volumetric flow, (m^3/s), of phase φ, in a Control Volume *CV*
$\alpha_{CV,\varphi}$ area fraction for the flow of phase φ, (-)
$v_{CV,\varphi}$ average velocity, (m/s), of phase φ, in a Control Volume *CV*, (m^2)
A_{CV} area associated with Control Volume *CV*, (m^2)
$J_{JN,\varphi}$ volumetric flow, (m^3/s), of phase φ, in junction *JN*
φ phase identifier (liquid or gas)

The sums in the above equation are over all junctions that connect *to* or *from* the Control Volume *CV*.

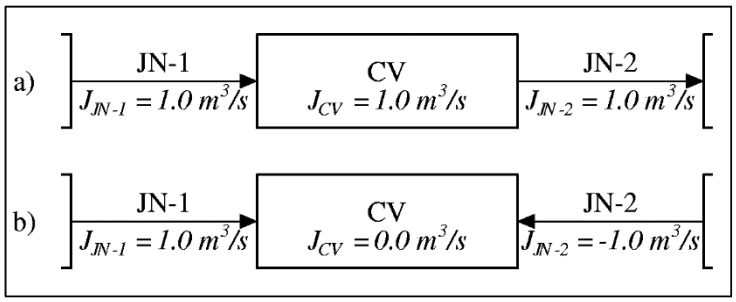


Figure 2-7 CV average velocities in RELAP formulation.

This approach has been criticized by the MELCOR developers, by pointing out that the above formulation depends on the logical direction of junctions [46]. For example, reversing both the sign of a junction velocity and the associated direction of positive flow (as shown in Figure 2-7, JN-2), does not preserve the Control Volume flow. While in case a) the CV flow is 1 m^3/s, case b), which physically is no different from case a), will give the CV flow equal to zero.

This dependence of results on the choice of the positive junction flow direction is a consequence of the assumption taken in RELAP5, that all *to* connections are on the left of a Control Volume, and all the *from* connections are on the right.

The formulation, which is applied in the MELCOR code, is [46]:

$$J_{CV,\varphi} = \alpha_{CV,\varphi} v_{CV,\varphi} A_{CV} = \frac{1}{2} \sum_{JN\ in\ CV} \left| J_{JN,\varphi} \right| \qquad \text{(MELCOR)}$$

where the summation is performed over all junctions connected *to* or *from* the Control Volume *CV*. As may easily be checked, the above formulation is independent on the positive flow direction assumed for junctions.

The approach taken is SPECTRA is somewhat similar to the one in MELCOR, it has however been extended in order to model adequately the transport of dispersed components (droplets or bubbles). To model the transport of dispersed components in a Control Volume it is necessary to distinguish the vertical and horizontal velocities. As will be shown below, the movement of dispersed components (droplets, bubbles) is tracked using the drift flux model, which gives different equations in case of vertical and horizontal flow. Therefore two velocities, horizontal and vertical, are used for each component present in a Control Volume.

The calculation of the Control Volume average velocities is described in sections 2.5.1 and 2.5.2 for the continuous and the dispersed components respectively. In the following discussion the continuous components are marked by the superscript *Ccomp*, while the dispersed components are marked with the symbol *Dcomp*.

2.5.1 Continuous Components

The total volumetric flows of the continuous components are calculated in a similar way as in the MELCOR code:

$$J_{Ccomp} = \frac{1}{2} \sum_{m=1}^{N} \left| J_{m,Ccomp} \right|$$

J_{Ccomp} total volumetric flow, (m^3/s), of the continuous component (*Ccomp = atms, pool*),

$J_{m,Ccomp}$ volumetric flows, (m^3/s), of the continuous component (*Ccomp = atms, pool*), associated with all mass sources relevant for given component). The flows, $J_{m,Ccomp}$, are defined as positive when the flow is into the control volume, and negative when the flow is out of the control volume, independently on the choice of positive junction flow (this additional assumption is not needed here, but it is needed for vertical flow calculation, as shown below).

The total volumetric flow, J_{Ccomp}, is split into two parts: the vertical flow, $J_{v,Ccomp}$, and the horizontal flow, $J_{h,Ccomp}$. The vertical flow is calculated from:

$$J_{v,Ccomp} = \sum_{m=1}^{N} J_{m,Ccomp}\, z_m$$

$J_{v,Ccomp}$ vertical volumetric flow, (m^3/s), of the continuous component,
z_m dimensionless distance, characteristic for the mass source *m*, (-), defined as:

$$z_m = \frac{Z_{Ccomp}/2 - Z_m}{Z_{Ccomp}}$$

Z_{Ccomp} height of the continuous component *Ccomp*, (m),
Z_m distance between the mass source *m*, and the reference point, (m). The reference points are the bottom of CV in case of pool and top of CV in case of atmosphere.

From the above definition of z_m it follows that $-½ \leq z_m \leq +½$, and that the dimensionless distance is positive if the center point of the continuous component is closer to the pool-atmosphere interphase boundary than the mass source (sources high in atmosphere or low in pool), and negative otherwise.

When both total and vertical flows are known, then the horizontal flow is obtained from the relation:

$$J_{Ccomp} = \sqrt{(J_{h,Ccomp})^2 - (J_{v,Ccomp})^2}$$

The equation for vertical flow calculation is illustrated by an example consisting of one CV with two connecting junctions, JN-1, JN-2. The example is shown in Figure 2-8.

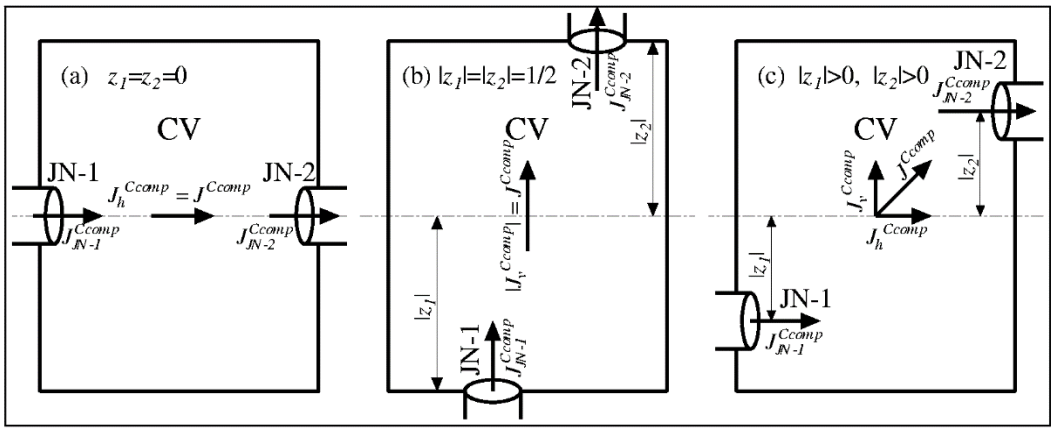


Figure 2-8 Vertical and horizontal velocities in the continuous components.

If both junctions are connected exactly in the middle of the component *Ccomp* then the values of z_m, are equal to zero for both junctions, and the vertical flow is equal to zero, $J_{v,Ccomp} = 0.0$ (Figure 2-8.a). In this case the horizontal flow is equal to the total flow, $J_{h,Ccomp} = J_{Ccomp}$.

If junctions are located at the bottom and at the top of CV then the absolute values of both z_1 and z_2 are equal to ½. In this case the equation for vertical flow will give the same absolute value of flow as the equation for the total flow. In such case the horizontal flow is equal to zero, as shown in Figure 2-8.b. The sign of vertical flow depends on the component being considered. In case of atmosphere, z_1 is equal to –½, while z_2 is equal to +½, and the vertical flow is equal to:

$$J_{v,Ccomp} = \left(-\frac{1}{2}\right)\left(+\left|J_{JN-1}\right|\right) + \left(+\frac{1}{2}\right)\left(-\left|J_{JN-2}\right|\right) = -J_{Ccomp}$$

in agreement with the convention that defines downwards flow of atmosphere as positive. If on the other hand the pool is considered, then according to the convention mentioned above, the signs of z_1, z_2 reverse, and the vertical flow is equal to:

$$J_{v,Ccomp} = \left(+\frac{1}{2}\right)\left(+\left|J_{JN-1}\right|\right) + \left(-\frac{1}{2}\right)\left(-\left|J_{JN-2}\right|\right) = +J_{Ccomp}$$

again in agreement with the convention that defines upwards flow of pool as positive.

In the case shown in Figure 2-8.c both junctions have intermediate locations. There are nonzero flows in both vertical and horizontal direction. Note that as a result of the definition of the dimensionless distance, the following relation is always fulfilled:

$$|z| \le \frac{1}{2}$$

which, in turn, leads to the following relation between the vertical and the total flows:

$$\left|J_{v,Ccomp}\right| \le J_{Ccomp}$$

It should be noted that although in the above example only junction flows were considered, the code takes into account all mass transfer processes.

The elevations of all those sources are defined through the system geometry entered in input data, in particular the elevations of all mass sources are as follows:

- Junction flows (m=1)

  The source elevation is assumed to be equal to the center elevation of this part of the junction, which is above water level, in case of atmosphere flow calculation, and the center elevation of this part of the junction, which is below water level, in case of pool flow calculation.

- Condensation or boiling on Solid Heat Conductors (m=2)

  The source elevation is assumed to be equal to the center elevation of this part of the Solid Heat Conductor, which is above water level, in case of atmosphere flow calculation, and the center elevation of this part of the Solid Heat Conductor, which is below water level, in case of pool flow calculation.

- Non-equilibrium mass transfer (m=3)

  In case of atmosphere the non-equilibrium mass transfer is assumed to occur in the whole atmosphere volume, therefore the source location is in the center of the CV atmosphere, thus: $Z_{3,atms} = ½ \cdot Z_{atms}$, and the source relative elevation, $z_{3,atms}$, is always equal to zero.

  In case of pool the non-equilibrium mass transfer is assumed to occur in the whole pool volume only in case of shallow pools, with the depths no greater than Z_{max}. The source elevation is therefore defined as follows:
  For shallow pools ($Z_{pool} \leq Z_{max}$): $Z_{3,pool} = ½ \cdot Z_{pool}$, which means that the relative elevation, $z_{3,pool}$, is equal to zero.

  For deep pools ($Z^{pool} > Z_{max}$): $Z_3^{pool} = ½ \cdot (Z_{pool} - Z_{max})$, which means that the relative elevation is determined as: $z_{3,pool} = (½ \cdot Z_{pool} - ½ \cdot (Z_{pool} - Z_{max})/Z_{pool} = ½ \cdot Z_{max}/Z_{pool}$. The value of maximum depth at which non-equilibrium bubble source is located, ($½ \cdot Z_{max}$), is by default set to 1.0 m, and may be changed by the user through input data (see Volume 2).

- Interphase mass transfer (m=4)

  In case of pool-atmosphere interphase the source location is at the pool surface elevation. That means the relative elevation, $z_{4,Ccomp}$, is equal to –½ in case of both atmosphere and pool.

  In case of droplet-atmosphere interphase, the mass source for the atmosphere flow calculation is assumed to be located at the current average droplet position.

  In case of bubble-pool interphase, the mass source for the pool flow calculation is assumed to be located at the current average bubble position.

- De-entrainment (m=5)

  The source of gas for the atmosphere flow calculation (in case of bubble de-entrainment), as well the source of water for the pool flow calculation (in case of droplet vertical de-entrainment), are located at the pool surface. Therefore the relative elevation, $z_{5,comp}$, is always equal to –½.

- Tabular mass sources (m=6)

  The relative elevations of all mass sources are calculated based on the source positions, defined in the input data.

- Hydrogen burn (m=7)

  The mass sources associated with hydrogen burn are neglected, and thus the relative elevations are set to zero.

- Oxidation reactions (m=8)

  The mass sources associated with the oxidation reactions are neglected, and thus the relative elevations are set to zero.

When the vertical and horizontal flows, J_v^{Ccomp}, J_h^{Ccomp}, are known, the corresponding volumetric fluxes and velocities may be calculated. The volumetric flux is defined as volumetric flow per unit flow area. (Volumetric flux is sometimes referred to in literature as "superficial velocity". It is felt that the term volumetric flux is better, since it gives the physical sense of the parameter.) The volumetric fluxes are equal to:

$$j_{v,Ccomp} = \frac{J_{v,Ccomp}}{A_v}$$

$$j_{h,Ccomp} = \frac{J_{h,Ccomp}}{A_{h,comp}}$$

$j_{v,Ccomp}$ vertical volumetric flux of the continuous component *Ccomp*, ((m^3/s)/m^2) =(m/s)
$j_{h,Ccomp}$ horizontal volumetric flux of the continuous component *Ccomp*, (m/s)
A_v representative area for flow in vertical direction, (m^2), (the value is specified in input)
$A_{h,Ccomp}$ representative area for the horizontal flow of the continuous component *Ccomp*, (m^2). The total area for flow in horizontal direction, A_h, which is specified in input, is divided into two areas, used for the atmosphere and the pool, based on the current pool level in CV: $A_{h,atms} = A_h \cdot Z_{atms}/(Z_{atms}+Z_{pool})$, and $A_{h,pool} = A_h \cdot Z_{pool}/(Z_{atms}+Z_{pool})$.

Finally the velocities of the continuous components are calculated based on known volumetric fluxes in the vertical and horizontal direction, as follows:

$$v_{v,Ccomp} = \frac{j_{v,Ccomp}}{1-\alpha_{Dcomp}} = \frac{J_{v,Ccomp}}{A_v(1-\alpha_{Dcomp})}$$

$$v_{h,Dcomp} = \frac{j_{h,Ccomp}}{1-\alpha_{Dcomp}} = \frac{J_{h,Ccomp}}{A_h(1-\alpha_{Dcomp})}$$

where: α_{Dcomp} is the volumetric fraction of dispersed component, (-), (bubbles, droplets).

2.5.2 Dispersed Components

The calculation of discontinuous (dispersed) component velocities is based on the drift flux model ([48], chapter 4) and the assumption that the vertical and horizontal velocities can be treated independently, using the one-dimensional drift flux equations.

The discussion of the discontinuous components velocity calculation starts with the derivation of the general form of the drift flux equation. Below that discussion, the equations used for the vertical and the horizontal flows are shown.

The starting point for the derivation of the dispersed components velocities is the equation defining drift flux between the discontinuous component (D) and the continuous component (C). The drift flux, j_{DC}, is equal to ([48], section 4.3):

$$j_{DC} = v_\infty \alpha_D (1-\alpha_D)^x$$

α_D volumetric fraction of the dispersed component, (-), (bubbles, droplets)
v_∞ Terminal velocity, (m/s), (velocity of a single particle in an infinite, motionless continuous component)
x exponent, depending on the flow regime, (-)

The velocity of the dispersed component, v_D, is related to the drift flux by the following equation ([48], section 4.4):

$$v_D = C_0 j + \frac{j_{DC}}{\alpha_D}$$

j total volumetric flux, (m/s)
j_{DC} drift flux, (m/s)
C_0 model constant, (-), depending on the flow regime

To rearrange the above equation, the following basic relations (see [48], section 1.4) are used:

$$j = j_C + j_D$$
$$j_C = v_C (1-\alpha_D)$$
$$j_D = v_D \alpha_D$$

In the above equations j_C and j_D are the volumetric fluxes, (m/s), of the continuous and the discontinuous component respectively, v_C and v_D are the velocities, (m/s), of the continuous and the discontinuous component respectively.

Using the above relations and the drift flux equation the equation for the dispersed component velocity may be rearranged to give:

$$v_D = \frac{C_0 v_C (1-\alpha_D) + v_\infty (1-\alpha_D)^x}{1 - C_0\ \alpha_D}$$

Since the velocities of the continuous components, v_C, are known, then the velocities of the dispersed components may be calculated if the values of the terminal velocity, v_∞, the constant C_0, and the exponent x_v, are known. The same general form of drift flux equation is applied for both vertical and horizontal flow directions. The equations are shown below.

- *Vertical velocity*

  The equation defining vertical velocity of the discontinuous components is written as follows:

$$v_v^{Dcomp} = \frac{C_{0,v}^{Dcomp} v_v^{Ccomp} (1-\alpha^{Dcomp}) + v_{\infty,v}^{Dcomp} (1-\alpha^{Dcomp})^{x_v^{Dcomp}}}{1 - C_{0,v}^{Dcomp} \alpha^{Dcomp}}$$

  where the subscripts *v* are used to signify vertical direction. The indicators Dcomp Ccomp were placed in superscripts rather than subscripts to preserve space. The values of $v_{\infty,v}^{Dcomp}$, $C_{0,v}^{Dcomp}$, x_v^{Dcomp} depend on the flow regime, and are calculated differently for the bubbly and drop flows. The values may also be defined for each Control Volume through input data.

- *Horizontal velocity*

  The equation defining horizontal velocity of the discontinuous components is written as follows:

$$v_h^{Dcomp} = \frac{C_{0,h}^{Dcomp} v_h^{Ccomp} (1-\alpha^{Dcomp}) + v_{\infty,h}^{Dcomp} (1-\alpha^{Dcomp})^{x_h^{Dcomp}}}{1 - C_{0,h}^{Dcomp} \alpha^{Dcomp}}$$

  where the subscripts *h* mean horizontal direction. For the horizontal flow the values: $v_{\infty,h}^{Dcomp}$, $C_{0,h}^{Dcomp}$ are assumed to be equal to zero and one respectively, but may be changed for each Control Volume by input. With the default values, the above equation reduces to: $v_h^{Dcomp} = v_h^{Ccomp}$.

  The fact that the drift flux model parameters may be defined on input, allows the user a flexible modeling of the dispersed component flow. If, for example, a spray system is modeled, for which an average horizontal velocity of droplets is known, then the following definition:

$$v_{\infty,h}^{drop} = v_{hor}$$
$$C_{0,v}^{drop} = 10^{-10}$$
$$x_h^{drop} = 10^{-10}$$

2.6 Treatment of Dispersed Components

Two methods are available in SPECTRA for dispersed components (bubbles and droplets): a simplified treatment and a detailed treatment.

- Simplified treatment

  Note that the simplified treatment is by definition used in the homogeneous Control Volumes (see section 2.8). With the simplified treatment of the dispersed components (droplets, bubbles), the code assumes constant particle diameters and average particle positions. This is the default option, giving fast but less accurate results for bubbles/droplets. The bubble diameter is calculated using flow regime maps, similar to those used by RELAP. Two regimes: bubbly flow and slug flow are distinguished. Different correlations are used for the bubbly and the slug flow regimes. The flow regime boundaries are determined as follows:

$$bubbly: \quad \alpha < \alpha_{BS} - \Delta\alpha/2$$
$$slug: \quad \alpha > \alpha_{BS} + \Delta\alpha/2$$

  Here α_{BS} is the transition void fraction and $\Delta\alpha$ is the interpolation zone (default 0.05). The transition void fraction depends on mass flux:

  - Low flow, $G < G_{1,BS}$ (default $G_{1,BS}$ =2000 (kg/m^2-s))

$$\alpha_{BS} = \alpha_{1,BS}$$

    $\alpha_{1,BS}$ is the transition void fraction for low flow (default value 0.25)

  - High flow, $G < G_{2,BS}$ (default $G_{2,BS}$ =3000 (kg/m^2-s))

$$\alpha_{BS} = \alpha_{2,BS}$$

    $\alpha_{2,BS}$ is the transition void fraction for high flow (default value 0.5)

  - Transition flow, $G_{1,BS} < G < G_{2,BS}$

$$\alpha_{BS} = \alpha_{1,BS} + (\alpha_{2,BS} - \alpha_{1,BS}) \frac{G - G_{1,BS}}{G_{2,BS} - G_{1,BS}}$$

  In case of bubbly flow the bubble diameter is obtained using the Taylor instability model (see 2.6.1). In case of slug flow the bubble diameter is obtained from Ishii Mishima correlation [124]:

$$D_{bubb} = 0.88 \cdot D_{hyd}$$

  D_{hyd} hydraulic diameter

- Detailed treatment
  With this approach for each dispersed component two additional equations are used:

  - The particle count equation, used to determine the size of the average particle
  - The particle position equation, which is a sort of a volumetric balance of sources and sinks along the vertical axis, and is used to calculate the average position of the dispersed components.

  The two additional equations required for the discontinuous component: particle count and particle average position, are described in sections 2.6.1 and 2.6.2 respectively.

2.6.1 Particle Count

If the detailed treatment of dispersed components is used, then to determine the representative (average) size of the dispersed particles the particle count equation is used. To write the particle count for a given component in a Control Volume, one needs to know the number of particles per second produced from each source of the dispersed component. Each source has its own characteristic size of the particles that are created by it. The models used to determine particle sizes from different sources are described below in this section. With the average size of the particle and the mass flux of particles known for the given source, the number of particles per second for the source is calculated.

With the number of particles per second known from each source one can "count" particles in the Control Volume. Finally, knowing the total number of particles and the total mass of particles (from the mass balance equation) the average particle size is calculated.

The particle count equations have the following form:

$$\frac{dN_{Dcomp}}{dt} = \sum_{m=1}^{8} S_{m,Dcomp}$$

N_{Dcomp} total number of particles (in code a real number), of the dispersed component *Dcomp* (*Dcomp = drop, bubb*)

$S_{m,Dcomp}$ number of particles per second, produced by the source *m*, (s^{-1}) (source strength)

The summation is made over all relevant sources for a given discontinuous component (see section 2.4). The calculation of the source strength is discussed below for each of the eight mass source types. With a known total number of particles, the current volume of a single particle is obtained as:

$$V_{0,Dcomp} = \frac{M_{Dcomp}}{N_{Dcomp}\rho_{Dcomp}}$$

where M_{Dcomp} is the total mass (kg) of the discontinuous component *Dcomp*, ρ_{Dcomp} is the density (kg/m^3) of the discontinuous component *Dcomp*. With $V_{0,Dcomp}$ known, the single particle diameter is assumed based on the assumption that the particles form regular spheres:

$$D_{0,Dcomp} = \sqrt[3]{6V_{0,Dcomp}/\pi}$$

The source strength, $S_{m,Dcomp}$, for each particular source is obtained from the general formula:

$$S_{m,Dcomp} = \frac{W_{m,Dcomp}}{\rho_{m,Dcomp}V_{0,m,Dcomp}} = \frac{6W_{m,Dcomp}}{\pi\rho_{m,Dcomp}\left(D_{0,m,Dcomp}\right)^3}$$

where $\rho_{m,Dcomp}$ is the density (kg/m^3) of particles created at the source *m*, while $V_{0,m,Dcomp}$ is the volume (m^3) of a single particle created at the mass source *m*. The densities of particles are easily obtained from the fluid property routines, when the source parameters (pressure, temperature, composition) are known. The calculation of the single particle volume is somewhat more complicated, and is discussed below for each of the mass sources.

- Junction flows (*m*=1), and tabular mass sources (*m*=6)

  The two terms: junction flows and tabular mass sources, are treated in very similar manner, since they both represent similar physical phenomenon; fluid flow from some other part of the system into a given Control Volume. One important difference is that in the first case the mass transfer rates are determined by the junction flows, calculated by the CV Junction Package, while in the second case they are determined by the user, via tabular or control functions. There is another difference. Junctions transport four components (atmosphere, droplets, pool, bubbles). Therefore a distinction must be made between for example bubbles created by atmosphere flow into a pool, and bubbles being simply transported from the pool of one Control Volume to another. The tabular mass sources supply only two components: gas or liquid (or both), which are considered to be continuous components prior to injection, and which, upon injection, are distributed among CV components depending on the source position, as shown in Figure 2-6. Therefore the masses from tabular sources are treated in the same way as the junction sources from the continuous components.

  Transport of discontinuous components through junctions
  This case includes droplets being transported through junctions along with atmosphere flow, and bubbles being transported along with the pool flow. The particles which appear in the given CV have the same dimension as the particles in the source CV (upstream the junction). Thus:

$$D_{0,m=1,Dcomp} = D_{0,Source,Dcomp}$$

  Droplets created due to junction flow of pool to the atmosphere of a Control Volume, and tabular sources of liquid located in the atmosphere of a Control Volume
  A set of correlations is available to determine the source of droplets created when water is injected into the atmosphere of a Control Volume. Those correlations rely on fluid properties as well as geometrical parameters, like flow area and diameter. The parameters are supplied with the input data for all junctions, as well as the tabular mass sources.

The correlation set consists of:

- *Small flows*
  The Kutateladze and Styrikovich [161] formula is applicable for small flow ([48], equation 12.1):

  K-S: $$D_{0,m=1,drop} = 2.0\left(\frac{\sigma D/2}{g(\rho_{liq}-\rho_{gas})}\right)^{1/3}$$

  g gravity acceleration, (m/s^2)
  σ surface tension, (N/m)
  ρ_{liq} liquid density, (kg/m^3)
  ρ_{gas} gas density, (kg/m^3)
  D source diameter, (m)

  The droplet diameter is limited by the wavelength of the Taylor instability ([48], section 12.5). Therefore the maximum droplet diameter is set by the Taylor instability model [162]:

  Taylor: $$D_{0,m=1,drop} = 2.0\left(\frac{\sigma}{g(\rho_{liq}-\rho_{gas})}\right)^{1/2}$$

- *Jet flows*
  When the liquid velocity through the orifice is increased the critical velocity for transition to jetting is soon exceeded, and the liquid leaves the orifice in a form of jet, which, as shown by Rayleigh [163], is always unstable and breaks into individual droplets. The liquid velocity required for the formation of the liquid jet, is calculated based on the Kutateladze and Styrikovich equation [161] (reproduced from [48], equation 9.8) with the roles of the gas and the liquid reversed:

  $$v_{jet,drop} = 1.25\left(\frac{\sigma}{g(\rho_{liq}-\rho_{gas})(D/2)^2}\right)^{1/2}\left(\frac{\sigma g(\rho_{liq}-\rho_{gas})}{\rho_{liq}^2}\right)^{1/4}$$

  Droplets which are produced in this way have a diameter which, according to Wallis, is equal to 1.9 times the diameter of the orifice ([48], section 12.2):

  Wallis: $$D_{0,m=1,drop} = 1.9D$$

- *Atomization*
  If the jet velocity is very large the jet becomes violently unstable and breaks into a shower of very small droplets. This regime is called atomization ([48], section 12.2). An important parameter for determining the stability of a single droplet is the Weber number, We, based on the relative velocity and the gas density ([48], equation 12.4):

$$We = \frac{\rho_{gas}(v_{liq} - v_{gas})^2 D_{0,drop}}{\sigma}$$

v_{liq} liquid velocity, (m/s)
v_{gas} gas velocity, (m/s)

The critical value of the Weber number is 12 ([48], section 12.3). Thus the drop diameter during atomization is calculated from:

$$D_{0,drop} = 12 \frac{\sigma}{\rho_{gas} v_{liq}^2}$$

In the above equation the gas velocity is neglected since in typical conditions when the We number is used the gas velocity is expected to be small, compared to the velocity of the liquid jet.

Bubbles created due to junction flow of atmosphere to the pool of a Control Volume, and tabular sources of gas located in the pool of a Control Volume
A set of correlations is available to determine the source of bubbles created when atmosphere is injected into the pool of a Control Volume. The correlation set consists of:

- *Small flows*
  Kutateladze and Styrikovich [161] formula is applicable for small flow ([48], equation 12.1):

  K-S: $$D_{0,m=1,bubb} = 2.0 \left(\frac{\sigma D / 2}{g(\rho_{liq} - \rho_{gas})} \right)^{1/3}$$

  g gravity acceleration, (m/s^2)
  σ surface tension, (N/m)
  ρ_{liq} liquid density, (kg/m^3)
  ρ_{gas} gas density, (kg/m^3)
  D source diameter, (m)

- *Large flows*
  In case of large flows the bubble diameter is determined by the volumetric flow rate. The Davidson equation [175] is used to calculate the bubble volume, V_b, (reproduced from [48], equation 9.5):

  Davidson: $$V_{0,m=1,bubb} = 1.138 \frac{Q_{gas}^{6/5}}{g^{3/5}}$$

  Q_{gas} volumetric flow, (m^3/s)
  g gravity acceleration, (m/s^2)

- *Jet flows*
  When the gas velocity through the orifice is made sufficiently high the gas leaves the orifice in a form of jet, which eventually breaks into individual bubbles. The gas velocity required for the formation of the gas jet is calculated by two equations. In case of small diameters the Kutateladze and Styrikovich equation [161] is used (reproduced from [48], equation 9.8):

Jet:
$$v_{jet,bubb} = 1.25 \cdot \left(\frac{\sigma}{g(\rho_{liq} - \rho_{gas})(D/2)^2} \right)^{1/2} \cdot \left(\frac{\sigma\, g\, (\rho_{liq} - \rho_{gas})}{\rho_{gas}^{\,2}} \right)^{1/4}$$

For large orifice diameters the Bugg and Rowe equation [176] is used:

$$Fr = \frac{\pi}{4} \frac{v_{jet,bubb}}{\sqrt{g / D}} = 50$$

where Fr is the orifice Froude number, (-), and g is the gravity acceleration, (m/s^2). Bubbles which are produced in this way have a diameter which, according to Wallis, is about twice the diameter of the orifice ([48], section 9.2):

Wallis:
$$D_{0,m=1,bubb} = 2.0 \cdot D$$

- Condensation or boiling on Solid Heat Conductors (m=2)

  The size of droplets, which may be created at the lower edge of a Solid Heat Conductor in case of condensation, is calculated using the Taylor instability model. Therefore:

$$D_{0,m=2,bubb} = 2.0 \left(\frac{\sigma}{g(\rho_{liq} - \rho_{gas})} \right)^{1/2}$$

  In case of nucleate boiling the equation proposed by Fritz and Ende [177] is used:

$$D_{0,m=2,bubb} = 0.0208 \cdot \theta \cdot \left(\frac{\sigma}{g(\rho_{liq} - \rho_{gas})} \right)^{1/2}$$

  where θ is the contact angle which, in general, depends on both fluid properties and surface conditions. The contact angle is an input parameter, with the default value of 96°. Because of difficulties in finding the values of the contact angle, the default value was selected to give the same bubble size as the Taylor instability model. Therefore θ = (2.0/0.0208) ≈ 96°.

  In case of film boiling, the Taylor instability model is used to determine the initial bubble size).

- Nonequilibrium mass transfer (*m*=3)
  In case of nonequilibrium boiling (flashing) the diameter of the created bubbles is calculated using the Taylor instability model. In case of nonequilibrium condensation (fogging) the diameter of the created droplets is set to a constant value. The value is $5 \cdot 10^{-4}$ m. Typical droplets of fog are of course smaller than that, however very small droplets may cause numerical convergence problems, which will slow down the calculations. Therefore relatively large fog particles are assumed for general application, with the possibility of re-definition of the value on input, if the users wishes to make a study the influence of fog particle size on the results.

- Interphase mass transfer (*m*=4)
  Since the interphase mass transfer consists of evaporation or condensation at the surface dividing phases, there is no particle source. The source strength for this term is always zero.

- De-entrainment (*m*=5)
  De-entrainment processes are the processes of removal of the existing discontinuous components from the continuous components. Therefore the particle sizes, associated with this term, are simply equal to the current sizes of the particles present in the Control Volume:

$$D_{0,m=5,Dcomp} = D_{0,Dcomp}$$

2.6.2 Particle Position

If the detailed treatment of dispersed components is used, then the following procedure is used to determine the representative (average) position of the dispersed particles. The average position of particle is defined as the position for which the total volumes of particles above and below are equal. Consider a volume V_{Dcomp} of dispersed particles, moving with vertical velocity $v_{v,Dcomp}$, in an environment with volumetric sources and sinks. The vertical movement of the average particle position is in this case given by:

$$\frac{dZ_{Dcomp}}{dt} = v_{v,Dcomp} + \sum_{m=1}^{8} \frac{J_{m,Dcomp}}{V_{Dcomp}} (Z_m - Z_{Dcomp})$$

Z_{Dcomp} elevation of the average position of the dispersed component, (m). The reference (zero) elevation is assumed to be the bottom of Control Volume for bubbles and the top of Control Volume for droplets. The positive direction for Z^{Dcomp} is the same as the positive direction for vertical velocity: upwards for bubbles and downwards for droplets.

$J_{m,Dcomp}$ volumetric flow, (m^3/s), of the discontinuous component, associated with mass source *m* (negative in case of mass sink).

Z_m position of the center point of mass source *m*, (m). The reference elevation and positive direction for Z_m are the same as for Z_{Dcomp}.

V_{Dcomp} total volume of the dispersed component *Dcomp*, (m^3)

The summation is made over all relevant sources for a given discontinuous component (see section 2.4).

The derivation of the above equation is quite simple. If one considers a coordinate system associated with the current average particle position, and moving with the vertical velocity of $v_{v,Dcomp}$, then the equation will take the form: $dZ_{Dcomp}/dt=\Sigma(J_{m,Dcomp}/V_{Dcomp})\cdot Z_m$. Replacing the derivative by the finite difference this may be written as: $\Delta Z_{Dcomp}=\Sigma(J_{m,Dcomp}\cdot\Delta t/V_{Dcomp})\cdot Z_m$, or finally:

$$\Delta Z_{Dcomp} = \sum_{m=1}^{8} Z_m (\Delta V_{m,Dcomp} / V_{Dcomp})$$

where $\Delta V_{m,Dcomp}=(J_m^{Dcomp}\cdot\Delta t)$ is the volume added at source m, during the time step.

According to the definition of average, the right hand size of the last equation represents the average value of Z, weighted by volumetric strengths of all sources, and thus represents the middle point of the total volume of the component *Dcomp* or, in other words, the average location of the component *Dcomp*.

The location of all sources, Z_m, has to be known to evaluate the particle position equation. Assumptions taken in defining the source elevations for all sources have already been discussed in section 2.5.1, at the discussion of the continuous components vertical velocity calculation. Note that the equation used for vertical velocity calculation uses the dimensionless elevations, z_m, while the particle position equation uses the dimensional elevations, Z_m (m).

Results obtained with the dispersed particle modelling are illustrated by two simple examples, shown below.

The first example consists of one Control Volume at the atmospheric pressure, filled up to 5 m with pool at 315 K. Dry nitrogen bubbles are injected with the rate of 0.1 kg/s, at the elevation of 2 m (using tabular mass sources), during the period 0 - 150 s. Figure 2-9 shows the calculated water level and the average bubble position for the times t<200 s. Initially the injected bubbles are present near the source and pool swelling is observed. At about 30 s first bubbles reach the pool surface. From this moment a slow decrease in pool level is observed, since bubbles are disappearing through the pool surface. Finally a stable situation is reached, when the amount of injected bubbles is the same as the amount of bubbles flowing through the surface. The average bubble position stabilizes approximately in the middle between the source location and the pool surface. When the source is stopped at 150 s, the bubbles remaining in the pool rise up, and finally disappear about 10 s later.

It may easily be checked that this time is consistent with the calculated bubble velocity, which is about 0.3 m/s. At 150 s the average bubble is about 1.5 m below the pool surface, which means that the lowest bubble is about 3 m below the pool surface. The transport time is therefore (3 m)/(0.3 m/s) = 10 s, and indeed the last bubbles are expected to disappear from the pool at about 160 s - Figure 2-9.

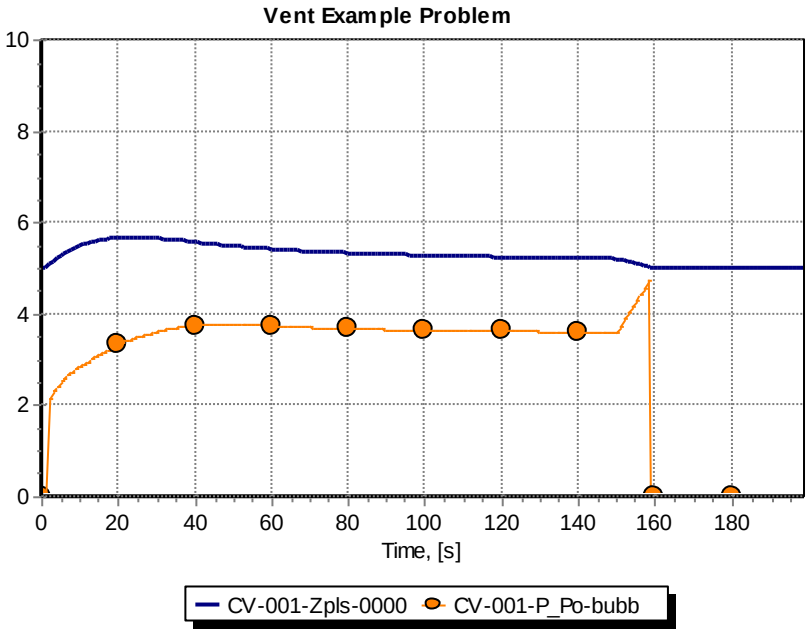


Figure 2-9 Results of the dry vent example problem

In the second example problem the pool level is 4 m, the pool is nearly saturated, and a few steam/hydrogen bubbles are injected at the bottom of the pool at the time t=0.0 s. The bubble diameter is plotted in Figure 2-10. Instead of plotting the bubble diameter as a function of time, it is plotted as a function of bubble average position: $D_{bubb}(t) = D_{bubb}(Z_{bubb}(t))$. It is seen that near the pool surface the bubbles rapidly grow, which is a consequence of intensive evaporation into the bubbles, as the bubble pressure decreases along with the decreasing submergence. The results of this example problem are verified by comparison with the results of analytical bubble thermodynamics model, presented in [159]. The results obtained by Powers [159] are shown in Figure 2-11. It is seen that the results are in good agreement.

The particle position equation requires the knowledge of particle vertical velocity. As shown in section 2.5.2 the particle velocities are calculated using the drift flux model, which uses three parameters: terminal velocity, v_∞, the constant C_0, and the exponent x. Those parameters are calculated using correlations appropriate for droplets and bubbles. The correlations used for droplets and bubbles are described in the next two sections.

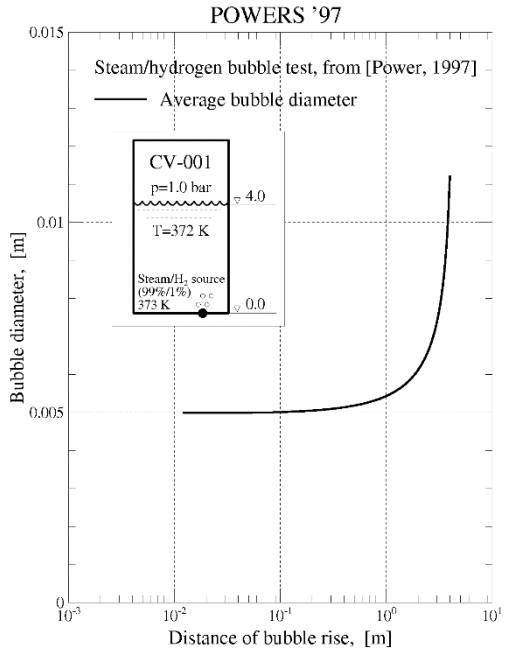


Figure 2-10 Bubble diameter, SPECTRA results

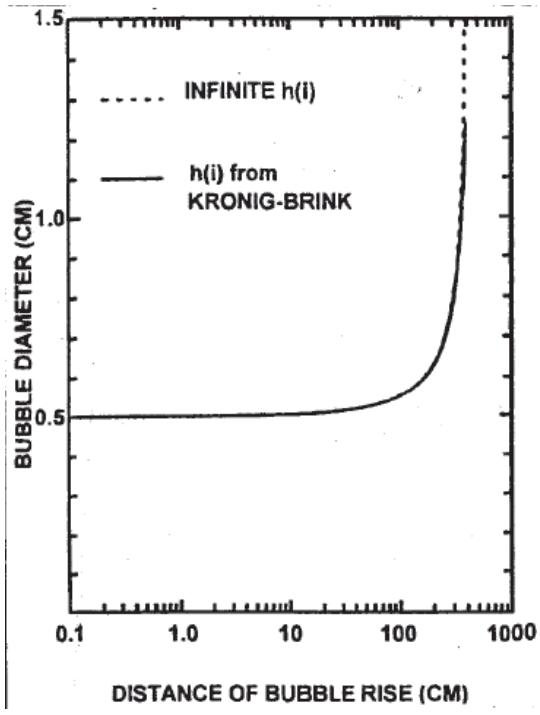


Figure 2-11 Bubble diameter [159]

2.6.3 Droplet Velocities

This section gives a description of the parameters needed for the drift flux equation in case of droplet flows. Two sets of parameters are described. First the vertical flow parameters: the droplet terminal velocity, $v_{\infty,v,drop}$, the constant $C_{0,v,drop}$, and the exponent $x_{v,drop}$, are described. Next the horizontal flow parameters: $v_{\infty,h,drop}$, $C_{0,h,drop}$, and $x_{h,drop}$, are described. (NOTE: the drift flux model parameters C_0 and x are described in section 2.5.2.)

- *Region 1, Small droplets*

  The Hadamard [164] and Rybczynski [165] correlation is used (reproduced from [48], equation 12.28). Compared to the correlation for bubbles the roles of liquid and gas are reversed:

$$\text{(H-R):}\quad v_{\infty,v,drop} = \frac{1}{18}\cdot\frac{D_d^2\, g(\rho_{liq}-\rho_{gas})}{\eta_{gas}}\cdot\frac{3\eta_{liq}+3\eta_{gas}}{3\eta_{liq}+2\eta_{gas}}$$

  where g is the gravity acceleration (m/s^2), ρ is density (kg/m^3), η is viscosity, (kg/m/s), and D_d is the droplet diameter. The appropriate value of the exponent for this region is $x_{v,drop}$ = 2.0 ([48], section 12.6). The value of constant $C_{0,v,drop}$ is equal to 1.0 (see section 2.5.2 for definition of C_0 and x).

- *Region 2, Surface tension dominant*

  In this region the equation recommended by Wallis is used (see [48], sections 12.5 and 8.2):

$$\text{(Wallis):}\qquad v_{\infty,v,drop} = \left(\frac{4}{3}\frac{D_d\, g(\rho_{liq}-\rho_{gas})}{C_D\rho_{gas}}\right)^{0.5}$$

  with the value of the drag coefficient, C_D, equal to 0.44 for Re > ~1000. For Reynolds numbers below ~1000 the drag coefficient is given by Schiller et al. [166]: $C_D = (24/Re)(1+0.15\cdot Re^{0.687})$. With this definition of C_D the calculation of the droplet velocity must be performed iteratively, since the velocity depends on the drag coefficient, C_D, which itself is a function of velocity (through the Reynolds number). To avoid iteration the following approximations for C_D have been derived:

$$C_D = \begin{cases} \dfrac{27.0}{\mathrm{Re}^{0.8}} & \text{if} \quad \mathrm{Re} < 30 \\ \dfrac{6.9}{\mathrm{Re}^{0.4}} & \text{if} \quad 30 < \mathrm{Re} < 1000 \\ 0.44 & \text{if} \quad 1000 < \mathrm{Re} \end{cases}$$

  The approximation formulae for Re < 1000 give good agreement with the formula recommended by Wallis - see Figure 2-12.

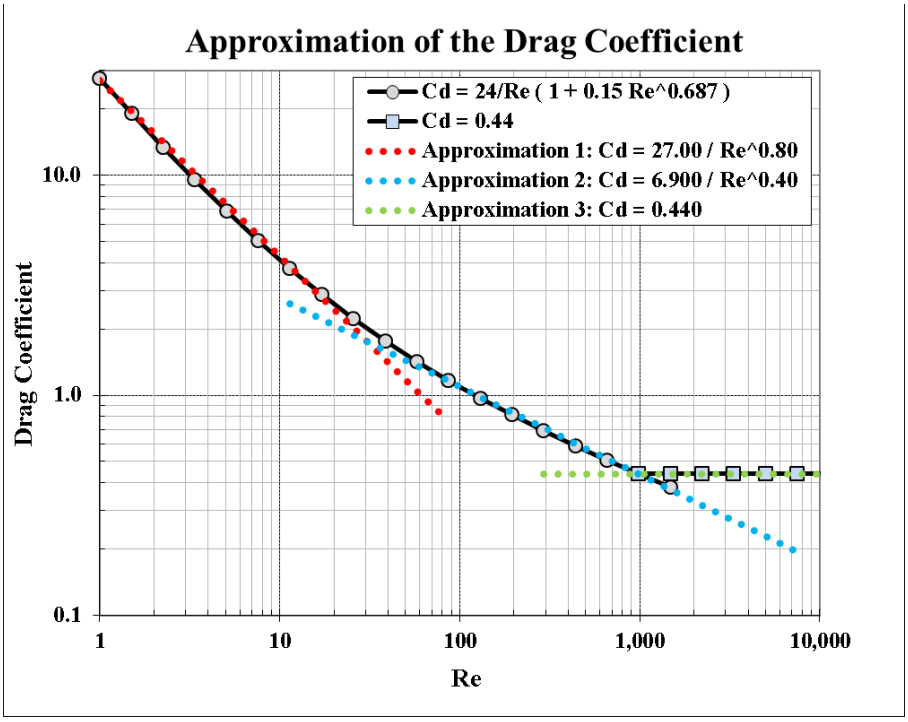


Figure 2-12 Approximations of drag coefficient for low (Re<30) and high (Re>30) Re numbers

The above approximations allow to obtain an explicit expression for the droplet velocity, by substituting $Re = v_\infty D_d \rho_{gas} / \eta_{gas}$:

$$v_{\infty,v,drop} = \begin{cases} \left(\dfrac{0.021728}{C_{D,\,\mathrm{Re}\to\infty}} \cdot \dfrac{D_d^{1.8} g(\rho_{liq} - \rho_{gas})}{\rho_{gas}^{0.2} \eta_{gas}^{0.8}} \right)^{0.833} & if \quad \mathrm{Re} < 30 \\ \left(\dfrac{0.085024}{C_{D\,\mathrm{Re}\to\infty}} \cdot \dfrac{D_d^{1.4} g(\rho_{liq} - \rho_{gas})}{\rho_{gas}^{0.6} \eta_{gas}^{0.4}} \right)^{0.625} & if \quad 30 < \mathrm{Re} < 1000 \\ \left(\dfrac{1.33333}{C_{D\,\mathrm{Re}\to\infty}} \cdot \dfrac{D_d g(\rho_{liq} - \rho_{gas})}{\rho_{gas}} \right)^{0.5} & if \quad \mathrm{Re} > 1000 \end{cases}$$

Here $C_{D,Re\to\infty}$ is the value appropriate for $Re \to \infty$, namely $C_{D,Re\to\infty} = 0.44$. The drag coefficient may be defined by the user (input parameter CDRGCV in the record 161000 - Volume 2). The numerical coefficients are equal to:

- first formula: $4/(3\times27.0) \times 0.44$
- second formula: $4/(3\times6.9) \times 0.44$
- third formula: $4/3$

The boundary values of Reynolds numbers only approximate numbers. Selection of correlation is performed by choosing the one that gives maximum value of the drag coefficient (i.e. minimum velocity). This provides smooth transition from one region to another.

The appropriate value of the exponent for this region is $x_{v,drop}$ = 2.0 ([48], section 12.6). The value of constant $C_{0,v,drop}$ is equal to 1.0 (see section 2.5.2 for definition of C_0 and x).

- *Region 3, Large distorted droplets*

  In this region the Levich equation ([167], page 431) is used (reproduced from [48], equation 12.32):

$$\text{(Levich):} \qquad v_{\infty,v,drop} = 1.4 \cdot \left(\frac{\sigma g(\rho_{liq} - \rho_{gas})}{\rho_{gas}^2} \right)^{1/4}$$

  The appropriate value of the exponent for this region is x_v^{drop} = 2.0 ([48], section 12.6). The value of constant $C_{0,v,drop}$ is equal to 1.0 (see section 2.5.2 for definition of C_0 and x).

Selection of correlation

Among the three correlations, shown above, this correlation is selected which gives the lowest value of the droplet terminal velocity. This method provides a smooth trasition from one correlation to another (continuity of the velocity function). The final droplet terminal velocity is plotted in Figure 2-13.

For horizontal velocity calculations the following values of the parameters $v_{\infty,h,drop}$, $C_{0,h,drop}$, and $x_{h,drop}$, are assumed:

$$v_{\infty,h,drop} = 0.0$$
$$C_{0,h,drop} = 1.0$$
$$x_{h,drop} = 1.0$$

With these values the drift flux equation reduces to (see section 2.5.2):

$$v_{h,drop} = v_{h,atms}$$

That means droplets are moving in the horizontal direction with the same velocity as atmosphere. Typically droplets have a relatively high inertia and they are expected to have a relatively small horizontal velocity. On the other hand in cases like for example a spray system injecting droplets at some angle, the horizontal velocity may be relatively large. It is not possible to take into account all situations. Therefore the above settings are considered as a good first approximation, while simultaneously a user is encouraged to change the parameters to values more appropriate for his particular problem.

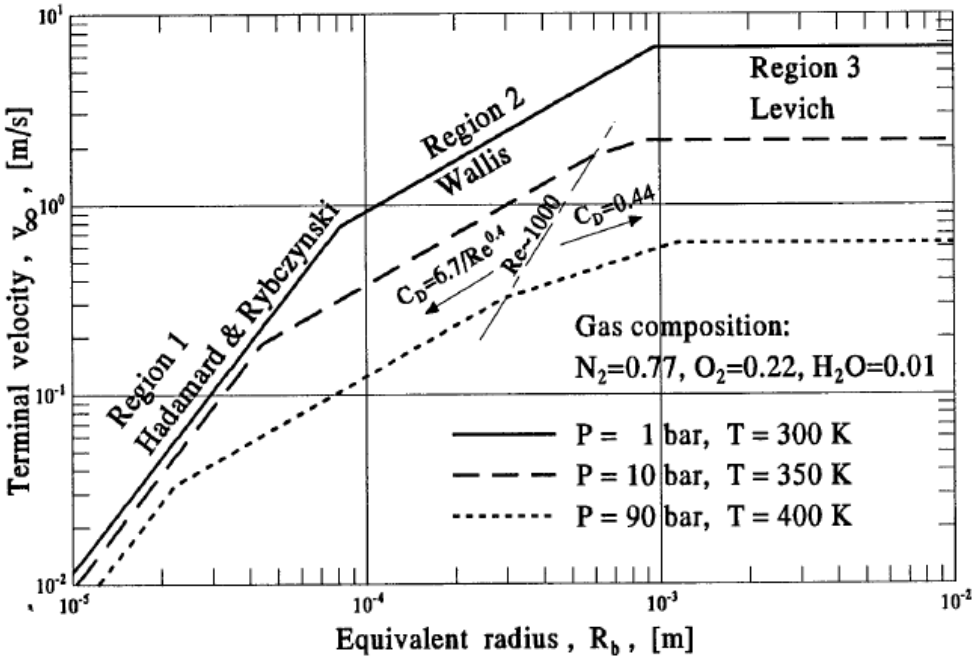


Figure 2-13 Droplet terminal velocity

2.6.4 Bubble Velocities

This section gives a description of the parameters needed for the drift flux equation in case of bubble flows. Two sets of parameters are described. First the vertical flow parameters: the bubble terminal velocity, $v_{\infty,v,bubb}$, the constant $C_{0,v,bubb}$, and the exponent $x_{v,bubb}$, are described. Next the horizontal flow parameters: $v_{\infty,h,bubb}$, $C_{0,h,bubb}$, and $x_{h,bubb}$, are described.

- *Region 1, Small bubbles*

  The Hadamard [164] and Rybczynski [165] correlation is used (reproduced from [48], equation 9.14):

$$\text{(H-R):}\quad v_{\infty,v,bubb} = \frac{1}{18}\cdot\frac{D_b^2 g(\rho_{liq}-\rho_{gas})}{\eta_{liq}}\cdot\frac{3\eta_{gas}+3\eta_{liq}}{3\eta_{gas}+2\eta_{liq}}$$

  where g is the gravity acceleration (m/s^2), ρ is density (kg/m^3), η is viscosity, (kg/m/s), and D_b is the bubble diameter (m).

  The appropriate value of the exponent for this region is $x_{v,bubb}$ = 2.0 ([48], section 9.3). The value of the constant $C_{0,v,bubb}$ is discussed below.

- *Region 2, Intermediate-small bubbles*

  In this region the Peebles and Gerber equation [168] is used (reproduced from [48], table 9.1):

$$\text{(P-G):}\quad v_{\infty,v,bubb} = 0.33 g^{0.76}\left(\frac{\rho_{liq}}{\eta_{liq}}\right)^{0.52}\cdot\left(\frac{D_b}{2}\right)^{1.28}$$

Although the above equation was written originally for CGS units, it is also correct in SI units. The appropriate value of the exponent for this region is $x_v^{bubb} = 1.75$ ([48], section 9.3).

- *Region 3, Intermediate bubbles*

  In this region the Peebles and Gerber equation [168] is used (reproduced from [48], table 9.1):

$$\text{(P-G):} \quad v_{\infty,v,bubb} = 1.35 \cdot \left(\frac{\sigma}{\rho_{liq} D_b / 2} \right)^{0.5}$$

  The value of the exponent for this region has been set to 1.6, an intermediate value between the Region 2 and Region 4 values: $x_{v,bubb} = 1.75$ and 1.50 respectively.

- *Region 4, Intermediate-large bubbles*

  In this region the Zuber equation is used (reproduced from [48], equation 9.34):

$$\text{(Zuber):} \qquad v_{\infty,v,bubb} = 1.53 \cdot \left(\frac{\sigma g (\rho_{liq} - \rho_{gas})}{\rho_{liq}^2} \right)^{1/4}$$

  The appropriate value of the exponent for this region is $x_{v,bubb} = 1.5$ ([48], section 9.3).

- *Region 5, Large bubbles*

  In this region the Davies and Taylor equation [169] is used (reproduced from [48], equation 9.19):

$$\text{(Davies):} \qquad v_{\infty,v,bubb} = 1.00 \cdot \sqrt{g D_b / 2}$$

  The value of exponent for this region has been set to 1.5, the same as in region 4.

  In case of large bubbles the influence of the containing walls is taken into account. The equation of Collins [170] is used, which is valid for Region 5. The equation gives the ratio of real terminal velocity to the terminal velocity in an infinite medium, $v_{\infty,v,bubb}(D \to \infty)$, and is reproduced from [48], section 9.3):

$$\frac{v_{\infty,v,bubb}}{v_{\infty,v,bubb}(D \to \infty)} = \begin{cases} 1.0 & \text{if} \quad D_b / D < 0.125 \\ 1.13 \cdot \exp(-D_b / D) & \text{if} \quad 0.125 < D_b / D < 0.6 \\ 0.496 \cdot D / D_b & \text{if} \quad 0.6 < D_b / D \end{cases}$$

  where D is the diameter of the container, (m), which in the code is an input parameter, with the default value equal to square root of the horizontal cross section area.

Selection of correlation

The following method is used to select the correlation:

- In the large diameter regions: Region 3, 4 and 5, the correlation is selected that gives the largest bubble velocity.

- Out of the small diameter regions: Region 1, and 2, and the large diameter regions, the correlation is selected that gives the smaller bubble velocity.

The selection scheme may be written as:

$$v_{\infty,v,bubb} = Min[Reg.1, Reg.2, Max(Reg.3, Reg.4, Reg.5)]$$

With this selection scheme the application ranges of correlations are in agreement with the ranges calculated as shown in [48], table 9.1. The final bubble terminal velocity is plotted in Figure 2-14.

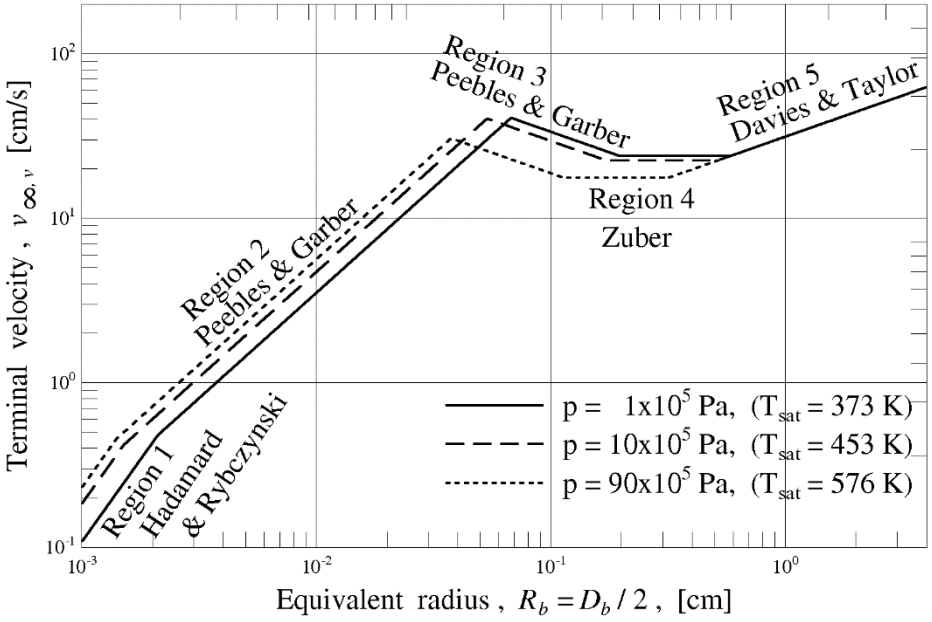


Figure 2-14 Bubble terminal velocity

For the calculation of the distribution parameter, C_0, several correlations are available. Two correlations are implemented in the code.

- *Zuber and Findlay* correlation [171]. The distribution parameter, C_0, is equal to:

$$C_{0,v,bubb} = 1.2$$

- *Sun et al.* correlation [172]. The distribution parameter, C_0, is equal to:

$$C_{0,v,bubb} = \frac{1}{0.82 + 0.12\dfrac{p}{p_{crit}}}$$

p pressure, (Pa)
p_{crit} critical pressure, (Pa)

For calculations one of the two correlations must be selected. The selection of correlation is based on an input parameter. The default selection is Zuber and Findlay.

For the horizontal velocity calculations the following values of the parameters $v_{\infty,h,bubb}$, $C_{0,h,bubb}$, and $x_{h,bubb}$, are assumed:

$$v_{\infty,h,bubb} = 0.0$$
$$C_{0,h,bubb} = C_{0,v,bubb}$$
$$x_{h,bubb} = 1.0$$

With these values the drift flux equation reduces to (see section 2.5.2):

$$v_{h,bubb} = C_{0,h,bubb} v_{h,pool}$$

In the earlier SPECTRA version the default setting was $C_{0,v,bubb} = 1.0$, which means that bubbles are moving in the horizontal direction with the same velocity as pool. Several test calculations and comparisons with RELAP showed that use of $C_{0,v,bubb} > 1.0$ gives better results, therefore this is the current default setting.

2.6.5 Alternative Correlation for Terminal Velocity of Particles

An alternative correlation for the terminal velocity of particles may be requested by the user (input parameter MODVI - Volume 2). The model, if activated, is applied for the following particles:

- bubbles,
- droplets,
- particles suspended in the liquid pool.

The applied convention concerning positive direction of particle velocity in a CV is:

- bubbles: up (towards the pool surface or junction)
- droplets: down (towards the pool surface or junction)
- particles: up (towards the pool surface or junction) or down (towards junction or a solid structure - sedimentation), depending on the density of particles (DENART - Volume 2) compared to the density of the liquid.

The model is based on the drag coefficient. The terminal velocity is obtained from:

$$v_\infty = \left(\frac{4/3}{C_D} \cdot \frac{D_p g(\rho_f - \rho_p)}{\rho_f} \right)^{0.5}$$

Here g is the gravitational acceleration (m/s^2), ρ is density (kg/m^3), η is viscosity, (kg/m/s), D is diameter. Subscripts p refers to particle, while f to fluid. Five regimes are distinguished (in contrast to the default droplet model - section 2.6.3, which has three regimes). The correlations for drag coefficient are based on Morsi et al. [204]. The correlations from [204] are not used directly; instead the following approximation, leading to an explicit formula for terminal velocity (section 2.6.3), is used:

$$C_D = \frac{A}{\mathrm{Re}^b}$$

Where $Re = v_\infty D_p \rho_f / \eta_f$. When C_D is substituted by this formula, the terminal velocity is obtained explicitly as:

$$v_\infty = \left(\frac{4/3}{A} \cdot \frac{D_d^{1+b} g(\rho_f - \rho_p)}{\rho_p^{1-b} \eta_f^b} \right)^{\frac{2}{2-b}}$$

The coefficients that were used in different regimes are shown in Table 2-2. In the first two ranges the same approximations as used in section 2.6.3, are used:

- Range 1: $C_D = 27 / Re^{0.8}$ $Re < 30$
- Range 2: $C_D = 6.9 / Re^{0.4}$ $30 < Re < 1000$

The approximations 1 and 2 are compared to the function from Schiller et al. [166] in Figure 2-15. In this range Morsi et al. [204] are using three correlations: Re < 10, 10 < Re < 100, and 100 < Re < 1000. Comparison of Schiller and Morsi functions for 1 < Re < 1000 is shown in Figure 2-19. For 1000 < Re < 5000 Morsi used: $C_D = 0.357 + 148.62/Re – 475000/Re^2$, which is also shown in Figure 2-15.

In the next three ranges the following approximations are used:

- Range 3: $C_D = 0.575 / Re^{0.05}$ $1{,}000 < Re < 10{,}000$
- Range 4: $C_D = 0.235 \cdot Re^{0.06}$ $10{,}000 < Re < 100{,}000$
- Range 5: $C_D = 0.470$ $100{,}000 < Re$

Comparison of the developed approximations with the functions from Morsi et al. [204] is shown in Figure 2-16 (the asymptotic value, for Re→∞, was shifted from 0.52 to 0.47, to be in agreement with the generally recommended value for a sphere - Figure 12-5). The approximations developed for SPECTRA agree very well with the functions from Morsi et al. [204].

The boundary values of Reynolds numbers given above are only approximate numbers. Selection of correlation is performed by choosing the minimum of 4 and 5 and then the maximum of 1, 2, 3, and the above mentioned minimum: Max (1, 2, 3, Min (4, 5)). This provides smooth transition from one region to another.

The value of C_D is plotted in Figure 2-18 for the full range of Reynolds numbers. For comparison, Figure 2-17 shows C_D for the default correlation (section 2.6.3).

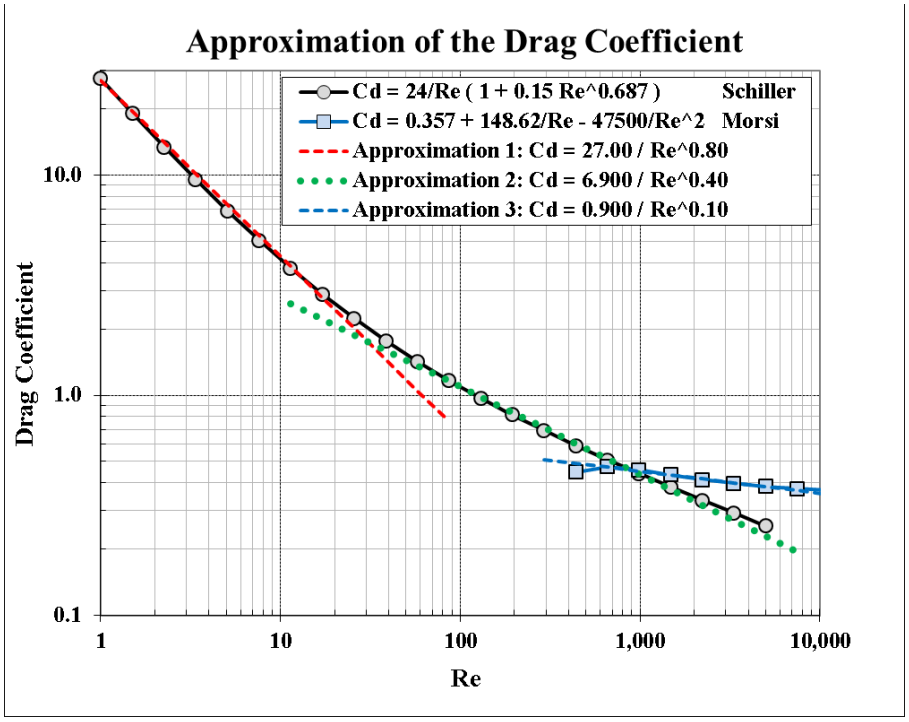


Figure 2-15 Terminal velocity correlation, approximations 1, 2, and 3

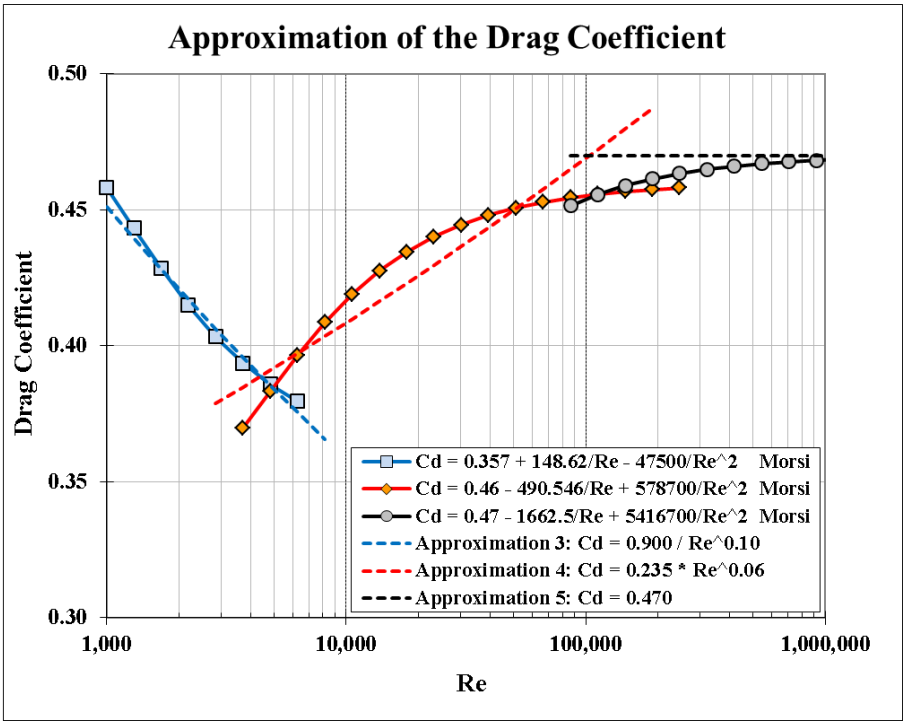


Figure 2-16 Terminal velocity correlation, approximations 3, 4, and 5

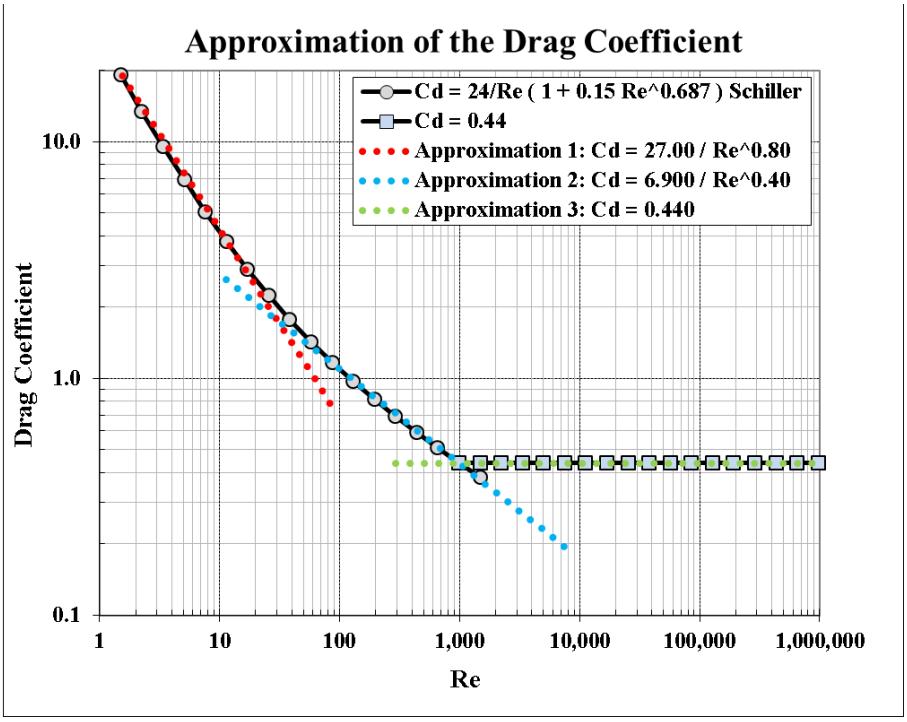


Figure 2-17 Default correlation for droplets, 3 ranges of Re, section 2.6.3

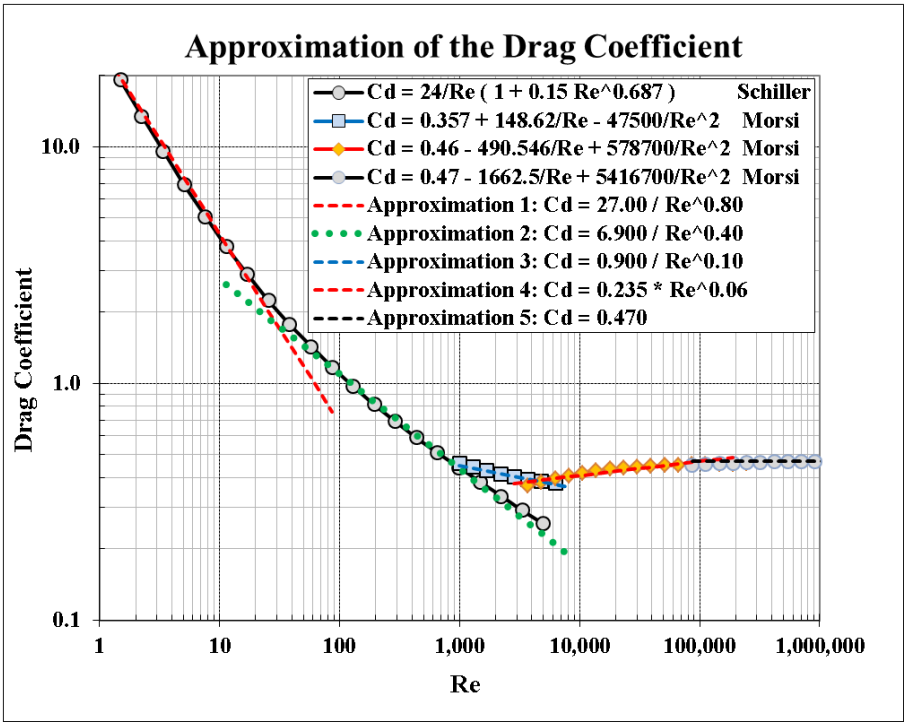


Figure 2-18 Alternative correlation (droplets, bubbles, particles), 5 ranges of Re

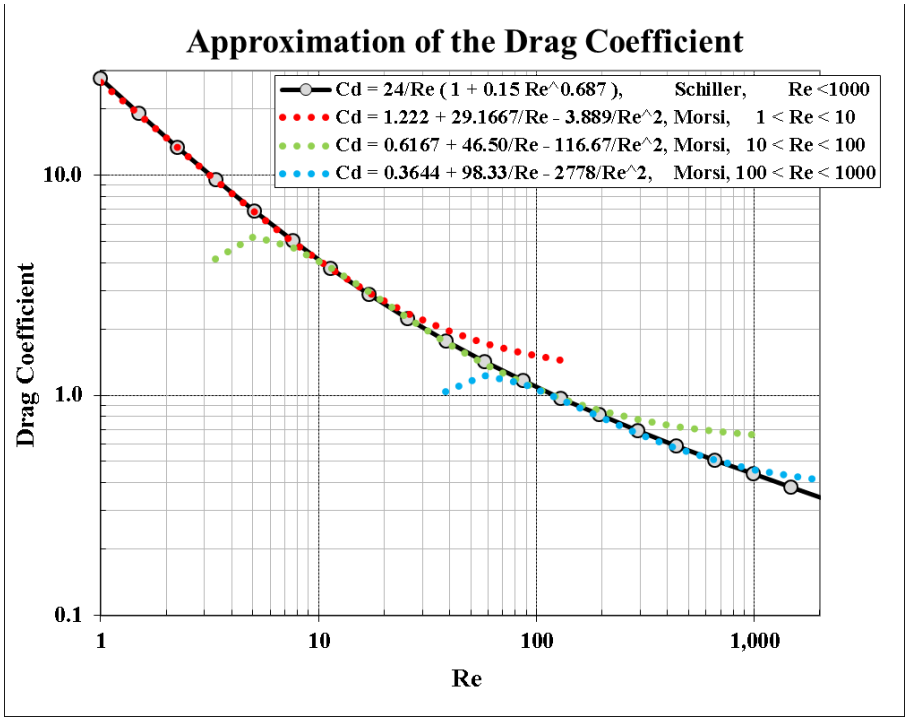


Figure 2-19 Comparison of Schiller et al. [166] and Morsi et al. [204] correlations

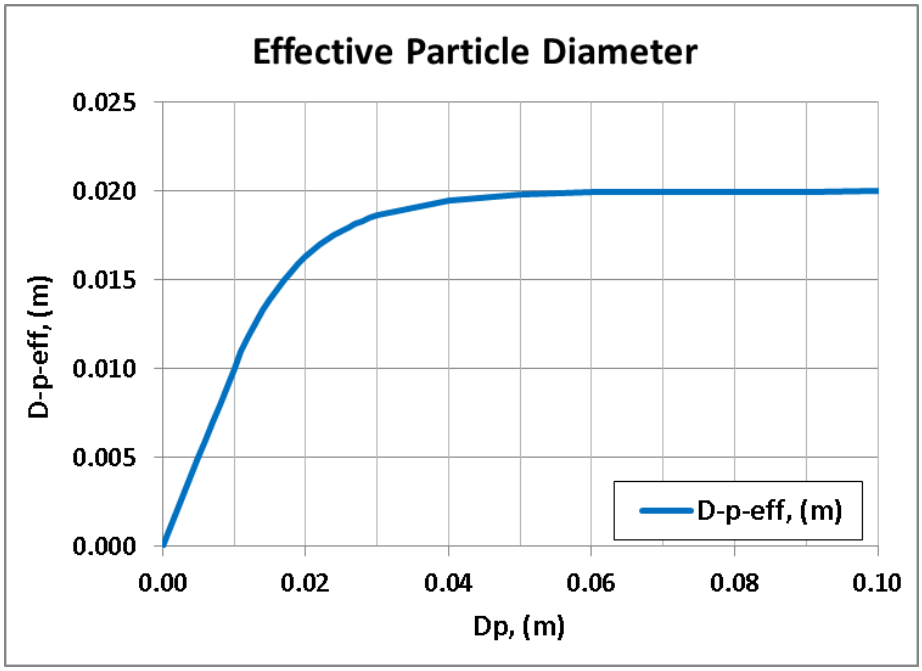


Figure 2-20 Particle diameter, as used by the correlation

The range of applicability is particle diameter D_p < 0.02 m. The effective diameter used in the correlation is not larger than 0.02 m. In order to avoid a sharp change of the calculated values at the cut-off value, a special procedure is defined to ensure smooth transition from the linear range, where $D_{p,eff} = D_p$ and the maximum range, where $D_{p,eff} = D_{p,max}$, as shown in Figure 2-20.

All coefficients used in the five regions are shown in Table 2-2. For comparison, Table 2-3 shows the coefficients used in the default correlation for droplet terminal velocity (section 2.6.3). The coefficients are obtained from:

$$A' = \frac{4/3}{A} \cdot C_{D,\,\mathrm{Re}\to\infty} \qquad B' = \frac{1}{2-B} \qquad X = 1+B \qquad Y = 1-B \qquad Z = B$$

The constant $C_{D,\mathrm{Re}\to\infty}$ is introduced here by multiplying and dividing the first constant (4/3/A) by this value. The formula is:

$$v_\infty = \left(\frac{A'}{C_{D,\,\mathrm{Re}\to\infty}} \cdot \frac{D_d^X \, g(\rho_f - \rho_p)}{\rho_p^Y \eta_f^Z} \right)^{B'}$$

$C_{D,\mathrm{Re}\to\infty}$ is introduced for flexibility. This is a user-defined input (CDRGCV, VINFRT, VINPRT, see Volume 2). This is also the asymptotic value of C_D, for Re→∞. With this parameter the user may shift the whole line, shown in Figure 2-18, up or down. This adds to the modeling flexibility.

Table 2-2 Values of constants in the alternative terminal velocity correlation (Figure 2-18)

C_D (Re→∞) = 0.47

Approx.	*A*	*4/(3A)*	*A'*	*b*	*2-b*	*B'*	*X*	*Y*	*Z*
(1)	27.00	0.049383	0.023210	0.80	1.200	0.833	1.80	0.20	0.80
(2)	6.90	0.193237	0.090821	0.40	1.600	0.625	1.40	0.60	0.40
(3)	0.900	1.48148	0.69630	0.10	1.900	0.526	1.10	0.90	0.10
(4)	0.235	5.67376	2.66667	-0.06	2.060	0.485	0.94	1.06	-0.06
(5)	0.470	2.83688	1.33333	0.00	2.000	0.500	1.00	1.00	0.00

Table 2-3 Values of constants in the default droplet terminal velocity correlation, sec. 2.6.3 (Figure 2-17)

C_D (Re→∞) = 0.44

Approx.	*A*	*4/(3A)*	*A'*	*b*	*2-b*	*B'*	*X*	*Y*	*Z*
(1)	27.00	0.049383	0.021728	0.80	1.200	0.833	1.80	0.20	0.80
(2)	0.44	3.03030	1.33333	0.00	2.000	0.500	1.00	1.00	0.00

When using the alternative correlation, the user may activate an option to tabulate the drag coefficient versus Reynolds number (CDRGCV>1000). The terminal velocity is obtained from:

$$v_\infty = \left(\frac{4/3}{C_D} \cdot \frac{D_p g(\rho_f - \rho_p)}{\rho_f} \right)^{0.5}$$

with C_D defined by a tabular function providing $C_D = f(\mathrm{Re})$ for the whole relevant range of Reynolds numbers. The argument for this function will be the Reynolds number, $Re = v_\infty D_p \rho_f / \eta_f$. The values obtained from the Tabular Function will be limited to the range of $10^{-2} < C_D < 10^2$.

An option with C_D tabulated for the entire range of Reynolds numbers is available for droplets and bubbles only. In case of particles, a constant value may be used (section 12.2.4) by setting VINPRT > 1000, VINFRT > 1000. This is done for every size section. The drag coefficients are equal to:

- C_D = const. = VINPRT – 1000 (CV pool)
- C_D = const. = VINFRT – 1000 (pool flow through JN)

On top of the drag coefficient correlation, a correlation specific for very small particles may be used, as follows.

$$v_\infty = \frac{1}{18} \cdot \frac{D_p{}^2 g(\rho_f - \rho_p)}{\eta_f} \cdot C_{small}$$

Here C_{small} is the user-defined constant (CSMLCV). This correlation is applicable for very small particles, $D_p < \sim 10^{-4}$ m. The best estimate value of C_{small} is 1.0. If a positive value is provided, then the correlation is used if the absolute value obtained from this correlation is smaller than the absolute value obtained from the drag of the correlation.

2.6.6 Particle Distribution

The local volume fraction of a dispersed component is needed in a number of cases. For example, to calculate bubble de-entrainment one needs to know the local void fraction at the pool surface. In case of bubble or droplet transport through a junction one has to know the local volumetric fractions of those components at the junction elevation.

The equation of particle position, described in section 2.6.2 provides the position of the average particle in the continuous component. From this value the local volumetric fractions must be deduced. This is done using mathematical functions, called here "shape functions". The local volumetric fraction of the dispersed component *Dcomp* at the elevation *Z*, is equal to the average volumetric fraction of this component, multiplied by the value shape function, *f*, which depends on the elevation, *Z*, average particle position, Z_{Dcomp}, and height of continuous component, Z_{Ccomp}:

$$\alpha_{Dcomp}(Z) = \overline{\alpha}_{Dcomp} \cdot f(Z, Z_{Dcomp}, Z_{Ccomp})$$

The shape functions depend on the local elevation *Z*, as well as the average particle position, Z_{Dcomp}, and the height of the continuous component, Z_{Ccomp}. The functions were selected based on the following criteria:

- The average values must be kept:

$$\frac{1}{Z_{Ccomp}} \cdot \int_0^{Z_{Ccomp}} \overline{\alpha}_{Dcomp} \cdot f(Z, Z_{Dcomp}, Z_{Ccomp}) dZ = \overline{\alpha}_{Dcomp}$$

$$\frac{1}{Z_{Ccomp}} \cdot \int_0^{Z_{Ccomp}} Z \cdot f(Z, Z_{Dcomp}, Z_{Ccomp}) dZ = Z_{Dcomp}$$

- Local void fraction should be a smooth function of all variable parameters: *Z*, Z_{Dcomp}, Z_{Ccomp}.
- If the average position of the dispersed component is in the middle of the continuous component, $Z_{Dcomp}=Z_{Ccomp}/2$, then the distribution is uniform, $f = 1.0$.
- If the average position of the dispersed component is in the ipper part of the continuous component, $Z_{Dcomp}<Z_{Ccomp}/2$, then the distribution should decrease to zero at the elevations close to about $2 \cdot Z_{Dcomp}$.

A combination of a linear and an elliptical functions was selected to represent the dispersed component distribution, since with those functions it is possible to construct in a simple way the shape functions that fulfill all the requirements mentioned above.

The distribution functions are shown in Figure 2-21. The equation determining the shape function *f* is shown below for the case when particle position is close to the interphase (Case (A) in Figure 2-21).

$$f(Z, Z_{Dcomp}, Z_{Ccomp}) = \begin{cases} \dfrac{Z_{Ccomp}}{2Z_{Dcomp}} & for: \quad Z < Z_{Dcomp} \\ \dfrac{Z_{Ccomp}}{2Z_{Dcomp}} \sqrt{1 - [(Z - Z_{Dcomp})/(Z_1 - Z_{Dcomp})]^2} & for: \quad Z_{Dcomp} < Z < Z_1 \\ 0.0 & for: \quad Z_1 < Z \end{cases}$$

where Z_1 is equal to: $Z_1 = Z_{Dcomp} \cdot (1 + 4/\pi)$.

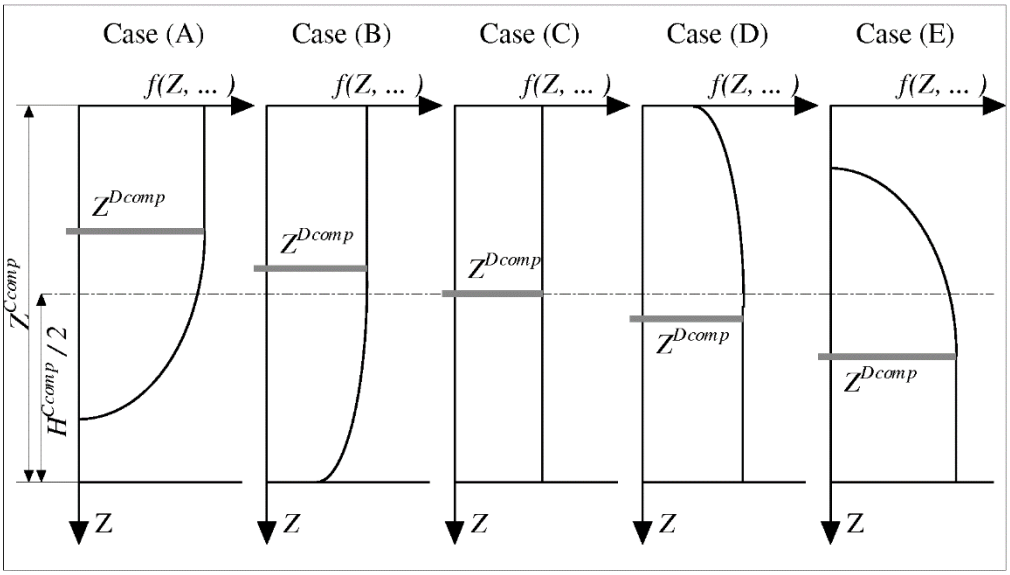


Figure 2-21 "Shape functions" for calculation of local fractions of dispersed components.

2.6.7 Treatment of Dispersed Components in the Pool

There is a difference in treatment of the dispersed component in the pool (bubbles) compared to the dispersed component in the atmosphere (droplets). In the atmosphere, all droplet sources are taken directly to the discontinuous component mass and energy equations. In the pool however, the bubble models are divided into two groups, applied in two "zones" in which the bubble behavior is different:

- *Bubble collapse in the injection zone*

  In the vicinity of the bubble source, bubble collapse is assumed to occur. Typically bubble collapse occurs if steam rich bubbles enter a relatively cold pool. The condensation of steam during bubble collapse is quite rapid and proceeds until the bubble equilibrates with the surrounding pool. There is a separate model for bubble collapse heat and mass transfer available in the Heat & Mass Transfer Package (section 7.2.5). The properties of bubbles, used for the bubble collapse calculations, are those specific for the individual bubble source. Bubble collapse is assumed to occur instantaneously, at the location of the source. This assumption is justified in Volume 3 (see also section 7.2.5).

- *Average bubble zone*

  When the bubble collapse process is finished, the bubbles begin their rise up towards the pool surface. The equilibrated bubbles are taken as a source into the pool discontinuous component equations. Thus, following the bubble collapse, the bubbles from all sources present in the pool of a Control Volume are "averaged", and represented by their average properties: bubble diameter, velocities, etc. The Heat & Mass Transfer Package contains a separate model for the heat and mass transfer during the bubble rise up period (section 7.2.6).

A separate treatment is provided for bubbles created at boiling surfaces. When the pool temperature is clearly below saturation, then the bubbles fully condense, since pure steam bubbles cannot exist in the subcooled water environment. If the bubble collapse was used when the pool is close to, or at saturation temperature, then the bubbles would still be collapsing totally, since boiling surfaces are located at some depth below the pool surface, where saturation temperature is higher. In this case nonequilibrium bubbles would be created near the pool surfaces. This approach would introduce additional numerical stiffness into the equations, and would not be physically correct. To provide a more "smooth" treatment of bubbles from boiling surfaces, the following procedure is applied.

When the pool subcooling is smaller than the limit ΔT_{BC} (default value of ΔT_{BC} is equal to 0.5 K - Volume 2) then the bubble collapse at boiling surfaces is switched off. Just switching off the collapse model would not be much helpful, since a very quick condensation of bubbles placed near the SC elevation would be calculated by the model for bubble mass transfer (section 7.2.6). Therefore it is additionally assumed that the bubbles rise up in a plume of relatively warm water (Figure 2-22). Normally the water temperature which the bubbles "see" is equal to the pool average temperature (or the pool local temperature, at the average bubble location, if stratification models are active). In the case shown in Figure 2-22, the pool temperature is: $T_{pool} > T_{sat}(p_{atms}) - \Delta T_{BC}$. In such cases the water temperature "seen" by the bubbles is equal to the warm plume temperature, assumed to be equal to the saturation at the average bubble location, $T_{sat}(p_{bubb})$.

Additional options in the program allow to define the location of boiling bubbles. The bubbles are created at the SC/TC surface elevation, but not deeper than a certain maximum depth (default value is 2.0 m - Volume 2). This parameter compensates for the fact that in fully developed boiling, the bubble collapse model is switched off. Additional consequences which the warm plume model has on the calculation of the boundary fluid temperature for Solid Heat Conductors are described in section 5.5 and 6.3.

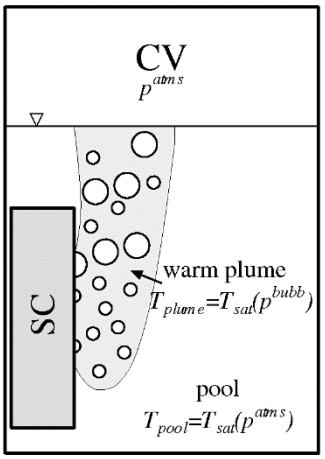


Figure 2-22 Warm plume

2.7 Stratification Models

Stratified conditions are created in a containment if a source of light fluid is located in the upper part of the containment fluid volume (of course the inverse situation, heavy fluid injected into lower part, would also promote stratification, but is of no importance for containment safety).

The light fluid may be created in various ways. In case of a severe accident, hot hydrogen is produced in the core. This hydrogen may be released to the containment through a break in the reactor coolant system. If the break is located high in the containment, stratified conditions will be promoted in the containment atmosphere. The light fluid may be created not only by mass sources but also by energy sources. For example, if steam is vented into the suppression pool, the heat of condensation warms up the water, creating light fluid near the pool surface, and thus promoting stratification of the pool. Also hydrogen recombiners may promote stratification in the containment atmosphere. The energy released during recombination may be sufficient to decrease the gas density (due to the temperature increase), in spite of the consumption of hydrogen.

The development of stratified layers is illustrated in Figure 2-23. If the source of light fluid is located in the lower part of the containment (Case A) then a circulation flow develops, driven by the density difference between the fluid column in the left part (with the source), and the right part. In this case the circulation flow continues until the whole containment is relatively well mixed. If the source is located high (Case B), then initially the density difference drives the circulation flows, just as in the previous case. However, soon enough the volume of light fluid in the upper part of the right part becomes sufficiently large to balance the light gas in the left part. The driving force ceases to exist, and circulation practically stops. The stratified layer of the light gas remains in the upper part of the containment.

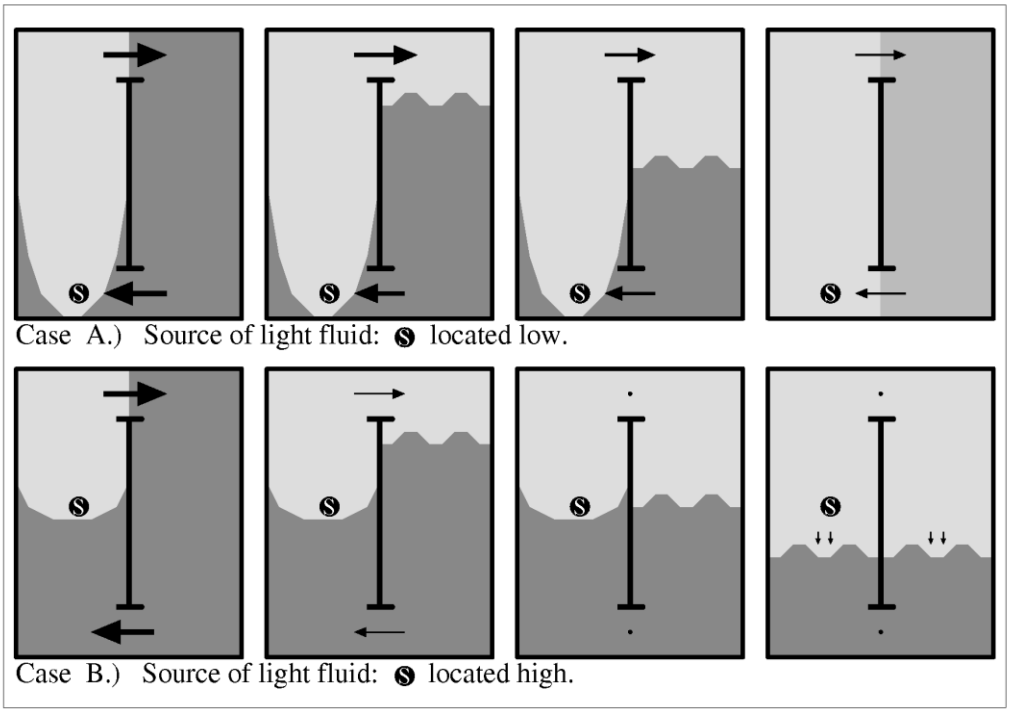


Figure 2-23 Influence of source location on development of stratified conditions

From the above discussion one may get an impression that a simple rule of thumb: "source is high in the containment", allows to determine whether stratification will occur. It is however not that simple. Probably everyone has experienced the difficulty in heating up a cold room using a relatively small, electrical heater. The warmer air rises up, as a warm plume, and creates a stratified warm layer near the ceiling, while the floor is still freezing. The same mechanism is observed during long term venting of gases from a PCCS unit into the suppression pool. The upper part of the pool is heated even if the vent submergence is large, because the warm water rises to the pool surface as a plume, and deposits the energy there.

Analyzing stratification is quite complicated. There is no doubt that the "Lumped Parameter" type codes will not be suitable to analyze stratification, since the containment compartments are represented typically by large, perfectly mixed, control volumes. The results might be to some extend improved by dividing the compartments into a number of relatively small control volumes. The LP codes however do not provide a sufficiently accurate flow solution. The codes typically use one-dimensional momentum equation (section 4.2), which is not suitable for a detailed analysis of three-dimensional effects.

For a detailed analysis one has to solve the three-dimensional Navier-Stokes equation, with a turbulence model. The most widely used is the k-ε model [173]. Such codes are typically referred to as "CFD codes", or "field codes". Recently a growing interest in CFD codes and their application to containment analyses, is observed. However, the application of field codes is a developing methodology requiring a great deal of experience, further validation efforts, and the proof of the codes' predictive capability for full-plant analysis [174].

Since a really accurate tool for the analysis of stratification has to involve the CFD solution technique, the best strategy would be to superimpose a fine grid on the Control Volumes, and solve the Navier-Stokes equation. The inclusion of such model into the SPECTRA code would be preferred, but it would require substantial effort to build such model and to integrate it into the existing numerical scheme without losing the main merits of the current code: relatively fast calculations, solution with no mass or energy errors. Therefore the concept of CFD Volumes in SPECTRA is left as an eventual option for the future, while the present version contains some simple, parametric models, which allow to perform a sensitivity study and estimate an error being made by the lack of detailed stratification models.

The present stratification models were intended to perform a parametric study, in order to obtain a conservative estimation of the containment pressure. The degree of stratification in each Control Volume can be controlled by the user, with certain input parameters, "stratification parameters" (see Volume 2). Apart from the model controlled by the user, an option is included in the code to calculate the stratification parameters internally. This was done to obtain a model which is independent of the user, and may be considered as the "SPECTRA best estimate model" for stratification analysis.

The model consists of three parts:

- *Thermal stratification of the atmosphere*
  The temperature of the atmosphere of a Control Volume is assumed to be a linear function of elevation.

$$\frac{dT_{atms}}{dZ_{atms}} = f(TSP)$$

  The local temperature at any elevation is calculated based on the average atmosphere temperature (which is a results of the energy conservation equation), and the temperature gradient dT_{atms}/dZ_{atms}, which is determined by the Thermal Stratification Parameter, TSP. The Thermal Stratification Parameter is either defined by user (in case of conservative analysis), or calculated by the code (in case of best estimate analysis).

- *Thermal stratification of the pool*
  The temperature of the pool of a Control Volume is assumed to be a linear function of elevation determined by the Pool Stratification Parameter, PSP.

$$\frac{dT_{pool}}{dZ_{pool}} = f(PSP)$$

- *Density stratification of the atmosphere*
  Thew volumetric concentrations of gases in the Control Volume atmosphere are assumed to be a linear function of elevation.

$$\frac{dx_{k,atms}}{dZ_{atms}} = f(DSP)$$

  The local gas fractions at any elevation are calculated based on the average fractions (which are obtained from the mass conservation equations), and the density gradient $dx_{k,atms}/dZ_{atms}$, which is determined by the Density Stratification Parameter, *DSP*. The Density Stratification Parameter is either defined by the user (see Volume 2), or calculated by the code.

The description below show the stratification parameters are obtained in case of best estimate analysis, that is when the stratification parameters are left to be calculated by the code.

The method used to calculate the thermal stratification parameters is to divide a Control Volume into an upper and a lower halve, and write a sort of an energy balance for both halves. Warm sources in the upper part and cold sources in the lower part promote stratification. Additionally a plume model is available. When a plume model is activated then the same procedure is applied for the stratification calculation, except that the source for which the plume model is activated is considered to be located in the upper part of the fluid volume, independently of its physical location. A criterion is present to switch off the plume model when the flow rate from the source becomes very rapid. This is done to simulate mixing induced by an incoming jet of fluid. Here a few examples are shown to illustrate how the model works.

The first example consists of a single Control Volume and a heater/cooler, represented by a Solid Heat Conductor. The temperature of the heater is first linearly increased from the initial temperature of 370 K to 400 K (between 20 s to 250 s), and then linearly decreased to 350 K (between 250 s and 480 s). Three cases are considered:

- Case 1: Heater placed high; SC-101 is located near the top of CV-101
- Case 2: Heater placed low; SC-201 is located near the bottom of CV-201
- Case 3: Heater placed low; SC-201 is located near the bottom of CV-201, plume model is activated for the heater

Results for cases 1, 2, and 3, are shown as visualization picture in Figure 2-24 and time dependent graphs in Figure 2-25, Figure 2-26, and Figure 2-27. In the first case (Figure 2-25) the SC-101 is at the top of the CV and when its temperature increases, it causes an increase of gas temperature only in the upper part of CV. The temperature at the bottom of CV remains unchanged. When SC-101 temperature decreases, and SC acts as a cooler for CV atmosphere, it first cools down the warm gas in the upper part of CV, and then (after about 430 s) it cools down the whole gas volume uniformly.

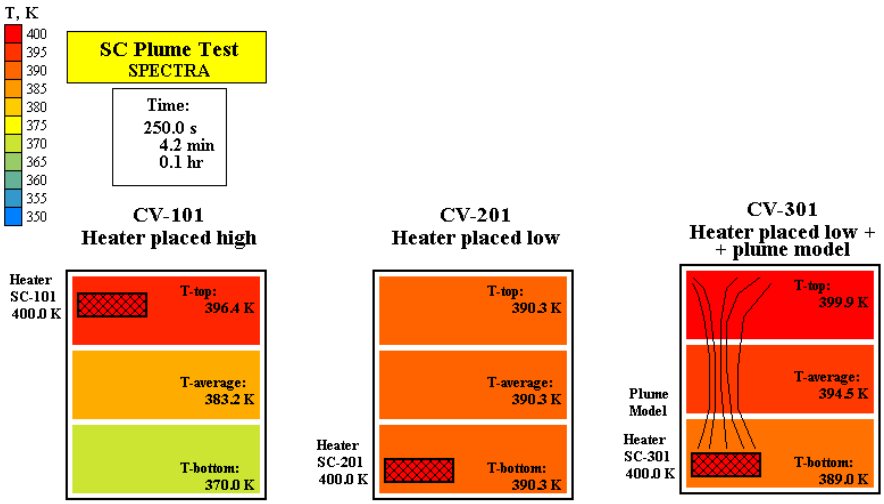


Figure 2-24 SC-Plume test, t = 250 s (maximum heater temperature).

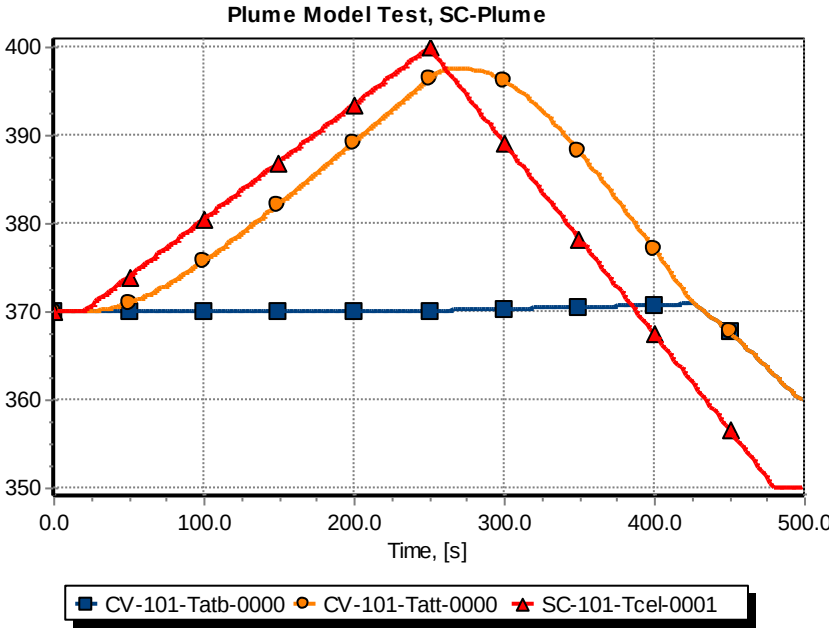


Figure 2-25 Temperatures: CV bottom, CV top, heater surface, heater at the CV top.

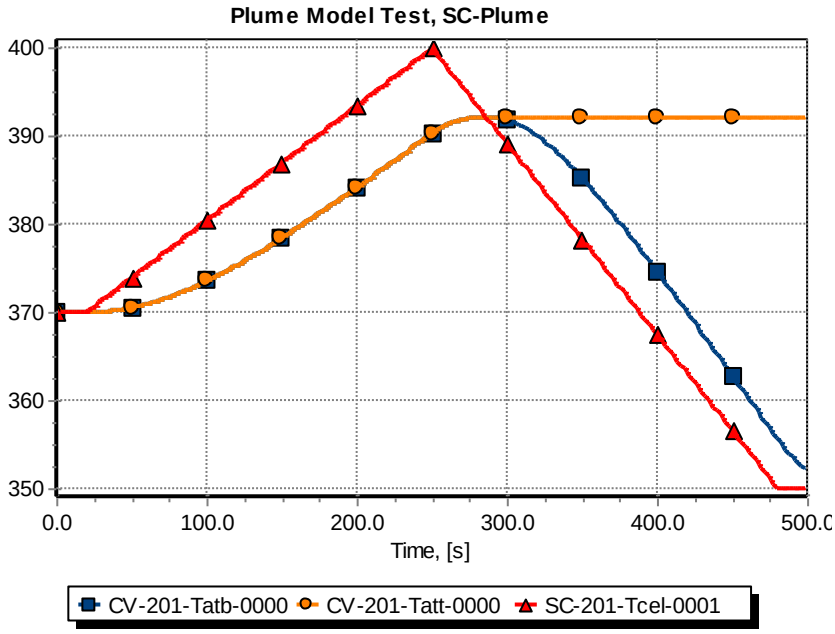


Figure 2-26 Temperatures: CV bottom, CV top, heater surface, heater at the CV bottom.

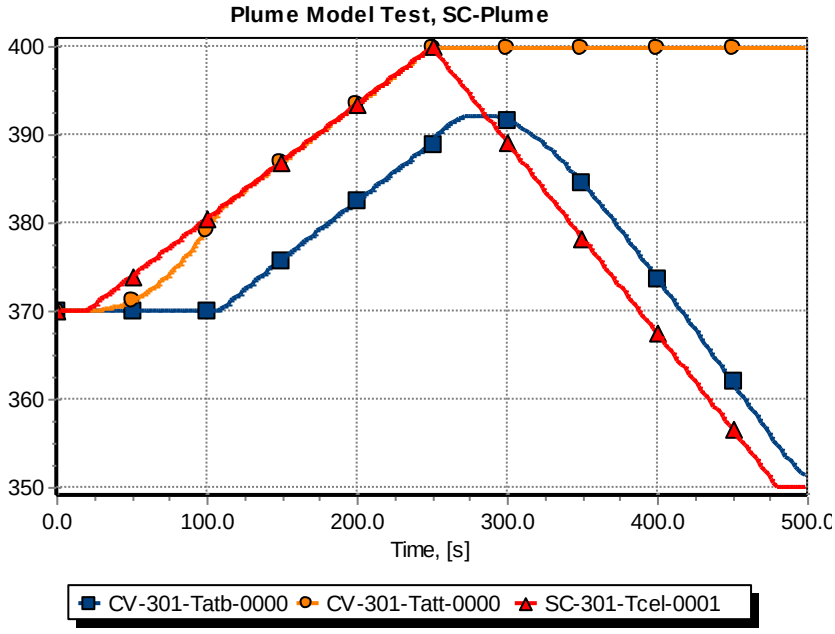


Figure 2-27 Temperatures: CV bottom, CV top, heater surface, heater at the CV bottom + plume model.

In the second case (Figure 2-26) the SC-201 is at the bottom of the CV and when its temperature increases, it heats up uniformly the whole gas space. When SC-201 temperature decreases, and it acts as a cooler for CV atmosphere, it cools down only the gas in the lower part of CV. The temperature of the gas in the upper part remains unchanged.

In the third case (Figure 2-27) the SC-301 is at the bottom of the CV, but as a result of the active plume model the upper part of the atmosphere heats up when SC-301 temperature increases (similarly as in the first case). After a while also the lower part heats up, which is different from the case 1. When SC-301 temperature decreases, and it acts as a cooler for CV atmosphere, it cools down only the gas in the lower part of CV (as in the second case).

The visualization picture shows the temperatures in all three cases at the time when the heater temperature reaches maximum (250 s). It is seen that in the first case only the upper part of CV-101 is heated, in the second case all CV-201 is heated. In the third case the whole volume of CV-301 is heated but due to the plume the highest temperatures are observed at the top of CV.

The second example is a simple model of a small "containment", shown in Figure 2-28. The model consists of five Control Volumes representing the containment compartments, and one CV representing the environment. Connections between the containment compartments are provided by six junctions. The seventh junction represents leakage from the containment to environment. The leakage was included in the model to keep the containment pressure approximately constant and avoid gas temperature rise due to compression, which would make the results somewhat more difficult to interpret. Five Solid Heat Conductors are used to represent the containment walls and structures.

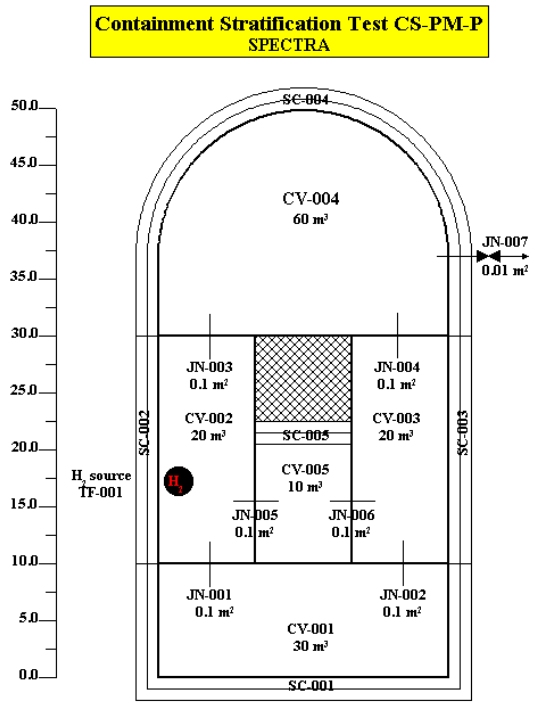


Figure 2-28 Nodalization for containment stratification test

Hot (500 K) hydrogen is injected into CV-002 in the period 500 ÷ 1500 s. Calculations were performed for three cases:

- perfect mixing in Control Volumes
- best estimate stratification in Control Volumes
- maximum stratification in Control Volumes

The results of perfect mixing run are shown in Figure 2-29 and Figure 2-33. As a result of the hydrogen injection a natural circulation flow develops, driven by the density difference between CV-002 and CV-003 (Figure 2-33). The accuracy of the flow solution for this case has been confirmed by separate calculations of the same example problem with the MELCOR code [46] (Figure 2-34). The circulation flow persists during the whole injection time, and results in quite good mixing of the gas in the containment (Figure 2-30).

Results of the best estimate stratification run are shown in Figure 2-31 and Figure 2-35. Most of the hydrogen is in the upper part of the containment; circulation flows are very small (Figure 2-35). Results of the maximum stratification run are shown in Figure 2-32. In this case the hydrogen is almost exclusively located in the upper part of the containment.

The stratification model in SPECTRA allows the user to perform a conservative estimation of containment behavior by using different assumptions concerning stratification. Examples of such use of the stratification model are ISP-42, PANDA Tests, as well as the analysis of the SWR-1000 reactor and PANDA BC tests. Those analyses are shown in Volume 3.

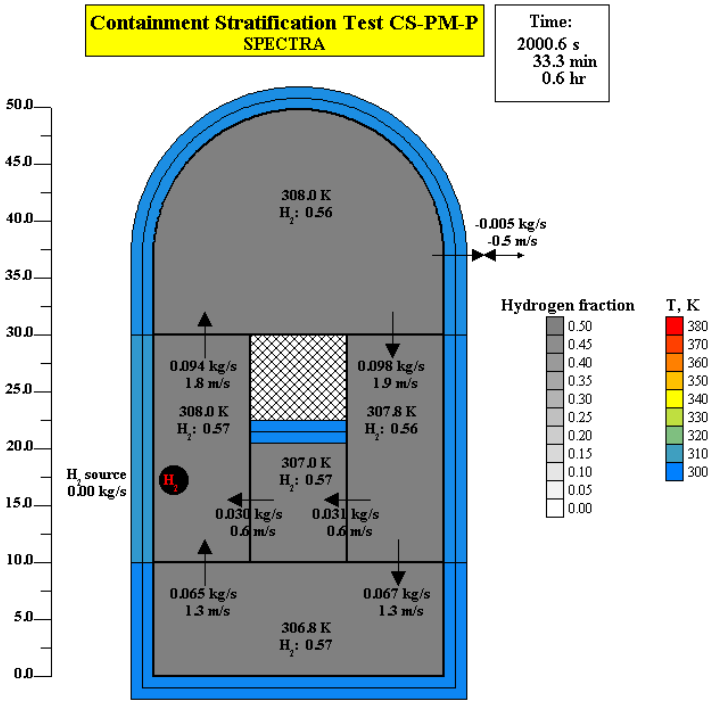


Figure 2-29 Hydrogen concentrations, SPECTRA, perfect mixing case.

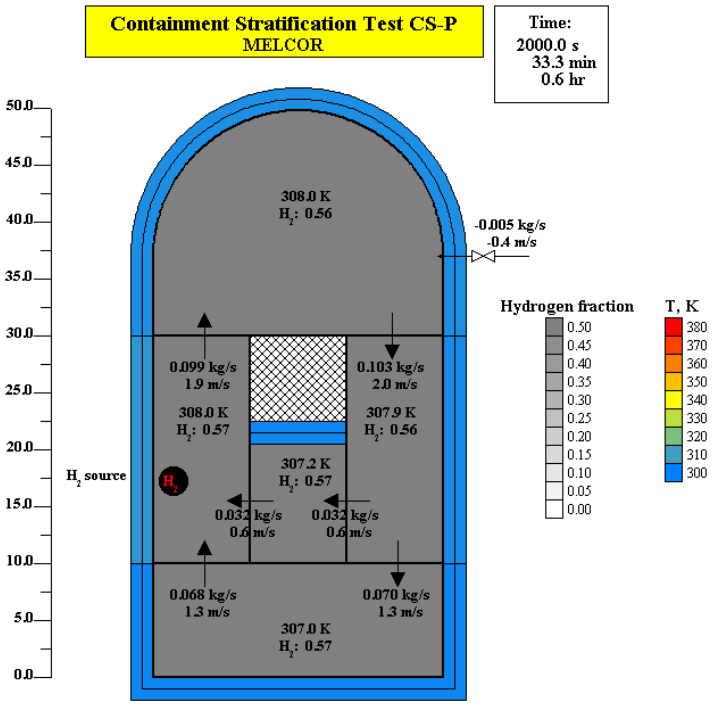


Figure 2-30 Hydrogen concentrations, MELCOR

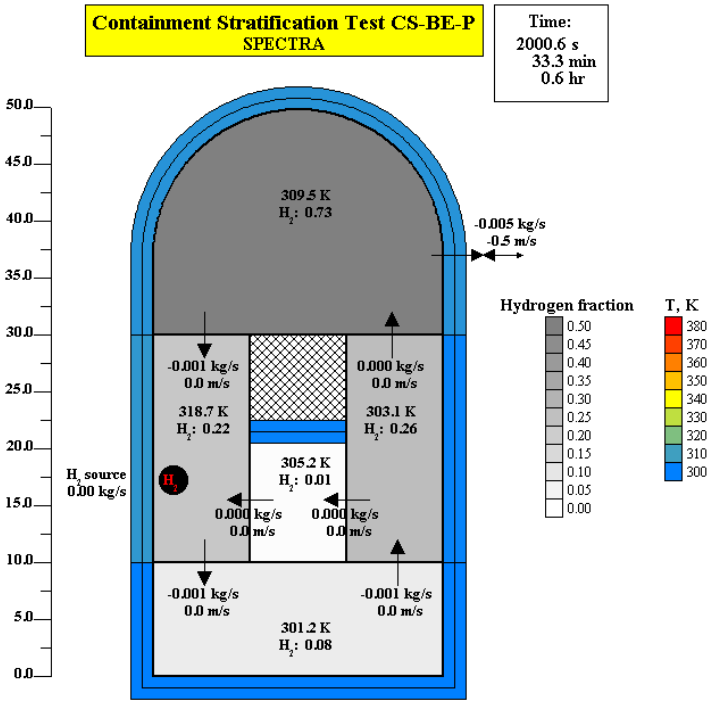


Figure 2-31 Hydrogen concentrations, SPECTRA, “best estimate” stratification case.

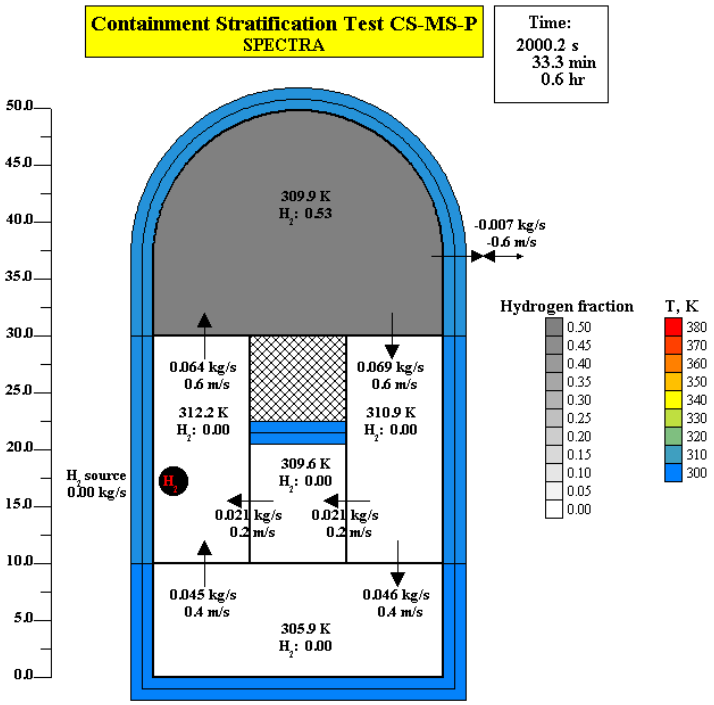


Figure 2-32 Hydrogen concentrations, SPECTRA, maximum stratification case.

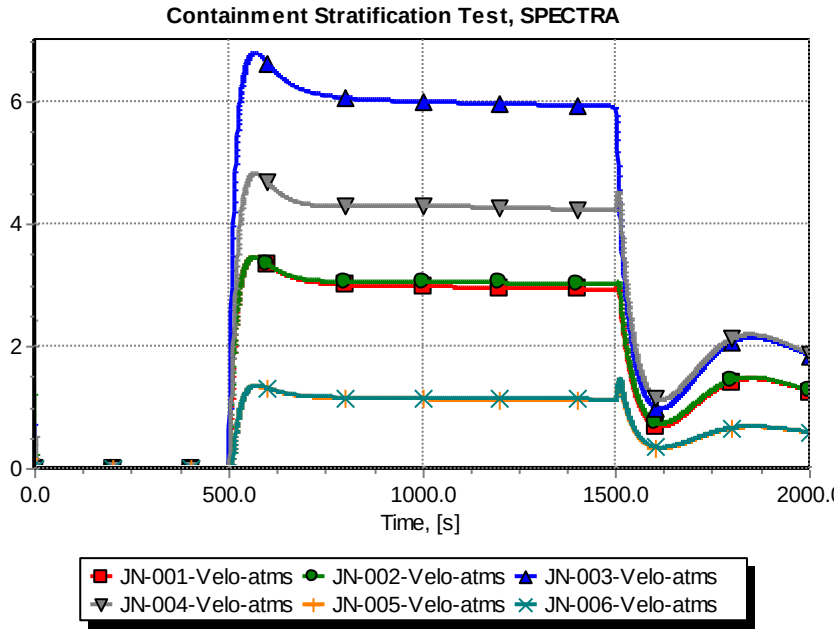


Figure 2-33 Circulation velocities, SPECTRA, perfect mixing case

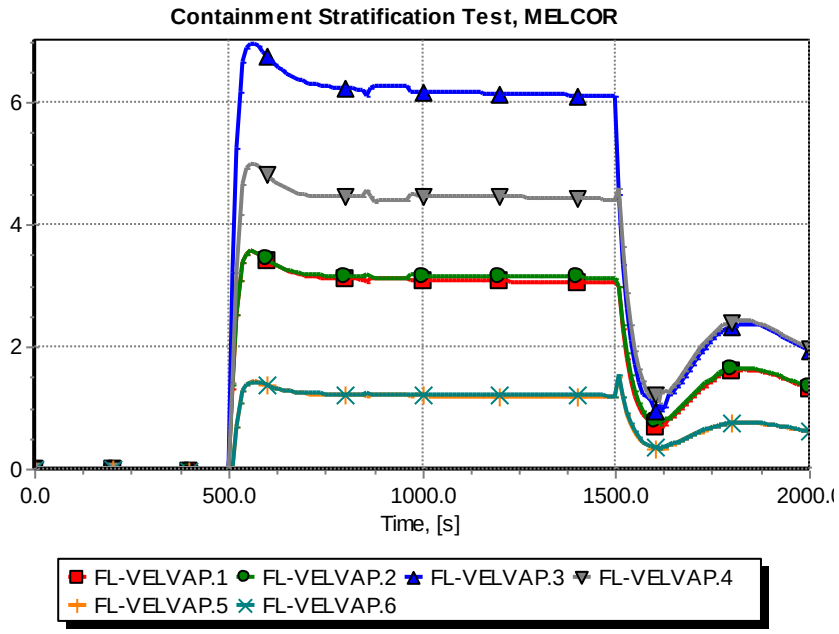


Figure 2-34 Circulation velocities, MELCOR

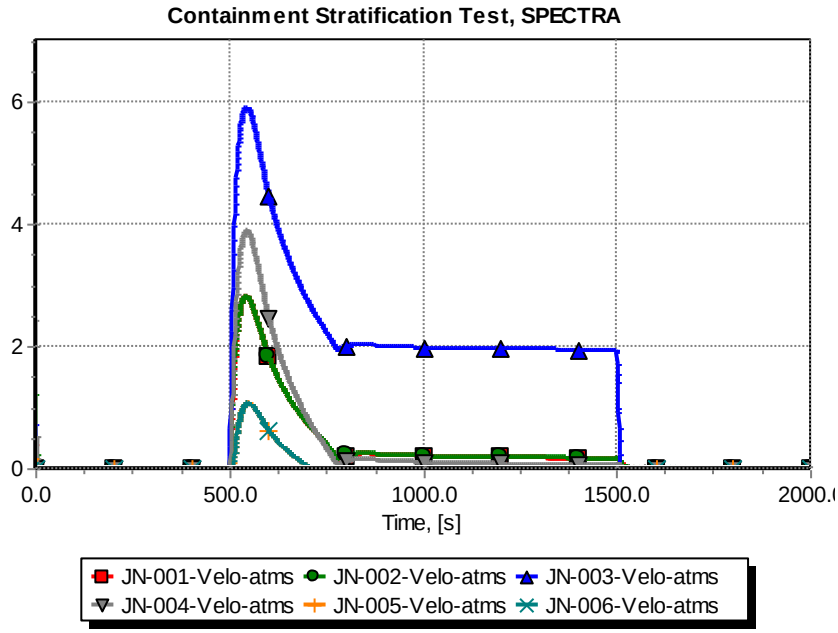


Figure 2-35 Circulation velocities, SPECTRA, best estimate case

2.8 Homogeneous Control Volumes

An option is provided to treat selected or all Control Volumes as homogeneous Control Volumes. In a Homogeneous CV only pool and atmosphere exist in a Control Volume. Atmosphere and pool are assumed to be homogeneously mixed. This approach is similar to the one taken in RELAP5 for example. The thermodynamics parameters in a homogeneous Control Volume are limited to:

- p pressure, (Pa) CV-XXX-Pres-atms
- T_{gas} gas temperature, (K) CV-XXX-Temp-atms
- T_{liq} liquid temperature, (K) CV-XXX-Temp-pool
- α void fraction, (-) CV-XXX-VolF-atms

Note that for homogeneous CV-s the plotable volume fractions and mass fractions are limited to:

- CV-XXX-VolF-atms void fraction (volume fraction of atmosphere gas)
- CV-XXX-MasF-atms mass fraction of atmosphere gas

Other volume and mass fractions, (-bubb, -drop -pool), as well as the pool level, should not be used as output and plot parameters. Similarly the flow parameters are limited to gas and liquid:

- v_{gas} gas velocity, (m/s) JN-XXX-Velo-atms
- v_{liq} liquid velocity, (m/s) JN-XXX-Velo-pool

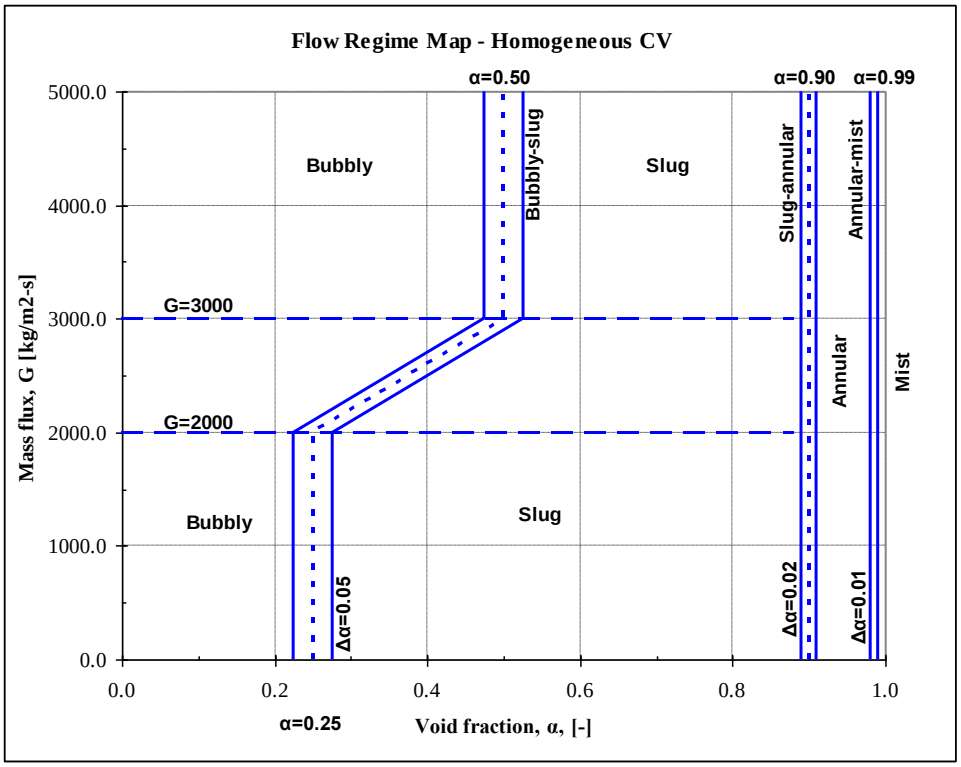


Figure 2-36 Flow regime maps for homogeneous Control Volumes

The flow regime map applied for homogeneous volumes is shown in Figure 2-36. The flow regimes include four basic flow regimes, namely bubbly, slug, annular, and mist, with three intermediate, transition flow regimes. The following flow regimes numbers are used:

1.0	bubbly flow	1.5	bubbly-slug transition
2.0	slug flow	2.5	slug-annular transition
3.0	annular flow	3.5	annular-mist transition
4.0	mist flow		
5.0	stratified flow	5.5	transitions between stratified and other regimes

The flow regime map is determined by the following parameters:

- $\alpha_{1,BS}$ void fraction for bubbly-slug transition in low mass flux region (default $\alpha_{1,BS}$=0.25)
- $\alpha_{2,BS}$ void fraction for bubbly-slug transition in high mass flux region (default $\alpha_{2,BS}$=0.5)
- $G_{1,BS}$ low mass flow limit (default $G_{1,BS}$=2000 kg/m^2-s)
- $G_{2,BS}$ high mass flow limit (default $G_{1,BS}$=3000 kg/m^2-s)
- $\Delta\alpha_{BS}$ "width" of the bubbly-slug transition (default $\Delta\alpha_{BS}$=0.05)
- $\alpha_{1,SA}$ void fraction for slug-annular transition (default $\alpha_{1,SA}$=0.9)
- $\Delta\alpha_{SA}$ "width" of the slug-annular transition (default $\Delta\alpha_{SA}$=0.02)
- $\alpha_{1,AM}$ void fraction for annular-mist transition (default $\alpha_{1,AM}$=0.99)
- $\Delta\alpha_{AM}$ "width" of the annular-mist transition (default $\Delta\alpha_{AM}$=0.01)

The bubble diameter is calculated in each flow regime as described below.

- Bubbly flow (Flow regime 1.0)

  In this flow regime the bubble diameter is obtained using the Tylor instability model:

$$D_b = D_{TI} = 2 \cdot \left(\frac{\sigma}{g \cdot (\rho_f - \rho_g)} \right)^{1/2}$$

- Slug flow

  In this flow regime the bubble (or rather the slug) diameter is obtained using Ishi-Mishima correlation (applicable for pipes):

$$D_b = D_{IM} = 0.88 \cdot D_{hyd}$$

  It is assumed that for geometries other than circular pipes the bubble diameter may be obtained from the same formula using the hydraulic diameter. The following limits are imposed on the hydraulic diameter:

$$2 \cdot D_{TI} < D_{hyd} < 1.0$$

  Therefore the maximum size of a slug is 0.88 m.

The interface area in the bubbly and the slug flow regimes is obtained from:

$$A_i = \frac{6 \cdot \alpha \cdot V_{CV}}{D_b}$$

Here V_{CV} is the total volume of CV, and α is the void fraction. The formula follows from a bubble count in a volume V_{CV} with the fraction α occupied by the bubbles:

$$N_b = \frac{\alpha \cdot V_{CV}}{V_b} = \frac{\alpha \cdot V_{CV}}{\pi \cdot D_b^{\,3} / 6} = \frac{6 \cdot \alpha \cdot V_{CV}}{\pi \cdot D_b^{\,3}}$$

The total interface area of N_b bubbles is equal to:

$$A_i = N_b \cdot A_b = \frac{6 \cdot \alpha \cdot V_{CV}}{\pi \cdot D_b^{\,3}} \cdot \pi \cdot D_b^{\,2} = \frac{6 \cdot \alpha \cdot V_{CV}}{D_b}$$

- Annular flow

  The interface area for an annular flow in a pipe with diameter D_{hyd} and length L is equal to:

$$A_i = \pi \cdot D_{hyd} \cdot L$$

Since the total volume $V_{CV} = \pi D_{hyd}^2/4 \cdot L$, the interface area is given by:

$$A_i = \frac{4 \cdot V_{CV}}{D_{hyd}}$$

It is assumed that for geometries other than circular pipes the interface area may be obtained from the same formula, using the hydraulic diameter.

- Mist flow

The interface area in the mist flow regime follows from a droplet count in a volume V_{CV} with the fraction (1–α) occupied by the droplets:

$$A_i = N_d \cdot A_d = \frac{6 \cdot (1-\alpha) \cdot V_{CV}}{\pi \cdot D_d^{\,3}} \cdot \pi \cdot D_d^{\,2} = \frac{6 \cdot (1-\alpha) \cdot V_{CV}}{D_d}$$

The droplet diameter in the mist flow is obtained from:

$$D_d = D_{fog}$$

Here D_{fog} is the diameter of droplets created in a non-equilibrium condensation process (fogging). The default value of D_{fog} is 10^{-4} m (input parameter DFOGCV).

- Stratified flow

Stratified flow may occur only in Control Volumes with horizontal flow. A horizontal flow in a Control Volume is indicated in input deck by the parameter IHORCV (see Volume 2). If the parameter is not set, the code automatically detects horizontal flow conditions by comparing horizontal and vertical velocity components in a Control Volume. The conditions for horizontal flow are checked when the horizontal velocity exceeds the vertical velocity by a factor of C_{hor}:

$$v_{hor} > v_{ver} \cdot C_{hor}$$

Here C_{hor} is a user-defined parameter (CHORCV) with a default value of 2.0. In the horizontal flow volumes the stratified flow is possible. It occurs when the mass flux and the velocity difference are below certain critical values:

$$G < G_{strat}$$
$$\left| v_{gas} - v_{liq} \right| < \Delta v_{strat}$$

Here G_{strat} and Δv_{strat} are user-defined parameters (GSTRCV, VSTRCV) with default values of 3000.0 kg/m^2-s and 1.0 m/s respectively. In case of stratified flow the interface area is equal to the horizontal cross section area of the Control Volume.

3 Fluid Property Package

3.1 Introduction

The Fluid Property Package contains properties of subcooled water, superheated steam, and several non-condensable gases. Steam as well as noncondensables are treated as real gases, the virial equation of state is used for the latter.

This section provides a brief discussion of the fluid property data package. The description is divided into three parts. The first part, shown in section 3.2, describes the saturated properties of steam and water. Next, the method of calculating subcooled water properties is described in section 3.3. Finally, section 3.5 gives a description of gas properties.

3.2 Steam/water Saturation Properties

The saturation properties are calculated using tabulated data, obtained from [29]. Tabulated data are preferred over using approximation functions, since the calculations are faster on a digital computer, at the expense of larger memory requirement. The memory size is practically unlimited on modern computers. The property data are tabulated for the range from 0.01°C to 373.976°C. The properties may be obtained by a call to the procedures with one of the following arguments:

- Temperature, (K) T
- Pressure, (Pa) p
- Enthalpy of saturated liquid, (J/kg) $h_{sat.liq}$

The properties which are being returned by the saturated steam/water property routines consist of two out of the three parameters listed above (the third one is an input parameter), as well as:

- Enthalpy of saturated vapor, (J/kg) $h_{sat.vap}$
- Density of saturated liquid and vapor, (kg/m^3) $\rho_{sat.liq}$ $\rho_{sat.vap}$
- Entropy of saturated liquid and vapor, (J/kg/K) $s_{sat.liq}$ $s_{sat.vap}$
- Dynamic viscosity of saturated liquid and vapor, (kg/m/s) $\eta_{sat.liq}$ $\eta_{sat.vap}$
- Thermal conductivity, (W/m/K) $k_{sat.liq}$ $k_{sat.vap}$
- Specific heat, (J/kg/K) $c_{p,\,sat.liq}$, $c_{p,\,sat.vap}$
- Prandtl number ($\eta c_p/k$), (-) $Pr_{sat.liq}$ $Pr_{sat.vap}$
- Density derivative, (kg/m^3/K) $\partial\rho_{sat.liq}/\partial T$ $\partial\rho_{sat.vap}/\partial T$
- Thermal expansion coefficient for saturated water, (1/K) $\beta_{sat.liq} = (1/\rho_{sat.liq})\times(\partial\rho_{sat.liq}/\partial T)$
- Surface tension between liquid and vapor phase, (N/m) $\sigma_{liq\text{-}vap}$

The properties are calculated using third order (cubic) interpolation between the nearest tabulated values. The third order interpolation ensures continuity of the function and its first derivative (see section 17.1).

The density derivatives are obtained based on the assumption that the derivative may be well represented by finite difference, as follows:

$$\frac{\partial \rho}{\partial T} = \frac{\Delta \rho}{\Delta T} = \frac{\rho(T + \Delta T) - \rho(T)}{\Delta T}$$

The temperature difference, ΔT, used to calculate derivatives is assumed as 0.1 K. The densities at temperatures T and $T+\Delta T$, are calculated using third order interpolation.

3.3 Water Properties

Properties of subcooled water depend generally on both temperature and pressure. The pressure dependence is however very weak, and may be neglected in practical calculations. The assumption taken in building the water property routines was that all properties, except for density and internal energy, are independent of pressure and are therefore represented by their saturation properties:

$$X_{liq}(p,T) = X_{sat.liq}(T)$$

In the above equation X is a replacement symbol, representing all liquid properties listed in section 3.2, except for the density and the enthalpy. The density is calculated as follows:

$$\rho_{liq}(p,T) = \rho_{sat.liq}(T) \cdot \left(\frac{\rho_{liq}(p_{sat},T)}{\rho_{liq}(p,T)} \right)^{-1}_{Thomson}$$

where the ratio of saturated water density to the actual water density is calculated using the method of Thomson [30]:

$$\left(\frac{\rho_{liq}(p_{sat},T)}{\rho_{liq}(p,T)} \right)^{-1}_{Thomson} = 1 + c \cdot \ln\left(\frac{\beta + p}{\beta + p_{sat}} \right)$$

$$\beta = \left[-1 + a \cdot \left(1 - \frac{T}{T_{crit}} \right)^{1/3} + b \cdot \left(1 - \frac{T}{T_{crit}} \right)^{2/3} + d \cdot \left(1 - \frac{T}{T_{crit}} \right)^{1/3} + e \cdot \left(1 - \frac{T}{T_{crit}} \right)^{4/3} \right] \cdot p_{crit}$$

where T_{crit} and p_{crit} are the critical temperature and pressure respectively. The values of T_{crit} and p_{crit}, as well as the values of constants, a, b, c, ..., in the Thomson's model are obtained from [31]. The values of critical parameters are:

- T_{crit} = 647.3 K
- p_{crit} = 2.212×10^7 Pa

The values of the model constants are shown in Table 3-1.

Table 3-1 Values of constants in the Thompson model

a	*b*	*c*	*d*	*e*
–9.070217	62.45326	0.1988366	–135.1102	157.8569

The present version of SPECTRA does not use the Thompson model to calculate the water density. It was found out that use of this model resulted in flow problems in the water flow solution. Therefore the model is currently deactivated and the density is obtained from:

$$\rho_{liq}(p,T) = \rho_{sat.liq}(T)$$

Thus the liquid is assumed to be incompressible. Application of the compressibility resulted in "phantom flows". In order to prevent the phantom flows the gravity head in the flow solution should be calculated using an effective water density of:

$$\rho_{eff} = \rho \cdot \left(1 + \frac{1}{2} \cdot \frac{\partial \rho}{\partial p} \cdot gz\right)$$

This may be done in a future version of the code, if necessary.

The specific enthalpy and specific internal energy of water are obtained from:

$$h_{liq}(p,T) = h_{sat.liq}(T) + \frac{p - p_{sat}}{\rho_{liq}(p,T)}$$

$$u_{liq}(p,T) = h_{liq}(p,T) - \frac{p}{\rho_{liq}(p,T)}$$

The fluid property package offers two methods of getting the water properties, using different sets of input arguments:

- Pressure and temperature as input arguments (subroutine WATERT):

$$WATERT: \quad Properties = f(p,T)$$

- Pressure and internal energy as input arguments (subroutine WATERU):

$$WATERU: \quad Properties = f(p,u_{liq})$$

While the first method is quite straightforward, the second requires an internal iteration to calculate the properties. The iteration continues until the desired convergence is achieved (relative error between the assumed and the calculated internal energy is less than 10^{-9}). The second method is more useful in practice. Since it is the internal energy which is obtained from the energy balance, the second method allows to obtain quickly all properties for the new energy.

3.4 Alternative Fluid (Liquid Metals, Molten Salts, etc.) Properties

The alternative fluid may be specified in SPECTRA instead of water. In such case water properties are replaced by the properties of the alternative fluid. This option is included primarily for analyzing liquid metals.

When the alternative fluid is used, the user must supply the saturation properties of the desired fluid. These are defined by two tables, giving the thermodynamic and the thermo-physical properties versus temperature. The temperature may be expressed in Kelvin or degree Celsius. The tables include:

- Thermodynamic properties:
  - Temperature, (K or ˚C) T
  - Saturation pressure, (Pa) p_{sat}
  - Enthalpy of saturated liquid and vapor, (J/kg) $h_{sat.liq}$ $h_{sat.vap}$
  - Density of saturated liquid and vapor, (kg/m^3) $\rho_{sat.liq}$ $\rho_{sat.vap}$
  - Entropy of saturated liquid and vapor, (J/kg/K) $s_{sat.liq}$ $s_{sat.vap}$

- Thermo-physical properties:
  - Dynamic viscosity of saturated liquid and vapor, (kg/m/s) $\eta_{sat.liq}$ $\eta_{sat.vap}$
  - Thermal conductivity of sat. liquid and vapor, (W/m/K) $k_{sat.liq}$ $k_{sat.vap}$
  - Prandtl number of saturated liquid and vapor ($\eta c_p/k$), (-) $Pr_{sat.liq}$ $Pr_{sat.vap}$
  - Surface tension, (N/m) $\sigma_{liq\text{-}vap}$

Only selected tables need to be filled-in (see Volume 2). The user may omit some, or even all the tables. In such case the missing tables are filled-up using the built-in water properties. Since the alternative fluid is intended to represent liquid metal coolants, typically only the saturated liquid properties need to be entered, while the saturated vapor properties may be omitted. An example of defining liquid lead as an alternative fluid is provided in Volume 3.

Note that when the alternative fluid (liquid metal) is defined, it is used in all Control Volumes and therefore liquid water is not available. If the user wishes to perform calculations with a model in which both liquid metal and water are needed, then the model must be split into two parts, one containing liquid metal and one containing water. The two models may then be run in parallel using the synchronized option in the EDF Package (section 18.2). An example of such synchronized run with liquid metal on the primary side and water on the secondary side is shown in Volume 3.

As mentioned above, typically the alternative fluid is only used in terms of liquid, while vapor properties are not defined. Therefore any evaporation should be avoided (steam does not properly represent the properties of the vapor of given metal). This includes evaporation from a pool surface, a droplet surface, and evaporation into a bubble, all of which are automatically disabled by the input procedures (see Volume 2). Secondly, boiling is prevented by setting the maximum liquid temperature to a value of ten Kelvins below the saturation: $T \leq T_{sat}(p) - 10.0$.

Furthermore the critical flow model in SPECTRA is appropriate for water as a liquid. Therefore for the liquid metal calculations the critical flow model is deactivated by setting the limiting velocity to a large number (see Volume 2).

3.5 Gas Properties - Built-in Gases

The built-in gas property data base contains properties of five non-condensable gases: hydrogen, helium, nitrogen, oxygen, and carbon dioxide, as well as the properties of steam. All gases are treated as real gases. For non-condensable gases the virial equation of state is used. Steam properties are calculated using a method based on Helmholtz function, described in [29].

The equations defining gas properties are relatively complicated, in some cases involve iterative calculations, and therefore are time consuming. To save computing time the properties were pre-computed, and tabulated in a large data base as functions of temperature and pressure. During calculations the properties are obtained by bi-linear interpolation between the nearest tabulated points. This method is very fast, specifically since the data are tabulated in regular intervals, and consequently no search is needed to find the nearest data points. A disadvantage of this method is a rather large memory requirement. The relatively large memory requirement is not a serious disadvantage of the method. Modern computers come with practically infinite memory.

3.5.1 Gas Property Data Base

The following properties:

- Density, (kg/m^3): $\rho = \rho(p, T)$, (equation of state)
- Internal energy, (J/kg): $u = u(p, T)$
- Viscosity, (kg/m/s): $\mu = \mu(p, T)$
- Thermal conductivity, (W/m/K): $k = k(p, T)$

are pre-computed and tabulated in the gas property data base. Properties of the following six gases are defined in the data base:

- hydrogen, H_2,
- helium, He,
- steam, H_2O,
- nitrogen, N_2,
- oxygen, O_2,
- carbon dioxide, CO_2.

The gas property data base is shown schematically in Figure 3-1. The properties are tabulated for the temperature range from 270.0 K to 3070.0 K, and for the pressure range from 1.0×10^5 Pa to 2.1×10^7 Pa. An extrapolation scheme is provided for low pressures, so that the data base effectively covers pressures from 0 to 2.1×10^7 Pa.

The properties are tabulated for temperatures every 10.0 K, and pressures every $1.0\cdot10^5$ Pa (the pressure dependence of gas properties is rather weak, therefore this large pressure spacing is considered adequate - see Volume 3 – Verification and Validation. The data base consists of 281 temperature points, and 210 pressure points.

The properties of steam are extrapolated into the subcooled, meta-stable region, so there is no problem in obtaining properties if steam is somewhat subcooled. This may occur in calculations, because of the non-equilibrium treatment of fluids in Control Volumes.

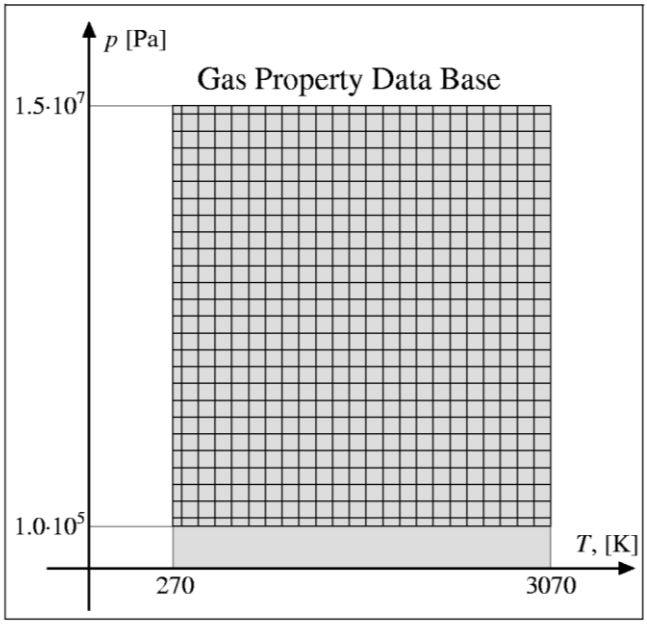


Figure 3-1 Gas property data base.

The methods of calculation of the properties which are defined in the data base are described below for each property. The discussion is limited to non-condensable gases. The steam properties were obtained using the program listed in [29].

- *Density, ρ(p,T)*
  For all gases except for steam the density has been calculated using a virial equation of state. For steam the density has been calculated using Helmholtz function [29].

  The virial equation of state ([31], equation 3-5.1) is used to calculate the specific volume:

$$\frac{pv}{RT} = Z = 1 + \frac{B(T)}{v} + \frac{C(T)}{v^2} + \frac{D(T)}{v^3} + \ldots$$

  where: p - pressure, (Pa),
  R - individual gas constant, (J/kg/K),
  T - temperature, (K),
  v - specific volume, (m^3/kg),
  Z - gas compressibility, (-),
  B - second virial coefficient (function of temperature), (m^3/kg),
  C - third virial coefficient (function of temperature), (m^6/kg^2),
  D - fourth virial coefficient (function of temperature), (m^9/kg^3).

The virial equation may also be written as a power series of p instead of v ([31], equation 3-5.2):

$$Z = 1 + B(T)\frac{p}{RT} + C(T)\left(\frac{p}{RT}\right)^2 + D(T)\left(\frac{p}{RT}\right)^3 + \ldots$$

Sometimes it is written in the form:

$$Z = 1 + B'(T)p + C'(T)p^2 + D'(T)p^3 + \ldots$$

where the virial coefficients, B', C', D', are of course expressed in different units. Often the virial equation is truncated to contain only the second virial coefficient:

$$\frac{pv}{RT} = Z = 1 + B(T)\frac{p}{RT}$$

The virial equations used to calculate gas densities are discussed separately for two groups of gases. First, the equations used for hydrogen, helium, and carbon dioxide are described. Next, the equations used for nitrogen and oxygen are described.

Gases: H_2, He, CO_2
The virial equation of state truncated to the second virial coefficient is used for these gases. The second virial coefficient is calculated using the following polynomial ([32], p.6-28):

$$B_m(T) = 10^{-3}\left(a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4\right)$$

where B_m is in (m^3/kmole). Since the virial coefficient is in "molar units", it is given here with the subscript m. The constants a_1, a_2, a_3, a_4, a_5, are taken from [32] (pages 6-28 through 6-37) and are reproduced in Table 3-2. The factor 10^{-3} is the conversion from (cm^3/mole) to (m^3/kmole).

Table 3-2 Polynomial coefficients to calculate second virial coefficient, [32].

Gas	a_1 cm^3/mole	a_2 cm^3/(mole K)	a_3 cm^3/(mole K^2)	a_4 cm^3/(mole K^3)	a_5 cm^3/(mole K^4)
H_2	15.4	–9.0	–0.2	0.0	0.0
He	12.0	–1.0	0.0	0.0	0.0
CO_2	–127	–288	–118	0.0	0.0

Since the virial coefficient is in "molar units", the molar gas constant, R_m, must be used to obtain the compressibility factor:

$$Z = 1 + \frac{B_m(T)\,p}{R_m T}$$

When the second virial coefficient, $B_m(T)$, is known, the density is obtained from:

$$\rho(p,T) = \frac{p}{RT}\left(1 + \frac{B_m(T)\,p}{R_m T}\right)^{-1}$$

Gases: N_2, O_2
The virial equation in the form of the power series of pressure with second, third and fourth virial coefficients is used. The equation has the form: ([39], chapters 7 and 8):

$$Z = 1 + B'(T)\,p + C'(T)\,p^2 + D'(T)\,p^3$$

where: p - pressure, (atm), (the pressure in (Pa) is divided by $1.01325 \cdot 10^5$, to obtain the atmospheres),
B' - second virial coefficient, (atm^{-1}),
C' - third virial coefficient, (atm^{-2}),
D' - fourth virial coefficient, (atm^{-3}).

The virial coefficients, B', C' and D', are tabulated in [39], for temperatures between 100 and 3000 K. The second and third virial coefficients, B' and C', were obtained through the Lennard-Jones potential. The fourth virial coefficient was fitted empirically (see [39], pages 297 and 369). The virial coefficients are calculated using cubic interpolation between tabulated data points. The virial coefficients, as calculated by the cubic interpolation, are shown in Figure 3-2 (a) and (b). The gas density is obtained from:

$$\rho(p,T) = \frac{p}{RT}\left[1 + B'(T)\,p + C'(T)\,p^2 + D'(T)\,p^3\right]^{-1}$$

- *Specific internal energy, u(p,T)*
The specific internal energy is calculated as:

$$u(p,T) = \int_{T_{ref}}^{T} c_p(T')\,dT' - \frac{p}{\rho(p,T)}$$

where: c_p - isobaric specific heat, (J/kg/K),
T_{ref} - reference temperature, (K), for all non-condensable gases assumed to be equal to 0 K.

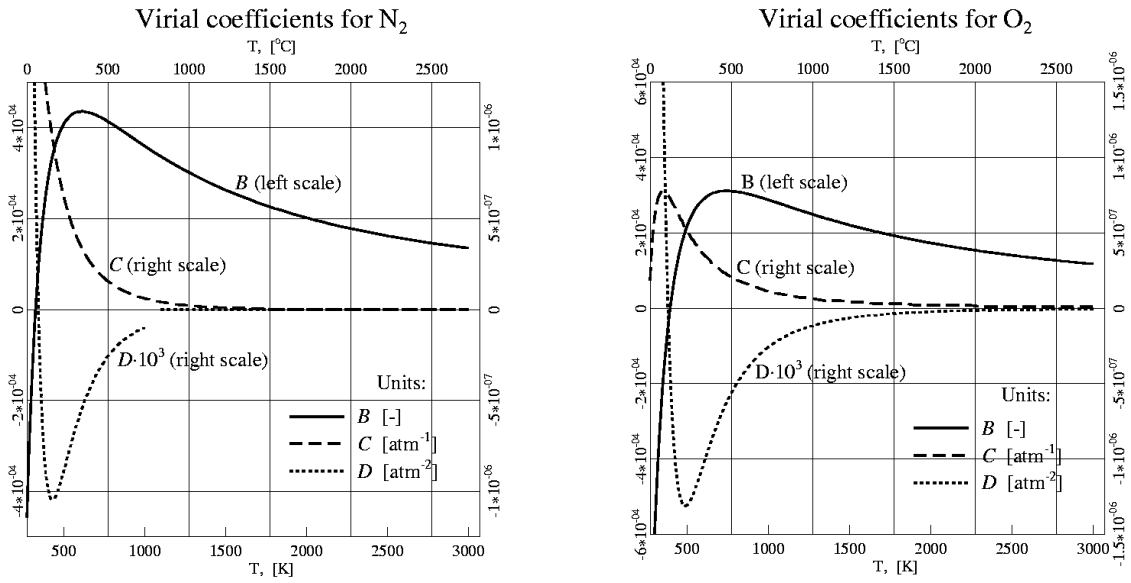


Figure 3-2 Virial coefficients for nitrogen (a) and oxygen (b).

The specific heat, $c_p(T)$ is calculated using the following polynomial ([31], Appendix A, page 657):

$$c_p(T) = \frac{10^3}{\mu}\left(C_A + C_B T + C_C T^2 + C_D T^3 \right)$$

where μ is molecular weight and C_i are the polynomial coefficients, given in [31], Appendix A, pages 661 - 668. The values are reproduced in Table 3-3.

Table 3-3 Polynomial coefficients to calculate specific heat, [31].

Gas	C_A (J/mol/K)	C_B (J/mol/K^2)	C_C (J/mol/K^3)	C_D (J/mol/K^4)
H_2	$2.714\cdot10^{1}$	$9.274\cdot10^{-3}$	$-1.381\cdot10^{-5}$	$7.645\cdot10^{-9}$
He	$2.080\cdot10^{1}$	0.0	0.0	0.0
N_2	$3.115\cdot10^{1}$	$-1.357\cdot10^{-2}$	$2.680\cdot10^{-5}$	$-1.168\cdot10^{-8}$
O_2	$2.811\cdot10^{1}$	$-3.680\cdot10^{-6}$	$1.746\cdot10^{-5}$	$-1.065\cdot10^{-8}$
CO_2	$1.980\cdot10^{1}$	$7.344\cdot10^{-2}$	$-5.602\cdot10^{-5}$	$1.715\cdot10^{-8}$

- *Viscosity, $\mu(p,T)$*
  Low pressure gas viscosity is calculated using the method of Chung et al. [40], or the method of Lucas [41]. For gases other than the quantum gases (H_2 and He) the method of Chung et al. is used. For quantum gases the method of Chung et al. is not valid. Therefore for those gases the method of Lucas is used. The equations of both methods are shown in [31].

  After the low pressure viscosity has been calculated using one of the methods presented above, the influence of pressure on viscosity is calculated. To account for the pressure influence on the gas viscosity the method of Richtenberg is used [42]. In this method the viscosity ratio, $\eta/\eta_{p\approx 0}$, is calculated based on the reduced pressure, reduced temperature and dimensionless dipole moment. The equations of the Richtenberg method are shown in [31].

- *Thermal conductivity, k(p,T)*
  Low pressure thermal conductivity is calculated using the method of Miller et al. [43]. The conductivity is obtained from the following formula:

$$k_{p=0}(T) = A + BT + CT^2 + DT^3$$

where: $k_{p\approx 0}$ - gas thermal conductivity at low pressure, (W/m/K),
$A, B, ...$ - polynomial coefficients.

The values of polynomial coefficients, *A, B, C, D,* are shown in [31]. For the six gases being considered they are reproduced in Table 3-4.

Table 3-4 Polynomial coefficients to calculate low-pressure thermal conductivity.

Gas	*A*, (W/m/K)	*B*, (W/m/K^2)	*C*, (W/m/K^3)	*D*, (W/m/K^4)
H_2	$8.099\cdot10^{-3}$	$6.689\cdot10^{-4}$	$-4.158\cdot10^{-7}$	$1.562\cdot10^{-10}$
He	$3.722\cdot10^{-2}$	$3.896\cdot10^{-4}$	$-7.450\cdot10^{-8}$	$1.290\cdot10^{-11}$
H_2O	$7.341\cdot10^{-3}$	$-1.013\cdot10^{-5}$	$1.801\cdot10^{-7}$	$-9.100\cdot10^{-11}$
N_2	$3.919\cdot10^{-4}$	$9.816\cdot10^{-5}$	$-5.067\cdot10^{-8}$	$1.504\cdot10^{-11}$
O_2	$-3.273\cdot10^{-4}$	$9.966\cdot10^{-5}$	$-3.743\cdot10^{-8}$	$9.732\cdot10^{-12}$
CO_2	$-7.215\cdot10^{-3}$	$8.015\cdot10^{-5}$	$5.477\cdot10^{-9}$	$-1.053\cdot10^{-11}$

To account for the pressure influence on the gas conductivity the method of Stiel and Thodos [35] is used. In this method thermal conductivity, *k*, is equal to ([31], section 10-5, page 521):

$$k(p,T) = k_{p\to 0}(T)\cdot f(T_{crit},\ p_{crit},\ Z_{crit},\ v_{crit},\ v,\ M_w)$$

T_{crit} critical temperature, (K),
p_{crit} critical pressure, (Pa),
Z_{crit} critical compressibility, (-),
v_{crit} critical specific volume, (m^3/kg),
v specific volume, (m^3/kg),
M_w molar weight, (kg/kmole).

The equations defining the pressure dependent multiplier, are shown in [31].

- *Thermal conductivity of Helium, k(p,T)*
  It was found out that the method describe above does not provide a sufficient accuracy in case of Helium. Therefore, for this gas, the formula from KTA rules [202] is used. The formula is:

$$k(p,T) = 2.682 \cdot 10^{-3} \cdot (1 + 1.123 \cdot 10^{-8} \cdot p) \cdot T^{0.71 \cdot (1 - 2 \times 10^{-9} \cdot p)}$$

  The improvement of accuracy of this formula, compared to the formerly used formula from [31], is discussed in Volume 3.

The properties of all gases, as calculated by the methods described above, are shown in graphs below; each graph gives of the following properties:

- Compressibility, defined as: $Z = pv/RT$. The value of compressibility is a measure of deviation of the gas density from the density given by the ideal gas equation of state.

- Ratio of the constant volume specific heat at given temperature, $c_v(T)$, to the specific heat at low temperature, $c_v(T_0)$, where T_0 is the lower-most temperature data point in the SPECTRA Fluid Property Data Package (T_0 = 270.0 K). In case of steam the internal energy is shown instead of the specific heat ratio. This is because steam properties are generated without using specific heat. The steam internal energy data table is obtained directly, while in case of all other gases the internal energy data tables are obtained by integrating the specific heat.

- Dynamic viscosity.

- Thermal conductivity.

The above properties are shown for six gases available in the Fluid Property data tables, H_2, He, H_2O, N_2, O_2, CO_2.

- **H_2**.
  Hydrogen properties are shown in Figure 3-3. It may be observed that in the range of parameters considered, hydrogen behaves very much like a perfect gas. Deviation from the perfect gas equation of state is below 10% in the considered pressure range. Almost no influence of pressure on the thermal properties is observed.

- **He**
  Helium properties are shown in Figure 3-4. Helium, like hydrogen, behaves like a perfect gas. Deviation from the perfect gas equation of state is below 8% in the considered pressure range. Almost no influence of pressure on thermal properties is observed.

- **H_2O**
  Steam properties are shown in Figure 3-5. The compressibility of steam decreases close to the saturation line. In the subcooled steam range the density was extrapolated in such way that the compressibility smoothly levels with increasing subcooling. A similar extrapolation in the subcooled range was applied for the thermal conductivity. In case of internal energy and viscosity a simple linear extrapolation of properties in subcooled range is used. For practical application the extrapolated values are not important, since subcooled steam will quickly condense and only very small subcooling may be encountered. Nevertheless the data points in the data tables had to be filled. This was done to obtain reasonable and safe extrapolated

properties, so that, should a significant subcooling ever been encountered, the property calculating subroutines should not fail in calculating the state and the obtained values should be in reasonable agreement with the values expected based on observations.

- **N2**
  Nitrogen properties are shown in Figure 3-6. Deviation from the perfect gas equation of state is, for the considered range of pressure, up to about 7 %. Some small influence of pressure on viscosity and thermal conductivity is observed, specifically for low temperatures.

- **O2**
  Oxygen properties are shown in Figure 3-7. Deviation from the perfect gas equation of state is the largest for temperatures close to 0°C. At this temperature the maximum deviation is almost 10%. The influence of pressure on viscosity and thermal conductivity is similar to that observed for nitrogen.

- **CO2**
  Carbon dioxide properties are shown in Figure 3-8. Deviation from the perfect gas equation of state is quite large for low temperatures. Above 200°C the deviation does not exceed 10%, but below that temperature it becomes larger. The influence of pressure on viscosity and thermal conductivity is significant especially below 200°C.

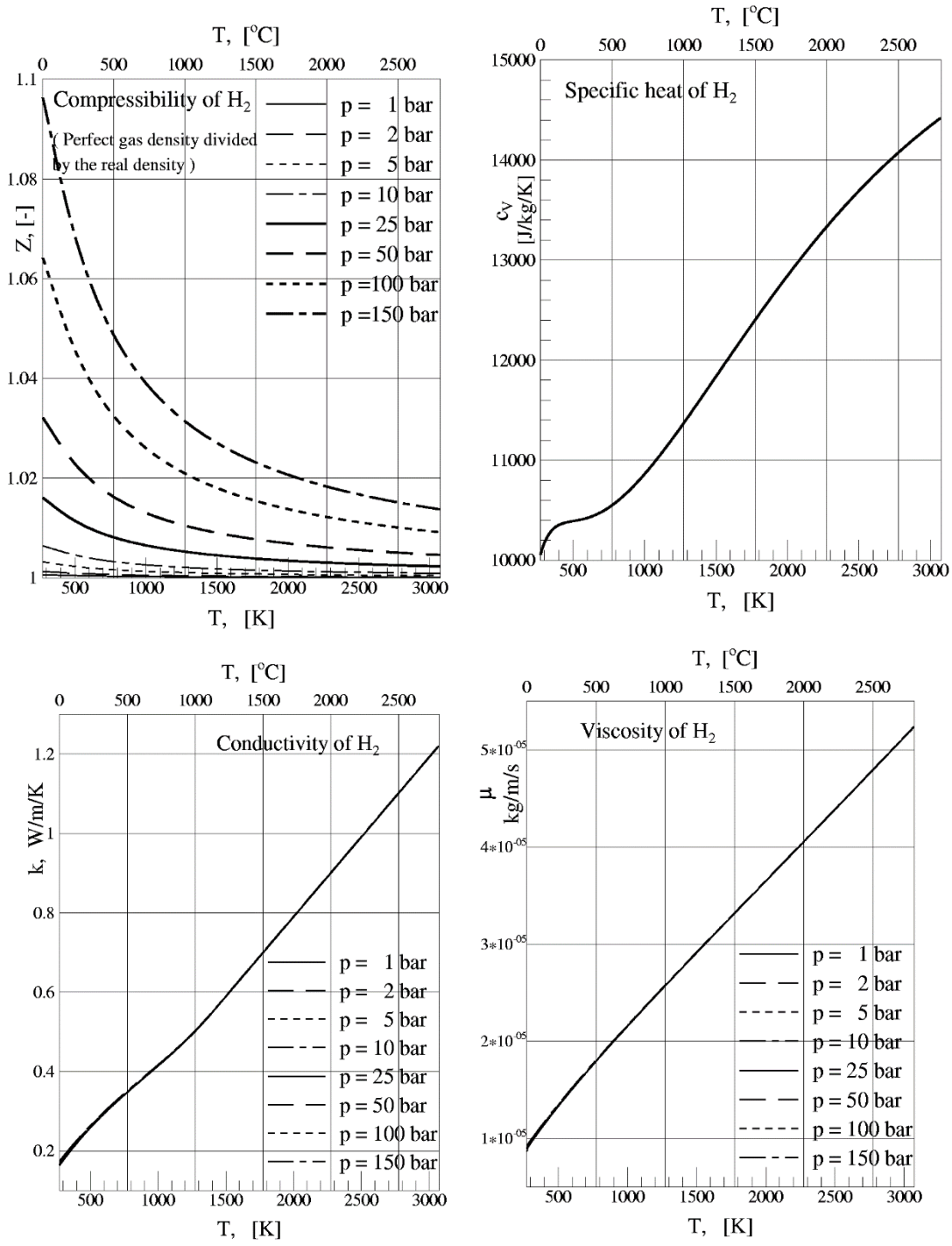


Figure 3-3 Properties of H_2.

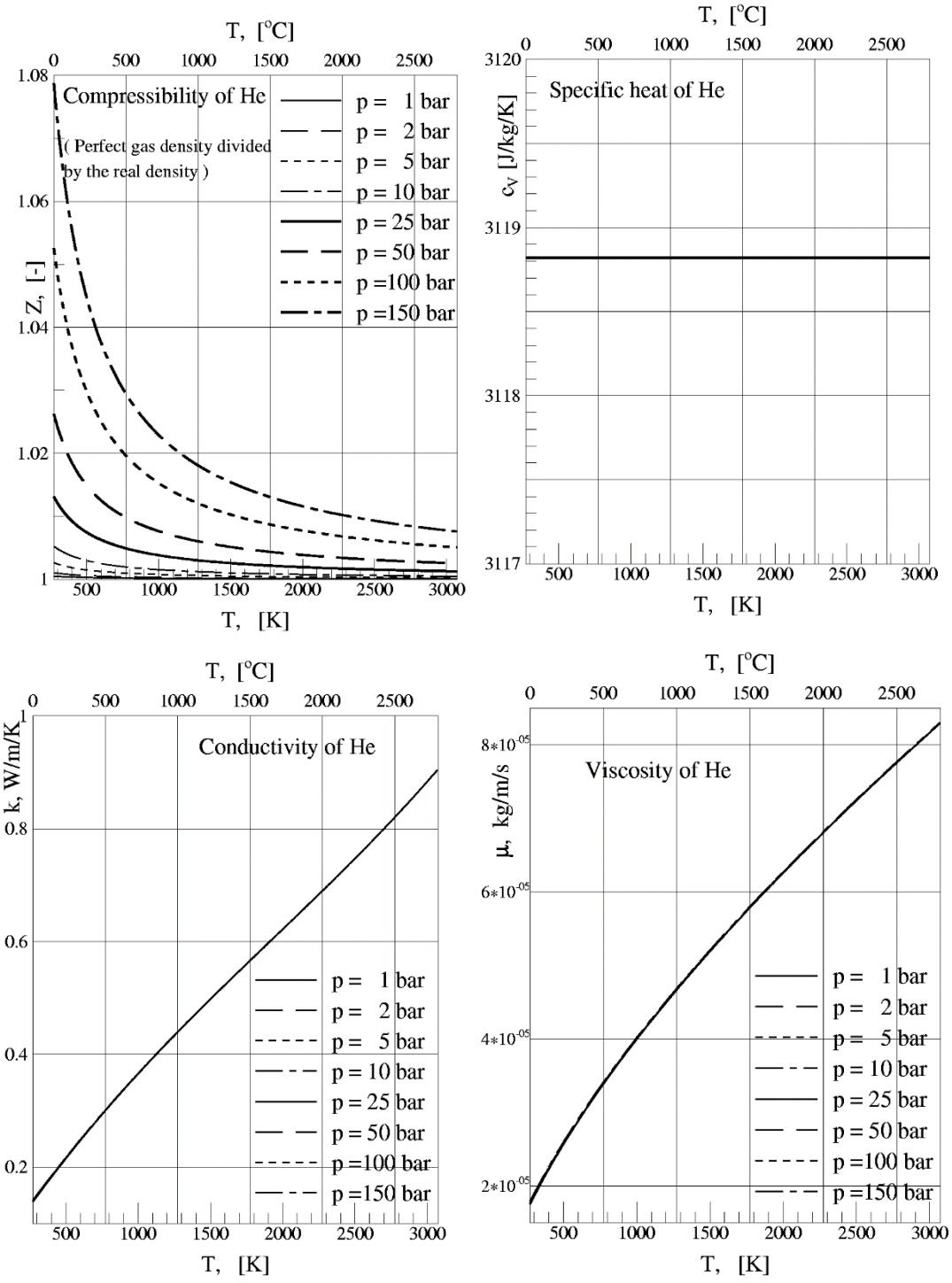


Figure 3-4 Properties of He.

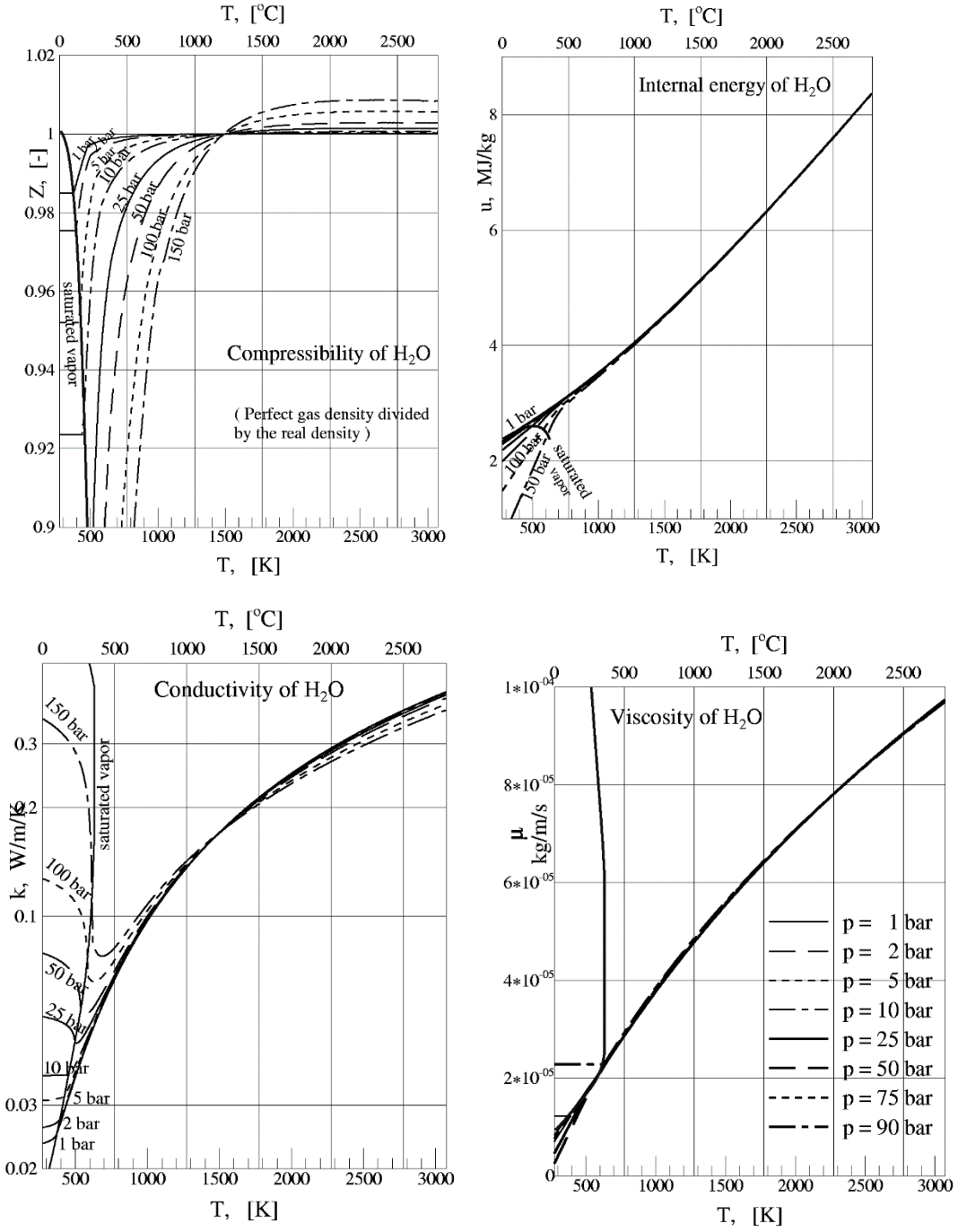


Figure 3-5 Properties of H_2O.

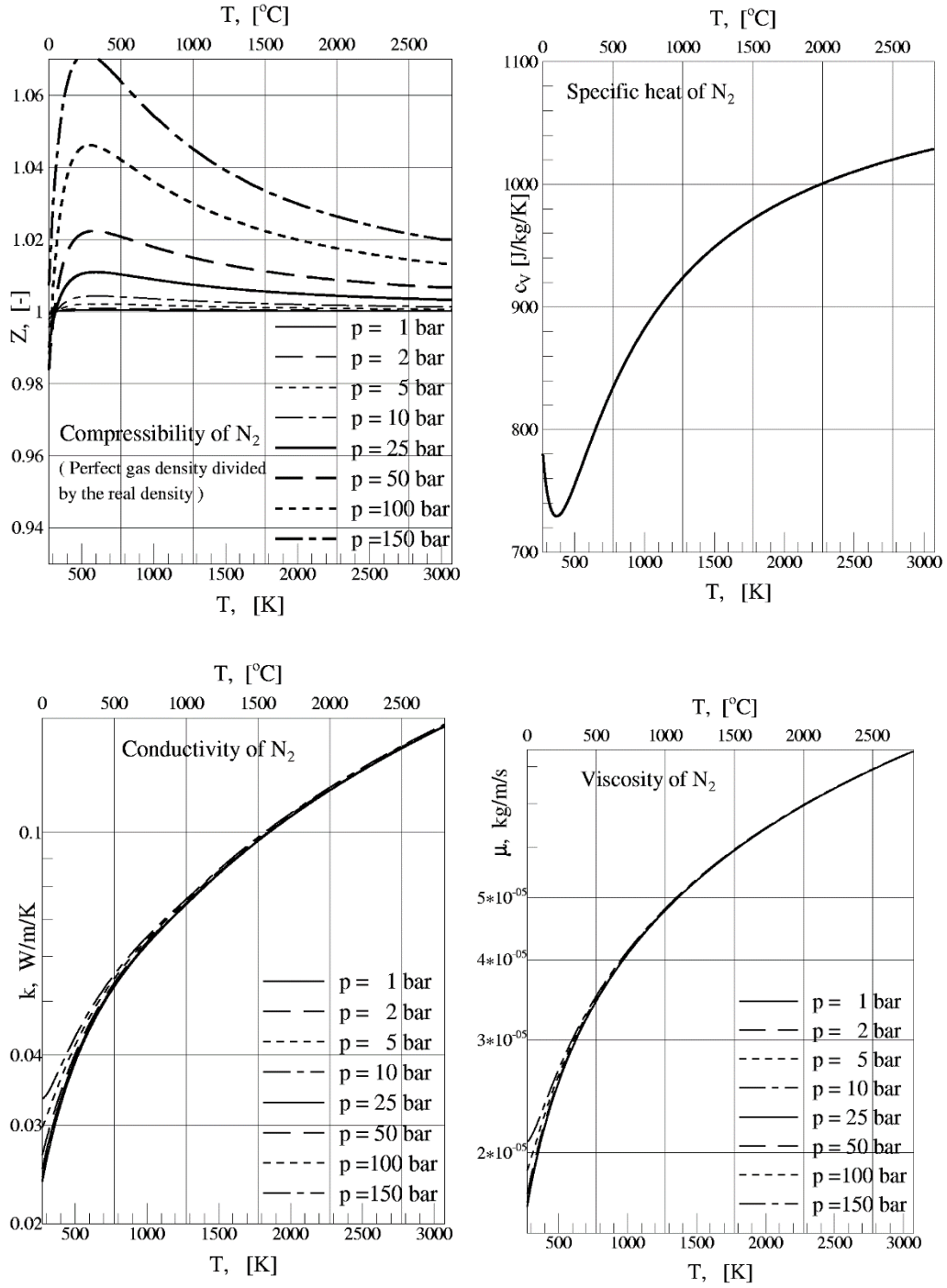


Figure 3-6 Properties of N_2.

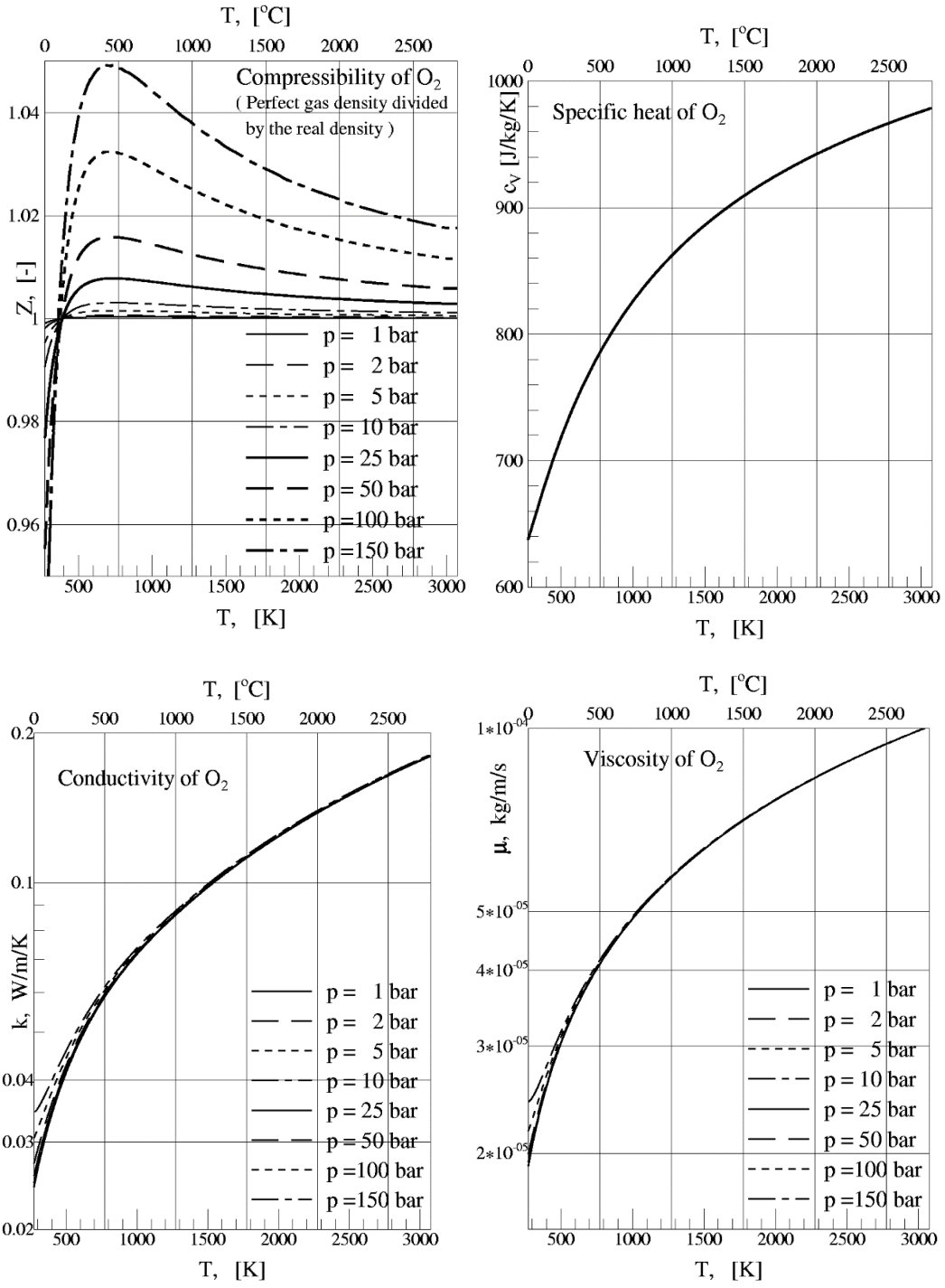


Figure 3-7 Properties of O_2.

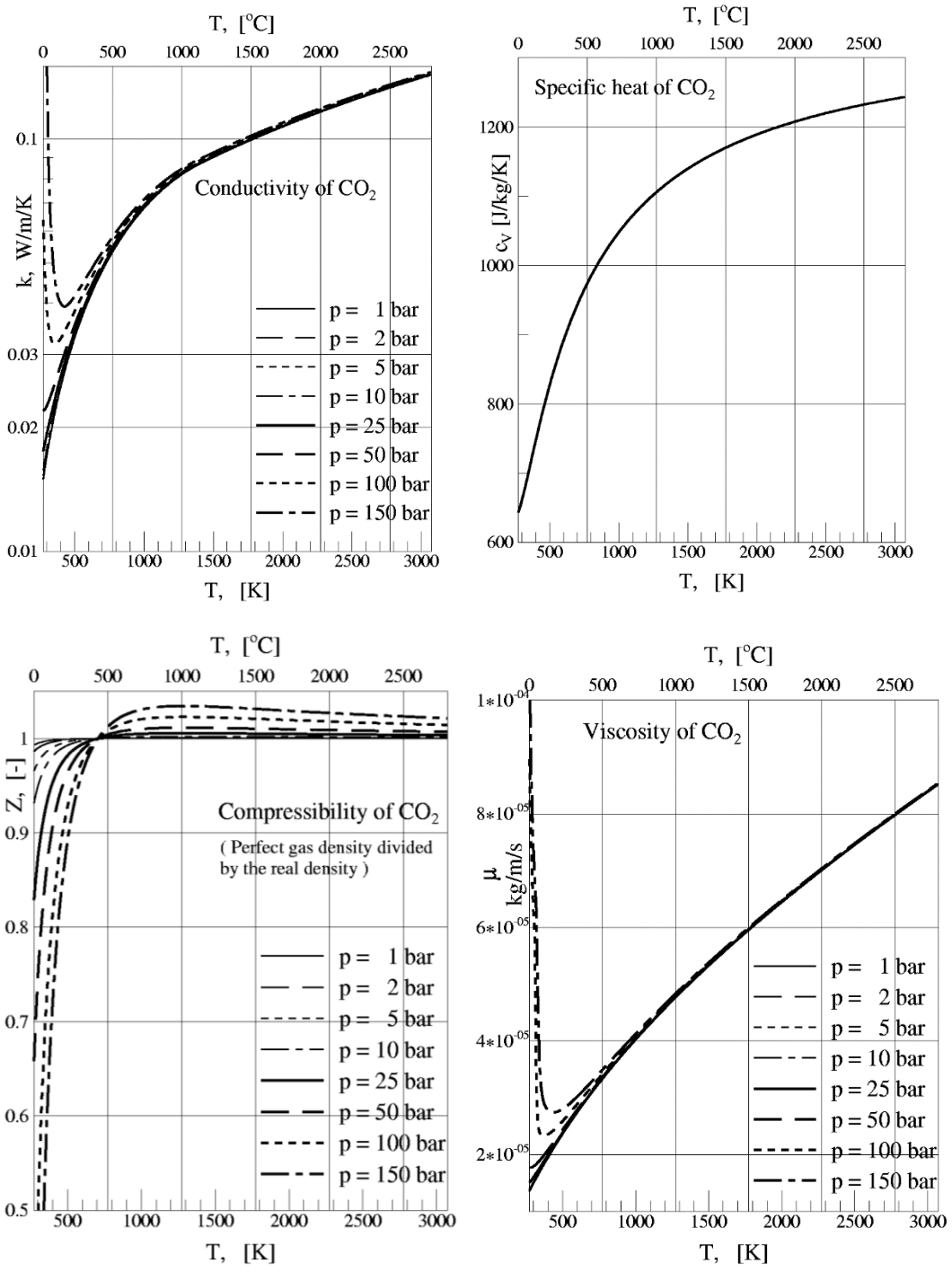


Figure 3-8 Properties of CO_2.

3.5.2 Calculation of Gas Properties

As mentioned above, the gas property subroutines are built based on the assumption that not only steam, but all gases are treated as real gases. One consequence of this assumption is that a simple relation between the gas partial pressure, p_k, and the molar fraction, x_k:

$$p_k = x_k \cdot p$$

is only approximately correct, and cannot be used to determine partial pressures of gases. The discussion below shows how the gas pressures are calculated from the basic definitions, using the gas property data tables described in the previous section.

The following definitions are assumed:

density of gas mixture	$\rho = M / V$
specific volume of gas mixture	$v = V / M = 1 / \rho$
mass fraction of gas k	$c_k = M_k / M$
density of gas k	$\rho_k = M_k / V = c_k \rho = c_k / v$

M_k mass of gas *k*, in given component (atmosphere or bubbles) in a Control Volume, (kg)
M total mass of gases in given component of a Control Volume, (kg)
V total volume of given component in a Control Volume, (m^3)

In the symbols used in this section the superscripts comp, referring to the component number, are skipped for simplicity. One has to remember that, since this section discusses the gas properties, the CV components to which it applies are the atmosphere and bubbles.

Note that the definitions stated above are general, and are valid whether the gas is ideal or not. The gas property data tables provides the relation between the gas density and the gas partial pressure, as:

$$\rho_k = \rho_k(p_k, T)$$

where p_k is the partial pressure of gas *k*, (Pa), and T is the temperature of gas mixture, (K). The meaning of the above equation is, that if temperature and all partial pressures are known, then the gas densities are obtained from the data tables, by a simple bi-linear interpolation between the tabulated points. (It should be noted that because of the shape of the density function: $\rho \sim p$; $\rho \sim 1/T$, the density is interpolated in the pressure "direction", while $1/\rho$ is interpolated in the temperature "direction").

Similarly the internal energy of the mixture can be obtained by a straightforward bi-linear interpolation, if temperature and all partial pressures are known:

$$u_k = u_k(p_k, T)$$

In practice one has to solve an inverse problem: the gas densities are known (since the masses are known from the mass conservation equation), and the internal energy is known (from the energy conservation equation), and one needs to find the "state" of the mixture, that is to determine temperature, all partial pressures, and finally all physical properties of the mixture. In such case an iterative solution is needed. As a first step the temperature is guessed. Use is made of the following relations:

$$u = \sum_{k=1}^{N_{gas}} c_k u_k \qquad u_k \approx \left(\partial u_k / \partial T\right) \cdot \left(T_G - T_{ref,k}\right)$$

T_G first guess of gas temperature, (K)
$T_{ref,k}$ reference temperature of gas *k*, (K)
u_k specific internal energy of gas *k*, (J/kg)

The above equations lead to the following formula for the first guess of the gas temperature:

$$T_G = \frac{u + \sum_{k=1}^{N_{gas}} c_k \cdot \frac{\partial u_k}{\partial T} \cdot T_{ref,k}}{\sum_{k=1}^{N_{gas}} c_k \cdot \frac{\partial u_k}{\partial T}}$$

As a next step, the total pressure and the individual gas partial pressures are guessed, using the perfect gas law, as:

$$p_G = \frac{RT_G}{v}$$

$$p_{k,G} = x_k p_G$$

p_G first guess of gas pressure, (Pa)
$p_{k,G}$ first guess of a partial pressure of gas *k*, (K)
R gas constant, (J/kg-K), obtained from: $R = c_k R_k$

With the guessed values of the gas partial pressures, the temperature of mixture is calculated. This is done using the following relation:

$$u = \sum_{k=1}^{N_{gas}} c_k u_k (p_{k,G}, T)$$

Since the internal energy, *u*, is known, the temperature is obtained from the above equation, by performing inverse interpolation of the internal energy tables.

With the calculated temperature and known gas densities, the partial pressures are calculated for each gas, by inverse interpolation of the density tables. The total pressure is then calculated as the sum of partial pressures. The calculated pressure is compared to the guessed value. If the relative difference between the calculated and assumed pressure is smaller than the convergence criterion, equal to 10^{-8}, the iteration is terminated. If not, the calculated values of partial pressures are assumed as the next guess, and the procedure continues, by re-calculating the gas temperature by performing inverse interpolation of the internal energy tables, etc. Usually 5 to 10 iterations are sufficient to obtain the desired accuracy.

To provide an easy way of obtaining gas properties for any combination of input parameters, which may be encountered during the calculation procedure, five subroutines are provided, which use different methods to calculate gas properties.

- Partial pressures and temperature as arguments (subroutine GASEQP):

  GASEQP Properties = f (p_k , T)

  In this case properties are obtained by a bi-linear interpolation in the gas property data base. The method is very fast, since no iteration is required. The method is used for example during input processing, to get the properties of gases for which temperature and composition have been specified in the input data.

- Specific volume, internal energy, and mass fractions as arguments (subroutine GASEQV):

  GASEQV Properties = f (v , u , c_k)

  In this case iterative calculation is required. The calculation procedure is described above. This is the most frequently used method during transient calculations. The arguments: v, and c_k, are obtained from the masses of all gases (using the definitions shown above). The masses are obtained from the mass balance. The energy balance provides the internal energy, u.

- Pressure, temperature, and mass fractions as arguments (subroutine GASEQT):

  GASEQT Properties = f (p , T , c_k)

  In this case iterative calculation is required. The calculation procedure is similar to the one described above. This subroutine is used frequently by the heat and mass transfer models, for example to obtain fluid properties at the film temperature, etc. It is also used for bubble property calculation, should GASEQV fail to obtain consistent results. The bubble component is the most difficult one to solve. As a result of huge overall area for heat and mass transfer, and practically no heat capacity, the equations for the bubble component are very "stiff". It sometimes happens that GASEQV fails to calculate bubble properties. In such case the subroutine GASEQT is used, as an emergency measure, to prevent time step cut and continue the iterations. The gas temperature, required by GASEQT, is set to the pool temperature (the bubble temperature is typically very close to the pool temperature).

- Pressure, internal energy, and mass fractions as arguments (Subroutine GASEQU):

  GASEQU Properties = f (p , u , c_k)

  This subroutine is used for the bubble thermodynamics calculation. During the first step in the thermodynamics calculation, the bubble properties are obtained from this subroutine, with the pressure determined by the current bubble submergence. During the final step the bubble properties are obtained in a "standard" way, from the GASEQV, described above.

- Specific volume, temperature, and mass fractions as arguments (Subroutine GASEQC):

  GASEQU Properties = f (v , T , c_k)

This subroutine is presently never used by the code, but has been built for completeness and for possible future use.

All five subroutines provide the same results, within an accuracy determined by the iteration convergence criteria, which is a relative error of 10^{-8}. This has been verified by extensive automatic testing of all five subroutines, in which a random number generator was used to provide input arguments.

The parameters that are returned by the five subroutines are as follows. First, three out of the six parameters listed below (the remaining three are input arguments):

Temperature, (K),	T
Pressure, (Pa),	p
Specific volume, (m^3/kg),	v
Specific internal energy, (J/kg),	u
Partial pressures of individual gases, (Pa),	p_k
Mass fractions of individual gases, (-),	c_k

Next, the following properties are obtained:

Density, (kg/m^3),	ρ
Enthalpy, (J/kg),	$h = u + p \times v$
Viscosity, (kg/m/s),	μ
Thermal conductivity, (W/m/K),	k
Isobaric specific heat, (J/kg/K),	$c_p = h / T$
Isochoric specific heat, (J/kg/K),	$c_v = u / T$
Prandtl number, (-),	$Pr = \mu\, c_p / k$
Thermal expansion coefficient, (1/K),	$\beta = 1/T$
Diffusion coefficient, (m^2/s),	D

The properties of a gas mixture are not always easily obtained from the individual gas values. In case of internal energy it is assumed that the value for the gas mixture is simply equal to the average weighted by mass fractions:

$$u = \sum_{k=1}^{N_{gas}} c_k u_k$$

In case of thermal properties of gases the above assumption cannot be used and special procedures are needed. Below three methods are described, used to calculate:

- Viscosity of gas mixture
- Thermal conductivity of gas mixture
- Diffusion coefficient in gas mixture

3.5.2.1 Viscosity of a Gas Mixture

The viscosity of a mixture of gases is calculated using the method of Wilke [33]:

$$\mu_{mix} = \sum_{i=1}^{N_{gas}} \frac{x_i \mu_i}{\sum_{j=1}^{N_{gas}} x_j \Phi_{ij}}$$

where:

$$\Phi_{ij} = \frac{\left[1 + \left(\mu_i / \mu_j\right)^{1/2} \left(M_{w,j} / M_{w,i}\right)^{1/4}\right]^2}{\sqrt{8} \cdot \left(1 + M_{w,i} / M_{w,j}\right)^{1/2}}$$

μ_{mix} viscosity of the gas mixture, (kg/m-s)
μ_i viscosity of the gas *i*, (kg/m-s)
x_i mole fraction of gas *i* in the mixture, (-)
$M_{w,i}$ molar weight of the gas *i*, (kg/kmole)

If the mole fractions, x_i, are replaced by the mass fractions, c_i, using the relation:

$$x_i = \frac{(c_i / M_{w,i})}{\sum_{j=1}^{N_{gas}} (c_j / M_{w,j})}$$

the equation becomes (see [46]):

$$\mu_{mix} = \sqrt{8} \cdot \sum_{i=1}^{N_{gas}} \frac{c_i \mu_i}{\sum_{j=1}^{N_{gas}} x_j \cdot \left(\frac{M_{w,i}}{M_{w,i} + M_{w,j}}\right)^{1/2} \cdot \left[\left(\frac{M_{w,i}}{M_{w,j}}\right)^{1/4} + \left(\frac{\mu_i}{\mu_j}\right)^{1/2}\right]^2}$$

This formula is used by the Fluid Property Package to calculate the dynamic viscosity of a gas mixture.

3.5.2.2 Thermal Conductivity of a Gas Mixture

The thermal conductivity of a mixture of gases is calculated using the relation proposed by Wassiljewa [34], modified by Mason and Saxena [36]:

$$k_{mix} = \sum_{i=1}^{N_{gas}} \frac{x_i k_i}{\sum_{j=1}^{N_{gas}} x_j \Phi_{ij}}$$

k_i is the individual thermal conductivity of gas *i*, and Φij is the same as above. Again the mole fractions are replaced by mass fractions and the final expression for the thermal conductivity of a mixture becomes [46]:

$$k_{mix} = \sqrt{8} \cdot \sum_{i=1}^{N_{gas}} \frac{c_i k_i}{\sum_{j=1}^{N_{gas}} x_j \cdot \left(\frac{M_{w,i}}{M_{w,i} + M_{w,j}} \right)^{1/2} \cdot \left[\left(\frac{M_{w,i}}{M_{w,j}} \right)^{1/4} + \left(\frac{\mu_i}{\mu_j} \right)^{1/2} \right]^2}$$

This formula is used by the Fluid Property Package to calculate thermal conductivity of a gas mixture.

3.5.2.3 Diffusion Coefficient of a Gas Mixture

The diffusion coefficient is calculated using the relation, sometimes called the Blanc's law, between the diffusion coefficient in the gas mixture and the binary diffusion coefficients [37], [31]. The equation is:

$$D_{C,i-mix} = \left(\sum_{\substack{j=1 \\ j \neq i}}^{N_{gas}} \frac{x_j}{D_{C,i-j}} \right)^{-1}$$

$D_{C,i\text{-}mix}$ diffusion coefficient of the gas *i* in the gas mixture, (m^2/s)
$D_{C,i\text{-}j}$ binary diffusion coefficient for the $i - j$ system, (m^2/s)
x_j mole fraction of gas *j* in the mixture, (-)

The method is general and may be used to calculate the diffusion coefficient for any gas present in the gas mixture. Currently the program calculates only diffusion coefficients for steam in the gas mixture, other coefficients are not needed. Steam diffusion coefficients are needed to calculate the Sherwood number, used in mass transfer correlations (see Chapter 7).

The binary diffusion coefficients are calculated using the method of Fuller et al. [38]. The original equation of Fuller is:

$$D_{C,i-j} = 0.00143 \cdot \frac{T^{1.75}}{p(\Sigma_i^{1/3} + \Sigma_j^{1/3})^2} \frac{1}{\sqrt{2}} \left(\frac{1}{M_{w,i}} + \frac{1}{M_{w,j}} \right)^{1/2}$$

The original equation was using diffusion coefficients in cm^2/s, pressure in bar and temperature in K. In order to apply pressure in Pa the right hand side of the formula must be multiplied by 10^5. In order to obtain diffusion coefficients in m^2/s the right hand side of the formula must be multiplied by 10^{-4}. Thus the conversion factor to SI units is 10. Including the square root of 2 gives the total multiplier of $10/2^{1/2} = 7.07$, and the final value of the coefficient is $0.00143 \times 7.07 = 1.011 \times 10^{-2}$. The final equation is:

$$D_{C,i-j} = 1.011 \times 10^{-2} \cdot \frac{T^{1.75}}{p(\Sigma_i^{1/3} + \Sigma_j^{1/3})^2} \left(\frac{1}{M_{w,i}} + \frac{1}{M_{w,j}} \right)^{1/2}$$

$D_{C,i\text{-}j}$ diffusion coefficient, (m^2/s)
T temperature, (K)
p pressure, (Pa)
$M_{w,i}$ molar weight of gas i, (kg/kmole)
Σ_i molecular diffusion volume of gas i, given in Table 3-5

Table 3-5 Molar weights and molecular diffusion volumes

Element	M_W	Σ	Compound	M_W	Σ
H_2	2.01588	6.12	N_2	28.0134	18.5
He	4.00260	2.67	O_2	31.9988	16.3
H_2O	18.0153	13.1	CO_2	44.0098	26.9

Further discussion of the binary diffusion coefficients is provided in section 3.6.4. There are other methods of calculating diffusion coefficients. For example temperature correlations are being used. The correlations have usually the form:

$$D_C = A \cdot \frac{T^B}{p}$$

In SPECTRA the method of Fuller and the Blanc's law is used because they offer a general formulation, which is applicable for any gas mixture, while the correlations are given for a particular gas mixture. The correlations are used for example by MELCOR (RN Package, reference [46]), where two correlations are available, one for the steam-air mixtures and one for the steam-hydrogen mixtures.

- Steam-air:

$$D_C = 4.7931 \times 10^{-5} \cdot \frac{T^{1.9}}{p}$$

- Steam-H_2:

$$D_C = 6.60639 \times 10^{-4} \cdot \frac{T^{1.68}}{p}$$

Results of these correlations are compared to the results of the method applied in SPECTRA in Figure 3-9 and Figure 3-10. It is seen that a very good agreement is obtained.

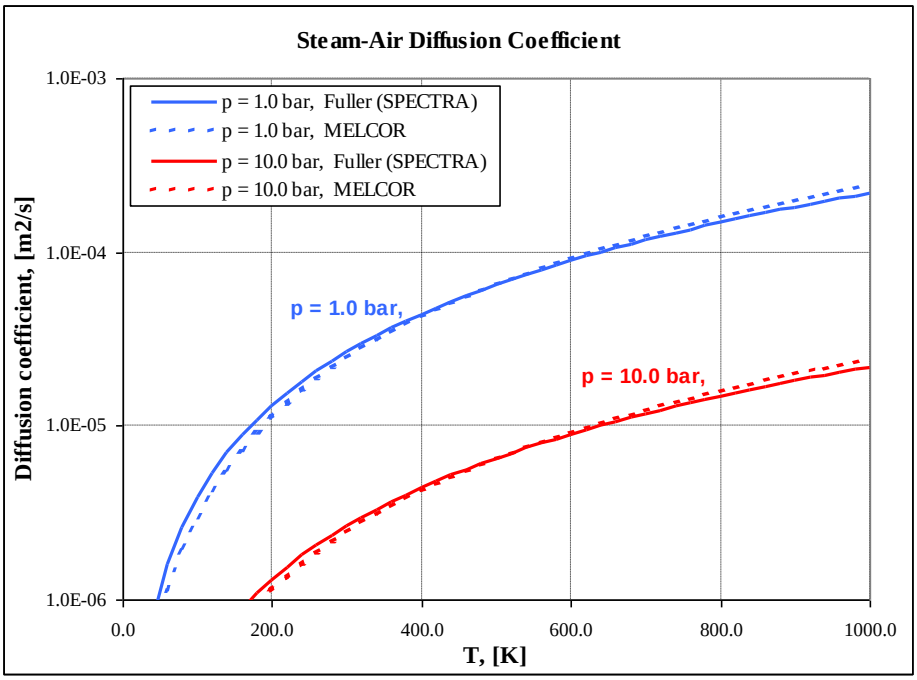


Figure 3-9 Comparison of steam-air diffusion coefficients.

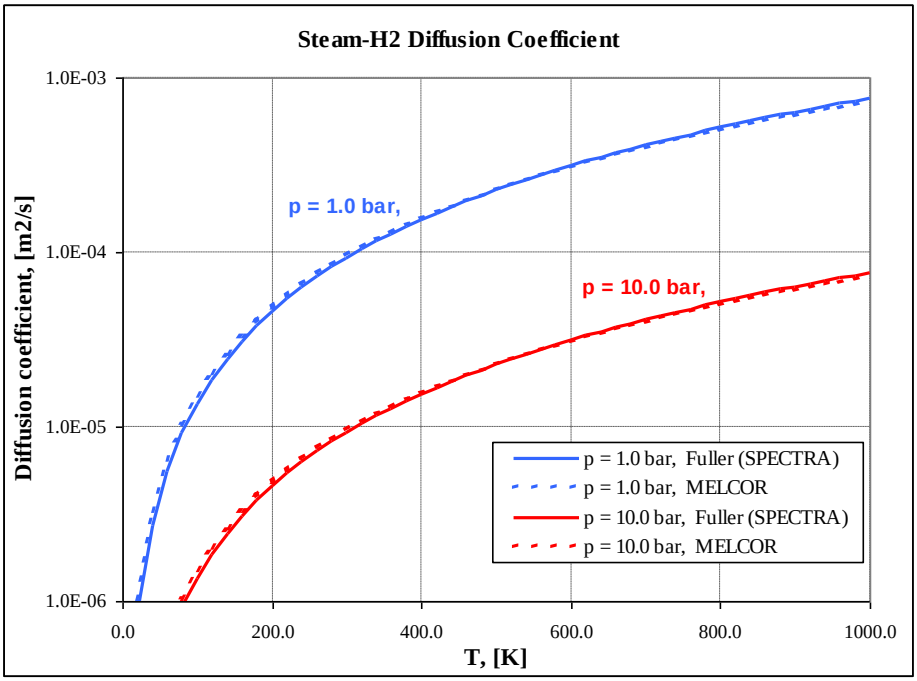


Figure 3-10 Comparison of steam-H_2 diffusion coefficients.

3.6 Gas Properties - User-Defined Gases

On top of the built-in gases, user-defined gases are available. The total number of gases (built-in + user-defined) is 20. The user-defined gases are treated as semi-perfect gases. The perfect gas equation of state is used. The specific heat, the viscosity, and the thermal conductivity, are assumed to be functions of temperature.

The user has to define the following parameters:

- Molar weight, M_w, (kg/kmol)
- Molecular diffusion volume, Σ, (-)
- Specific heat, $c_p(T)$, (J/kg-K)
- Viscosity, $\mu(T)$, (kg/m-s)
- Thermal conductivity, $k(T)$, (W/m-K)

The data is used by the program to compute the following parameters:

- Density, $\rho(T, p)$, (kg/m^3), section 3.6.1
- Internal energy, $u(T)$, (J/kg), section 3.6.2
- Dynamic viscosity, $\mu(T)$, (kg/m-s), section 3.6.3
- Thermal conductivity, $k(T)$, (W/m-K), section 3.6.3
- Diffusion coefficient, $D_C(T, p)$, section 3.6.4

3.6.1 Equation of State, $\rho(T, p)$

The ideal gas equation of state is used for all user-defined gases:

$$\rho = \frac{p}{RT}$$

ρ density, (kg/m^3)
p pressure, (Pa)
T temperature, (K)
R individual gas constant, (J/kg-K)

The individual gas constant is obtained from:

$$R = \frac{R_u}{M_w}$$

R_u universal gas constant (=8314.51), (J/kmol-K)
M_w gas molar weight, (kg/kmol)

3.6.2 Internal Energy, $u(T)$

The user has to provide a specific heat table, $c_p(T)$, and the value of the internal energy at the reference point, u_{ref} (T_0) (J/kg). The specific heat is assumed to be a function of temperature only (influence of pressure on internal energy of the user-defined gases is ignored). This is justified for most gases (see for example the specific heat of H_2 (Figure 3-3), He (Figure 3-4), N_2 (Figure 3-6), O_2 (Figure 3-7).

The reference point, T_0, is assumed to be the first point in the specific heat data table. The internal energy is calculated by integrating the specific heat data provided by the user, as:

$$u(T) = u_{ref} + \int_{T_0}^{T} [c_p(T') - R]\, dT'$$

R individual gas constant, (J/kg/K)

3.6.3 Thermal Properties, Viscosity, $\mu(T)$, Conductivity, $k(T)$

The thermal properties, dynamic viscosity and thermal conductivity, are calculated as functions of temperature (influence of pressure on the thermal properties of the user-defined gases is ignored). This is justified for most gases (see for example the viscosity and conductivity of H_2 (Figure 3-3), He (Figure 3-4), N_2 (Figure 3-6), O_2 (Figure 3-7).

The values of $\mu(T)$, $k(T)$, are obtained from linear interpolation of the tabulated data. Once the values for individual gases are obtained, the mixture properties are obtained from the equations shown in sections 3.5.2.1 and 3.5.2.2.

3.6.4 Binary Diffusion Coefficient, $D_C(T, p)$

The diffusion coefficient is calculated from (see section 3.5.2.3):

$$D_{C,i-j} = 1.011 \times 10^{-2} \cdot \frac{T^{1.75}}{p(\Sigma_i^{1/3} + \Sigma_j^{1/3})^2} \left(\frac{1}{M_{w,i}} + \frac{1}{M_{w,j}} \right)^{1/2}$$

$D_{C,i-j}$ diffusion coefficient, (m^2/s)
T temperature, (K)
p pressure, (Pa)
$M_{w,i}$ molar weight of gas i, (kg/kmol)
Σ_i molecular diffusion volume of gas i, given in Table 3-6. The data shown in Table 3-6 has been used to develop the following correlations that may be used for other materials:

- elements: $\Sigma \sim 1.3 \cdot M_w^{0.66}$
- compounds: $\Sigma \sim 2.5 \cdot M_w^{0.66}$

Results obtained with the above correlations are shown in Figure 3-11 against the data from Table 3-6. Once the values for the individual gases are obtained, the mixture properties are obtained from the equations shown in section 3.5.2.3.

Table 3-6 Molar weights and molecular diffusion volumes ([31] Table 11-1, [32] page 1-7)

Element	M_W	Σ	Compound	M_W	Σ
C	12.01	15.90	H_2	2.02	6.12
H	1.01	2.31	D_2	4.03	6.84
O	16.00	6.11	N_2	28.00	18.50
N	14.00	5.43	O_2	32.00	16.30
F	19.00	14.70	Air	28.01	19.70
Cl	35.45	21.00	CO	28.80	18.0
Br	79.90	21.90	CO_2	44.01	26.90
I	126.90	29.80	N_2O	44.00	35.90
S	32.06	22.90	NH_3	16.02	20.70
He	4.00	2.67	H_2O	18.02	13.10
Ne	20.18	5.98	SF_4	108.06	71.30
Ar	39.95	16.20	Cl_2	70.91	38.40
Kr	83.80	24.50	Br_4	159.80	69.00
Xe	131.30	32.70	SO_2	64.06	41.80

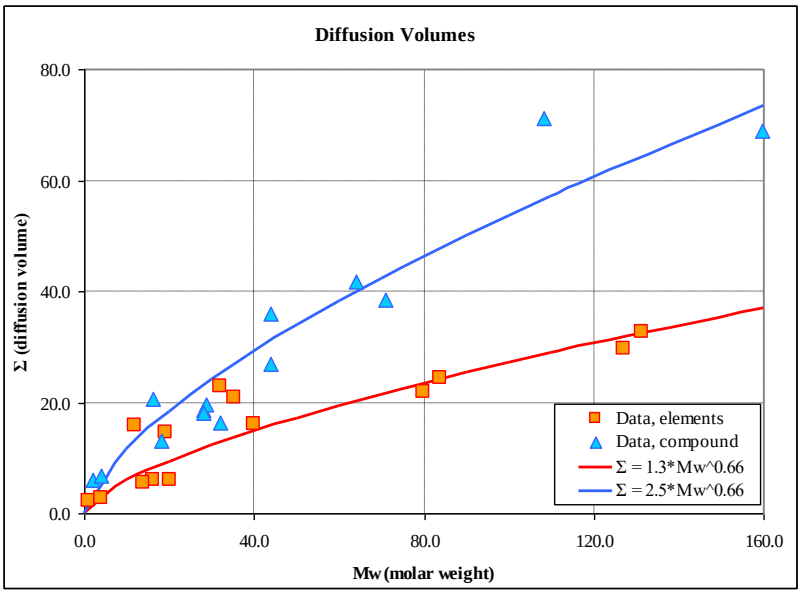


Figure 3-11 Diffusion volumes - data and correlations: $\Sigma = 1.3 \cdot M_w^{0.66}$, $\Sigma = 2.5 \cdot M_w^{0.66}$

4 Junction Package

4.1 Introduction

A Junction is a connection between two Control Volumes, through which fluids may be transported from one CV to another. No fluid volume is associated with a Junction itself. Consequently a Junction represents just an opening between two Control Volumes. Junctions are characterized by their flow area, flow direction, height (which is related to the flow direction), as well as momentum length, friction length etc. Since Junctions represent just openings, the bottom and top elevation of each junction must lie within the elevations of both Control Volumes which it connects. Examples of the junction connections are shown in Figure 4-1.

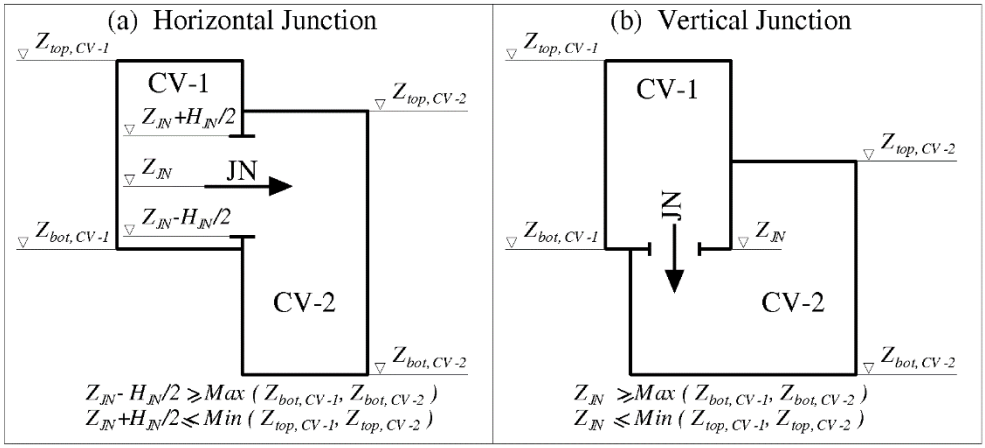


Figure 4-1 Junction geometry

This concept of a Junction is somewhat different than the concept of a Flow Path in the MELCOR code [46], where a Flow Path can be connected to the Control Volumes at different elevations. It is felt that, although the approach taken in MELCOR allows the user more flexibility in setting up a nodalization, it results in a somewhat non-physical treatment of fluid, which is transported instantaneously from one elevation to another (with no intermediate stage, since no mass is associated with a Flow Path). There are several implications of such treatment, which users are often unaware of, and which may lead to an inappropriate choice of the nodalization scheme. The present approach is considered safer and easier for physical interpretation.

Consistent with the treatment within Control Volumes, the CV Junction Package recognizes four components: atmosphere, droplets, pool, and bubbles. All of these components are transported through a junction. The velocities of those components are determined differently in case of continuous components (atmosphere, pool) and dispersed components (droplets, bubbles).

In case of continuous components a momentum equation is written for each component flowing through a junction. The flow solution is performed by pressure linearization in order to obtain the "flow matrix", which allows to find simultaneously the flows in all junctions. The flow solution based on this concept is commonly applied in many computer codes, for example MELCOR [46], MAAP [47], etc. The flow solution method is described in section 4.2. Several special type junctions are not included in the flow solution matrix. Those are junctions with user defined flow, junctions where critical flow occurs, and junctions for which flow is forced by pumps or fans. The treatment of such junctions is given below, in sections 4.4 and 4.6.

4.2 Junction Flows

The description of the junction flow solution is provided in three parts. In the first part, shown in section 4.2.1, the method used to calculate the flows of continuous components, atmosphere and pool, is described. Next, the method to determine velocities of the dispersed components, is described in section 4.2.3. Finally, calculation of a pool level in a junction, and the flow areas for all components, is discussed in section 4.2.4.

4.2.1 Flow Solution for Continuous Components - Method 1 - FFM

Before the flow equation is introduced, a short discussion is needed to explain how the pressure distribution within a Control Volume is calculated. As mentioned in section 2.3, pressures in a Control Volume are defined at the pool-atmosphere interphase. The pressure at any other elevation, for example Z meters above the reference point, is typically obtained by taking into account the gravity head of the fluid column:

$$p(Z) = p_{ref} - g \cdot \rho(p_{ref}) \cdot Z$$

where the subscript ref means value at the reference point (pool-atmosphere interphase, or CV bottom in absence of pool). In case of gas flows, the flows occur typically at very small pressure differences. In such case the formulation shown above may cause artificial circulation flows. Such flows may occur for example in case of two volumes, connected by two junctions at different elevations, as shown in Figure 4-2.

In the case shown in Figure 4-2 (a) the bottom elevations of CV-1 and CV-2 are the same, and no flow occurs. However, if the bottom elevations are different, as in Figure 4-2 (b), then the pressures (defined in absence of pool at CV bottom) are slightly different in stable conditions ($p_{CV\text{-}1} > p_{CV\text{-}2}$). This results in slightly different densities ($\rho_{CV\text{-}1} > \rho_{CV\text{-}2}$), and finally different pressure gradients: $(dp/dZ)_{CV\text{-}1} = g\rho_{CV\text{-}1} > (dp/dZ)_{CV\text{-}2} = g\rho_{CV\text{-}2}$. This fact will cause circulation flow from CV-1 through JN-1 to CV-2, and back through JN-2, which will not decrease in time.

The circulation flows described above are known as "phantom circulation flows" and have been observed also in the MAAP code [47].

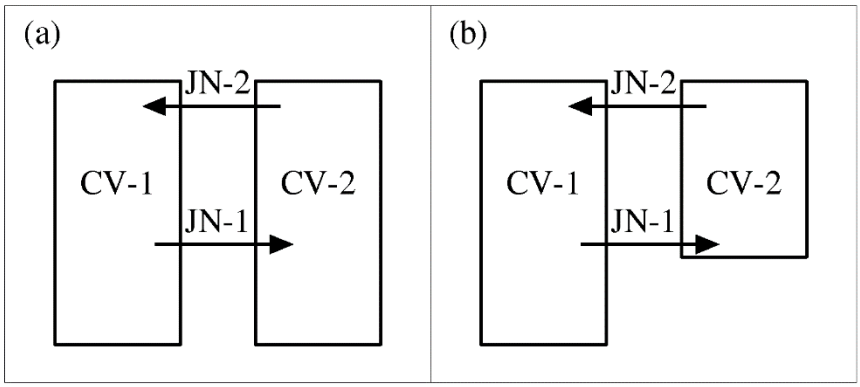


Figure 4-2 Nodalization for phantom circulation flow test

The solution suggested in MAAP Manuals is the use of the following differential equation:

$$\frac{dp(Z)}{dZ} = g \cdot \rho(p_{ref})$$

Which for the ideal gas has the following solution:

$$p(Z) = p_{ref} \cdot \exp\left(- \frac{g \cdot \rho(p_{ref}) \cdot Z}{p_{ref}} \right)$$

This formulation is used in the SPECTRA code to determine the pressure change due to atmospheric static gravity head in a CV. The formulation allows to minimize the phantom circulation flows.

The general concept of calculating the pressure correction due to static gravity heads for atmosphere and pool is illustrated in Figure 4-3. The elevation change is always taken between the reference point, and the center of that part of the junction, which is within the given component (atmosphere or pool). In case of atmosphere, the equation shown above is used. In case of pool, a simple form: $g\rho Z$, is used, in which the density is taken as an average of pool and bubbles. The final equations for the pressures at the junction elevation, in the upstream and downstream Control Volumes are:

$$p_U^{atms} = p_{CV,U}^{atms} \cdot \exp\left(- \frac{g \cdot \rho_{CV,U}^{atms} \cdot Z_U^{atms}}{p_{CV,U}^{atms}} \right)$$

$$p_D^{atms} = p_{CV,D}^{atms} \cdot \exp\left(- \frac{g \cdot \rho_{CV,D}^{atms} \cdot Z_D^{atms}}{p_{CV,D}^{atms}} \right)$$

$$p_U^{pool} = p_{CV,U}^{pool} + g \cdot \left(\rho_{CV,U}^{pool} (1 - \alpha_{CV,U}^{bubb}) + \rho_{CV,U}^{bubb} \alpha_{CV,U}^{bubb} \right)$$

$$p_D^{pool} = p_{CV,D}^{pool} + g \cdot \left(\rho_{CV,D}^{pool} (1 - \alpha_{CV,D}^{bubb}) + \rho_{CV,D}^{bubb} \alpha_{CV,D}^{bubb} \right)$$

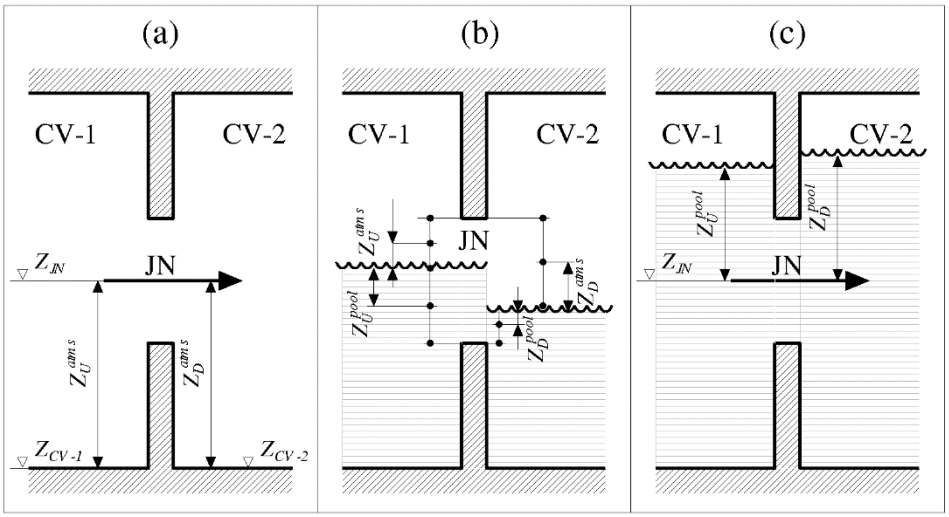


Figure 4-3 Fluid pressure at junction elevation

In the above equations subscripts *U* and *D* are used to indicate the Control Volumes upstream and downstream of junction JN respectively. Note that the reference pressures for atmosphere and pool of a Control Volume are the same (section 2.3). Therefore:

$$p_{CV,U}^{pool} = p_{CV,U}^{atms}$$
$$p_{CV,D}^{pool} = p_{CV,D}^{atms}$$

The equations governing the flow of atmosphere and pool through junctions are derived from the basic one-dimensional equation of motion:

$$\frac{d(mv)}{dt} = \sum F$$

where m is fluid mass, v is velocity, and Σ*F* is the sum of all forces acting on fluid. The equation is written for a fluid volume: *AL*, where *A* is the flow area (m^2), and *L* is the momentum (or inertial) length (m). The momentum length and the total area of junction are input parameters. A discussion on defining the momentum length is provided in section 4.5. The value of the area for a given fluid component may change in time, as a consequence of a varying water level in the junction. If the derivative, d*A*/d*t*, can be neglected, then the equation is written as:

$$A\rho L\frac{dv}{dt} = \sum F$$

Three forces are taken into account in the equation. These are pressure force, wall friction, and interphase friction.

$$\sum F = F_{pres} + F_{fric} + F_{int}$$

The individual terms are equal to:

- Pressure force

$$F_{pres} = A(p_U - p_D)$$

where p_U and p_D are the upstream pressure and downstream pressures, equal to the reference pressures in the upstream and downstream CV's, corrected for the static gravity head, as shown above.

- Wall friction

$$F_{fric} = -A\left(K\frac{\rho v|v|}{2} + f_{wall}\frac{L_f}{D_f}\frac{\rho v|v|}{2} \right)$$

K form loss factor, (-), either constant or given by a Control Function
f_{wall} wall friction factor, (-), calculated as described in sections 4.3.1, 4.3.2
ρ density, (kg/m^3)
D_f diameter, (m), used for friction calculation, an input value for the code, which in general may be different than the "physical" diameter, which is used to determine bottom and top elevations
L_f length, (m), used for friction calculation, an input value for the code, which in general may be different than the inertial length, L. A discussion on defining the friction length and the inertial length is provided in section 4.5.

- Interphase friction

$$F_{int} = -A\left(f_{int}\frac{L_f}{D_f}\frac{\rho|v - v^{-\phi}|(v - v^{-\phi})}{2} \right)$$

f_{int} interphase friction factor, (-), calculated as described in section 4.3.3
$v^{-\phi}$ velocity of component with different phase (pool in the equation for atmosphere flow, and atmosphere in the equation for pool flow), (m/s)

Finally, the momentum equations for atmosphere and pool flows in Junction No. i, are:

$$\rho_i^{comp} L_i \frac{dv_i^{comp}}{dt} = p_{U,i}^{comp} - p_{D,i}^{comp} - C_{f,i}^{comp} v_i^{comp} - C_{int,i}^{comp}(v_i^{comp} - v_i^{-comp})$$

where:

$$C_{f,i}^{comp} = \left(K_i^{comp} + f_{wall,i}^{comp}\frac{L_{f,i}}{D_{f,i}} \right) \cdot \frac{\rho_i^{comp}|v_i^{comp}|}{2}$$

$$C_{int,i}^{comp} = f_{wall,i}^{comp}\frac{L_{f,i}}{D_{f,i}} \cdot \frac{\rho_i^{atms}|v_i^{comp} - v_i^{-comp}|}{2}$$

In the above equations the meaning of the superscript *comp* is atmosphere and pool, while the meaning of superscript *–comp* is pool and atmosphere respectively.

The momentum equations provide up to $2N_{JN}$ equations, where N_{JN} is the total number of junctions. In practice the number of equations is smaller, since not all junctions have simultaneous flow of atmosphere and pool, and some of the junctions (junctions with user specified flow, junctions with critical flow) are treated differently. Of course, since phenomena like two phase flow in a junction and the occurrence of critical flow are time dependent, also the total number of equations determining the flow solution varies in time.

The method of solving the flow equations is based on pressure linearization which leads to, what is called in SPECTRA a "Full Flow Matrix" (FFM), and is shortly described below. Control Volume pressures are linearized, by introducing pressure derivatives, as:

$\left(\frac{\partial p}{\partial M}\right)_{m,CV}^{comp}$ pressure change of the component *comp* in CV caused by the mass source *m*

$\left(\frac{\partial p}{\partial E}\right)_{n,CV}^{comp}$ pressure change of the component *comp* in CV caused by the energy source *n*

Using the pressure derivatives, the end-of-time-step pressure in control volume CV, can be expanded as:

$$p_{CV}^{comp} = \overline{p}_{CV}^{comp} + \sum_m \left(\frac{\partial p}{\partial M}\right)_{m,CV}^{comp} \cdot W_{m,CV}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E}\right)_{n,CV}^{comp} \cdot Q_{n,CV}^{comp} \Delta t$$

Here $\overline{p}_{CV}^{comp}$ is the the pressure at the beginning of the time step. The sums are over all mass sources and energy sources that may be present in a Control Volume.

The first term in the sum of the mass sources represents the junction flow. This term is written separately as:

$$\left(\frac{\partial p}{\partial M}\right)_{m=1,CV}^{comp} \cdot W_{m=1,CV}^{comp} \Delta t = \sum_{j \in CV} \left(\frac{\partial p}{\partial M}\right)_{m=1,CV}^{comp} \cdot \sigma_{j,CV} A_j^{comp} \rho_j^{comp} v_j^{comp} \Delta t$$

where the sum is taken over all junctions *j*, which are connected to, or from, the control volume CV. The symbol $\sigma_{j,CV}$ is defined as follows:

$$\sigma_{j,CV} = \begin{cases} -1 & \textit{if the volume CV is the "from CV" for the junction } j \\ +1 & \textit{if the volume CV is the "to CV" for the junction } j \end{cases}$$

The pressures are linearized for the upstream CV (*CV=U*), and the downstream CV (*CV=D*), and substituted into the momentum equation. The momentum equation becomes:

$$\rho_i^{comp} L_i \frac{v_i^{comp} - \bar{v}_i^{comp}}{\Delta t} =$$

$$= \left[\bar{p}_{U,i}^{comp} + \sum_{j \in U} \left(\frac{\partial p}{\partial M} \right)_{m,U}^{comp} \sigma_{j,U} A_j^{comp} \rho_j^{comp} v_j^{comp} \Delta t + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,U}^{comp} W_{m,U}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,U}^{comp} Q_{n,U}^{comp} \Delta t \right] -$$

$$- \left[\bar{p}_{D,i}^{comp} + \sum_{j \in D} \left(\frac{\partial p}{\partial M} \right)_{m,D}^{comp} \sigma_{j,D} A_j^{comp} \rho_j^{comp} v_j^{comp} \Delta t + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,D}^{comp} W_{m,D}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,D}^{comp} Q_{n,D}^{comp} \Delta t \right] -$$

$$- \quad C_{f,i}^{comp} v_i^{comp} \quad - \quad C_{int,i}^{comp} (v_i^{comp} - v_i^{-comp})$$

Note that in the above equation the subscripts *U* and *D*, mean everywhere the Control Volumes upstream and downstream of the junction *i*, and not the junction *j*.

As may be easily noticed, the above equation defines a system of linear equations, with unknown junction component velocities, v_i^{comp}. To obtain the formula for the matrix elements, the equation is re-written, by grouping all terms dependent on v_i^{comp} on the left hand side, and all other terms on the right hand side. Additionally the whole equation is divided by: $(\rho_i^{comp} L_i / \Delta t)$, and a use is made of the following relation:

$$\sigma_{i,U} = -1$$

$$\sigma_{i,D} = +1$$

These relations are valid only for junction *i*, since the subscripts *U* and *D* represent always Control Volumes upstream and downstream the junction *i*. The resulting equation has the form:

$$\left\{ 1 + \frac{\Delta t}{\rho_i^{comp} L_i} (C_{f,i}^{comp} + C_{int,i}^{comp}) + \frac{\Delta t}{\rho_i^{comp} L_i} A_i^{comp} \rho_i^{comp} \Delta t \left[\left(\frac{\partial p}{\partial M} \right)_U^{comp} + \left(\frac{\partial p}{\partial M} \right)_D^{comp} \right] \right\} v_i^{comp} -$$

$$- \frac{\Delta t}{\rho_i^{comp} L_i} \sum_{\substack{j \in U \\ j \neq i}} \left(\frac{\partial p}{\partial M} \right)_{m,U}^{comp} \sigma_{j,U} A_j^{comp} \rho_j^{comp} v_j^{comp} \Delta t + \frac{\Delta t}{\rho_i^{comp} L_i} \sum_{\substack{j \in D \\ j \neq i}} \left(\frac{\partial p}{\partial M} \right)_{m,D}^{comp} \sigma_{j,D} A_j^{comp} \rho_j^{comp} v_j^{comp} \Delta t =$$

$$= \bar{v}_i^{comp} +$$

$$+ \frac{\Delta t}{\rho_i^{comp} L_i} \left[\bar{p}_{i,U}^{comp} + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,U}^{comp} W_{m,U}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,U}^{comp} Q_{n,U}^{comp} \Delta t \right] -$$

$$- \frac{\Delta t}{\rho_i^{comp} L_i} \left[\bar{p}_{i,D}^{comp} + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,D}^{comp} W_{m,D}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,D}^{comp} Q_{n,D}^{comp} \Delta t \right] +$$

$$+ C_{int,i}^{comp} v_i^{-comp}$$

This equation set may be written shortly in a matrix form:

$$\mathcal{A} \boldsymbol{v} = \boldsymbol{B}$$

where $\boldsymbol{A}$ is a "junction flow matrix", consisting of the elements a_{ij}, described below, $\boldsymbol{v}$ is the vector of junction component velocities, v_i^{comp}, and $\boldsymbol{B}$ is the right hand side vector, with elements b_i, described below.

The elements of the junction flow matrix $\boldsymbol{A}$, a_{ij}, are equal to:

- $i = j$:

$$a_{ij} = \left\{ 1 + \frac{\Delta t}{\rho_i^{comp} L_i} (C_{f,i}^{comp} + C_{int,i}^{comp}) + \frac{\Delta t}{\rho_i^{comp} L_i} A_i^{comp} \rho_i^{comp} \Delta t \left[\left(\frac{\partial p}{\partial M} \right)_U^{comp} + \left(\frac{\partial p}{\partial M} \right)_D^{comp} \right] \right\}$$

- $i \neq j$:

$$a_{ij} = \frac{\Delta t}{\rho_i^{comp} L_i} A_j^{comp} \rho_j^{comp} \Delta t \left[- \left(\frac{\partial p}{\partial M} \right)_U^{comp} \sigma_{j,U} + \left(\frac{\partial p}{\partial M} \right)_D^{comp} \sigma_{j,D} \right]$$

The elements of the right hand side vector, $\boldsymbol{B}$, are equal to:

$$\begin{aligned} b_i = & \bar{v}_i^{comp} + \\ & + \frac{\Delta t}{\rho_i^{comp} L_i} \left[\bar{p}_{i,U}^{comp} + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,U}^{comp} W_{m,U}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,U}^{comp} Q_{n,U}^{comp} \Delta t \right] - \\ & - \frac{\Delta t}{\rho_i^{comp} L_i} \left[\bar{p}_{i,D}^{comp} + \sum_{m>1} \left(\frac{\partial p}{\partial M} \right)_{m,D}^{comp} W_{m,D}^{comp} \Delta t + \sum_n \left(\frac{\partial p}{\partial E} \right)_{n,D}^{comp} Q_{n,D}^{comp} \Delta t \right] + \\ & + C_{int,i}^{comp} \bar{v}_i^{comp} \end{aligned}$$

The junction flow matrix is a typical, diagonally dominant matrix, with relatively few non-zero elements ("sparse matrix"). In the early versions of SPECTRA an iterative method - the bi-conjugate gradient method (section 17.4) - was used if the number of elements is large. Currently a non-iterative standard matrix solver (section 17.4) is applied. It was found out that the non-iterative solvers (Gauss-Jordan elimination or LU decomposition) methods are safer; that means they always finds a solution, while the bi-conjugate method sometimes fails to obtain the solution.

It should be mentioned that in the expression defining the elements of matrix $\boldsymbol{A}$ and vector $\boldsymbol{B}$, the values of the pressure derivatives are calculated using the beginning of time step conditions, while the mass and energy transfer terms: W_m^{comp}, Q_n^{comp}, are those at the end of time step, calculated in the main iteration to obtain implicit solution. This was done because it has been observed that the use of explicit (old time step) values for pressure derivatives, results in faster and better convergence of the flow solution. No mathematical proof, or any explanation can be given why this should be the case. This is a simple observation based on testing calculation.

The method of calculating pressure derivatives is based on the assumption that gases may be treated as ideal gases. The complexity of the possible conditions that may be encountered during calculations lead to the fact that quite a large number of different formulae are used to calculate the pressure derivatives. In particular, each of the mass and energy sources, described in section 2.4, has its own pressure derivatives, defined for each of the four components. Below two simple cases are shown, to demonstrate the general method used to obtain the pressure derivatives.

- Pressure derivative for liquid mass sources or sinks ($\partial p / \partial M_{liq}$)

As a first example a derivative is considered which is associated with a positive or negative source of liquid, for example in case of a tabular mass source. The pressure change in Control Volume is caused by a changing volume of gas, V_{gas}, as a result of liquid mass source, M_{liq}. The pressure derivative can be therefore written as:

$$\left(\frac{\partial p}{\partial M_{liq}}\right)=\left(\frac{\partial p}{\partial V_{gas}}\right)\cdot\left(\frac{\partial V_{gas}}{\partial M_{liq}}\right)$$

To calculate the first derivative, the perfect gas equation of state is used:

$$p=\frac{M_{gas}R_{gas}T_{gas}}{V_{gas}}$$

where R_{gas} is the individual gas constant, and T_{gas} is the gas temperature. Differentiation with respect to the gas volume, V_{gas}, with $T_{gas} \approx$ const., leads to:

$$\left(\frac{\partial p}{\partial V_{gas}}\right)=-\frac{M_{gas}R_{gas}T_{gas}}{V_{gas}^2}$$

The second derivative is obtained simply by observing that the change in gas volume is equal to the change in the water volume, taken with the negative sign: $(\partial V_{gas}/\partial M_{liq}) = -(\partial V_{liq}/\partial M_{liq})$. The change of the water volume, caused by adding M_{liq} kg of water, is equal to: M_{liq}/ρ_{liq}, so that: $(\partial V_{liq}/\partial M_{liq})=1/\rho_{liq}$. Therefore the derivative of the gas volume is equal to:

$$\left(\frac{\partial V_{gas}}{\partial M_{liq}}\right)=-\frac{1}{\rho_{liq}}$$

The full derivative is equal to:

$$\left(\frac{\partial p}{\partial M_{liq}}\right)=\left(-\frac{M_{gas}R_{gas}T_{gas}}{V_{gas}^2}\right)\cdot\left(-\frac{1}{\rho_{liq}}\right)=\frac{M_{gas}R_{gas}T_{gas}}{V_{gas}^2\rho_{liq}}$$

or finally:

$$\left(\frac{\partial p}{\partial M_{liq}}\right)=\frac{p}{V_{gas}\rho_{liq}}$$

The above formula defines pressure derivatives for the atmosphere component. In case of pool component, the formula is a little more complex. In that case one has to know whether the junction for which pressure derivative is written, is partly or fully below the water level. If that is the case, then the mass source of liquid results in an increasing pressure for pool flow, because of increasing the submergence of the junction. The additional term, caused by the hydrostatic head is given by:

$$\left(\frac{\partial p_{hydr}}{\partial M_{liq}}\right)=\left(\frac{\partial p_{hydr}}{\partial Z}\right)\cdot\left(\frac{\partial Z}{\partial M_{liq}}\right)$$

The first derivative is determined as:

$$p_{hydr}=\rho_{liq}gZ \quad \Rightarrow \quad \left(\frac{\partial p_{hydr}}{\partial Z}\right)=\rho_{liq}g$$

The second derivative is:

$$M_{liq}=A^{pool}Z\rho_{liq} \quad \Rightarrow \quad \left(\frac{\partial Z}{\partial M_{liq}}\right)=\frac{1}{A^{pool}\rho_{liq}}$$

Therefore the hydrostatic derivative is:

$$\left(\frac{\partial p_{hydr}}{\partial M_{liq}}\right)=\left(\rho_{liq}g\right)\cdot\left(\frac{1}{A^{pool}\rho_{liq}}\right)=\frac{g}{A^{pool}}$$

The full derivative for partially or fully submerged junctions is:

- JN below the water level:

$$\left(\frac{\partial p}{\partial M_{liq}}\right)=\frac{p}{V_{gas}\rho_{liq}}+\frac{g}{A^{pool}}$$

- JN above the water level:

$$\left(\frac{\partial p}{\partial M_{liq}}\right)=\frac{p}{V_{gas}\rho_{liq}}$$

Note that even if the junction is above water level in the given CV, the derivative may still be needed, because the water level in the other CV may be high, and consequently pool flow through the junction will occur.

- Pressure derivative for gas mass sources or sinks ($\partial p/\partial M_{gas}$)

First, consider a negative gas source. Differentiation of the perfect gas equation of state gives:

$$\left(\frac{\partial p}{\partial M_{gas}}\right)=R_{gas}\frac{T_{gas}}{V_{gas}}=\frac{pV_{gas}}{M_{gas}T_{gas}}\frac{T_{gas}}{V_{gas}}=\frac{p}{\rho_{gas}V_{gas}}$$

In the above equation the effect of temperature change on pressure has been neglected, so that the derivative shown above is obtained for isothermal conditions. Theoretically the temperature effect could be taken into account by considering adiabatic conditions, rather than isothermal. This however would not improve the solution procedure, on the contrary it would make the solution worse. The reason is that the pressure change in a multi-compartment system is relatively slow. Typically inter-compartment flows of practically incompressible gas is observed. In other words, the gas disappearing in one negative source is replaced by approximately the same volume of gas, coming from other sources. Therefore the isothermal

assumption results in better pressure projection and faster convergence. This behavior has been observed in multiple test calculations.

Next, consider a positive source of gas. This case is slightly more complex, since now the incoming gas may have a different temperature, and also a different composition (for example hydrogen flowing into a volume filled with air), and the pressure derivative is affected by the enthalpy of the incoming gas. In this case the pressure derivative is given by:

$$\left(\frac{\partial p}{\partial M_{gas}}\right)_m = \frac{p_m}{\rho_m T_m}\frac{T_{gas}}{V_{gas}} + \frac{p}{\rho_{gas} V_{gas}}\frac{c_{v,m}(T_m - T_{gas})}{c_{v,gas} T_{gas}}$$

The subscript *m* refers to the source parameters, while the parameters in the Control Volume are given the subscript *gas*. The symbol c_v represents isochoric specific volume (J/kg/K).

Note that in case of the gas outflow from a Control Volume the source parameters are the same as the Control Volume parameters: $T_m = T_{gas}$, $p_m = p$, $\rho_m = \rho_{gas}$, and the above equation becomes identical with the one shown above for the negative sources.

Finally a few remarks should be made about the calculation of the friction terms: C_f, C_{int}. Those terms depend on the absolute value of junction velocities, and are calculated based on beginning of time step velocities, $\bar{v}_i^{comp}$. The C_f and C_{int} are treated in the flow solution matrix as constants, independent of the end of time step velocities, v_i^{comp}.

Limits are imposed on the values of C_f and C_{int}, to prevent the friction terms from being very small when the beginning of time step velocities are close to zero. The minimum value for the wall friction term, $C_{f,\,min}$, is set based on the steady state friction pressure drop. The velocity corresponding to the steady state pressure drop Δp_i in junction *i*, is equal to:

$$v_i = \left(\frac{2\Delta p_i}{\rho_i (K_i + f_{wall,i} L_{f,i} / D_{f,i})}\right)^{1/2}$$

The value of $C_{f,\,min}$ is set to the value corresponding to the steady state pressure drop, multiplied by a constant factor, X_{Cfm}:

$$C_{f,\min,i} = \frac{\Delta p_i}{v_i} X_{Cfm} = \Delta p_i \left(\frac{\rho_i (K_i + f_{wall,i} L_{f,i} / D_{f,i})}{2\Delta p_i}\right)^{1/2} X_{Cfm}$$

or:

$$C_{f,\min,i} = X_{Cfm}\left[\frac{1}{2}\rho_i\left(K_i + f_{wall,i}\frac{L_{f,i} D_{f,i}}{D_{f,i}}\right)\Delta p_i\right]^{1/2}$$

The default value of the multiplier X_{Cfm} is 1.0. The value may be re-defined in the input deck (see Volume 2). The minimum value for the interphase friction term, $C_{int,\,min}$, is set to a small positive number. Currently the value of 10^{-10} is used.

4.2.2 Flow Solution for Continuous Components - Method 2 - LFM

The previous section describes the full flow solution, referred as a "Full Flow Matrix" (FFM). The present section describes the flow solution which leads to a "Limited Flow Matrix" (LFM). The Full Flow Matrix is always used at the beginning phase of the time step solution. In the subsequent iterations the FFM or the LFM may be solved, depending on the input parameter MSFMJN (see Volume 2).

The general form of the momentum equation is the same as described in the previous section.

$$\rho_i \cdot L_i \cdot \frac{dv_i}{dt} = p_{U,i} - p_{D,i} - C_{f,i} \cdot v_i - C_{int,i} \cdot (v_i - v_{i,-\phi}) - C_{acc} \cdot \frac{\rho_i ({v_D}^2 - {v_U}^2)}{2}$$

Here the subscripts *i, U, D* are used to for the junction, the upstream, and the downstream CV, respectively. C_{acc} is the acceleration pressure drop multiplier, equal to zero or one, depending on the activator IACCJN (see Volume 2), and $C_{f,i}$ $C_{int,i}$ are the wall friction and the interface friction terms, equal to:

$$C_{f,i} = \left(K_i + f_{wall,i} \cdot \frac{L_{f,i}}{D_{f,i}} \right) \cdot \frac{\rho_i \cdot |v_i|}{2}$$

$$C_{int,i} = f_{int,i} \cdot \frac{L_{f,i}}{D_{f,i}} \cdot \frac{\rho_i \cdot |v_i - v_{i,-\phi}|}{2}$$

In contrast to the FFM solution, which is the "first line of attack", i.e. it must estimate the end-of-time-step pressures based on the beginning-of-time-step values only, the LFM takes advantage of the end-of-time-step pressures, already computed by FFM in the earlier iterations. The general approach of the LFM is to find what is a velocity correction needed to obtain a correct pressure drop over the junction. The velocity correction for a given junction *i* is defined as:

$$\Delta v_i = (v_i - v_i')$$

Here v_i' is the velocity estimate obtained in the previous iterations (using FFM). In the FFM approach, the pressures are linearized around the beginning-of-time-step values. In the LFM approach the pressures are linearized around the end-of-time-step pressures, estimated earlier (by FFM).

$$p_{CV} = p_{CV}' + \left(\frac{\partial p}{\partial M} \right)_{m=1,CV} \cdot \Delta W_{m=1,CV} \cdot \Delta t$$

Here p_{CV}' is the pressure obtained in earlier iterations (by using the FFM solution). Only the terms related to the junction flows (m=1) are taken into account in the linearization. The other terms (mass and energy transfers from solid surfaces, non-equilibrium mass transfers, interface transfers, etc. are assumed not to change since the previous iterations.

The $\Delta W_{m=1,CV}$ is the mass flow correction corresponding to the velocity correction Δv_i, and is obtained by summing the corrections for all junctions connected to given CV. Therefore the junction flow term is written as:

$$\left(\frac{\partial p}{\partial M}\right)_{m=1,CV} \cdot \Delta W_{m=1,CV} \cdot \Delta t = \sum_{j \in CV} \sigma_{j,CV} \cdot \left(\frac{\partial p}{\partial M}\right)_{m=1,CV} \cdot A_j \cdot \rho_j \cdot (v_j - v_j') \cdot \Delta t$$

Here the sum is taken over all junctions *j*, which are connected to or from the control volume CV. The momentum equation becomes:

$$\begin{aligned}
&\rho_i \cdot L_i \cdot \frac{v_i - \bar{v}_i}{\Delta t} = \\
&= \left[p_{U,i}' + \sum_{j \in U} \sigma_{j,U} \cdot \left(\frac{\partial p}{\partial M}\right)_{m=1,U} \cdot A_j \cdot \rho_j \cdot (v_j - v_j') \cdot \Delta t \right] - \\
&- \left[p_{D,i}' + \sum_{j \in D} \sigma_{j,D} \cdot \left(\frac{\partial p}{\partial M}\right)_{m=1,D} \cdot A_j \cdot \rho_j \cdot (v_j - v_j') \cdot \Delta t \right] - \\
&- \quad C_{f,i} \cdot v_i \quad - \quad C_{int,i}(v_i - v_{i,-\phi}) \quad - \quad C_{acc} \cdot \frac{\rho_i ({v_D}^2 - {v_U}^2)}{2}
\end{aligned}$$

The values of $C_{f,i}$, $C_{int,i}$ are calculated using the velocity obtained in previous iterations.

$$C_{f,i} = \left(K_i + f_{wall,i} \cdot \frac{L_{f,i}}{D_{f,i}} \right) \cdot \frac{\rho_i \cdot |v_i'|}{2}$$

$$C_{int,i} = f_{wall,i} \cdot \frac{L_{f,i}}{D_{f,i}} \cdot \frac{\rho_i \cdot |v_i' - v_{i,-\phi}'|}{2}$$

Similarly as in the case of FFM, in order to obtain the formula for the matrix elements, the equation is re-written, by grouping all terms dependent on the unknown velocities (v_i , v_j) on the left hand side, and all other terms on the right hand side. The resulting equation has the form:

$$\begin{aligned}
&(C_{f,i} + C_{int,i} + \frac{\rho_i \cdot L_i}{\Delta t}) \cdot v_i \quad - \\
&- \quad \sum_{j \in U} \sigma_{j,U} \cdot \left(\frac{\partial p}{\partial M}\right)_U \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j \quad + \quad \sum_{j \in D} \sigma_{j,D} \cdot \left(\frac{\partial p}{\partial M}\right)_D \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j \quad = \\
&= \quad p_{i,U}' \quad - \quad p_{i,D}' \quad - \\
&- \quad \sum_{j \in U} \sigma_{j,U} \cdot \left(\frac{\partial p}{\partial M}\right)_U \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j' \quad + \quad \sum_{j \in D} \sigma_{j,D} \cdot \left(\frac{\partial p}{\partial M}\right)_D \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j' \\
&+ \quad C_{int,i} v_{i,-\phi} \quad - \quad C_{acc} \cdot \frac{\rho_i ({v_D}^2 - {v_U}^2)}{2} \quad - \quad \frac{\rho_i \cdot L_i}{\Delta t} \cdot \bar{v}_i
\end{aligned}$$

This equation set may be written shortly in a matrix form:

$$A\nu = B$$

The elements of the matrix ***A***, a_{ij}, are equal to:

- $i = j$:

$$a_{ii} = (C_{f,i} + C_{int,i} + \frac{\rho_i \cdot L_i}{\Delta t}) + \left(\frac{\partial p}{\partial M}\right)_U \cdot A_i \cdot \rho_i \cdot \Delta t + \left(\frac{\partial p}{\partial M}\right)_D \cdot A_i \cdot \rho_i \cdot \Delta t$$

  Here use has been made of the relations:

$$\sigma_{i,U} = -1$$

$$\sigma_{i,D} = +1$$

- $i \neq j$:

$$a_{ij} = -\sigma_{j,U} \cdot \left(\frac{\partial p}{\partial M}\right)_U \cdot A_i \cdot \rho_i \cdot \Delta t + \sigma_{j,D} \cdot \left(\frac{\partial p}{\partial M}\right)_D \cdot A_i \cdot \rho_i \cdot \Delta t$$

  with:

$$\sigma_{j,CV} = \begin{cases} -1 & \textit{if the volume CV is the "from CV" for the junction } j \\ +1 & \textit{if the volume CV is the "to CV" for the junction } j \end{cases}$$

The elements of the right hand side vector, ***B***, are equal to:

$$b_i = p_{i,U}' - p_{i,D}' - \left(\frac{\partial p}{\partial M}\right)_U \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j' + \sum_{j \in D} \sigma_{j,D} \cdot \left(\frac{\partial p}{\partial M}\right)_D \cdot A_j \cdot \rho_j \cdot \Delta t \cdot v_j' + C_{int,i} v_{i,-\phi} - C_{acc} \cdot \frac{\rho_i (v_D^{\,2} - v_U^{\,2})}{2} + \frac{\rho_i \cdot L_i}{\Delta t} \cdot \bar{v}_i$$

The Limited Flow Matrix is a typical, diagonally dominant matrix. It is solved using the LU decomposition method.

4.2.3 Flow of Dispersed Components

The calculation of discontinuous component velocities is based on the drift flux model ([48], chapter 4) and is very similar to the calculation of vertical and horizontal velocities of dispersed components within the Control Volume (section 2.5.2). The derivation of the final form of the drift flux equation is presented in section 2.5.2. Here only the final equations are shown, and discussed.

- Vertical junctions

  The velocities of discontinuous components in vertical junctions are calculated from:

$$v_v^{Dcomp} = \frac{C_{0,v}^{Dcomp} v_v^{Ccomp}(1-\alpha^{Dcomp}) + v_{\infty,v}^{Dcomp}(1-\alpha^{Dcomp})^{x_v^{Dcomp}}}{1-C_{0,v}^{Dcomp}\alpha^{Dcomp}}$$

  The meaning of the symbols is described in section 2.5.2. The subscript v is used here to indicate vertical junctions.

  The values of $C_{0,v}^{Dcomp}, v_{\infty,v}^{Dcomp}, x_v^{Dcomp}$ depend on the flow regime, and are calculated differently for the bubbly and droplet flows, as described in section 2.6. The values may also be defined for each junction through input data.

  The positive flow direction in a vertical junction is defined in input data, and may be either upwards or downwards. Therefore, to preserve the applicability of the drift flux equation, the following calculation procedure is applied:

  - The "natural positive direction" for given components is identified, and the continuous component velocity is expressed for that direction. For example, in case of pool/bubble components the natural positive direction is upwards. If the junction direction is upwards, then velocities of those components remain the same. If the junction direction is downwards then the velocities are taken with reversed signs.
  - The drift flux equation is applied to calculate the dispersed component velocity.
  - The calculated velocity of the dispersed component is transformed back to the value appropriate for the actual junction positive direction.

- Horizontal junctions

  The velocities of the discontinuous components in horizontal junctions are calculated from:

$$v_h^{Dcomp} = \frac{C_{0,h}^{Dcomp} v_h^{Ccomp}(1-\alpha^{Dcomp}) + v_{\infty,h}^{Dcomp}(1-\alpha^{Dcomp})^{x_h^{Dcomp}}}{1-C_{0,h}^{Dcomp}\alpha^{Dcomp}}$$

where the subscripts *h* mean horizontal junction. For the horizontal flow the values: $C_{0,h}^{Dcomp}, v_{\infty,h}^{Dcomp}$ are assumed to be equal to zero and one respectively, but may be changed for each junction by input. With the default values, the above equation reduces to:

$$v_h^{Dcomp} = v_h^{Ccomp}$$

If the user wishes to obtain a constant velocity of the dispersed component in a horizontal junction, v_{hor}, then the following values should be applied (see section 2.5.2):

$$v_{\infty,h}^{Dcomp} = v_{hor}$$
$$C_{0,h}^{Dcomp} = 10^{-10}$$
$$x_h^{Dcomp} = 10^{-10}$$

With the above values the velocities of dispersed components will be equal to v_{hor}, and the flow will always occur in positive direction. To obtain constant flow in negative direction the terminal velocity must be set to a negative number: $v_{\infty,h}^{Dcomp} = -v_{hor}$

4.2.4 Water Level in a Junction

The junction flow matrix can be solved if all matrix coefficients are known. The matrix coefficients depend on the component flow areas, that means flow areas available for atmosphere and pool flow. The methods of determining the component flow areas are described in this section.

The component flow areas are determined based on the water level in a junction. The junction water level in turn is calculated, as described below, based on water levels in the control volumes, as well as the component velocities in the junction. Beginning of time step velocities are used to determine the junction water level.

The following approach is used to calculate the water level in junctions.

In case when both atmosphere and pool velocities are in the same direction, the water level in the junction is the same as the water level in the control volume where the flow originates. If the velocities of atmosphere and pool are in different directions, then a counter-current flow limit is used to determine the water level in junctions.

The procedure is illustrated in Figure 4-4. In the case (a) the whole junction area is available for the atmosphere flow, independently of the gas velocity. In the case (b) both atmosphere and pool velocities are from CV-1 to CV-2, and therefore the water level in the junction is the same as the water level in CV-1. In the case (c) the flow is again unidirectional, since no gas flow occurs. Consequently the whole junction area is available for pool flow. Possible back-flow of atmosphere is in such case considered, and a counter current flow limit is applied whenever the atmosphere pressure on the CV-2 side of the junction is larger than the atmosphere pressure on the CV-1 side of the junction (including the hydrostatic head of water column above the top of the junction).

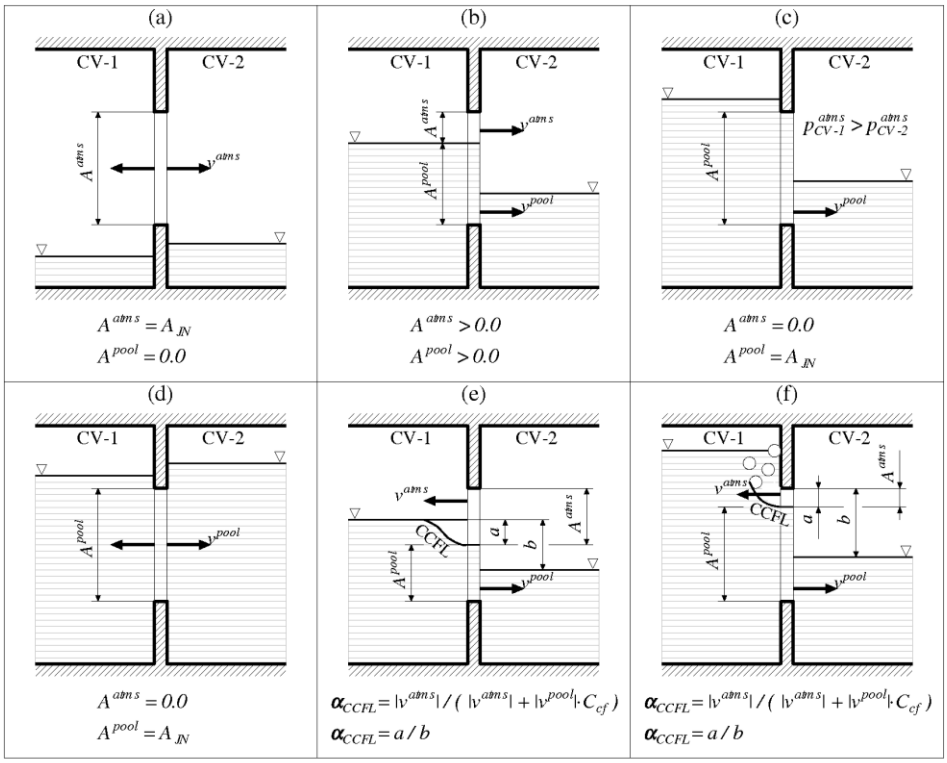


Figure 4-4 Pool level in junctions

In the case (d) the whole junction area is available for the pool flow, independently of the liquid velocity. Finally the last two cases represent the situations when the counter-current flow limit (CCFL) is applied.

In the case (e) the water level in the junction is equal to:

$$Z_{JN}^{pool} = Z_{CV-1}^{pool} - (Z_{CV-1}^{pool} - Z_{CV-2}^{pool}) \cdot \frac{a}{b} =$$
$$= Z_{CV-1}^{pool} - (Z_{CV-1}^{pool} - Z_{CV-2}^{pool}) \cdot \alpha_{CCFL}$$

where the counter-current flow limit, α_{CCFL}, is defined by a criterion similar to that used in MELCOR [46]:

$$\alpha_{CCFL} = \frac{\left|v^{atms}\right|}{\left|v^{atms}\right| + \left|v^{pool}\right| C_{cf}}$$

In the above formula C_{cf} is a constant, dependent on the flow direction. The default values of C_{cf} are 0.1 for vertical flow and 10.0 for horizontal flow respectively. The values may be redefined for each junction through input data (Volume 2). Note that with the approach described above the component flow areas are smooth functions of the component velocities. When in case (e) the gas velocity approaches zero, then $\alpha_{CCFL} \rightarrow 0$, and $Z_{JN}^{pool} = Z_{CV-1}^{pool}$, which is the same as the value obtained in case (b).

Finally, in the case (f) the water level in the junction is equal to:

$$Z_{JN}^{pool} = Z_{JN}^{top} - (Z_{JN}^{top} - Z_{CV-2}^{pool}) \cdot \frac{a}{b} =$$
$$= Z_{JN}^{top} - (Z_{JN}^{top} - Z_{CV-2}^{pool}) \cdot \alpha_{CCFL}$$

where Z_{JN}^{top} is the junction top elevation. The limit, α_{CCFL}, is calculated as above. Again, when the gas velocity approaches zero, then $Z_{JN}^{pool} = Z_{JN}^{top}$, and the whole junction area becomes available for the pool flow, as in case (c).

In the discussion above only horizontal junction was considered. The calculation procedure in the vertical junctions is very similar, the differences consist of a different value of the constant C_{cf}, and the values of the bottom and top junction elevations. In case of horizontal junctions the values are the same for the "from" and "to" volumes, and equal to: ($Z_{JN} - H_{JN}/2$), and ($Z_{JN} + H_{JN}/2$), respectively (Figure 4-5 a).

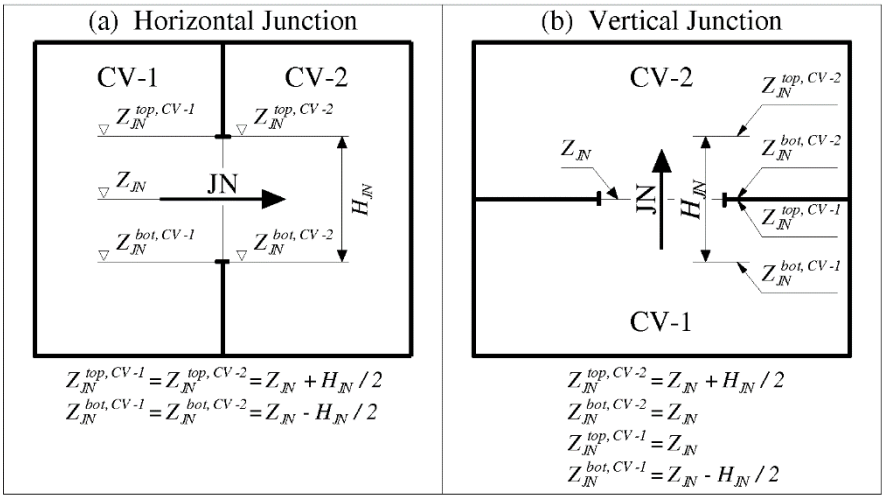


Figure 4-5 Bottom and top elevations for horizontal and vertical junctions

In case of vertical junctions the values are different for the "from" and "to" volumes. Figure 4-5 (b) shows how the values of bottom and top elevation of junction JN in the control volumes CV-1 and CV-2, are defined. The values are:

$$CV\ above\ junction: \begin{cases} Z_{JN}^{top,CV-002} = Z_{JN} + H_{JN}/2 \\ Z_{JN}^{bot,CV-002} = Z_{JN} \end{cases}$$

$$CV\ below\ junction: \begin{cases} Z_{JN}^{top,CV-001} = Z_{JN} \\ Z_{JN}^{bot,CV-001} = Z_{JN} - H_{JN}/2 \end{cases}$$

In the above formulas H_{JN} is the junction height, or the "opening height" of a junction. This is a user input parameter with the default value of:

$$horizontal\ junctions: \quad H_{JN} = D_{JN}$$
$$vertical\ junction: \quad H_{JN} = D_{JN}/2$$

where D_{JN} is the junction diameter.

When the pool level in the junction is determined, then the component flow areas are calculated from the following formulae:

$$A^{atms} = A_{JN} X_{JN}^{atms} (1 - \alpha^{drop})$$
$$A^{drop} = A_{JN} X_{JN}^{atms} \alpha^{drop}$$
$$A^{pool} = A_{JN} X_{JN}^{pool} (1 - \alpha^{bubb})$$
$$A^{bubb} = A_{JN} X_{JN}^{pool} \alpha^{bubb}$$

A_{JN} total junction flow area, (m^2)

X_{JN}^{atms} fraction of area available for atmosphere flow, equal to:

$$X_{JN}^{atms} = (Z_{JN}^{top} - Z_{JN}^{pool}) / (Z_{JN}^{top} - Z_{JN}^{bot})$$

X_{JN}^{pool} fraction of area available for pool flow, equal to:

$$X_{JN}^{pool} = (Z_{JN}^{pool} - Z_{JN}^{bot}) / (Z_{JN}^{top} - Z_{JN}^{bot})$$

α_{drop} volumetric fraction of droplets in the junction, equal to the local droplet volumetric fraction in the CV where the droplet flow originates, at the elevation of the center point of this part of the junction which is above the CV water level.

α_{bubb} volumetric fraction of bubbles in the junction, equal to the local bubble volumetric fraction in the CV where the bubble flow originates, at the elevation of the center point of this part of the junction which is below the CV water level.

4.3 Friction Models

4.3.1 Wall Friction

4.3.1.1 Friction Factor Models

Three models are available for the calculation of the single-phase wall friction factor. Those are:

- Model for non-uniform wall roughness, with the Colebrook-White formula for the friction factor in fully developed turbulent flow.
- Model for uniform wall roughness, with the Nikuradse formula for the friction factor in fully developed turbulent flow.
- Simplified model, with the Blasius equation for smooth walls, and Prandtl-Nikuradse for rough walls. While the two models mentioned above require iterative calculation, this method gives a straightforward formula for the friction factor.

The three models are shortly described below. The wall roughness for typical materials are given in Table 4-2.

- ***Non-uniform roughness (most commercial pipes)***

  The model is based on [2]. Four regions are distinguished: laminar flow, two transition regions, and turbulent flow.

  - Laminar flow: $Re < Re_0 = 0.0754 \cdot \exp(0.0065/e)$
    The meaning of the symbols used above is: Re - Reynolds number, e - relative equivalent roughness. In the laminar range the Hagen-Poiseuille law for circular tubes is used with a correction factor:

$$f_{wall} = \frac{64 \cdot C_{lam}}{\mathrm{Re}}$$

    Here C_{lam} is the correction factor for non-circular tube geometry, defined by the user. A minimum value of Reynolds number is imposed, $Re_{min} = 64$. If $Re < Re_{min}$ then it is set to Re_{min}.

  - First transition region: $Re_0 < Re < Re_1 = 1160/e$
    A log-linear interpolation is performed between the Hagen-Poiseuille law for Re_0: $f_0 = 64/Re_0$, and the Samoilenko formula [3] for Re_1.

    The Samoilenko formula is:

$$f_{wall} = \frac{4.4}{\mathrm{Re}^{0.595}} \exp\left(-\frac{0.00275}{e_1} \right)$$

    where: e_1 - maximum of e and 0.007.

- Second transition region: $Re_1 < Re < Re_2 = 2090/e^{0.0635}$
  A third order interpolation is performed between a value obtained from the Samoilenko formula for Re_1, and a value obtained from the Colebrook-White formula for Re_2.

- Turbulent flow: $Re > Re_2$
  The friction factor is calculated using the Colebrook-White formula [4]:

$$f_{wall} = \left[2\log_{10}\left(\frac{2.51}{\mathrm{Re}\sqrt{f_{wall}}} + \frac{e}{3.7} \right) \right]^{-2}$$

  Since f_{wall} is given by an implicit formula, an iterative calculation is needed. The first guess for the iteration is obtained using the Prandtl-Nikuradse formula, valid for the stabilized turbulent flow ($Re \to \infty$), [5]:

$$f_{wall} = \left[2\log_{10}\left(\frac{3.7}{e} \right) \right]^{-2}$$

The final friction factors, calculated using the procedure described above, are shown in Figure 4-6. In the laminar range a linear decrease of the friction factor with increasing Re is observed in the logarithmic scale. In the first transition region the factor is still decreasing, but a number of different curves are obtained for different values of wall roughness. In the second transition region the friction factor increases with increasing Re. Finally a slow decrease of f_{wall} with Re is observed in the turbulent region.

- ***Uniform roughness***

Three regions are distinguished: laminar, transition, and turbulent flow.

- Laminar flow, $Re < Re_{lam} = 2000$
  In the laminar region the Hagen-Poiseuille law is used. As in case of non-uniform roughness a minimum value of Re is imposed: $Re_{min} = 64$.

- Transition region, $Re_{lam} \le Re \le Re_{tur} = 4000$
  In this region a third order interpolation is performed between the values of the laminar friction factor, f_{lam}, calculated from the Hagen-Poiseuille law for $Re=Re_{lam}$, and turbulent friction factor, f_{tur}, calculated from the Nikuradse formula for $Re=Re_{tur}$.

- Turbulent region, $Re > Re_{tur}$
  The friction factor is calculated using the Nikuradse equation [5]:

$$f_{wall} = \left[a + b\log_{10}\left(\mathrm{Re}\sqrt{f_{wall}}\right) + c\log_{10}(e) \right]^{-2}$$

  the values of the constants *a, b, c* depend on the value of the product: $(e \cdot \mathrm{Re} \cdot f_{wall}^{1/2})$, and are shown in Table 4-1.

Table 4-1 Coefficients in the Nikuradse equation for friction factor.

$e\,\mathrm{Re}\,f_{wall}^{1/2}$	a	b	c
3.6 ÷ 10	–0.800	+2.000	0.000
10 ÷ 20	+0.068	+1.130	–0.870
20 ÷ 40	+1.538	0.000	–2.000
40 ÷ 191.2	+2.471	–0.588	–2.588
> 191.2	+1.138	0.000	–2.000

Since f_{wall} is given implicitly, an iterative calculation is needed. The first guess for the iteration is obtained using the Prandtl-Nikuradse formula:

$$f_{wall} = \left[2\log_{10}\left(\frac{3.7}{e}\right)\right]^{-2}$$

The values of friction factors are shown in Figure 4-7. It is seen that the friction factors calculated for uniform roughness are smaller than those obtained for non-uniform roughness for small Reynolds numbers. In the stabilized turbulent flow ($Re\rightarrow\infty$) the values are practically the same.

- ***Simplified Model***

Three regions are distinguished: laminar, transition, and turbulent flow.

- Laminar flow, $\mathrm{Re} < \mathrm{Re}_{lam} = 2000$
  In the laminar region the Hagen-Poiseuille law is used. As in case of non-uniform roughness a minimum value of Re is imposed: $\mathrm{Re}_{min} = 64$.

- Transition region, $\mathrm{Re}_{lam} \le \mathrm{Re} \le \mathrm{Re}_{tur} = 4000$
  In this region a maximum value is used of laminar friction factor, f_{lam}, calculated from Hagen-Poiseuille law for $\mathrm{Re}=\mathrm{Re}_{lam}$, and turbulent friction factor, f_{tur}, calculated as shown below for $\mathrm{Re}=\mathrm{Re}_{tur}$.

- Turbulent region, $\mathrm{Re} > \mathrm{Re}_{tur}$
  The friction factor is calculated as a maximum of the smooth and the rough wall models:
  - Smooth walls, Blasius formula:

$$f_{smooth} = \frac{0.3164}{\mathrm{Re}^{0.25}}$$

  - Rough walls, Prandtl-Nikuradse formula, [5]:

$$f_{wall} = \left[2\log_{10}\left(\frac{3.7}{e}\right)\right]^{-2}$$

The values of friction factors are shown in Figure 4-8. The simple model gives a relatively good approximation of the friction factor, without the necessity of iterative calculation.

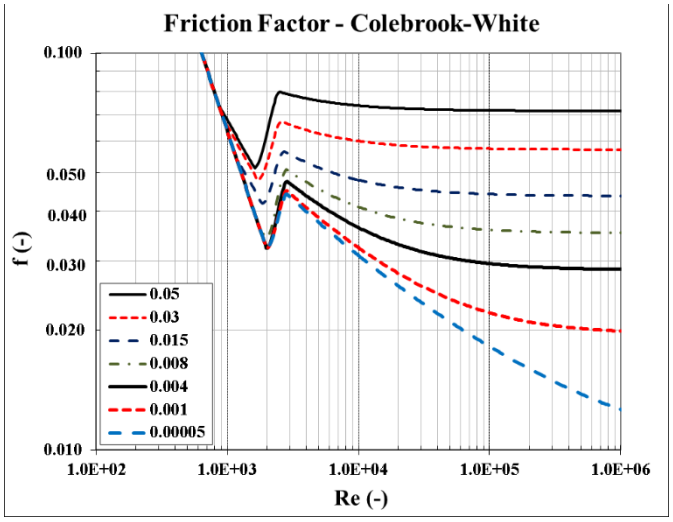


Figure 4-6 Friction factor, non-uniform roughness

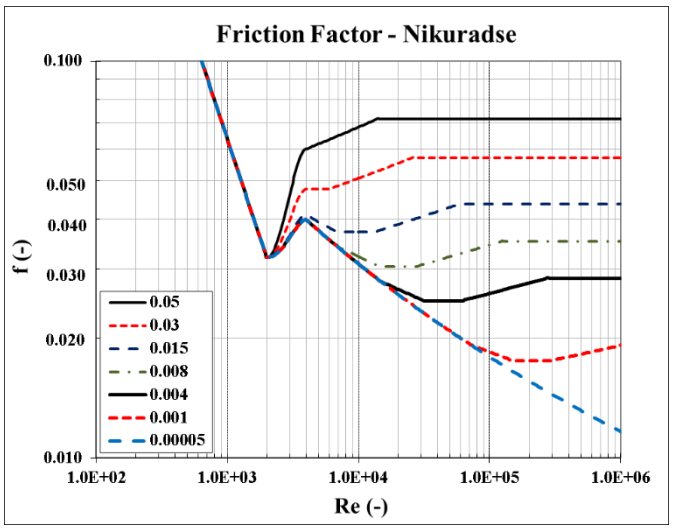


Figure 4-7 Friction factor, uniform roughness

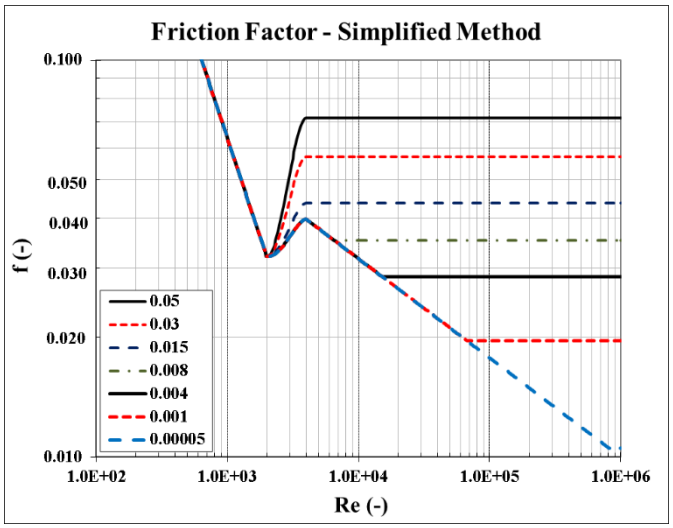


Figure 4-8 Friction factor, simplified model

4.3.1.2 Non-Uniform Roughness - Approximations of Colebrook-White Formula

The Colebrook-White formula requires iteration, since the friction factor depends on itself:

$$f_{wall} = \left[2\log_{10}\left(\frac{2.51}{\mathrm{Re}\sqrt{f_{wall}}} + \frac{e}{3.7} \right) \right]^{-2}$$

Several approximations of the Colebrook-White formula have been developed and published in open literature, for example [222]. Two approximations are available in SPECTRA. These are discussed below. In each case only the turbulent correlation (Colebrook-White) is replaced by the approximation. The laminar and transition models are the same.

- Beluco and Camano approximation [222]:

$$f_{wall} = \left[2\log_{10}\left(\left(\frac{5.9802}{\mathrm{Re}} \right)^{0.9695} + \left(\frac{e}{3.7315} \right)^{1.0954} \right) \right]^{-2}$$

  The friction factor calculated based on the Beluco and Camano approximation in the turbulent range is shown in Figure 4-10. For comparison, the results based on the Colebrook-White are shown in Figure 4-9. As shown in Volume 3, the average relative difference between the Beluco and Camano formula and the Colebrook-White formula is:

$$\delta = 0.08\%$$

  The agreement is excellent and the correlation is significantly faster in numerical calculations, therefore it is the default correlation in SPECTRA.

- Churchill approximation is [222]:

$$f_{wall} = \left[2\log_{10}\left(\frac{5.74}{\mathrm{Re}} + \frac{e}{3.7} \right) \right]^{-2}$$

  The friction factor calculated based on the Churchill approximation in the turbulent range is shown in Figure 4-11. For comparison, the results based on the Colebrook-White are shown in Figure 4-9. As shown in Volume 3, the average relative difference between the Churchill formula and the Colebrook-White formula is:

$$\delta = 4.7\%$$

  This approximation is clearly not as good as the previous one, but since it is frequently used in engineering applications, it is available in SPECTRA.

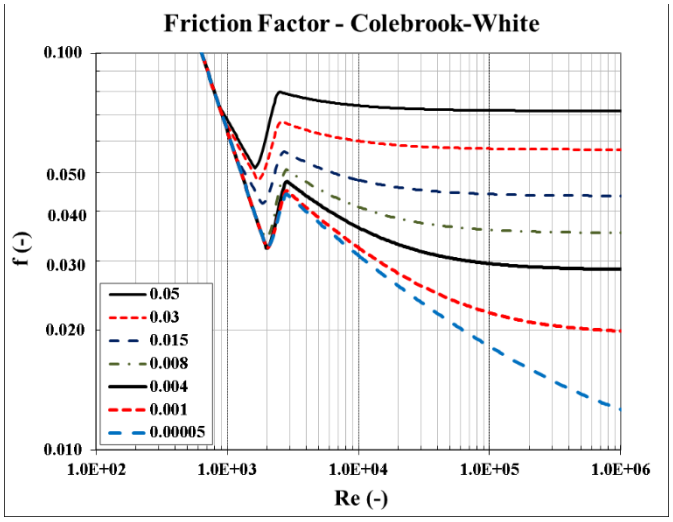


Figure 4-9 Friction factor, non-uniform roughness, Colebrook-White formula

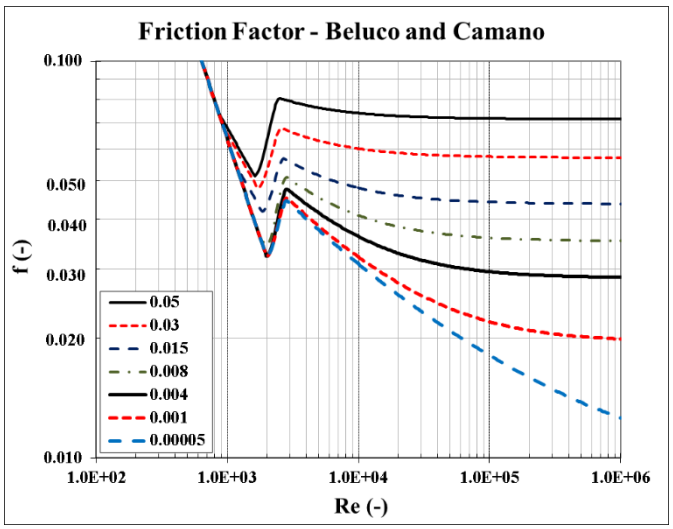


Figure 4-10 Friction factor, non-uniform roughness, Beluco and Camano formula

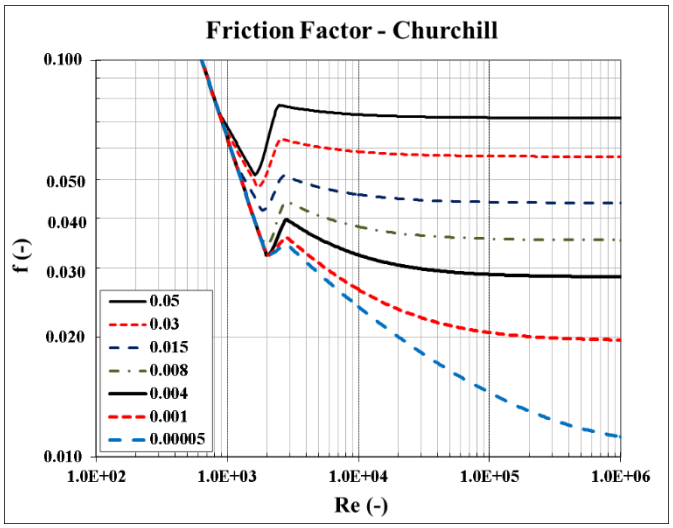


Figure 4-11 Friction factor, non-uniform roughness, Churchill formula

4.3.1.3 Surface Roughness of Materials

The surface roughness for several typical materials are given in Table 4-2.

Table 4-2 Wall roughness of typical materials [19].

Material	Roughness	
	ft	m
Drawn tubing	0.000,005	1.5×10^{-6}
Commercial steel pipe	0.000,15	4.6×10^{-5}
Galvanised iron	0.000,5	1.5×10^{-4}
Cast iron	0.000,85	2.6×10^{-4}
Riveted pipe	0.003 - 0.03	9.1×10^{-4} - 9.1×10^{-3}

4.3.2 Two-Phase Friction Factor and Form Loss Multipliers

Two different two-phase multipliers are used by the code: the friction factor multiplier, Φ_f^2, and the form loss factor multiplier, Φ_K^2. Those factors are used with the wall friction factor, f_{wall}, and the form loss factor, K, respectively. The two-phase friction factor multipliers are used when pool and bubble components are present in the same junction. In such case both the friction factor and the form loss factor are multiplied by the appropriate two phase factor. In case of atmosphere/droplet flow through a junction no two phase multiplier is applied.

Therefore:

$$f_{wall,2\Phi}^{pool} = f_{wall}^{pool} \cdot \Phi_f^2 \qquad K_{wall,2\Phi}^{pool} = K_{wall}^{pool} \cdot \Phi_K^2$$

$$f_{wall,2\Phi}^{atms} = f_{wall}^{atms} \qquad K_{wall,2\Phi}^{atms} = K_{wall}^{atms}$$

Two models are available for calculation of the two-phase friction multiplier: Hancox-Nicoll, and Levy. Only one correlation, Griffith-Rohsenow, is available for the two-phase form loss multiplier. All these models are described below.

- Hancocx-Nicoll correlation for Φ_f^2

  The Hancox-Nicoll correlation [50] is:

$$\Phi_f^2 = \left\{1+\left[\left(\frac{\rho_{liq}}{\rho_{gas}}\right)\left(\frac{\eta_{liq}}{\eta_{gas}}\right)^{1/5}-1\right]X\right\}\cdot\left[1+RX^{1/2}(1-X)^{1/4}\right]$$

Φ_f^2 two-phase friction factor multiplier, (-)
ρ_{liq} density of liquid, (kg/m^3)
ρ_{gas} density of gas, (kg/m^3)
η_{liq} viscosity of liquid, (kg/m-s)
η_{gas} viscosity of gas, (kg/m-s)
X quality, (-)

The value of R is given by the formula:

$$R = 3.1 \cdot \left(1 - \frac{p_{sat}}{p_{crit}}\right) \cdot \exp(-0.000565\, G)$$

p_{sat} saturation pressure, (Pa)
p_{crit} critical pressure of water, (Pa), (equal to: 2.212×10^7 [31]
G mass flux, (kg/m2/s).

The constant multiplier in the exponent has been wrongly given as 0.0135 in the article [50]. The correct value is obtained by dividing 0.00276 in the Hancox' original report by 4.885, which is the conversion factor from British to SI units [51]. The value in the article was obtained by multiplying instead of dividing the original value by the conversion factor.

- Levy correlation for Φ_f^2

The Levy correlation [52] is:

$$\Phi_f^2 = \frac{1}{(1-\alpha)^2}$$

Here α is the void fraction (volumetric fraction of bubbles in the pool).

- Griffith-Rohsenow correlation for Φ_K^2

The Griffith-Rohsenow correlation [54] is:

$$\Phi_K^2 = 1 + r_f \cdot X \cdot \frac{v_{gas} - v_{liq}}{v_{liq}}$$

v_{liq} specific volume of liquid, (m^3/kg)
v_{gas} specific volume of gas, (m^3/kg)
X quality, (-)
r_f empirical constant, tabulated values of r_f are shown in [55], and are reproduced in Table 4-3.

Table 4-3 Values of r_f for two phase loss factor multiplier (reproduced from [55], table 7.4).

Fitting	Pressure range	Quality range (%)	r_f
Bend-short	$p/p_{crit} < 0.15$	0 - 10	1.5
Bend-short	$p/p_{crit} > 0.10$	0 - 50	4.0
Bend-long	$p/p_{crit} > 0.10$	0 - 50	2.2
Tee (serving as L)	$p/p_{crit} > 0.10$	0 - 50	1.6
Gate valve	$p/p_{crit} < 0.10$	0 - 50	1.5
Contractions	$0.015 < p/p_{crit} < 1$	0 - 50	1.0
Expansions	$0.15 < p/p_{crit} < 1$	0 - 50	1.1

4.3.3 Interphase Friction

The interphase friction factor is calculated based on the formula from [49]:

$$f_{\text{int}}^{Wallis} = 0.005 \cdot [1 + 75 \cdot (1 - \alpha)]$$

The definition of the friction factor applied in SPECTRA requires that the factor defined by the Wallis formula must be multiplied by 4:

$$f_{\text{int}} = 4 \cdot f_{\text{int}}^{Wallis} = 0.02 \cdot [1 + 75 \cdot (1 - \alpha)] = 1.5 \cdot (1.013333 - \alpha)$$

The above formula is used in the code, with the void fraction defined as the ratio of the atmosphere flow area to the sum of the atmosphere and pool flow areas: $\alpha = A_{atms}/(A_{pool} + A_{atms})$.

4.4 Critical Flow Model

The approach to model the critical flow in junctions is somewhat similar to the approach taken in the MELCOR code [46]. After the solution of the junction flow matrix, the computed mass fluxes are compared with critical mass fluxes, to determine if choking should be imposed. The test is bypassed for junctions where velocities do not exceed a threshold of 10 m/s (this value may be changed through input, see Volume 2, record 260000). If the flow exceeds the critical value, then critical flow is imposed in the junction, the junction is eliminated from the flow solution matrix, and the flow solution is repeated. In such case the velocities in the critical junction are set by the critical flow model, as described below, and therefore are treated as constant (like for example junctions with user specified flow), independent of other junction flows.

A special situation arises when after a flow solution is complete, more than one junction is calculated to be critical. In such cases setting velocities by the critical flow model in all those junctions at once would be wrong, because if the critical flow in one of them is taken into account, then the flow in other junctions may become subcritical. In such case the following procedure is applied. A junction is selected with the largest mass flux. Critical flow is set only in this junction, and the calculation of the junction flow matrix is repeated. If a supercritical flow is still encountered in another junction(s), then the procedure is repeated again. This is done until all critical junctions are identified. It has been found out in multiple test calculations that this procedure allows to identify correctly the critical junctions and provide an adequate flow solution.

The critical mass flux is calculated from:

$$G_c = C_M \cdot G_{c,Moody} \cdot f_{L/D}$$

G_c critical mass flux, (kg/m^2-s),
C_M Moody coefficient (see [57]), the value of C_M is by default equal to 0.70, but may be redefined for each junction by input data (Volume 2, record 210XXX).
$G_{c, Moody}$ critical mass flux, (kg/m^2-s), calculated from the base model. The Moody model is used as the base critical flow model.
$f_{L/D}$ influence of *L/D* (length to diameter) ratio, (-)

The base critical flow model is an approximation of the Moody model [58]. The approximation formulae allow to obtain fast calculations and are in good agreement with the original model. In the two phase range the approximation is based on a recent conclusion, made by Gauntt et. al. [46]. It has been found out that the Moody model is very well approximated by interpolating an inverse of the mass averaged velocities between the saturated liquid and saturated vapor values (see section 4.4.3). In the subcooled and superheated regions approximation formulae have been developed to fit the data tabulated in RELAP-4 [59]. These approximation formulae represent very well (difference below 1 % - Volume 3) the exact data of the model.

The equations of the critical flow model are shown below, in five sections. Section 4.4.1 presents the model used for subcooled water. Section 4.4.2 presents the model for gas flow. Section 4.4.3 presents the critical flow model for two-phase mixtures. Next, the method of calculating the influence of L/D on the critical flow is shown in section 4.4.4. Finally, section 4.4.5 shows the method of calculating the slip ratio, and setting the junction component velocities once the overall critical mass flux has been determined.

4.4.1 Subcooled Liquid Critical Flow

The subcooled liquid critical flow model uses an equation that approximates the Moody model subcooled flow data tables from RELAP-4 ([59], subroutine DATZ). The critical mass flux is calculated using the following equation:

$$G_{c,liq} = A_1(p) + A_2(p) \cdot \{1 - \exp[A_3(p) \cdot (h - h_0(p))]\}$$

Here $G_{c,liq}$ is the critical mass flux (kg/m^2-s), p is the stagnation pressure (Pa), h is the stagnation enthalpy (J/kg). The coefficients A_1, A_2, A_3, h_0, are functions of pressure, and are tabulated for the pressure range: from 0.689×10^5 Pa to 165.0×10^5 Pa. Third order "smooth" interpolation (section 17.1) is performed to obtain the coefficients A_1, B_1, C_1, h_0, for the given pressure, p. The values of these coefficients, as calculated using the third order interpolation, are shown in Figure 4-12 and Figure 4-13.

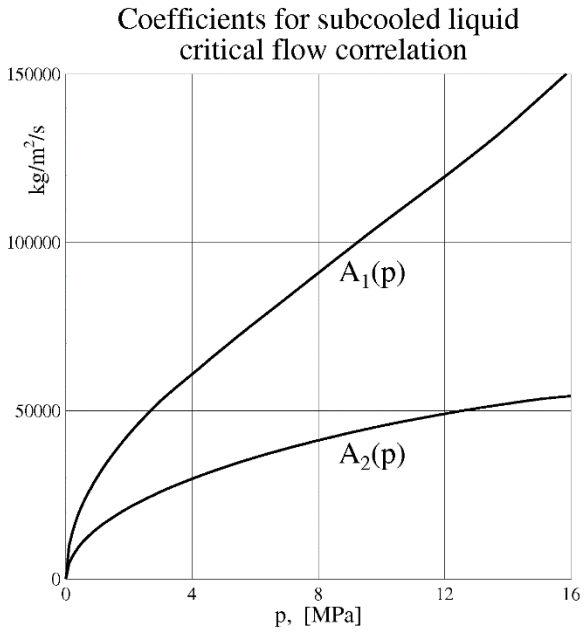


Figure 4-12 Coefficients A_1 and A_2, subcooled liquid critical flow correlation

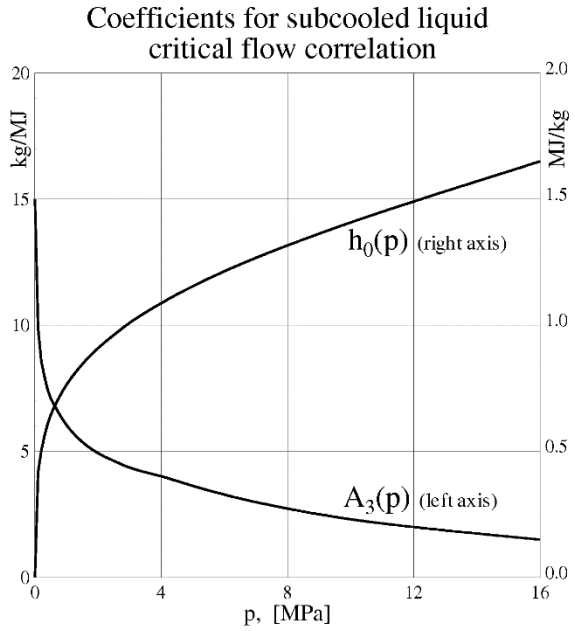


Figure 4-13 Coefficients A_3 and h_0, subcooled liquid critical flow correlation

The above correlation provides a good fit to the tables from RELAP-4. The difference between the calculated G_c and the value from the tables is typically below 1% (see Volume 3).

Use of the correlation was preferred over other methods (tabulated data, as in RELAP-4; polynomial expansion, as in RETRAN (see [46]) or direct calculation of fluid properties and their derivatives, for the following reasons:

- It is relatively simple, and simultaneously fast in calculations.
- There are no discontinuities in the calculated critical flux G_c.
- It has good extrapolation properties (see Volume 3).

Comparison of the equation with the source table is shown in Figure 4-14. A more detailed comparison is shown in Volume 3, for the full pressure range, from 10 psia to 2400 psia (0.06895 - 16.55 MPa). It should be noted that although the original tables in RELAP-4 are in British units, the approximation equations, as well as the full model, are written completely in SI units.

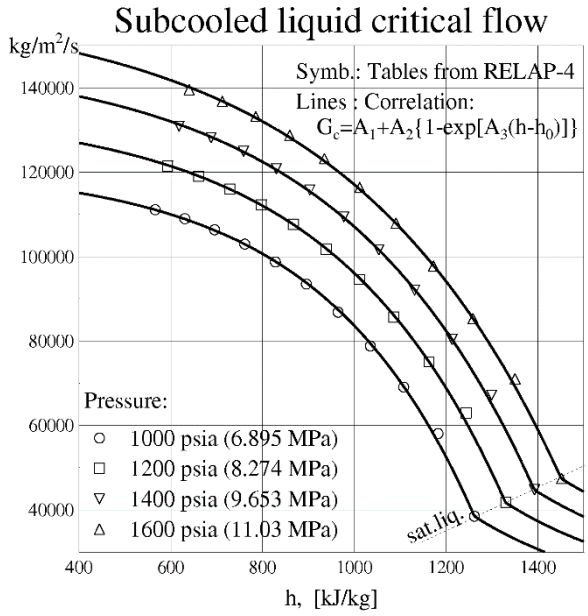


Figure 4-14 Subcooled liquid critical flow - SPECTRA correlation and data

4.4.2 Critical Flow of a Gas Mixture

In case of a gas mixture the critical flow model uses the perfect gas equation [60]:

$$G_{c,gas} = \sqrt{\gamma\, p\, \rho \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}$$

Here ρ is the gas density, γ is the ratio of specific heats, $\gamma = c_p/c_v$. The values of the specific heats for a gas mixture are calculated by averaging the values for individual gases flowing through a junction, as follows:

$$\left(\frac{c_p}{c_v}\right)_{gas} = \frac{|W_{atms}|c_{p,atms} + |W_{bubb}|c_{p,bubb}}{|W_{atms}|c_{v,atms} + |W_{bubb}|c_{v,bubb}}$$

$c_{p,\,atms}$ specific heat at constant pressure for the atmosphere gas
$c_{v,\,atms}$ specific heat at constant volume for the atmosphere gas
$c_{p,\,bubb}$ specific heat at constant pressure for the bubble gas
$c_{v,\,bubb}$ specific heat at constant volume for the bubble gas
W_{atms} mass flow, (kg/s) of the atmosphere gas
W_{bubb} mass flow, (kg/s) of the bubble gas

The individual specific heats for the atmosphere and bubble gas are obtained from a general gas mixture property procedure, described in section 3.5.2. For pure steam a correlation has been developed that provides a good fit to the Moody model data tables from RELAP-4. The superheated steam critical flow correlation is:

$$G_{c,steam} = \frac{Cp}{\sqrt{h - h_0}}$$

C and h_0 are constants, equal to 1.427 (-) and 1.837×10^6 (J/kg), respectively. Although the correlation provides a very good fit to the Moody data tables (Figure 4-15), it is currently not used by the code, to avoid discontinuity in the model when a pure steam flow changes into a flow of a mixture of gases. Therefore the perfect gas equation is always used to calculate the critical flow of gas, and consequently: $G_{c,\,steam} = G_{c,\,gas}$. The perfect gas correlation gives very similar results to the superheated steam correlation, as shown in Volume 3.

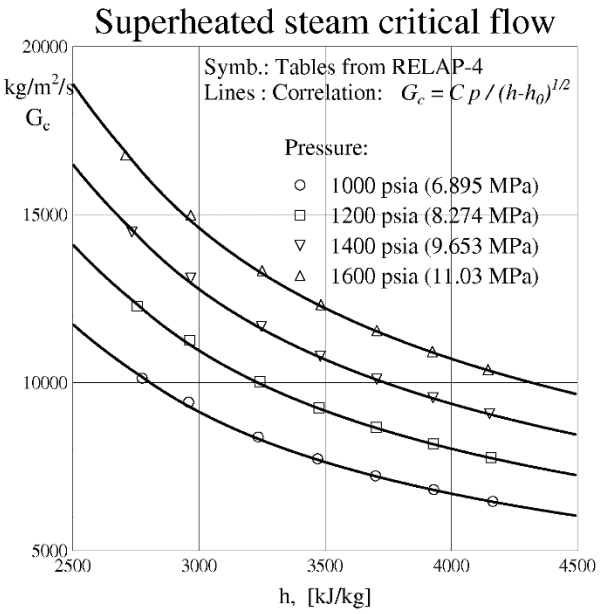


Figure 4-15 Superheated steam critical flow - SPECTRA correlation and data

4.4.3 Two-Phase Critical Flow

The two phase critical flow model uses an interpolation between the pure liquid and the pure vapor critical flows. The equation given in [46] is used:

$$\frac{\rho_{2\phi}}{G_{c,2\phi}} = \frac{(1-\alpha)\rho_{liq}}{G_{c,liq}} + \frac{\alpha\rho_{gas}}{G_{c,gas}}$$

α void fraction, (-)
$G_{c,liq}$ critical mass flux for pure liquid, (kg/m^2-s)
$G_{c,gas}$ critical mass flux for pure gas, (kg/m^2-s)
ρ_{liq} density of liquid phase, (kg/m^3)
ρ_{gas} density of gas phase, (kg/m^3)
$\rho_{2\varphi}$ mixture density, (kg/m^3), equal to: $\rho_{2\varphi} = (1 - \alpha) \rho_{liq} + \alpha \rho_{gas}$

The equation states that the inverse of the mass averaged velocity in critical flow is a linear function of void fraction [46]. It was found out by Gauntt et al. that this interpolation provides an extremely good fit to the Moody critical flow tables from RELAP-4 (see [46], CVH/FL Package Reference Manual, figure C.1).

4.4.4 Influence of *L/D*

In case of critical discharge of low quality fluids through a pipe, there is a strong influence of the ratio of tube length to diameter, *L/D*, on the flow rate. This fact is taken into account in the model by the *L/D* multiplier on critical flow, $f_{L/D}$, defined as the ratio of the mass flux for the actual *L/D* ratio to the value obtained for $L/D \rightarrow \infty$:

$$f_{L/D} = \frac{G_{c,L/D}}{G_{c,L/D\rightarrow\infty}}$$

The values of *L* and *D* are specified for each junction in input data (the values of friction length and diameter: L_f, D_f are used). In case of the pure liquid flow the multiplier, $f_{L/D,\,X=0}$, is calculated based on the data of Fauske [61]. The Fauske data give critical flow rates for *L/D* between 0 and 40, and pressures up to 2000 psig, for saturated water discharge through a tube with an inside diameter of 0.25 in.

The values of $f_{L/D,\,X=0}$ were tabulated using the data from [61], using the last point (*L/D* = 40) as a reference point. It is assumed that for *L/D* > 40 the critical flow is the same as for *L/D* = 40. Thus the tabulated values were calculated from:

$$f_{L/D,X=0} = \frac{G^{Fauske}_{c,L/D}}{G^{Fauske}_{c,L/D=40}}$$

$G_{c,L/D}^{Fauske}$ critical mass flux for the length-to-diameter ratio equal to *L/D*, taken from the Fauske data, (lb/ft^2-s)

$G_{c,L/D=40}^{Fauske}$ critical mass flux for the length-to-diameter ratio equal to 40, taken from the Fauske data, (lb/ft^2-s)

The values of saturated liquid multipliers, $f_{L/D,\,X=0}$, are shown in Table 4-4.

Table 4-4 The values of *L/D* multipliers for pure liquid

L/D	0.0	3.0	8.0	12.0	16.0	40.0
$f_{L/D}(X=0)$	2.46	1.77	1.16	1.08	1.05	1.00

The influence of the *L/D* ratio is observed only in case of pure liquid or low quality two-phase flow. Based on the experimental data shown in [57] the quality below which the *L/D* effect becomes visible has been estimated as 0.02.

The calculation of the multiplier, $f_{L/D}$, is performed as follows:

- Pure liquid, quality: $X \le 0.0$
  The multiplier is calculated using the linear interpolation of the tabulated data (Table 4-4).

- Low quality mixture, $0.0 < X < 0.02$
  The multiplier is calculated using the logarithmic, third order (smooth) interpolation. The interpolation is performed between the pure liquid value (for $X = 0.0$), and 1.0 (for $X = 0.02$):

$$f_{L/D} = f_{L/D,X=0} + (1.0 - f_{L/D,X=0})(3.0 - 2.0\lambda_{\log})\lambda_{\log}^2$$

  $\lambda_{\log}$ is a logarithmic interpolation factor, defined as:

$$\lambda_{\log} = \log_{10}(1.0 + 9.0\lambda_{lin}) \qquad 0.0 \le \lambda_{\log} \le 1.0$$

  λ_{lin} is a linear interpolation factor, defined as:

$$\lambda_{lin} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \qquad 0.0 \le \lambda_{lin} \le 1.0$$

  with $X_{\min}$ and $X_{\max}$ equal to 0.0 and 0.02, respectively. As shown in Volume 3, this interpolation allows to obtain good agreement with experimental data.

- High quality mixture, $X \ge 0.02$
  There is no influence of *L/D*. The multiplier, $f_{L/D}$, is set to 1.0.

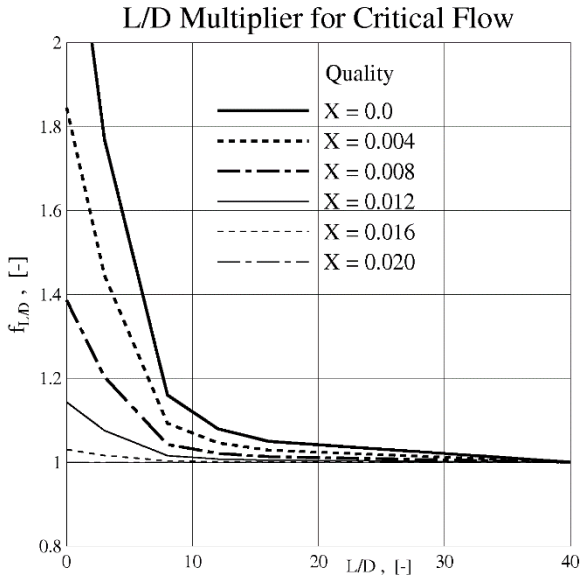


Figure 4-16 *L/D* multiplier, $f_{L/D}$

The values of the multiplier, $f_{L/D}$, calculated as described above, are plotted in Figure 4-16, for qualities between 0.0 and 0.02.

4.4.5 Slip Ratio, Component Velocities

The critical flow model calculates the overall mass flux through a critical junction, G_c. When the mass flux is known, individual velocities must be assigned to all components which may flow through a junction. The Junction Package distinguishes four components: atmosphere, droplets, pool, and bubbles. This section describes how the velocities of those components are determined in a critical junction.

First, it is assumed that a critical junction has only two velocities: the gas velocity and the liquid velocity. Thus, it is assumed that if both atmosphere and bubbles flow through the critical junction, their velocities are the same. Similarly, if both pool and droplets are present, then their velocities are the same:

$$v_{gas} = v_{atms} = v_{bubb}$$

$$v_{liq} = v_{pool} = v_{drop}$$

v_{gas} gas velocity, (m/s)
v_{liq} liquid velocity, (m/s)
v_{atms} velocity applied for the atmosphere flow in a junction, (m/s)
v_{bubb} velocity applied for the bubble flow in a junction, (m/s)
v_{pool} velocity applied for the pool flow in a junction, (m/s)
v_{drop} velocity applied for the droplet flow in a junction, (m/s)

The velocities v_{gas} and v_{liq} have to fulfill the following relation:

$$(A_{atms}\rho_{atms} + A_{bubb}\rho_{bubb})v_{liq} + (A_{pool}\rho_{pool} + A_{drop}\rho_{drop})v_{gas} = A_{JN}G_c$$

A_{JN} is the total flow area of the junction and G_c is the critical mass flux. The above equation must be fulfilled if the total flow through the junction has to be equal to the critical flow. To determine v_{gas} and v_{liq} one more equation is needed. It is assumed that the ratio between v_{gas} and v_{liq} (slip ratio) is known (it is calculated based on a modified Moody model, as shown below). Thus the second equation is:

$$\frac{v_{gas}}{v_{liq}} = S$$

Here S is the slip ratio, (-). The above two equations are solved for v_{gas}, v_{liq}, as follows. The areas of gas and liquid flows are defined as:

$$A_{gas} = A_{atms} + A_{bubb}$$
$$A_{liq} = A_{pool} + A_{drop}$$

Similarly the products: (area times density) for the gas and liquid flows, are defined as:

$$(A\rho)_{gas} = A_{atms}\rho_{atms} + A_{bubb}\rho_{bubb}$$
$$(A\rho)_{liq} = A_{pool}\rho_{pool} + A_{drop}\rho_{drop}$$

Using those definitions the equation set is written in somewhat shorter form:

$$(A\rho)_{gas}v_{gas} + (A\rho)_{liq}v_{liq} = A_{JN}G_c$$
$$v_{gas} = v_{liq}S$$

This equation set is solved for v_{gas}, v_{liq}, to give:

$$v_{gas} = \frac{A_{JN}G_c}{(A\rho)_{gas} + (A\rho)_{liq}/S}$$
$$v_{liq} = \frac{A_{JN}G_c}{(A\rho)_{liq} + (A\rho)_{gas}\cdot S}$$

The above equations are used to determine the individual velocities of components. The value of the slip ratio, S, is calculated using a modified Moody model. In the original Moody model the slip ratio is equal to:

$$S = \sqrt[3]{\rho_{liq}/\rho_{gas}}$$

The slip ratio is quite difficult to calculate. Several models have been proposed in the past: [58], [61], [53], etc. Reference [55] presents comparison of experimental data of Henry [56] with predictions of several models (see [55], section 11.4.3.2). Experiments indicate that S is a little below 10, decreases with increasing void fraction, α, and $S\rightarrow 1$ when $\alpha\rightarrow 1$ (see Figure 4-17). The Levy's model, $S\sim\alpha^{(1/2)}$, gives $S\rightarrow 1$ when $\alpha\rightarrow 1$. Both the Moody model, $S=(\rho_{liq}/\rho_{gas})^{(1/3)}$, and the Henry model, $S=(\rho_{liq}/\rho_{gas})^{(1/2)}$, give S independent of the void fraction. In the presented example the Moody model gives $S\approx 8$, which is in good agreement with the experiment for $\alpha\approx 0.3$. The Henry model gives $S\approx 23$, which is too high.

The comparison leads to a general conclusion that it is very difficult to obtain correct prediction of slip ratio. "The question of slip ratio is still unanswered" [55].

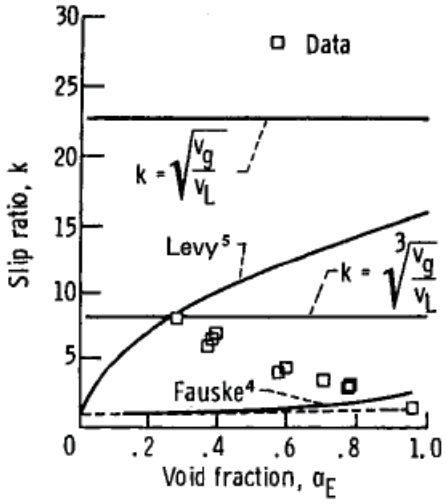


Figure 4-17 Comparison of experimental and theoretical slip ratios [55]

The method applied in SPECTRA is based on the Moody model. One reason for selecting the Moody slip ratio is to be consistent with the rest of the critical flow model, which is also based on the Moody theory. The other reason is the fact that, as can be observed in Figure 4-17, the model gives reasonably good prediction for the middle values of void fractions. For the extreme values of void fractions: $\alpha\rightarrow 0$ and $\alpha\rightarrow 1$, it is expected that the slip ratio should approach unity: $S\rightarrow 1$. This condition has to be fulfilled by the model, otherwise the calculated values of critical velocities will suffer strong discontinuities during transient calculations. Consequently the results will not only be difficult for physical interpretation but may be also difficult for numerical solution.

Therefore it was decided to modify the original Moody model, to obtain $S\rightarrow 1$ for $\alpha\rightarrow 0$ and $\alpha\rightarrow 1$, and thus allow "smooth calculations". A multiplier f is introduced and the slip ratio is defined as:

$$S = 1 + (S_{Moody} - 1)\cdot f$$

The function defining the multiplier *f* has been selected as follows:

$$f = 4Y \cdot (1-Y)$$

where:

$$Y = 2\alpha \cdot (1-\alpha)$$

With this formulation the value of *S* is practically equal to S_{Moody} in the range $0.3 < \alpha < 0.7$, and decreases smoothly to 1.0 for $\alpha \rightarrow 0$ and $\alpha \rightarrow 1$, as shown in Figure 4-18.

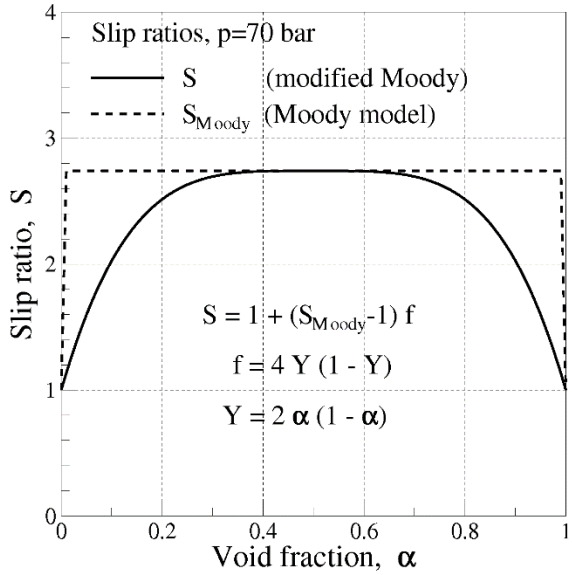


Figure 4-18 Slip ratios, Moody model and modified Moody model

Gas and liquid velocities, calculated with the Moody model and the modified model, are shown in Figure 4-19 and Figure 4-20 respectively. When the modified Moody model is used, both liquid and gas velocities are becoming equal for nearly single phase flows. Therefore:

$$v_{gas} \rightarrow v_{liq} \quad when \quad \alpha \rightarrow 0.0$$
$$v_{liq} \rightarrow v_{gas} \quad when \quad \alpha \rightarrow 1.0$$

This is clearly seen in Figure 4-20.

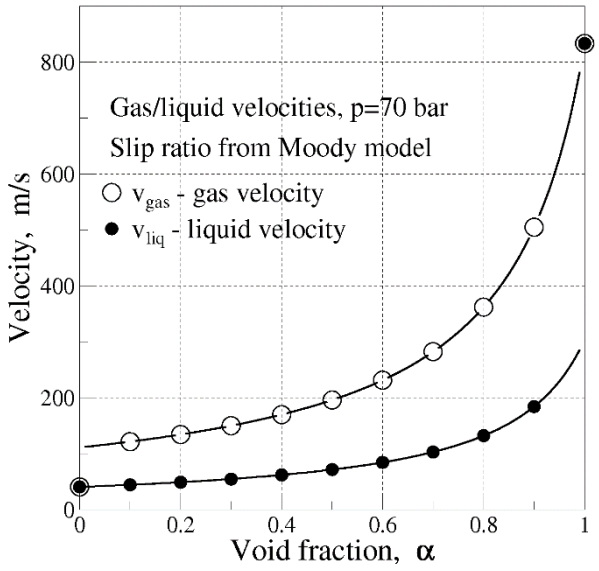


Figure 4-19 Gas and liquid velocities - slip ratio from the Moody model.

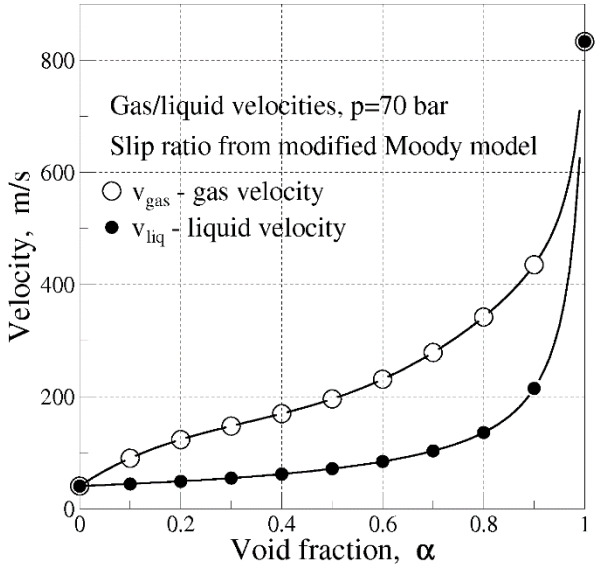


Figure 4-20 Gas and liquid velocities - slip ratio from the modified Moody model.

4.4.6 User-Defined Critical Flow

The critical flow model, described in the previous sections, is applicable for steam-water flow or gas mixtures. The user-defined flow, described in this section, is intended for alternative fluids (liquid metals) or sensitivity calculations. The critical mass flux is obtained from:

$$G_{c,gas} = \rho_{gas} \cdot c_{gas}$$
$$G_{c,liq} = \rho_{liq} \cdot c_{liq}$$

$G_{c,liq}$ critical mass flux for pure liquid, (kg/m^2-s)
$G_{c,gas}$ critical mass flux for pure gas, (kg/m^2-s)
ρ_{liq} density of liquid phase, (kg/m^3)
ρ_{gas} density of gas phase, (kg/m^3)
c_{liq} critical velocity of the liquid phase, (m/s)
c_{gas} critical velocity of the gas phase, (m/s)

The critical velocities must be defined by the user for the gas and the liquid phase. This is done using Tabular Functions giving the speed of sound as a function of the fluid temperature:

$$c_{gas} = f_{gas}(T_{gas})$$
$$c_{liq} = f_{liq}(T_{liq})$$

Here f_{gas} and f_{liq} are user-defined Tabular Functions (see Volume 2, record 262000). The following limits are imposed internally on the values obtained from the Tabular Functions:

$$1.0 \le c_{gas} \le 10^6$$
$$1.0 \le c_{liq} \le 10^6$$

In case of a two phase flow in a junction the procedure described in section 4.4.3 is applied:

$$\frac{\rho_{2\phi}}{G_{c,2\phi}} = \frac{(1-\alpha)\rho_{liq}}{G_{c,liq}} + \frac{\alpha\rho_{gas}}{G_{c,gas}} = \frac{(1-\alpha)}{c_{liq}} + \frac{\alpha}{c_{gas}}$$

α void fraction, (-)
$\rho_{2\varphi}$ mixture density, (kg/m^3), equal to: $\rho_{2\phi} = (1 - \alpha) \rho_{liq} + \alpha \rho_{gas}$

The final expression is:

$$G_{c,2\phi} = \frac{(1-\alpha) \cdot \rho_{liq} + \alpha \cdot \rho_{gas}}{(1-\alpha) / c_{liq} + \alpha / c_{gas}}$$

4.5 Momentum Length and Friction Length

SPECTRA, like many other system thermal-hydraulic codes (e.g. MELCOR, MAAP) distinguishes between the momentum (or inertial) length and the friction length. Both parameters must be defined by the user in the input deck. This section provides a short discussion on the meaning of those parameters and the methods to define them.

- Momentum length

The momentum length determines the inertia of the fluid. In general, the fluid acceleration term is given by *m dv/dt*, where *m* is the fluid mass (kg), *v* is the fluid velocity (m/s). The fluid mass is equal to m = ρV, where ρ is the fluid density (kg/m^3) and *V* is the fluid volume. The parameter that is used by the code is the momentum length, L_m, defined such that the product of the junction area *A* (m^2) and the momentum length L_m, is the volume of fluid that is being accelerated:

$$V = A \cdot L_m$$

This is the volume between the centers of the control volumes that are connected by the junction. The fluid acceleration term is in the code given by:

$$\rho \cdot A \cdot L_m \cdot \frac{dv}{dt}$$

The momentum length, L_m, is always equal to the sum of half-lengths of the connected fluid volumes, as shown below. As a simple example, we take the case 1, shown in Figure 4-21. The cross section area is the same for the junction and the two connected volumes. The fluid volume is equal to:

$$\frac{V_1 + V_2}{2} = \frac{A_1 L_1 + A_2 L_2}{2}$$

Since $A_1 = A_2 = A$, the fluid volume between the CV centers is equal to:

$$A \cdot \frac{L_1 + L_2}{2}$$

Since the fluid volume is in the code defined as $A \cdot L_m$, we have:

$$L_m = \frac{L_1}{2} + \frac{L_2}{2}$$

A more general case is shown in the case 2 in Figure 4-21. In this case each flow area is different. Furthermore, there is a small part length with the junction area. This could be for example a thick orifice or wall thickness in case JN-1 represents door, hatch, etc. Since JN represents an opening and does not by itself contain any volume, this volume (*V*) must be included in one or both Control Volumes that are connected by this junction.

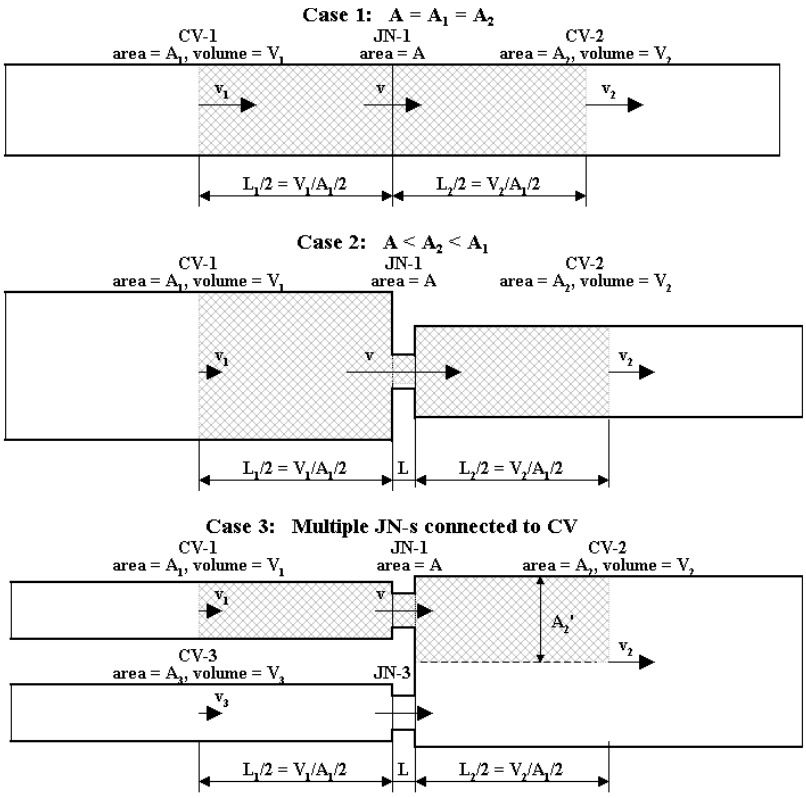


Figure 4-21 Momentum length and friction length of a Junction

In this case fluids are accelerated differently in CV-1 and CV-2 and in the junction region. The fluid acceleration term is given by:

$$\rho \cdot \left(V \frac{dv}{dt} + \frac{V_1}{2} \frac{dv_1}{dt} + \frac{V_2}{2} \frac{dv_2}{dt} \right) = \rho \cdot \left(AL \frac{dv}{dt} + \frac{A_1 L_1}{2} \frac{dv_1}{dt} + \frac{A_2 L_2}{2} \frac{dv_2}{dt} \right)$$

Taken that $A_1 v_1 = A_2 v_2 = A v$, we have:

$$\rho \cdot \left(AL \frac{dv}{dt} + \frac{AL_1}{2} \frac{dv}{dt} + \frac{AL_2}{2} \frac{dv}{dt} \right) = \rho \cdot A \cdot \left(L + \frac{L_1}{2} + \frac{L_2}{2} \right) \frac{dv}{dt}$$

Therefore:

$$\rho \cdot A \cdot L_m \cdot \frac{dv}{dt} = \rho \cdot A \cdot \left(L + \frac{L_1}{2} + \frac{L_2}{2} \right) \frac{dv}{dt}$$

and:

$$L_m = L + \frac{L_1}{2} + \frac{L_2}{2}$$

Finally, let's consider a more general case, with multiple junctions connected to a single control volume. The case is shown as Case 3 in Figure 4-21. The acceleration term for the junction JN-1 is written by assuming an imaginary line, shown in Figure 4-21 as dashed line. The fluid acceleration term is given by:

$$\rho \cdot \left(AL\frac{dv}{dt} + \frac{A_1 L_1}{2}\frac{dv_1}{dt} + \frac{A_2' L_2}{2}\frac{dv_2}{dt} \right)$$

The position of the imaginary division line depends on relative flows in the JN-1 and the junction connecting CV-3. Quite independently of the location of the division line, we have a general relation: $A_1 v_1 = A_2' v_2 = A v$. Therefore:

$$\rho \cdot \left(AL\frac{dv}{dt} + \frac{AL_1}{2}\frac{dv}{dt} + \frac{AL_2}{2}\frac{dv}{dt} \right) = \rho \cdot A \cdot \left(L + \frac{L_1}{2} + \frac{L_2}{2} \right)\frac{dv}{dt}$$

The final result is:

$$L_m = L + \frac{L_1}{2} + \frac{L_2}{2}$$

We may conclude that, independently of the geometry, the momentum length is always determined by the above formula. Assuming that the small volume at the JN is negligible, compared to the volumes of the connected Control Volumes, this may be written using volumes and CV flow areas:

$$L_m = L + \frac{V_1}{2A_1} + \frac{V_2}{2A_2}$$

- Friction length

The friction length is used in the friction pressure drop calculation:

$$\Delta P = f\frac{L_f}{D}\frac{\rho v^2}{2}$$

Let's consider first the case 1, shown in Figure 4-21. The pressure loss from volume center to volume center is given by:

$$\Delta P = f\frac{L_1}{2D_1}\frac{\rho v_1^{\,2}}{2} + f\frac{L_2}{2D_2}\frac{\rho v_2^{\,2}}{2}$$

In this case $A = A_1 = A_2$ and $v = v_1 = v_2$, therefore:

$$\Delta P = f\frac{L_1}{2D}\frac{\rho v^2}{2} + f\frac{L_2}{2D}\frac{\rho v^2}{2} = \frac{f}{D}\left(\frac{L_1}{2} + \frac{L_2}{2} \right)\frac{\rho v^2}{2}$$

Therefore:

$$L_f = \frac{L_1}{2} + \frac{L_2}{2}$$

Next, let's consider the more general case 2 in Figure 4-21. In this case each flow area is different. The friction loss term:

$$\Delta P = f\,\frac{L}{D}\,\frac{\rho v^2}{2} + f\,\frac{L_1}{2D_1}\,\frac{\rho v_1^{\,2}}{2} + f\,\frac{L_2}{2D_2}\,\frac{\rho v_2^{\,2}}{2}$$

is rewritten using:

$$v_1 = v\cdot\frac{A}{A_1} \qquad v_2 = v\cdot\frac{A}{A_2}$$

We obtain:

$$\Delta P = f\,\frac{L}{D}\,\frac{\rho v^2}{2} + f\,\frac{L_1}{2D_1}\left(\frac{A}{A_1}\right)^2\frac{\rho v^2}{2} + f\,\frac{L_2}{2D_2}\left(\frac{A}{A_2}\right)^2\frac{\rho v^2}{2}$$

If we multiply the numerator and the denominator by the junction diameter, *D*, we obtain:

$$\Delta P = \frac{f}{D}\left[L + \frac{L_1}{2}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{L_2}{2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{A}{A_2}\right)^2\right]\cdot\frac{\rho v^2}{2}$$

Consequently, the friction length is in this case given by:

$$L_f = L + \frac{L_1}{2}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{L_2}{2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{A}{A_2}\right)^2$$

This formula, written using volumes and CV flow areas, gives for the **Case 2:**

$$L_f = L + \frac{V_1}{2A_1}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{V_2}{2A_2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{A}{A_2}\right)^2$$

Finally, we consider the case 3 in Figure 4-21. The solution is the same as above, except that A_2 is replaced by A_2'. Therefore:

$$L_f = L + \frac{L_1}{2}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{L_2}{2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{A}{A_2'}\right)^2$$

Here the solution depends on the imaginary dividing line and therefore, the exact solution cannot be found in general. Assuming the same flow in both junctions of the case 3, we have $A_2' \approx A_2/2$. A practically important case in *N* channels entering the reactor plenum. In such case $A_2' \approx A_2/N$, and the formula for **Case 3** is:

$$L_f = L + \frac{V_1}{2A_1}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{V_2}{2A_2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{NA}{A_2}\right)^2$$

The user may select an option to calculate the momentum lengths and the friction lengths by the code (see Volume 2, input parameter IMFLJN). In such case the code uses the following formulae (Case 3 is not considered):

$$L_m = L + \frac{V_1}{2A_1} + \frac{V_2}{2A_2}$$

$$L_f = L + \frac{V_1}{2A_1} \cdot \left(\frac{D}{D_1}\right) \cdot \left(\frac{A}{A_1}\right)^2 + \frac{V_2}{2A_2} \cdot \left(\frac{D}{D_2}\right) \cdot \left(\frac{A}{A_2}\right)^2$$

Example 1:

In this example, junction JN-001 connects two pieces of identical pipe; each has length of 0.5 m and diameter of 0.1 m. This situation is that of case 1 in Figure 4-22 (a). Therefore, the momentum length and the friction length are calculated from the following equations:

$$L_m = \frac{L_1}{2} + \frac{L_2}{2} = \frac{0.5}{2} + \frac{0.5}{2} = 0.5$$

$$L_f = \frac{L_1}{2} + \frac{L_2}{2} = \frac{0.5}{2} + \frac{0.5}{2} = 0.5$$

In this case both the momentum length and the friction length are equal.

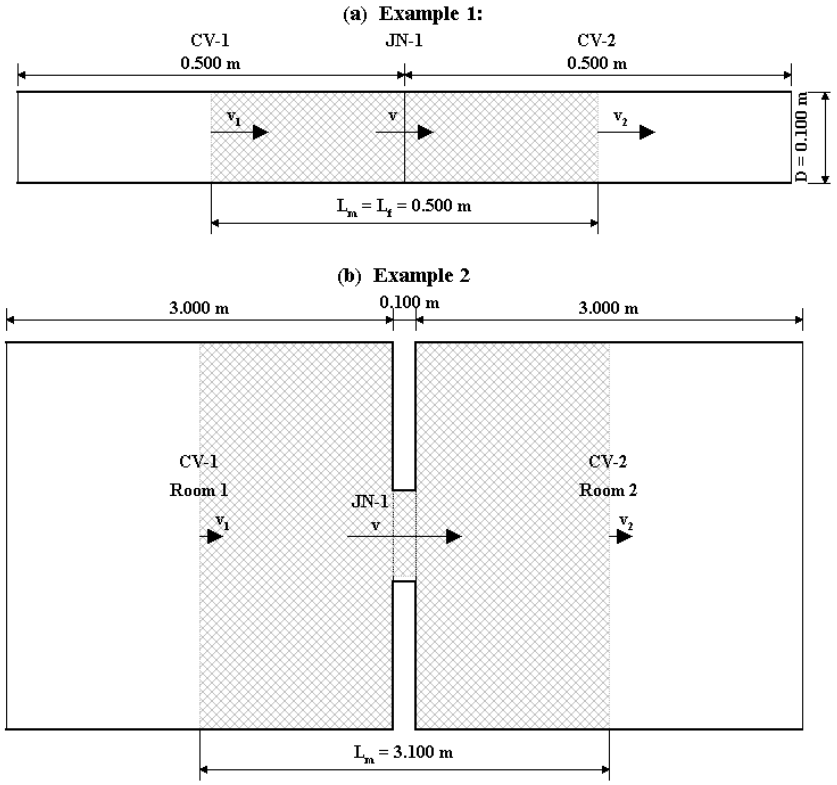


Figure 4-22 Geometry of (a) Example 1, (b) Example 2

Example 2:

In the example, junction JN-001 represents a door between control volumes CV-001 and CV-002. The door is 1 m wide and 2 m high. The flow area is 2.0 m^2. The wall thickness is 0.1 m. The two connected rooms are 3 m by 3 m area and are 3 m height. Therefore the cross section area is 9 m^2 and the volume is 27 m^3. This situation is that of case 1 in Figure 4-22 (b). Therefore the momentum length and the friction length are calculated from the following equations:

- Momentum length:

$$L_m = L + \frac{V_1}{2A_1} + \frac{V_2}{2A_2}$$

The length of the opening is equal to the wall thickness, L = 0.1 m. The volumes are $V_1 = V_2$ = 27.0 m^3. The momentum length is equal to:

$$L_m = 0.1 + \frac{27.0}{2\times 9.0} + \frac{27.0}{2\times 9.0} = 0.1 + 1.5 + 1.5 = 3.1$$

- Friction length:

$$L_f = L + \frac{V_1}{2A_1}\cdot\left(\frac{D}{D_1}\right)\cdot\left(\frac{A}{A_1}\right)^2 + \frac{V_2}{2A_2}\cdot\left(\frac{D}{D_2}\right)\cdot\left(\frac{A}{A_2}\right)^2$$

The hydraulic diameter of the control volumes are equal to $D_1 = D_2 = 4\times 9.0/12.0 = 3.0$ m, where 12.0 is the wetted perimeter of the rooms (control volumes). The hydraulic diameter of the junction is equal to $D = 4\times 2.0/6.0 = 1.33$ m, where 6.0 is the wetted perimeter of the door (junction). The friction length is equal to:

$$L_f = 0.1 + \frac{27.0}{2\times 9.0}\cdot\left(\frac{1.33}{3.0}\right)\cdot\left(\frac{2.0}{9.0}\right)^2 + \frac{27.0}{2\times 9.0}\cdot\left(\frac{1.33}{3.0}\right)\cdot\left(\frac{2.0}{9.0}\right)^2 =$$
$$= 0.1 + 0.033 + 0.033 = 0.166$$

It is seen that in this case:

- the major contribution to the momentum length is coming from the control volumes,
- the major contribution to the friction length is coming from the junction.

4.6 Additional Models

Several additional models are available in CV Junction Package. In order to model in a simple way forced flow (pumps, compressors), junctions with user defined flow may be used. Such junctions are described in section 4.6.1. For a more detailed analysis of pumps, compressors, fans, etc., a pump model is available, described in section 4.6.2. A turbine model is described in section 4.6.3. A valve model, described in section 4.6.5, allows to model check valves, motor operated valves, etc. Finally, the flow composition parameters, described in section 4.6.6, may be used to limit the flow of one or several gases in a gas mixture.

4.6.1 User-Defined Flow

Flows through any junction may be defined by the user, either as a Tabular Function of time, or as a Control Function of any argument in the program data base. The flows of atmosphere and pool, W_{atms}, W_{pool}, are defined independently, by different Tabular or Control Functions. The velocities of these components are calculated as:

$$v_{atms} = \frac{W_{atms}}{\rho_{atms} A_{atms}}$$

$$v_{pool} = \frac{W_{pool}}{\rho_{pool} A_{pool}}$$

where ρ_{comp} and A_{comp} are the densities and flow areas of the components in the control volume where the flow originates. The component flow areas are calculated based on the junction water level. The water level in the junction is assumed to be equal to the water level in the control volume where the pool flow originates. If the flow area of a given component is zero, then also the flow is equal to zero, independently of the value specified by tabular or control function. In other words, there will be no pool flow if the water level in the source CV is below the junction lower edge elevation, and no atmosphere flow if the level is above the upper edge elevation. To avoid discontinuities in the model, an interpolation zone is defined, so that the flow decreases smoothly to zero, when the component flow area decreases to zero. This is done as follows:

$$W_{comp} = \begin{cases} W_{TF,CF} & if \quad A_{comp} > A_{\min} \\ W_{TF,CF} \cdot \left(\dfrac{A_{comp}}{A_{\min}} \right)^2 & if \quad A_{comp} < A_{\min} \end{cases}$$

The subscript *TF,CF* means the value obtained from tabular or control function. $A_{\min}$ is the minimum area for which actual flow is still equal to that obtained from tabular or control function. The value of $A_{\min}$ is set to 10% of the full junction flow area.

With the above definition of mass flows, the junction velocities are equal to:

$$v_{comp} = \begin{cases} \dfrac{W_{TF,CF}}{\rho_{comp} A_{comp}} & if \quad A_{comp} > A_{\min} \\ \dfrac{W_{TF,CF} A_{comp}}{\rho_{comp} A_{\min}^2} & if \quad A_{comp} < A_{\min} \end{cases}$$

The velocities decrease linearly to zero with component flow area decreasing to zero. The dependence of velocity on the component flow area is illustrated in Figure 4-23.

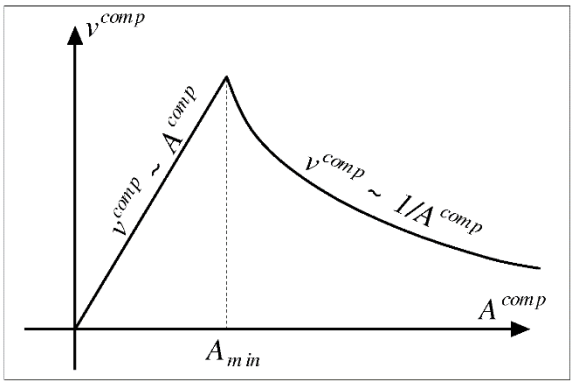


Figure 4-23 Velocity versus flow area for a given component.

The velocity calculation is illustrated by the following example problem. The model consists of two control volumes, CV-001 and CV-002, as shown in Figure 4-24. The volumes are connected by a horizontal junction, with an opening height of 1.0 m (elevations between 4.5 and 5.5 m). The initial water level in CV-001 is 4.0 m, 0.5 below the lower edge of junction JN-001. A mass source is present in CV-001, which adds 100 kg/s of water into CV-001.

The junction JN-1 is a user defined junction with flows specified by Tabular Functions, TF-001 and TF-002, as constants, equal to: W_{atms} = TF-001 = –0.1 kg/s, W_{pool} =TF-002 = 10.0 kg/s.

Results are shown in Figure 4-24, Figure 4-25, and Figure 4-26. Initially the water level in CV-001 is too low, and there is no pool flow through JN-001 although the value imposed by the Tabular Function is 10 kg/s. Due to the mass source of 100 kg/s, the water level in CV-001 rises, and at about 16 s the water level reaches bottom of JN-001 and pool flow begins. For about 1 s the flow is governed by the low component flow area interpolation, described above. At about 17 s the atmosphere flow area is greater than A_{min}, and the full flow of 10 kg/s is observed. About 10 seconds later the pool level in CV-001 approaches the top of JN-001. Initially the atmosphere velocity increases, to provide the requested atmosphere flow in spite of the decreasing atmosphere flow area. When the atmosphere flow area reaches A_{min}, both atmosphere velocity and flow decrease, to reach zero when the junction is completely covered by water on the CV-001 side, and the atmosphere flow area is equal to zero.

Note that the definition of the junction pool level in a user defined junction is simplified, compared to the "normal" junction. Therefore when the pool level in CV-001 is above the junction top elevation, then no atmosphere flow occurs, although in the normal junction it would still be possible, and would be determined by the counter-current flow limit, as described in section 4.2.4 (Figure 4-4.f).

The above discussion was related to the flow of continuous components. The dispersed components are also transported through junctions with user defined flow, if they are present upstream the junction. The transport of dispersed components is performed using the drift flux equation, the same as in case of regular junctions (section 4.2.3). Thus, when the velocities of the continuous components are determined using the method described above, the velocities of the dispersed components are obtained from:

$$v^{Dcomp} = \frac{C_0^{Dcomp} v^{Ccomp}(1-\alpha^{Dcomp}) + v_\infty^{Dcomp}(1-\alpha^{Dcomp})^{x^{Dcomp}}}{1-C_0^{Dcomp}\alpha^{Dcomp}}$$

where the superscript *Dcomp* means dispersed component (droplets, bubbles), while the superscript *Ccomp* means the corresponding continuous component (atmosphere, pool). The methods of determining the drift model constants, $v_\infty^{Dcomp}, C_0^{Dcomp}, x^{Dcomp}$, are described in section 2.6.

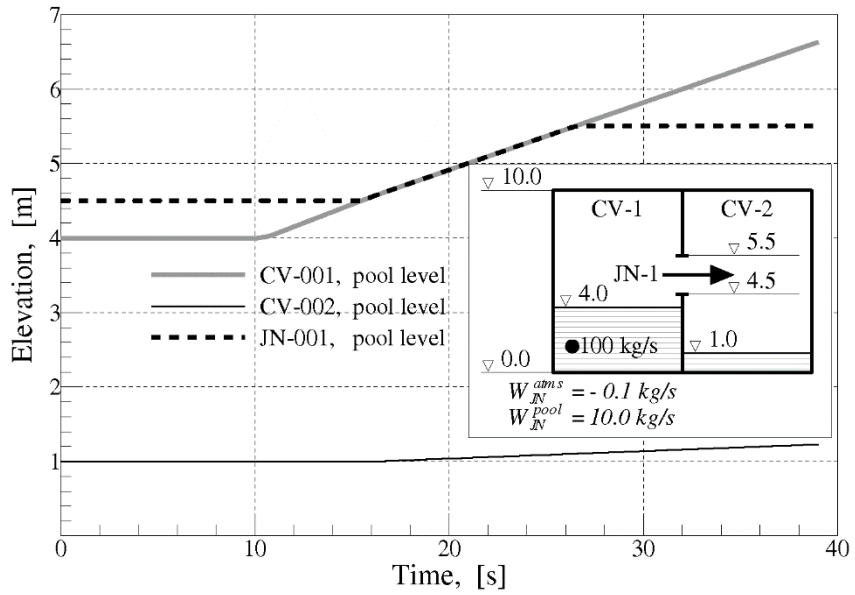


Figure 4-24 Water levels, JN with user-defined flow

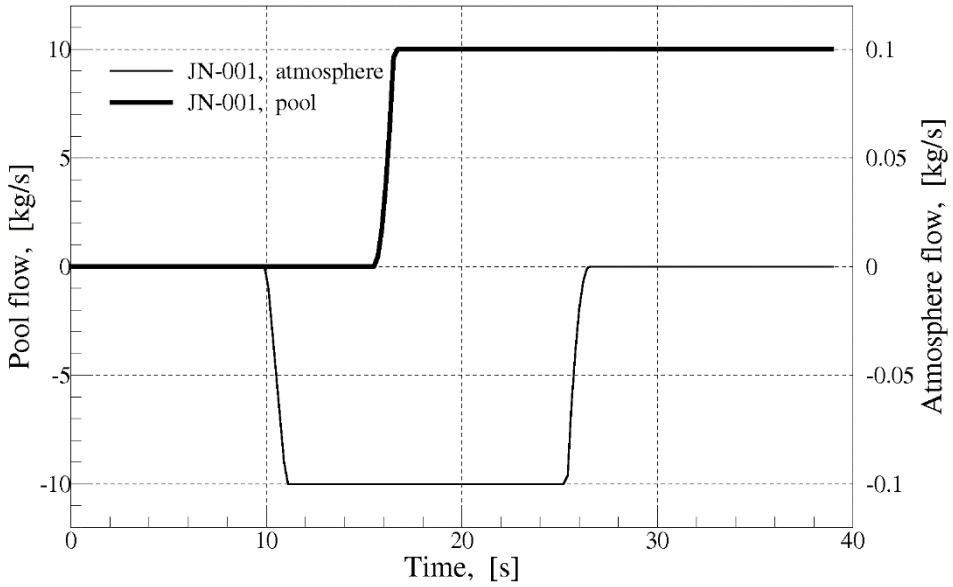


Figure 4-25 Mass flows of atmosphere and pool, JN with user-defined flow

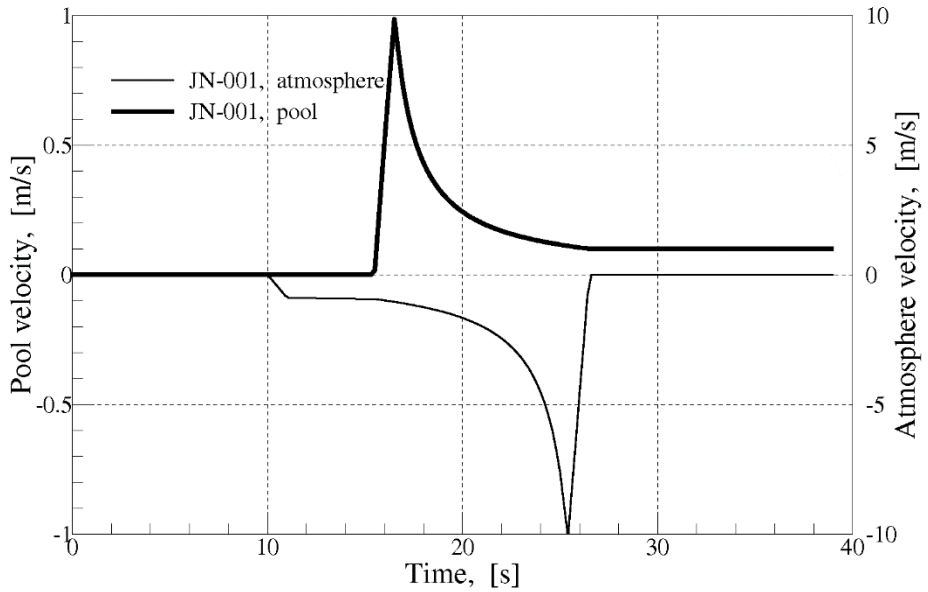


Figure 4-26 Velocities of atmosphere and pool, JN with user-defined flow

4.6.2 Pump/Compressor Model

Two models are available for pumps/compressors in SPECTRA. The first model (Type 1) simply sets the flow in junction based on the pressure difference across the junction and the user-defined maps. The second model (Type 2) calculates the pressure head based on the user-defined maps and then uses the general momentum equation to calculate the flow. Type 1 is seldom used. This was the first pump model applied in an early version of SPECTRA, and it is preserved in the code to keep compatibility with earlier input decks. Type 2 is generally recommended, because it is physically correct, while Type 1 is only an approximation valid for quasi-stationary conditions. Moreover Type 2 offers several additional capabilities, like:

- More flexible map generation (there are more map defining parameters that can be specified by the user - exponents *a, b, c*).
- Internal power calculation. When Type 1 is used, the user must define Control Functions to define power entering the fluid. A set of recommended functions had been provided in the manual to an early SPECTRA version. When Type 2 is used, the power that enters the fluid is calculated internally by the code, using efficiency tables.
- Speed can be calculated using a rotor inertia equation.

The discussion below explains how a pump/compressor map is defined using relatively few input parameters. The pump maps are discussed basically for the Type 2 pumps/compressors. The Type 1 model has the same maps, except that the exponents *a, b, c* have fixed values, and cannot be changed by the user.

4.6.2.1 Approach

The basic approach in the SPECTRA code is quite different than in codes such as RELAP, TRAC, etc. These codes require the user to provide data tables specifying maps for all possible conditions (four quadrant curves - see [45]). Alternatively, they offer built-in curves for typical pumps (Bingham Westinghouse pump curves are available in RELAP).

In general, it is quite difficult to find sufficient data to determine the four quadrant curves; specifically the data for reverse speed and reverse flow are typically not available from the manufacturer. The required input data is extensive; the user must provide a sufficient amount of data to cover all possible conditions, even if some conditions are never expected to occur. Consequently users of RELAP or TRAC typically prefer to use the built-in maps. Thus the modelling is either tedious or not flexible.

The pump/compressor model in SPECTRA was designed to provide flexible modelling, while simultaneously limiting the amount of effort in input data preparation. The pump/compressor maps are approximated by a standard equation, which is built into the code. The coefficients in this equation are defined by the user. The user has therefore to define only a few input parameters that will provide the full pump/compressor map.

The pump/compressor map is approximated in SPECTRA by a parabolic equation. The shape of map depends on the user-defined parameters, such as V_0, C_{RS}, C_{RF}, C_{DS}, C_{VS} and exponents *a, b, c*. The user-defined parameters and their effect on the pressure head are illustrated in Figure 4-27.

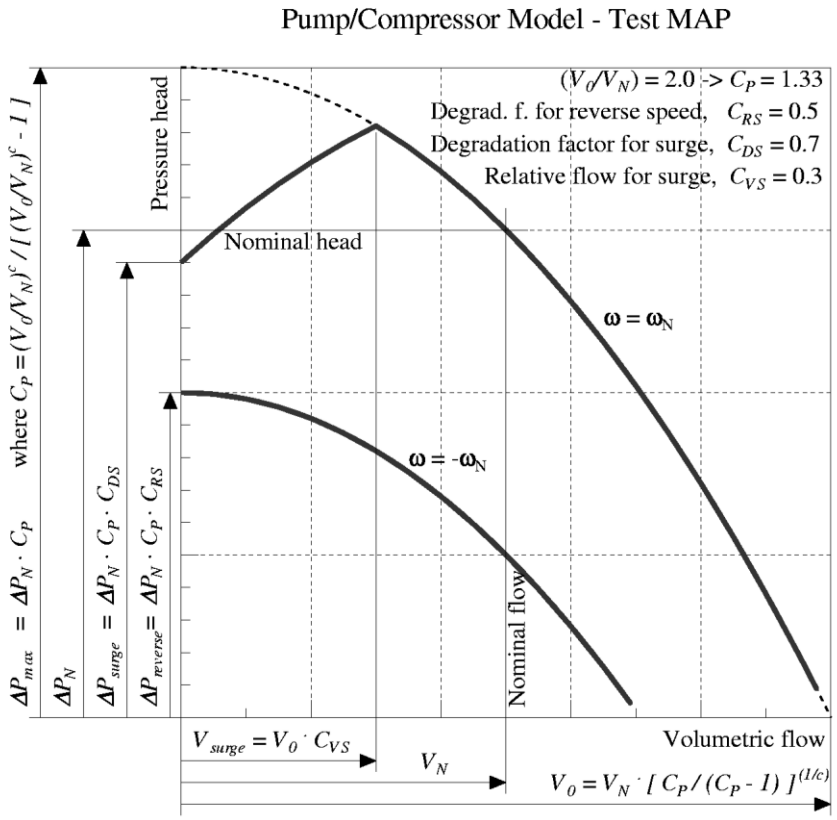


Figure 4-27 Influence of input parameters: C_P, C_{RS}, C_{DS}, C_{VS}, on the pump map.

Note that in the previous code versions C_P was the pump/compressor input parameter, instead of V_0. This was changed in order to make the pump/compressor input parameters the same as the turbine input parameters. There is a simple relation between C_P and V_0. The relation is shown in the section describing the input parameters (Volume 2), so the user can easily convert old input decks.

The user must in each case check his maps. This is done quite simply. A test run is set up, with a pump in between two volumes with time dependent conditions. The conditions should be selected in such a way as to force the pump to operate in the range that needs to be plotted to create a full map. Several identical pumps may be set, each operating with different speed. The resulting flow and pressure head (or pressure ratio) of each pump are then plotted. Instead of plotting them against time (*V=f(t), ΔP=f(t)*), they are plotted against each other (*ΔP=f(V)*). In this way maps with curves for several different pump/compressor speeds are obtained. Such maps are discussed in section 4.6.2.3. Examples of input files for creating such maps are given in Volume 3.

Section 4.6.2.2 shows definitions of reduced (dimensionless) parameters, which are being used with the pump/compressor maps. The discussion of the pump/compressor maps is provided in two parts. In the first part (section 4.6.2.3) an "ideal" map is described, that is a map based on parabolic equation, in which surge does not appear (surge degradation factor, $C_{DS} = 1.0$). The second part (section 4.6.2.4) describes how the surge is modelled. Finally section 4.6.2.5 shows how the pump/compressor power is calculated.

4.6.2.2 Reduced Parameters

The reduced (dimensionless) parameters, which are being used with the pump/compressor maps, are:

- Reduced speed, ω_R,
- Reduced flow, V_R.
- Reduced pressure head P_R, or reduced pressure ratio, Π_R.

The user has several options as to how these parameters are defined. The available definitions are discussed below.

Reduced speed, ω_R

The reduced speed ω_R, is calculated based on the user defined nominal speed, ω_N, (s^{-1}), and eventually the nominal temperature T_N, (K). If the nominal temperature, T_N, is not specified, then the reduced speed is defined as:

$$\omega_R = \left(\frac{\omega}{\omega_N} \right)$$

If a nominal temperature, T_N, is specified, then the reduced speed is defined as:

$$\omega_R = \frac{\left(\frac{\omega}{\omega_N} \right)}{\sqrt{\frac{T}{T_N}}}$$

In the above definitions ω is the pump speed and T is the temperature at the inlet to the pump/compressor. The first definition is appropriate for typical pumps (see for example [62]), while the second is appropriate for typical gas compressors (see for example [63]).

Reduced flow, V_R

The reduced flow V_R, is calculated based on the user defined nominal volumetric flow, V_N, (m^3/s) and eventually the nominal temperature T_N, the nominal fluid velocity, v_N, the nominal gas constant, R_N. Several options are available.

- Option 1
  If only the nominal flow, V_N, is entered, then the reduced flow is defined as:

$$V_R = \left(\frac{V}{V_N} \right)$$

- Option 2
  If a nominal temperature, T_N, is specified, then the reduced flow is defined as:

$$V_R = \frac{\left(\frac{V}{V_N}\right)}{\sqrt{\frac{T}{T_N}}}$$

- Option 3
  If a nominal fluid velocity, v_N, is specified, then the reduced flow is defined as:

$$V_R = \frac{\left(\frac{V}{V_N}\right)}{\sqrt{\frac{T}{T_N}}} \quad \frac{\sqrt{\frac{1+v^2/2c_pT}{1+v_N^2/2c_pT_N}}}{\left(\frac{1+v^2/2RT}{1+v_N^2/2RT_N}\right)}$$

- Option 4
  Finally, if a nominal gas constant, R_N, is specified, then the reduced flow is defined as:

$$V_R = \frac{\left(\frac{V}{V_N}\right)}{\left(\frac{R}{R_N}\right)\sqrt{\frac{T}{T_N}}} \quad \frac{\sqrt{\frac{1+v^2/2c_pT}{1+v_N^2/2c_pT_N}}}{\left(\frac{1+v^2/2RT}{1+v_N^2/2RT_N}\right)}$$

In the above definitions V is the pump volumetric flow, T is the gas temperature at the inlet to the pump, v is the fluid velocity in the pump/compressor. R and c_p are the perfect gas constant and the specific heat at constant pressure for the gas mixture flowing through the pump/compressor.

The first definition (option 1) is appropriate for typical pumps (see for example [62]). The last definition (option 4) is appropriate for typical gas compressors, as will be shown below. The intermediate definitions are approximations of the exact formula for compressors.

In case of gas compressors, the maps are frequently defined using so called, "corrected flow", W_{corr}, defined as (see for example [63]):

$$W_{corr} = W\frac{\sqrt{T_{tot}}}{p_{tot}}$$

Where W is the mass flow, T_{tot} is the total temperature (based on the enthalpy, $h=c_p\,T$, and the kinetic energy, $v^2/2$), and p_{tot} is the total pressure (static, p, and dynamic, $\rho v^2/2$). The above formula can be transformed as follows:

$$W\frac{\sqrt{T_{tot}}}{p_{tot}} = W\frac{\sqrt{T+v^2/2c_p}}{p+\rho v^2/2} = W\frac{\sqrt{T}}{p}\frac{\sqrt{1+v^2/2c_pT}}{1+\rho v^2/2p} =$$

$$= \rho V\frac{\sqrt{T}}{p}\frac{\sqrt{1+v^2/2c_pT}}{1+\rho v^2/2p} = \frac{V}{R\sqrt{T}}\frac{\sqrt{1+v^2/2c_pT}}{1+v^2/2RT}$$

In the above transformation use was made of the perfect gas law: $p = \rho RT$, and the relation $W=\rho V$. The above formula is written in SPECTRA in a dimensionless form, by replacing each dimensional parameter by the ratio of the parameter and it's nominal value. The result is:

$$\frac{\left(\frac{V}{V_N}\right)}{\left(\frac{R}{R_N}\right)\sqrt{\frac{T}{T_N}}} \quad \frac{\sqrt{\frac{1+v^2/2c_pT}{1+v_N^2/2c_pT_N}}}{\left(\frac{1+v^2/2RT}{1+v_N^2/2RT_N}\right)}$$

The above formula is identical to the last formula for V_R. Therefore the option 4 gives a dimensionless form of the corrected flow, W_{corr}.

If the gas composition does not change during calculations, then the gas constant R is always equal to R_N, and the gas constant ratio can be removed from the formula. Therefore option 3 is accurate as long as the gas composition does not change. If additionally the dynamic terms can be neglected compared to the static terms, then the terms with velocity squared disappear and option 2 may be used. Summarizing, the following options may be used to calculate the reduced flow:

- Option 1, appropriate for pumps,
- Option 2, appropriate for compressors with no gas composition changes and dynamic terms neglected.
- Option 3, appropriate for compressors with no gas composition changes.
- Option 4, appropriate for compressors without any restrictions and simplifications.

Reduced pressure head P_R, or reduced pressure ratio, Π_R

The user can define either the nominal pressure head (Pa), or the nominal pressure ratio, (-). If the nominal pressure head, ΔP_N, is specified, then the pump/compressor map uses the reduced pressure head, ΔP_R, defined as:

$$\Delta P_R = \left(\frac{\Delta P}{\Delta P_N}\right)$$

If the nominal pressure ratio, Π_N is specified, then the pump/compressor map uses the reduced pressure ratio, Π_R, defined as:

$$\Pi_R = \left(\frac{\Pi}{\Pi_N}\right)$$

In the above definitions ΔP and Π are the pressure head and the pressure ratio respectively. If the pressure ratio is used, then the reduced pressure head, ΔP_R, is replaced by $(\Pi–1)/(\Pi_R–1)$.

4.6.2.3 Pump/Compressor Maps - Ideal Maps with no Surge

Definitions and detailed descriptions of all parameters are provided in the User's Guide (Volume 2). Below a short description is provided how each parameter affects the shape of the pump/compressor pressure head. The pump/compressor model is based on the map, which is approximated by the following equation:

$$\Delta P_R = C_{RS} C_P \ \omega_R^a \mp C_{RF} \left(C_P - 1\right) \omega_R^b V_R^c$$

ΔP Reduced pressure head, (-). If the nominal pressure ratio, Π_N, is specified, the nominal pressure head is equal to: $\Delta P_N(t)=(\Pi_N-1)\cdot P_0(t)$, where $P_0(t)$ is the current pressure upstream the pump. In this case the reduced pressure head, ΔP_R, in the above formula is replaced by $(\Pi-1)/(\Pi_N-1)$, where $\Pi=(\Delta P/P_0+1)$, ΔP is the actual pressure head in (Pa), and Π_N is the nominal pressure ratio (input parameter) that is related to the nominal pressure head, $\Delta P_N(t)$ (time-dependent) by $\Pi_N = \Delta P_N(t)/P_0(t) + 1$

V_R Reduced pump flow, (-)

ω_R Reduced pump speed, (-).

C_P Constant defining the shape of pressure head in pump map, (-), $C_P > 1.0$. The value of C_P is related to the input parameters V_N (VFNPJN), and V_0 (VF0PJN), as: $C_P = (V_0/V_N)^c / ((V_0/V_N)^c - 1)$

C_{RS} Degradation factor for reverse speed, (-)
$\omega > 0.0$: $C_{RS} = 1.0$
$\omega < 0.0$: C_{RS} is given by a user-defined parameter

C_{RF} Degradation factor for reverse flow, (-)
$V > 0.0$: $C_{RF} = 1.0$
$V < 0.0$: C_{RF} is given by a user-defined parameter

a First exponent (default value of 2.0).

b Second exponent (default value of 1.0).

c Third exponent (default value of 2.0).

In the above equation the sign is – in case of positive flow, and + in case of negative flow. The parameters V_0, C_{RS}, C_{RF}, a, b, c, are user-defined. The influence of these parameters on the shape of pump map is discussed below.

Influence of C_P (V_0/V_N) on the pump map is shown in Figure 4-28. The following nominal parameters were used to make the plot.

ΔP_N = 1.0×10^5 Pa
V_N = 1.0 m^3/s
ω_N = 50.0 s^{-1}

Figure 4-28 shows pressure heads made for values of V_0/V_N between 10.0 (corresponding C_P of 1.01) and 1.01 (corresponding C_P of 50.8). The corresponding values of C_P, calculated for the default value of the exponent c (c=2.0) are shown below:

V_0/V_N = 10.0	C_P = 1.01	
V_0/V_N = 3.3	C_P = 1.10	
V_0/V_N = 2.0	C_P = 1.33	(typical water pump)
V_0/V_N = 1.2	C_P = 3.27	(typical gas compressor)
V_0/V_N = 1.01	C_P = 50.8	(piston pump)

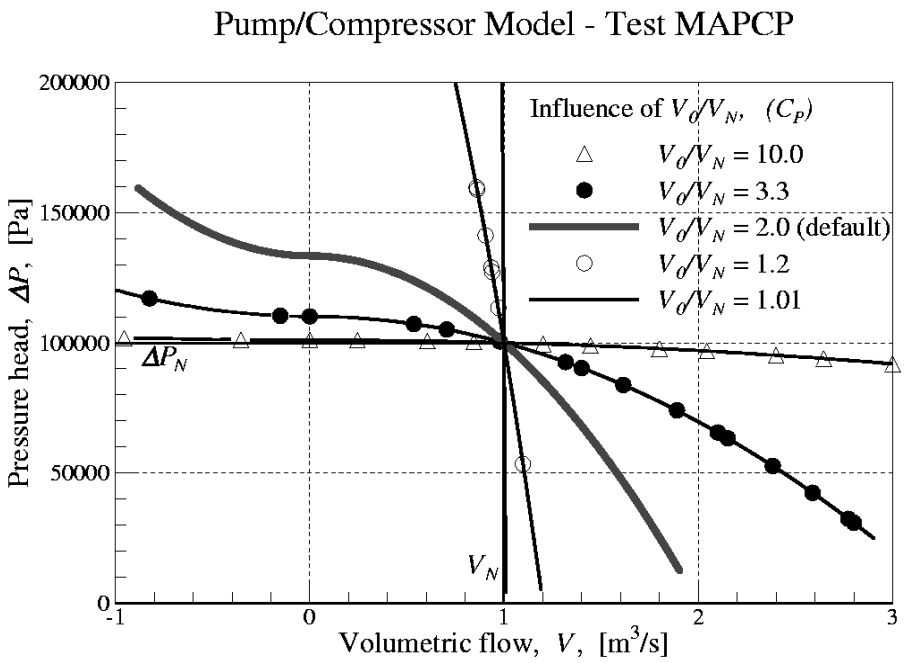


Figure 4-28 Influence of the parameter V_0, on the pump maps

It is seen that when V_0/V_N is very small (V_0/V_N=1.01) then the pressure head makes a nearly vertical line. This means that the pump flow is nearly constant, independently of the pressure head. This shape is typical for the piston pumps. When V_0/V_N is very large, the pressure head line becomes flat. (If the Type 1 pump is used, then very large values of V_0/V_N may also cause numerical solution problems - the flow is quite sensitive to any pressure changes.).

The influence of the degradation factor for reverse flow, C_{RF}, on the pump map is illustrated in Figure 4-29. The value of C_{RF} affects the pump map only in the reverse flow range. If the default value is used (C_{RF} = 1.0) then the pressure head lines in the negative flow range are symmetrical to the lines in the positive flow range. If C_{RF} < 1.0 then the pressure head lines in the negative flow are flatter than in the positive flow. If C_{RF} > 1.0 then the pressure head lines in the negative flow are steeper than in the positive flow. Comparison with the pump map from RELAP-5 [45], shown in Volume 3, shows that the latter value is appropriate to represent the pump map from RELAP-5.

The influence of the degradation factor for reverse speed, C_{RS}, on the pump map is illustrated in Figure 4-30 through Figure 4-33. The figures show pump maps on the ω-V diagram for several different pressure heads, ranging from –1.0 bar to +1.0 bar. The pump nominal values assumed to plot the maps shown in these figures are:

ΔP_N = 1.0×10^5 Pa,
V_N = 1.0 m^3/s,
ω_N = 50.0 s^{-1}.

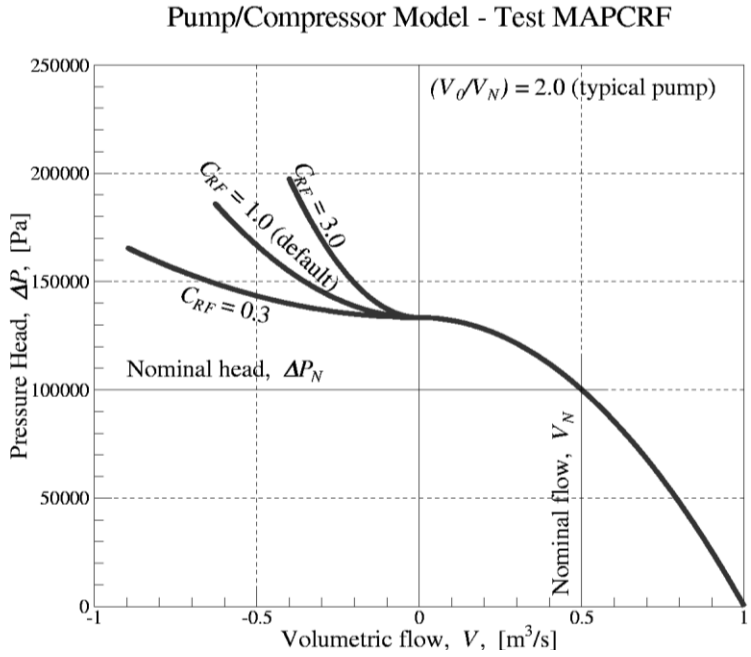


Figure 4-29 Influence of the parameter C_{RF}, on the pump maps

Figure 4-30 shows the pump map for C_{RS} = 0.1. This is the map of a typical centrifugal pump (such as built-in RELAP code - compare [45], figure 8-1, constant head lines). It is seen that when the pump speed is reversed the lines bend to the right, which means the flow tends to be in the positive direction, although it is smaller than in case of the positive pump speed.

To illustrate the influence of C_{RS}, the pump map for C_{RS} = 1.0 is shown in Figure 4-31. The lines are symmetrical with respect to the horizontal axis, which means the pump behaves exactly the same when it turns forward or reverse. Such pumps do not exist in practice. Theoretically one could build such a pump if the blades were made flat, and several outlet collectors were placed symmetrically around the pump. The pump would have to look something like the one shown in Figure 4-31. Such a pump would of course be very inefficient, and the example is shown here only to illustrate the influence of C_{RS} on the shape of the pump map.

The above two examples show that for the centrifugal machines (pumps, centrifugal compressors or centrifugal fans) an appropriate value of C_{RS} is positive and less than 1.0:

$$centrifugal\ \ pumps: \quad 0.0 < C_{RS} < 1.0$$

It is observed that with decreasing C_{RS} the lines of the pump map become more vertical in the negative pump speed range.

Figure 4-32 shows the pump map for C_{RS} = –0.1. This is a map of an axial pump/compressor. When the pump speed is reversed, the flow also reverses.

Figure 4-33 shows the pump map for C_{RS} = –1.0. For this value the lines are symmetrical with respect to the coordinate centre. That means when the pump speed is reversed the behaviour is the same as if it was placed in the opposite direction and running forward. An example of such pump is a simple, low efficiency axial fan, with flat blades.

The above two examples show that for the axial machines (typical compressors, fans) an appropriate value of C_{RS} is negative, and not smaller than –1.0:

$$axial\ \ pumps: \quad -1.0 < C_{RS} < 0.0$$

It is observed that with decreasing absolute value of C_{RS} the lines of the pump map become more vertical in the negative pump speed range.

The maps discussed above were produced using the value of exponents: *a*=2.0, *b*=0.0 (default settings for the Type 1 pump). Comparisons with typical pump and compressors maps show that a better value for the exponent *b* is 0.5 - 1.0. Figure 4-34 and Figure 4-35 show two maps for *a*=2.0, *b*=0.5. The first map (Figure 4-34) is a "centrifugal pump" map, with C_{RS} = +0.1. The second map (Figure 4-35) is an "axial pump" map, with C_{RS} = –0.1. The value of *b*=1.0 is a default setting for the Type 2 pump (the pump type recommended for general use). When *b* > 0.0 the pressure head is equal to zero when the pump/compressor speed is zero. This is more realistic than having a non-zero head with zero speed, which is the case when *b* = 0.0.

The influence of the exponents *a, b, c,* on the pump maps is shown in Figure 4-36 through Figure 4-41. The Type 1 pump settings (*a*=2.0, *b*=0.0) are shown in Figure 4-36. With these values of exponents the pressure head lines run approximately parallel, and never cross. See also the turbine map with the same *a, b*, section 4.6.3. Note that compared to the pump/compressor maps, the turbine maps are "upside-down" because of the different definition of the turbine pressure ratio (inverse).

Figure 4-37 shows the pressure head lines for (*a*=2.0, *b*=1.0). The lines cross each other in negative head range, which is not visible here, but will be clearly visible in the turbine maps - see section 4.6.3, Figure 4-54. Note that compared to the pump/compressor maps, the turbine maps are "upside-down".

Figure 4-38 shows the pressure head lines for (*a*=2.0, *b*=2.0). The lines cross each other in the same point, at the value of pressure head equal to zero. This is not visible here, but will be clearly visible in the turbine maps (see section 4.6.3, Figure 4-55 - note that compared to the pump/compressor maps, the turbine maps are "upside-down").

Figure 4-39 shows the pressure head lines for (*a*=3.0, *b*=1.0). Comparison of Figure 4-37 and Figure 4-39 shows that increasing the value of *a* one obtains a pump that is more sensitive to the speed changes, i.e. a certain (say 10%) change of speed running at nominal conditions results in larger change of the pressure head.

The influence of the exponent *c* on the pump/compressor map is shown in Figure 4-40 and Figure 4-41. These figures show the maps obtained with the exponent *c* equal to 3.0 (default value is 2.0), and all other parameters being the same as shown in Figure 4-36 and Figure 4-37. Thus the influence of the exponent *c* is seen by comparing Figure 4-40 with Figure 4-36, and Figure 4-41 with Figure 4-37.

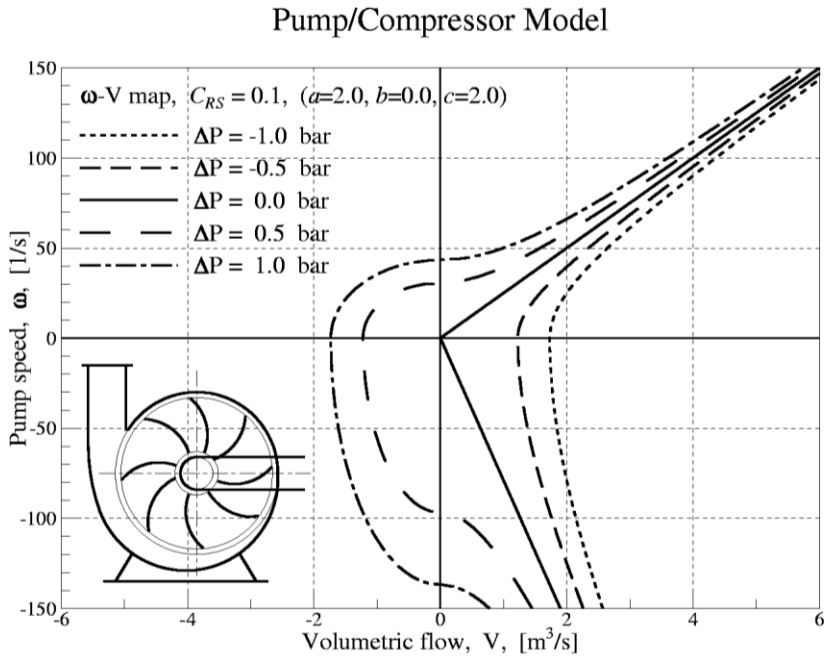


Figure 4-30 Influence of C_{RS} on the map, C_{RS} =0.1 (typical centrifugal pump).

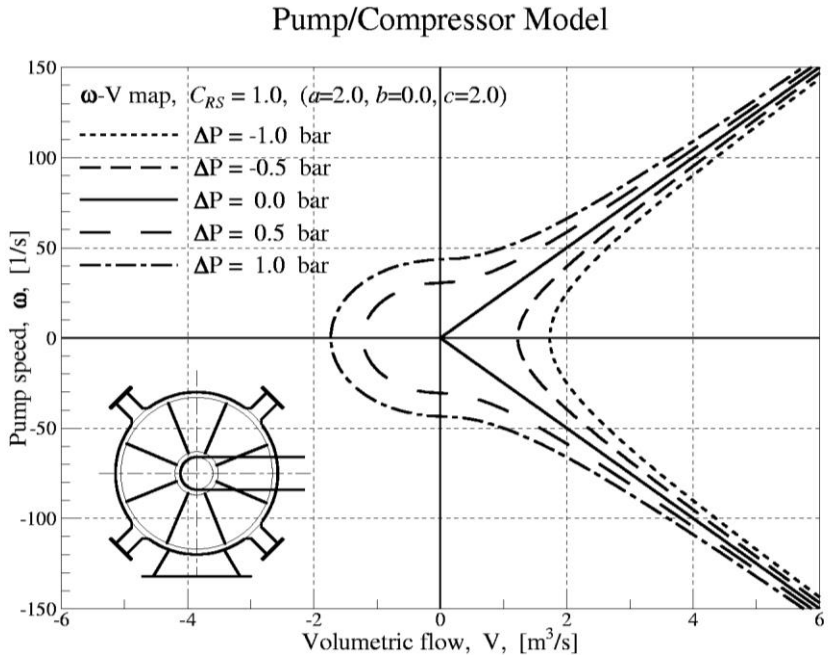


Figure 4-31 Influence of C_{RS}, on the maps.C_{RS} =1.0 (theoretical centrifugal “pump”).

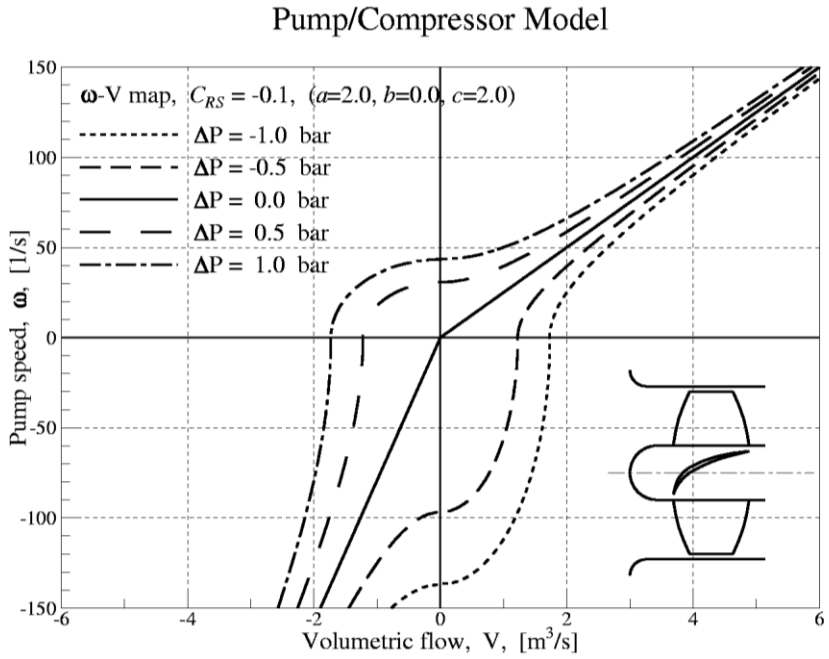


Figure 4-32 Influence of C_{RS}, on the pump maps.C_{RS} =0.1 (typical axial pump)

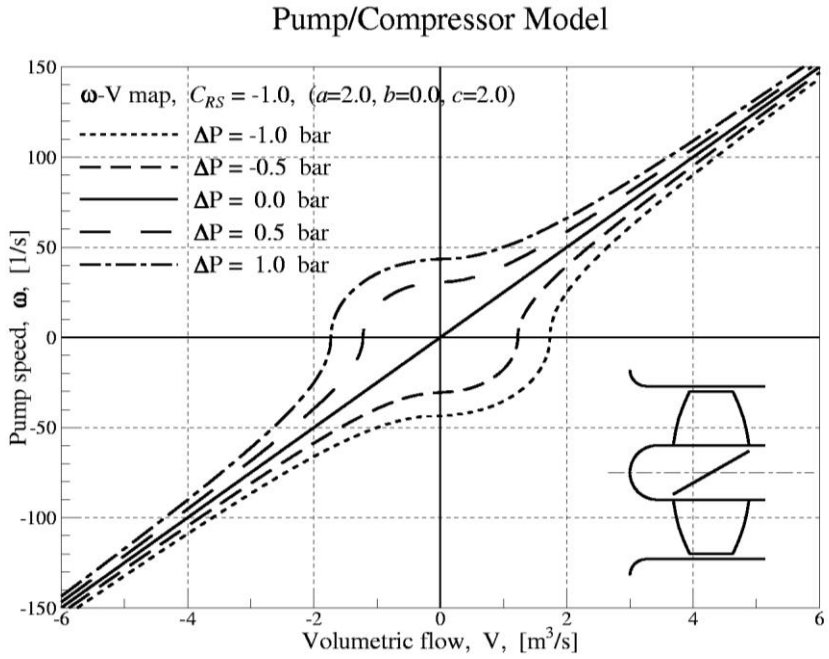


Figure 4-33 Influence of C_{RS}, on the pump maps, C_{RS} =–1.0 (theoretical axial “pump”).

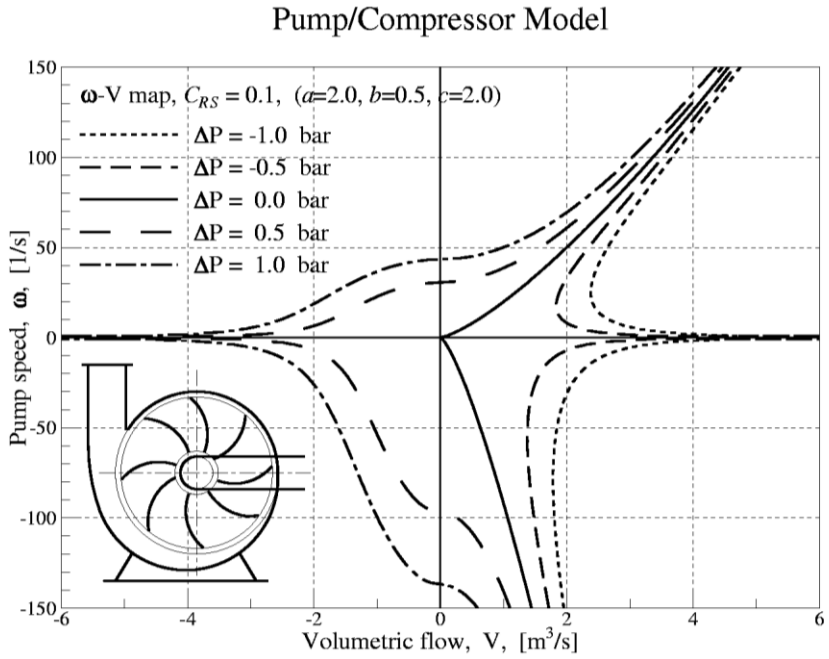


Figure 4-34 Influence of C_{RS}, on the pump maps, C_{RS} =0.1 (a=2.0, b=0.5, c=2.0)

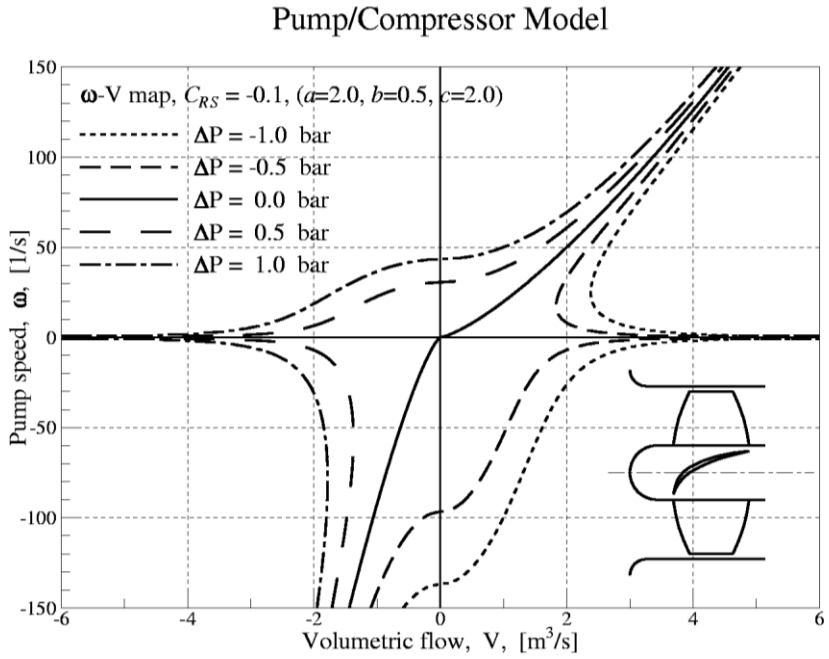


Figure 4-35 Influence of C_{RS}, on the pump maps, C_{RS} =–0.1 (a=2.0, b=0.5, c=2.0)

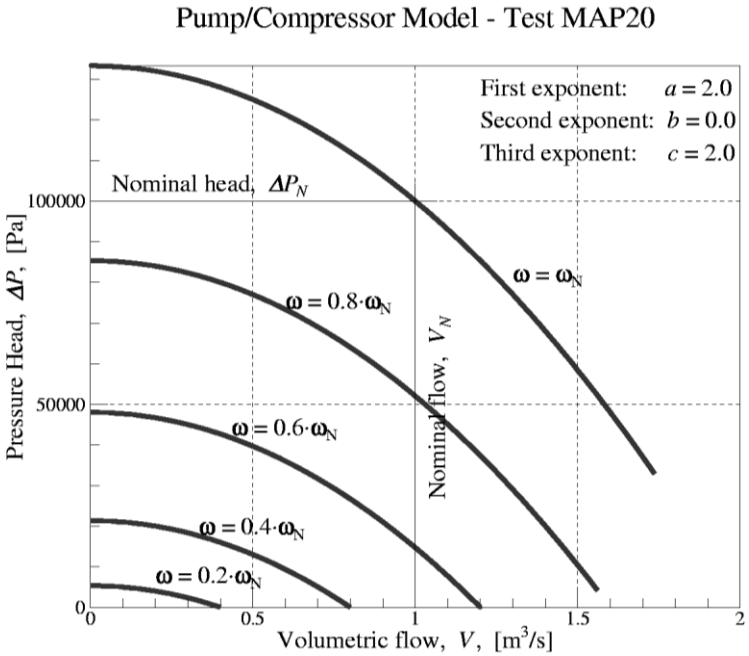


Figure 4-36 Influence of exponents on map, *a*=2.0, *b*=0.0, *c*=2.0 (Type 1 pump).

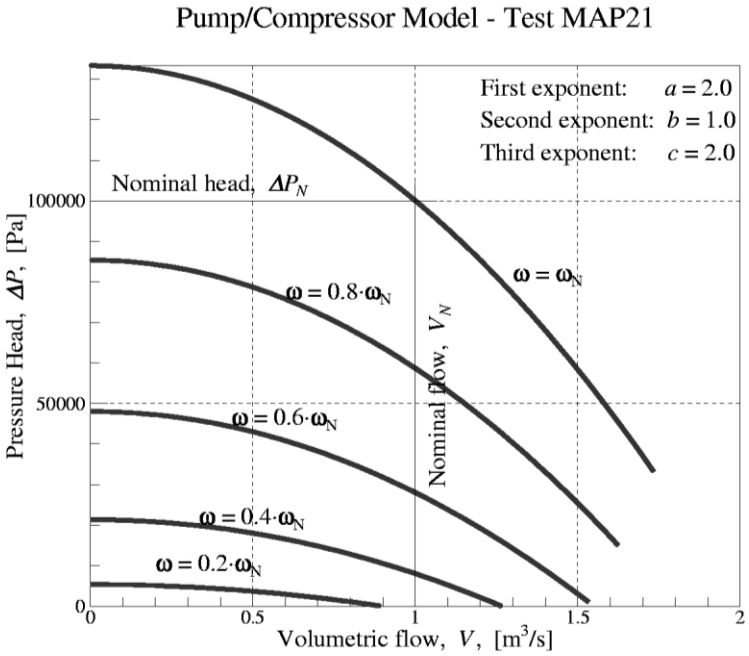


Figure 4-37 Influence of exponents on map, *a*=2.0, *b*=1.0, *c*=2.0 (default for Type 2 pump).

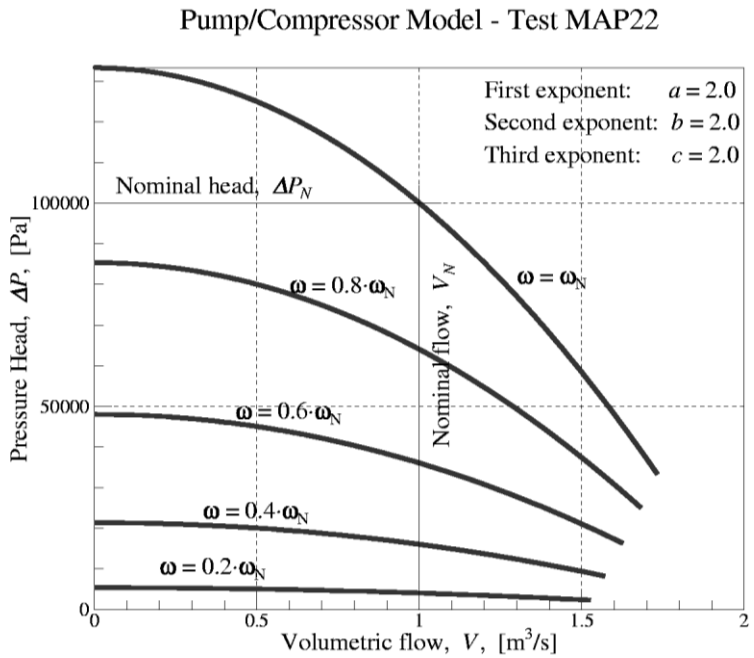


Figure 4-38 Influence of exponents on map, *a*=2.0, *b*=2.0, *c*=2.0.

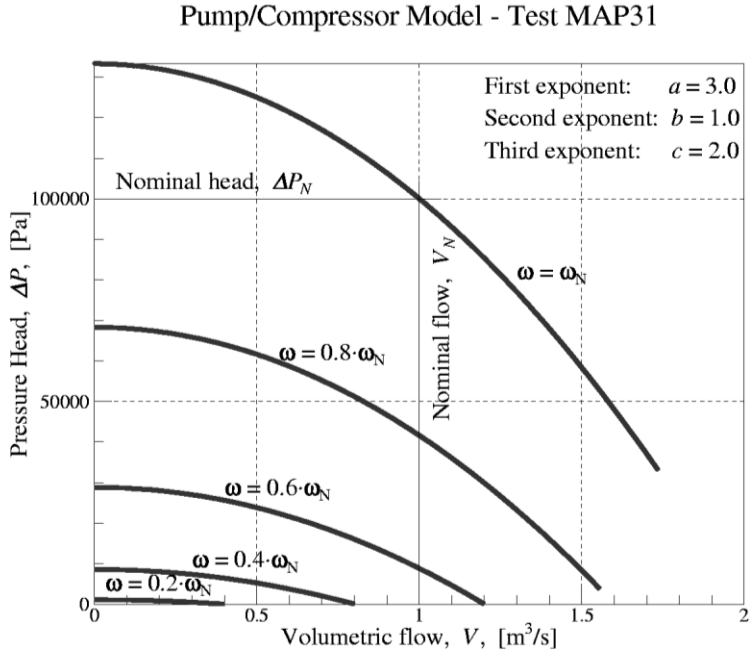


Figure 4-39 Influence of exponents on map, *a*=3.0, *b*=1.0, *c*=2.0.

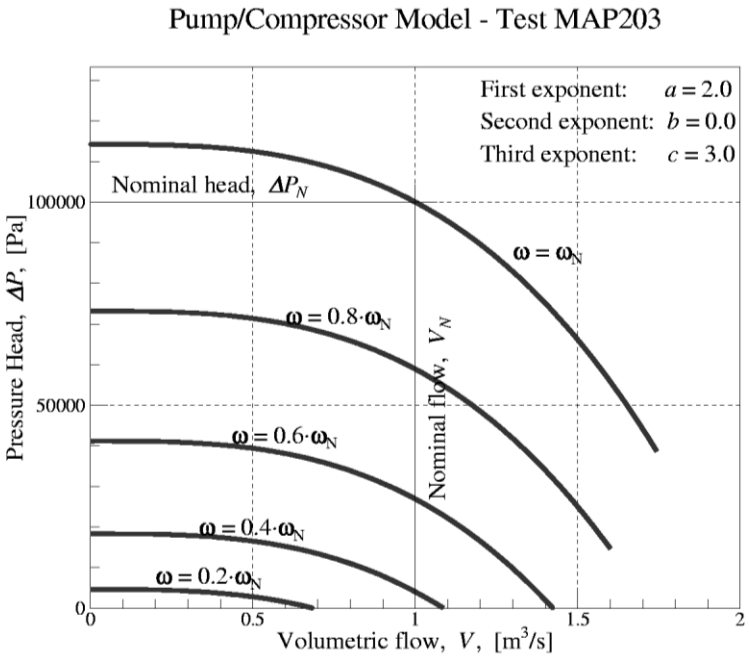


Figure 4-40 Influence of exponents on map, *a*=2.0, *b*=0.0, *c*=3.0 (compare Figure 4-36).

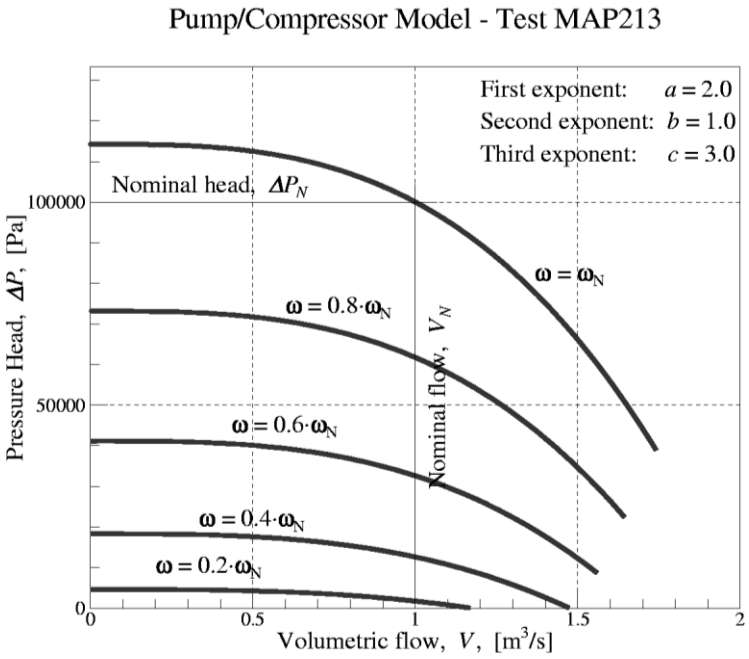


Figure 4-41 Influence of exponents on map, *a*=2.0, *b*=1.0, *c*=3.0 (compare Figure 4-37).

Negative values of the exponent *b*, although generally not recommended, was found useful for matching some compressor map data. If $b < 0.0$, then the term with ω_R^b is becoming large when the speed is small, giving unrealistically large pump head or compressor pressure ratio for a slowly turning machine. To prevent this a limit is imposed on ω_R^b: $\omega_R^b < \omega_{lim}^b$. For lower speeds the term is linearly interpolated to give zero for zero speed (see Figure 4-42). The default value of ω_{lim}^b is equal to 10.0. The data in Figure 4-42 is shown for the limit equal to 2.0. The pump/compressor map for the parameter *b* equal to –1.0 is shown in Figure 4-43. Comparison with Figure 4-41 shows how the map changes when the value of *b* is changed from +1.0 to –1.0.

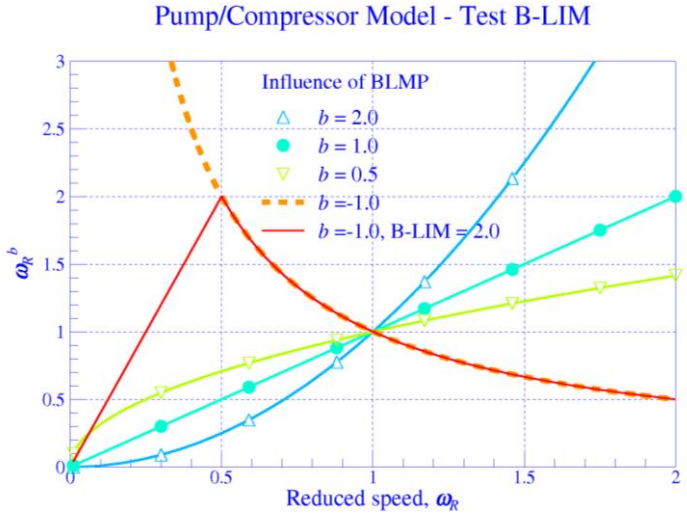


Figure 4-42 Influence of the limit B-LIM in case of negative *b*.

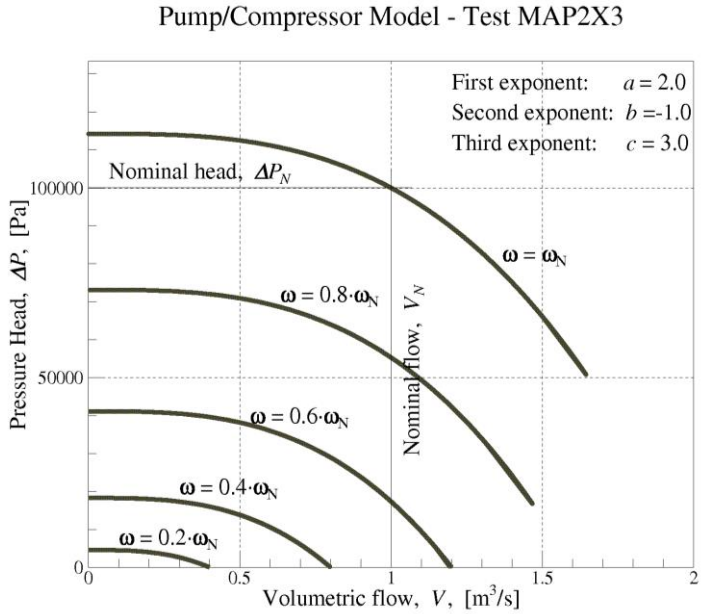


Figure 4-43 Influence of exponents on map, *a*=2.0, *b*=–1.0, *c*=3.0 (compare Figure 4-41).

4.6.2.4 Pump/Compressor Maps - Degradation of Head (Surge Model)

In the low flow range the machines such as compressors typically lose their capability to provide head. This is called "surge". In case of pumps the maps are close to the ideal maps, shown above. However, even in case of pumps a small decrease of head in low flow range is typically observed - see Volume 3 - comparison with pump model from RELAP-5.

In SPECTRA three input parameters are introduced in order to model the pump/compressor behavior in the low flow range. These are:

C_{VS} relative volumetric flow at which degradation begins, defined as a ratio of the flow at which surge starts to the flow at zero pressure head at nominal speed (see Figure 4-27).

C_{DS} degradation factor, defined as the ratio of the degraded pressure head at zero flow to the un-degraded pressure head at zero flow (resulting from the un-degraded pump/compressor model equation - see Figure 4-27).

s exponent in surge region interpolation.

The low flow region (the surge region) is defined as a region where the volumetric flow is:

$$V < V_{surge} = C_{VS} \cdot V_0(\omega) = C_{VS} \cdot V_N \cdot \left(\frac{C_P}{C_P - 1}\right)^{1/c} \cdot \left(\frac{\omega_R^a}{\omega_R^b}\right)^{1/c}$$

In the low flow range the pump head is decreased by introducing an effective speed, ω', defined as:

$$\omega' = \omega \cdot [C_{DS}^{1/a} + (1 - C_{DS}^{1/a}) \cdot (V / V_{surge})^{1/s}]$$

Substituting the above formula into the pump/compressor model equation, shown in section 4.6.2.3, one obtains the formula applied in the low flow (surge) region:

$$\Delta P_R = C_P \cdot \omega_R^a \cdot [C_{DS}^{1/a} + (1 - C_{DS}^{1/a}) \cdot (V / V_{surge})^{1/s}]^a \mp$$
$$\mp (C_P - 1) \cdot \omega_R^b \cdot [C_{DS}^{1/a} + (1 - C_{DS}^{1/a}) \cdot (V / V_{surge})^{1/s}]^b \cdot V_R^2$$

In the above equation the sign is – in case of positive flow, and + in case of negative flow. The influence of the surge parameters C_{VS} and C_{DS}. is discussed first for a typical compressor. Next, examples of a typical pump are shown and the influence of the exponent s on the stability of the pump is discussed.

For the next case a very small degradation factor was assumed:

relative flow:	C_{VS}	= 0.7,
degradation factor	C_{DS}	= 0.01.

Such a small value of C_{DS} is appropriate for gas compressors. The value of the constant C_P has been chosen as 3.0 (typical compressor). With this value of C_P the lines are relatively steep (C_P ~ 1.3 for typical pump maps - see discussion below), and the relative flow for surge, C_{VS}, is large (0.7) compared to a typical pump (~0.3 - see discussion below). The resulting pressure head map is shown in Figure 4-44.

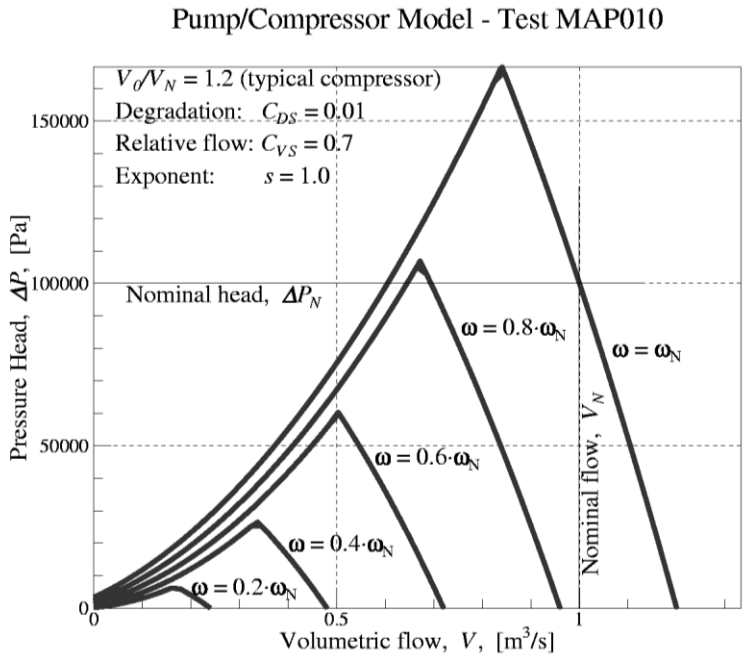


Figure 4-44 Influence of surge parameters on map, C_{DS}=0.01, C_{VS}=0.7, (V_0/V_N=1.2).

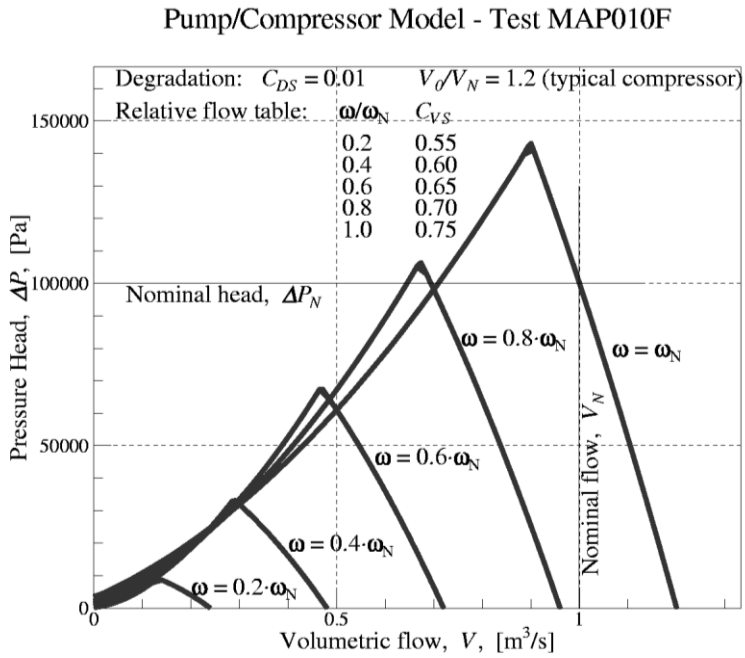


Figure 4-45 Influence of surge parameters on map, C_{DS}=0.01, C_{VS}=f(ω), (V_0/V_N=1.2).

The equation determining the pump/compressor map implies that at the nominal speed (ω_R=1.0), V_N, is related to V_0 by (see Figure 4-27):

$$\frac{V_N}{V_0} = \left(\frac{C_P - 1}{C_P} \right)^{1/c}$$

The relative flow for surge is typically smaller than the nominal flow, thus:

$$C_{VS} = \frac{V_{surge}}{V_0} < \frac{V_N}{V_0} = \left(\frac{C_P - 1}{C_P} \right)^{1/c}$$

Therefore: for C_P = 1.3 C_{VS} < 0.480
for C_P = 3.0 C_{VS} < 0.816

If C_{VS} is greater than the limit defined by the above formula, the code issues a warning message.

The next example is presented to illustrate the ability of the user to specify a variable surge limit: $C_{VS} = f(\omega)$. All the parameters are the same as above, except for the C_{VS}. In the present case the value is tabulated, as shown in the table below.

Table 4-5 Example of volumetric flow limit for surge as a function of speed.

ω/ω_N	0.20	0.40	0.60	0.80	1.00
C_{VS}	0.55	0.60	0.65	0.70	0.75

The results are shown in Figure 4-45. The pressure head lines are very similar to those observed in Figure 4-48. The only difference is the location of the surge limit. In Figure 4-45 the surge limit varies from 0.55 to 0.75, depending on the speed, while in Figure 4-44 it is constant and equal to 0.7.

The next examples show maps with low values of the degradation factor. Such maps are appropriate for typical water pumps. Before the pump examples are shown, a short discussion is presented on the stability of pump characteristics. The pump map has locally unstable characteristics if there is a region where the pressure head decreases with decreasing fluid flow. In this region stable operation of the pump is impossible; the fluid flow will oscillate between the two points bounding the unstable part of the map.

In order to obtain conditions for stable pump characteristics, it is important to find out what is the pressure head at zero flow, compared to the pressure head at the surge point. The pump/compressor pressure head at zero flow is equal to:

$$\Delta P_R\big|_{V=0.0} = C_P \cdot \omega_R^a \cdot C_{DS}$$

The surge point flow is equal to:

$$\frac{V_{surge}}{V_0} = C_{VS}$$

Using the relation between V_N and V_0:

$$\frac{V_0}{V_N} = \left(\frac{C_P}{C_P - 1}\right)^{1/c} \cdot \left(\frac{\omega_R^a}{\omega_R^b}\right)^{1/c}$$

the relative flow at surge can be expressed as:

$$\left(\frac{V_{surge}}{V_N}\right) = \left(\frac{V_{surge}}{V_0}\right) \cdot \left(\frac{V_0}{V_N}\right) = C_{VS} \cdot \left(\frac{\omega_R^a}{\omega_R^b}\right)^{1/c} \cdot \left(\frac{C_P}{C_P - 1}\right)^{1/c}$$

The pressure head at the surge point is therefore equal to:

$$\Delta P_R\big|_{V=surge} = C_P \cdot \omega_R^a - (C_P - 1) \cdot \omega_R^b \cdot \left(\frac{\omega^a}{\omega^b}\right) \cdot \left(\frac{C_P}{C_P - 1}\right) \cdot C_{VS}^c$$

or:

$$\Delta P_R\big|_{V=surge} = C_P \cdot \omega_R^a \cdot (1 - C_{VS}^c)$$

The pressure head at zero flow is equal to the pressure head at the surge point if:

$$C_P \cdot \omega_R^a \cdot C_{DS} = C_P \cdot \omega_R^a \cdot (1 - C_{VS}^c)$$

or:

$$C_{DS} = 1 - C_{VS}^c$$

If the head at zero flow is smaller than the pressure head at the surge point, then there will certainly be an unstable region in the pump map. The necessary condition to obtain a stable pump is therefore: $C_{DS} \geq 1 - C_{VS}^c$. As will be shown below, this condition is not sufficient to obtain a stable pump. A proper value of the exponent *s* is needed to ensure a fully stable pump map.

The example cases presented below were calculated using the default value of V_0/V_N = 2.0 (typical centrifugal pump). In the first case the value of C_{VS} was set to 0.3 (This value is still somewhat too high compared to a typical pump - see Volume 3 - but it gives a more clearly visible unstable region.)

relative flow: C_{VS} = 0.3,

To obtain a stable map, the value of C_{DS} must be greater than or equal to $1 - 0.3^2 = 0.91$. The first example was run with this value:

degradation factor C_{DS} = 0.91.

The resulting pressure head map is shown in Figure 4-46. It is seen that although the pressure head at the surge point and at the zero flow are the same, there is an unstable region in the pump map. The unstable region is marked in the figure.

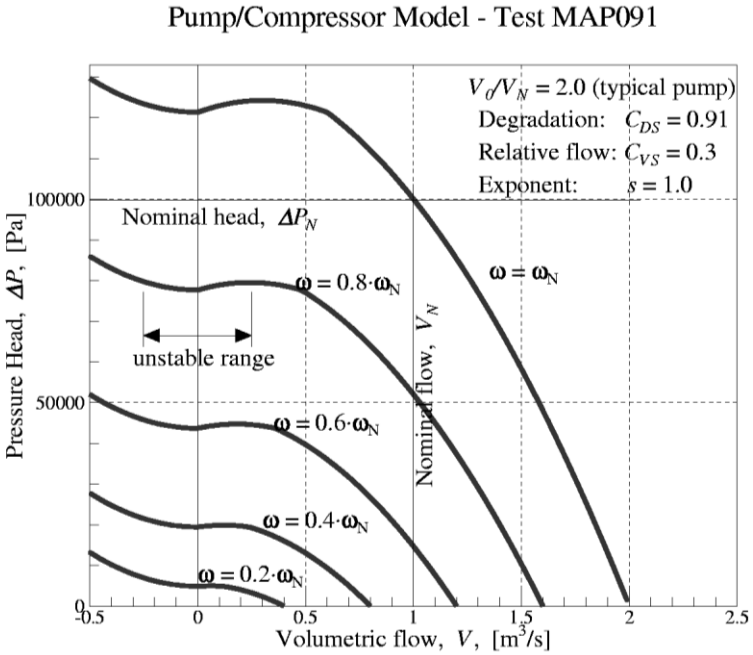


Figure 4-46 Influence of surge parameters, C_{DS}=0.91, C_{VS}=0.3, s=1.0, (V_0/V_N=2.0).

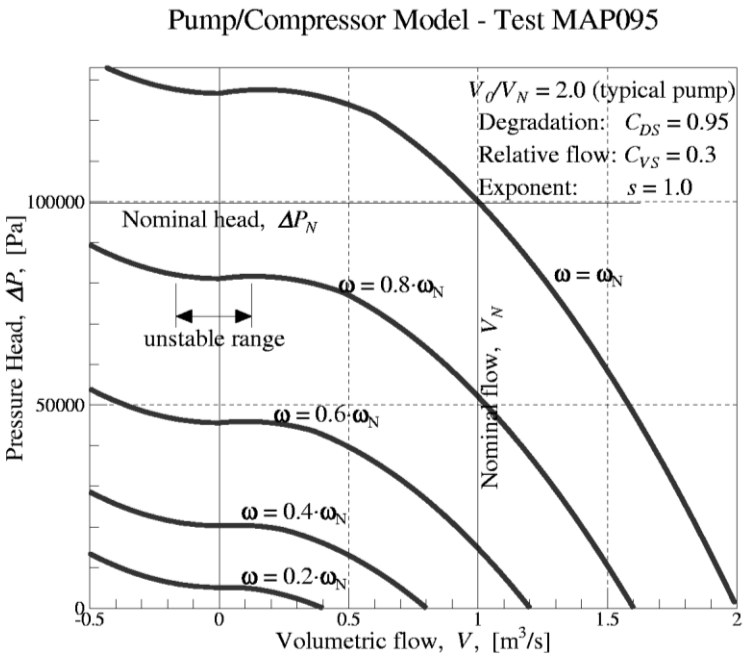


Figure 4-47 Influence of surge parameters, C_{DS}=0.95, C_{VS}=0.3, s=1.0, (V_0/V_N=2.0).

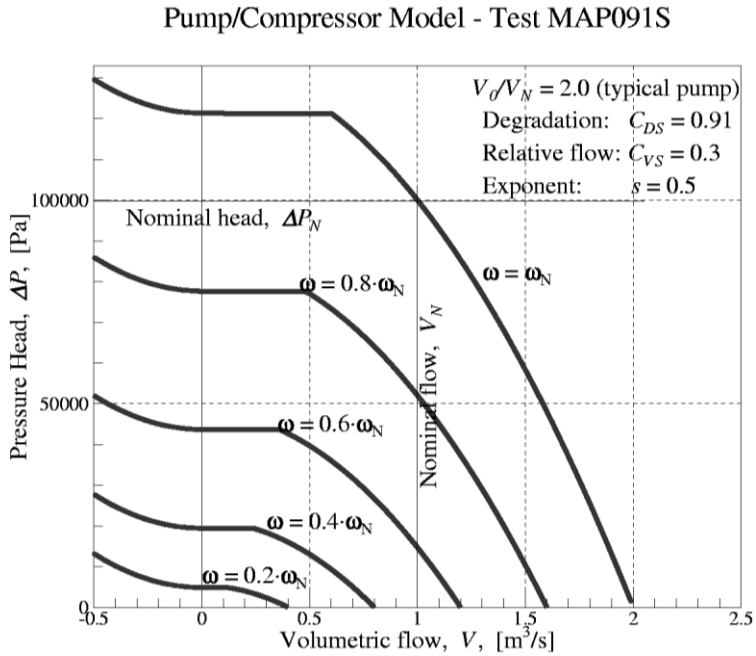


Figure 4-48 Influence of surge parameters on map, C_{DS}=0.91, C_{VS}=0.3, s=0.5, (V_0/V_N=2.0).

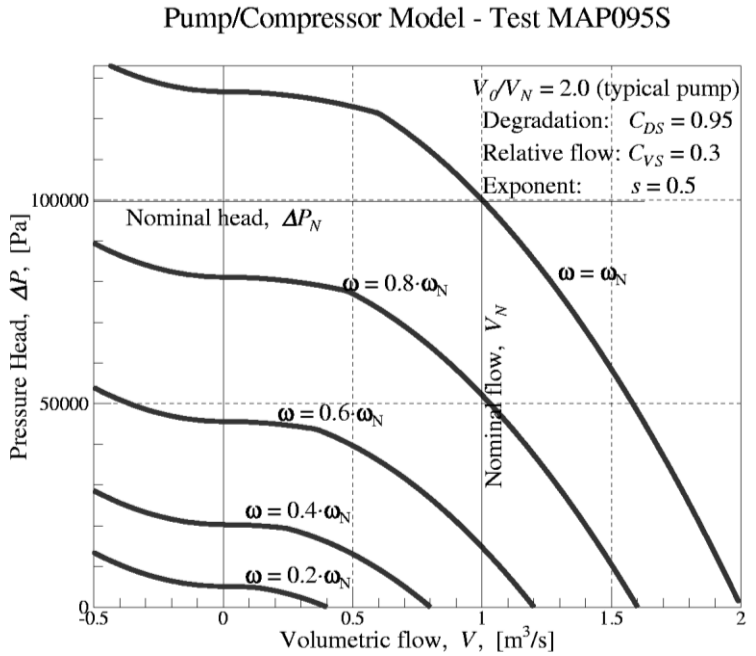


Figure 4-49 Influence of surge parameters on map, C_{DS}=0.95, C_{VS}=0.3, s=0.5, (V_0/V_N=2.0).

In the next example the degradation factor was set to a higher value:

degradation factor C_{DS} = 0.95.

The resulting pressure head map is shown in Figure 4-47. It is seen that the unstable region still exists, although it is smaller than in the previous case.

The next two cases were run with the same degradation factors, namely 0.91 and 0.95 respectively, but with the exponent *s* decreased from 1.0 (the default value) to 0.5. The results are shown in Figure 4-48 and Figure 4-49. It is seen that there is no unstable region in both maps.

It is interesting to note that the map shown in Figure 4-48 is similar to the pump map applied in the MELCOR code (see [46], Reference Manual, CVH/FL Package). MELCOR however does not take into account a change of the pressure head with the pump speed; the pump is simply on (nominal speed) or off (zero pressure head). Moreover in MELCOR the pump head does not change in negative speed range, so the MELCOR model and the model shown in Figure 4-48 are identical provided that a check valve is present, prohibiting backflow through the pump.

The above examples show that in order to obtain a stable map the exponent *s* should be set to 0.5. The default value of *s* (1.0) was selected because this is most appropriate for gas compressors, and the surge model is of primary importance for gas compressors. If *s* = 0.5 is used for the example shown in Figure 4-44, then the pressure head will become negative in the low flow range - see the example case shown in Figure 4-50.

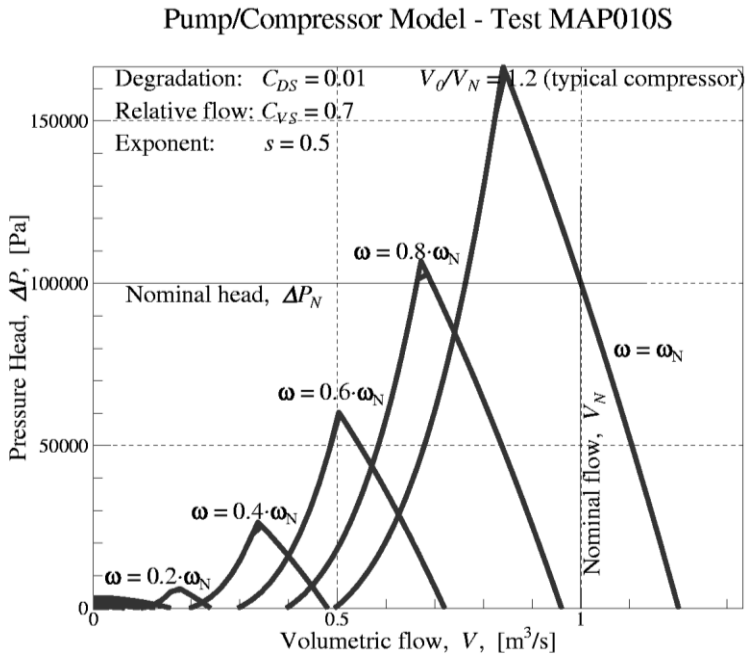


Figure 4-50 Influence of surge parameters on map, C_{DS}=0.01, C_{VS}=0.7, s=0.5, (V_0/V_N=2.0).

The use of the exponent *s* in order to provide a stable map may be important in analyses involving typical pumps. For example, the pump model applied in RELAP-5 [45] is best approximated using a stable map, with *s* = 0.5 - see Volume 3.

4.6.2.5 Power

For gas compressors the power which enters the fluid (atmosphere of CV downstream the compressor) is computed from:

$$Q_{hydr,atms} = \frac{1}{\eta} V \rho c_P T \left(\Pi^{(\kappa-1)/\kappa} - 1 \right)$$

Here η is the compressor efficiency, V is the volumetric flow (m^3/s), ρ is the density (kg/m^3), c_P is the specific heat (J/kg/K), T is the inlet temperature (K), Π is the pressure ratio, and Q_{hydr} is the total power source for the fluid (W) (the value is positive during normal pump/compressor operation).

For water pumps the power which enters the fluid (pool of CV downstream the pump) is computed from:

$$Q_{hydr,pool} = \frac{1}{\eta} V \Delta P$$

Here η is the pump efficiency, V is the volumetric flow (m^3/s), ΔP is the pressure head (Pa), and Q_{hydr} is the total power source for the fluid (W) (the value is positive during normal pump/compressor operation).

4.6.3 Turbine Model

Two models are available for turbines in SPECTRA. The first (simplified) model simply puts a flow resistance factor, which provides the desired pressure difference across the turbine. The second model, referred to as the advanced turbine model or shortly the turbine model, is based on user-defined maps, and is very similar to the pump model type 2, described in section 4.6.2. The simplified turbine model is seldom used. This was the first turbine model applied in an early version of SPECTRA, and it is preserved in the code to keep compatibility with earlier input decks. The advanced turbine model is recommended for general application.

4.6.3.1 Approach

The turbine model is based on the same approach as the pump Type 2 model (see section 4.6.2). In fact the turbine model uses exactly the same subroutine as the Type 2 pump/compressor. Only the input procedures are slightly different. All input parameters are the same as in case of the Type 2 pump. The turbine nominal parameters, such as nominal flow and pressure ratio or head, are internally converted by the code into nominal parameters of an "equivalent pump", as shown in Figure 4-51. The equivalent pump is defined as a pump/compressor that has exactly the same map as the turbine, if the map is plotted in terms of the pump pressure ratio (outlet divided by inlet pressure) rather than the turbine pressure ratio (inlet divided by outlet pressure).

The "equivalent pump" approach allows to perform calculations using the same subroutines that are calculating pumps/compressors. The equation defining turbine behavior is therefore exactly the same as the equation defining the pump/compressor map (see section 4.6.2), and is therefore not discussed here. Only the conversion to the "Equivalent Pump" is shown below.

4.6.3.2 Turbine Maps

Using the equation defining the pump map (section 4.6.2), it may be shown that nominal parameters of the equivalent pump are related to the turbine parameters as follows:

- Nominal volumetric flow:

$$V_{E.P.} = V_0 \cdot \left[(C_P - 1) / C_P \right]^{(1/c)}$$

- Nominal pressure ratio (if the pressure ratio is entered):

$$\Pi_{E.P.} = 1 + \frac{1 - 1/\Pi_N}{(C_P - 1)\,(V_N / V_{E.P.})^c - C_P}$$

- Nominal pressure head (if the pressure head is entered):

$$\Delta P_{E.P.} = \frac{\Delta P_N}{(C_P - 1)\,(V_N / V_{E.P.})^c - C_P}$$

Π_N nominal turbine pressure ratio, (-), (user defined)
ΔP_N nominal turbine pressure head, (Pa), (alternatively user defined)
V_N nominal turbine flow, (m^3/s), (user defined)
V_0 flow at zero-head $\Delta P = 0.0$ (or Π=1.0), (m^3/s), (user defined)
$\Pi_{E.P.}$ nominal pressure ratio of the "equivalent pump", (-)
$\Delta P_{E.P.}$ nominal pressure head of the "equivalent pump", (Pa)
C_P constant in pump map (see 4.6.2.3). Note that the value of this constant affects the nominal parameters of the "equivalent pump", but does not influence the turbine map, which is fully defined by V_N , V_0 , and Π_N or ΔP_N
c third exponent in turbine map definition, (-) (user defined)

The conversion to the "equivalent pump" is done automatically within the code; it is "transparent" for the user, i.e. the user works only with the turbine parameters (V_N , V_0 , Π_N or ΔP_N) while internally calculations are performed using the equivalent pump parameters ($V_{E.P.}$, $\Pi_{E.P.}$ or $\Delta P_{E.P.}$).

The parameters available to tune the turbine map are: V_0, the exponents a, b, c, and the constant C_{RS}, determining the turbine behavior in case of negative speed. The influence of these parameters on the turbine map is discussed below.

The influence of the parameter V_0 on the turbine maps is quite obvious from the definition of the parameter. It is illustrated below, in Figure 4-52. The lines shown in this figure were obtained for a turbine with Π_N = 1.5, V_N = 1.0 m^3/s, ω_N = 50.0 s^{-1}, a = 2.0, b = 1.0, c = 1.0, running at nominal speed.

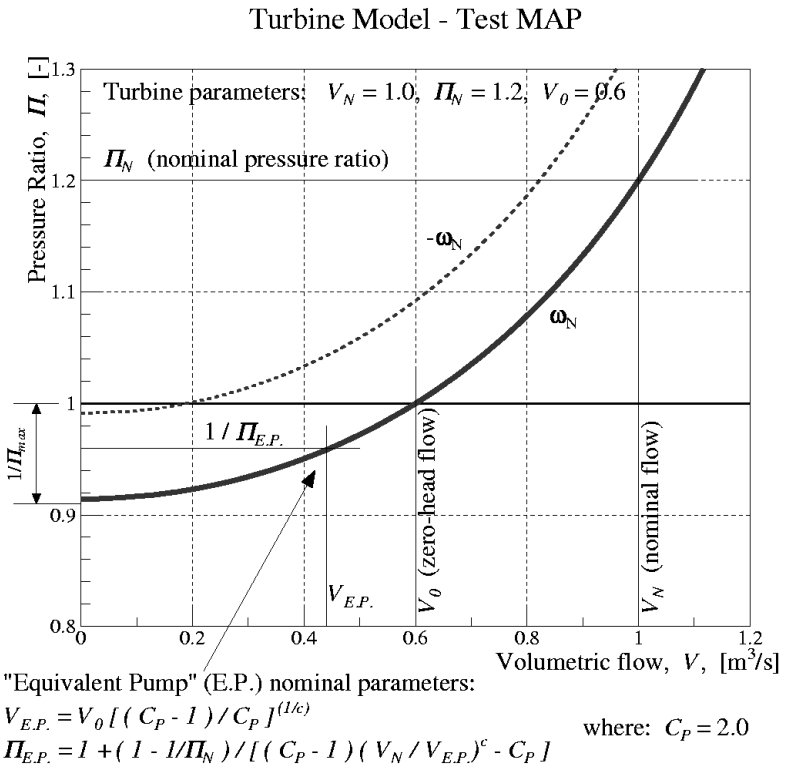


Figure 4-51 Turbine map - conversion of the turbine parameters into the "Equivalent Pump" parameters, performed internally by the code.

The parameters available to tune the turbine map are: V_0, the exponents a, b, c, and the constant C_{RS}, determining the turbine behavior in case of negative speed. The influence of these parameters on the turbine map is discussed below.

The influence of the parameter V_0 on the turbine maps is quite obvious from the definition of the parameter. It is illustrated below, in Figure 4-52. The lines shown in this figure were obtained for a turbine with $\Pi_N = 1.5$, $V_N = 1.0$ m³/s, $\omega_N = 50.0$ s^{-1}, $a = 2.0$, $b = 1.0$, $c = 2.0$, running at nominal speed.

The influence of the exponents a, b, c, on the turbine maps is shown in Figure 4-53 through Figure 4-58. The following values were used to produce maps shown in these figures:

Π_N = 1.5,
V_N = 1.0 m³/s,
V_0 = 0.5 m³/s,
ω_N = 50.0 s^{-1}.

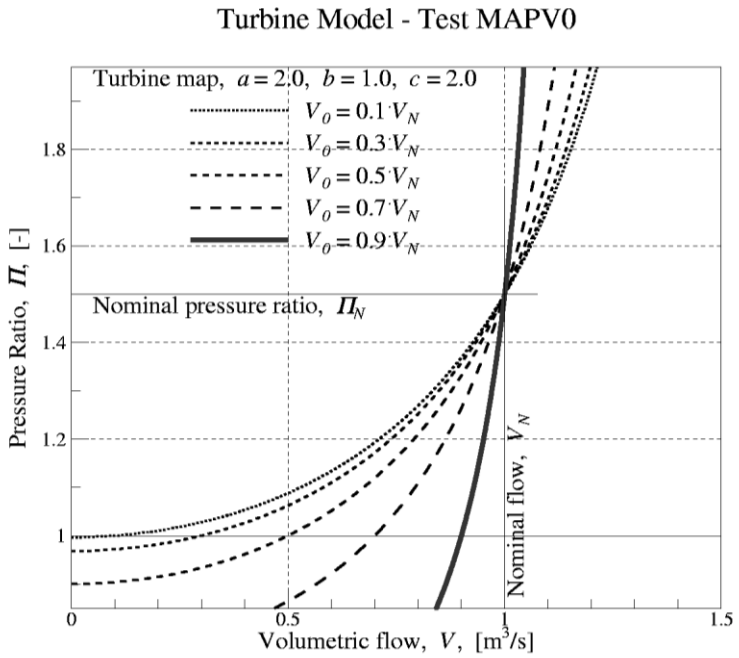


Figure 4-52 Influence of V_0 on turbine map (a=2.0, b=1.0, c=2.0)

The map obtained for (a=2.0, b=0.0, c=2.0) is shown in Figure 4-53. With these values of the exponents the pressure ratio lines run approximately parallel, and never cross. See also the pump/compressor map with the same *a, b, c,* - Figure 4-36. Note that compared to the turbine maps, the pump/compressor maps are "upside-down" because of a different definition of the pressure ratio (inverse).

Figure 4-54 shows the pressure ratio lines for (a=2.0, b=1.0, c=2.0). The lines cross each other at flow $V>V_0$. (See also the pump/compressor maps with the same *a, b, c,* - Figure 4-37. Note that compared to the turbine maps, the pump/compressor maps are "upside-down".)

Figure 4-55 shows the pressure ratio lines for (a=2.0, b=2.0, c=2.0). The lines cross each other in the same point; the point where the pressure head is equal to zero (or the pressure ratio equal to one, or the volumetric flow equal to: $V=V_0$). With this set of constants the turbine becomes more sensitive to changes of speed; the pressure ratio is small for small turbine velocity and large for large velocity. (See also the pump/compressor map with the same *a, b, c,* - see Figure 4-38. Note that compared to the turbine maps, the pump/compressor maps are "upside-down").

Figure 4-56 shows the pressure ratio lines for (a=3.0, b=1.0, c=2.0). Comparison of Figure 4-54 and Figure 4-56 shows that increasing the value of *a,* the lines cross at higher value of flow. (See also the pump/compressor map with the same *a, b, c,* - see Figure 4-39. Note that compared to the turbine maps, the pump/compressor maps are "upside-down").

Note that in the case of a turbine, a machine more sensitive to speed changes is obtained by increasing the exponent *b*, while in case of pumps/compressors the same is achieved by increasing the exponent *a*.

The influence of the third exponent, *c*, is illustrated in Figure 4-57 and Figure 4-58. The maps shown in these figures should be compared to the map shown in Figure 4-54, to see how the map changes when *c* is increased from the default value of 2.0 (Figure 4-54) to 3.0 (Figure 4-57), and decreased to 1.5 (Figure 4-58).

The influence of the reverse speed degradation factor, C_{RS} , on the turbine maps is shown in Figure 4-59, Figure 4-60 and Figure 4-61. The following values were used to produce maps shown in these figures:

Π_N = 1.5,
V_N = 1.0 m^3/s,
V_0 = 0.5 m^3/s,
ω_N = 50.0 s^{-1},
a = 2.0,
b = 1.0,
c = 2.0.

Figure 4-59 shows turbine maps for C_{RS} = 0.1 for the turbine speed, ω, between +100% and –100%. It is seen that in case of negative (reverse) speed the pressure ratio is larger than for a corresponding positive speed. The difference is small for small $|\omega|$, and increases with increasing $|\omega|$.

Figure 4-60 shows turbine maps for C_{RS} = 2.0 for the turbine speed, ω, between +100% and –100%. In contrast to the previous case, the negative (reverse) speed gives a smaller pressure ratio than that for a corresponding positive speed. Again, the difference is small for small $|\omega|$, and increases with increasing $|\omega|$.

Generally, it may be observed that for $C_{RS} < 1.0$ the lines at reverse speed are above those at the corresponding normal speed, while for $C_{RS} > 1.0$ the lines at reverse speed are below those at the corresponding normal speed.

Finally, Figure 4-61 shows turbine maps for C_{RS} = –1.0 for the turbine speed, ω, between +100% and –100%. With decreasing the value of C_{RS} , the pressure lines are higher in the negative (reverse) speed range.

The influence of the parameter C_{RS} may be shortly described by looking at the maps for the (nearly) zero fluid flow and ω = +100% and ω = –100%. If C_{RS} is positive, then at nearly zero flow the turbine acts as compressor, i.e. it provides pressure head in the normal flow direction, for both forward and reverse speeds. On the other hand, if C_{RS} is negative, then at zero flow and ω = +100% the turbine acts as compressor, proving (a small) pressure head in the normal flow direction, while at zero flow and ω = –100% it acts as a compressor providing (a small) head in the reversed flow direction. It should be noted that for large fluid flows and ω = –100% it acts as a compressor providing a large pressure head in the reversed flow direction.

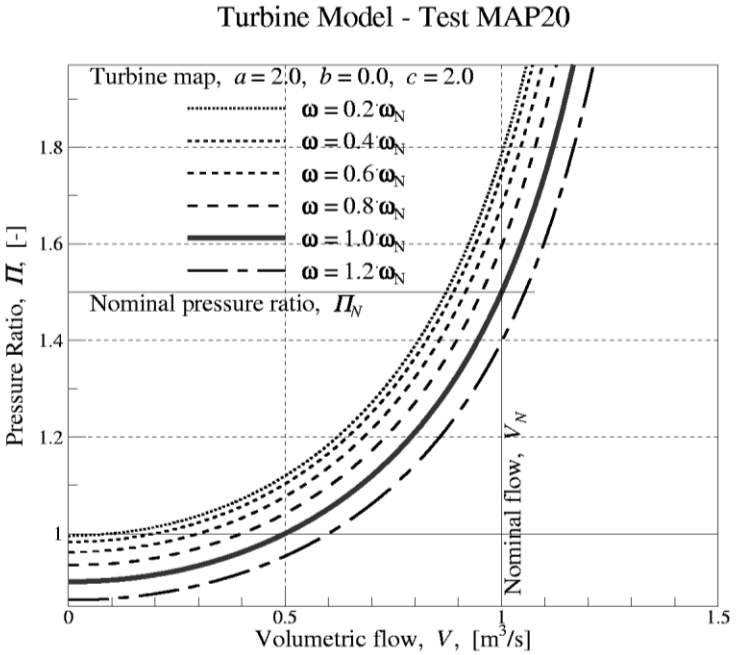


Figure 4-53 Influence of exponents on turbine map, *a*=2.0, *b*=0.0, *c*=2.0.

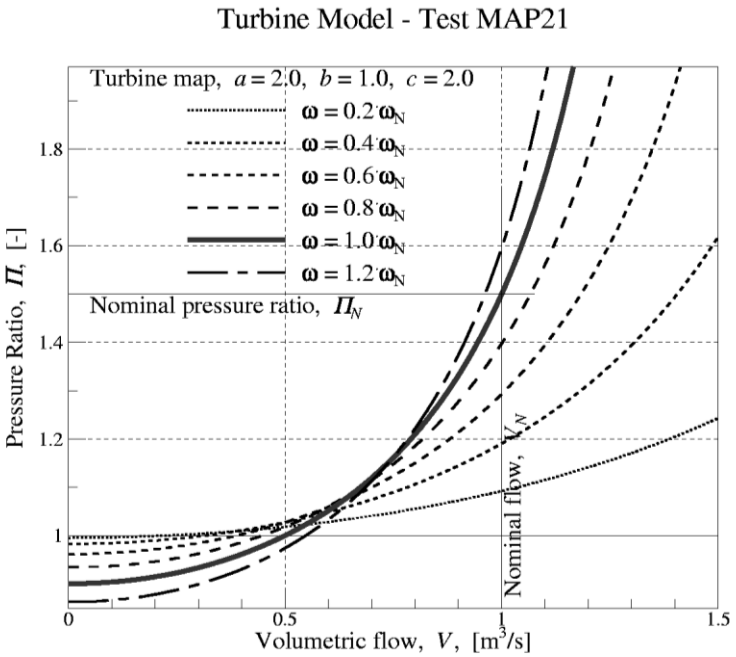


Figure 4-54 Influence of exponents on turbine map, *a*=2.0, *b*=1.0, *c*=2.0.

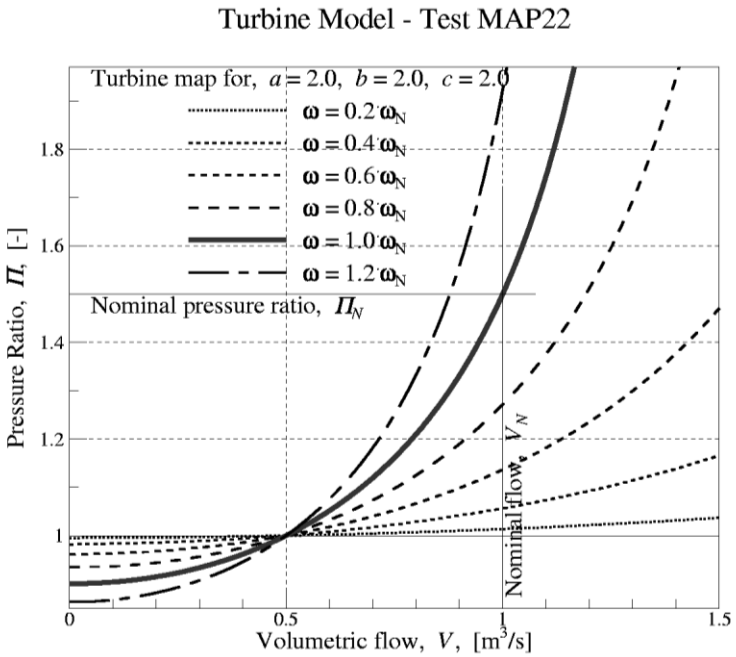


Figure 4-55 Influence of exponents on turbine map, *a*=2.0, *b*=2.0, *c*=2.0.

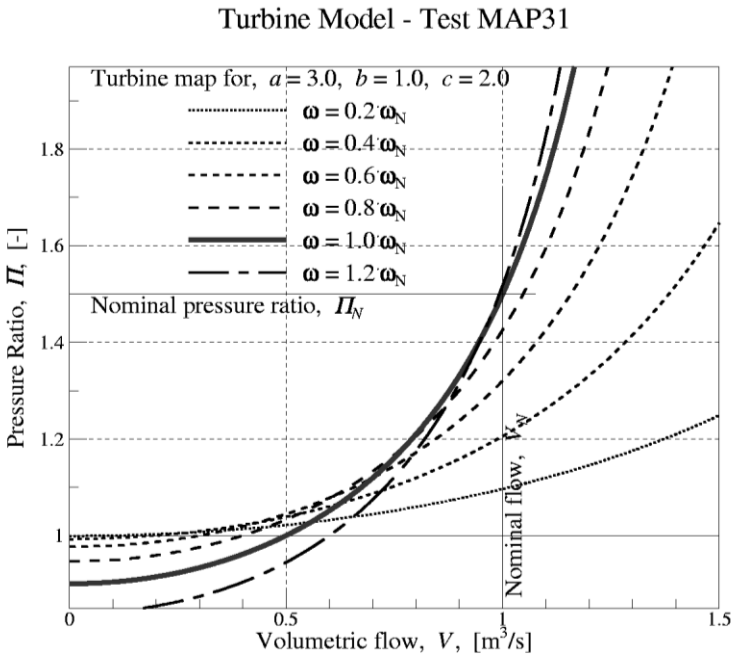


Figure 4-56 Influence of exponents on turbine map, *a*=3.0, *b*=1.0, *c*=2.0.

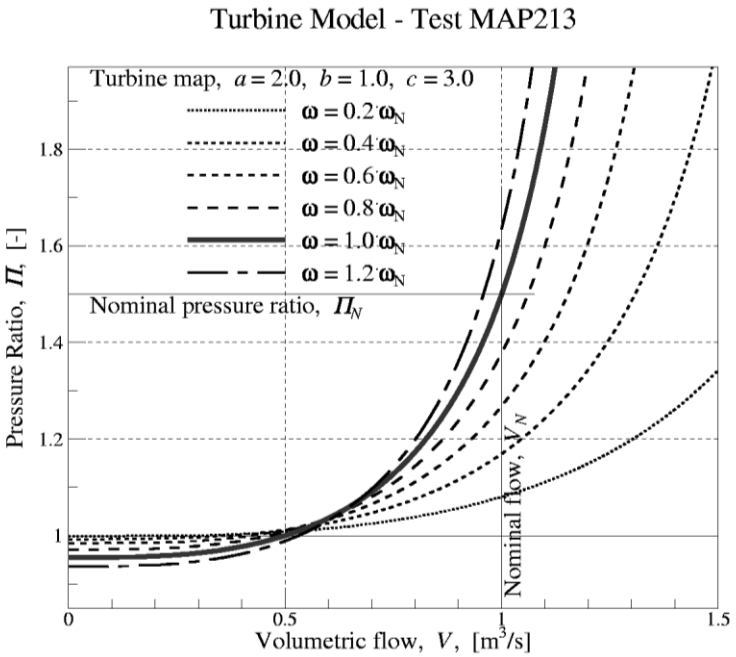


Figure 4-57 Influence of exponents on turbine map, *a*=2.0, *b*=1.0, *c*=3.0.

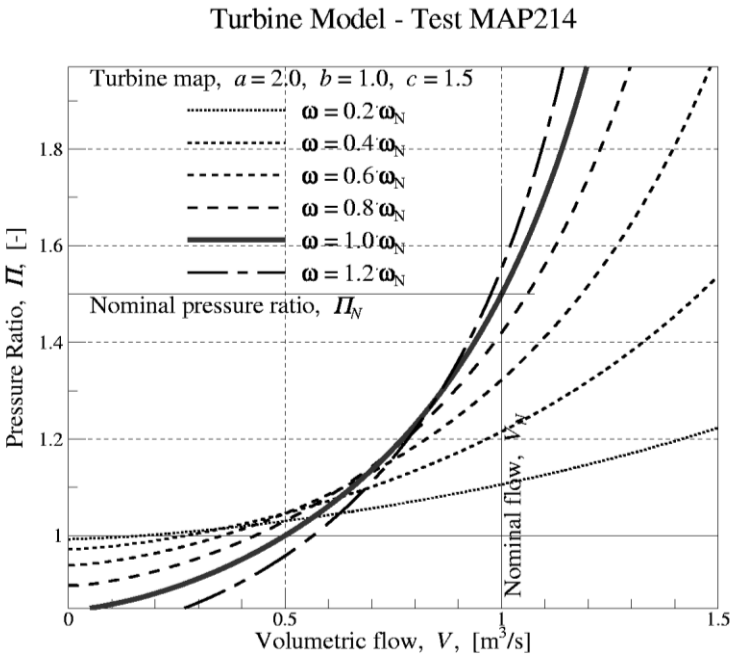


Figure 4-58 Influence of exponents on turbine map, *a*=2.0, *b*=1.0, *c*=1.5.

The default value of C_{RS} is –1.5. The value was chosen based on comparisons with four-quadrant turbine data, shown in [64]. In the previous SPECTRA versions the default value of C_{RS} was equal to 0.1.

The influence of the reverse flow degradation factor, C_{RF} , on the turbine maps is shown in Figure 4-62 and Figure 4-63. The following values were used to produce the maps shown in these figures:

Π_N = 1.5,
V_N = 1.0 m^3/s,
V_0 = 0.5 m^3/s,
ω_N = 50.0 s^{-1},
a = 2.0,
b = 1.0,
c = 2.0,
C_{RS} = –1.5.

The default value of value of C_{RS} was used in both figures. Comparison of Figure 4-62 and Figure 4-63 gives an indication of the influence of the reverse flow degradation factor, C_{RF}. The default value of C_{RF} is 0.05. The value was chosen based on comparisons with four-quadrant turbine data, shown in [64]. In the previous SPECTRA versions the reverse flow factor was not used, which means that C_{RF} was by definition equal to 1.0.

Comparison of the maps obtained with different values of C_{RS} and C_{RF} with measurement data from [64], is shown in Figure 4-65 and Figure 4-66. The turbine nominal data were assumed following the data in [64]:

Π_N = 3.91,
V_N = 3.0 m^3/s,
ω_N = 116.7 s^{-1}, (7000 rev/min).

The values of V_0 and the exponents, suitable to represent reasonably well the turbine behavior in the normal operating conditions (positive speed, positive flow) were selected in a few trials as:

V_0 = 1.0 m^3/s,
a = 2.0,
b = 0.5,
c = 2.0.

The parameters determining the turbine behavior in the reverse speed and reverse flow conditions were selected to be:

C_{RS} = –1.5,
C_{RF} = 0.05.

Figure 4-65 shows the measured power for positive (normal) and negative (reversed) flows and speeds. The positive speed values are plotted using solid lines and empty markers, while the negative speed values are plotted using dashed lines and filled markers.

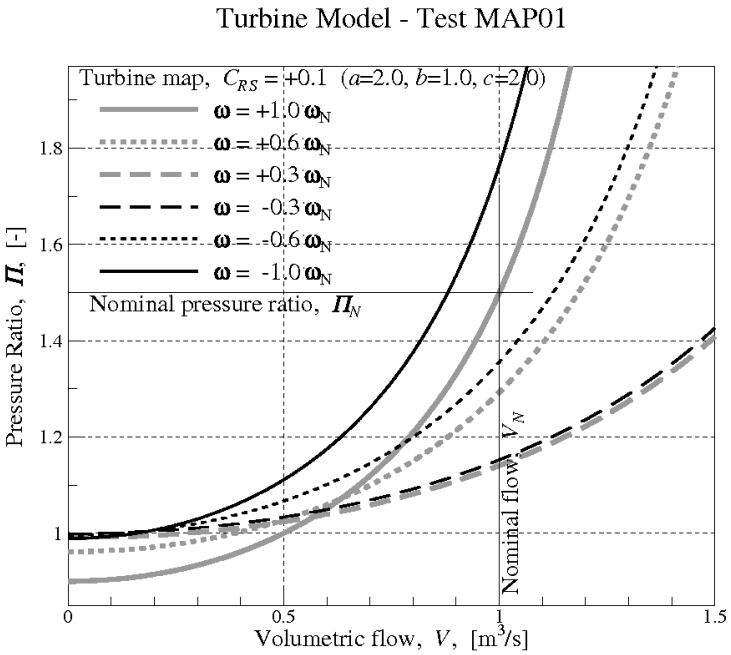


Figure 4-59 Influence of C_{RS} on turbine map, C_{RS}=0.1.

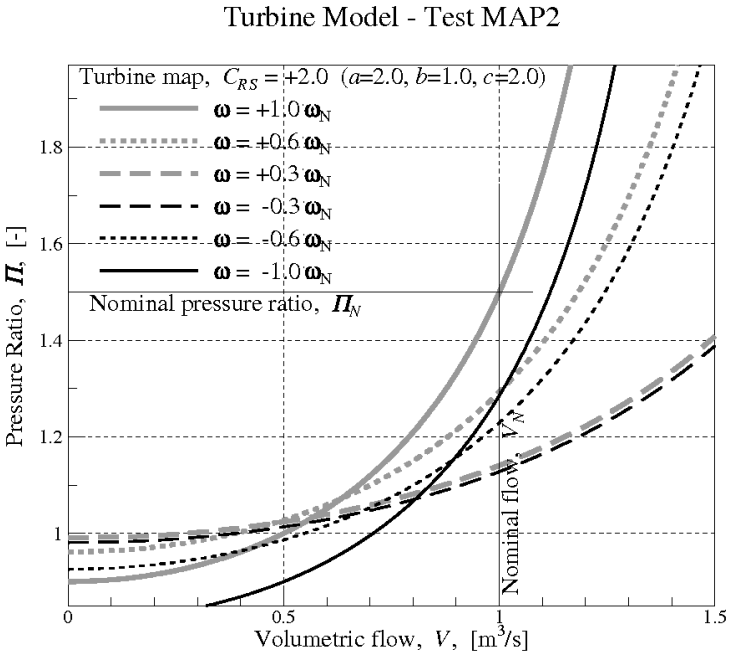


Figure 4-60 Influence of C_{RS} on turbine map, C_{RS}=2.0.

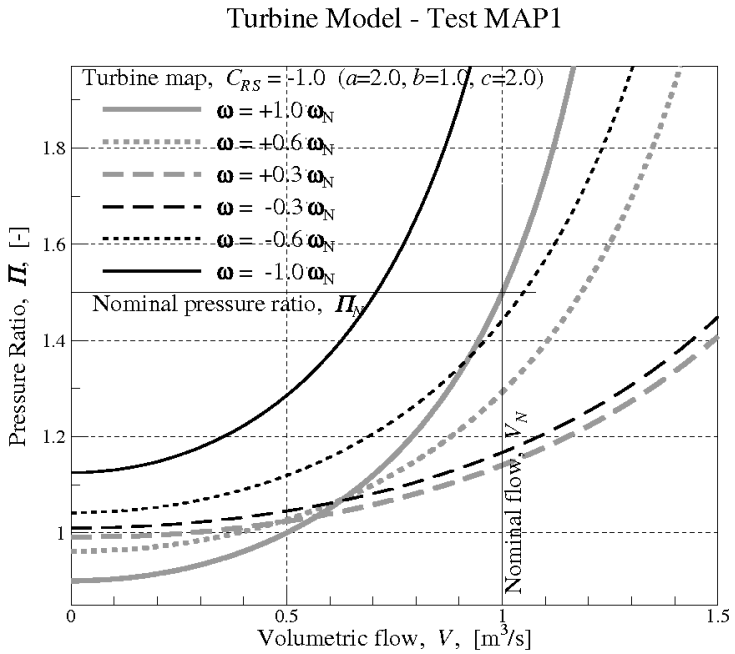


Figure 4-61 Influence of C_{RS} on turbine map, C_{RS}=–1.0.

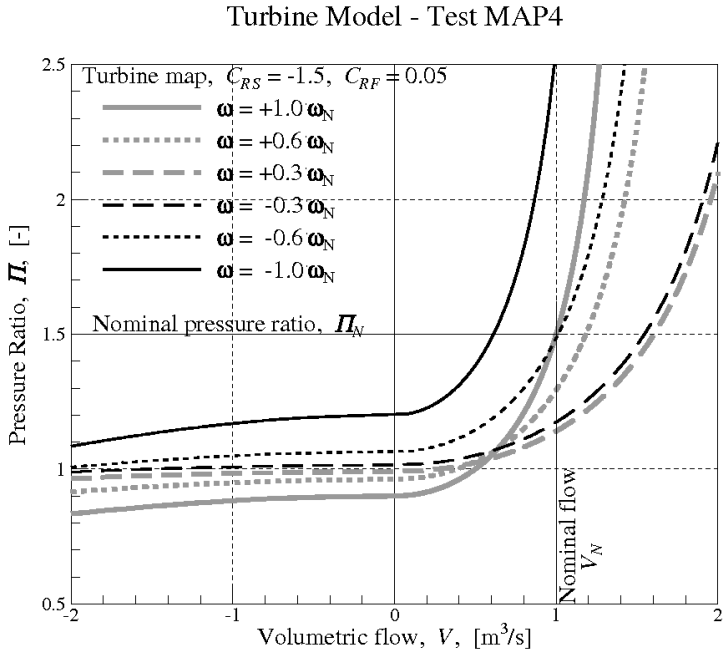


Figure 4-62 Influence of C_{RF} on turbine map, C_{RF}=0.2, C_{RS}=–1.5.

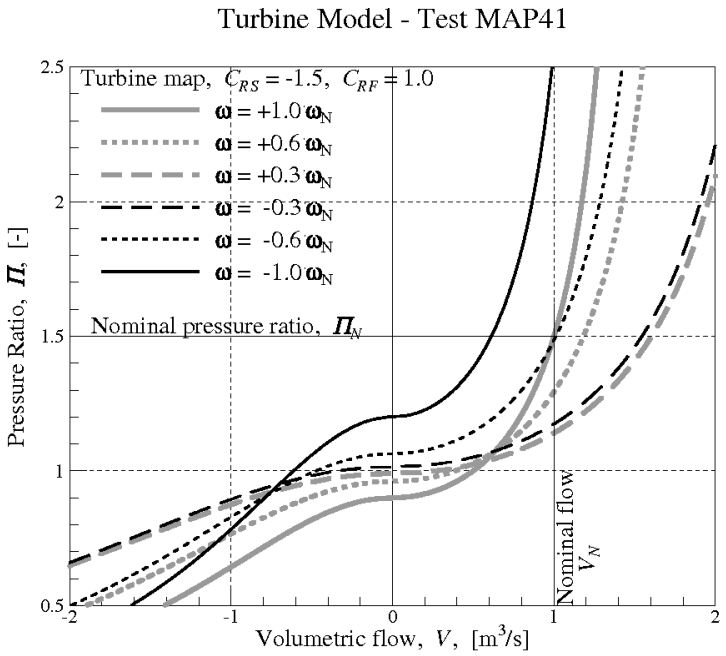


Figure 4-63 Influence of C_{RF} on turbine map, C_{RF}=1.0, C_{RS}=–1.5.

Figure 4-66 shows the calculated values. The positive speed values are plotted using grey lines and empty markers, while the negative speed values are plotted using dark lines and filled markers. The left graph in Figure 4-66 shows values obtained with a constant efficiency of 1.0. In case of reversed speed and normal (positive) flow the fluid decompression is very similar to the decompression occurring in normal speed. Therefore the power calculated with the efficiency of 1.0 is negative (meaning the power is obtained from the machine – watch the scale in Figure 4-66). Based on the measured data it is clear that the power should be positive (the power is consumed by the turbine turning on backwards). This may be best explained using a *h-s* (enthalpy-entropy) graph, shown in Figure 4-64.

In case of an ideal isentropic expansion from a given pressure p_1 to another pressure p_2, one obtains maximum power – this is the line 1-2a. This is an ideal process and it does not occur in practice. A practical process in a turbine at its normal operating conditions is shown by the line 1-2b. In such case some energy is dissipated as heat and the energy that is obtained from the turbine is smaller. The isentropic efficiency is in this case equal to:

$$\eta = \frac{\Delta h_1}{\Delta h_0}$$

A typical turbine has the efficiency of 0.7 - 0.9 in the typical operating conditions. The case of turbine turning on reverse (Figure 4-65 dashed lines), the fluid decompression is shown by the line 1-2c in Figure 4-64. The amount of energy dissipated as heat is so large that the exit enthalpy is higher than the inlet enthalpy. Consequently one has to put energy into the turbine to keep it running. In this case the turbine efficiency is negative, as shown in Figure 4-64.

Looking at Figure 4-65 it is clear that the efficiency is negative in all cases with reverse speed (except maybe the lowest negative speed case, in the negative flow range). For the calculations shown in the right graph in Figure 4-66 the following efficiencies were used:

	Positive flow	Negative (reverse) flow
• N = +7000 rev/min:	η = +0.9	η = +0.6
• N = +3500 rev/min:	η = +0.8	η = +0.4
• N = +1750 rev/min:	η = +0.6	η = +0.2
• N = –1750 rev/min:	η = –3.0	η = –0.1
• N = –3500 rev/min:	η = –5.0	η = –0.3
• N = –7000 rev/min:	η = –7.0	η = –3.0

The values of the efficiencies were obtained in a few trials, in order to provide a reasonably good match to the measured data, shown in Figure 4-65. For simplicity a constant efficiency was defined for each speed, thus the efficiency was in fact only a function of speed: $\eta = \eta(N)$. The efficiency could be made as a function of both flow and speed, $\eta = \eta(V, N)$ and in this way a better match could be obtained.

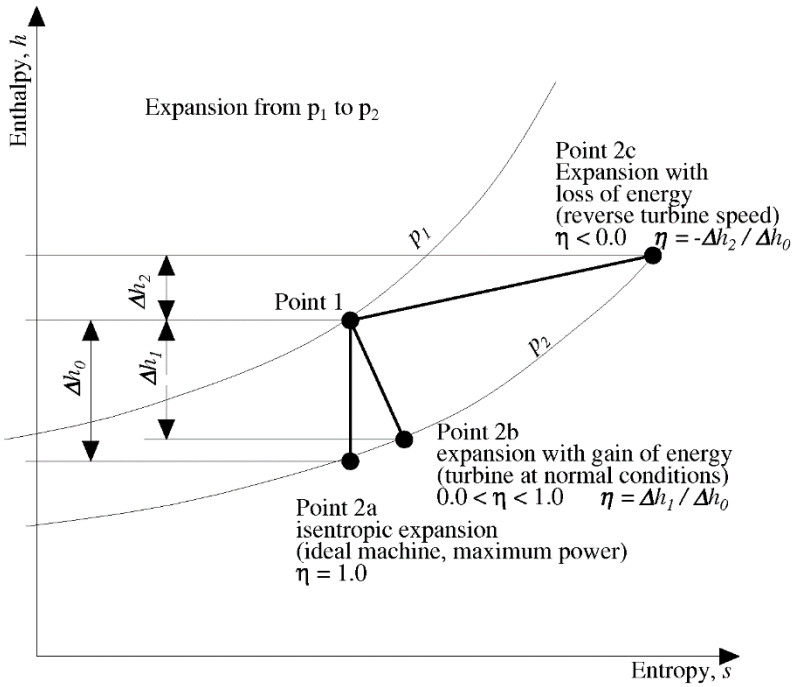


Figure 4-64 Illustration of the gas expansion in a turbine on the *h-s* diagram.

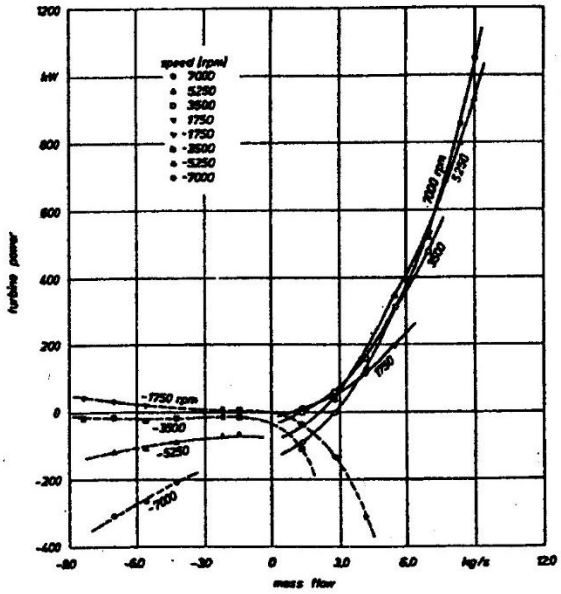


Figure 4-65 Power – reference [64].

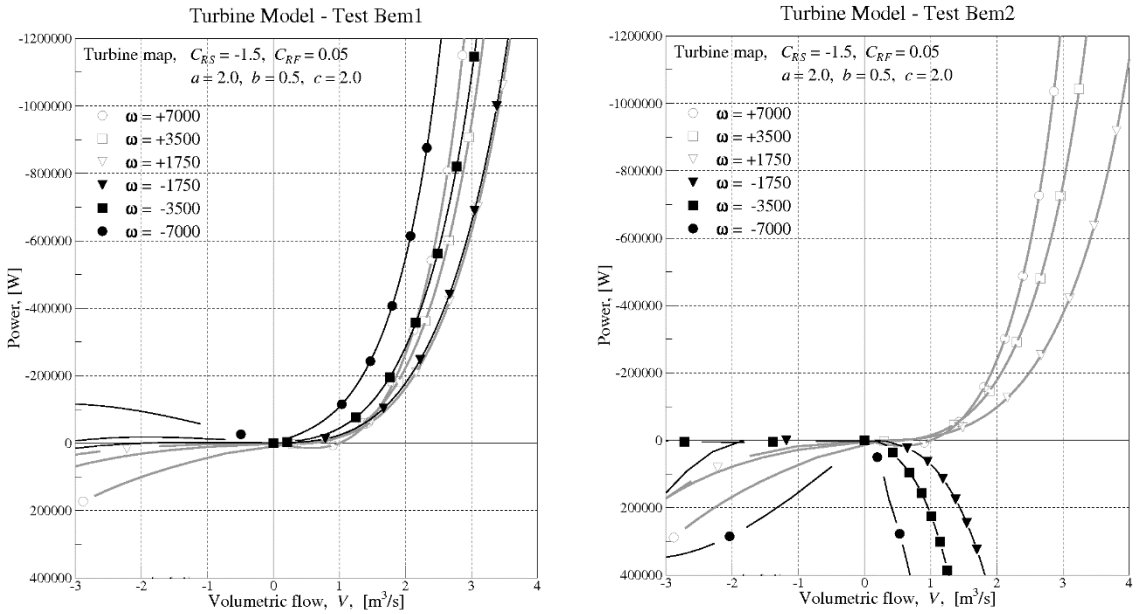


Figure 4-66 Turbine power C_{RS}=–1.5 C_{RF} =0.05, left: only positive efficiency, right: negative efficiency in reverse speed.

4.6.3.3 Power

For gas turbines the power which enters the fluid (atmosphere of CV downstream the turbine) is computed from:

$$Q_{hydr,atms} = \eta V \rho \, c_P \, T \, [\, (1/\Pi)^{(\kappa-1)/\kappa} - 1 \,]$$

Here η is the turbine efficiency, V is the volumetric flow (m^3/s), ρ is the density (kg/m^3), c_P is the specific heat (J/kg/K), T is the inlet temperature (K), Π is the pressure ratio, and Q_{hydr} is the total power source for the fluid (W) (the value is negative during normal turbine operation).

For water turbine the power which enters the fluid (pool of CV downstream the turbine) is computed from:

$$Q_{hydr,pool} = \eta V \Delta P$$

Here η is the turbine efficiency, V is the volumetric flow (m^3/s), ΔP is the pressure head (Pa), and Q_{hydr} is the total power source for the fluid (W) (the value is negative during normal turbine operation).

4.6.4 Simple Pump / Turbine Model

A simple model is available that can be used to represent pumps or turbines. With this model, the user can define the pump head through a Tabular or Control Function (Volume 2, record 236XXX).

$$\Delta P_{SPT} = \begin{cases} TF(XXX) & if \;\; XXX > 0 \\ CF(-XXX) & if \;\; XXX < 0 \end{cases}$$

Here ΔP_{SPT} is the momentum source (pressure head) from the simplified pump / turbine model (Pa), TF and CF are the values of a Tabular and a Control Function respectively.

The value of the head is added to the momentum equation for a given junction. This model is very similar to the QUICK-CF model in MELCOR. The value obtained from the TF / CF may be positive (pump) as well as negative (turbine). No energy effect is associated with this type of pump / turbine.

4.6.5 Valve Model

A valve may be included in any junction. The valve operation is modelled as a change in the fraction of the junction area that is open as well as a change of the pressure loss coefficients. Three types of valves are included: motor valve, check valve, and burst discs. These valve types are described below.

- Motor valve

  The open fraction of a motor valve is controlled by a Tabular Function of time, or a Control Function of any arguments from the program data base. The fraction open is limited to the range 0.0 - 1.0 and, if the controlling function returns a value outside this range, it will be suitably truncated.

- Check valve

  The check valve opens and closes based on pressure difference. The valve operation is determined by the following three parameters:

  Pressure difference to open valve, Δp_{open} Opening of a closed valve is initiated if the pressure upstream the valve junction is larger than the pressure downstream the valve plus Δp_{open}. Note that Δp_{open} can be both positive or negative. Negative values represent check valves which opening is assisted by a spring, such as for example valves on the core flooding lines in SWR-1000.

  Pressure difference to close valve, $\Delta p_{close} < \Delta p_{open}$. Closing of an open valve is initiated if the pressure upstream the valve junction is smaller than the pressure downstream the valve plus Δp_{open}.

  Rate of area change, $(1/A)(\mathrm{d}A/\mathrm{d}t)$. The action of valve opening or closing, once initiated, proceeds with the speed determined by this parameter.

- Burst disc

  The burst disc opens based on pressure difference. Once open it never closes. The opening setpoints are defined for the positive and the negative flow directions. The valve opens when the pressure difference in the positive flow direction exceeds $\Delta p_{open,+}$ or when the pressure difference in the negative flow direction exceeds $\Delta p_{open,-}$. Both $\Delta p_{open,+}$ and $\Delta p_{open,-}$ must be positive.

For both valve types a leakage area fraction can be specified. If it is specified, then the valve flow area is never smaller than the leakage area.

Both motor and check valve (or burst valve) may be used simultaneously. Such valve is referred to as “serial valve”:

- Serial valve

  In this case the valve is open when both motor and check valves are open. This means such a valve is treated as a serial valve with the valve fraction open equal to:

$$\left(\frac{A_V}{A_0}\right) = Min\left[\left(\frac{A_V}{A_0}\right)_{motor}, \left(\frac{A_V}{A_0}\right)_{check}\right]$$

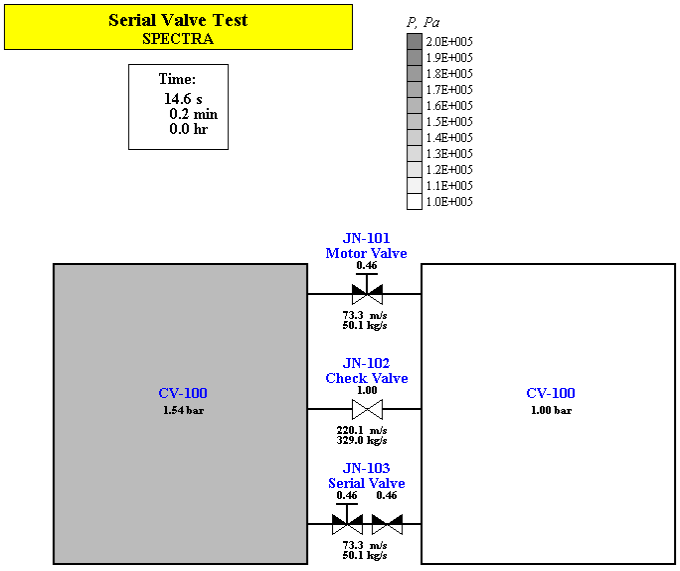


Figure 4-67 Example of a motor valve, a check valve, and a serial valve.

Figure 4-67 shows example of a motor valve, a check valve and a serial valve. The motor valve is controlled by a TF, which at the time shown in the picture gives the fraction open of 0.46. The check valve opens at the pressure difference of 0.5×10^5 Pa, so it is fully open. The serial valve shows the value of TF controlling the motor valve (left valve) and the current valve position (right valve). Since the valve open fraction is equal to the minimum of the check valve (1.0) and the motor valve (0.46), the smaller value is shown in the picture.

In general the pressure loss coefficient of a valve changes with the valve open fraction. Sometimes the change is so significant that it is important to include a variable loss coefficient in the computational model.

- The following expression may be used to calculate the loss factors (forward and reverse):

$$K_V = K_0\left[C_V + \left(1 - C_V\right)\frac{A_V}{A_0}\right]$$

- Tabular Function may be used. The TF value is calculated by the program using the open fraction as an argument. The loss factors (forward and reverse) are calculated from:

$$K_V = K_0 \cdot TF(A_V / A_0)$$

These two methods are described in the following subsections.

4.6.5.1 Pressure Loss Coefficient Given by an Analytical Expression

SPECTRA uses two values to define valve characteristics: K_0 and C_V. The actual loss coefficient is calculated from:

$$K_V = K_0 \cdot \left[C_V + \left(1 - C_V\right) \frac{A_V}{A_0} \right]$$

K_V current loss coefficient, (-)
K_{JN} loss coefficient for fully open valve (junction loss coefficient), (-)
C_V valve coefficient, (-)
A_V current open area, (m^2)
A_{JN} area for fully open valve (junction flow area), (m^2)

The coefficients K_0 and C_V are input parameters. The values of K_0 and C_V may be estimated, using the following relations:

$$C_V = \frac{K_V / K_0 - A_V / A_0}{1 - A_V / A_0}$$

$$K_V \rightarrow K_0 \qquad when\ A_V \rightarrow A_0$$

$$K_V \rightarrow K_0 \cdot C_V \qquad when\ A_V \rightarrow 0.0$$

It is therefore needed to know loss coefficients for fully open and nearly closed valve. These can easily be obtained from appropriate handbooks. Below a calculation procedure is shown, to obtain the coefficients for two valve types: a butterfly valve and a rectangular gate valve.

- Butterfly Valve

  Reference [2] gives the loss coefficients for a butterfly valve in a tube with a circular cross section (Diagram 9-16 in [2]). In case of a thin plane disk (Curve 2 in Diagram 9-16) the values are:

θ°	0	10	20	30	40	50	60	70
K_{Idel}	0.30	0.52	1.54	4.50	11.0	29.0	108.0	625.0

  It must be remembered that the loss factors given in [2], K_{Idel}, are related to the velocity in a fully open channel. To obtain values related to the actual valve flow area one must multiply these values by the area ratio squared: $(A/A_0)^2$. For a given angle, θ, the open area is given by: $A=A_0 \cdot (1-\sin\theta)$. Therefore the loss factors related to the actual flow area are equal to: $K=K_{Idel} \times (1-\sin\theta)^2$. The values of K, as well as K_{Idel}, are shown in Figure 4-68.

The value of K_0, needed for SPECTRA input, is equal to the loss factor for the fully open valve, thus 0.30. It has been found using a trial and error procedure, that a value of C_V=7.0 represents well the K values in Figure 4-68. Thus, for the butterfly valve considered here the appropriate values are:

$$K_0 = 0.30 \qquad C_V = 7.0$$

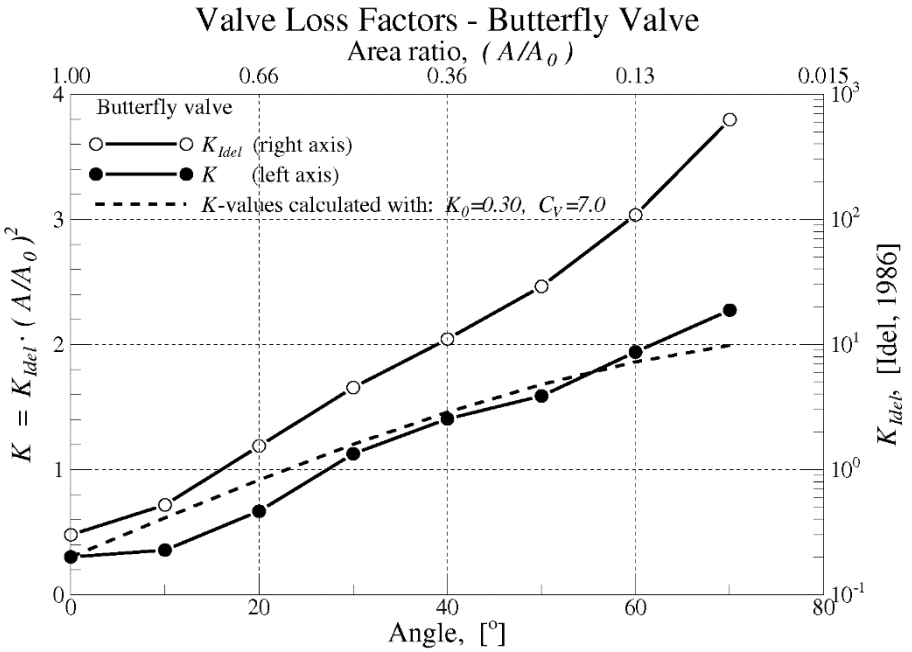


Figure 4-68 Loss factors, butterfly valve.

- Gate Valve

For a rectangular, single disk, gate valve the loss coefficients are given as a function of the ratio of the disk position (h), to the channel width (a) ([2], Diagram 9-5). The values are:

h/a	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
K_{Idel}	193	44.5	17.8	8.12	4.02	2.08	0.95	0.39	0.09

Again the values of K_{Idel} are related to the velocity in a fully open channel. The values are therefore multiplied by the area ratio squared: $(A/A_0)^2$. For a rectangular valve $(A/A_0)^2$ is simply equal to $(h/a)^2$. The values of K, as well as K_{Idel}, are shown in Figure 4-69.

The value of K_0, needed for SPECTRA input, is equal to the loss factor for the fully open valve, thus 0.09. It has been found using a trial and error procedure, that a value of C_V=25.0 represents well the K values in Figure 4-69. Thus, for the rectangular gate valve considered here the appropriate values are:

$$K_0 = 0.09 \qquad C_V = 25.0$$

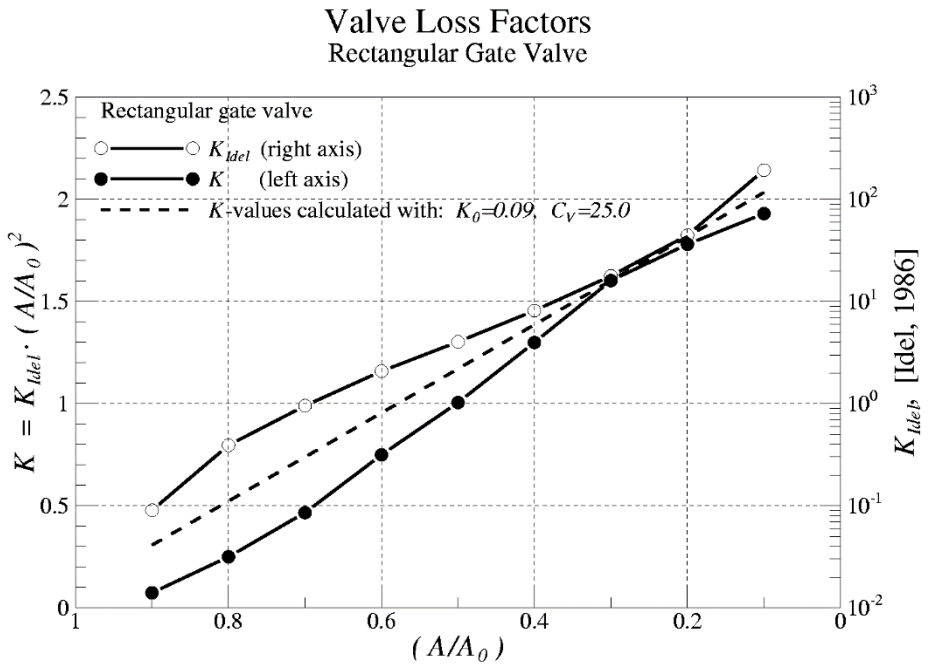


Figure 4-69 Loss factors, gate valve.

4.6.5.2 Pressure Loss Coefficient Given by a Table

With this option the pressure loss factors (forward and reverse) are calculated from:

$$K_V = K_0 \cdot TF(A_V / A_0)$$

K_0 is the constant multiplier on the pressure loss coefficient (forward and reverse)
TF value of Tabular Function evaluated for the current valve open fraction, A_V/A_0, (-)

For example, in case of a butterfly valve the values are:

θ°	0	10	20	30	40	50	60	70
A/A_0	1.00	0.826	0.658	0.500	0.357	0.234	0.134	0.0603
K_{Idel}	0.30	0.52	1.54	4.50	11.0	29.0	108.0	625.0
K	0.30	0.355	0.667	1.13	1.40	1.59	1.94	2.27

The input deck defining such valve is discussed in Volume 3.

4.6.6 Flow Composition Parameters

A model is provided which allows to restrict or promote flow of one, or several gases, from a gas mixture. This model is primarily intended for a conservative studies. For example, in advanced BWR containments, if the steam flow through vacuum breakers is restricted, then the obtained containment pressure will be typically more conservative (higher), because more noncondensables will be present in the drywell, where it will adversely affect the performance of containment coolers such as PCC.

If the flow composition parameters are used, then the actual concentration of gas i in the atmosphere flow through a junction is calculated from the following formula:

$$x_{i,JN}^{atms} = \frac{x_{i,CV}^{atms} \cdot FCP_i}{\sum_{k=1}^{N_{gas}} x_{k,CV}^{atms} \cdot FCP_k}$$

Here $x_{i,JN}^{atms}$, $x_{i,CV}^{atms}$ are the volumetric fractions of gas i in atmosphere flowing through junction JN, and in the Control Volume (local fraction at the junction elevation is used in case of stratifications). The values of flow composition parameters, FCP_i, are defined in input data. A restriction is imposed on FCP_i : $10^{-3} \le FCP_i \le 10^3$.

The influence of flow composition parameters is illustrated in Figure 4-70. The figure gives the dependence of the gas composition in a junction, versus the gas composition in the source volume, for gas i, in case of FCP_i ranging from 10^{-3} to 10^3. The flow composition parameters for gases other than *i* were all set to 1.0.

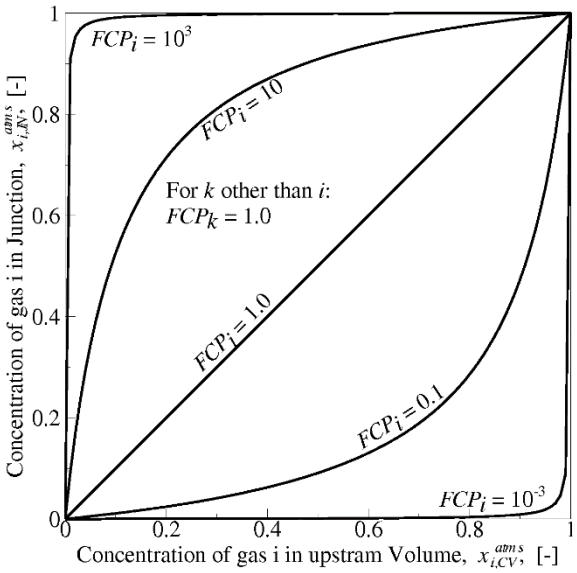
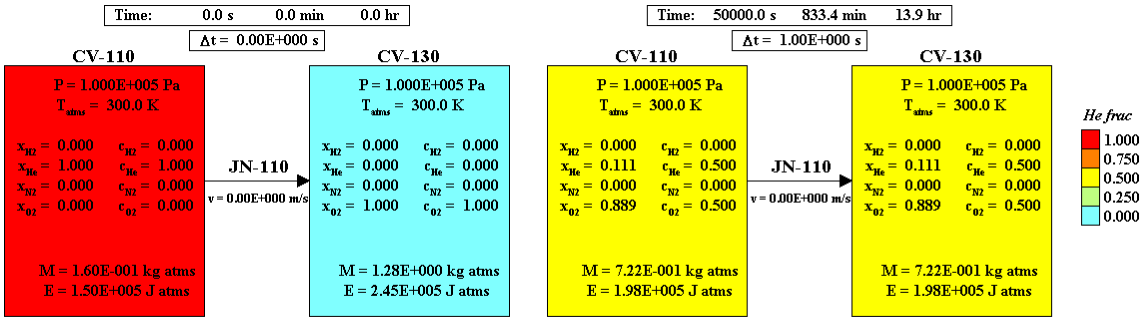


Figure 4-70 Influence of FCP on gas composition in a Junction

4.6.7 Diffusion Model

This section discusses transport of gases from one Control Volume to another by diffusion. The model needs to be activated by the user (see Volume 2, IDIFJN). Transport is based on differences in gas concentrations in the neighboring Control Volumes. Transport by diffusion is important only if there is no flow or very small flow through a junction.

Consider two Control Volumes, filled by gases and connected by a junction, as shown in Figure 4-71. The pressures and temperatures are identical in both volumes. The only differences between the volumes is the gas composition; the left volume, CV-110, is filled with Helium while the right volume, CV-130, is filled with Oxygen. Since there is no pressure difference, there will be no transport through the connecting junction, JN-110, and in absence of diffusion the conditions shown in Figure 4-71 (a) will not change in time.



(a) initial state (t = 0.0 s) (b) final state (t = 50,000 s)

Figure 4-71 Diffusion test case

If the diffusion model is turned on, the code calculates transfer of gases through the connecting junction using the diffusion equation:

$$J_i = -D_i \cdot \frac{dC_i}{dx}$$

J_i volumetric flux of gas i, (m/s)
D_i diffusion coefficient of gas i in the mixture of gases, (m^2/s)
C_i molar concentration of gas i in the mixture of gases, (-)
dC_i change of concentration of gas i in the mixture of gases, (-)
dx flow length, (m)

The diffusion equation is written for a junction with a certain diffusion length, L_{JN}:

$$J_i = D_i \cdot \frac{\Delta C_i}{L_{JN}}$$

ΔC_i difference in concentration of gas i in the connected Control Volumes:
$\Delta C_i = C_i$('from-CV') – C_i('to-CV')
L_{JN} diffusion length of the junction JN, (m)

The main elements of the model, namely the diffusion coefficient, the conservation of volume of exchanged gases and the effect of the gas velocity are discussed below.

- ***Diffusion coefficient***
  The diffusion coefficient is calculated as described in section 3.5.2.3. The diffusion coefficient of gas *i* in a gas mixture is obtained using the method of Blanc, with binary diffusion coefficients calculated using the method of Fuller. The diffusion coefficient of gas *i* is calculated for both mixtures of gases in the 'from-CV' and the 'to-CV'. Considering that the diffusion coefficient in the 'from-CV' and the 'to-CV' is in general different, the diffusion equation is written as::

$$J_i = \frac{\Delta C_i}{R_i}$$

  where R_i (diffusion resistance) is equal to:

$$R_i = \frac{L_{from}}{D_{i,from}} + \frac{L_{to}}{D_{i,to}}$$

  Here L_{from} and L_{to}, are the diffusion lengths in the 'from-CV' and the 'to-CV' respectively. The values are defined in the input deck (Volume 2: X1DFJN, X2DFJN). The default values are equal to the half of the junction friction length.

- ***Conservation of the exchanged volume of gas***
  The volume of gas transported by diffusion is equal to :

$$\frac{\Delta V_i}{\Delta t} = A_{JN} \cdot \frac{\Delta C_i}{R_i}$$

  A_{JN} flow area of the atmosphere gas flow through the junction JN, (m^2)
  $\Delta V_i/\Delta t$ volumetric rate of change of gas *i* due to diffusion (m^3/s)

  The volumetric change, ΔV_i, is calculated for each gas. If only two gases are diffusing, as is shown in Figure 4-71, the net effect is zero, i.e. the positive value of ΔV_{He} will be the same as the negative value of ΔV_{O2}. However, in a more general case of multiple gases present in both connected volumes this is not necessarily the case. The following procedure is applied to make sure that the net volume transferred by diffusion is zero. First the sum of positive and negative terms is calculated:

$$\Delta V^+ = \sum_i \Delta V_i \qquad for\ \Delta V_i > 0$$
$$\Delta V^- = \sum_i |\Delta V_i| \qquad for\ \Delta V_i < 0$$

Next, the smaller (more restrictive) absolute value is selected as the true value. The values (either the positive or the negative ones) that give larger absolute sum, are re-scaled to give the same total value as the smaller absolute sum. This way the program assures that the net volume exchanged in the diffusion process is always zero. In other words, the diffusion does not cause any gas flow that would inevitably appear if $\Delta V^{+} \neq \Delta V^{-}$. The absence of such flows is shown in the diffusion test cases shown in Volume 3.

- ***Effect of gas velocity***

  The diffusion velocities (volumetric fluxes) are very small. Typical values (see Volume 3) are smaller that 10^{-3} m/s. If there is no gas flow, or the gas velocities are very small, then the diffusion caused by differences in gas concentrations is important. However, when the gas velocities are large compared to the diffusion velocities, the transport by diffusion is not possible.

  In order to take that into account, a velocity limit is introduced, above which diffusion is not calculated (set to 0.0). In order to avoid an abrupt change of the diffusion term, that could lead to numerical difficulties, an interpolation zone is defined where the diffusion term is interpolated from the full value to zero. In the interpolation zone the diffusion velocity is obtained from:

$$J_i(v_{gas}) = \frac{VMDFJN - J_i(v_{gas} = 0.0)}{VMDFJN - VMDFJN \times XMDFJN}$$

  VMDFJN velocity above which diffusion mass transfer calculation is bypassed (m/s)
  XMDFJN interpolation range for diffusion mass transfer calculation (-)

  The default values are (see Volume 2) VMDFJN=0.01 m/s, XMDFJN=0.5. The velocity-dependent multiplier obtained for those values is shown in Figure 4-72.

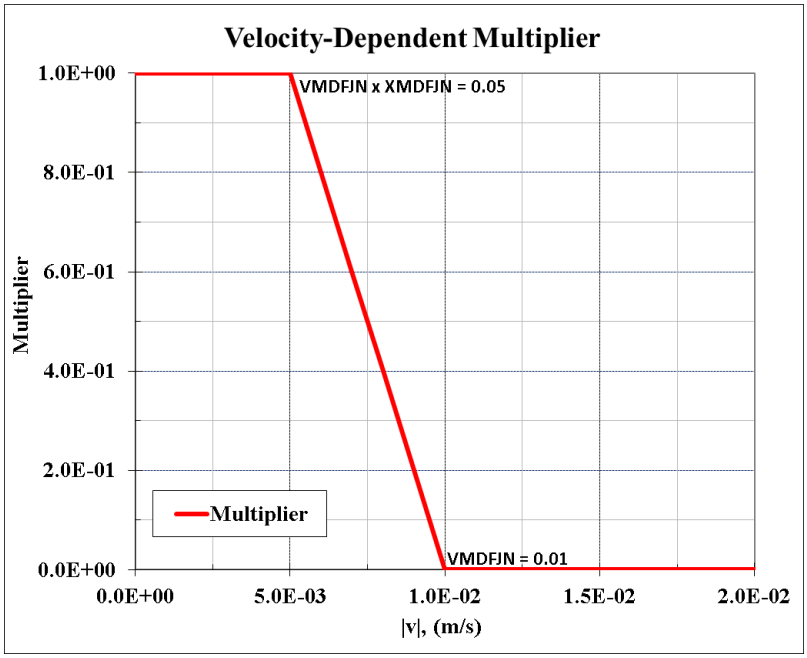


Figure 4-72 Velocity-dependent multiplier for diffusion

In the example test presented in Figure 4-71, the gases will fully mix by diffusion in roughly 10 hours. The gas velocity equal to zero all of the time. More details about this test case are shown in Volume 3.

The diffusion model is applied to the gases present in Control Volumes, defined in the FL Package. Additionally, the diffusion model may be applied for the fission product vapors, defined in the RT Package. In order to do so, the user must also supply the diffusion volumes (SGFPRT - see Volume 2), needed to calculate diffusion coefficients. The values of diffusion volumes for frequently used gases are shown in .Table 3-6.

5 1-D Solid Heat Conductor Package

5.1 Introduction

The 1-D Solid Heat Conductor Package calculates heat conduction within solid structures, and energy transfer across its boundary surfaces into control volumes.

A 1-D Solid Heat Conductor is a structure that is represented by one-dimensional heat conduction, with specified boundary conditions at each of its two boundary surfaces. The modeling capabilities of solid heat conductors are general, and can represent walls, containment structures, fuel rods with nuclear or electrical heating, piping walls, heat exchangers with smooth or finned tubes, etc.

Three different geometries are available for the 1-D Solid Heat Conductors:

- Rectangular geometry.
- Cylindrical geometry.
- Spherical geometry.

The rectangular and cylindrical heat conductors may have basically two orientations in space:

- Vertical orientation.
- Horizontal orientation.

The heat and mass transfer models contain correlations valid for low inclination, therefore a third orientation is possible:

- Low inclination (inclination angle < 20°).

The conduction is calculated within solid heat conductors by dividing each conductor into N "mesh cells", or "mesh intervals". The nodalization is specified by user on input and may be non-uniform, i.e. the distance between the mesh cell boundaries need not be the same. Cell 1 is always at the left boundary of a rectangular conductor, or at the inside boundary surface for a cylindrical or a spherical geometry. Cell N is always at the right boundary of a rectangular conductor, or at the outer boundary surface for a cylindrical or a spherical geometry.

Nodalization of the interior of a solid heat conductor is shown in Figure 5-1. Each mesh cell may contain a different material. The thermo-physical properties of the materials, such as thermal conductivity, k, specific heat, c_p, and density, ρ, are specified by user input, as functions of temperature. The nodes where temperature is defined are marked in Figure 5-1. The nodes are assumed to be located at the centers of the cells in case of interior cells, and at the surface in case of boundary cells. The boundary cells are therefore considered to be half-filled-cells, or simply "half-cells". This concept is similar to the one applied for 2-D Solid Heat Conductors, which leads to half-cells, one-quarter-cells, and three-quarter cells - see Chapter 6, but in contrast to the 2-D Conductors the half-cells are quite "transparent" for the user - see section 5.2.

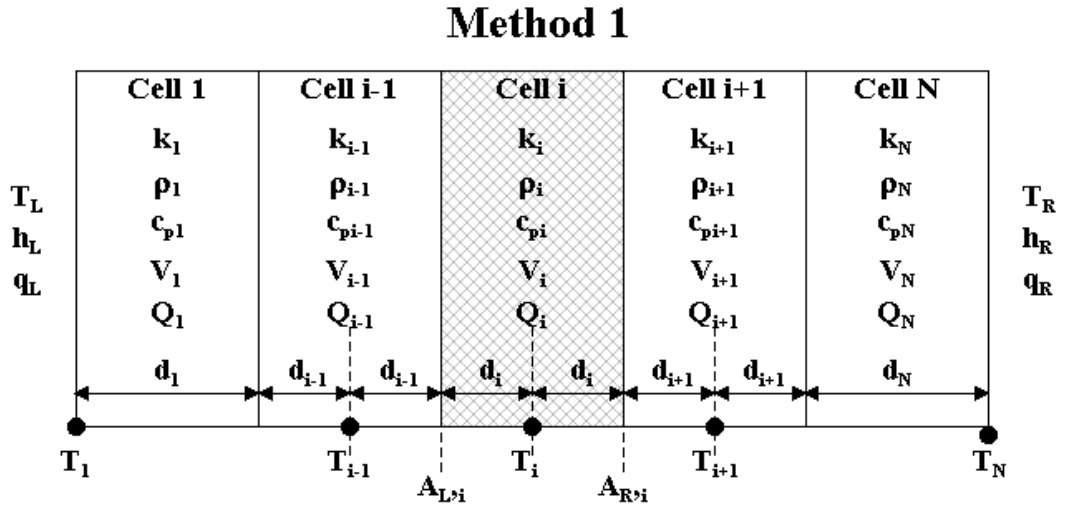


Figure 5-1 Nodalization of a 1-D Solid Heat Conductor, Method 1

An internal power source may be specified for each solid heat conductor. The space distribution is specified by user input and may vary for each mesh interval (Q_i). The time dependence is given by a user specified tabular function of time or control function of any variables from the program.

Each 1-D Solid Heat Conductor has two surfaces - left and right. In case of cylindrical and spherical geometries, the inner surface is the left surface, and the outer surface is the right surface of a solid heat conductor. At each surface the boundary conditions must be specified. A boundary condition consists of three parameters:

- Fluid temperature, T_{fluid}
- Convective heat transfer coefficient, h
- Non-convective heat flux, q_r (typically used to model thermal radiation).

The total boundary heat flux consists of the convective flux, $h\times(T_{wall} - T_{fluid})$, and the non-convective flux, q_r:

$$q_{tot} = h \cdot (T_{wall} - T_{fluid}) + q_r$$

The following types of boundary conditions may be specified:

- Adiabatic, $h = 0.0$, $q_r = 0.0$.

- User specified, with heat transfer coefficient, fluid temperature and non-convective heat flux given by tabular function of time, or control function of any parameters in the program.

- Heat transfer to a Control Volume, with fluid temperature taken from this control volume, which is specified as the boundary volume, and heat transfer coefficient calculated by the Heat and Mass Transfer Package (Chapter 7). The non-convective flux is calculated either by the Thermal Radiation Package (Chapter 8), or by the simplified radiation model (section 5.9).

Any combination of the above boundary conditions may be used. For example, heat transfer to a control volume may be used with the non-convective flux, q_r, equal to zero, or specified by tabular or control function.

If heat transfer to a boundary control volume is used, it is possible to use tabular or control function to define the heat transfer coefficient, rather than using the default models from the Heat and Mass Transfer Package. The fluid temperature however is in such case always that from the boundary control volume and cannot be redefined by a tabular or control function.

If a control volume is specified as boundary volume for either the left or the right surface, then the entire surface must fit within the boundary volume. That is, the bottom of the surface must equal or exceed the elevation of the bottom of a control volume, and the top of the surface must not exceed the top of the control volume. It is also forbidden to place a horizontal flat surface (rectangular geometry) facing up at the top elevation of the boundary control volume, or flat surface facing down at the bottom elevation of the control volume.

The mathematical treatment of 1-D Solid Heat Conductors is described in the sections 5.2 through 5.9. The conduction equation is discussed in section 5.2 and 5.3. Sections 5.4 and 5.5 describes how the boundary conditions are defined in case when a surface is partly covered with pool. Section 5.6 describes the temperature averaging concept, important for modeling heat exchangers. Section 5.7 describes how the boundary conditions are defined when there is a simultaneous heat transfer to pool and atmosphere. Section 5.8 describes the treatment of extended surfaces, such as fins and spines. Finally, a simplified radiation model is described in section 5.9.

5.2 Transient Heat Conduction (Method 1)

The general form of transient heat conduction equation is ([20], chapter 3):

$$\rho \cdot c_p \cdot \frac{\partial T}{\partial t} = \nabla\left(k \cdot \nabla T\right) + q_V$$

T	temperature, (K)
t	time, (s)
ρ	density, (kg/m^3)
c_p	specific heat, (J/kg/K)
k	thermal conductivity, (W/m-K)
q_V	internal heat source per unit volume, (W/m^3)

In case of one-dimensional heat conduction, with the material properties depending on temperature, and the internal heat source being a function of time, the equation is written as:

$$\rho(T) \cdot c_p(T) \cdot \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k(T) \cdot \frac{\partial T}{\partial x} \right) + q_V$$

The above equation is a parabolic partial differential equation. This equation has to be solved numerically with appropriate boundary conditions.

To solve the heat conduction equation, a finite difference version of the equation is constructed. In order to do that, the derivatives are approximated by the finite differences. For the node *i* the time derivative is approximated as:

$$\frac{\partial T_i}{\partial t} = \frac{T_i - \overline{T_i}}{\Delta t}$$

Δt is the time step, and $\overline{T_i}$ is the temperature of the node *i* at the beginning of the time step. Similarly the space derivative between the nodes *i*–1 and *i* may be approximated by:

$$\left(\frac{\partial T}{\partial x}\right)_{i-1,i} = \frac{T_i - T_{i-1}}{x_i - x_{i-1}}$$

Note that in the above approximation the end of time step temperatures are used: T_i, T_{i-1}. That means an implicit solution scheme is applied, which gives stable solution, independently of the time step size [21]. If the thermal conductivity was constant between the nodes *i*–1 and *i*, then the heat flux would be given simply by multiplying the derivative by *k*: $q = -k\cdot(T_i - T_{i-1})/(x_i - x_{i-1})$. However, the material properties (ρ, c_p, k) may be different in different cells. The node-to-node heat flux sees two cell materials, which in general may have different properties. The heat flux, written from the node *i*–1 to the node *i*, is obtained by using a summed resistance [21]:

$$q = -k \cdot \left(\frac{\partial T}{\partial x}\right)_{i-1,i} = \frac{T_{i-1} - T_i}{\frac{d_{i-1}}{k_{i-1}} + \frac{d_i}{k_i}}$$

d_i half-thickness of the cell *i* for interior cells, full thickness for boundary cells, (m)
k_i thermal conductivity of the material in the node *i*, (W/m-K)

Note that with the adopted scheme the temperature nodes are assumed to be located at the centers of the cells (see Figure 5-1), for all except the left and right boundary cells. The boundary cells are considered as "half-cells", with temperatures defined at the boundary surface. In the input deck, the concept of "half-cells" is not visible for the user. In the input deck, the user always specifies CELLSC as the cell thickness, and the code internally assumes that the parameter d_i, (Figure 5-1) needed for the conduction equation. The relation between CELLSC and d_i is:

$$CELLSC = \begin{cases} d_i & for\ boundary\ cells \\ 2d_i & for\ interior\ cells \end{cases}$$

Note that in order to have the same node-to-node lengths, thickness of the boundary cells should be half of the thickness of the interior cells. To make the node locations clear, the code prints out the location of every node in the output file, e.g.:

```
       NODE POS.    TEMPERATURE
 NODE      [m]          [K]
   1  1.00000E-02  3.00000E+02
   2  1.10000E-02  3.00000E+02
 . . .
```

The heat flux written for the nodes i and i+1, is very similar way as for the nodes i–1 and i. After simple transformations the finite difference approximation of the heat conduction equation takes the following form:

- Interior nodes, $1 < i < N$ (shaded area in Figure 5-1) :

$$V_i \rho_i c_{p,i} \cdot \frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}} \left(T_{i-1} - T_i \right) + \frac{A_{R,i}}{R_{R,i}} \left(T_{i+1} - T_i \right) + Q_i$$

$A_{L,i}$ heat transfer area at the left boundary of cell i, (m^2)
$A_{R,i}$ heat transfer area at the right boundary of cell i, (m^2)
V_i volume of cell i, (m^3)
Q_i internal heat power in cell i, (W)
$R_{L,i}$ conduction resistance between the node i–1 and i, (m^2-K/W)
$R_{R,i}$ conduction resistance between the node i and i+1, (m^2-K/W)

The conduction resistances, $R_{L,i}$ and $R_{R,i}$, are defined as:

$$\textit{left resistance} \quad R_{L,i} = \frac{d_{i-1}}{k_{i-1}} + \frac{d_i}{k_i}$$

$$\textit{right resistance} \quad R_{R,i} = \frac{d_{i+1}}{k_{i+1}} + \frac{d_i}{k_i}$$

- Left boundary node, $i = 1$:

$$V_i \rho_i c_{p,i} \cdot \frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}} \left(T_{fluid,L} - T_i \right) + \frac{A_{R,i}}{R_{R,i}} \left(T_{i+1} - T_i \right) + Q_i - A_{L,i} q_{r,L}$$

with: $T_{fluid,L}$, h_L, and $q_{r,L}$, being the fluid temperature, the heat transfer coefficient, and the non-convective (radiative) heat flux on the left surface, respectively. The conduction resistances are in this case equal to:

$$\textit{left resistance} \quad R_{L,i} = \frac{1}{h_L}$$

$$\textit{right resistance} \quad R_{L,i} = \frac{d_{i+1}}{k_{i+1}} + \frac{d_i}{k_i}$$

- Right boundary node, $i = N$:

$$V_i \rho_i c_{p,i} \cdot \frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}} \left(T_{i-1} - T_i \right) + \frac{A_{R,i}}{R_{R,i}} \left(T_{fluid,R} - T_i \right) + Q_i - A_{R,i} q_{r,R}$$

with: $T_{fluid,R}$, h_R, and $q_{r,R}$, being the fluid temperature, the heat transfer coefficient, and the non-convective (radiative) heat flux on the right surface, respectively.

The conduction resistances are in this case equal to:

$$\textit{left resistance} \qquad R_{L,i} = \frac{d_{i-1}}{k_{i-1}} + \frac{d_i}{k_i}$$

$$\textit{right resistance} \quad R_{L,i} = \frac{1}{h_R}$$

The above formulae present a set of N equations, with unknown temperatures T_i. This equation set may be written shortly in a matrix form:

$$AT = B$$

The matrix $\boldsymbol{A}$ is a tridiagonal matrix, because the equations for internal nodes contain three unknown variables: T_{i-1}, T_i, and T_{i+1}. Therefore the matrix equation may be written as:

$$a_{i-1,i} \cdot T_{i-1} + a_{i,i} \cdot T_i + a_{i+1,i} \cdot T_{i+1} = b_i$$

The matrix coefficients, $a_{i,j}$, and the right-hand side terms, b_i, are equal to:

- Interior nodes, $1 < i < N$:

$$a_{i-1,i} = -\frac{A_{L,i}}{R_{L,i}}$$

$$a_{i+1,i} = -\frac{A_{R,i}}{R_{R,i}}$$

$$a_{i,i} = \frac{V_i \rho_i c_{p,i}}{\Delta t} + \frac{A_{L,i}}{R_{L,i}} + \frac{A_{R,i}}{R_{R,i}}$$

$$b_i = \frac{V_i \rho_i c_{p,i} \overline{T_i}}{\Delta t} + Q_i$$

- Left boundary node, $i = 1$:

$$a_{i+1,i} = -\frac{A_{R,i}}{R_{R,i}}$$

$$a_{i,i} = \frac{V_i \rho_i c_{p,i}}{\Delta t} + \frac{A_{L,i}}{R_{L,i}} + \frac{A_{R,i}}{R_{R,i}}$$

$$b_i = \frac{V_i \rho_i c_{p,i} \overline{T_i}}{\Delta t} + Q_i + T_{fluid,L} \cdot \frac{A_{L,i}}{R_{L,i}} - A_{L,i} \cdot q_{r,L}$$

- Right boundary node, $i = N$:

$$a_{i-1,i} = -\frac{A_{L,i}}{R_{L,i}}$$

$$a_{i,i} = \frac{V_i \rho_i c_{p,i}}{\Delta t} + \frac{A_{L,i}}{R_{L,i}} + \frac{A_{R,i}}{R_{R,i}}$$

$$b_i = \frac{V_i \rho_i c_{p,i} \overline{T_i}}{\Delta t} + Q_i + T_{fluid,R} \cdot \frac{A_{R,i}}{R_{R,i}} - A_{R,i} \cdot q_{r,R}$$

The tridiagonal matrix is solved using the procedure specifically suitable for this type of matrices (section 17.4). Since the properties are temperature dependent, the matrix coefficients are not known at the start of calculations and a small internal iteration is required. To begin the iterations the old (beginning of time-step) temperatures are used as a first guess to calculate the matrix coefficients. In the subsequent iterations the new (calculated) node temperatures are used as the guessed temperatures and the matrix coefficients are updated accordingly. The iteration is performed until the discrepancy between the guessed and calculated temperature is for each node smaller than the convergence criterion, chosen as 10^{-8} relative error. The convergence of this internal iteration is typically very fast.

The values of boundary conditions, namely fluid temperatures, heat transfer coefficients, and non-convective fluxes, which are applied in the conduction equation are the end of time step values, which are obtained in the main iteration loop to obtain implicit solution. At the beginning of each time advancement the old-time-step values are used as the first guess for the end-of-time-step value. Next, the new (end-of-time-step) values are updated by the Solver, in the main iteration loop. During each iteration the conduction equation is re-solved with the updated values of boundary parameters. The main iteration continues until all variables of the system are converged, that means until the difference between the value assumed for iteration and obtained during iteration is smaller than the desired criteria (see Chapter 19).

The conduction model described above is quite general and accurate in practical applications. Below two example problems are shown, where the model results are compared to analytical solutions of the heat conduction equations.

- *Example Problem 1: Steady State Conduction, Variable Thermal Conductivity*

  The steady state conduction in a conductor with variable thermal conductivity is considered for two different geometries: rectangular and cylindrical. The left and right surface temperatures are known. The thermal conductivity is linearly dependant on local temperature as:

  $$k = k_0 \cdot (1 + a_0 \cdot T)$$

  The theoretical solutions for rectangular and cylindrical geometries result in the following steady state temperature distribution inside the conductors:

  - Rectangular geometry ([20], chapter 3, equation 112):

  $$T(x) = \frac{\sqrt{(1 + a_0 \cdot T_L)^2 - [(1 + a_0 \cdot T_L)^2 - (1 + a_0 \cdot T_R)^2] \cdot x / D} - 1}{a_0}$$

- Cylindrical geometry ([20], chapter 3, equation 130):

$$T(r) = \frac{\sqrt{(1+a_0 \cdot T_L)^2 - [(1+a_0 \cdot T_L)^2 - (1+a_0 \cdot T_R)^2] \cdot \ln(r/R_L)/\ln(R_R/R_L)} - 1}{a_0}$$

In the above formulae D is the thickness of rectangular conductor (in the present example assumed to be equal to 0.1 m), R_L, R_R are left and right radii of the cylinder (assumed to be equal to 0.1, 0.2 m), and T_L, T_R are the temperatures at the left and right surface respectively (assumed to be equal to 300, 500 K). Calculations were performed for thermal conductivity equal to:

$$k = 1.0 \cdot (1 + 0.01 \cdot T)$$

The heat conductor was nodalized using 11 nodes with equal distance between the nodes, that means cell thickness of 0.01 m for internal nodes and 0.005 m for the boundary nodes. The boundary temperatures were set by tabular functions, defining the desired fluid temperatures, and a very large heat transfer coefficient (10^{99}) to minimize the convective resistances. Calculations were performed until stable temperatures were obtained (10,000 s). SPECTRA results are shown in Figure 5-3 and Figure 5-3. The calculated temperatures are compared to the theoretical values in Figure 5-3. A very good agreement is observed.

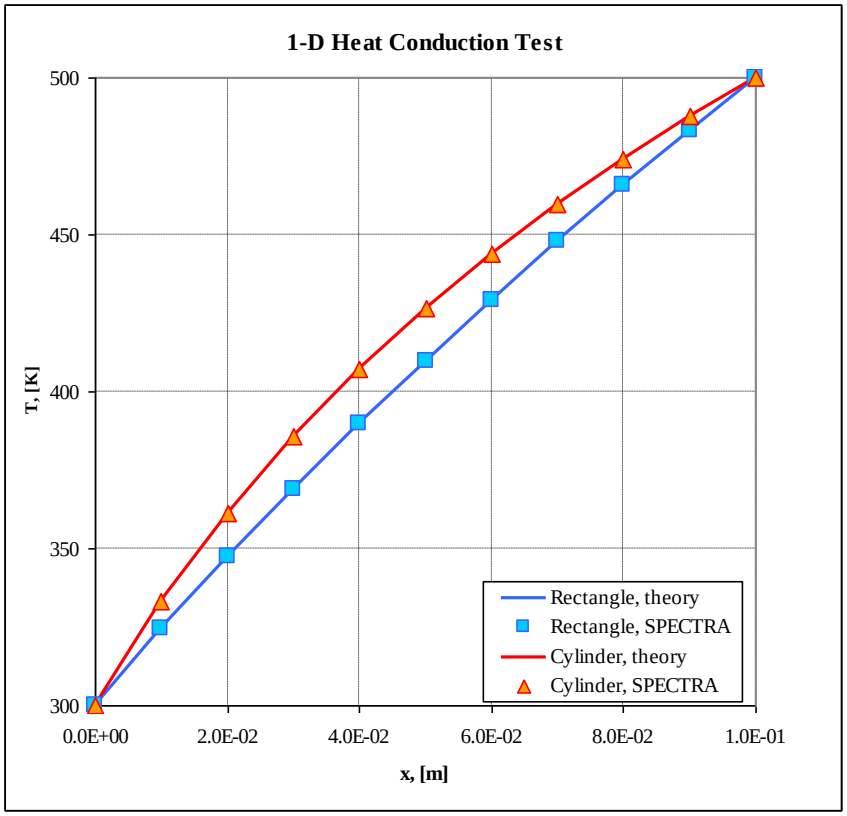


Figure 5-2 Variable thermal conductivity test, SPECTRA and analytical solution.

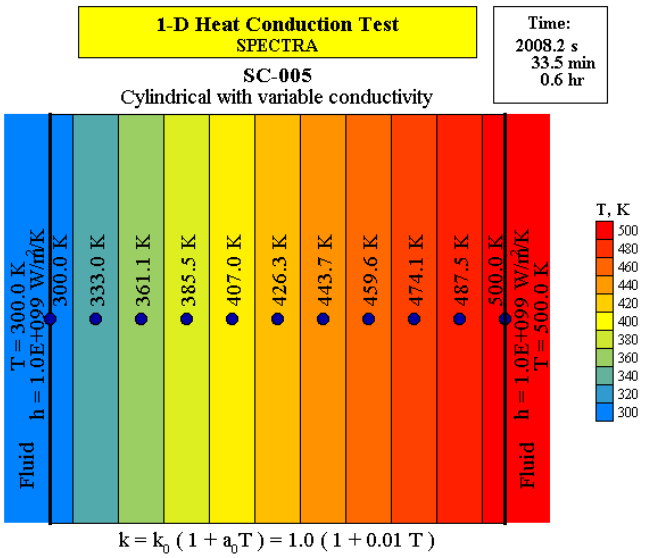


Figure 5-3 Variable thermal conductivity test, SPECTRA, t = 2000 s.

- *Example Problem 2: Transient Conduction in a Semi-Infinite Slab*

  Transient conduction in a semi-infinite slab with a step change of surface temperature is considered. The initial temperature of the slab is 300 K. At time equal to zero the surface temperature is set to 400 K. The theoretical solution of heat conduction equation result in the following temperature distribution inside the slab (see [20], chapter 3, equation 55):

$$T(x,t) = T_0 + (T_w - T_0) \cdot erfc\left(\frac{x}{2\sqrt{at}}\right)$$

  $T(x,t)$ temperature at location x from the left boundary, at time t, (K)
  T_0 initial temperature, (=300 K)
  T_w surface temperature, (=400 K)
  a thermal diffusivity, $k/(\rho c_p)$, (=5×10^{-7} m^2/s)

  Calculations were performed using a solid heat conductor with 0.01 m thickness. As shown below, this thickness is sufficient to represent a semi-infinite slab for the present, short term calculations. The heat conductor was nodalized using 21 nodes with equal distance between the nodes, that means cell thickness of 5×10^{-4} m for internal nodes and 2.5×10^{-4} m for the boundary nodes- Figure 5-4. The right surface is adiabatic, while at the left surface the desired temperature step is obtained by suitable tabular functions. The heat transfer coefficient at the left surface was again set to a large value (10^{99}) to minimize the convection resistance and make surface temperature equal to fluid temperature.

Figure 5-4 through Figure 5-7 show the temperatures at times 1.0, 5.0, 10.0, and 20.0 s. Comparison of the calculated results with the analytical solution is shown in Figure 5-8. The calculated values are in good agreement with the theoretical values, except near the right boundary, at t = 20 s. At that time temperature has penetrated into the right side of the conductor and the analytical solution, obtained for a semi-infinite slab becomes a bad approximation of the real geometry in the region close to the right boundary.

The above examples show that an accurate solution is obtained provided that sufficiently fine nodalization is selected. The mesh cell sizes are defined by the user, but the code internally checks whether the nodalization is fine enough, using the thermal penetration depth criterion:

$$d_{pen}(t) = C \cdot \sqrt{at}$$

Here a is material thermal diffusivity, $k/(\rho c_p)$, (m^2/s), Δt is time step (s), and C is a constant. The value of this constant depends on how exactly the penetration depth is defined. If the temperature gradient at the surface is used, then $C = \pi^{1/2} = 1.77$ ([21], section 3.4.2, p. 163). If the full penetration distance is used, then the constant is 4 times larger: $C = 4\pi^{1/2} = 7.09$. The latter value is used in SPECTRA. For each SC, the boundary cell size is compared to the thermal penetration depth obtained for 100.0 s time:

$$d_{pen}(t = 10^2) = 10 \cdot C \cdot \sqrt{a} = 70.9 \cdot \sqrt{a}$$

If the cell size exceeds this limit a warning message is printed. If the cell size exceeds the the thermal penetration depth obtained for 10^4 s:

$$d_{pen}(t = 10^4) = 100 \cdot C \cdot \sqrt{a} = 709 \cdot \sqrt{a}$$

then an error message is printed and the run stops.

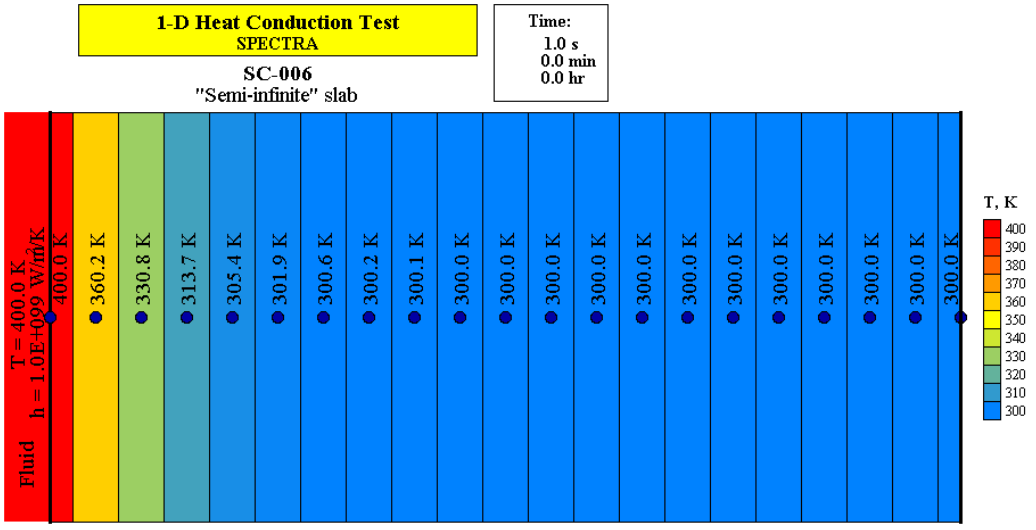


Figure 5-4 Transient conduction in a “semi-infinite” slab, SPECTRA, t = 1 s.

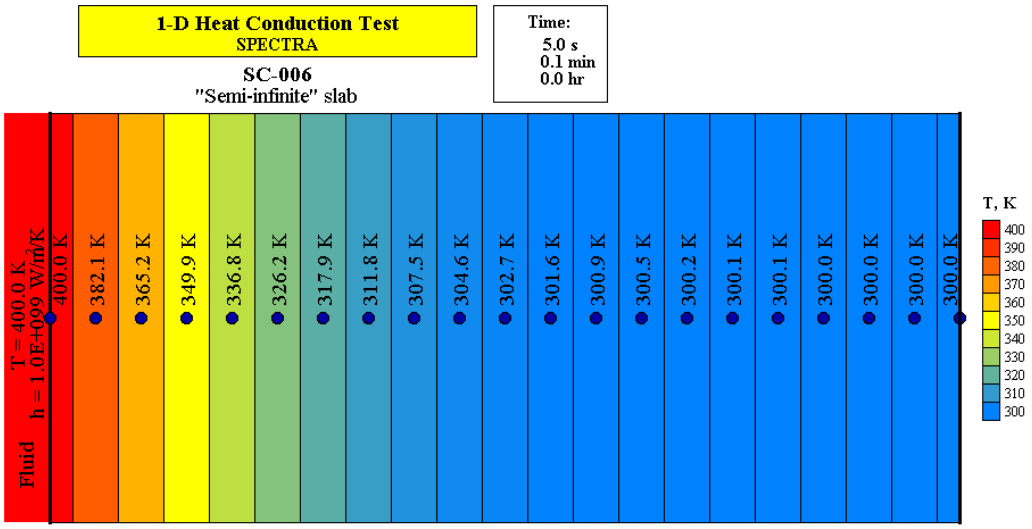


Figure 5-5 Transient conduction in a “semi-infinite” slab, SPECTRA, t = 5 s.

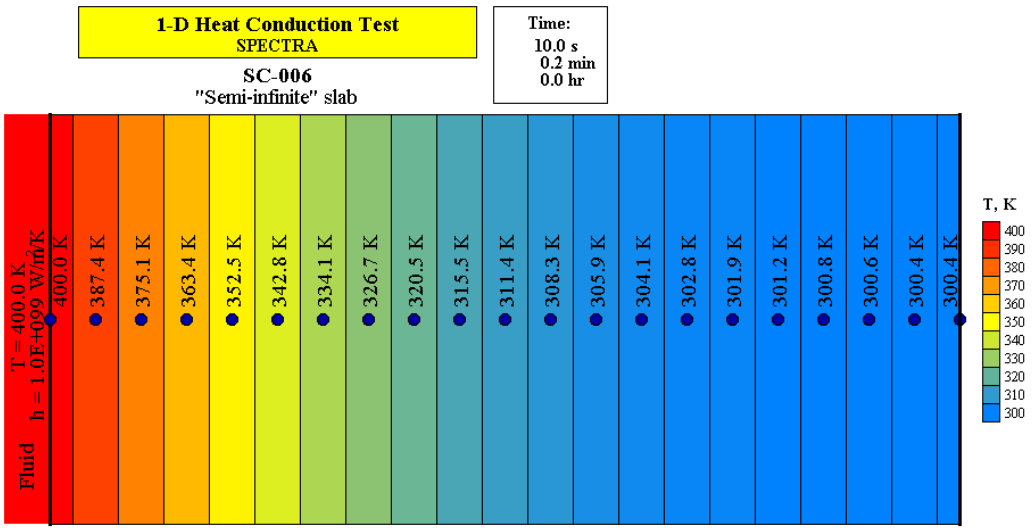


Figure 5-6 Transient conduction in a “semi-infinite” slab, SPECTRA, t = 10 s.

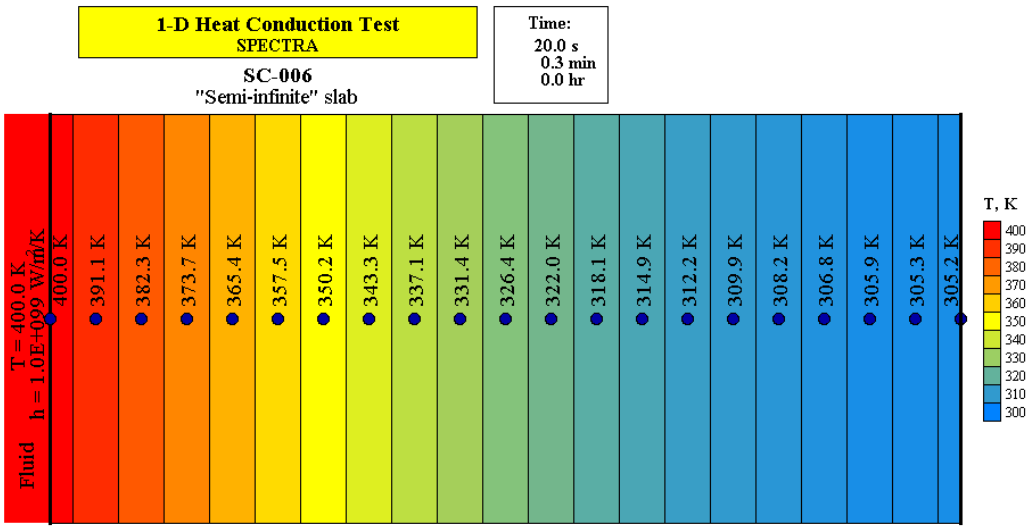


Figure 5-7 Transient conduction in a “semi-infinite” slab, SPECTRA, t = 20 s

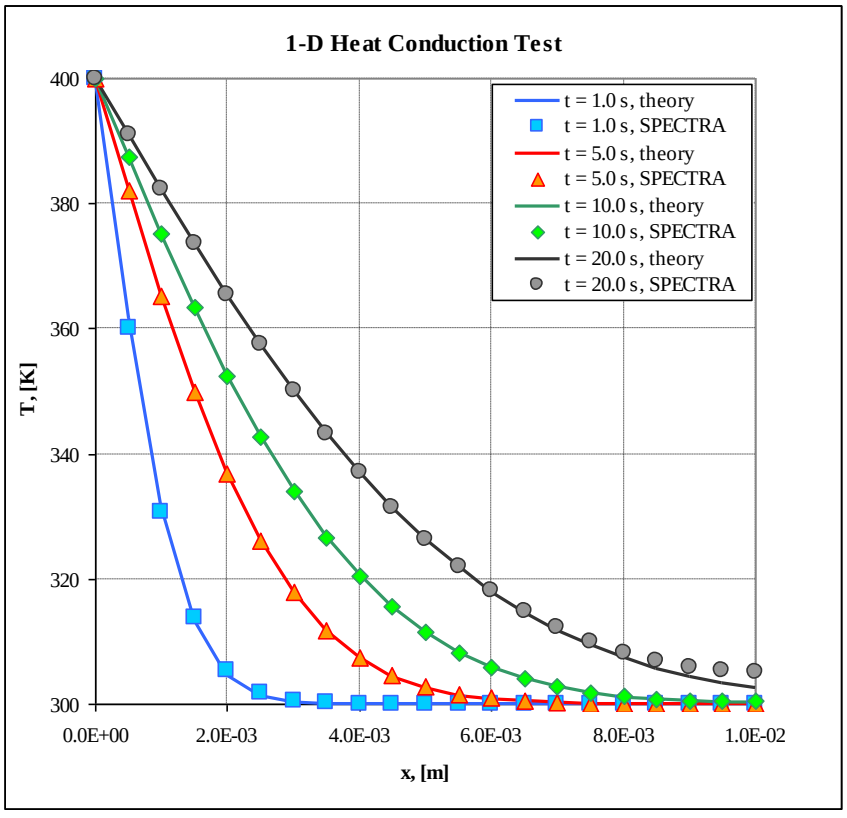


Figure 5-8 Transient conduction in a semi-infinite slab, SPECTRA and analytical solution.

5.3 Transient Heat Conduction (Method 2)

For Method 2, the nodes are placed at cell edges rather than cell centers. Nodalization is shown in Figure 5-9. This method is applied in most other system codes, for example MELCOR, RELAP.

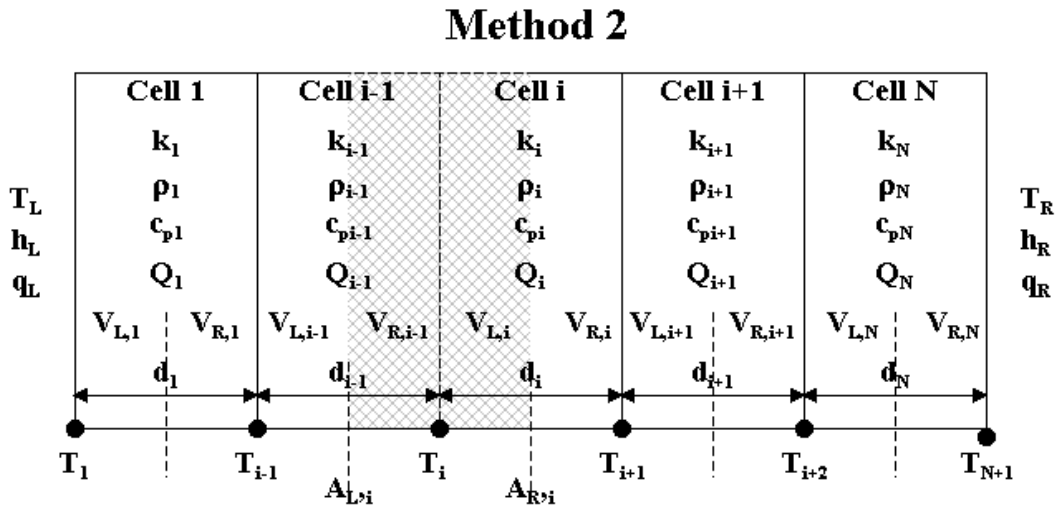


Figure 5-9 Nodalization of a 1-D Solid Heat Conductor, Method 2

For such case, the computational cell, for which the energy conservation is written, consists of half of the cell to the left of the temperature node *i*, and half of the cell to the right of the temperature node *i* (shaded area in Figure 5-9). The heat accumulation term is given by

$$(V_{R,i-1}\rho_{i-1}c_{p,i-1} + V_{L,i}\rho_i c_{p,i}) \cdot \frac{T_i - \overline{T_i}}{\Delta t}$$

Here $V_{L,i}$ and $V_{R,i}$ are the volumes of the left and the right "half-cells" of the cell *i*. For rectangular geometry the values are: $V_{L,i} = V_{R,i} = V_i / 2$, where V_i is the volume of cell *i* (m^3). For cylindrical and spherical geometry the values of $V_{L,i}$ and $V_{R,i}$ are not equal. The density and specific heat for each half cell is calculated using average temperature in each half-cell. For example, the density is defined as:

$\rho_{i-1} = \rho\ (\ ¾ \cdot T_i + ¼ \cdot T_{i-1}\ )$
$\rho_i \quad = \rho\ (\ ¾ \cdot T_i + ¼ \cdot T_{i+1}\ )$

The heat conduction on the path between the temperature nodes *i* and *i*+1 is:

$$q = -k \cdot \left(\frac{\partial T}{\partial x}\right)_{i,i+1} = \frac{T_{i+1} - T_i}{\frac{d_i}{k_i}}$$

The thermal conductivity k_i is calculated for the average temperature between the nodes *i* and *i* + 1. , which means: $k_i = f\ (\ ½ \cdot T_i + ½ \cdot T_{i+1}\ )$. Similarly, the thermal conductivity k_{i-1} is calculated for the average temperature between the nodes *i* and *i* – 1.

Therefore, the thermal conductivity is defined as:

$k_{i-1} = k\,(\,½\cdot T_i + ½\cdot T_{i-1}\,)$
$k_i \;\; = k\,(\,½\cdot T_i + ½\cdot T_{i+1}\,)$

The energy balance equations used for the temperature nodes have in the Method 2 the following forms:

- Interior nodes, $1<i<N+1$ (shaded area in Figure 5-9) :

$$(V_{R,i-1}\rho_{i-1}c_{p,i-1} + V_{L,i}\rho_i c_{p,i})\cdot\frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}}(T_{i-1} - T_i) + \frac{A_{R,i}}{R_{R,i}}(T_{i+1} - T_i) + \left(f_{R,i-1}Q_{i-1} + f_{L,i}Q_i\right)$$

  with:

$$R_{L,i} = \frac{d_{i-1}}{k_{i-1}} \qquad R_{R,i} = \frac{d_i}{k_i}$$

- Left boundary, $i=1$:

$$(V_{L,i}\rho_i c_{p,i})\cdot\frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{R,i}}{R_{R,i}}(T_{i+1} - T_i) + \left(f_{L,i}Q_i\right) + \frac{A_{L,i}}{R_{L,i}}\left(T_{fluid,L} - T_i\right) + A_{L,i}\cdot q_{r,L}$$

  with:

$$R_{L,i} = \frac{1}{h_L} \qquad R_{R,i} = \frac{d_i}{k_i}$$

- Right boundary, $i=N+1$:

$$(V_{r,i-1}\rho_{i-1}c_{p,i-1})\cdot\frac{T_i - \overline{T_i}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}}(T_{i-1} - T_i) + \left(f_{R,i}Q_i\right) + \frac{A_{R,i}}{R_{R,i}}\left(T_{fluid,R} - T_i\right) + A_{R,i}\cdot q_{r,R}$$

  with:

$$R_{L,i} = \frac{d_{i-1}}{k_{i-1}} \qquad R_{R,i} = \frac{1}{h_R}$$

$A_{L,i}$ heat transfer area at the left boundary of the computational cell i, ("half-cell i–1 and "half-cell" i - Figure 5-9), (m^2).
$A_{R,i}$ heat transfer area at the right boundary of the computational cell i, (m^2)
$f_{L,i}$ volume fraction of the left "half-cell" i, equal to $f_{L,i} = V_{L,i} / V_i$
$f_{R,i}$ volume fraction of the right "half-cell" i, equal to $f_{L,i} = V_{R,i} / V_i$
h heat transfer coefficient (W/m^2-K) on the left (L) or right (R) surface
q_r non-convective heat flux (W/m^2) on the left (L) or right (R) surface (obtained from thermal radiation model or from a user-defined Tabular/Control Function).

A transient heat conduction test, the same as the one shown in the previous section for the method 1, is shown in Figure 5-10, Figure 5-11 and Figure 5-12. More tests and comparisons between the Method 1 and Method 2 results are shown in Volume 3.

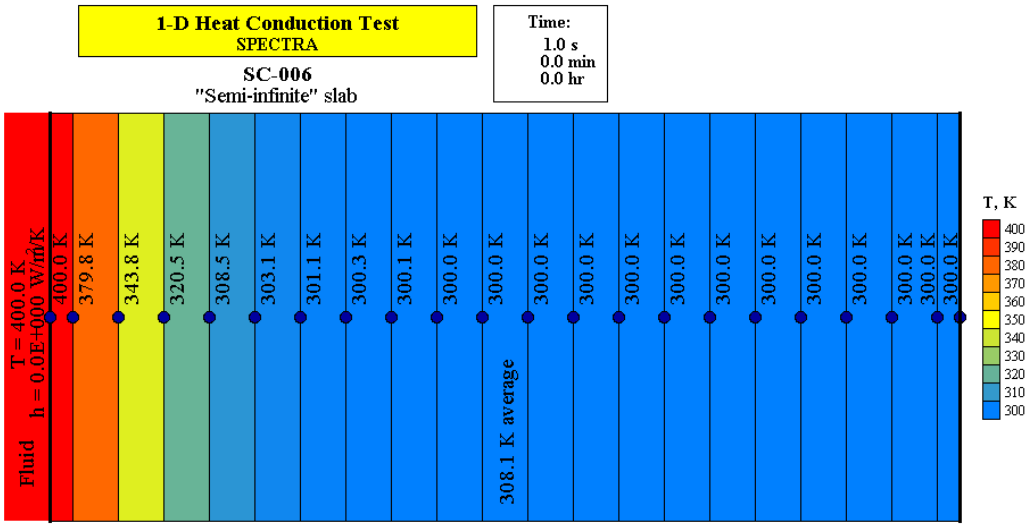


Figure 5-10 Transient conduction in a “semi-infinite” slab, Method 2, t = 1 s.

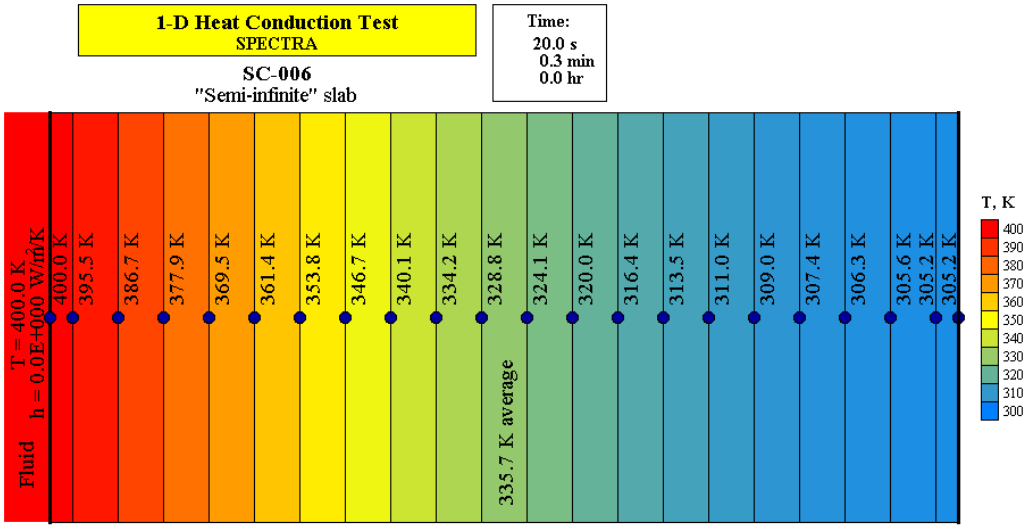


Figure 5-11 Transient conduction in a “semi-infinite” slab, Method 2, t = 20 s

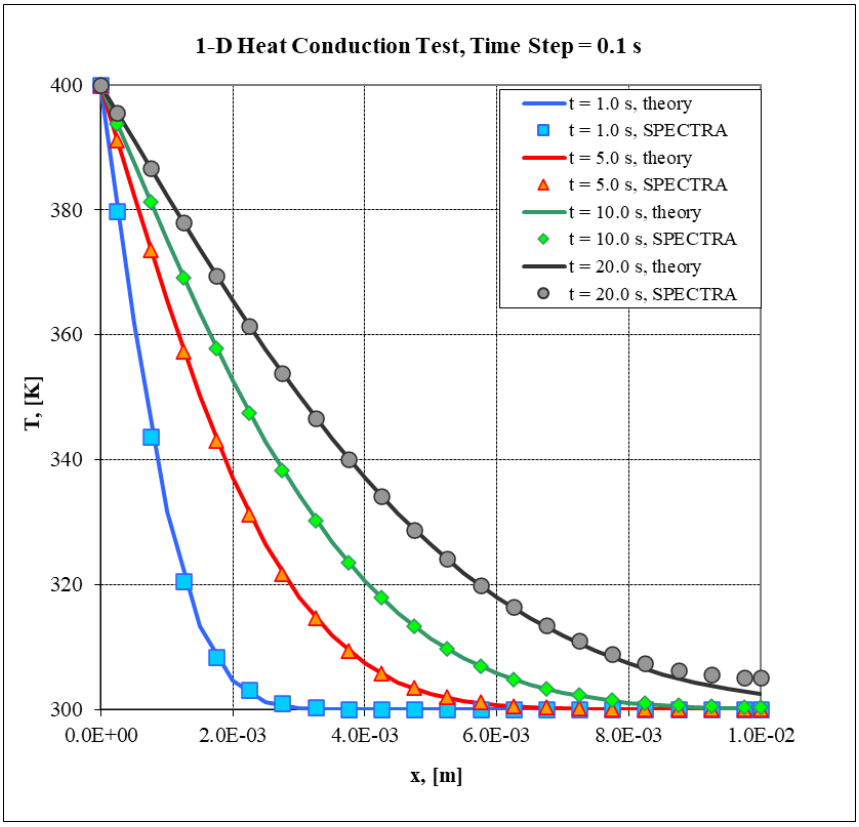


Figure 5-12 Transient conduction in a semi-infinite slab, Method 2 and analytical solution.

5.4 Boundary Conditions

Each 1-D Solid Heat Conductor has two boundary surfaces. One of the following types of boundary conditions must be used for each surface:

- Perfect insulation (no heat transfer at the boundary surface)
- Fluid temperature and heat transfer coefficient defined by Tabular Functions
- Fluid temperature taken from Control Volume, heat transfer coefficient defined by a user-defined Tabular Function
- Fluid temperature taken from Control Volume, heat transfer coefficient calculated by the Heat and Mass Transfer Package from fluid conditions in the boundary Control Volume

Typically the user wants to connect the SC surfaces to Control Volumes and let the code compute the heat transfer at the surfaces. In such case the user must make sure that the bottom and top elevation of the SC surfaces lie within the Control Volumes that are linked to those surfaces. The program checks the elevations and gives an error message if the SC elevations are not within the range defined by the boundary Control Volume elevations.

Accurate definition of the SC elevations is specifically important in cases when liquid is present in the boundary Control Volume. The fluid temperature and heat transfer coefficient above and below the water level are of course different and the program must determine appropriate representative boundary conditions. The method used to calculate the representative boundary conditions for the surfaces partly submerged in water is discussed in section 5.5. A necessary input for this method are the bottom and top elevations of the SC surfaces, which must be defined with good accuracy. In cases when a single-phase heat transfer is always expected to occur (for example gas cooled reactors) the exact SC elevations are important only in the case when stratification models are used within the boundary CV.

For user convenience an automatic SC allocation scheme has been implemented in the program. The scheme is activated by setting the SC mid-point elevation to –999. In such case the SC elevation, $Z(SC)$, is automatically set by the code based on the elevations of the boundary volumes as follows:

$$Z(SC) = (1/2) \cdot (Z_{MAX} + Z_{MIN})$$

where:

$$Z_{MAX} = Min\,[\, Z_{top}(CV_{left}),\, Z_{top}(CV_{right})\,]$$

$$Z_{MIN} = Max\,[\, Z_{bot}(CV_{left}),\, Z_{bot}(CV_{right})\,]$$

Here Z_{top} and Z_{bot} ate the top and the bottom elevations of the boundary volumes respectively. CV_{left} and CV_{right} are the left and the right boundary volume. Note that if only one surface is linked to a CV, then the automatic allocation will result in the SC mid-point being in the center of the boundary CV. For example, if only left surface is linked to a CV, then:

$$Z(SC) = (1/2) \cdot (\, Z_{top}(CV_{left}) + Z_{bot}(CV_{left})\,)$$

The automatic SC allocation scheme is illustrated in Figure 5-13. Four cases are shown:

- In the first case a vertical SC-100 is allocated in such a way that the surfaces span all common elevations of the left and right volumes, CV-110 and CV-120.
- In the second case a horizontal rectangular SC-200 is allocated in such a way that the left surface is in CV-210 and the right surface is in CV-220. The gap between CV-210 and CV-220 must not be larger than the thickness of SC-200.
- In the third case a vertical SC-300 cannot be allocated because CV-310 and CV-320 do not have common elevations. The mid-point elevation will be computed according to the equation shown above, but input error will occur.
- In the fourth case a horizontal rectangular SC-400 cannot be allocated because the gap between CV-410 and CV-420 is larger than the SC-400 thickness. Input error will occur.

When the option of automatic SC allocation is used, it is a good practice to check in the output file what are the SC-elevations.

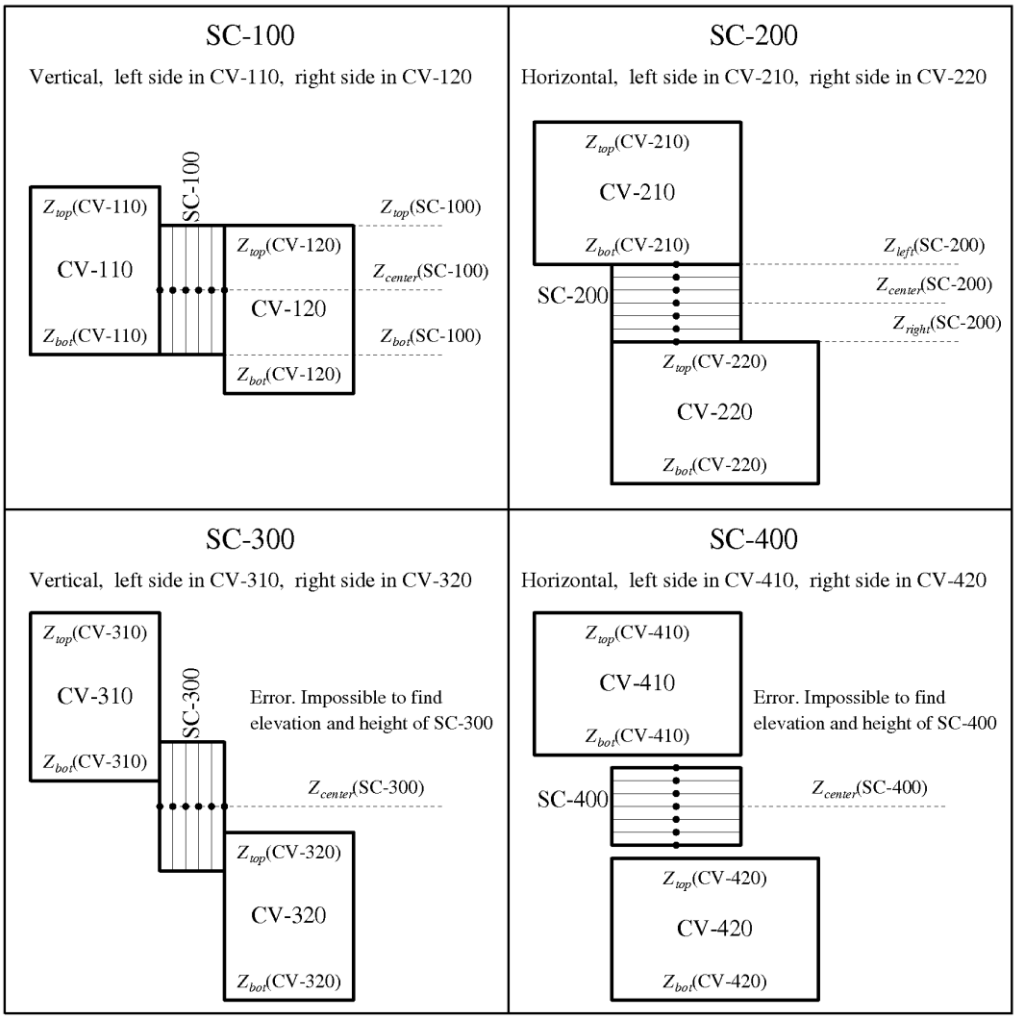


Figure 5-13 Automatic allocation of SC within the boundary Control Volumes

5.5 Representative Boundary Conditions

The Solid Heat Conductor Package requires one temperature and one heat transfer coefficient on for each Solid Conductor (SC) surface. If SC surface is partly immersed in water, then different heat transfer coefficients as well as fluid temperatures are calculated for the pool and the atmosphere. Nevertheless, for the conduction equation representative values of heat transfer coefficient, h, and fluid temperature, T_{fluid}, must be specified. The non-convective heat flux, q_r, is applied only for the atmospheric part of SC surface and therefore do not need averaging. The procedure used to calculate representative boundary conditions for all possible conditions is described below.

- Surface in atmosphere of a Control Volume. The representative boundary conditions are those for the atmosphere:

$$h = h_{atms}$$
$$T_{fluid} = T_{atms}$$

- Surface in pool of a Control Volume. The representative boundary conditions are those for the pool:

$$h = h_{pool}$$
$$T_{fluid} = T_{pool}$$

- Surface partly submerged in pool of Control Volume. The representative heat transfer coefficient is assumed to be equal to the average of the heat transfer coefficients for the atmosphere and the pool.

$$h = X_{atms} h_{atms} + X_{pool} h_{pool}$$

X_{atms} is the fraction of SC surface in the atmosphere of Control Volume, and X_{pool} is the fraction of surface in pool of a Control Volume.

The representative fluid temperature is calculated from the following relation:

$$T_{fluid} = \begin{cases} T_{SC} - \dfrac{q_{atms} + q_{pool}}{h} & if \;\; h > 0.0 \\ T_{SC} & if \;\; h = 0.0 \end{cases}$$

where T_{SC} is the SC surface temperature, while q_{atms} and q_{pool} are the heat fluxes to atmosphere and pool respectively, and are equal to:

$$q_{atms} = X_{atms} \cdot h_{atms} \cdot \left(T_{SC} - T_{atms}\right)$$
$$q_{pool} = X_{pool} \cdot h_{pool} \cdot \left(T_{SC} - T_{pool}\right)$$

As a consequence of the averaging procedure described above, the SC surface "sees" only the representative values of heat transfer coefficient and fluid temperature. The overall wall heat flux is equal to:

$$q = q_{atms} + q_{pool}$$

This flux is partitioned between the atmosphere (q_{atms}) and the pool (q_{pool}) of Control Volume using the above equations. Note that the individual atmosphere and pool heat fluxes may have different signs. This happens for example, when SC surface is in contact with cold pool and hot atmosphere, with intermediate surface temperature.

An example of the calculation of representative fluid temperature is shown in Figure 5-14. Only if both pool and atmosphere heat transfer coefficients are equal, then the representative fluid temperature changes linearly from the atmosphere temperature (when the pool fraction is zero) to the pool temperature (when the pool fraction is one). In other cases the change is nonlinear.

If, for example, pool heat transfer coefficient is larger than the atmosphere heat transfer coefficient, then the representative fluid temperature is closer to the pool temperature (the lowest line, marked with triangles, in Figure 5-14).

The representative boundary conditions are available for plotting. For any SC, the user may plot:

- atmosphere values (T_{atms}, h_{atms}, q_{atms})
- pool values (T_{pool}, h_{pool}, q_{pool})
- representative values (T_{fluid}, h, q)

The representative values are recommended for general use. In the case of 2-D Solid Heat Conductors only the representative values are available for plotting.

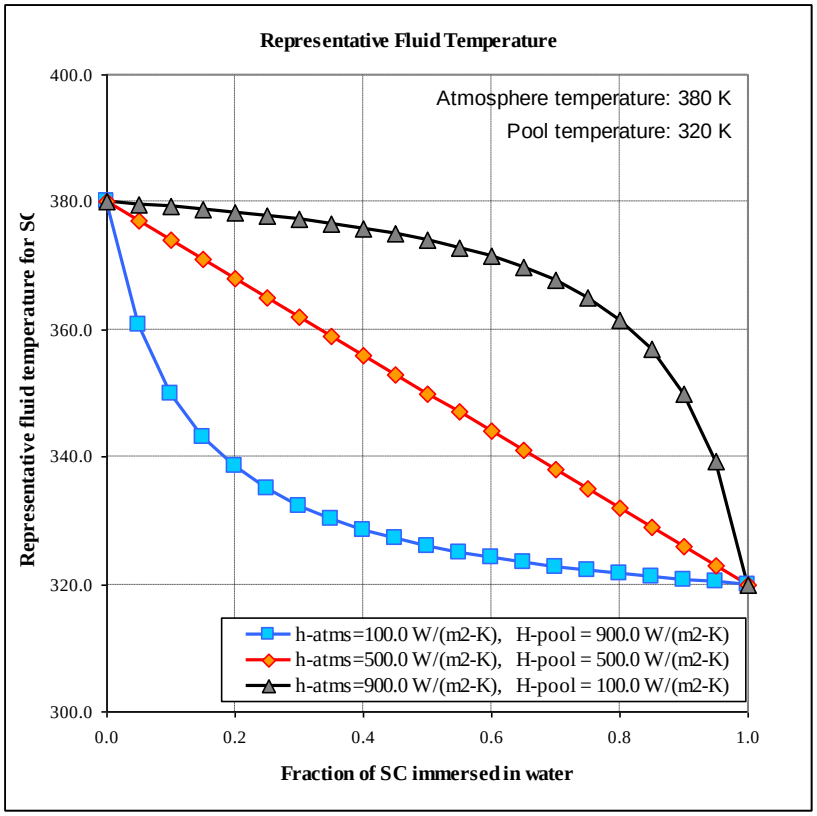


Figure 5-14 Representative fluid temperature calculation

In case of boiling surfaces a special treatment is provided to take into account the warm plume (section 2.6.7). In this case the fluid temperature is calculated by:

$$T_{fluid} = T_{pool} + (T_{\text{int}} - T_{pool}) \cdot f_{\alpha}(\alpha) \cdot f_{T}(T) \cdot f_{Z}(Z)$$

T_{int} is the liquid temperature at the pool-bubble interphase (thus the warm plume temperature, equal to the saturation temperature for the pressure at the elevation of average bubble - see section 2.6.7), and f_{α}, f_T, f_Z, are the void fraction factor, the temperature factor, and the submergence factor, described below.

- Void fraction factor, $f_{\alpha}(\alpha)$

$$f_{\alpha}(\alpha) = \begin{cases} 0.0 & if \quad \alpha \le \alpha_1 \\ (3-2\lambda)\lambda^2 & if \quad \alpha_1 < \alpha < \alpha_2 \\ 1.0 & if \quad \alpha_2 \le \alpha \end{cases}$$

  α is the average void fraction in the pool; λ is the interpolation factor, equal to:

$$\lambda = \frac{\alpha - \alpha_1}{\alpha_2 - \alpha_1}$$

  The limiting void fractions have default values of: $\alpha_1=10^{-3}$, $\alpha_2=0.05$, and may be redefined via input. Physically the use of the void fraction factor means that a SC surface will be exposed to warm plume only if the average void fraction of bubbles in the pool is sufficiently large.

- Surface temperature factor, $f_T(T)$

$$f_{T}(T) = \begin{cases} 0.0 & if \quad T_{SC} \le T_{int} + \Delta T_1 \\ (3-2\lambda)\lambda^2 & f \quad T_{int} + \Delta T_1 < T_{SC} < T_{int} + \Delta T_2 \\ 1.0 & f \quad T_{int} + \Delta T_2 \le T_{SC} \end{cases}$$

  T_{SC} is the surface temperature of the SC; λ is the interpolation factor, equal to:

$$\lambda = \frac{T_{SC} - (T_{int} + \Delta T_1)}{(T_{int} + \Delta T_2) - (T_{int} + \Delta T_1)}$$

  The limiting temperature differences have default values of: ΔT_1=1.0 K, ΔT_2=3.0 K, and may be redefined via input. Physically the use of the surface temperature factor means that an SC surface will be exposed to the warm plume only if boiling occurs at this surface, which means if the surface temperature is sufficiently high.

- Surface submergence factor, $f_Z(Z)$

$$f_Z(Z) = \begin{cases} (3-2\lambda_1)\lambda_1^2 & if \quad 0.0 \le Z < Z_{bubb} \\ 0.0 & if \quad Z_{bubb} \le Z \le Z_{bubb} \cdot Z_{m1} \\ (3-2\lambda_2)\lambda_2^2 & if \quad Z_{bubb} \cdot Z_{m1} < Z < Z_{bubb} \cdot Z_{m2} \\ 1.0 & if \quad Z_{bubb} \cdot Z_{m2} \le Z \end{cases}$$

Z_{bubb} is the submergence of average bubble; λ_1 and λ_2 are the interpolation factors, equal to:

$$\lambda_1 = \frac{Z}{Z_{bubb}}$$

$$\lambda_2 = \frac{Z - Z_{bubb}Z_{m2}}{Z_{bubb}Z_{m2} - Z_{bubb}Z_{m1}}$$

The limiting submergence multipliers have default values of: $Z_{m,1}$ = 2.0, $Z_{m,2}$ = 4.0, and may be redefined via input. Physically the use of the surface submergence factor means that an SC surface will be exposed to the warm plume if the surface is not located very deeply in the pool. The warm plume temperature is approaching pool temperature when surface submergence approaches zero.

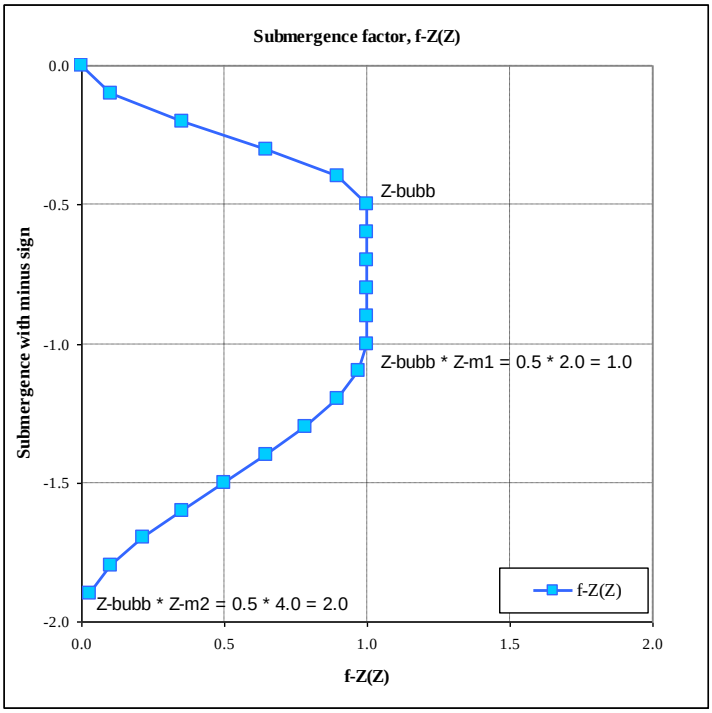


Figure 5-15 Submergence factor, $f_Z(Z)$.

5.6 Temperature Averaging for Heat Exchangers

The temperature averaging concept is best explained using a counter-current heat exchanger, shown in Figure 5-16. The discussion below is relevant for modeling heat exchangers not only with the SPECTRA code, but also with most computer codes, for example MELCOR, CONTAIN etc.

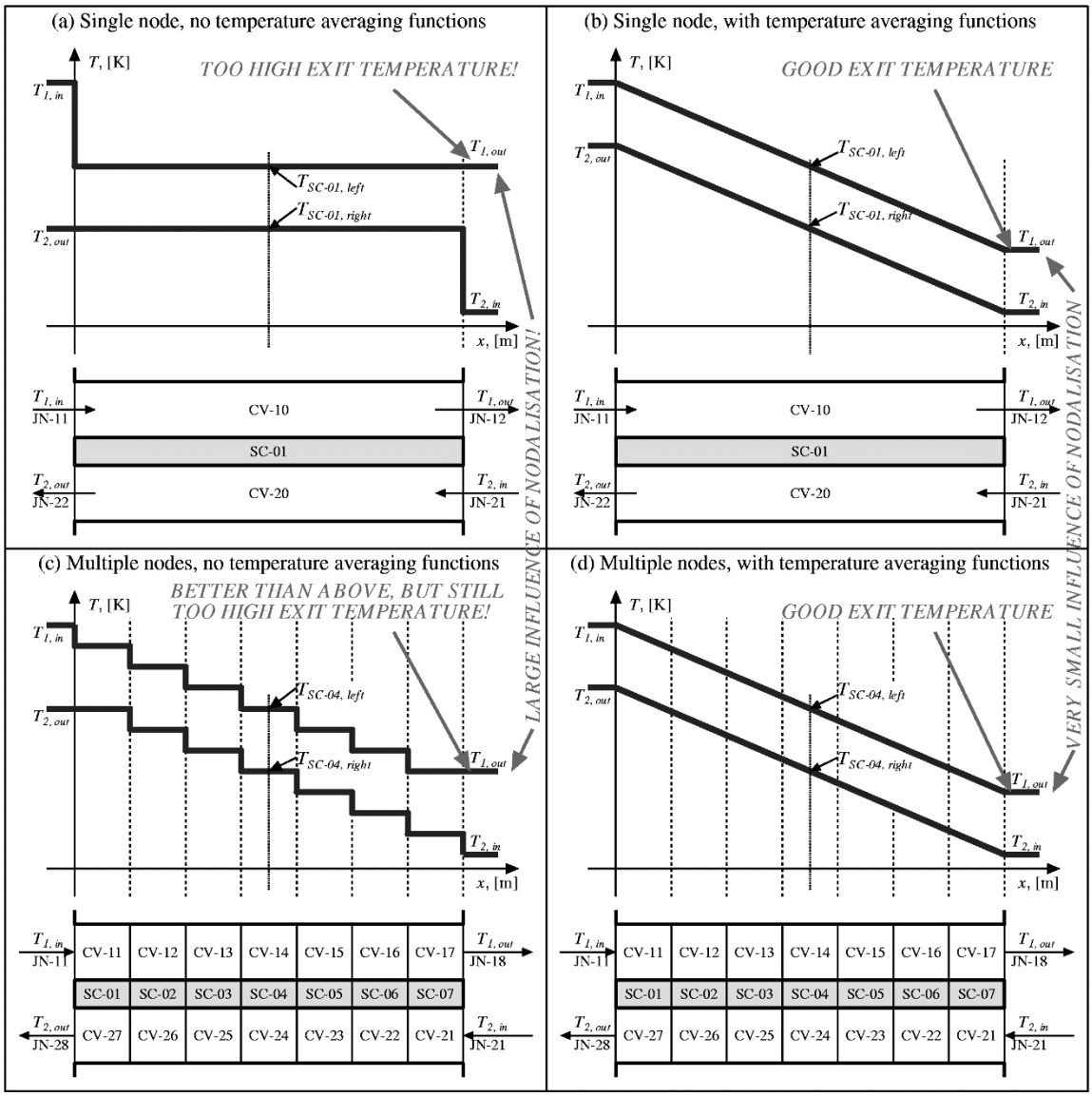


Figure 5-16 Counter flow heat exchanger modelling.

Figure 5-16 (a) shows a heat exchanger modeled using a single "node". By the "node" one should understand here a heat conducting structure and two control volumes, representing the cold and the hot side of the heat exchanger. The temperatures in both control volumes will stabilize at a certain level, denoted in Figure 5-16 (a) by $T_{SC\text{-}01,left}$ and $T_{SC\text{-}01,right}$. The temperature difference depends on the geometry and the hydraulic conditions of the analyzed system. Even if the computer code is able to predict this temperature difference perfectly well, the exit temperatures will be quite different from reality. This is because the wall, in this case SC-01, "sees" on both sides only a single temperature on its whole length Figure 5-16 (a). In reality the temperature changes along the wall, on both hot and cold side.

In order to obtain more accurate exit temperature prediction one should use multiple nodes to model the heat exchanger. An example of such model is shown in Figure 5-16 (c), where 7 nodes are used. This representation allows improving the exit temperature prediction, but the error still remains significant. A nodalization sensitivity study, performed for the IHX (intermediate heat exchanger) of the HTR-ICC plant (High Temperature Reactor - Indirect Cycle Co-generation) showed that for this type of heat exchanger the error in the exit temperature prediction was:

- ~80 K for 5 nodes,
- ~40 K for 10 nodes,
- ~20 K for 20 nodes.

These data are illustrated in Figure 5-17 (the line: "without temperature averaging functions"). An extrapolation of the above numbers leads to an expected error of about 10 K for 40 nodes, and 5 K for 80 nodes. Thus, even if 100 nodes are applied the exit temperature will still be off by several degrees.

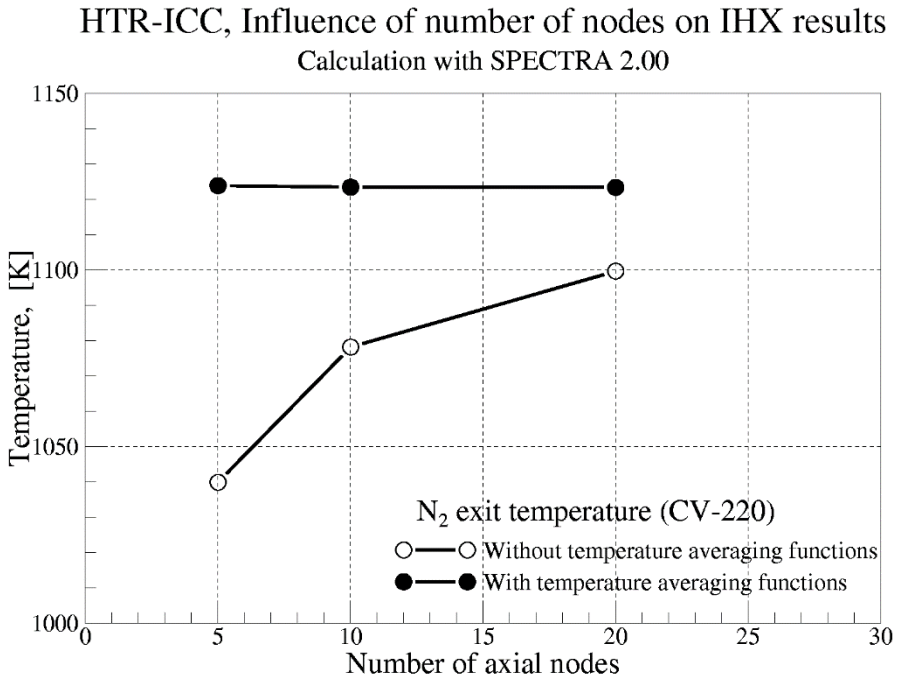


Figure 5-17 Influence of the number of nodes on HTR-ICC intermediate heat exchanger.

Therefore, for the analyzed heat exchanger one would need more than a hundred nodes to achieve accuracy similar to that obtained with the temperature averaging. One must remember that a node consists of a single structure, two control volumes, and two junctions to allow fluid flow. Thus the total number of components needed to represent only the heat exchanger is more than 500. A typical HTR system consists of at least several heat exchangers, not to mention turbo-compressors, and the reactor itself.

Application of this large amount of nodes would inevitably lead to long computation times because of small time steps forced by the "Courant Limit". Consequently, such approach would be impractical for analyzing long transients.

In order to improve the accuracy without the necessity of using very large amount of nodes the concept of fluid temperature averaging at the structure boundary has been developed. The concept is quite simple. The structure boundary fluid temperature is defined as the average of the fluid temperature at the inlet to the boundary control volume and at the exit from it (note that the exit temperature is equal to the temperature in the boundary control volume itself), rather than the fluid temperature in the boundary control volume.

Use of the temperature averaging concept is shown in Figure 5-16 (b) and (d). The boundary fluid temperatures, which the conducting structure "sees", are equal to the average of the control volume and the inlet temperature. The consequences might be quite surprising in case of one-node representation - such as shown in Figure 5-16 (b). The exit temperature on the hot side (which is also the temperature in the boundary volume CV-10) is lower than the exit temperature on the cold side (CV-20)! Thus, although the fluid in CV-10 is colder than the fluid in CV-20, the heat is still transferred from CV-10 to CV-20. The results are correct even in one-node representation, nevertheless to avoid confusion in the results interpretation, at least few nodes should be used - see Figure 5-16 (d).

A general temperature-averaging scheme is available in SPECTRA. The scheme is allows correct temperature averaging in steady state, as well as transients with fluid flow direction changes. The method relies on a user-defined nominal fluid flow, W_{TA}, for full temperature averaging. This value of W_{TA} is defined as the minimum mass flow at which the temperature distribution within a single Control Volume of the heat exchanger can be considered as approximately linear. Therefore W_{TA} depends on the applied nodalization; with a small number of nodes applied; W_{TA} must be large, while with large number of nodes W_{TA} may be smaller. The "full" temperature averaging is performed only when the fluid flow exceeds W_{TA} :

$$T_{SC} = \frac{1}{2} T_{CV} + \frac{1}{2} T_{inlet}$$

T_{SC} is the averaged fluid temperature, used as a boundary condition for the Solid Heat Conductor SC, T_{CV} is the fluid temperature in the Control Volume CV, where the heat is transferred (boundary volume of the SC), and T_{inlet} is the fluid temperature at the inlet to the CV.

When the flow is smaller, the temperature is obtained from a weighted average, as:

$$T_{SC} = \left[1 - a(W)\right] \cdot T_{CV} + \left[a(W)\right] \cdot T_{inlet}$$

W is the inlet mass flow. The interpolation is performed using a smooth 3-rd order interpolation, as:

$$a(W) = \frac{1}{2}(3 - 2X)X^2 \qquad X = \frac{W}{W_{TA}}, \qquad 0.0 \le a(W) \le 0.5$$

With this definition the average temperature becomes identical with the Control Volume temperature if the inlet flow decreases to zero:

$$T_{SC} \to T_{CV}, \quad when\, W \to 0.0$$

This procedure of temperature averaging gives good results for the flow range between zero and a large flow. It does not however guarantee good results if the flow direction changes. When flow direction changes then a counter-current heat exchanger becomes a co-current one (or vice versa). Of course the behavior of the exchanger is quite different in that case. Essentially the difference is that the value of W_{TA}, appropriate for the normal (counter-current) flow conditions, is inappropriate for the new (co-current) flow conditions. This may be easy understood in case of a typical recuperator, where the inlet temperatures differ by few hundred degrees and the primary to secondary side temperature difference is of order of ten degrees. If the flow direction on one side changes, then the primary to secondary side temperature difference becomes several hundred degrees. In such circumstances there is a huge heat transfer near the inlet and the temperatures rapidly change. In order to have a linear temperature profile over the same control volume the mass flow needs to be much larger. Thus in the co-current situation the appropriate value of W_{TA} is typically much larger than in normal counter-current situation. This can be achieved in the model by specifying different value of W_{TA} for co- and counter-current flows. The input consists of four entries of W_{TA}, which are then used depending on the current flow directions. The concept is illustrated in Figure 5-18.

The input entries are:

- $W_{TA,\,1}$, limiting W_{TA}, for flow entering through JN-11 (normal flow direction).
- $W_{TA,\,2}$, limiting W_{TA}, for flow entering through JN-12 (reversed flow direction).
- $W_{TA,\,1\,sec}$, limiting W_{TA}, for flow entering through JN-11 and reversed secondary flow.
- $W_{TA,\,2\,sec}$, limiting W_{TA}, for flow entering through JN-12 and reversed secondary flow.

For a typical counter-current flow heat exchanger, reasonable values for the T-A model would be:

- $W_{TA,\,1}$ = Nominal mass flow (counter-current).
- $W_{TA,\,2}$ = Large value (co-current).
- $W_{TA,\,1\,sec}$ = Large value (co-current).
- $W_{TA,\,2\,sec}$ = Nominal mass flow (counter-current).

Results do not depend on the logical junction direction, but only on the current flow direction.

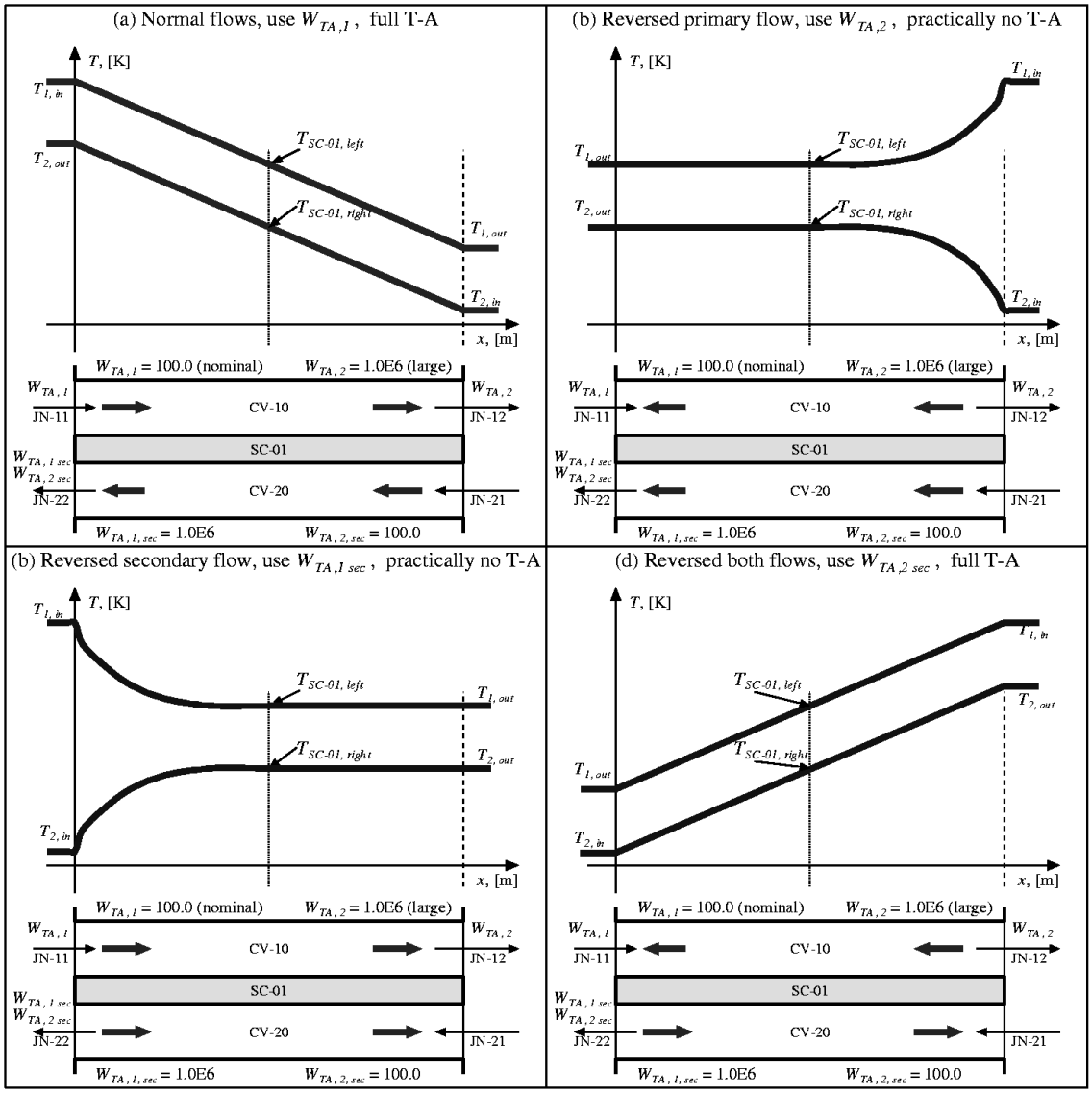


Figure 5-18 Generalized temperature-averaging scheme.

5.7 Simultaneous Heat Transfer to Pool and Atmosphere

Simultaneous heat transfer to pool and atmosphere may lead to too large condensation in some cases. This situation is illustrated in Figure 5-19. An artificial negative heat flux at the surface of SC (Figure 5-19, middle), keeps the steam temperature close to the liquid temperature and prevents heat up of the uncovered nodes in the upper part of the fuel channel.

To eliminate this problem (at least to a certain extent; the best strategy here would be to use fine nodalization - Figure 5-19, left), the user may restrict the simultaneous heat transfer to a situation when the pool-covered fraction is smaller than a user-defined critical value. An interpolation zone is defined. The full transfer to gas occurs when the pool fraction is below 0.9 of the critical value, a linear interpolation is performed in the range between 0.9 and 1.0 times the critical value. In the considered example a value of 0.01 was used (Figure 5-19, right).

The procedure illustrated above allows to eliminate heat transfer from atmosphere to pool, through the solid structure (SC or TC). However, a direct heat transfer still exists at the pool-atmosphere interphase. Therefore it is advisable to eliminate the pool-atmosphere heat transfer at the same time. This was done in the example shown below.

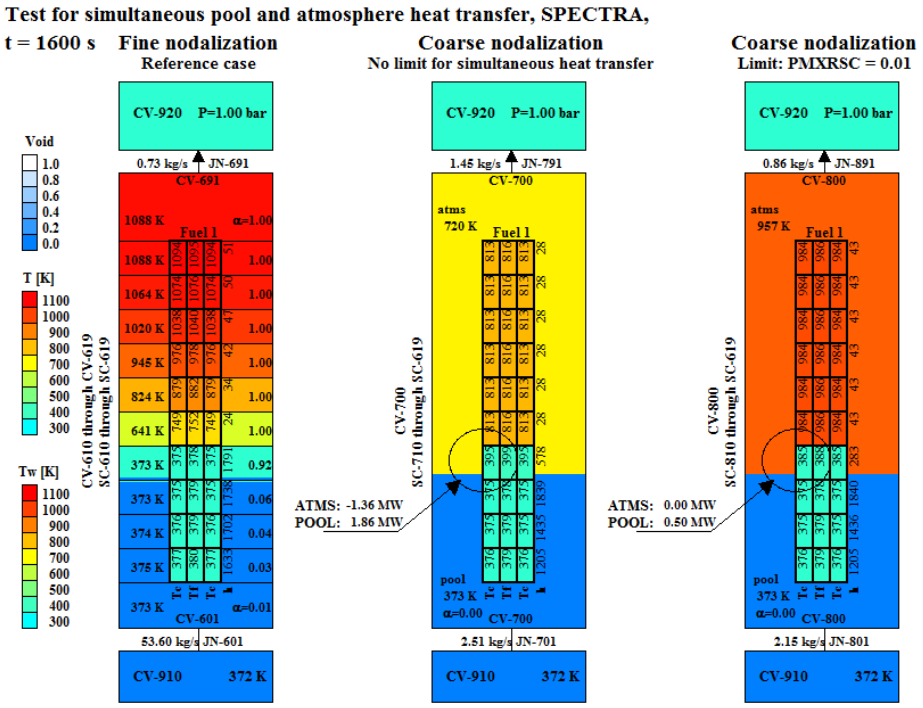


Figure 5-19 Simultaneous heat transfer to pool and atmosphere

5.8 Extended Surfaces – Fins and Spines

Fins or spines may be present on rectangular or cylindrical solid heat conductors. Modeling of fins and spines is performed in a simplified way. A detailed model would require solution of multi-dimensional conduction equation. The present model treats fins or spines just as an addition to the SC surface (extended surface), which changes the effective heat flux from the true SC surface. The wall nodalization is identical to that which would be applied in absence of fins/spines. The conduction equation is therefore the same; the only difference is that the conduction model now "sees" different values of boundary conditions.

The influence of extended surfaces is calculated based on the assumption that the temperature distribution inside the fins/spines may be represented by the theoretical stationary distribution. With this assumption the fin/spine efficiency is calculated, using the usual efficiency definition [20]. Finally, the enhancement of heat (and mass) transfer on the SC surface is obtained. The calculation procedure is shortly described below.

Three types of extended surfaces are considered (see Figure 5-20):

- Rectangular fins.
- Cylindrical fins.
- Spines.

Spines may be applied for any geometry. Rectangular fins may be applied on both surfaces of rectangular heat conductors. Cylindrical fins may be applied only on the right (outer) surface of cylindrical heat conductors.

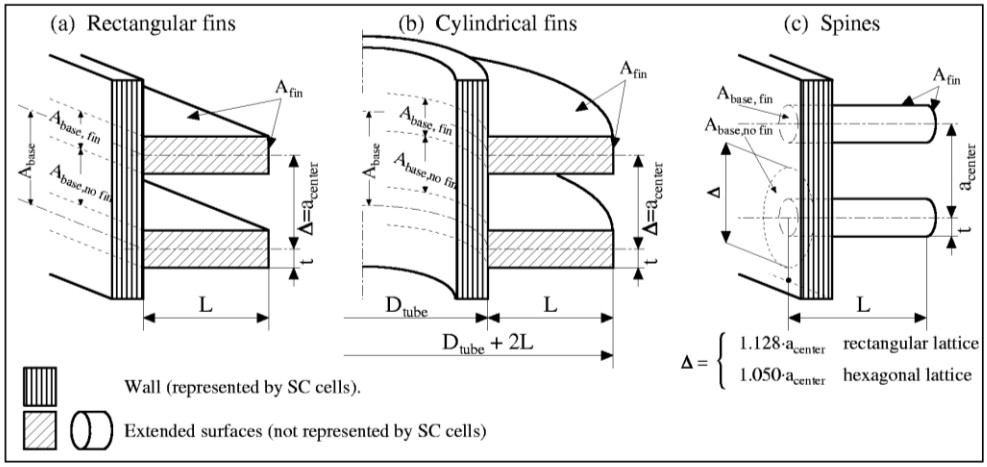


Figure 5-20 Extended surfaces for 1-D Solid Heat Conductors, fins and spines.

The effective heat transfer coefficient, h_{eff}, and mass transfer flux, m_{eff}, at the SC surface with fins or spines are calculated by multiplying the true values (obtained from the Heat and Mass Transfer Package) by the enhancement factor, E.

Different values of enhancement factors are used for the parts of SC exposed to the atmosphere and the pool of CV:

$$h_{eff}^{atms} = E^{atms} \cdot h^{atms} \qquad m_{c,eff} = E^{atms} \cdot m_c$$
$$h_{eff}^{pool} = E^{pool} \cdot h^{pool} \qquad m_{c,eff} = E^{pool} \cdot m_c$$

The symbols m_c and m_b mean the condensation and boiling mass transfer rates (kg/m^2-s), respectively.

The enhancement of heat/mass transfer due to fins or spines, E, is defined as the ratio of the average heat/mass transfer flux which is transferred through the outer surface of the wall (at the base of fins or spines), to that which would be transferred through the same surface in absence of fins (but with the same heat transfer coefficient). That means: if the fins were isothermal (temperature in all fin equal to the temperature at the base of the fin) the enhancement would be equal to the ratio of the total heat transfer area in presence of fins or spines, to that in absence of fins/spines. Since fins/spines are not isothermal, the heat transfer area of the fins/spines must be multiplied by the fin/spine efficiency, which is defined as the ratio of heat transferred over the total fin or spine area to that which would be transferred if the fins or spines were isothermal ([20], section 3.B.2.b). Thus the fin enhancement is given by:

$$E^{comp} = \frac{A_{base,no\ fin} + A_{fin} \cdot \eta^{comp}}{A_{base}}$$

$A_{base,\ no\ fin}$ this part of the surface area at the fin/spine base, which is not covered by the fin/spine, (m^2), (Figure 5-20)

A_{base} surface area at the fin/spine base, (m^2), (equal to: $A_{base,\ no\ fin} + A_{base,\ fin}$), where $A_{base,\ fin}$ is this part of the surface area at the fin/spine base which is covered by fin/spine, (m^2), (Figure 5-20)

A_{fin} area of heat/mass transfer of the fin/spine, (m^2), (Figure 5-20)

η^{comp} fin/spine efficiency for a given component. The fin/spine efficiency is defined as the ratio of the actual heat transfer to the heat that would be transferred if the fins/spines were isothermal (i.e. thermal conductivity $k \to \infty$) - see for example [20] page 3-115.

The subscript *comp* is used to signify atmosphere or pool. The final formula for E depends on the geometry and is described below, separately for each geometry type.

- Rectangular Fins:

  The surface areas per unit length are equal to (see Figure 5-21.a):

  $A_{base,\ no\ fin} = \Delta - 2t$
  $A_{base,\ fin} = 2t$
  $A_{fin} = 2(L+t)$

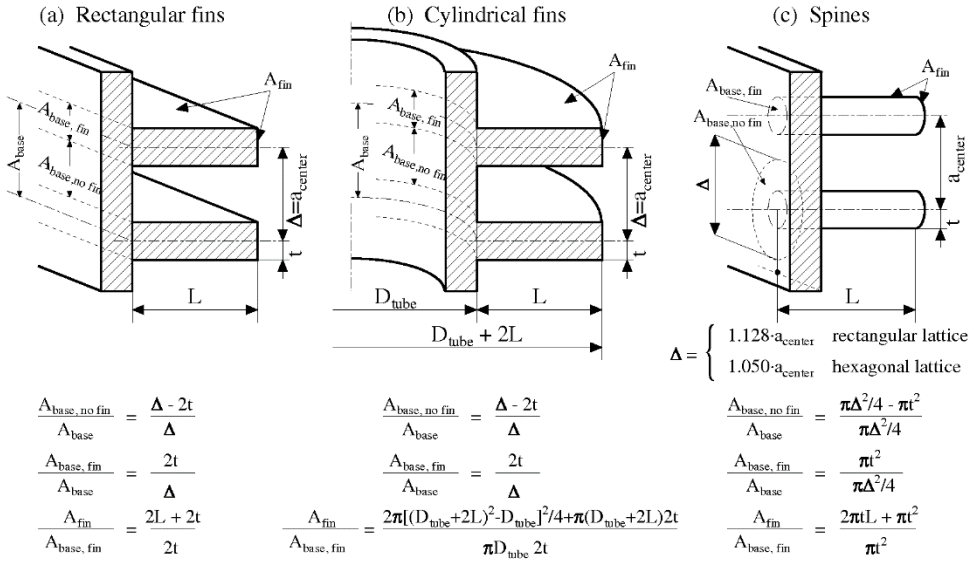


Figure 5-21 Calculation parameters for fins and spines.

Consequently, the fin enhancement is given by:

$$E^{comp} = \left(1 - \frac{2t}{\Delta}\right) + \left(\frac{2t}{\Delta}\right) \cdot \left(1 + \frac{L}{t}\right) \cdot \eta^{comp}$$

L is fin length, t is half thickness of a fin, and Δ is the distance between two neighboring fins (Figure 5-21.a). The fin efficiency for given component, η^{comp} is given by ([21], section 2.4):

$$\eta^{comp} = \frac{\tanh(\beta^{comp} \cdot L)}{\beta^{comp} \cdot L}$$

β^{comp} is defined as:

$$\beta^{comp} = \sqrt{h^{comp} / (k \cdot t)}$$

In the above formula h^{comp} is the heat transfer coefficient on the fin surface (calculated by the Heat and Mass Transfer Package), while k, t, are the thermal conductivity of the fin material, and half fin thickness, respectively.

- Cylindrical fins

The surface areas per unit length are equal to (see Figure 5-21.b):

$A_{base,\ no\ fin} = (\Delta - 2t) \cdot \pi \cdot D_{tube}$
$A_{base,\ fin} = 2t \cdot \pi \cdot D_{tube}$
$A_{fin} = 2 \cdot [\pi \cdot (D_{tube}+2L)^2/4 - \pi \cdot (D_{tube})^2/4] + 2t \cdot \pi \cdot (D_{tube}+2L)$

Consequently, the fin enhancement is given by:

$$E^{comp} = \left(1 - \frac{2t}{\Delta}\right) + \left(\frac{2t}{\Delta}\right) \cdot \left[1 + \frac{L}{t} \cdot \left(1 + \frac{2t + L}{D}\right)\right] \cdot \eta^{comp}$$

The equation is valid for the right (outer) surface. For the left (inner) surface. the formula is very similar, with $2t+L$ replaced by $2t–L$. The fin efficiency is given by ([21], section 2.4):

$$\eta^{comp} = \frac{c}{\beta^{comp} L} \cdot \frac{K_1(\beta^{comp} r_1) \cdot I_1(\beta^{comp} r_2) - I_1(\beta^{comp} r_1) \cdot K_1(\beta^{comp} r_2)}{K_0(\beta^{comp} r_1) \cdot I_1(\beta^{comp} r_2) + I_0(\beta^{comp} r_1) \cdot K_1(\beta^{comp} r_2)}$$

K, I are Bessel functions, β^{comp} is defined as above, $c = r_1/r_2$, and r_1, r_2, are the inner and the outer radius of the fins. Thus r_1 and r_2 are equal to: $r_1 = D_{tube}/2$, $r_2 = D_{tube}/2 + L$ for the right (outer) surface, while: $r_1 = D_{tube}/2 – \mathrm{L}$, $r_2 = D_{tube}/2$ for the inner (left) surface.

- Spines

The surface areas per unit length are equal to (see Figure 5-21.c):

$A_{base,\ no\ fin} = \pi \cdot (\Delta^2/4 - t^2)$
$A_{base,\ fin} = \pi \cdot t^2$
$A_{fin} = 2 \cdot \pi \cdot L \cdot t + \pi \cdot t^2$

Consequently, the fin enhancement is given by:

$$E^{comp} = 1 - \left(\frac{2t}{\Delta}\right)^2 + \left(\frac{2t}{\Delta}\right)^2 \cdot \left(1 + \frac{2L}{t}\right) \cdot \eta^{comp}$$

The fin efficiency is given by ([21], section 2.4):

$$\eta^{comp} = \frac{\tanh(\sqrt{2} \cdot \beta^{comp} \cdot L)}{\sqrt{2} \cdot \beta^{comp} \cdot L}$$

β^{comp} is defined as above.

The program calculates an average fin/spine temperature (it is not used in the calculations but is included for plotting purposes). The average fin temperature is obtained using the definition of the fin efficiency. The fin/spine efficiency is defined as the ratio of the actual heat transfer to the heat that would be transferred if the fins/spines were isothermal [20] page 3-115. Therefore:

$$\eta = \frac{T_{fin} - T_{fluid}}{T_{wall} - T_{fluid}}$$

Here T_{wall} is the wall temperature (at the base of the fin/spine) and T_{fluid} is the fluid temperature. Therefore:

$$T_{fin} = T_{fluid} + \eta \cdot (T_{wall} - T_{fluid})$$

The heat capacity of the fin/spine material is accounted for in the following way. The fin does not exist as a separate mesh cell in the 1-D conduction model. In the previous SPECTRA versions the fin/spine heat capacity was ignored altogether. In the current version the heat capacity of the fin/spine material is accounted for by enlarging the heat capacity of the boundary mesh cell (the cell to which the fin/spine is attached). In order to do this a "cell volume increase factor" is calculated by the program (and printed during the input processing). The cell volume increase factors are calculated from the following formulae.

- Rectangular Fins:

  The volumes of fins and the boundary cell are equal to:

  $V_{fin} = 2 \cdot L \cdot t$
  $V_{cell} = \Delta \cdot x$

  Here x is the thickness of the boundary cell. The volume increase factor is equal to:

$$\frac{V_{cell} + V_{fin}}{V_{cell}} = 1 + 2 \cdot \left(\frac{L \cdot t}{\Delta \cdot x} \right)$$

- Cylindrical fins

  Left (inner) surface:
  $V_{fin} = (\pi/4) \cdot [\, D^2 - (D - 2 \cdot L)^2 \,] \cdot 2 \cdot t = 2\pi \cdot L \cdot t \,(D - L)$
  $V_{cell} = (\pi/4) \cdot [\, (D + 2 \cdot x)^2 - D^2 \,] \cdot \Delta = \pi \cdot \Delta \cdot x \,(D + x)$

$$\frac{V_{cell} + V_{fin}}{V_{cell}} = 1 + 2 \cdot \left(\frac{L \cdot t}{\Delta \cdot x} \right) \cdot \left(\frac{D - L}{D + x} \right)$$

  Right (outer) surface:
  $V_{fin} = (\pi/4) \cdot [\, (D + 2 \cdot L)^2 - D^2 \,] \cdot 2 \cdot t = 2\pi \cdot L \cdot t \,(D + L)$
  $V_{cell} = (\pi/4) \cdot [\, D^2 - (D - 2 \cdot x)^2 \,] \cdot \Delta = \pi \cdot \Delta \cdot x \,(D - x)$

$$\frac{V_{cell} + V_{fin}}{V_{cell}} = 1 + 2 \cdot \left(\frac{L \cdot t}{\Delta \cdot x} \right) \cdot \left(\frac{D + L}{D - x} \right)$$

- Spines

  $V_{fin} = \pi \cdot t^2 \cdot L$
  $V_{cell} = (\pi/4) \cdot \Delta^2 \cdot x$

$$\frac{V_{cell} + V_{fin}}{V_{cell}} = 1 + 4 \frac{L \cdot t^2}{\Delta^2 \cdot x}$$

During the transient the code calculates the actual heat capacity of the boundary cell as the true heat capacity of the cell material (at the current cell temperature) multiplied by an appropriate volume increase factor. This implies that the fin material is the same as the material of the boundary cell. This is the default setting (see Volume 2). The user may choose different material for the fin than for the boundary cell. In such case a warning message is issued informing the user that this is appropriate if the heat capacities of both materials are the same (or at least similar).

5.9 Simplified Thermal Radiation Model - Structure-to-Gas Radiation

An elaborate thermal radiation model is available through the Thermal Radiation Package (Chapter 8). Use of such model often involves significant effort, needed to prepare the view factor and beam length data. Therefore a simple radiation model is available for cases when thermal radiation is less important. The simplified radiation model calculates radiative heat exchange between wall surface and surrounding gas. The gas is assumed to be opaque (gas emissivity and absorptivity are equal to one). For such cases the radiative heat flux is given by [113]:

$$q_r = \sigma \cdot \varepsilon_w(T_w) \cdot \left(T_w^4 - T_g^4\right)$$

σ is the Stefan-Boltzmann constant, σ=5.67×10^{-8} W/(m^2-K^4), $\varepsilon_w(T_w)$ is the wall emissivity, defined by the user a function of temperature (IQRLSC / IQRRSC > 1000), T_w, T_g, are the wall and the gas temperatures, respectively.

The assumption of gas opacity is taken for modeling consistency. Gas emissivity could be introduced into the above equation by replacing the wall emissivity, $\varepsilon_w(T_w)$, by an effective wall-gas emissivity, ε_{wg}, defined as [113]:

$$\varepsilon_{wg} = \frac{1}{\frac{1}{\varepsilon_w(T_w)} + \frac{1}{\varepsilon_g(T_g)} - 1}$$

Such method is used for example in the MELCOR code, Heat Structure Package [46]. This approach may be criticized for the following reason. If the detailed net enclosure model (Chapter 8) or the structure-to-structure model (section 5.10) is not used, the radiative heat exchange can be correctly modeled only if the atmosphere is opaque. In any other case, i.e. if the gas emissivity and absorptivity are smaller than one, then the use of the effective emissivity is approximately correct only in case of hot gas radiating to relatively cold walls, for example in case of a burning chamber.

In cases when there is no internal energy source in the gas, the gas temperature equilibrates quickly with the temperatures of the surrounding walls, because of the relatively low heat capacity of gas. In such case the gas acts mainly as a screen between radiating walls, and radiative exchange between walls is minimal in case of opaque atmosphere (ε_g=1.0), increases with decreasing gas emissivity, and reaches maximum in case of transparent atmosphere (ε_g=0.0) - Figure 5-22 (see section 8.3).

Use of the effective emissivity will result in the radiative exchange being maximal in case of opaque atmosphere (ε_g=1.0), decreasing with decreasing gas emissivity, and equal to zero in case of transparent atmosphere (ε_g=0.0) - Figure 5-22. This dependence is opposite to that observed in reality. Therefore, if the detailed thermal radiation model (the model described in Chapter 8) is not used, then typically the best approximation is obtained by assuming that the atmosphere is opaque.

Structure-to-gas radiation is calculated only on the surfaces that are not covered by liquid (the uncovered surface area is used). Verification of the structure-to-gas radiation model is shown in Volume 3. It is shown that the results of the simplified model are the same as the results of the more accurate net enclosure model, with the gas emissivity equal to 1.0.

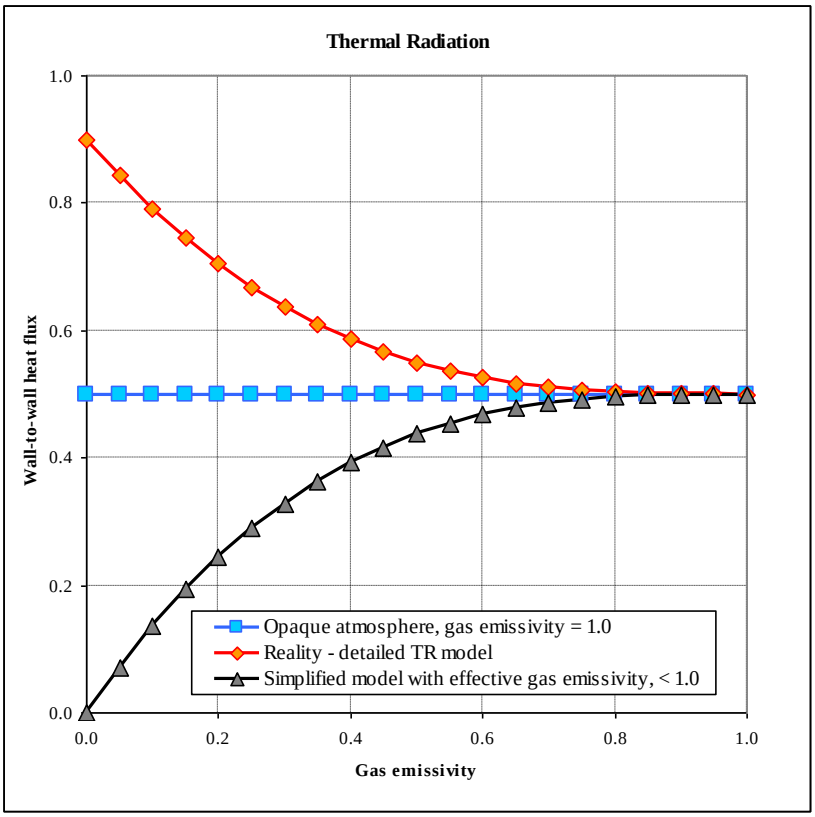


Figure 5-22 Influence of gas emissivity on wall-to-wall radiative heat transfer.

5.10 Simplified Thermal Radiation Model - Structure-to-Structure Radiation

While the structure-to-gas radiation model, described in the previous section, represents the situation where gas emissivity is equal to 1.0, the structure-to-structure radiation model represents the situation where gas emissivity is equal to 0.0. The user has to define pairs of surfaces radiating to each other. The same results may be obtained using the net enclosure model with gas emissivity equal to zero, but the input is much simpler. In case when there is multiple pairs of surfaces radiating to one another (e.g. reactor vessel wall and RCCS wall in High Temperature Reactor) it is much easier to define a set of radiating pairs of surfaces, rather that defining a large number of enclosures with large matrices of view factors.

The model is available only for 1-D Solid Heat Conductors. An arbitrary number of 1-D Solid Heat Conductor surface pairs may be defined by the user as described in Volume 2 (records 327XXX, 328XXX). A single SC may be a member of a single radiating pair only. In order to activate the model, the surface emissivity must be defined for both surfaces of the pair, exactly the same way as it is done for the structure-to-gas model (IQRLSC / IQRRSC > 1000). The radiative exchange is calculated using the following relationship:

$$q_{1-2} = \varepsilon_{1-2} \cdot \sigma \cdot \left(T_1^4 - T_2^4 \right)$$

q_{1-2} heat flux from surface 1 to 2, W/m^2,
σ Stefan-Boltzmann constant, =5.67×10^{-8} W/(m^2-K^4),
T_1 temperature of surface 1, K,
T_2 temperature of surface 2, K,
ε_{1-2} effective emissivity between surfaces 1 to 2.

The effective emissivity is obtained from

$$\varepsilon_{1-2} = \left(\frac{1}{\varepsilon_1} + \frac{A_1}{A_2} \cdot \left(\frac{1}{\varepsilon_2} - 1 \right) + \frac{1}{F_{1-2}} - 1 \right)^{-1}$$

A_1 area of surface 1, m^2, (the smaller surface)
A_2 area of surface 2, m^2, (the larger surface)
ε_1 emissivity (temperature-dependent) of surface 1,
ε_2 emissivity (temperature-dependent) of surface 2,
F_{1-2} view factor from surface 1 (the smaller surface) to surface 2 (the larger surface).

If both surfaces have the same area, then the formula reduces to:

$$\varepsilon_{1-2} = \left(\frac{1}{\varepsilon_1} + \frac{1}{\varepsilon_2} - 1 \right)^{-1}$$

The formula is based on view factor F_{1-2} = 1.0. If one of the surfaces is larger, the calculation procedure is as follows. If the user does not specify any view factor, the code assumes that the view factor from the smaller to the larger surface is 1.0. The view factor from the larger surface to the smaller one follows from the reciprocity relation: $F_{2-1} = A_1 / A_2$. The effective emissivity is given by:

$$\varepsilon_{1-2} = \left(\frac{1}{\varepsilon_1} + \frac{A_1}{A_2} \cdot \left(\frac{1}{\varepsilon_2} - 1 \right) \right)^{-1}$$

This is the so-called Christiansen system, discussed in Volume 3 (section: Five Radiating Surfaces - Christiansen System). The user may define the emissivity $F_{1-2} < 1.0$. In such case $F_{2-1}=F_{1-2}\times A_1/A_2$ and the effective emissivity is obtained from the most general formula, shown above (the same formula is used in MELCOR for structure-to-structure radiation model [46]). The heat fluxes obtained from the smaller surface, q_{1-2}, and the larger surface, q_{2-1}, are equal to:

$$q_{1-2} = \varepsilon_{1-2} \cdot \sigma \cdot \left(T_1^4 - T_2^4 \right)$$
$$q_{2-1} = \varepsilon_{1-2} \cdot \sigma \cdot \left(T_2^4 - T_1^4 \right) \cdot \frac{A_1}{A_2}$$

Structure-to-structure radiation is calculated only on the surfaces that are not covered by liquid (the uncovered surface area is used). Verification of the structure-to-structure radiation model is shown in Volume 3. It is shown that the results of the simplified model are the same as the results of the more accurate net enclosure model, with the gas emissivity equal to 0.0.

5.11 Structural Failure Model

The failure model may be applied for 1-D and 2-D Solid Heat Conductors. If it is applied, it calculates if failure has occurred, time to rupture, and cumulative damage. These parameters are available in the SPECTRA output and as arguments to Control Functions. The user may use these parameters to model an opening of a flow path when the rupture is calculated to occur .

The failure model considers the following three failure mechanisms: the meltdown, the creep rupture, and the ultimate strength failure. Those mechanisms are discussed subsequently below.

- Meltdown

  The meltdown occurs when the current stress exceeds the melting temperature:

  $$T \geq T_M$$

  Here T and T_M are the temperature and the melting temperature, respectively.

- Creep rupture

  Creep rupture is calculated using the method proposed by Larson and Miller [178]. The time to rupture is obtained from the following relation ([179], section B-5.1):

  $$LMP' = 0.001 \cdot T' \cdot [C' + \log_{10}(t_r')]$$

  Here T' is temperature in Rankine, t_r' is time to rupture in hours, C' is a material constant. In SPECTRA SI units are used, therefore the temperature is expressed in Kelvins, while the time to rupture is expressed in seconds. The time to rupture is expressed as:

  $$\log_{10}(t_r) = \frac{LMP}{T} - C$$

  Conversion from Rankine to Kelvins and from hours to seconds means that when the formula is expressed in SI units $LMP = 1000/1.8 \times LMP'$ and $C = C' - \log_{10}(3600) = C' - 3.56$.

  The failure is calculated using the concept of cumulative damage, described in [179], section B-5.1. A damage that the material suffers when subjected to temperature T and stress σ is defined as equal to the inverse of the time to rupture calculated for the current parameters:

  $$D = \frac{1}{t_r(T, \sigma)}$$

  A cumulative damage is defined as:

  $$CD = \int D dt$$

During the calculations the integration is approximated by a summation:

$$CD(t) = CD(t - \Delta t) + D \cdot \Delta t$$

The failure occurs when the cumulative damage reaches the value of 1.0. It may easily be checked that when the stress and temperature are constant, the cumulative damage becomes equal to 1.0 after the time equal to t_r.

The Larson-Miller parameter, *LMP*, is approximated in SPECTRA by the following correlation:

$$LMP = A - B \cdot \log_{10}(\sigma)$$

Here *A* and *B* are material constants and σ is stress in Pa. In literature English units are often used, where stress is expressed in ksi, equal to 1000 psi:

$$LMP' = A' - B' \cdot \log_{10}(\sigma')$$

One ksi is 1000 psi. Since 1 bar = 14.504 psi, 1 ksi is equal to $1000/14.504 \times 10^5 = 6.89465 \times 10^6$ Pa. Therefore when the Pa are used for stress, then: $A'' = A' + B' \times \log_{10}(6.89 \times 10^6) = A' + B' \times 6.8385$. A conversion of *LMP'* to *LMP* is still needed. As shown above, the factor 1000/1.8 is needed to account for that conversion. Therefore: $A = A'' \times (1000/1.8)$. For the same reason the factor *B'* must be multiplied by the factor 1000/1.8. Therefore: $B = B' \times (1000/1.8)$. Finally if the SI units are to be applied, then the coefficients must be converted as follows:

$$A = (A' + B' \times 6.8385) \times (1000 / 1.8)$$
$$B = B' \times (1000 / 1.8)$$
$$C = C' - 3.56$$

- Ultimate strength failure

  The ultimate strength failure occurs when the current stress exceeds the ultimate strength:

  $$\sigma \geq \sigma_U$$

  Here σ and σ_U are the stress and the ultimate strength respectively. The ultimate strength depends on temperature. In SPECTRA the following approximation is taken:

  - At low temperatures a constant value is used:

    $$\sigma \geq \sigma_{U,lowT}$$

  - At high temperatures the *LMP* is used, with the time to rupture, $t_r = t_{U,highT}$, obtained to provide a good fit to the experimental data. The data and the fit used in SPECTRA are discussed below for several frequently used materials.

Reference [179] ultimate strength and creep rupture data for four materials: Inconel-600, stainless steel 304SS, carbon steel SA106B and carbon steel SA533B1. These materials are discussed below.

- Inconel-600

  For the Inconel-600, reference [179] Figure B-31 gives:

  $LMP' = 36.196 - 8.9433 \times \log_{10}(\sigma')$
  $LMP' = T'(13 + \log_{10}(t_r'))(10^{-3})$

  In the SI units:

  $A = (36.196 + 8.9433 \times 6.8385) \times (1000/1.8) = 54086$
  $B = 8.9433 \times (1000/1.8) = 4968.5$
  $C = 13 - 3.56 = 9.44$

  The ultimate strength experimental data for Inconel-600 is shown in [179], Figure B-29. Using a trial and error method it was found out that the experimental data may be well approximated by the creep rupture line with the time to rupture of $t_r = 634$ s.

  The results are shown in Figure 5-23. The solid red line represents the high temperature ultimate strength line, obtained with $t_{U,highT} = 634$ s. The low temperature ultimate strength is 7.3×10^8 Pa (reference [179], Figure B-29). The dashed red line represents the low temperature ultimate strength. For comparison two creep rupture lines are shown in Figure 5-23. These are the 1 hr and the 10 hr creep lines, shown using blue lines, the dashed line for the 1 hr creep and the solid line for the 10 hr creep.

- Stainless steel 304SS

  For the stainless steel 304SS, reference [179] Figure B-41 gives:

  $LMP' = 43.163 - 9.1555 \times \log_{10}(\sigma')$
  $LMP' = T'(16 + \log_{10}(t_r'))(10^{-3})$

  In the SI units:

  $A = (43.163 + 9.1555 \times 6.8385) \times (1000/1.8) = 58763$
  $B = 9.1555 \times (1000/1.8) = 5086.4$
  $C = 16 - 3.56 = 12.44$

  The ultimate strength experimental data for 304SS is shown in [179], Figure B-38. Using a trial and error method it was found out that experimental data may be well approximated by the creep rupture line with the time to rupture of $t_r = 1120$ s.

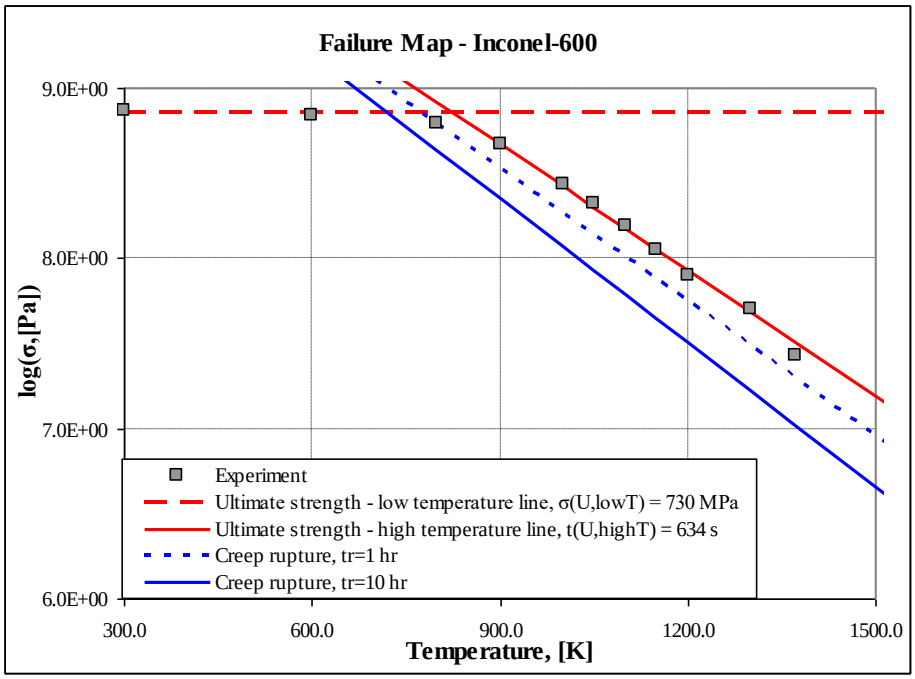


Figure 5-23 Ultimate strength for Inconel-600, data [179] and fit

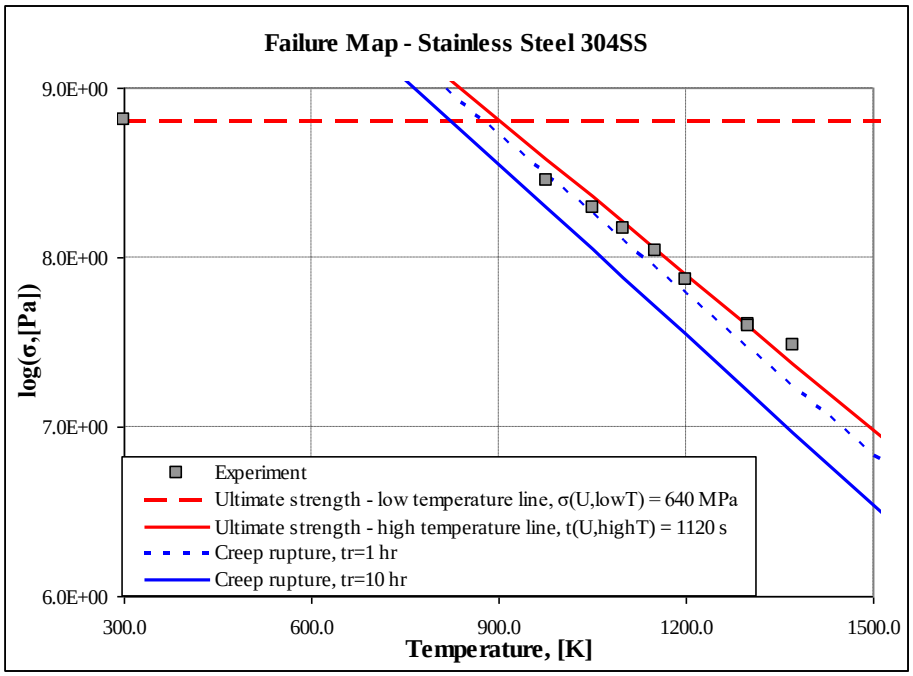


Figure 5-24 Ultimate strength for stainless steel 304SS, data [179] and fit

The results are shown in Figure 5-24. The solid red line represents the high temperature ultimate strength line, obtained with $t_{U,highT}$ = 1120 s. The low temperature ultimate strength is 6.45×10^8 Pa (reference [179], Figure B-38 - the engineering value is used). The dashed red line represents the low temperature ultimate strength. For comparison two creep rupture lines are shown in Figure 5-24. These are the 1 hr and the 10 hr creep lines, shown using blue lines, the dashed line for the 1 hr creep and the solid line for the 10 hr creep.

- Carbon steel SA106B

For the carbon steel SA106B, reference [179] Figure B-50 gives:

$LMP' = 30.867–7.6282\times\log_{10}(\sigma')$
$LMP' = T'(13+\log_{10}(t_r')\,)(10^{-3})$

In the SI units:

$A = (30.867 +7.6282\times6.8385)\times(1000/1.8) = 46129$
$B = 7.6282\times(1000/1.8) = 4237.9$
$C = 13 – 3.56 = 9.44$

The ultimate strength experimental data for SA106B is shown in [179], Figure B-47. Using a trial and error method it was found out that experimental data may be well approximated by the creep rupture line with the time to rupture of t_r = 515 s.

The results are shown in Figure 5-25. The solid red line represents the high temperature ultimate strength line, obtained with $t_{U,highT}$ = 515 s. The low temperature ultimate strength is 5.5×10^8 Pa (reference [179], Figure B-47). The dashed red line represents the low temperature ultimate strength. For comparison two creep rupture lines are shown in Figure 5-25. These are the 1 hr and the 10 hr creep lines, shown using blue lines, the dashed line for the 1 hr creep and the solid line for the 10 hr creep.

- Carbon steel SA533B1

For the carbon steel SA533B1, reference [179] provides creep data (Table B-10). Data from that table has been approximated as follows:

$LMP' = 48.775–12.5476\times\log_{10}(\sigma')$
$LMP' = T'(20+\log_{10}(t_r')\,)(10^{-3})$

In the SI units:

$A = (48.775 +12.5476\times6.8385)\times(1000/1.8) = 74768$
$B = 12.5476\times(1000/1.8) = 6970.9$
$C = 20 – 3.56 = 16.44$

The ultimate strength experimental data for SA533B1 is shown in [179], Figure F-1.7-5. Using a trial and error method it was found out that experimental data may be well approximated by the creep rupture line with the time to rupture of t_r = 30 s.

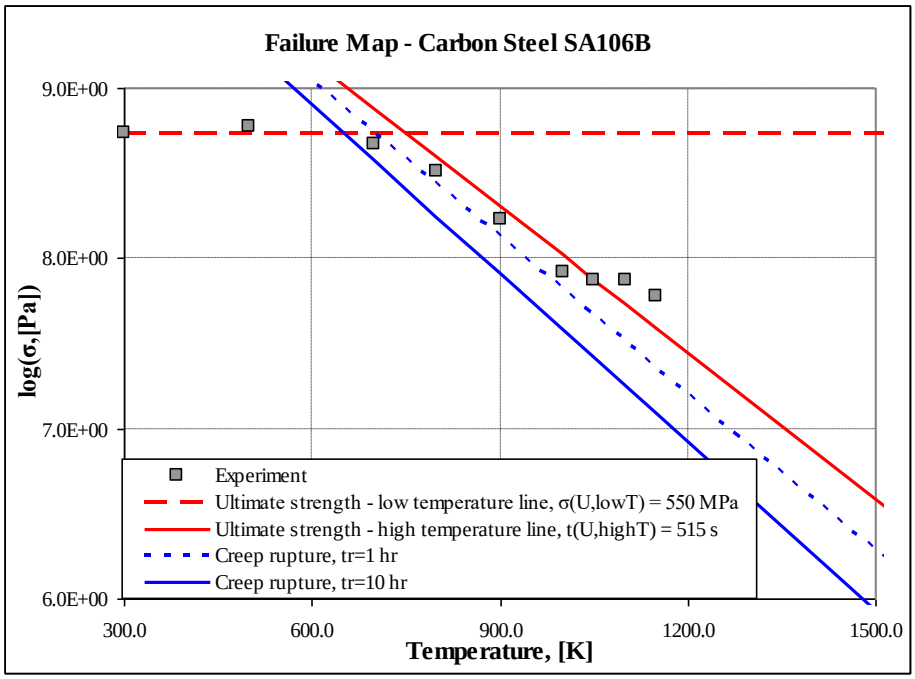


Figure 5-25 Ultimate strength for carbon steel SA106B, data [179] and fit

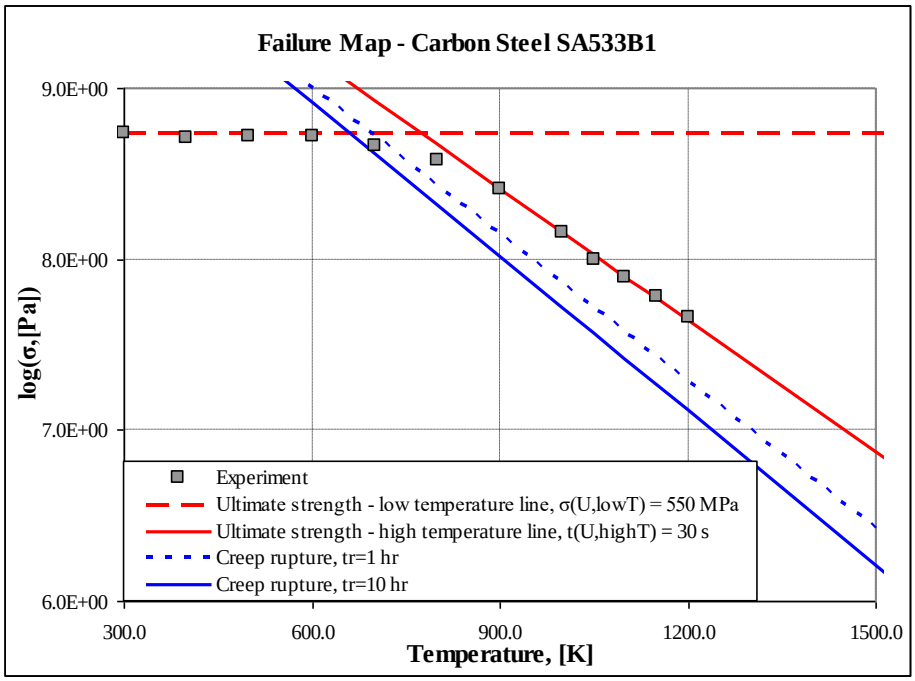


Figure 5-26 Ultimate strength for carbon steel SA533B1, data [179] and fit

The results are shown in Figure 5-26. The solid red line represents the high temperature ultimate strength line, obtained with $t_{U,highT}$ = 30 s. The low temperature ultimate strength is 5.5×10^8 Pa (reference [179], Figure F-1.7-5). The dashed red line represents the low temperature ultimate strength. For comparison two creep rupture lines are shown in Figure 5-26. These are the 1 hr and the 10 hr creep lines, shown using blue lines, the dashed line for the 1 hr creep and the solid line for the 10 hr creep.

Data for all materials is summarized in Table 5-1.

Table 5-1 Failure model data for some frequently used materials

Constants	Material			
	Inconel-600	Steel 304SS	Steel SA106B	Steel SA533B1
A	54,086	58,763	46,129	74,768
B	4,968.5	5,086.4	4,237.9	6,970.9
C	9.44	12.44	9.44	16.44
T_M	1644.0	1671.0	1789.0	1789.0
$\sigma_{U,lowT}$	7.3×10^8	6.4×10^8	5.5×10^8	5.5×10^8
$t_{U,highT}$	634.0	1120.0	515.0	30.0

5.12 Gap Conductance Model

The gap conductance model may be applied for 1-D and 2-D Solid Heat Conductors. If it is applied, it calculates the effective conductance and the effective heat transfer coefficient of the gap between fuel and cladding of the fuel element. The gap calculations are performed for those 1-D and/or 2-D Solid Heat Conductors, which are designated as the fuel regions. The fuel regions are discussed in section 5.12.1. The gap model is discussed in section 5.12.2.

5.12.1 Fuel Regions

As a first step the user must define the following:

- *Fuel regions*. The fuel regions define which 1-D and/or 2-D Solid Heat Conductors are used to represent the fuel rods in the reactor core.
- *Common fuel regions*. The common fuel regions define which of the fuel regions represent the same fuel rod (or group of rods) and therefore share the same gap properties, such as gap pressure and gas composition in the gap.

For example, suppose the reactor core is modelled as one hot rod and one average rod, each of them is represented by five 1-D Solid Heat Conductors in the axial direction (Figure 5-27). The hot rod is represented by SC-101 through SC-105. The average rod is represented by SC-201 through SC-205. The fuel regions and the common fuel regions are in such case defined as follows:

- Fuel regions (total reactor core):
  SC-101, SC-102, SC-103, SC-104, SC-105, SC-201, SC-202, SC-203, SC-204, SC-205

- Common fuel regions:
  - Common region 1 (hot rod):
    SC-101, SC-102, SC-103, SC-104, SC-105
  - Common region 2 (average rod):
    SC-201, SC-202, SC-203, SC-204, SC-205

The hot rod and the average rod, shown in Figure 5-27, are made of 6 nodes. Nodes 1 - 4 represent the fuel, node 5 represents the gap, node 6 represents the cladding. This is the case for all SC-s in Figure 5-27. This is not a formal requirement; i.e. in general the gap node number may be different for different fuel regions or even for the same common fuel region. For each fuel region the user must select a gap model to be applied in this region. The gap model is described in the following section. The user may define up to 10 gap models with different parameters, such as surface emissivities, surface roughness, direct contact pressure, etc.

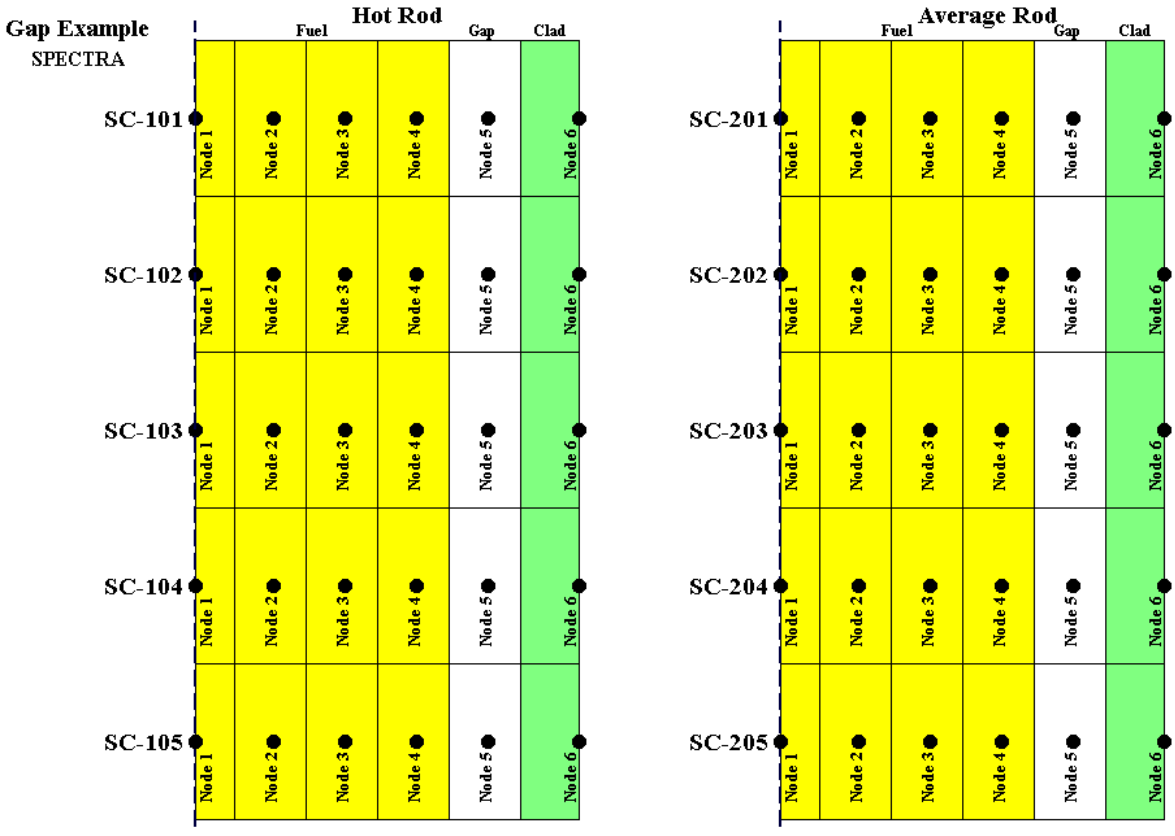


Figure 5-27 Example of fuel regions for gap calculations

From the point of view of the 1-D or 2-D conduction equation, the gap node is only different from other nodes by the way thermal conductivity of the material is obtained. While for the non-gap nodes, the thermal conductivity is obtained by interpolating material property data using the actual node temperature, for the gap node thermal conductivity is set after the gap conductance is calculated as:

$$k_{gap} = h_{gap} d_{gap}$$

Here h_{gap} is the gap conductance (W/m^2-K), calculated as described in section 5.12.2; d_{gap} is the gap thickness (m), which may be either constant or calculated from the dynamic expansion model, described in section 5.12.3. If the dynamic expansion model the effective conductivity of the gap node is obtained by taking into account the changes of gap geometry as:

$$k_{gap} = h_{gap} d_{gap} \frac{r_f' + r_c'}{r_f + r_c}$$

Here r_f and r_c are the original fuel outer radius and cladding inner radius (Figure 5-28), as defined by the Solid Conductor geometry data, while r_f' and r_c' are the actual dimensions, calculated by the dynamic expansion model. The correction factor is the ratio of the actual gap mean radius $(r_f' + r_c')/2$ to the original gap mean radius $(r_f' + r_c')/2$.

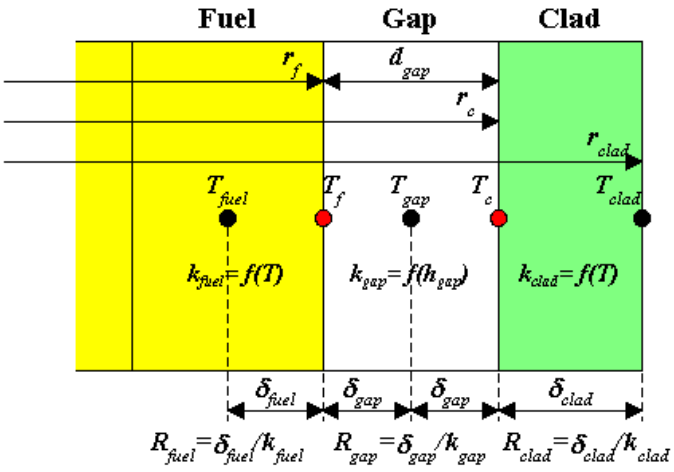


Figure 5-28 Gap parameters

5.12.2 Gap Conductance Model

Gap conductance calculation is discussed in this section. The gap conductance may be defined in one of the two ways:

- For each fuel region the gap conductance is equal to a constant, user-defined value.
- For each fuel region the gap conductance is calculated using the gap conductance model, described below.

With the gap conductance model, the overall heat transfer coefficient in a gap is computed as a sum of three terms:

$$h_{gap} = h_r + h_c + h_s$$

h_r radiation, (W/m^2-K)
h_c conduction through gas, (W/m^2-K)
h_s conduction through solid, due to contact of fuel and cladding, (W/m^2-K)

- *Radiation*. The radiation term is obtained as:

$$h_r = \sigma \cdot \left(\frac{1}{\varepsilon_f} + \frac{1}{\varepsilon_c} - 1 \right)^{-1} \cdot \left(T_f^2 + T_c^2 \right) \cdot \left(T_f + T_c \right)$$

Here ε_f, ε_c, are the emissivities of fuel and cladding, T_f, T_c, are the surface temperatures of fuel and cladding, σ is the Stefan-Boltzmann constant, equal to 5.67×10^{-8} (W/m^2-K^4).

The surface temperatures of fuel and cladding, T_f, T_c, are directly available from the SC/TC solution scheme when the Method 2 is used - Figure 5-9. If Method 1 is used, nodes are placed in the cell centers, except for the boundary cells - see Figure 5-1. In such case the temperatures T_f, T_c must be estimated, using the nearest node temperatures T_{fuel}, T_{gap}, T_{clad} (see Figure 5-28).

The estimation of T_f, T_c, in case of Method 1 is done using the following relations:

$$T_f = T_{gap} + X_{fuel} \cdot (T_{fuel} - T_{gap})$$
$$T_c = T_{gap} + X_{clad} \cdot (T_{clad} - T_{gap})$$

where:

$$X_{fuel} = \frac{R_{gap}}{R_{gap} + R_{fuel}} = \frac{\delta_{gap} / k_{gap}}{\delta_{gap} / k_{gap} + \delta_{fuel} / k_{fuel}}$$
$$X_{clad} = \frac{R_{gap}}{R_{gap} + R_{clad}} = \frac{\delta_{gap} / k_{gap}}{\delta_{gap} / k_{gap} + \delta_{clad} / k_{clad}}$$

T_{fuel} temperature of the fuel node, (K) (Figure 5-28)
T_{clad} temperature of the cladding node, (K) (Figure 5-28)
R_{fuel} thermal resistance of the fuel, (m^2-K/W)
R_{clad} thermal resistance of the cladding node boundary, (m^2-K/W)
R_{gap} thermal resistance of the gap node boundary, (m^2-K/W)
δ_{fuel} distance from the fuel node to the fuel node boundary, (m)
δ_{clad} distance from the cladding node to the cladding node boundary, (m)
δ_{gap} distance from the gap node to the gap node boundary, (m)
k_{fuel} thermal conductivity of fuel, (W/m-K)
k_{clad} thermal conductivity of cladding, (W/m-K)
k_{gap} effective thermal conductivity of the gap, (W/m-K), (= $h_{gap}\times2\delta_{gap}$)

The above relations are based on the thermal resistances of fuel ($R_{fuel} = \delta_{fuel}/k_{fuel}$), cladding ($R_{clad} = \delta_{clad}/k_{clad}$), and gap ($R_{gap} = \delta_{gap}/k_{gap}$) and strictly speaking are valid only in stationary conditions. In practice a very small error is introduced by applying these relations also in transient conditions. Since the interpolation factors depend on h_{gap}, which has to be calculated, the calculation must be performed iteratively. This is done within the internal iteration, which is needed not only because of the gap but also because the material properties may be temperature-dependent. Temperatures of Heat Conductors are always solved interactively. The old time step values are taken as a first step for the iteration.

- *Gas conduction*. The conduction term is obtained as:

$$h_c = \frac{k_g}{d_{gap} + d_{\min} + (g_f + g_c)}$$

Here k_g is the conductivity of the gas mixture, d_{gap} is the gap thickness ($d_{gap}=2\delta_{gap}$), $d_{\min}$ is related to the roughness of the two surfaces, g_f, g_c, are the temperature jump distances. The $d_{\min}$ is obtained from:

$$d_{\min} = C_d \cdot (r_f + r_c)$$

Here r_f, r_c, are the surface roughness of fuel and cladding respectively, and C_d is constant. The value of C_d recommended in literature ranges from 0.62 (Cetinkale and Fishenden [187]) to 3.2 (RELAP5 [44]) - see [188]. The default value of C_d in SPECTRA is taken from the Ross and Stoute model [186] and is equal to 1.5 (see Volume 2).

The temperature jump distances, g_f, g_c, are extrapolations of the true gap size to account for discontinuities in temperature at the boundary surface of the gas - see Figure 5-29.

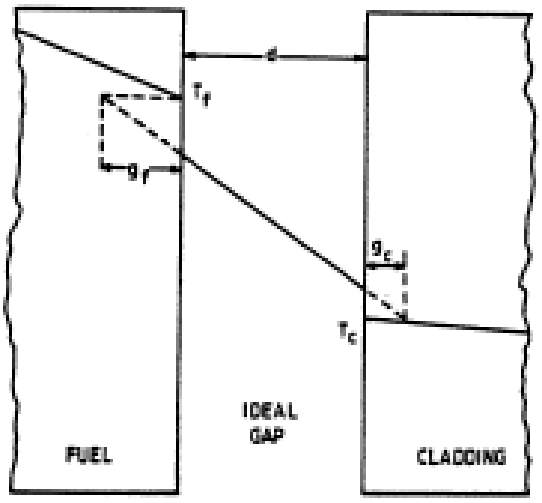


Figure 5-29 Temperature jump distances, g_1, g_2, reproduced from [189]

The discontinuities arise not only from imperfect energy exchange between the gas and the surface, but also because the probabilities of a gas molecule colliding with another gas molecule and with solid surface are markedly different [189]. The sum of the temperature jump distances is obtained from:

$$(g_f + g_c) = C_j \cdot \frac{k_g \cdot \sqrt{T_g}}{p_g \cdot \sum_{i=1}^{N_{gas}} X_i a_i M_i^{-1/2}} \cdot$$

Here C_j is a constant, a_i is the accommodation coefficient (-), T_g is the gap gas temperature (K), M_i is the gas molar weight (kg/kmol), X_i is the gas molar fraction (-), and p_g is the gas pressure in the gap (Pa). The constant C_j is equal to 0.0247 - see reference [44]. The thermal accommodation coefficient of gas *i* is based on Ullman data [185] (see also [189]):

$$a_{He} = 0.425 - 2.3 \times 10^{-4} \cdot T_g$$

$$a_{Xe} = 0.740 - 2.5 \times 10^{-4} \cdot T_g$$

$$a_i = a_{He} + \frac{M_i - M_{He}}{M_{Xe} - M_{He}} \cdot (a_{Xe} - a_{He})$$

Here M_{He} is the molar weight of Helium (=4.0), M_{Xe} is the molar weight of Xenon (=131.3), M_i is the molar weight of the gas *i*.

The gas pressure in the gap is computed from:

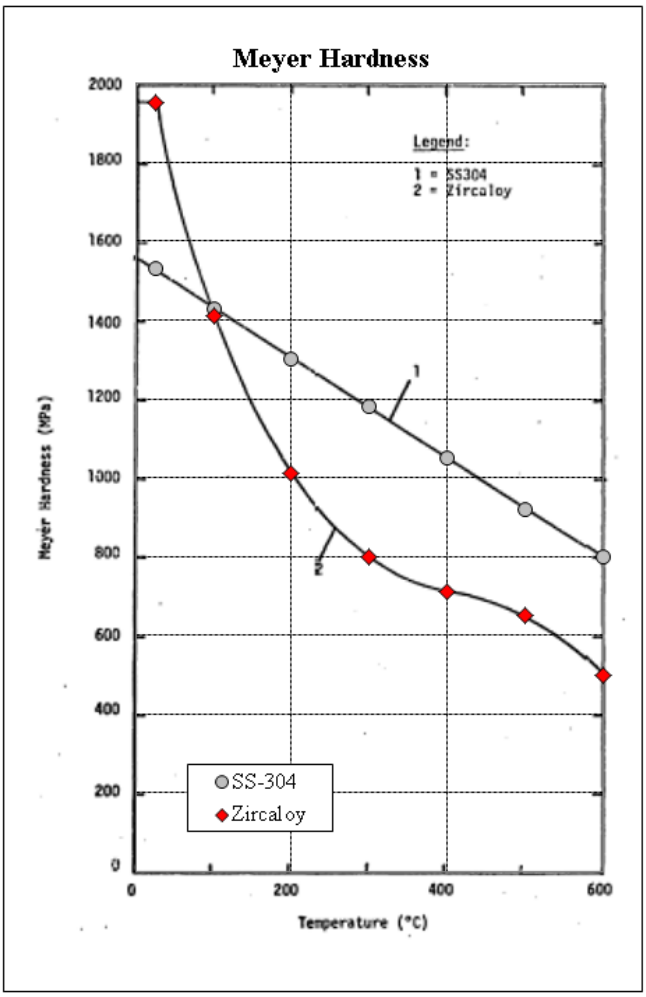
$$p_g = \frac{p_0}{T_0} \cdot T_g$$

Here p_0, T_0 are the gas pressure and temperature in reference state and T_g is the current gas temperature in the gas plenum. The gas plenum is defined by the user for each common region. A CV fluid temperature or SC/TC node temperature may be selected. This should be selected as the volume or structure most closely associated with the non-fuel region in a fuel pin at the top of a stack of fuel pellets.

- *Solid conduction*. The heat transfer due to the fuel-cladding contact is obtained as:

$$h_s = C_s \cdot \frac{k_s}{\sqrt{r_t}} \cdot \left(\frac{p_a}{H_c} \right)^n$$

Here C_s is a constant, k_s is the effective solid conductivity, p_a is the apparent interfacial pressure in the point of contact, H_c is the Meyer hardness of the softer material (cladding), r_t is the effective surface roughness. The constant C_s is equal to 11.89 $m^{-1/2}$ according to the Ross and Stoute model [186] (the original value of 1.189 $cm^{-1/2}$ is converted to SI units). For Zircaloy cladding the Meyer hardness is approximately equal to (see Figure 5-30, reproduced from [190] Figure A-5):



T [K]	T [°C]	SS-304	Zircaloy
298	25	1530	1950
373	100	1430	1410
473	200	1300	1010
573	300	1180	800
673	400	1050	710
773	500	920	650
873	600	800	500
973	700	*680*	*350*
1073	800	*560*	*200*
1173	900	*440*	*50*

Figure 5-30 Meyer hardness of SS-304 and Zircaloy [190]

$T = 300$ K $\quad H_c = 1960\times10^6$ Pa
$T = 900$ K $\quad H_c = 500\times10^6$ Pa

The effective solid conductivity is obtained from:

$$k_s = 2\cdot\frac{k_f \cdot k_c}{k_f + k_c}$$

Here k_f, k_c, are the conductivities (W/m-K) of fuel and cladding respectively. The effective surface roughness is obtained from the Ross and Stoute model [186]:

$$r_t = \sqrt{\frac{r_f^2 + r_c^2}{2}}$$

The solid conduction due to the direct fuel-cladding contact is calculated if a positive value of p_a is supplied in the input data. If no value is supplied, then:

- If no dynamic expansion model is not used, then $p_a = 0.0$
- If the dynamic expansion model is used (section 5.12.3), then p_a is obtained as follows.
  - If the fuel outer radius (r_f') and the cladding inner radius (r_c') obtained from the expansion models (see section 5.12.3) fulfill the relation: $r_f' \leq r_c'$, then:

$$p_a = 0.0$$

  - If $r_f' > r_c'$, then the interfacial pressure is calculated from the cladding elastic deformation, as follows:

$$p_a = \frac{E}{r_{cm}} \cdot \frac{r_{clad} - r_c}{r_c} \cdot \delta_{CE} \cdot X_P$$

E Young modulus of the cladding
r_{cm} cladding mean radius, $r_{cm} = (r_c + r_{clad}) / 2$ - Figure 5-28
δ_{CE} cladding elastic deformation due to fuel rod expansion, $\delta_{CE} = r_f' - r_c'$.
X_P user-defined multiplier

The above formula is obtained assuming that fuel expansion causes elastic deformation of cladding. The cladding elastic deformation due to fuel rod expansion is related to the hoop stress as:

$$\delta_{CE} = r_{cm} \cdot \frac{\sigma_h}{E}$$

Here r_{cm} is the cladding mid-point radius, σ_h is the cladding hoop stress (Pa. The hoop stress is given by:

$$\sigma_h = \frac{p_a \cdot r_I}{r_O - r_I}$$

Here the subscripts *I* and *O* are used to signify inner surface outer surface of cladding respectively, *r* is radius (m) and *P* is pressure (Pa). After rearrangement:

$$\delta_{CE} = \frac{r_{cm}}{E} \cdot \frac{r_I}{r_O - r_I} \cdot p_a$$

and:

$$p_a = \frac{E}{r_{cm}} \cdot \frac{r_O - r_I}{r_I} \cdot \delta_{CE}$$

Here the nomenclature from section 5.12.3 is used, which is somewhat different than the one used in Figure 5-28. One has to keep in mind that $r_I \equiv r_c$, $r_O \equiv r_{clad}$. Therefore:

$$p_a = \frac{E}{r_{cm}} \cdot \frac{r_{clad} - r_c}{r_c} \cdot \delta_{CE}$$

A user-defined multiplier, X_P, is introduced to account for "ridging" at the pellet-pellet interfaces (also called wheatsheafing, hour glassing) - Figure 5-31. Because of this phenomenon the contact pressure is smaller than it would have been if the cladding expansion had been uniform (i.e. fuel pellet remained as an ideal cylinder. Therefore $X_P \leq 1.0$ and the final formula is:

$$p_a = \frac{E}{r_{cm}} \cdot \frac{r_{clad} - r_c}{r_c} \cdot \delta_{CE} \cdot X_P$$

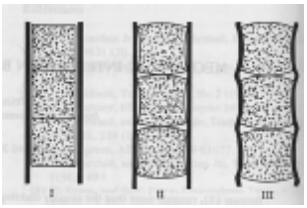


Figure 5-31 Expansion of pellet and cladding [203]

5.12.3 Dynamic Expansion Model

Gap size calculation is discussed in this section. The gap size may be defined in one of the two ways:

- For each fuel region the gap thickness is equal to a constant, user-defined value.
- For each fuel region the gap thickness is calculated using the dynamic expansion model, described below.

Note that the gap size is needed if the gap conductance model is used (section 5.12.2). If a constant, user-defined gap conductance is used, then the gap size, whichever way defined, is not used in calculations (it is still printed in the output/plot files).

With the dynamic expansion model, the gap size is calculated as follows:

$$d_{gap} = r_c' - r_f'$$

Here r_c' and r_f' are the cladding inner and the fuel outer radii respectively, obtained from:

$$r_f' = r_f + x_{FT} + x_{FS}$$
$$r_c' = r_c + x_{CT} + x_{CC} + x_{CE}$$

r_f 'cold' fuel outer radius, based on the SC nodalization - Figure 5-28, (m)
r_c 'cold' cladding inner radius, based on the SC nodalization - Figure 5-28, (m)
x_{FT} fuel thermal expansion, (m)
x_{FS} fuel swell induced by fission products, (m)
x_{CT} cladding thermal expansion, (m)
x_{CC} cladding creepdown, (m)

x_{CE} cladding elastic deformation, (m)

The individual terms are shortly described below.

- *Fuel thermal expansion.* The fuel thermal expansion term is calculated from:

$$x_{FT} = \sum_{i=1}^{N_G-1} d_i \cdot \varepsilon_{FT}(T_i)$$

Here d_i is thickness (m) of the node i, T_i is temperature (K) of the node i, ε_{FT} (-) is a function defining thermal expansion of fuel as a function of temperature. The summation is performed for all fuel nodes, $i < N_G$, N_G is the gap node number. The thermal expansion function must be entered as a table in the input data. For UO_2 and PuO_2 fuels appropriate functions are given by ([101] page 2-49):

$$UO_2 \;:\quad \varepsilon_{FT}(T) = 1.0\times10^{-5}T - 3.0\times10^{-3} + 4.0\times10^{-2}\exp\left(\frac{-6.9\times10^{-20}}{k_B T}\right)$$

$$PuO_2 :\quad \varepsilon_{FT}(T) = 9.0\times10^{-6}T - 2.7\times10^{-3} + 7.0\times10^{-2}\exp\left(\frac{-7.0\times10^{-20}}{k_B T}\right)$$

Here k_B is Boltzmann constant, equal to 1.38×10^{-23} J/K. Values tabulated for relevant temperature range are shown in Figure 5-32. When the fuel is composed of a mixture of UO_2 and PuO_2, the thermal expansion is found by taking a weighted average of the contributions from each component:

$$\varepsilon_{FT}(MOX) = \varepsilon_{FT}(UO_2)\cdot x(UO_2) + \varepsilon_{FT}(PuO_2)\cdot x(PuO_2)$$

- *Fuel swell.* The value of fuel swell due to fission gases, x_{FS}, depends on burn-up history and therefore must be supplied by the user. For the fresh fuel rods the value is zero. For fuel rods with significant burn-up the value of x_{FS} must be calculated by an appropriate code and provided in input.

- *Cladding thermal expansion.* The cladding thermal expansion term is calculated from:

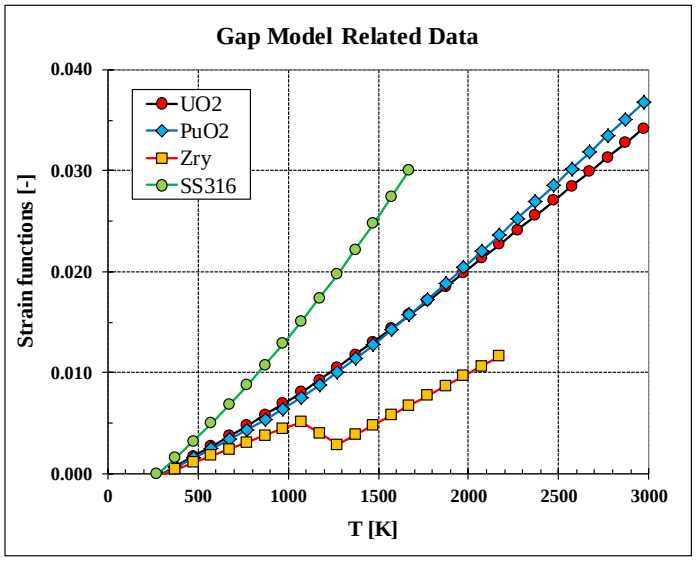
$$x_{CT} = r_{cm}\cdot\varepsilon_{CT}(T_c)$$

Here r_{cm} is the cladding mid-point radius, T_c is the average cladding temperature (K), ε_{CT} (-) is a function defining thermal expansion of cladding as a function of temperature. The summation is performed for all cladding nodes, $N_G < i < N$, N_G is the gap node number, N is the total number of nodes. The thermal expansion function must be entered as a table in the input data. For Zircaloy cladding appropriate functions are given by ([197] page 358):

$$\varepsilon_{CT}(T) = -2.373\times10^{-4} + 6.721\times10^{-6}(T-273) \qquad for \quad T < 1073$$

$$\varepsilon_{CT}(T) = -6.800\times10^{-3} + 9.700\times10^{-6}(T-273) \qquad for \quad T > 1273$$

Values tabulated for relevant temperature range are shown in Figure 5-32.



T [K]	UO2	PuO2	Zry	SS316
273.0	0.000	0.000	0.000	0.000
373.0	7.301E-04	6.571E-04	4.348E-04	1.580E-03
473.0	1.731E-03	1.559E-03	1.107E-03	3.248E-03
573.0	2.736E-03	2.467E-03	1.779E-03	5.004E-03
673.0	3.754E-03	3.394E-03	2.451E-03	6.847E-03
773.0	4.792E-03	4.356E-03	3.123E-03	8.778E-03
873.0	5.860E-03	5.367E-03	3.795E-03	1.080E-02
973.0	6.965E-03	6.438E-03	4.467E-03	1.290E-02
1073.0	8.109E-03	7.576E-03	5.140E-03	1.510E-02
1173.0	9.293E-03	8.784E-03	4.020E-03	1.738E-02
1273.0	1.052E-02	1.006E-02	2.900E-03	1.975E-02
1373.0	1.178E-02	1.140E-02	3.870E-03	2.220E-02
1473.0	1.307E-02	1.279E-02	4.840E-03	2.475E-02
1573.0	1.440E-02	1.424E-02	5.810E-03	2.738E-02
1673.0	1.574E-02	1.573E-02	6.780E-03	3.010E-02
1773.0	1.711E-02	1.726E-02	7.750E-03	
1873.0	1.850E-02	1.882E-02	8.720E-03	
1973.0	1.990E-02	2.041E-02	9.690E-03	
2073.0	2.132E-02	2.202E-02	1.066E-02	
2173.0	2.274E-02	2.364E-02	1.163E-02	
2273.0	2.416E-02	2.527E-02		
2373.0	2.559E-02	2.691E-02		
2473.0	2.703E-02	2.856E-02		
2573.0	2.846E-02	3.021E-02		
2673.0	2.989E-02	3.185E-02		
2773.0	3.132E-02	3.349E-02		
2873.0	3.275E-02	3.513E-02		
2973.0	3.417E-02	3.677E-02		

Figure 5-32 Gap model data for frequently used materials - strain function

- *Cladding elastic deformation.* The cladding thermal expansion term is calculated from:

$$x_{CE} = r_{cm}\left(\frac{\sigma_h - \nu\sigma_z}{E}\right)$$

Here r_{cm} is the cladding mid-point radius, σ_h is the cladding hoop stress (Pa), σ_z is the cladding axial stress (Pa), ν is the Poison ratio. The hoop and axial stresses are given by:

$$\sigma_h = \frac{P_I \cdot r_I - P_O \cdot r_O}{r_O - r_I} \qquad \sigma_z = \frac{P_I \cdot r_I^{\,2} - P_O \cdot r_O^{\,2}}{r_O^{\,2} - r_I^{\,2}}$$

Here the subscripts *I* and *O* are used to signify inner surface outer surface of cladding respectively, *r* is radius (m) and *P* is pressure (Pa) (according to nomenclature used in Figure 5-28: $r_I \equiv r_c$, $r_O \equiv r_{clad}$). NOTE: cladding elastic deformation is calculated only when there is a boundary volume defined (in 1-D IVLRSC>0, in 2-D IVLBTC>0), because only then the cladding outer pressure, P_O, is available as the pressure in the boundary volume.

The Young modulus must be entered as a table in the input data. For Zircaloy cladding appropriate functions are given by ([101] page 4-45):

$$E(T) = 1.088 \times 10^{11} - 5.475 \times 10^{7}\, T \qquad for \quad T < 1090$$
$$E(T) = 9.210 \times 10^{10} - 4.050 \times 10^{7}\, T \qquad for \quad T > 1240$$

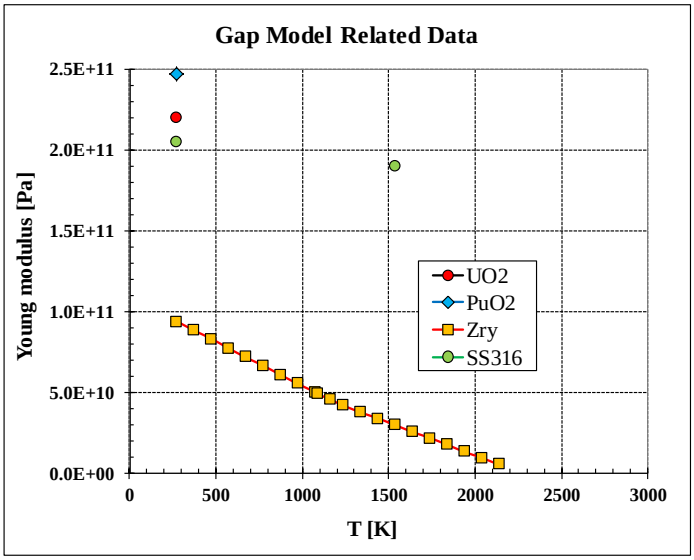
Values tabulated for relevant temperature range are shown in Figure 5-33.

Note that when the calculated deformed cladding radius is smaller than the calculated deformed fuel radius: $r_c' < r_f'$, then the cladding elastic deformation is assumed:

$$x_{CE} = r_f' - r_c'$$

At the same time the fuel-cladding interfacial pressure, p_a (used for the solid conduction term, section 5.12.2) is obtained from the formula for elastic deformation, rearranged to obtain the inner side pressure:

$$p_a = \frac{r_O^{\,2}(1-\nu) + r_I r_O}{r_I^{\,2}(1-\nu) + r_I r_O} \cdot P_O + \frac{r_O^{\,2} + r_I^{\,2}}{r_I^{\,2}(1-\nu) + r_I r_O} \cdot \frac{E}{r_{cm}} \cdot x_{CE} - P_I$$



T [K]	UO2	PuO2	Zry	SS316
273.0	2.200E+11	2.470E+11	9.385E+10	2.050E+11
373.0			8.838E+10	
473.0			8.290E+10	
573.0			7.743E+10	
673.0			7.195E+10	
773.0			6.648E+10	
873.0			6.100E+10	
973.0			5.553E+10	
1073.0			5.005E+10	
1090.0			4.912E+10	
1165.0			4.550E+10	
1240.0			4.188E+10	
1340.0			3.783E+10	
1440.0			3.378E+10	
1540.0			2.973E+10	1.900E+11
1640.0			2.568E+10	
1740.0			2.163E+10	
1840.0			1.758E+10	
1940.0			1.353E+10	
2040.0			9.480E+09	
2140.0			5.430E+09	
2240.0				
2340.0				
2440.0				
2540.0				
2640.0				
2740.0				
2840.0				

Figure 5-33 Gap model data for frequently used materials - Young modulus

- *Cladding creepdown.* The value of cladding creepdown, x_{CC}, depends on burn-up history and therefore must be supplied by the user. For the fresh fuel rods the value is zero. For fuel rods with significant burn-up the value of x_{FS} must be calculated by an appropriate code and provided in input.

The dynamic expansion model is invoked by defining any of the three parameters, fuel thermal expansion, cladding thermal expansion, cladding elastic deformation.

5.12.4 Fuel-Cladding Centerline Shift

In case of fuel-cladding centerline shift the overall gap conductance, h_g, is computed based on the local gap conductance, $h_g'(d_g')$, obtained for the local gap size, d_g', from the following formula:

$$h_g = \frac{1}{\pi}\int_0^{\pi} h_g'[d_g'(\theta)] \cdot d\theta = \frac{1}{N}\sum_{i=1}^{N} h_g'[d_g'(\theta_i)]$$

$$\theta_i = \frac{i-1}{N-1} \cdot \pi$$

The local gap size is obtained from (Figure 5-34):

$$d_g'(\theta_i) \approx R_{c,i} - R_{f,o} - \Delta \cdot \cos(\theta_i)$$

Approximation is good for gaps small compared to the cladding and fuel radii, $d_g \ll R_{c,i} \approx R_{r,o}$.

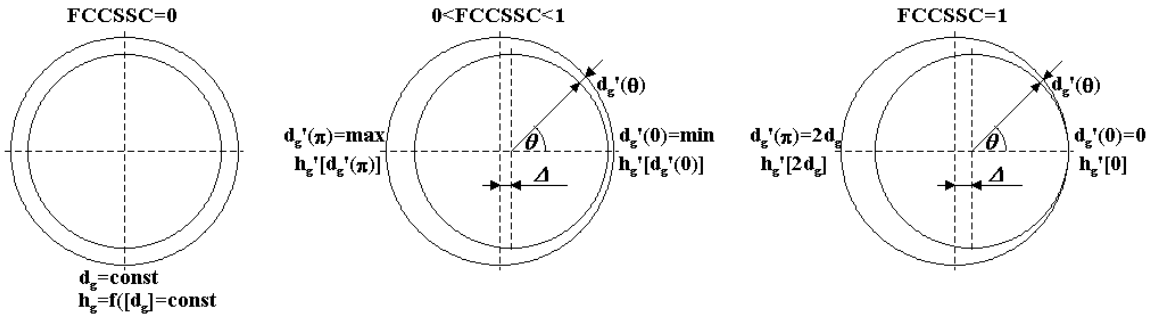


Figure 5-34 Local gap size calculation for fuel-cladding centerline shift

The number of integration points, N, is a user-defined parameter. The minimum number of points is 2. In such case the local gap conductance is calculated in two points only, the smallest gap ($\theta = 0$) and the largest gap ($\theta = \pi$). Influence of the number of integration points is shown in Figure 5-35. It is seen that in practice 50 points are sufficient to obtain an accurate (N-independent) value of h_{gap}.

Figure 5-36 shows the effect of centerline shift on h_{gap} for several different gap thickness. Note that the value of gap conductance for the fuel-cladding centerline shift of 1.0 is between a factor of 2 and a factor of 4 larger than the value for no shift.

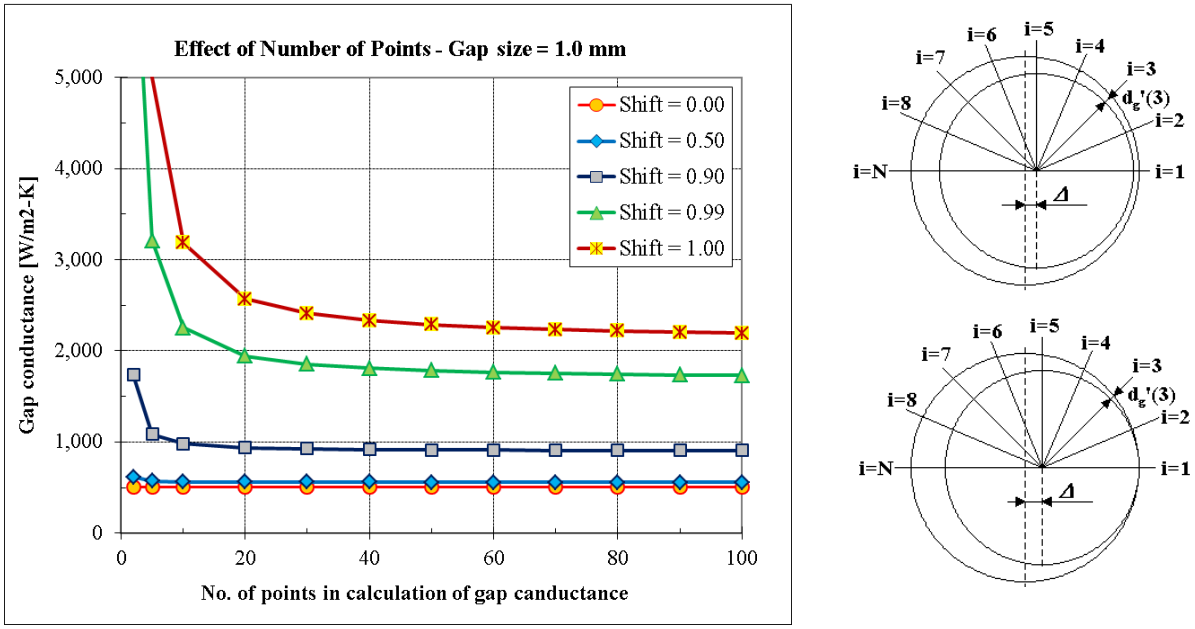


Figure 5-35 Influence of number of points, N, on gap conductance h_{gap}

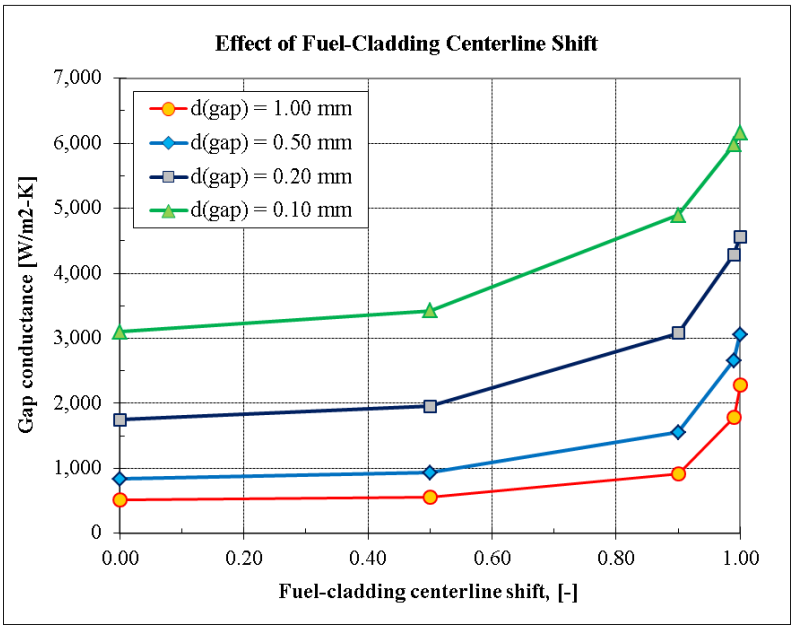


Figure 5-36 Influence of number of points, *N*

5.13 Axial Conduction and Direct Contact Conduction

1-D and 2-D Solid Heat Conductors may be linked to provide multidimensional heat transfer, for example in order to model effective conductivity of a pebble bed or a prismatic block reactor.

The axial heat conduction or direct contact heat transfer between two structures, *i* and *j*, is considered - Figure 5-37. The heat transfer between the two solid heat conductors is calculated from:

$$Q_{i-j} = A_{i-j} \frac{T_i - T_j}{R_{i-j}}$$

Q_{i-j} heat transferred between the solid heat conductors *i* and *j*, (W)
A_{i-j} area of contact between the solid heat conductors *i* and *j*, (m^2)
T_i temperature of the solid heat conductor *i*, (K)
T_j temperature of the solid heat conductor *j*, (K)
R_{i-j} thermal resistance for the axial heat flow between the solid conductors *i* and *j*, (m^2-K/W)

The thermal resistance for the heat flow is obtained from:

$$R_{i-j} = \frac{L_i}{k_i} + \frac{1}{h_{i-j}} + \frac{L_j}{k_j}$$

L_i length for axial conduction in the solid heat conductor *i* (user-input)
k_i thermal conductivity in the solid heat conductor *i*
L_j length for axial conduction in the solid heat conductor *j* (user-input)
k_j thermal conductivity in the solid heat conductor *j*
h_{i-j} heat transfer coefficient at the point of contact of solid heat conductors *i* and *j*

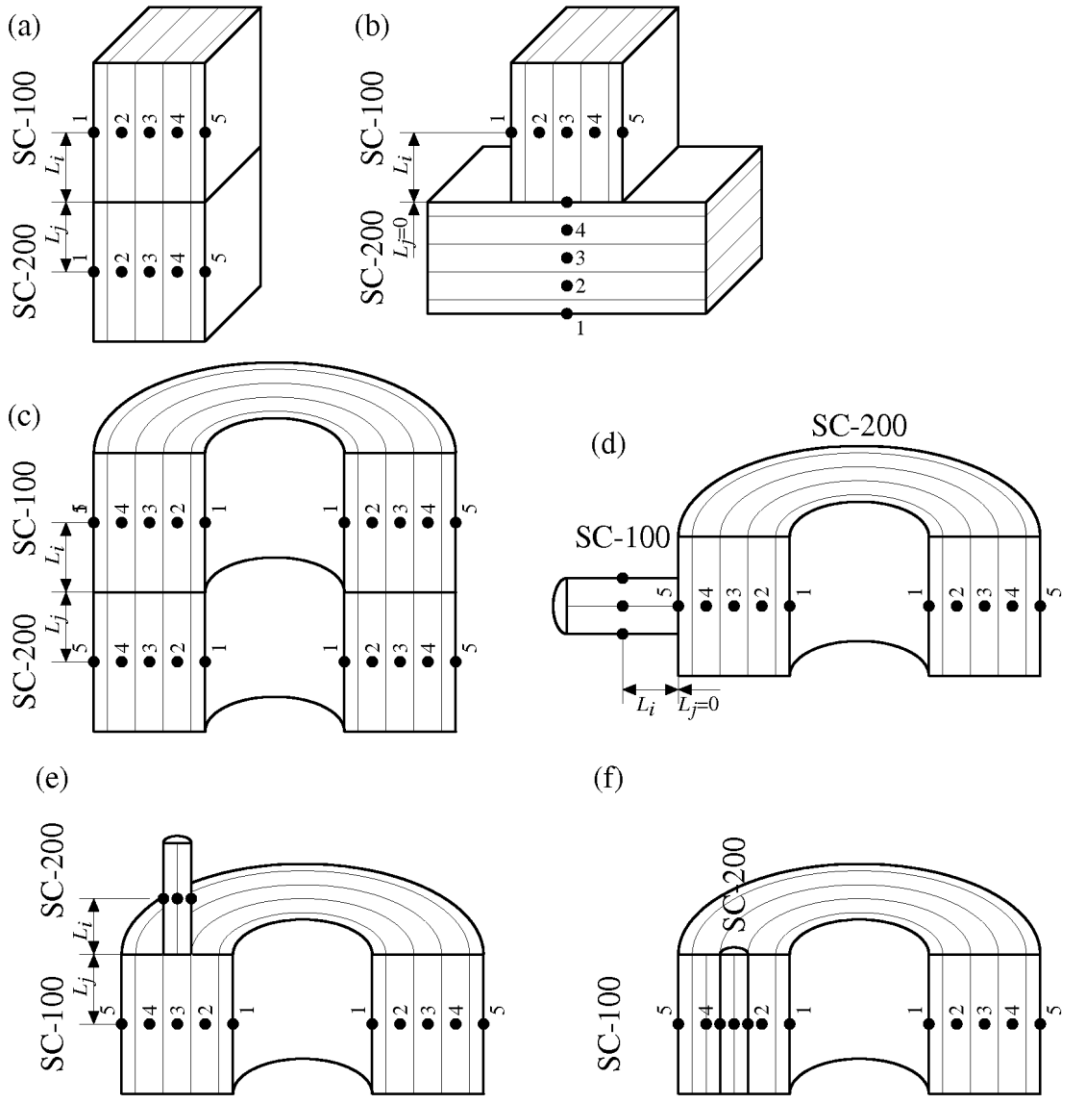


Figure 5-37 Axial conduction, direct contact between different SC

More information about linking different solid heat conductors is provided in Volume 2. Example cases are shown in Volume 3. The term "axial conduction" is used for situations where all cells are linked - Figure 5-37 (a) and (c). In such case both solid heat conductors must have identical geometry. The term "direct contact conduction" is used for situations where a single cell of one solid heat conductor is linked to one or more cells of the other solid heat conductor - Figure 5-37 (b) (d), (e), (f).

In case of axial transfer, individual cells are always linked. T_i and T_j are the local cell temperatures and the heat is deposited in these cells. In case of direct contact heat transfer, the user has an option (see Volume 2, input parameters L1AXSC, L2AXSC) to distribute the heat uniformly over all cells. In this case, a volume-averaged or a mass-averaged temperature of the solid heat conductor is used as

T_i and T_j . This option is most appropriate when modeling radial conduction through prismatic blocks (see Volume 3, test PMR-k-eff).

The direct contact conduction may be used to model the pebble bed reactor or the prismatic block reactor. In case of a pebble bed reactor the user creates spherical structures, representing pebbles. The pebble material is that of graphite and graphite-fuel mixture. The direct contact conduction is then pebbles from one location of the core to another.

An artificial material is defined to be used for the direct contact conduction (see Volume 2, input parameters M1AXSC, M2AXSC). The thermal conductivity of this material may be obtained from available correlations, for example Zehner-Schlunder or Robold correlation, which give effective conductivity of the pebble bed, including conduction, radiation, as well as convection. The heat capacity and the density of such material are irrelevant (since these are defined by the "true" material of the pebbles) and any positive number may be entered. Volume 3 shows in detail how the direct contact conduction may be used to model the pebble bed reactor (test PBR-k-eff) or the prismatic block reactor (test PMR-k-eff).

5.14 Solid Heat Conductors with Size Change during Transient

1-D Solid Heat Conductors may be modeled to change size in time. The variable SC size is defined by the user. When the model is applied, the size (both left and right surface area) of a SC is controlled by a Tabular or a Control Function. The function defines a fraction of the nominal size that is available for heat transfer. The value obtained from the Tabular or Control Function is internally limited to the range between 10^{-3} and 1.0. The thickness and node sizes are not affected.

If the size change is applied and the SC has an internal heat source, the source strength will follow the size change (i.e. the power density, W/m^3, will remain the same). The plot parameter SC-xxx-Qcel-xxxx, giving the power generation in each cell, is however not reflecting this fact (it gives power of the “full size” SC, independently of the value of the multiplier). The model can be used for SCs with direct contact (section 5.13) but in such case the SC size multiplier should be made the same for the SCs that are in contact.

The model may be applied in the following situations:

- SC surfaces are convecting heat,
- simple radiation-model between the surface and gas is used,
- structure-to-structure radiation model is used.

The model cannot be used if SC is a member of the detailed radiation model network, because the radiation view factors do not change in time. Test cases and verification of the model are provided in Volume 3.

6 2-D Solid Heat Conductor Package

6.1 Introduction

The 2-D Solid Heat Conductor Package calculates heat conduction within solid structures, and energy transfer across its boundary surfaces into control volumes.

A 2-D Solid Heat Conductor is a structure that is represented by two-dimensional heat conduction, with specified boundary conditions at each of its two boundary surfaces. The modeling capabilities of solid heat conductors are general, and can represent walls, containment structures, fuel rods with nuclear or electrical heating, piping walls, heat exchangers with smooth or finned tubes, etc. Two different geometries are available for the 2-D Solid Heat Conductors:

- Rectangular geometry.
- Cylindrical geometry.

The rectangular and cylindrical heat conductors may have two orientations in space (Figure 6-1):

- Vertical orientation.
- Horizontal orientation.

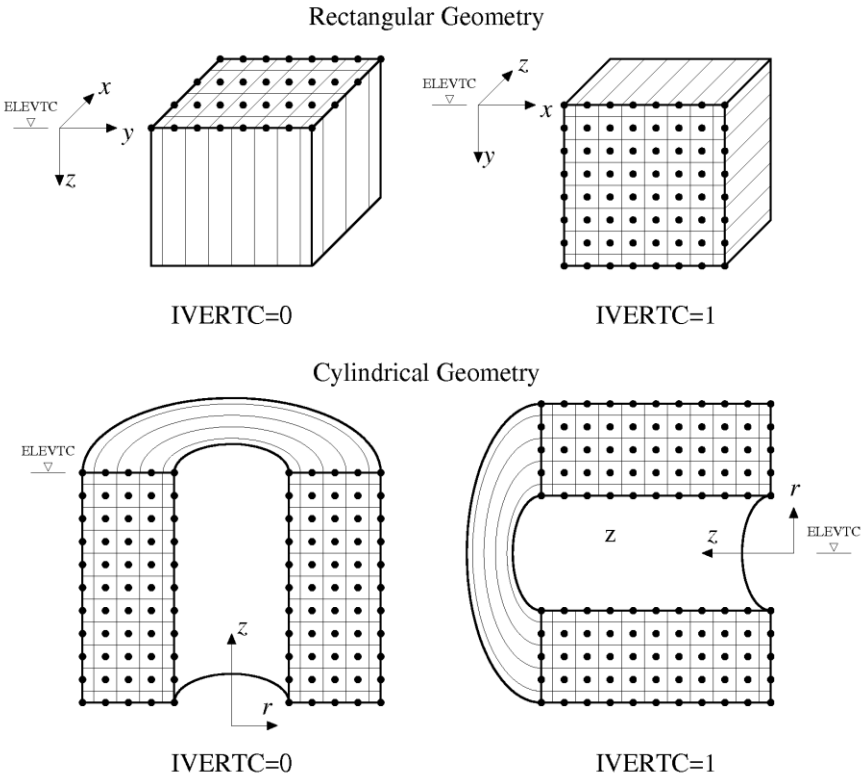


Figure 6-1 Vertical orientations of 2-D Solid Heat Conductors

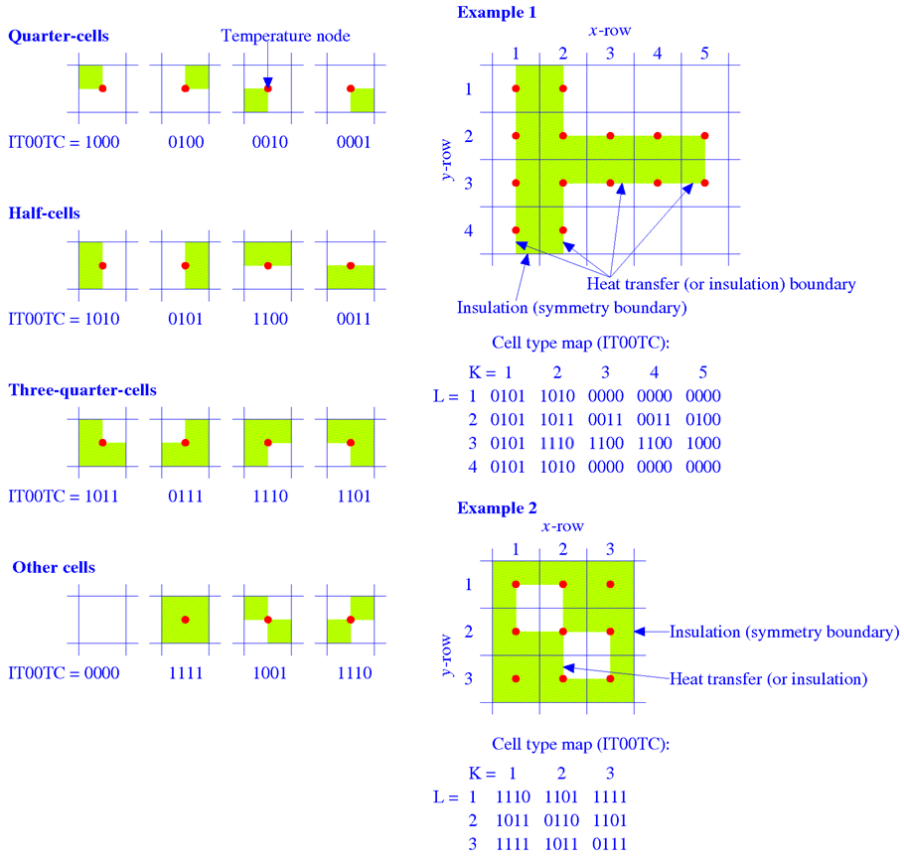


Figure 6-2 Nodalization of a 2-D Solid Heat Conductor.

Nodalization of the interior of a solid heat conductor is shown in Figure 6-2. Each mesh cell may contain a different material. The thermo-physical properties of the materials, such as thermal conductivity, k, specific heat, c_p, and density, ρ, are specified by user input, as functions of temperature. The nodes where temperature is defined are marked in red in Figure 6-2. The adopted modeling concept leads to full-cells (all-solid), half-cells, quarter-cells, three-quarter cells, and empty (all-fluid) cells.

An internal power source may be specified for each solid heat conductor. The space distribution is specified by user input and may vary for each mesh interval (Q_i). The time dependence is given by a user specified tabular function of time or control function of any variables from the program.

At each boundary surface the boundary conditions must be specified. Boundary conditions are defined in the same way as for 1-D Solid Heat Conductors.

The mathematical treatment of 2-D Solid Heat Conductors is described in the sections 6.2 through 6.5. The conduction equation is discussed in section 6.2. Sections 6.3 6.4 6.5 describe the representative boundary conditions, the temperature averaging, and the simplified radiation model, respectively. These models are used for boundary cells of the 2-D Solid Heat Conductors.

6.2 2-D Transient Heat Conduction

The general form of transient heat conduction equation is ([20], chapter 3):

$$\rho \cdot c_p \cdot \frac{\partial T}{\partial t} = \nabla\left(k \cdot \nabla T\right) + q_V$$

T temperature, (K)
t time, (s)
ρ density, (kg/m^3)
c_p specific heat, (J/kg/K)
k thermal conductivity, (W/m-K)
q_V internal heat source per unit volume, (W/m^3)

In case of two-dimensional heat conduction, with the material properties depending on temperature, and the internal heat source being a function of time, the equation is written as:

$$\rho(T) \cdot c_p(T) \cdot \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k(T) \cdot \frac{\partial T}{\partial x} + k(T) \cdot \frac{\partial T}{\partial y} \right) + q_V$$

The above equation is a parabolic partial differential equation. This equation has to be solved numerically with appropriate boundary conditions.

To solve the heat conduction equation, a finite difference version of the equation is constructed. In order to do that, the derivatives are approximated by the finite differences. For the node i,j the time derivative is approximated as:

$$\frac{\partial T_{i,j}}{\partial t} = \frac{T_{i,j} - \overline{T_{i,j}}}{\Delta t}$$

Δt is the time step, and $T_{i,j}$ is the temperature of the node i,j at the beginning of the time step. Similarly the space derivatives may be approximated by:

$$\left(\frac{\partial T}{\partial x}\right)_{i-1,i} = \frac{T_i - T_{i-1}}{x_i - x_{i-1}}$$
$$\left(\frac{\partial T}{\partial y}\right)_{j-1,j} = \frac{T_j - T_{j-1}}{y_j - y_{j-1}}$$

Here a simplified notation is used, where the index that does not change is skipped, i.e. $T_i = T_{i,j}$ in the first equation, $T_j = T_{i,j}$ in the second equation.

Note that in the above approximation the end of time step temperatures are used: T_i, T_{i-1}, T_j, T_{j-1}. After simple transformations the finite difference approximation of the heat conduction equation takes the following form for the interior nodes:

$$V_{i,j}\rho_{i,j}c_{p,i,j}\cdot\frac{T_{i,j}-\overline{T_{i,j}}}{\Delta t}=Q_i+$$
$$+\frac{A_{L,i}}{R_{L,i}}\left(T_{i-1}-T_i\right)+\frac{A_{R,i}}{R_{R,i}}\left(T_{i+1}-T_i\right)+$$
$$+\frac{A_{U,j}}{R_{U,j}}\left(T_{j-1}-T_j\right)+\frac{A_{D,j}}{R_{D,j}}\left(T_{j+1}-T_j\right)$$

$A_{L,i}$ heat transfer area at the left-facing boundary of cell i, (m^2)
$A_{R,i}$ heat transfer area at the right-facing boundary of cell i, (m^2)
$A_{U,i}$ heat transfer area at the up-facing boundary of cell j, (m^2)
$A_{D,i}$ heat transfer area at the down-facing boundary of cell i, (m^2)
V_i volume of cell i, (m^3)
Q_i internal heat power in cell i, (W)
$R_{L,i}$ conduction resistance between the node i–1 and i, (m^2-K/W)
$R_{R,i}$ conduction resistance between the node i and i+1, (m^2-K/W)

The conduction resistances are defined as:

$$left \quad R_{L,i}=\frac{d_{i-1}}{k_{i-1}}+\frac{d_i}{k_i}$$
$$right \quad R_{R,i}=\frac{d_{i+1}}{k_{i+1}}+\frac{d_i}{k_i}$$
$$up \quad R_{U,j}=\frac{d_{j-1}}{k_{j-1}}+\frac{d_j}{k_j}$$
$$down \quad R_{D,j}=\frac{d_{j+1}}{k_{j+1}}+\frac{d_j}{k_j}$$

d_i half-thickness of the cell i in the x-direction, (m)
d_j half-thickness of the cell j in the y-direction, (m)
$k_i k_j$ thermal conductivity of the material in the node i,j (W/m-K)

The temperature in each node is affected by the temperatures of the four neighboring nodes. The temperature nodes are given internal, consecutive numbers k. The equation set may be written shortly in a matrix form:

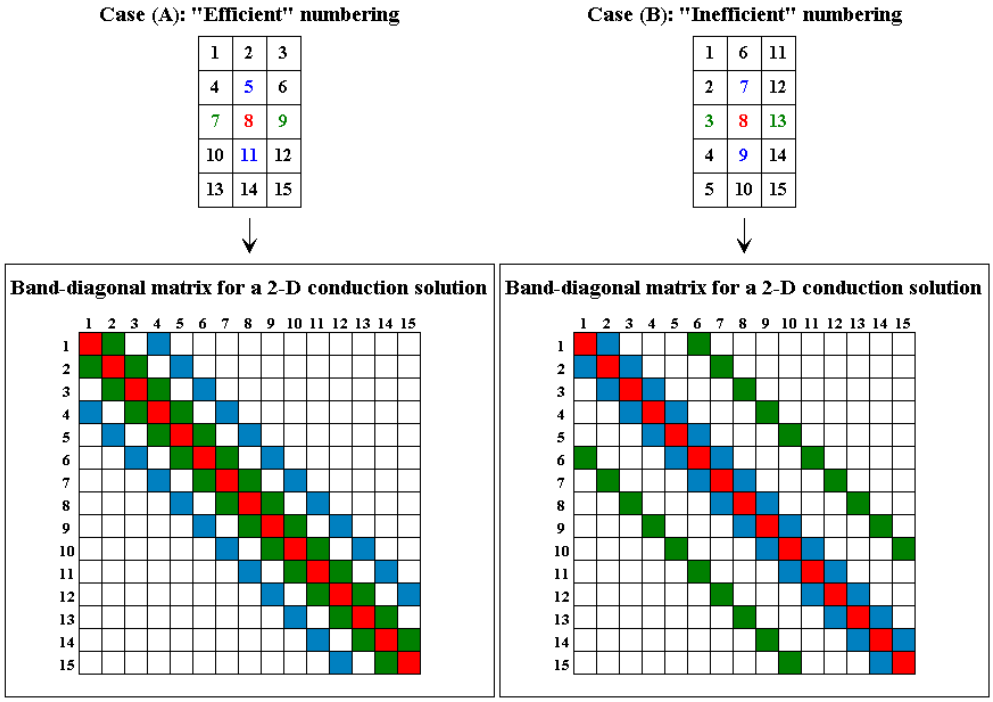
$$AT=B$$

The matrix $\boldsymbol{A}$ is a band-diagonal matrix, because the equations for internal nodes contain five unknown variables: T_{k-N}, T_{k-1}, T_k, T_{k+1}, and T_{k+N}. Therefore the matrix equation may be written as:

$$a_{k-N,k}\cdot T_{k-1}+a_{k-1,k}\cdot T_{k-1}+a_{k,k}\cdot T_k+a_{k+1,k}\cdot T_{k+1}+a_{k+N,k}\cdot T_{k+1}=b_k$$

Here N is the number of cells in either the x-direction or the y-direction, depending which one is smaller.

In order to minimize the band size, the consecutive numbers are assigned to the nodes in such a way that the width of band matrix is minimized. For example, if the number of nodes is 3 in the *x*-direction and 5 in the *y*-direction, then the internal numbering will be as shown below in the Case (A), the "efficient" numbering. Thus node number 8 is affected by nodes 7, 9, 5, and 11. This leads to a band diagonal matrix of the width of $N = 3$. In the case of "inefficient" numbering, Case (B), the band width would be larger (equal to 5), which would lead to a matrix that is more difficult to store and solve. Therefore the input procedures of the TC Package always select the "efficient" numbering scheme for each TC individually, based on its geometry.



The conduction model described above is quite general and accurate in practical applications. Below two example problems are shown, where the model results are compared to analytical solutions of the heat conduction equations.

Figure 6-3 and Figure 6-4 show the 1-D and 2-D solution of the case of step change of temperatures at the boundary of a "semi-infinite" slab. Further discussion and comparison of the results calculated by SPECTRA with the analytical solution are provided in Volume 3 (test case "1-D Transient Conduction Tests for a 2-D Solid Heat Conductor").

Figure 6-5 and Figure 6-6 show 2-D solution and the theoretical solution of the case of step change of temperatures at the boundary of a "semi-infinite" slab. Further discussion is provided in Volume 3 (test case "2-D Transient Conduction Tests for a 2-D Solid Heat Conductor").

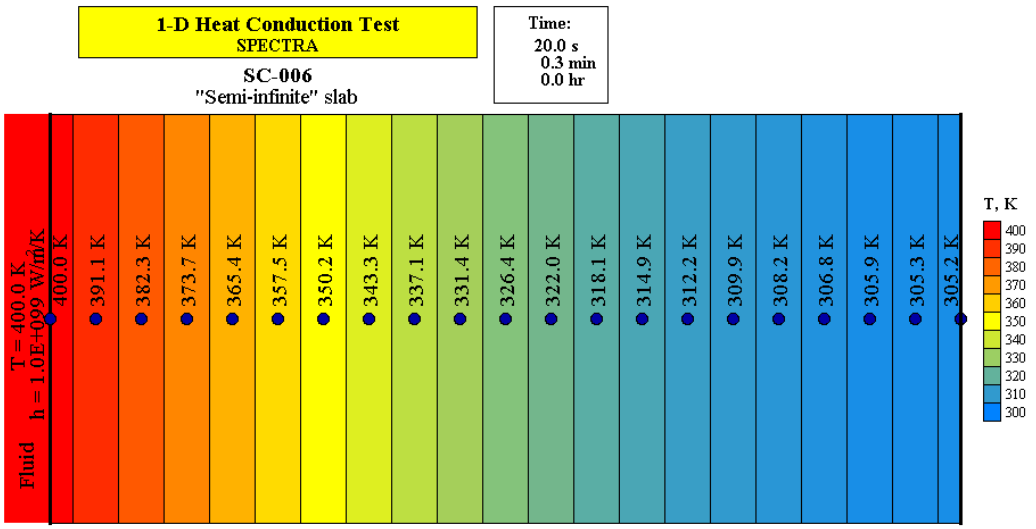


Figure 6-3 Transient conduction in a “semi-infinite” slab, SPECTRA, 1-D, t = 20 s.

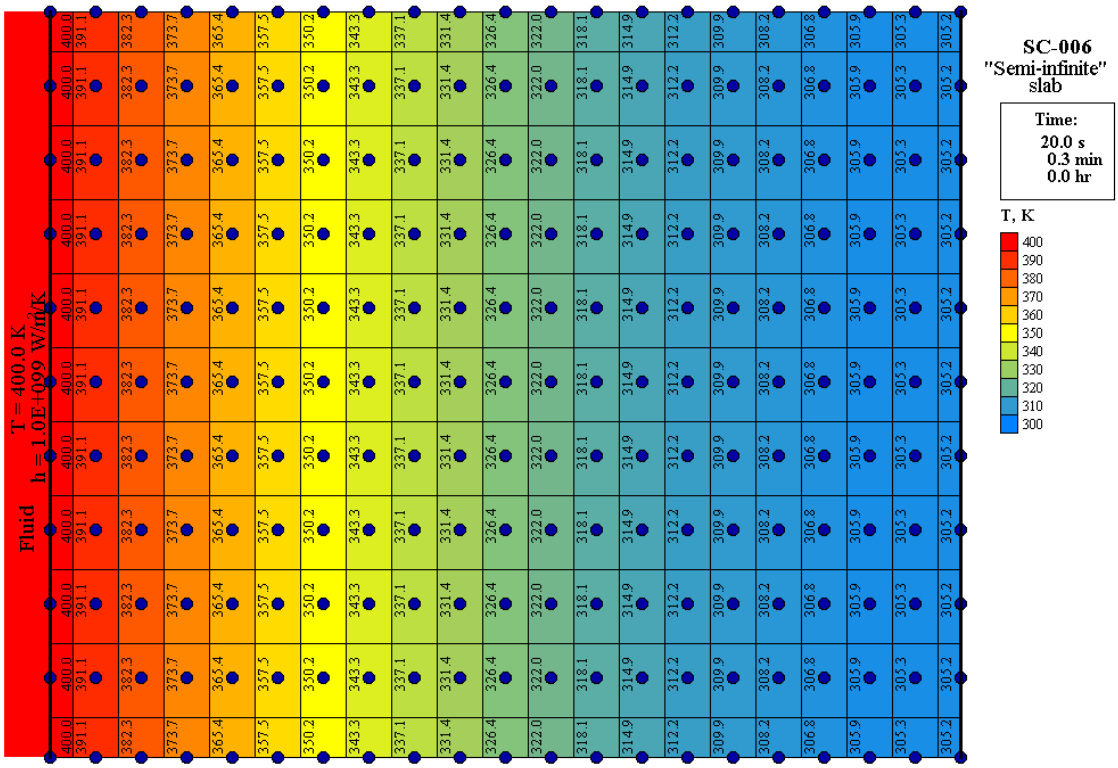


Figure 6-4 Transient conduction in a “semi-infinite” slab, SPECTRA, 2-D, t = 20 s.

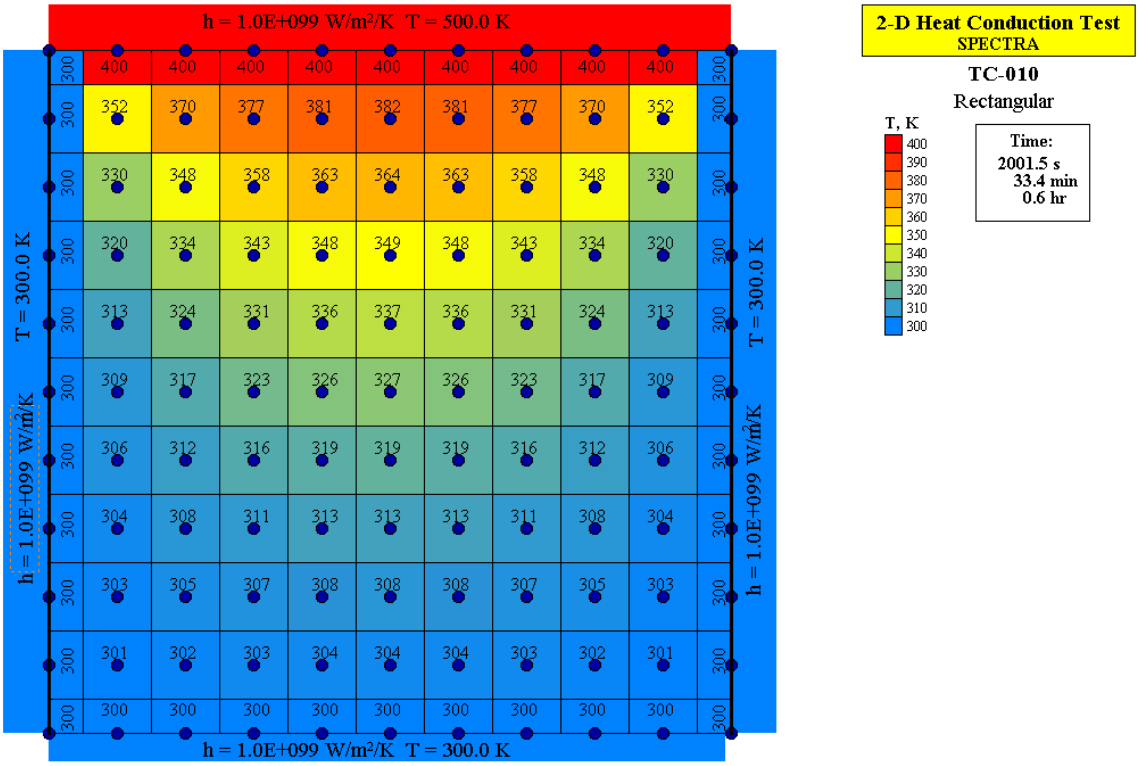


Figure 6-5 2-D conduction test - SPECTRA.

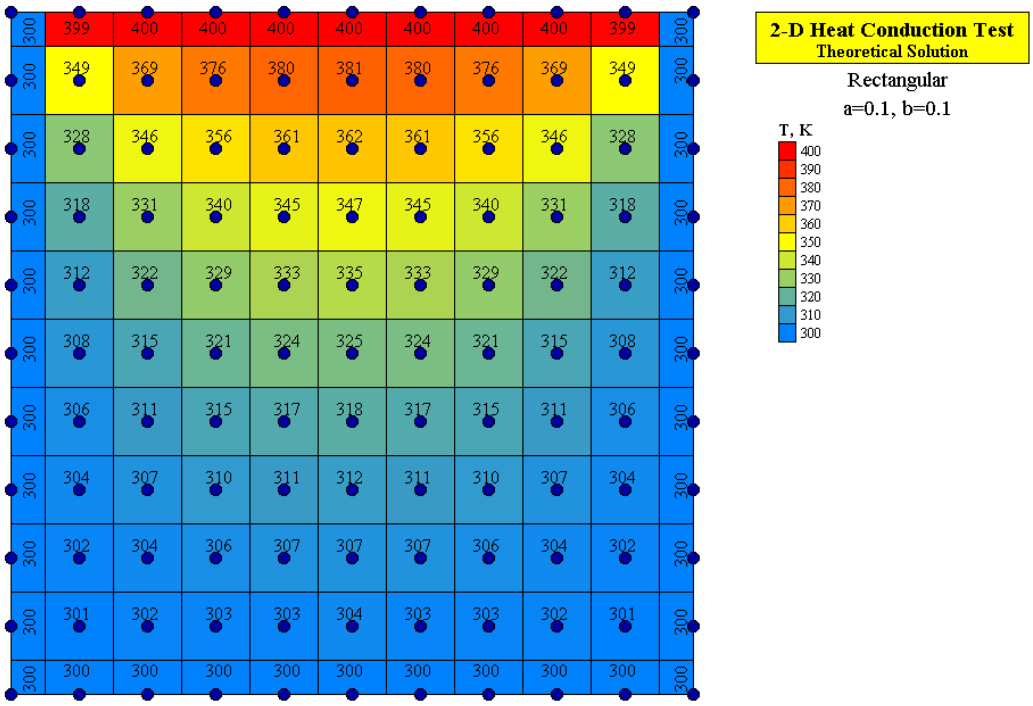


Figure 6-6 2-D conduction test - theoretical solution.

6.3 Representative Boundary Conditions

The representative boundary conditions for the boundary cells of the 2-D Solid Heat Conductors are calculated in exactly the same way as the representative boundary conditions for the left and right cells of the 1-D Solid Heat Conductors. The method is described in section 5.5.

6.4 Temperature Averaging for Heat Exchangers

The temperature averaging is described at 1-D Solid Heat Conductors, section 5.6. Exactly the same model may be applied for the 2-D Solid Heat Conductors.

6.5 Simplified Thermal Radiation Model

The simplified thermal radiation model is described at 1-D Solid Heat Conductors, section 5.9. Exactly the same model may be applied for the 2-D Solid Heat Conductors.

6.6 Structural Failure Model

The structural failure model is described at 1-D Solid Heat Conductors, section 5.11. Exactly the same model may be applied for the 2-D Solid Heat Conductors.

6.7 Gap Conductance Model

The gap conductance model is described at 1-D Solid Heat Conductors, section 5.12. Exactly the same model may be applied for the 2-D Solid Heat Conductors.

6.8 MCCI Model

Numerically, the Molten Core - Concrete Interactions (MCCI) is a part of the TC Package; the MCCI equations are solved on the 2-D network of the 2D Solid Heat Conductor. However, since MCCI is a separate subject, and quite broad in itself, it is described in a separate chapter, namely Chapter 13.

7 Basic Heat and Mass Transfer Package

The Heat and Mass Transfer Package contains all the models and correlations which are used to calculate heat and mass transfer rates. The models are used by the Solid Heat Conductor Package, to calculate boundary conditions for the solid heat conductors, in particular to calculate:

- Wall-atmosphere heat and mass transfer for those solid heat conductor surfaces which are in contact with atmosphere
- Wall-pool heat and mass transfer for those solid heat conductor surfaces which are in contact with pool

The models are also used by the Control Volume Package, to calculate interphase heat and mass transfer rates, in particular:

- Heat and mass transfer at the pool-atmosphere interphase for those control volumes which contain pool
- Heat and mass transfer at the droplet-atmosphere interphase for those control volumes which contain droplets
- Heat and mass transfer at the bubble-pool interphase for those control volumes which contain bubbles

The description of the Heat and Mass Transfer package is provided in this section. In the first part individual correlations are described. This part contains a description of the correlations used for the following conditions:

- Natural convection (section 7.1.1)
- Forced convection (section 7.1.2)
- Condensation (section 7.1.4)
- Nucleate boiling (section 7.1.5)
- Critical heat flux (section 7.1.6)
- Film boiling (section 7.1.9)
- Transition boiling (section 7.1.10)
- Leidenfrost transition (section 7.1.11)
- Heat transfer to two-phase flow (section 7.1.13)
- Non-equilibrium mass transfer (section 7.1.14)

The next part contains a description of the logic to calculate, based on the correlations described above, the heat and mass transfer for the following situations (Figure 7-1):

- Wall (SC/TC)-atmosphere heat and mass transfer, W-A, (section 7.2.1)
- Wall (SC/TC)-pool heat and mass transfer, W-P, (section 7.2.2)
- Pool-atmosphere heat and mass transfer, P-A, (section 7.2.3)
- Droplet-atmosphere heat and mass transfer, D-A, (section 7.2.4)
- Bubble-pool heat and mass transfer, B-P, (sections 7.2.5, 7.2.6)

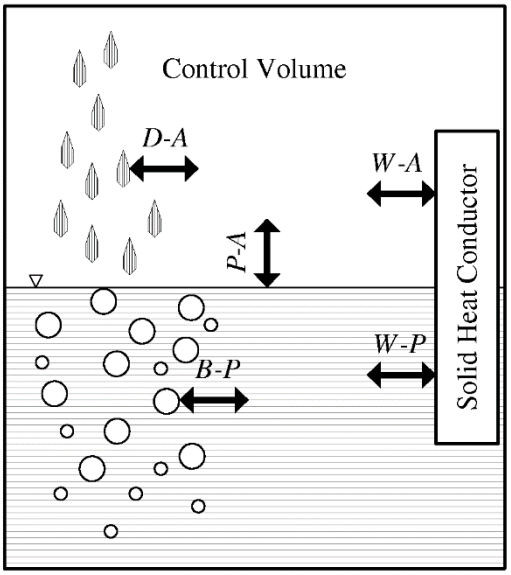


Figure 7-1 Heat and mass transfer processes in a Control Volume

7.1 Heat and Mass Transfer Correlations

7.1.1 Natural Convection

The natural convection model contains equations for three geometry types: rectangular (flat plates), cylindrical, and spherical. The convective heat transfer coefficient is obtained from:

$$h_{conv} = Nu \cdot \frac{k}{D}$$

where *Nu* is the Nusselt number, *k* is the thermal conductivity, and *D* is a characteristic dimension. For each geometry type different configurations of the surface (horizontal, vertical, inside, outside) are considered. The equations used to calculate the Nusselt number for natural convection are discussed below for each of these cases.

- *Vertical surfaces (vertical walls and vertical cylinders)*

  In this case the characteristic dimension is the plate height. Three ranges of (*Gr*·*Pr*) (Grashof number times Prandtl number) are distinguished:

  (*Gr*·*Pr*) < 10^4, [23], [24]:

  For this range the tabulated values, recommended by McAdams and Holman, are used. The values of the "recommended curve" are shown in Table 7-1. Linear interpolation is performed, using $\log_{10}$(*Gr*·*Pr*).

Table 7-1 Recommended curve for vertical plates

$(Gr \cdot Pr)$	10^0	10^1	10^2	10^3	10^4
Nu	1.44	1.90	2.63	3.89	6.03 ($\rightarrow$5.9)

Laminar range, $10^4 < (Gr \cdot Pr) < 1.78 \cdot 10^9$, [23], [24]:

$$Nu = 0.59 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

Turbulent range, $(Gr \cdot Pr) > 1.78 \cdot 10^9$, [24]:

$$Nu = 0.10 \cdot (Gr \cdot \mathrm{Pr})^{1/3}$$

The above formula is recommended by Holman [24]. McAdams [23] recommends a slightly different equation, with the value of the constant equal to 0.13 rather than 0.10. The background for selecting the equation from [24] is that it gives a somewhat better agreement with a number of available experimental data (see Volume 3).

To provide a smooth equation for the whole range of $(Gr \cdot Pr)$ the value of the Nusselt number for the last point of the "recommended curve", was changed from 6.03 to 5.9; the transition point between the laminar and turbulent equation was defined at $1.78071 \cdot 10^9$, rather than 10^9. The recommended curve, shown in Table 7-1, has been extrapolated down to the values of $(Gr \cdot Pr)$ of 10^{-4}. This was done using the curve for horizontal walls, shown in Table 7-2. For $(Gr \cdot Pr) = 1.0$ the vertical curve gives $Nu = 1.44$, while horizontal $Nu = 1.08$. The ratio of 1.44/1.08 was applied to all values $(Gr \cdot Pr) < 1.0$ from the horizontal curve, to obtain the extrapolated values for the vertical curve. The extrapolated values are:

$(Gr \cdot Pr) = 10^{-4}$ $\quad Nu = 0.65$
$(Gr \cdot Pr) = 10^{-3}$ $\quad Nu = 0.73$
$(Gr \cdot Pr) = 10^{-2}$ $\quad Nu = 0.88$
$(Gr \cdot Pr) = 10^{-1}$ $\quad Nu = 1.12$

- *Horizontal plates - "open" for natural circulation*

A surface is considered to be "open" for natural circulation when a hot surface is facing upwards, or a cold surface is facing downwards. The characteristic dimension is the plate width. Three ranges of $(Gr \cdot Pr)$ numbers are distinguished:

$(Gr \cdot Pr) < 10^4$
For this range no recommended equation or table was found in literature. The "recommended curve" for horizontal cylinders (Table 7-2) was applied for this range. The background of this choice is the fact that the laminar flow equation for this case is very similar to the laminar flow equation for the horizontal cylinder case.

Laminar range, $10^4 < (Gr \cdot Pr) < 1.085 \cdot 10^7$, [23]:

$$Nu = 0.54 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

Turbulent range, $(Gr \cdot Pr) > 1.085 \cdot 10^7$, [23]:

$$Nu = 0.14 \cdot (Gr \cdot \mathrm{Pr})^{1/3}$$

To provide a smooth equation for the whole range of ($Gr \cdot Pr$) number, the Nusselt number for the last point of the "recommended curve", (Table 7-2), was changed from 5.37 to 5.4, and the transition point between laminar and turbulent equation was changed from $2.0 \cdot 10^9$ to $1.085 \cdot 10^9$.

- *Horizontal plates - "closed" for natural circulation*

  A surface is considered to be "closed" for natural circulation when a hot surface is facing downwards, or a cold surface is facing upwards. The situation is similar to (small) enclosed cylindrical or vertical spaces, described below. The characteristic dimension is the plate width. Two ranges of ($Gr \cdot Pr$) numbers are distinguished:

  $(Gr \cdot Pr) < 10^4$
  For this range no recommended equation or table was found in literature. It was assumed that the values of one half of the horizontal wall "recommended curve" (Table 7-2) could be used for this case. The background of this choice is the fact that the laminar flow equation for this case gives a twice smaller Nusselt number than the laminar flow equation for the "open" surface.

  $(Gr \cdot Pr) > 10^4$:

$$Nu = 0.27 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

  The function defined above is smooth, since the last point of the recommended curve Nu=5.4/2=2.7 is the same as the value obtained from the correlation.

- *Horizontal cylinders*

  In this case the characteristic dimension is the cylinder diameter. Three ranges of ($Gr \cdot Pr$) are distinguished:

  $(Gr \cdot Pr) < 10^4$, [23], [24]:

  For this range the tabulated values, recommended by McAdams and Holman, are used. The values of the "recommended curve" are shown in Table 7-2. Linear interpolation is performed, using $\log_{10}(Gr \cdot Pr)$.

Table 7-2 Recommended curve for horizontal cylinders.

$(Gr \cdot Pr)$	10^{-4}	10^{-3}	10^{-2}	10^{-1}	10^{0}	10^{1}	10^{2}	10^{3}	10^{4}
Nu	0.49	0.55	0.661	0.841	1.08	1.51	2.11	3.16	5.37

Laminar range, $10^4 < (Gr \cdot Pr) < 2.11 \cdot 10^7$, [23], [24]:

$$Nu = 0.53 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

Turbulent range, $(Gr \cdot Pr) > 2.11 \cdot 10^7$, [24]:

$$Nu = 0.13 \cdot (Gr \cdot \mathrm{Pr})^{1/3}$$

To provide a smooth equation for the whole range of $(Gr \cdot Pr)$ number, the value of the Nusselt number for the last point of the "recommended curve", (Table 7-2), was changed from 5.37 to 5.3 (in contrast to the value 5.4, which is applied for the horizontal "closed" plates, described above), and the transition point between the laminar and turbulent equation was defined at the intersection of those lines, at $(Gr \cdot Pr) = 2.11 \cdot 10^7$ instead of 10^9.

- *Spheres*

  The characteristic dimension is the sphere diameter. The formula applicable for outside surface of spheres is [24]:

$$Nu = 2 + 0.43 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

- *Enclosed cylindrical spaces*

  The formula applicable for vertical or horizontal cylindrical enclosures was taken from [24]. The characteristic dimension is the cylinder length, with the limit of: $L/D < 2.0$. Only one formula is recommended for the whole range, but to be consistent with other formulations, two ranges have been distinguished

  $(Gr \cdot Pr) < 10^4$, [23], [24]:

  For this range the horizontal cylinders "recommended curve" is used (Table 7-2).

  $(Gr \cdot Pr) > 10^4$, [24]:

$$Nu = 0.55 \cdot (Gr \cdot \mathrm{Pr})^{1/4}$$

  To provide a smooth equation for the whole range of $(Gr \cdot Pr)$ number the value of the Nusselt number for the last point of the "recommended curve", (Table 7-2), was changed from 5.37 to 5.5.

- *Enclosed spherical spaces*

  In case of enclosed spherical spaces literature typically gives equations for the effective, or apparent thermal conductivity of the fluid. Those correlations allow to obtain wall-to-wall temperature difference for constant heat flux between two concentric spheres with a small gas space in between. Such equation may be found in [24]: $Nu = 0.228 \cdot (Gr \cdot Pr)^{0.226}$. The MELCOR code [46] uses this equation to calculate a transient heat transfer coefficient on internal sphere surfaces ([46], page HS-RM-39). It is felt however that such correlation is not suitable for general application in calculating the transient wall-to-fluid heat transfer coefficient. Therefore the use of this correlation in SPECTRA is avoided.

For the spherical geometry it is difficult to find an appropriate correlation. It is felt that for typical applications (reactor vessel head, etc) where the radius is large, an "open" configuration correlation is applicable. When the model is used for spheres with a very small radius, two cases are possible:

- A small single sphere is considered. In this case the problem practically does not exist, because there is very little heat exchange and an insulation boundary condition is sufficient.
- Small concentric spheres are considered. In this case a correlation like that shown above might be desired: $Nu=0.228\cdot(Gr\cdot Pr)^{0.226}$. Even then, the approach taken in MELCOR is not quite correct, since this correlation should be applied with the wall-to-wall temperature difference [24], while the code uses it with the wall-to-fluid temperature difference.

In SPECTRA it is assumed that typically spherical conductors will be assumed for large structures and the outer sphere equation is also used on the inner surface of the sphere. Models appropriate for very small enclosures are not implemented.

A user-defined multiplier (input parameters XRBLSC, XRBRSC, XRBBTC - see Volume 2) may be applied for the heat transfer coefficient obtained from the above correlations, for example to obtain a conservative estimation of the heat transfer.

Results obtained from all natural convection correlations presented above are compared in figures Figure 7-2. It is seen in Figure 7-2-left that vertical plates and horizontal plates "open" for natural circulation give approximately the same Nusselt number. Horizontal plates "closed" for natural circulation give roughly 2 ÷ 3 times smaller Nusselt number. Figure 7-2-right shows that in case of low $(Gr\cdot Pr)$ numbers, the Nusselt number is greater in case of heat transfer to spheres than to cylinders.

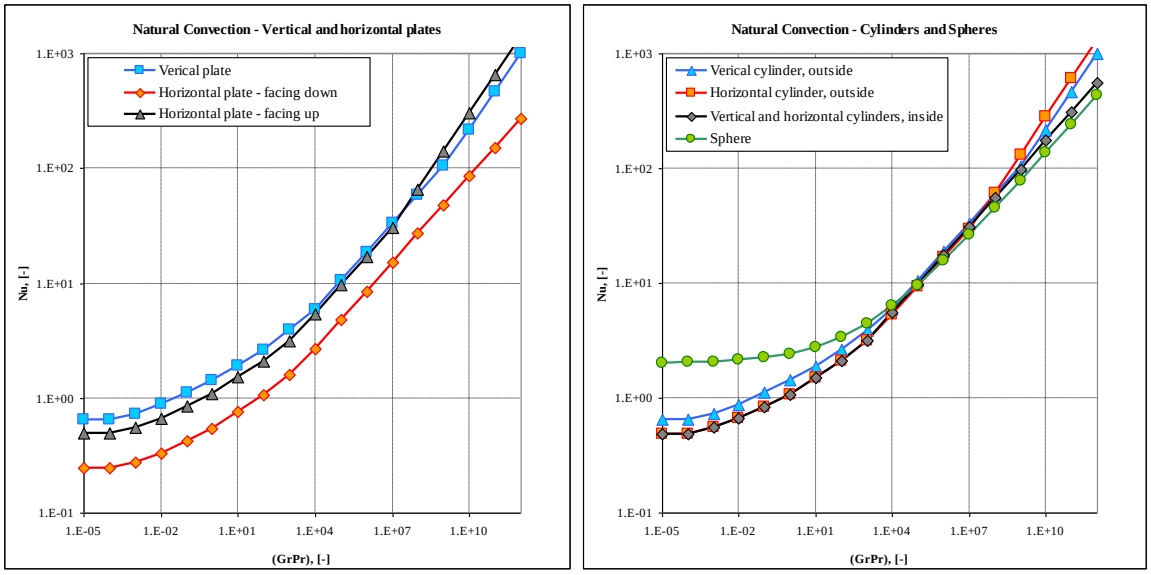


Figure 7-2 Natural convection model, left: flat plates, right: cylinders and sphere

7.1.2 Forced Convection

The forced convection model contains equations for internal and external flows. The equations give the Nusselt number, and are discussed below for each of those cases.

- *Internal flows*

  The equations described in this part are applicable for internal flows, fully developed velocity profile and isothermal wall surface. The characteristic dimension is the hydraulic diameter. Three ranges are distinguished: laminar, transition and turbulent.

  Laminar range: $Re < Re_{lam} = 0.5Re_c$, where Re_c is the crossing point of both laminar and turbulent correlations, $Re_c = (3.656 / (0.023Pr^{0.4}))^{1.25}$. The correlation is (see [20], page 7-20):

$$Nu_{lam} = 3.656$$

  Transition range: $Re_{lam} < Re < Re_{tur} = 2.0Re_c$
  In the transition range the following interpolation is performed, which provides smooth transition from the laminar to the turbulent region.

$$Nu_{tran}(Re) = Nu_{lam}(Re) - \{Nu_{lam}(Re) - Nu_{tur}(Re)\} \cdot \left\{ \frac{Re - Re_{lam}}{Re_{tur} - Re_{lam}} \right\}$$

  Turbulent range: $Re > Re_{tur} = 2.0Re_c$; Dittus-Boelter correlation [22]:

$$Nu_{tur} = 0.023\ Re^{0.8}\ Pr^{0.4}$$

  Experimental data for water shows that in case of internal flow in a rod bundle the heat transfer coefficient is higher than that obtained from the Dittus-Boelter correlation. In SPECTRA a multiplier, X_{RB}, defined in the input deck (input parameters XRBLSC, XRBRSC, XRBBTC - see Volume 2) may be used to account for the rod bundle. Appropriate values for parallel flow and cross-flow may be found in literature. The ratio of pitch over diameter, $X_{RB}=P/D$, is a good approximation for $P/D < 1.6$ [184]. In such case:

$$Nu_{tur} = X_{RB}\ 0.023\ Re^{0.8}\ Pr^{0.4} = \left(\frac{P}{D} \right) 0.023\ Re^{0.8}\ Pr^{0.4}$$

  For tube arrangements other than equilateral triangle pitch, the multiplier is given by:

$$X_{RB} = \left(\frac{P_1 \cdot P_2}{D^2} \right)$$

  Here P_1 and P_2 are the "pitches of the tubes in the bundle" and D is the tube diameter. Note that the rod bundle multiplier may be different for different fluids. For example, for the liquid metals even the trend may be different (decrease of the heat transfer coefficient with increase of P/D - see section 7.1.3).

- *External flows around cylinders*

  The characteristic dimension for this case is the cylinder diameter. Three ranges are distinguished: laminar, transition and turbulent.

  Laminar range: $Re < Re_{lam} = 10^5$, Holman correlation [24] :

$$Nu_{lam} = (0.35 + 0.56\ Re^{0.52})\ Pr^{0.3}$$

  Transition range: $Re_{lam} < Re < Re_{tur}$:

  In the transition range the following interpolation is performed, which provides smooth transition from laminar to turbulent regions.

$$Nu_{tran}(Re) = Nu_{lam}(Re) - \{Nu_{lam}(Re) - Nu_{tur}(Re)\} \cdot \left\{ \frac{Re - Re_{lam}}{Re_{tur} - Re_{lam}} \right\}^{1/2}$$

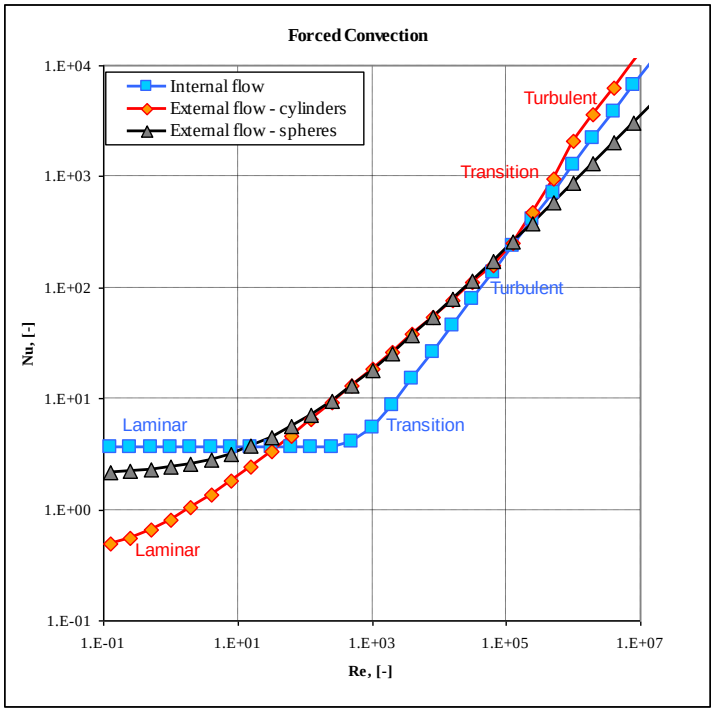
  Turbulent range: $Re > Re_{tur} = 10^6$; Bennet correlation [25].

$$Nu_{tur} = 0.037\ Re^{0.8}\ Pr^{1/3}$$

- *External flows around spheres.*

  The data available on forced convective heat transfer from the spheres has been brought together by Whitaker [26] to develop a single equation for gases and liquids flowing past the spheres. The characteristic dimension is the sphere diameter. The equation is [24]:

$$Nu = 2 + (0.4\ Re^{1/2} + Re^{2/3})\ Pr^{0.4}$$



where the viscosity change near the wall has been neglected. The above equation is valid for the laminar range, for Reynolds numbers up to $Re = 0.8 \cdot 10^5$ [24]. Since no equation was found in the literature for the turbulent region, the Whitaker formula is used for the whole range of Reynolds numbers.

Results obtained from all forced convection correlations are compared in Figure 7-3. The selection of a correlation is performed for each Solid Heat Conductor surface via input data. Default selections are given in the SPECTRA User's Guide.

Figure 7-3 Forced convection model.

7.1.3 Forced Convection Correlations for the Alternative Fluid

The alternative fluid is intended for liquid metal fluid. A generally applicable correlation is available in SPECTRA that may be used to represent forced convective heat transfer. The correlation is:

$$Nu = \left[A_0 + A_1 \cdot Re^{B_1} \cdot Pr^{C_1} \cdot \left(1 + A_2 \cdot Re^{B_2} \cdot Pr^{C_2}\right)^{D_2} \cdot \left(1 + A_3 \cdot Re^{B_3} \cdot Pr^{C_3}\right)^{D_3}\right] \times \left(\frac{\mu_f}{\mu_w}\right)^x$$

Here Nu is the Nusselt number, Re is the Reynolds number, Pr is the Prandtl number, μ_f is the fluid viscosity obtained at the bulk fluid temperature, μ_w is the fluid viscosity obtained at the wall surface temperature. The constants A_0, A_1, A_2, A_3, B_1, B_2, B_3, C_1, C_2, C_3, x, are defined by the user.

The correlation is defined in the records 843XXX (see Volume 2). These records define a single heat transfer correlation, valid for the entire range of Reynolds numbers. If multiple correlations are needed, then a set of up to 10 correlations may be defined in the records 843YYY.

The rod bundle multiplier, depending on the *P/D* ratio may be applied, the same as in the internal flow configuration, described in section 7.1.2.

An overview of heat transfer correlations is presented for example in references [27], [196]. Table 7-3 and Table 7-4 show correlations applicable for constant wall temperature, T_w = const., and constant wall heat flux, q_w = const.

Table 7-3 Correlations applicable for liquid metals, constant wall temperature [27]

Investigator	Correlation	Remark
[Gilliland, *et al*., 1951]	$Nu = 3.3 + 0.02\,Pe^{0.8}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, rather speculative description of the boundary conditions.
[Sleicher and Tribus, 1957]	$Nu = 4.8 + 0.015\,Re^{0.91} Pr^{1.21}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within a bandwidth of -33% and +19.5% of [Notter & Sleicher, 1972].
[Hartnett and Irvine, 1957]	$Nu = Nu_{slug} + 0.015\,Pe^{0.8}$ with $Nu_{slug} = 5.78$ assuming a slug velocity profile	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are 40% below the [Notter & Sleicher, 1972] correlation
[Azer and Chao, 1961]	$Nu = 5 + 0.05\,Re^{0.77} Pr^{1.02}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^5$, the predictions are within +14.2% and -18.6% of the [Notter & Sleicher, 1972] correlation.
[Notter and Sleicher, 1972]	$Nu = 4.8 + 0.0156\,Re^{0.85} Pr^{0.93}$	$0.004 \le Pr \le 0.1$ and $10^4 \le Re \le 10^6$, the predictions are based on a numerical analysis and the minimum error of the cloud of experimental data.
[Chen and Chiou, 1981]	$Nu = 4.5 + 0.0156\,Re^{0.85} Pr^{0.86}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within a bandwidth of +36% and -2% of [Notter & Sleicher, 1972].

Table 7-4 Correlations applicable for liquid metals, constant wall heat flux [27]

Investigator	Correlation	Remark
[Lyon, 1949, 1951] [Subbotin, *et al.*, 1962]	$Nu = 5 + 0.025\,Pe^{0.8}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within +33% and -6.5% of [Notter & Sleicher, 1972].
[Lubarski and Kaufman, 1955]	$Nu = 0.625\,Pe^{0.4}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 10^5$, significant underprediction of [Notter& Sleicher, 1972] by -43%.
[Sleicher and Tribus, 1957]	$Nu = 6.3 + 0.016\,Re^{0.91}Pr^{1.21}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within a bandwidth of -32% and +26% of [Notter & Sleicher, 1972].
[Hartnett and Irvine, 1957]	$Nu = Nu_{slug} + 0.015\,Pe^{0.8}$ with $Nu_{slug} = 8$ assuming a slug velocity profile	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are 44% below the [Notter & Sleicher, 1972] correlation
[Dwyer, 1963]	$Nu = 7 + 0.025\left[RePr - \frac{1.82Re}{(\varepsilon_M/\nu)_{max}^{0.14}}\right]^{0.8}$ where $(\varepsilon_M/\nu)_{max} = 0.037Re\sqrt{f}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within +31% and -6.5% of the [Notter & Sleicher, 1972] correlation.
[Skupinski, *et al.*, 1965]	$Nu = 4.82 + 0.0185Pe^{0.827}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within +22% and -18% of the [Notter & Sleicher, 1972] correlation.
[Notter and Sleicher, 1972]	$Nu = 6.3 + 0.0167\,Re^{0.85}Pr^{0.93}$	$0.004 \le Pr \le 0.1$ and $10^4 \le Re \le 10^6$, the predictions are based on a numerical analysis and the minimum error of the cloud of experimental data
[Chen and Chiou, 1981]	$Nu = 5.6 + 0.0165\,Re^{0.85}Pr^{0.86}$	$0 \le Pr \le 0.1$ and $10^4 \le Re \le 5{\cdot}10^6$, the predictions are within a bandwidth of +34% and -7% of [Notter & Sleicher, 1972]
[Lee,1983]	$Nu = 3.01\,Re^{0.0833}$	$0.001 \le Pr \le 0.02$ and $5{\cdot}10^3 \le Re \le 10^5$, where its predictions are within +25% and -44% of [Notter & Sleicher, 1972].

As a generally applicable correlation for internal flows the Notter and Sleicher correlation may be recommended. For external flows three correlations were recommended:

- Rectangular geometry:

  Two correlations are mentioned, first for a single plate and second for flow between parallel plates [27], [28]:

$$Nu = 0.565 \cdot Re^{0.5} \cdot Pr^{0.5}$$

$$Nu = 5.6 + 0.019 \cdot Re^{0.775} \cdot Pr^{0.775}$$

  The first correlation is defined by: $A_1 = 0.565$, $B_1 = 0.5$, $C_1 = 0.5$, the remaining coefficients are zero. The second correlation is defined by: $A_0 = 5.6$, $A_1 = 0.019$, $B_1 = 0.775$, $C_1 = 0.775$, the remaining coefficients are zero.

- Cylindrical geometry:

  Two correlations are mentioned, first for a single cylinder, Incropera [194], and second for flow in the tube bundle, Mikityuk [195]:

$$Nu = 0.3 + 0.62 \cdot Re^{0.5} \cdot Pr^{1/3} \cdot \left[1 + \left(\frac{Re}{282{,}000}\right)^{5/8}\right]^{4/5} \Big/ \left[1 + \left(\frac{0.4}{Pr}\right)^{2/3}\right]^{1/4}$$

$$Nu = 0.047 \cdot \left\{1 - \exp\left[-3.8 \cdot (P/D - 1)\right]\right\} \cdot \left(Pe^{0.77} + 250\right)$$

  The first correlation is defined by: A_0 = 0.3, A_1 = 0.62, B_1 = 0.5, C_1 = 0.333, A_2 = $(1/282{,}000)^{5/8} = 3.923\times10^{-4}$, B_2 = 0.625, D_2 = 0.8, $A_3 = (0.4)^{2/3}$ = 0.543, C_3 = –0.667, D_3 = –0.25, the remaining coefficients are zero. The second correlation is defined by: A_0 = 0.047×250 = 11.75, A_1 = 0.047, B_1 = 0.77, C_1 = 0.77, the remaining coefficients are zero. The term dependent on the pitch-over-diameter ratio, *P/D*, may be taken into account using the multiplier, $f_{L/D}$ (see Volume 2, words XRBLSC, XRBRSC, XRBTSC), defined as:

$$f_{L/D} = 1 - \exp\left[-3.8 \cdot (P/D - 1)\right]$$

  Note that the heat transfer decreases with *P/D* ratio. For example, if *P/D* = 1.5, then $f_{L/D}$ = 0.85. This is contradictory to the data for water - see section 7.1.2.

Figure 7-4 shows comparison of all correlations, performed for *Pr* = 0.01 and $10 < Re < 10^9$, and $f_{L/D}$ = 0.85. The following remarks can be made:

(1) For the turbulent region ($Re > 10^4$) the cylindrical geometry correlations give the highest heat transfer (*Nu* number).

(2) For the laminar and transition region ($Re < 10^4$) the single plate and single cylinder correlations give $Nu \rightarrow \sim0$ for $Pe \rightarrow 0$. These correlations are valid for the turbulent flows and should not be applied in the laminar flow range. It is however easy to modify these correlations by modifying the value of A_0. In the laminar range *Nu* is approximately constant and equal to between 3.66 and about 5. The value of A_0 was set to 3.66 in both correlations. The results are shown in Figure 7-5. The modified correlations are valid for the whole range of Reynolds numbers. In the turbulent range the value of A_0 is very quickly becoming negligible, since the term with *Re* is increasing very fast. Therefore in the turbulent range the modification of A_0 is invisible, i.e. the lines are practically identical in Figure 7-4 and Figure 7-5 for $Re > 10^4$.

(3) The internal flow correlation (Notter and Sleicher, 1972) may be used for all geometries in a simple approach. The geometry-dependent correlations should be used in a more detailed approach. All coefficients are listed in Table 7-5.

Table 7-5 Recommended heat transfer correlation coefficients for liquid lead

Case	A_0	A_1	B_1	C_1	A_2	B_2	C_2	D_2	A_3	B_3	C_3	D_3
Plate	3.66	0.565	0.5	0.5	-	-	-	-	-	-	-	-
Plates	5.60	0.019	0.775	0.775	-	-	-	-	-	-	-	-
Cylinder	3.66	0.620	0.5	0.333	$3.923\cdot10^{-4}$	0.625	0.0	0.8	0.543	0.0	–0.667	–0.25
T.bundle	11.75	0.047	0.77	0.77	-	-	-	-	-	-	-	-
Internal	4.8	0.0156	0.85	0.93	-	-	-	-	-	-	-	-

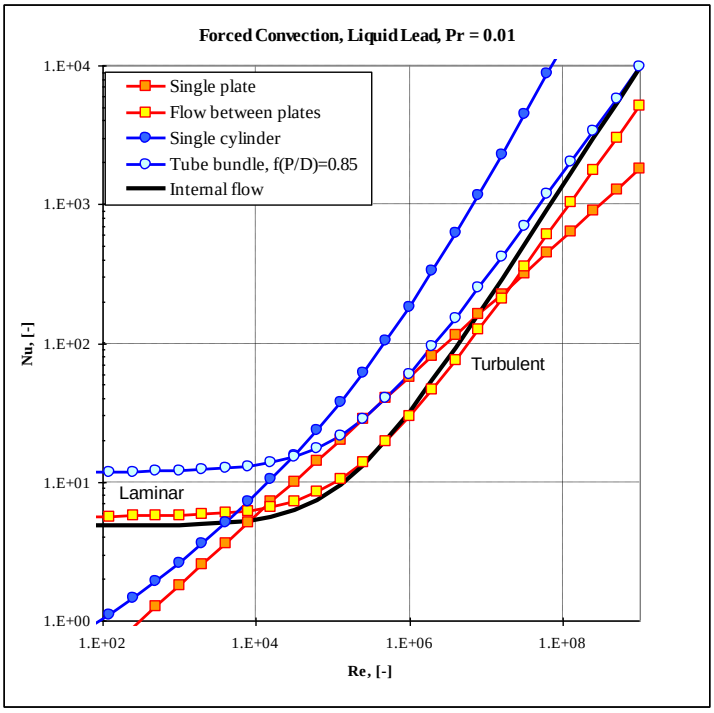


Figure 7-4 Comparisons of heat transfer correlations

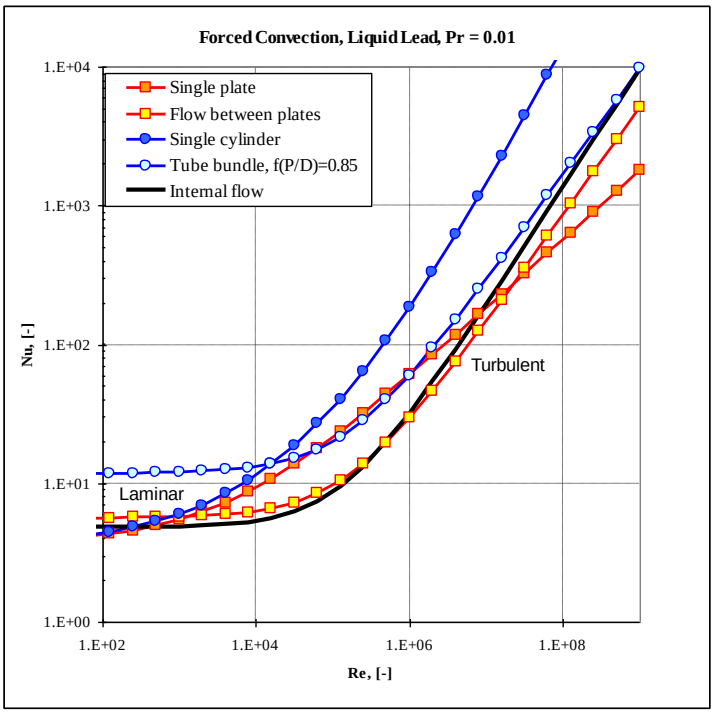


Figure 7-5 Heat transfer correlations modified for laminar range

7.1.4 Condensation

The condensation model consists of the following three sub-models:

- Pure steam condensation model
- Influence of noncondensables (non-condensable gas degradation factor)
- Influence of interfacial shear (shear enhancement factor).

The description below is divided into three parts, each part describes one of the sub-model, mentioned above.

- Pure steam condensation model

  In the theoretical model condensation is assumed to be limited by conduction through liquid film [20]:

$$h_{cond} = \frac{k_{liq}}{\delta_{cond}}$$

  where k_{liq} is thermal conductivity of the condensate, and δ_{cond} is the condensate film thickness. The condensate flow is calculated based on the static laminar flow model. Condensate transport through several, vertically stacked Solid Heat Conductors may be modelled, however not in a dynamic, but quasi static way. In such case the condensate is deposited on the surface on the lower SC/TC member of the stack. Otherwise the condensate either drains directly to the pool, or is converted to droplets, and suspended in the atmosphere of CV, at the elevation of the lower edge of a Solid Heat Conductor.

  The condensate film thickness and consequently the heat transfer coefficient depend on the geometry. The program contains models appropriate for several geometries (Figure 7-6): vertical wall or tube, internal condensation inside horizontal, or nearly horizontal tubes, tube banks, horizontal surfaces. The following correlations are applied:

  - *Vertical wall*, modified Nusselt laminar flow model [130]:

$$h_{cond} = 0.943\left(\frac{g\rho_{liq}(\rho_{liq}-\rho_{vap})k_{liq}^3 h_{l-v}}{D\eta_{liq}\Delta T_{sat}}\right)^{0.25}$$

  g gravity acceleration, (m/s^2)
  ρ density, (kg/m^3)
  k thermal conductivity, (W/m-K)
  η viscosity, (kg/m-s)
  h_{l-v} evaporation enthalpy, (J/kg)

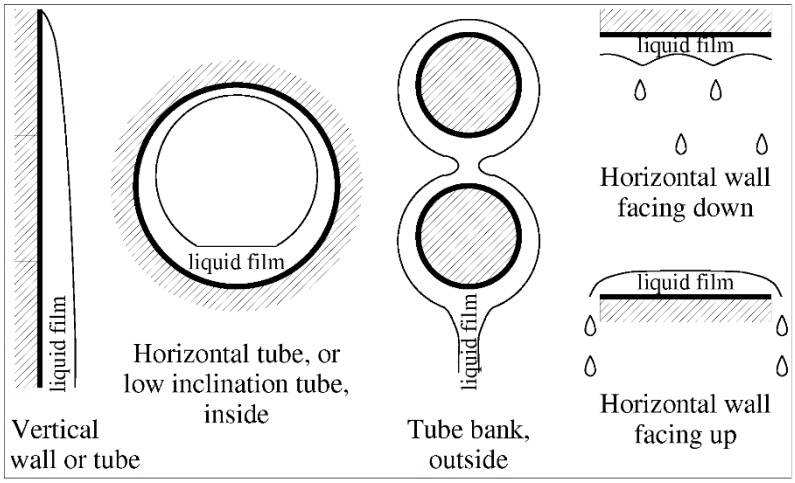


Figure 7-6 Geometries available for the condensation

ΔT_{sat} wall subcooling (K) (wall temperature minus saturation temperature)
D characteristic dimension for condensation (input parameter - see Volume 2)
Subscripts *liq* and *vap* signify liquid and vapor respectively.

- *Horizontal tubes or low inclination tubes, inside*, based on [20], with Chato [131] correction for inclination:

$$h_{cond} = 0.555\left(\frac{g\rho_{liq}(\rho_{liq}-\rho_{vap})k_{liq}^3 h_{l-v}}{D\eta_{liq}\Delta T_{sat}}\right)^{0.25}\left(1+\frac{\theta^\circ}{100}\right)$$

θ° tube inclination in degrees ($\theta^\circ < 20^\circ$)

- *Horizontal tubes, tube banks, outside*, modified Nusselt model [130]:

$$h_{cond} = 0.728\left(\frac{g\rho_{liq}(\rho_{liq}-\rho_{vap})k_{liq}^3 h_{l-v}}{N_{row}D\eta_{liq}\Delta T_{sat}}\right)^{0.25}$$

N_{row} number of vertical rows

- *Horizontal surfaces, facing down*, Gerstmann and Griffith [132]:

$$h_{cond} = \frac{k_{liq}}{D_x}\frac{0.9\tau^{-1/6}}{1+1.1\tau^{1/6}}$$

where:

$$D_x = \left(\frac{\sigma}{g(\rho_{liq} - \rho_{vap})} \right)^{0.5}$$

$$\tau = \frac{k_{liq}\eta_{liq}}{g\rho_{liq}(\rho_{liq} - \rho_{vap})h_{l-v}} \frac{\Delta T_{sat}}{D_x^3}$$

σ is the surface tension N/m. Note that this equation does not need a user input characteristic dimension. The characteristic dimension, D_x, is in this case determined by fluid properties.

- *Horizontal surfaces, facing up,* Chato model [131]:

$$h_{cond} = \frac{k_{liq}}{D}$$

Here D is the film thickness, equal to the equilibrium thickness of water layer which is formed while there is free fall of water at the edges of the surface. Note that if there is no free fall at the edges, condensation will stop almost immediately because the surface will be covered by pool, and the wall-atmosphere heat transfer will not be calculated. The default value of D is 10^{-3} m, but may be redefined for each Solid Heat Conductor in input data.

- Influence of noncondensables

The influence of non-condensable gases is taken into account by introducing a non-condensable degradation factor, defined as the ratio of actual local condensation heat transfer coefficient to the pure steam local condensation heat transfer coefficient, f_{NC}, versus mass fraction, X, of noncondensable gases. Three correlations are available to calculate f_{NC}: the KSP correlation, the Ogg correlation, and the Modified Ogg correlation. Each of the correlations mentioned above gives the value of degradation factor for a steam-air mixture, $f_{NC}(H_2O\text{-}air)$, and a steam-helium mixture, $f_{NC}(H_2O\text{-}He)$. These correlations are shown below.

- *Kuhn-Schrock-Petersen (KSP) correlations* [133]

$$f_{NC}(H_2O - air) = \begin{cases} 1 - 2.601 X_a^{0.708} & for \quad X_a < 0.1 \\ 1 - X_a^{0.292} & for \quad X_a > 0.1 \end{cases}$$

$$f_{NC}(H_2O - He) = \begin{cases} 1 - 35.81 X_{He}^{1.074} & for \quad X_{He} < 0.01 \\ 1 - 2.09 X_{He}^{0.457} & for \quad 0.01 < X_{He} < 0.10 \\ 1 - X_{He}^{0.137} & for \quad 0.10 < X_{He} \end{cases}$$

In SPECTRA the boundary values have been slightly modified compared to the original correlation. Values at the cross points of different correlations were selected, to provide a smooth transition from one correlation to another. The values are: X_a = 0.100477, X_{He} = 0.0100441, 0.0998948.

- *Ogg correlations* [134]

$$f_{NC}(H_2O-air)=\begin{cases}1-1.165\,X_a^{0.26} & for \quad X_a<0.3\\ 1-0.905\,X_a^{0.05} & for \quad 0.3<X_a<0.9\\ 1-X_a & for \quad 0.9<X_a\end{cases}$$

$$f_{NC}(H_2O-He)=\begin{cases}1-1.590\,X_{He}^{0.29} & for \quad X_{He}<0.11\\ 1-0.865\,X_{He}^{0.014} & for \quad 0.11<X_{He}<0.86\\ 1-X_{He} & for \quad 0.86<X_{He}\end{cases}$$

In SPECTRA the boundary values have been slightly modified compared to the original correlation. Values at the cross points of different correlations were selected, to provide a smooth transition from one correlation to another. The values are: X_a = 0.300419, 0.900258, X_{He} = 0.110179, 0.863221.

- *Modified Ogg correlations* [135]

The modified Ogg correlation has been introduced in [135], based on comparisons with experimental data for external condensation. For low noncondensable gas fractions the correlation gives very similar results to the original correlation; differences are observed in high noncondensable fractions (Figure 7-8). The correlation is:

$$f_{NC}(H_2O-air)=\begin{cases}1.00-1.165\,X_a^{0.26} & for \quad X_a<0.3\\ 0.21-0.210\,X_a & for \quad X_a>0.3\end{cases}$$

$$f_{NC}(H_2O-He)=\begin{cases}1.00-1.590\,X_{He}^{0.29} & for \quad X_{He}<0.12\\ 0.16-0.160\,X_{He} & for \quad X_{He}>0.12\end{cases}$$

In SPECTRA the boundary values at the cross points of different correlations were applied, to provide a smooth transition from one correlation to another. The values are: X_a = 0.302131, X_{He} = 0.119726.

Comparison of degradation factors obtained from the above three correlations, is shown in Figure 7-7 and Figure 7-8. It has been found out (see Volume 3) that the KSP correlation gives very good results for internal condensation (inside tubes), while the modified Ogg correlation provides good results for external condensation (outside tubes, walls). Such is therefore the default selection of correlation (Volume 2), which may be changed in input for each 1-D or 2-D Solid Heat Conductor by input parameters.

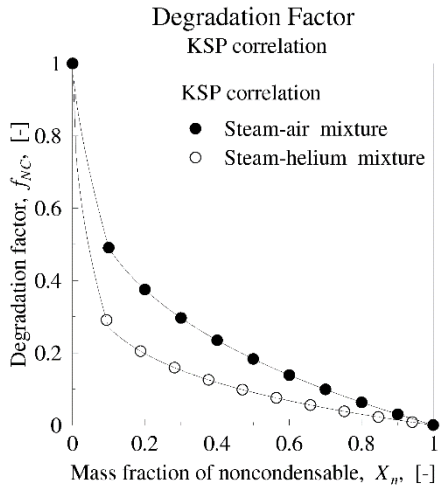


Figure 7-7 KSP correlation for f_{NC}

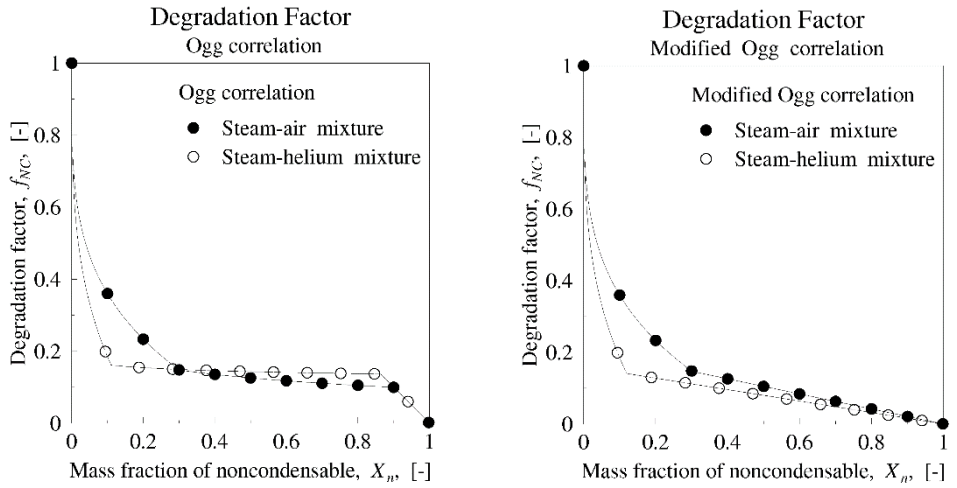


Figure 7-8 Ogg correlation (left) and modified Ogg (right) for f_{NC}

- Influence of shear stress

  Shear stress can considerably modify the condensate flow in case of high gas velocities. In case of co-current downward flow of gas and condensate, the interfacial shear will increase condensate velocity, and therefore decrease the film thickness, which finally leads to increase of the heat transfer coefficient. In case of counter-current flow, with gas flowing up, the opposite effect will be observed, and the heat transfer coefficient will be decreased.

  Nusselt derived equations for condensation in presence of high gas velocities, based on the assumption that the entire condensate layer remained in laminar flow. The resulting film velocity is (see [136]):

$$v = \frac{g(\rho_{liq} - \rho_{gas})\delta^2}{2\eta_{liq}} + \tau_{SH}\frac{\delta}{\eta_{liq}}$$

where δ is the film thickness, and τ_{SH} is the interfacial shear, obtained from:

$$\tau_{SH} = f\frac{\rho_{gas} v_{gas}|v_{gas}|}{2}$$

with v_{gas} being the gas downflow velocity, and f being the friction factor. The friction factor is taken from the data of Bergelin et al. [138], shown in Figure 7-9. The ratio (σ_w/σ) is a correction factor for fluids other than water, and represents the ratio between surface tension of water to the surface tension of the fluid being considered.

Bergelin data is approximated by a correlation equation, valid for low values of (Γ/ρ_{liq}). This is a region of practical interest, since the condensate mass flux per unit width, Γ, is typically smaller than 0.1 kg/m/s and the liquid density is of order of 1000 kg/m^3. The approximation is:

- *Laminar gas flow* $Re_{gas} < Re_{lam} = 2000$

$$f = \frac{16}{\mathrm{Re}}$$

- *Transition flow* $Re_{lam} < Re_{gas} < Re_{tur} = 3000$
Cubic interpolation is performed, which provides a smooth transition between laminar and turbulent regions.

$$f = f_{lam} - (f_{lam} - f_{tur})\left(3 - 2\frac{\mathrm{Re} - \mathrm{Re}_{lam}}{\mathrm{Re}_{tur} - \mathrm{Re}_{lam}}\right)\left(\frac{\mathrm{Re} - \mathrm{Re}_{lam}}{\mathrm{Re}_{tur} - \mathrm{Re}_{lam}}\right)^2$$

- *Turbulent gas flow* $Re_{gas} > Re_{tur} = 3000$

$$f = 0.07\,\mathrm{Re}^{0.24}$$

The values obtained from the correlation are shown in Figure 7-10. The correlation provides a very good fit to the Bergelin data for (Γ/ρ_{liq}) < 0.2. This formula has been introduced in reference [135], specifically for application in the SPECTRA code.

The final value of the interfacial shear enhancement factor, f_{SH}, is obtained as a ratio of film velocity with shear, to that which would be observed in absence of shear (τ_{SH}=0).

$$f_{SH} = 1.0 + \frac{2\tau_{SH}}{g(\rho_{liq} - \rho_{gas})\delta} = 1.0 + f\frac{\rho_{gas} v_{gas}|v_{gas}|}{g(\rho_{liq} - \rho_{gas})\delta}$$

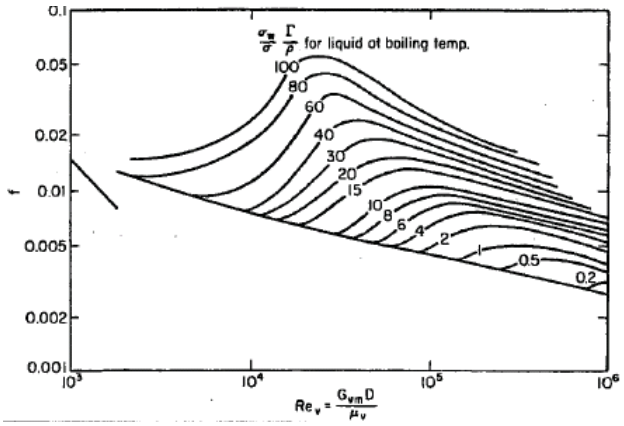


Figure 7-9 Friction factor for gas (air) flowing in a tube with liquid layer on the wall, Bergelin et al.

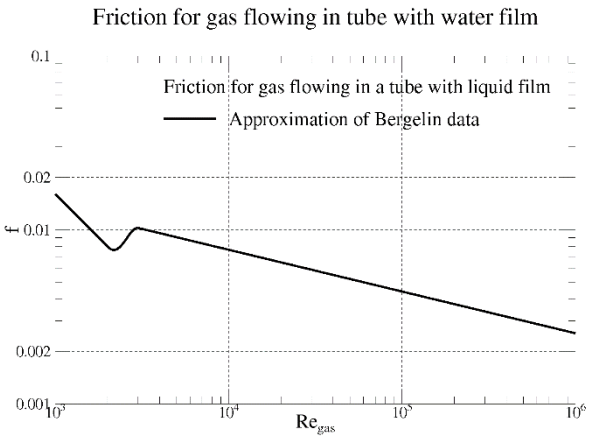


Figure 7-10 Approximation of friction factor applied in SPECTRA.

Note that the interfacial shear can either increase the heat transfer coefficient (f_{SH}>1.0 when v_{gas}>0.0), or decrease the heat transfer coefficient (f_{SH}<1.0 when v_{gas}<0.0). In SPECTRA a minimum and a maximum limit are imposed on interfacial shear: $0.5 < f_{SH} < 2.0$.

Two other models are available for the shear enhancement calculation: the Kuhn-Schrock-Petersen (KSP) correlation [133], and the Ogg correlation [134]. The Ogg correlation is:

$$f_{SH} = 1 + 1.2 \times 10^{-3} \, \mathrm{Re}_{gas}^{0.7}$$

The KSP correlation takes into account the enhancement of heat transfer due to shear and other factors, like waviness at film surface. The Ogg and KSP correlations for shear enhancement are currently not used by the code. Before they can be activated their applicability range should be extended into the counter-current flow range, where f_{SH}<1.0. The present forms of these correlations give an incorrect trend of shear influence in this range.

- Full condensation correlation

  The full condensation correlation consists of three factors: pure steam heat transfer coefficient, h_{cond}, noncondensables degradation factor, f_{NC}, and shear factor, f_{SH}. It has been found however that the direct product of the three factors would lead to incorrect results when non-condensable gases are present. The equation used to calculate the average film thickness must be modified to take into account the influence of the non-condensable gas degradation factor. The degradation results in a decrease of the average condensate film thickness by the factor of $f_{NC}^{1/4}$, and consequently an increase of the average heat transfer coefficient, by the factor of $f_{NC}^{-1/4}$. Therefore:

$$h = \left(\frac{h_{cond}}{f_{NC}^{1/4}} \right) \cdot f_{NC} \cdot f_{SH}$$

  where h_{cond} is the pure steam condensation heat transfer coefficient, obtained from the Nusselt model. This formulation provides a good agreement with experimental data for Berkeley single tube condensation tests [133].

  In the present formulation the condensation heat transfer coefficient is effectively proportional to $f_{NC}^{3/4}$. In the formulation used in the GE version of the TRACG code, TRACG, it is proportional to f_{NC}. This is because the TRACG formulation of condensation model is different. TRACG does not use the Nusselt laminar film model, but calculates the condensate film transport using two-phase momentum equations. Therefore the multiplier $f_{NC}^{-1/4}$ is not applicable in the TRACG model.

7.1.5 Nucleate Boiling

Two correlations are available for nucleate boiling: Rohsenow and Chen. The selection of correlation is made for each SC/TC surface via input data. The default selection is: Chen for an internal flow and Rohsenow for an external flow.

- Rohsenow correlation

  The Rohsenow correlation [139] is:

$$\frac{c_{p,liq}(T_{wall} - T_{sat})}{h_{l-v}} = C_{sf} \left[\frac{q_{wall}}{\eta_{liq} h_{l-v}} \left(\frac{\sigma}{g(\rho_{liq} - \rho_{gas})} \right)^{1/2} \right]^{0.33} \mathrm{Pr}_{liq}$$

  $c_{p,\,liq}$ specific heat of liquid, (J/kg/K)
  T_{wall} wall surface temperature, (K)
  T_{sat} saturation temperature, (K)
  $h_{l\text{-}v}$ latent heat of evaporation, (J/kg)
  q_{wall} wall heat flux, (W/m^2)
  η_{liq} viscosity of liquid, (kg/m/s)
  σ surface tension, (N/m)
  g gravity acceleration, (m/s^2)

ρ_{liq} density of liquid, (kg/m^3)
ρ_{gas} density of vapor, (kg/m^3)
Pr_{liq} Prandtl number of liquid, (-)
C_{sf} constant dependant on surface and fluid type (the values for water as a fluid are reproduced in Table 7-6), (-)

Table 7-6 Values of C_{sf}, Rohsenow correlation, water as coolant

Surface type	C_{sf}
Nickel	0.0060
Platinum	0.0130
Brass	0.0060
Polished copper	0.0128
Lapped copper	0.0147
Scored copper	0.0068
Ground and polished stainless steel	0.0080
Teflon pitted stainless steel	0.0058
Chemically etched stainless steel	0.0133
Mechanically polished stainless steel	0.0132

- Chen correlation

The Chen correlation [125] consists of two parts: the forced convective part, and the boiling part. The forced convective part is calculated based on Dittus-Boelter correlation, while the boiling part is calculated based on Forster-Zuber correlation:

$$h = h_{D-B}F + h_{F-Z}S$$

h_{D-B} convective heat transfer coefficient, given by Dittus-Boelter correlation (section 7.1.2), (W/m^2/K)
F two-phase Reynolds number factor, (-)
h_{F-Z} boiling heat transfer coefficient, given by Forster-Zuber correlation, (W/m^2/K)
S boiling suppression factor, (-)

In the present application two modifications were made into the original Chen correlation:

- Instead of using the Dittus-Boelter correlation, the convective part is calculated by the standard natural and forced convective correlation sets (sections 7.1.1, 7.1.2), the using logic described in section 7.2.2.
- The fluid-surface factor, C_{sf}, used by the Rohsenow correlation, is applied for the boiling part. In the Rohsenow correlation the heat flux is proportional to C_{sf}^{-3}. A typical value of C_{sf} (water on stainless steel) is 0.013. The Chen nucleate boiling correlation is multiplied by the ratio: $(0.013/C_{sf})^3$. C_{sf} is an input parameter, with the default value of 0.013. Thus, with the default value the multiplier is equal to 1.0. It has been shown that use of the C_{sf} factor allows to obtain better agreement with experimental data when the Chen correlation is used to calculate boiling on surfaces other than steel (see Volume 3)

The forced convective boiling model, as implemented, calculates the heat flux from the following expression:

$$q_{wall} = h_{conv} F (T_{wall} - T_{fluid}) + h_{F-Z} S \left(\frac{0.013}{C_{sf}} \right)^3 (T_{wall} - T_{sat})$$

The Forster-Zuber heat transfer coefficient, $h_{F\text{-}Z}$, the two-phase Reynolds number factor, F, and the boiling suppression factor, S, are described below.

- Forster-Zuber correlation for heat transfer coefficient, $h_{F\text{-}Z}$

  The Forster-Zuber correlation [140] is:

$$h_{F-Z} = 0.00122 \frac{\Delta T_{sat}^{0.24} \Delta p_{sat}^{0.75} c_{p,liq}^{0.45} k_{liq}^{0.79}}{\sigma^{0.5} h_{l-v}^{0.24} \eta_{liq}^{0.29} \rho_{gas}^{0.24}}$$

  ΔT_{sat} wall surface temperature minus saturation temperature, (K)
  Δp_{sat} saturation pressure difference corresponding to ΔT_{sat}, (Pa)
  k_{liq} thermal conductivity of liquid, (W/m/K)

  Other symbols are explained above. All symbols are in SI units (see [44]).

- Two-phase Reynolds Number multiplier, F

  The factor F is defined as the ratio of the Reynolds number in two-phase flow to the Reynolds number for pure liquid phase, raised to the power of 0.8 [136]:

$$F = \left(\frac{\mathrm{Re}_{TP}}{\mathrm{Re}_{liq}} \right)^{0.8}$$

  Re_{TP} two-phase flow Reynolds number, (-)
  Re_{liq} pure liquid Reynolds number, (-)

  To evaluate the F factor a correlation based on the Collier and Pulling correlation [137] is used. The Collier and Pulling correlation gives the F factor versus Martinelli parameter, χ_{tt}, as follows:

$$F = 2.5 \left(\frac{1}{\chi_{tt}} \right)^{0.7}$$

  This correlation gives values of F very similar to that recommended by Chen [125]. The values of F, recommended by Chen are plotted in Figure 7-11. The values obtained using the Collier and Pulling correlation are shown in Figure 7-12. Discrepancies are observed only for low ($1/\chi_{tt}$) values. To obtain F close to the Chen data the Collier-Pulling correlation has been modified as follows:

$$F = 2.5\left(\frac{1}{\chi_{tt}} + 0.17\right)^{0.7}$$

This formula has been introduced in reference [135], specifically for application in the SPECTRA code. The value of 0.17 was chosen since with that value F is equal to 1.0 when $(1/\chi_{tt}) \approx 0.1$. The values of F obtained with the modified correlation are shown in Figure 7-12. The modified Collier-Pulling correlation is in good agreement with the data given by Chen. The discrepancies are much smaller than the shadowed area representing the experimental data scatter shown in Figure 7-11.

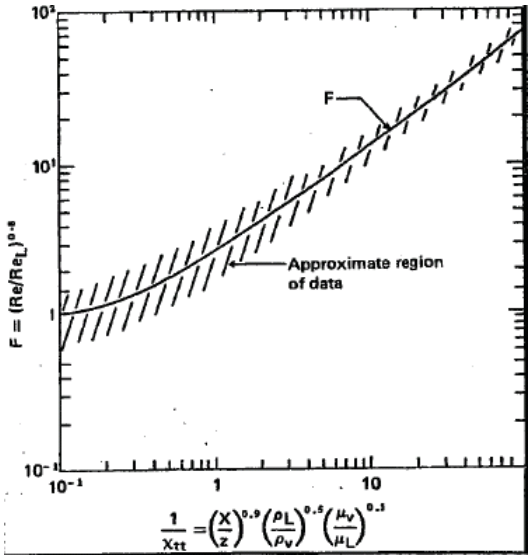


Figure 7-11 Two-phase Reynolds number factor, *F*, Chen [125]

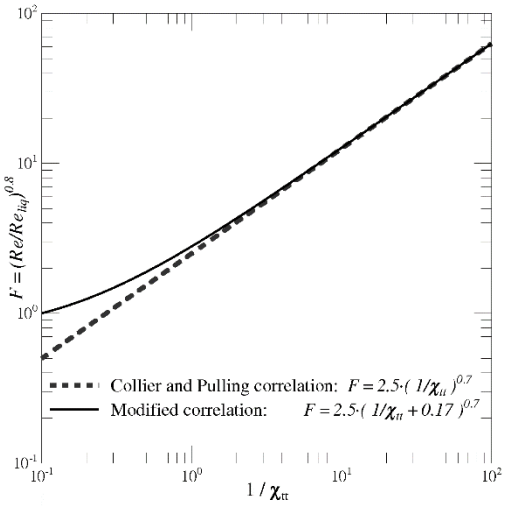


Figure 7-12 Two-phase Re factor, *F*, Collier-Pulling and modified Collier Pullig correlation

The modified Collier-Pulling correlation is used in SPECTRA to calculate F. The limits are imposed on the input parameter: $0.1000935875<\chi_{tt}<100.0$. The exact value of the lower limit was obtained as the χ_{tt} for which $F = 1.0$. The method of calculating χ_{tt} is shown in section 7.1.13.

- Boiling suppression factor, S

The suppression factor, S, is defined as the ratio of the mean superheat, ΔT_e, to the wall superheat, ΔT_{sat}, raised to the power of 0.99 [136]:

$$S = \left(\frac{\Delta T_e}{\Delta T_{sat}} \right)^{0.99}$$

ΔT_e mean superheat, (K), equal to the mean fluid temperature in the bubble growth zone, minus the saturation temperature, (K)

ΔT_{sat} wall superheat, (K), equal to the wall surface temperature, minus the saturation temperature

The value of the suppression factor depends on flow. For non-flow conditions (pool boiling) the S factor is close to one. With increasing flow the value of the S factor decreases. This is caused by the fact that with increasing flow the boundary layer becomes thinner, and the effective temperature that is seen by the bubble is lower (Figure 7-13).

The values of suppression factor are given by Chen versus two-phase Reynolds number, Re_{TP}, in the form of a graph. The graph defining the S factor is shown in Figure 7-14. The S factor is plotted against the two phase Reynolds number, which is equal to:

$$\mathrm{Re}_{TP} = \mathrm{Re}_{liq} F^{1.25}$$

To approximate the S factor the following correlation has been developed:

$$S = \frac{1}{1 + 0.5 \times 10^{-5} \mathrm{Re}_{TP}}$$

with the limits imposed on the two-phase Reynolds number: $0<Re_{TP}<600{,}000$. This formula has been introduced in reference [135], specifically for application in the SPECTRA code. The values of S obtained with this correlation are shown in Figure 7-15. Discrepancies between correlation and experiment are smaller than the experimental data scatter.

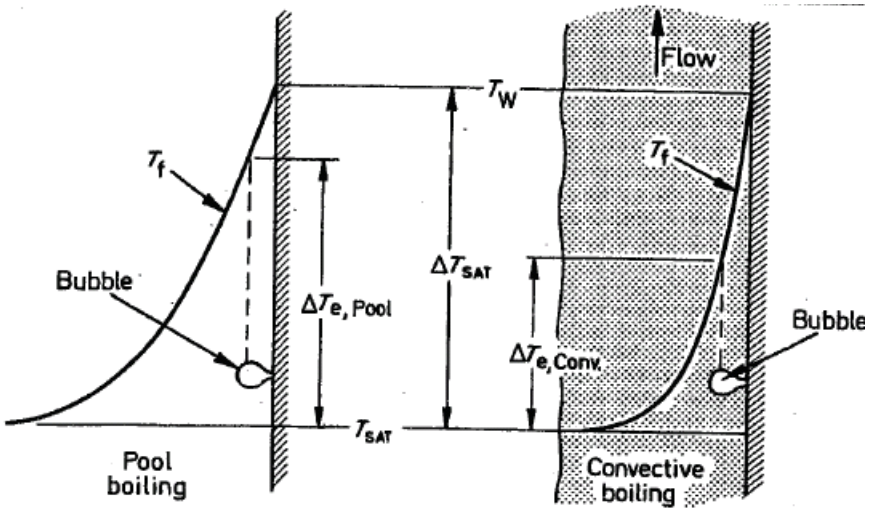


Figure 7-13 Temperature profiles for pool boiling and convective boiling [125].

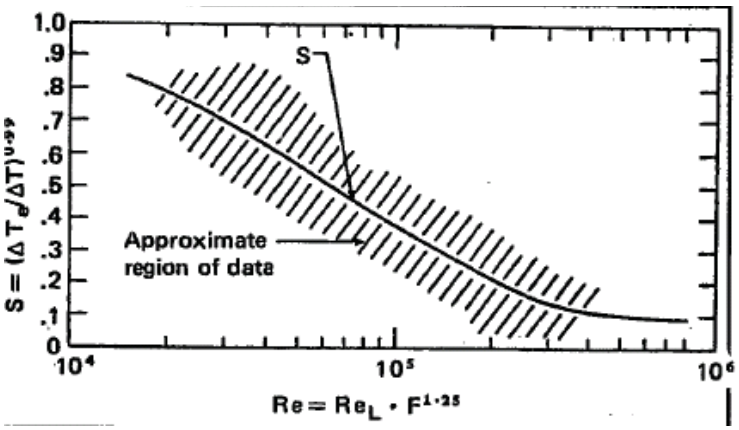


Figure 7-14 Suppression factor, *S*, Chen [125]

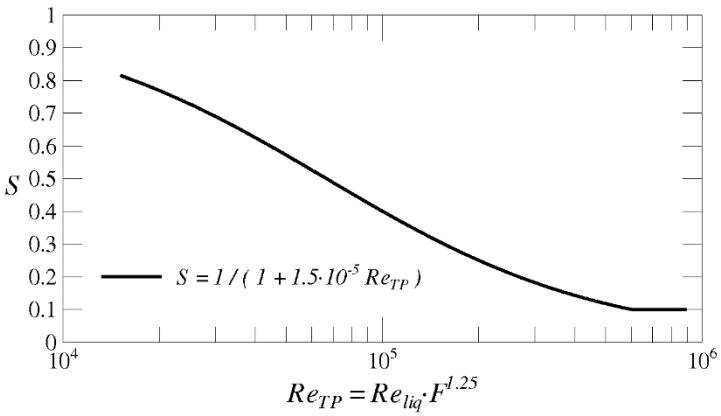


Figure 7-15 Suppression factor, *S*, correlation

7.1.6 Critical Heat Flux

Two models are available to calculate critical heat flux (CHF). The models are shortly referred to as the pool boiling, and the convective boiling critical heat flux models. The selection of a model is made for each SC/TC surface via input parameters (MCFLSC, MCFRSC, MCFBTC, see Volume 2). The default selection is: convective boiling model for internal flow, and pool boiling model for external flow. Both models are described in this section.

- Pool boiling

  The pool boiling critical heat flux model is based on the Zuber correlation, with the Ivey-Morris correction for subcooling, a geometry dependent multiplier, and a quality-dependent multiplier. The critical heat flux is calculated from the following formula:

$$q_{CHF} = q_{Zub} C_{I-M} C_{geom} (1 - X)$$

  q_{Zub} critical heat flux from Zuber correlation, (W/m^2)
  C_{I-M} Ivey-Morris correction for subcooling, (-)
  C_{geom} geometry dependent multiplier, (-)
  X quality, (-), the multiplier (1–X) may be deactivated by C0IMHT (see Volume 2)

  The Zuber critical heat flux correlation [141], q_{Zub}, is given by:

$$q_{Zub} = C_{Zub} \rho_{gas} h_{l-v} \left(\frac{\sigma g (\rho_{liq} - \rho_{gas})}{\rho_{gas}^2} \right)^{0.25} \left(\frac{\rho_{liq}}{\rho_{liq} + \rho_{gas}} \right)^{0.5}$$

  ρ_{liq} liquid density, (kg/m^3)
  ρ_{gas} vapor density, (kg/m^3)
  h_{l-v} latent heat of vaporization, (J/kg)
  σ surface tension, (N/m)
  g gravity constant, (m/s^2)

  The value of constant C_{Zub} is 0.13, originally suggested by Zuber. Reference [20], page 13-31 recommends 0.18 instead of 0.13, as a value that gives better agreement with experiment. However, based on reference [55] figure 4.3 it is safer to use the Zuber's original value.

$$C_{I-M} = 1 + C_0 \cdot \frac{\rho_{liq} c_{p,liq} (T_{sat} - T_{liq})}{\rho_{gas} h_{l-v}} \cdot \left(\frac{\rho_{gas}}{\rho_{liq}} \right)^{0.25}$$

  T_{liq} liquid temperature, (K)
  T_{sat} saturation temperature, (K)
  $c_{p,liq}$ liquid specific heat, (J/kg/K)
  C_0 user-defined coefficient (C0IMHT, default value = 0.1 - see Volume 2)

  The values of the geometry dependent multiplier, C_{geom}, are based on the Leinhard et al. extension to the Zuber theory (see [55]):

  Flat plates: C_{geom} = 1.140
  Cylinders: C_{geom} = 0.985

Spheres: $C_{geom} = 0.840$

In case of cylinders and spheres, the values for "large cylinders" and "large spheres" are used. This is because the "small" cylinders and spheres are those for which radius is of order of: $(\sigma/g(\rho_{liq}-\rho_{gas}))^{(1/2)}$ (see [55]). For water the value is of order of 10^{-3} m. Diameters of typical structures are much larger.

The term $(1-X)$ is used as a rough representation of the decrease of CHF with quality, observed in experiments and taken into account in the convective boiling CHF correlations, described below (Zuber correlation was developed for pool boiling conditions, thus X≈0).

- Convective boiling

When the convective boiling model is applied, the critical heat flux depends on the mass flux, G. For low mass fluxes, $G < G_{pool} = 1.0$ kg/m^2/s, the pool boiling model, described above, is applied. For high mass fluxes, $G > G_{conv} = 750$ kg/m^2/s, the convective model is applied. In the intermediate region, $G_{pool} < G < G_{conv}$, a third order interpolation is applied to provide smooth transition between the models. In the convective range, $G > G_{conv}$, three critical heat flux models are available:

- Biasi correlation,
- U.S.S.R Academy of Science lookup tables,
- Groeneveld lookup tabules.

Biasi correlation
With Biasi correlation [142], the critical heat flux, q_{CHF} (W/m^2) is given by:

$$q_{CHF} = Max(q_1, q_2)$$

The values of q_1 and q_2 are equal to:

$$q_1 = \frac{1883}{D^a G^{1/6}}\left(\frac{f(p)}{G^{1/6}} - X\right)$$
$$q_2 = \frac{3780 \cdot h(p)}{D^a G^{0.6}}(1 - X)$$

The pressure dependent functions, $f(p)$ and $h(p)$ are calculated from:

$$f(p) = 0.7249 + 0.099 \cdot p \cdot \exp(-0.032\, p)$$
$$h(p) = -1.159 + 0.149 \cdot p \cdot \exp(-0.019\, p) + \frac{8.99\, p}{10 + p^2}$$

The exponent a in the formulae for q_1 and q_2 is equal to:

$$a = \begin{cases} 0.4 & if \quad D > 1\, cm \\ 0.6 & if \quad D < 1\, cm \end{cases}$$

In the above formulae p is pressure in bars, G is the mass flux in g/cm^2/s, X is quality, D is diameter in cm, and q_{CHF} is heat flux in W/cm^2. (Note that internal calculation is performed

using the original units, but input and output from the appropriate subroutine is in SI units.) The range of application is:

$$\begin{array}{ccc} 2.7 & < p < & 140\ bar \\ 1/(1+\rho_{liq}/\rho_{gas}) & < X < & 1.0 \\ 10.0 & < G < & 600\ g/cm^2/s \\ 0.3 & < D < & 3.75\ cm \end{array}$$

Thus the Biasi correlation should not be applied in subcooled region. To improve results, the Ivey-Morris correction is used in that region (see discussion of Figure 7-18 below).

U.S.S.R Academy of Sciences critical heat flux look-up tables
The USSR Academy of Sciences has produced a series of standard tables of critical heat flux as a function of the local bulk mean water conditions [143]. The tables give critical heat flux for various pressures, mass fluxes and qualities (or subcooling, when quality is equal to zero), and for a reference tube diameter, D_{ref} = 8 mm:

$$q_{UAS}(p, G, X\ or\ \Delta T_{sub})$$

For the tube diameters other than D_{ref} the critical heat flux is calculated from the following formula:

$$q_{CHF} = q_{UAS}(p, G, X\ or\ \Delta T_{sub})\left(\frac{D_{ref}}{D}\right)^{0.5}$$

valid for : 4 mm < D < 16 mm. The applicability has been extended to 37.5 mm based on Matzner experiments [145] (see the discussion of Matzner experiments below). The calculation of critical heat flux is based on interpolation of the tabulated data. The tables were reproduced from [136]. The applicability range is shortly discussed below.

Range of subcooling, ΔT_{sub}:

$$0.0 \le \Delta T_{sub} \le 75.0\ \text{K}$$

This is quite a wide range and no extrapolation beyond that range has been implemented. If a subcooling larger than 75 K is encountered, the value for 75 K will be used to obtain the value of the critical heat flux.

Range of quality, X:

$$0.0 \le X \le 0.75$$

The maximum quality value varies for different pressures and different mass fluxes. For qualities larger than maximum the critical heat flux is linearly extrapolated to give the value of zero at $X = 1.0$.

$$q_{UAS}(p, G, X > X_{max}) = q_{UAS}(p, G, X = X_{max}) \cdot \frac{1 - X}{1 - X_{max}}$$

Range of mass flux, G:

$$750 \le G \le 5000 \text{ kg/m}^2\text{-s}$$

No extrapolation is performed. If the value is beyond this range it is set to the nearest value within this range. For $G > 5000$ kg/m^2/s the influence of mass flux on critical heat flux is typically very small. For mass fluxes below 750 kg/m^2/s, interpolation is applied to obtain smooth transition between the pool boiling and the convective boiling models.

Range of pressures, p:

$$29.5 \le p \le 196.0 \text{ bar}$$

The maximum pressure is quite large and no extrapolation is needed. The low pressure extrapolation is performed using the following formula:

$$q_{UAS}(p < p_{min}, G, X \text{ or } \Delta T_{sat}) = q_{UAS}(p = p_{min}, G, X \text{ or } \Delta T_{sat}) \cdot \left[1 - 0.60 \left(\frac{p_{min} - p}{p_{min}} \right)^2 \right]$$

This extrapolation has been selected to obtain qualitative agreement with the shape of the pool boiling curve in the low pressure region. Furthermore it was checked that it gives a qualitatively good agreement with the Groeneveld lookup tables (Figure 7-16), which are valid for pressures down to 1.0 bar.

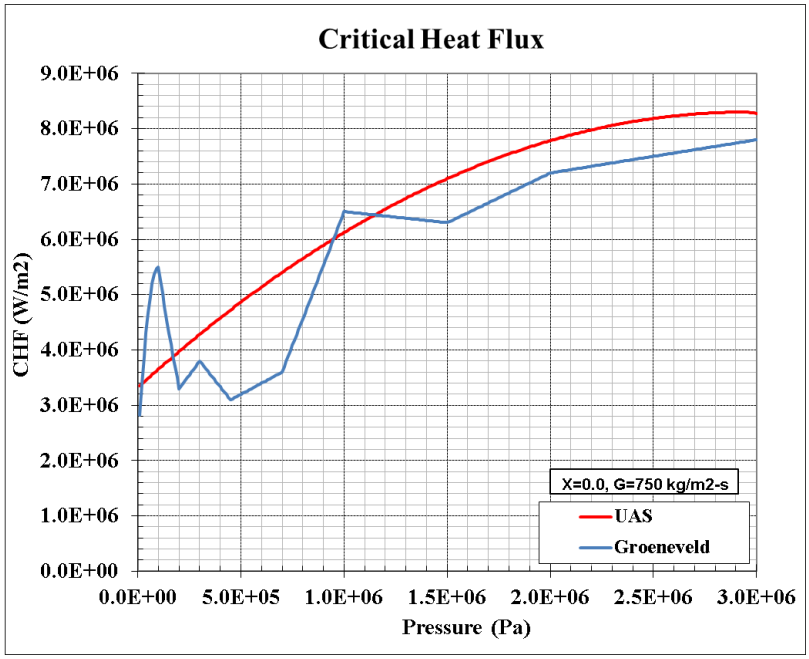


Figure 7-16 Comparison of UAS and Groenveld CHF model for pressures 0 - 30 bar

Groeneveld critical heat flux look-up tables

A wide range critical heat flux look-up tables has been prepared by Groeneved et al. (1986) [144]. The tables give critical heat flux for various pressures, mass fluxes and qualities (or subcooling, when quality is equal to zero), and for a reference tube diameter, D_{ref} = 8 mm:

$$q_{GRO}(p, G, X)$$

For the tube diameters other than D_{ref} the critical heat flux is calculated from the following formula:

$$q_{CHF} = q_{GRO}(p, G, X \text{ or } \Delta T_{sub})\left(\frac{D_{ref}}{D}\right)^{0.5}$$

valid for : 4 mm < D < 16 mm. The applicability has been extended to 37.5 mm based on Matzner experiments [145] (see the discussion of Matzner experiments below). The calculation of critical heat flux is based on interpolation of the tabulated data. The applicability range is shortly discussed below.

Range of quality, X:

$$-0.5 \le X \le 1.0$$

Range of mass flux, G:

$$0 \le G \le 7500 \text{ kg/m}^2\text{-s}$$

Range of pressures, p:

$$1.0 \le p \le 200.0 \text{ bar}$$

Comparison of different CHF models with available experimental data showed that the look-up tables (LUT) of the U.S.S.R Academy of Science and of Groeneveld et al. provide most accurate prediction in the wide range of relevant parameters. The look-up tables of UAS is currently the default model for convective boiling conditions. The look-up tables of Groeneveld et al. and the Biasi correlation are available as alternative models.

A disadvantage of the Biasi correlation is the fact that it is not applicable for subcooled liquid. This is illustrated based on Weatherhead experiments [126]. The experimental results are shown in Figure 7-17 (left). It is seen that in the subcooled region the influence of mass flux is different than in the two-phase region. While in the subcooled region q_{CHF} increases with increasing G, in the two-phase region q_{CHF} decreases with increasing G. These trends are correctly predicted by the look-up tables (Figure 7-18). If the Biasi correlation is used the trend observed in the two-phase region is also observed in the subcooled region (Figure 7-17, right). The fact that q_{CHF} increases with increasing subcooling is a consequence of using Ivey-Morris correction for subcooling.

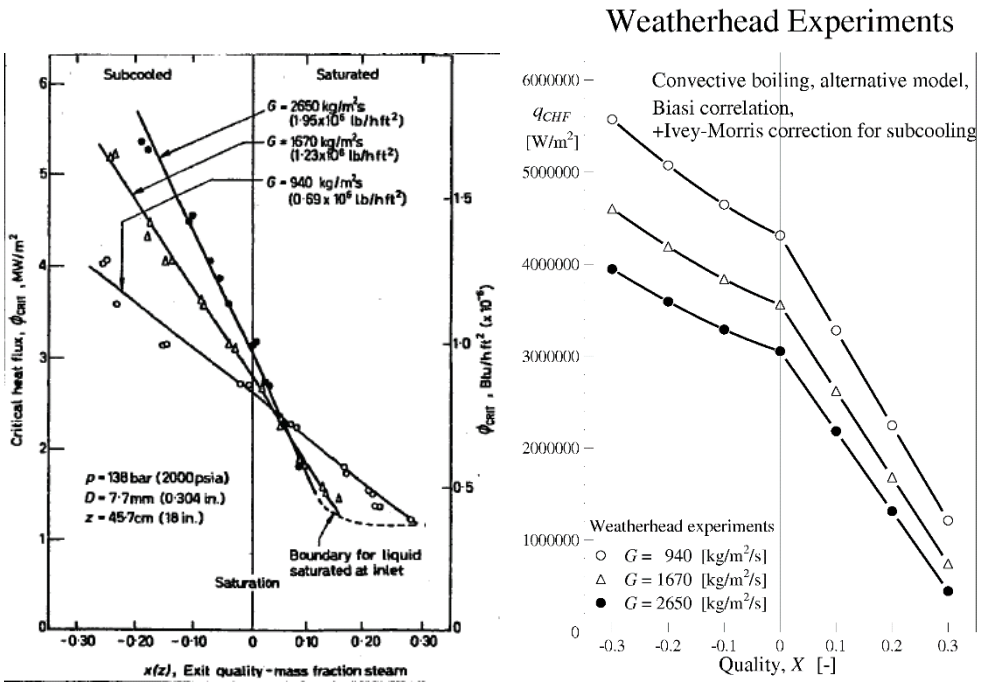


Figure 7-17 Weatherhead experiments: left: data, right: Biasi+Ivey-Morris

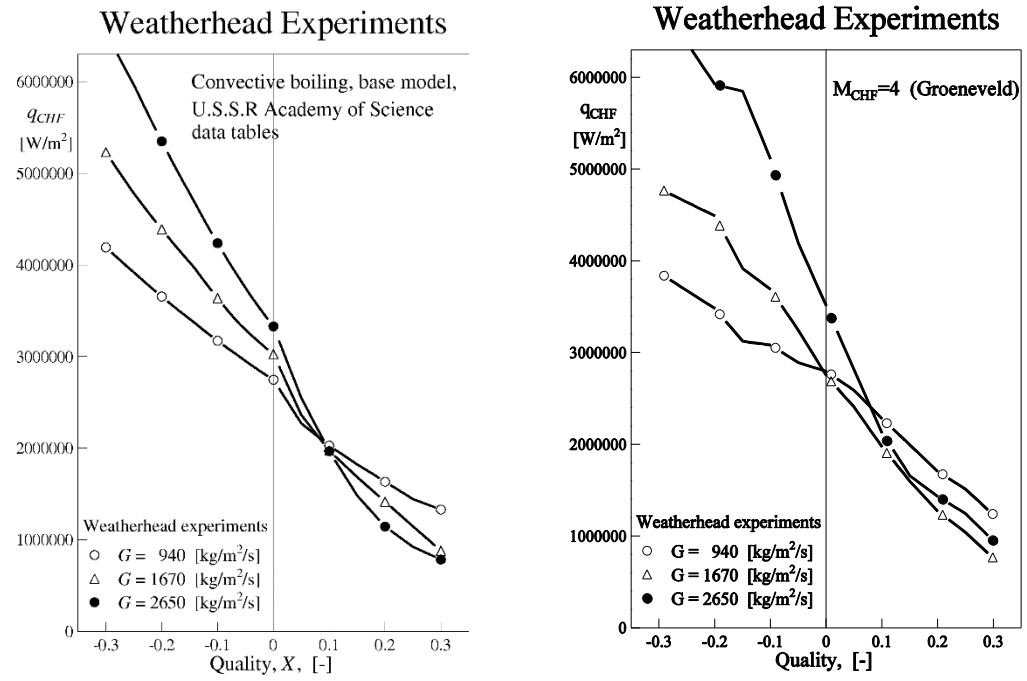
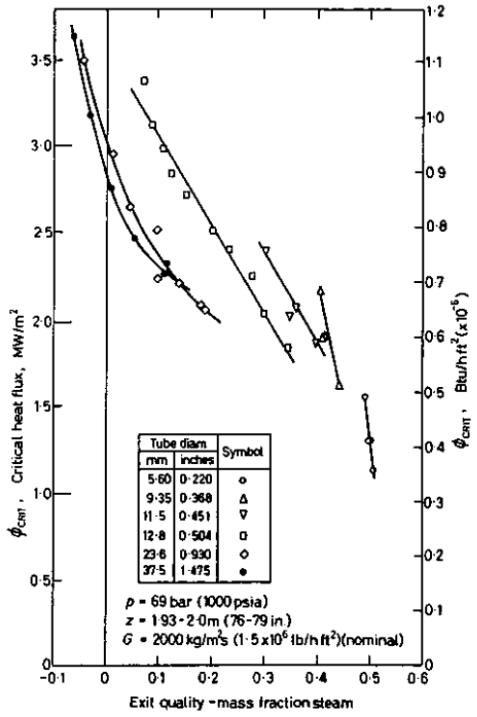


Figure 7-18 Weatherhead experiments: left: UAS LUT, right: Groeneveld LUT

- Influence of diameter on convective boiling CHF - Matzner experiments



[136]

Figure 7-19 Influence of diameter, reproduced from

Figure 7-19 shows the results of Matzner experiments [145]. The results calculated using the UAS lookup tables (the base model) are shown in Figure 7-20, left. The CHF is somewhat overestimated in case of the smallest diameter (D=5.9 mm) and somewhat underestimated in case of the largest diameter (D=37.5 mm). For other diameters a good agreement is observed.

Figure 7-20 right shows the results obtained with Biasi (the alternative model). Generally CHF is overestimated for small diameters (D=5.6, 9.35 mm) and overestimated for D = 12.8 mm.

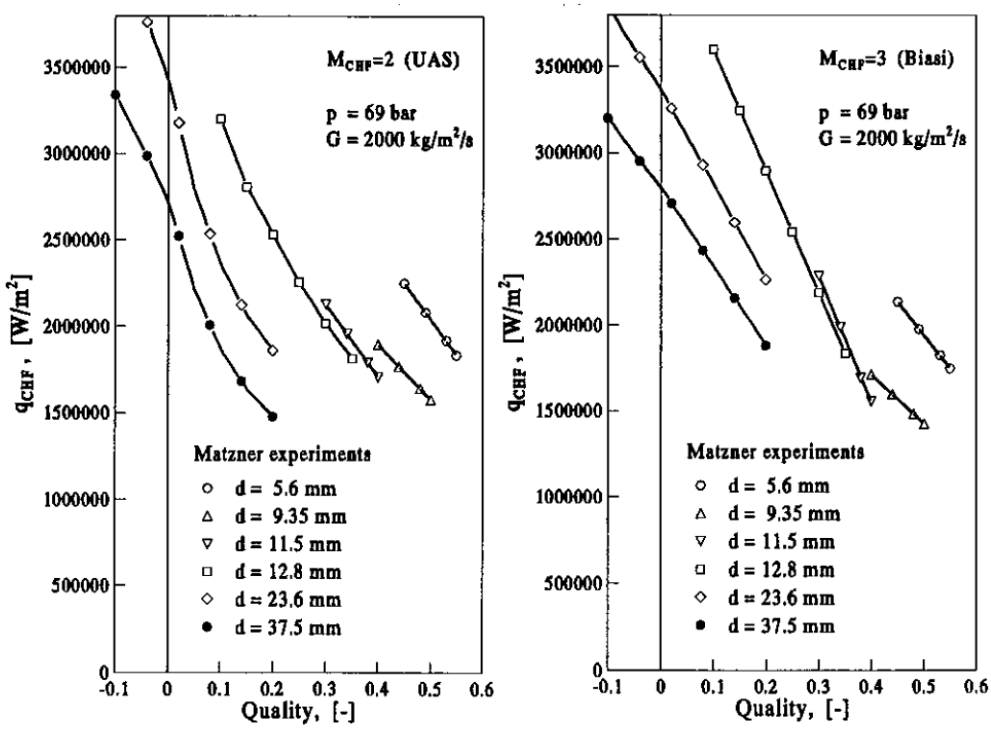


Figure 7-20 Influence of diameter, left: UAS (base model), right: Biasi (alternative model)

7.1.7 Critical Heat Flux in Small Channels

Typically the CHF increases with decreasing diameter. The dependency is well represented by:

$$CHF \sim \left(\frac{D}{D_0} \right)^n$$

where D_0 is a reference diameter (equal to 0.008 m) and n is between –1/3 and –1/2. This relation holds for diameters larger than a certain critical diameter, D_{crit}. Below the critical diameter CHF decreases with decreasing diameter. Measurement data indicate that the critical diameter depends on subcooling. For large subcooling (55 K) the critical diameter is smaller than 0.3 mm [191]. For subcooling of about 20 K it is about 1 - 2 mm [192]. For saturated liquid it is as large as 7 mm [193]. This data is approximated in SPECTRA by the following correlation:

$$D_{crit} = D_c \cdot \exp[-A_c X]$$

Here X is quality ($X \leq 0.0$) while D_c and A_c are constants. The data points are shown in Figure 7-21. The following constants are selected: D_c = 0.007 m and A_c = 30.0. The line obtained with the correlation is shown in Figure 7-21. The coefficients D_c and A_c may be redefined in the input (see Volume 2).

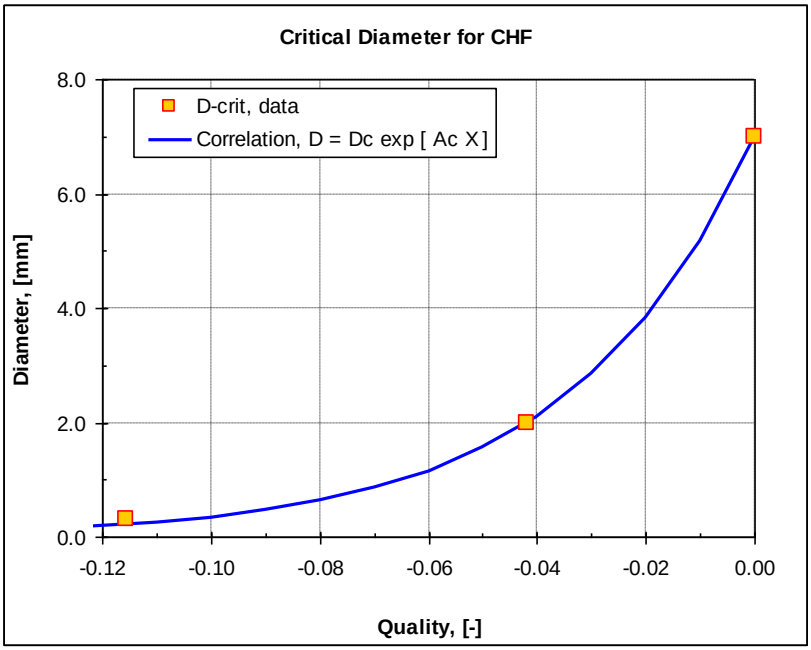


Figure 7-21 Critical diameter for CHF

For narrow channels the CHF is interpolated between the value obtained for the critical diameter and zero. A third order, smooth interpolation is applied, as follows:

$$CHF = CHF_{crit} \cdot [3 - 2 \cdot (D / D_{crit})] \cdot (D / D_{crit})^2$$

The resulting CHF is shown in Figure 7-22. The solid line shows the CHF in the large diameter region, where CHF is proportional to $(D/D_0)^{-0.5}$. The values of CHF in the low diameter region are shown with the square marks. The critical diameter is in this case assumed to be 0.004 m.

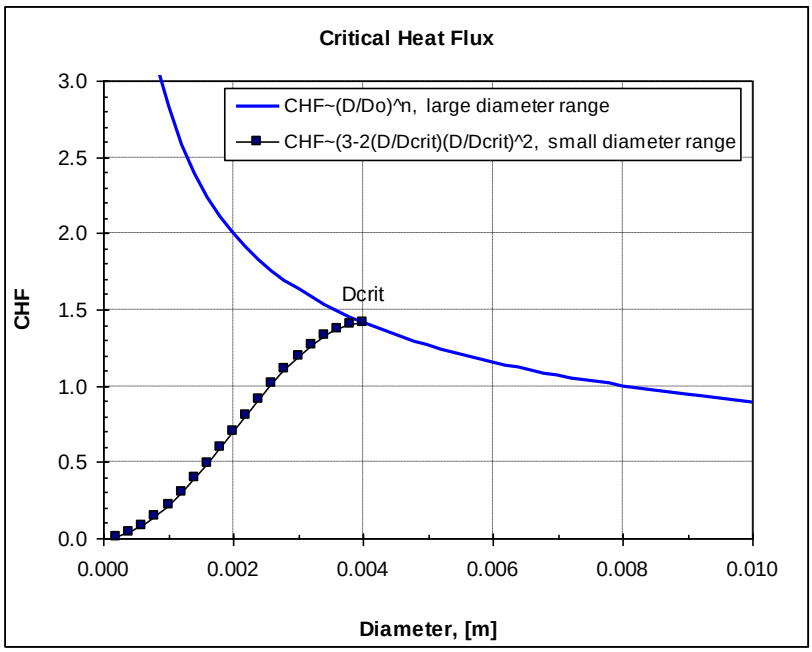


Figure 7-22 Influence of diameter on CHF

7.1.8 User-Defined Critical Heat Flux

On top of the built-in correlations for the critical heat flux, an option is provided for the user to define a different critical heat flux correlation. With this option the CHF is defined by a Control Function. Since the Control Functions have access to all parameters in SPECTRA, practically any CHF correlation can be built. The expense is rather tedious input development, since typically several Control Functions may be needed to define a CHF correlation for a single surface and correlations must be defined for each surface independently, because the arguments such as pressure, quality, etc. are generally different at each surface.

7.1.9 Film Boiling

The overall heat transfer coefficient consists of a convective and a radiative part. The convective heat transfer coefficient is calculated for pool conditions and forced convection conditions. Radiation through the liquid film is calculated using the black-body radiation law. The heat transfer coefficient in saturated film boiling is equal to (see [136]):

$$h_{FB} = h_{conv} \cdot F + 0.75 h_{rad}$$

h_{conv} convective heat transfer coefficient, (W/m^2/K)
h_{rad} radiative heat transfer coefficient, (W/m^2/K)
F void fraction-dependent multiplier

When the heat transfer coefficient, h, is known, the heat flux is calculated as:

$$q_{FB} = h_{FB}(T_{wall} - T_{sat})$$

Calculation of the convective and radiative heat transfer coefficient is described below.

- Convective heat transfer

  To calculate convective heat transfer coefficient, the following general formula is used:

$$h_{conv} = C\left(\frac{k_{vap}^3 \rho_{vap} g(\rho_{liq} - \rho_{vap})(h_{l-v} + 0.4 c_{p,vap}\Delta T_{sat}}{\eta_{vap} D_0 \Delta T_{sat}}\right)$$

  k_{vap} vapor conductivity, (W/m/K)
  ρ_{liq} liquid density, (kg/m^3)
  ρ_{vap} vapor density, (kg/m^3)
  $h_{l\text{-}v}$ latent heat of vaporization, (J/kg)
  $c_{p,\,vap}$ specific heat of vapor, (J/kg/K)
  ΔT_{sat} wall superheat, (K), equal to: wall temperature minus saturation temperature
  η_{vap} viscosity of vapor, (kg/m/s)
  C constant, (-)

  The values of the constant C and the characteristic dimension D_0 depend on geometry, and are given below.

  - *Vertical walls*. The Bromley correlation is used [146]. In this correlation the characteristic dimension is the wall height and the constant C is equal to 0.625 - 0.883 ([55], section 3.2). The lower limit for C is applied for calculations:

$$D_0 = H \quad (height)$$
$$C = 0.625$$

- *Horizontal plates (facing up)*. The Berenson correlation is used [147], with D_0 given by the wavelength group ([55], section 3.3):

$$D_0 = \sqrt{\sigma / g(\rho_{liq} - \rho_{vap})}$$
$$C = 0.425$$

- *Horizontal plates (facing down)*. The applicability of Berenson correlation is extended to down-facing surfaces, but for those surfaces the width is used as a characteristic dimension:

$$D_0 = W \quad (width)$$
$$C = 0.425$$

- *Horizontal cylinders and spheres*. The Bromley correlation is used [146]. The values of C and D_0 are given by ([55], section 3.4):

$$D_0 = D \quad (diameter)$$
$$C = 0.620$$

In the formula for convective heat transfer coefficient, the expression: $0.4 \cdot c_{p,\,gas} \cdot \Delta T_{sat}$ is added to the evaporation enthalpy. Small differences may be encountered in the value of the constant, used in this expression. The values recommended in reference [55] are 0.35 and 0.5 for Bromley and Berenson correlations, respectively. Rohsenow recommends 0.4 for both correlations. The code uses a single value of 0.4 for both correlations.

- Radiative heat transfer

The black-body radiation law is used for the calculations. The radiative heat flux is equal to:

$$q_{rad} = \sigma_{S-B}(T_{wall}^4 - T_{sat}^4)$$

where σ_{S-B} is the Stefan-Boltzmann constant, (W/m^4/K), equal to $5.67 \cdot 10^{-8}$ [32]. The radiative heat transfer coefficient is calculated from:

$$h_{rad} = \frac{q_{rad}}{T_{wall} - T_{sat}} = \sigma_{S-B}(T_{wall} + T_{sat})(T_{wall}^2 + T_{sat}^2)$$

- Void fraction-dependent multiplier

The full film boiling correlation is used when the void fraction is below the critical value, α_1, defined below. For larger void fractions a 3-rd order interpolation is performed to a mist flow at high void fraction ($\alpha = 0.99$). The heat transfer coefficient obtained from the film boiling correlation is multiplied by the following void fraction-dependent factor:

$$F = (3 - 2 \cdot X) \cdot X^2$$

where:

$$X = \frac{1-\alpha}{1-\alpha_1}$$

Here α is the void fraction and α_1 is the critical value, above which the convective part of the heat transfer coefficient correlation is reduced. The value of α_1 is defined by the user (see Volume 2, input parameter VFL0HT). The multiplier, *F*, is plotted in Figure 7-23.

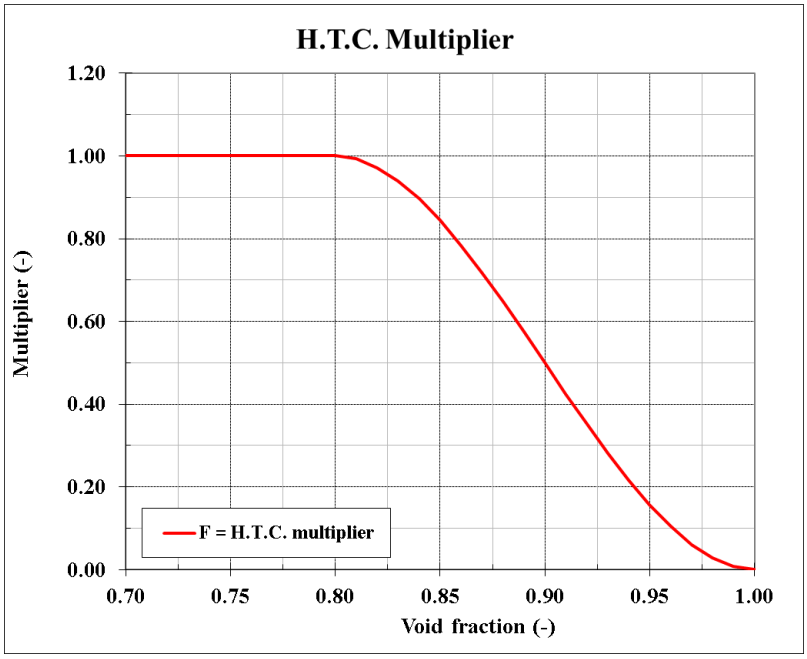


Figure 7-23 Void fraction-dependent multiplier for film boiling correlation, α_1 = 0.80

- Forced convective film boiling

The above formulae are applicable for saturated film boiling. In case of forced flow of subcooled liquid the film boiling heat transfer coefficient is higher. In SPECTRA the following formula is used:

$$q = q_{FB} + A_{SFB} \cdot h_{conv} \cdot \left(T_{sat} - T_{fluid}\right)$$

where q_{FB} is the saturated film boiling heat flux while h_{conv} is a convective heat transfer coefficient, equal to: Max(h_{FC}, h_{NC}) with h_{FC}, h_{NC} being the forced and the natural convective heat transfer coefficients. A_{SFB}, (-) is a user-defined coefficient with default value of 0.0.

7.1.10 Transition Boiling

The transition boiling model is used in the region between nucleate boiling and film boiling. The limiting heat flux in nucleate boiling is the critical heat flux, q_{CHF}. In the film boiling, it is the minimum film boiling q_{MFB}. Since the heat flux should be a continuous function of wall temperature, the transition boiling model must provide a smooth transition between the critical heat flux point, *CHF*, and the minimum heat flux point, *MFB* (Figure 7-24).

The heat flux in the transition region is calculated using the Kalinin correlation [128]. The basic formula is:

$$q = q_{CHF}\, a_{wet} + q_{MFB} a_{dry}$$

where the values of a_{dry} and a_{wet} must fulfill the following condition: $a_{dry} + a_{wet} = 1$. The formula for a_{dry}, as proposed by Kalinin et al. is:

$$a_{dry} = 1 - \left(\frac{\Delta T_{MFB} - \Delta T_{wall}}{\Delta T_{MFB} - \Delta T_{CHF}} \right)^7$$

ΔT_{MFB} wall superheat at minimum film boiling, (K), equal to: $T_{MFB} - T_{sat}$
ΔT_{CHF} wall superheat at critical heat flux, (K), equal to $T_{CHF} - T_{sat}$
ΔT_{wall} wall superheat, (K), equal to $T_{wall} - T_{sat}$

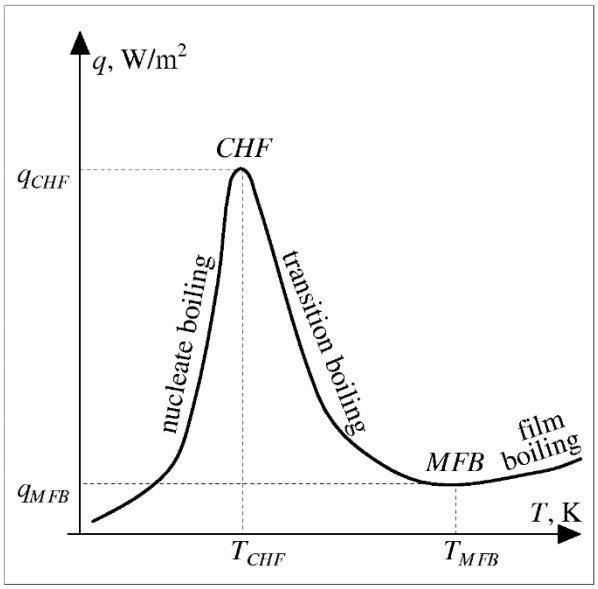


Figure 7-24 Transition boiling range

The factors in the Kalinin correlation may be written in the form:

$$a_{wet} = \left(\frac{\Delta T_{MFB} - \Delta T_{wall}}{\Delta T_{MFB} - \Delta T_{CHF}} \right)^7$$

$$a_{dry} = 1 - a_{wet}$$

The heat transfer coefficient in the transition boiling is obtained from:

$$h = \frac{q_{CHF} a_{wet} + q_{MFB} a_{dry}}{T_{wall} - T_{sat}}$$

7.1.11 Leidenfrost Transition

The transition from a dry heating surface (film boiling) to a partially wetted heating surface (transition boiling), is called the Leidenfrost transition [55]. As shown in section 7.1.10 above, to calculate the heat transfer in transition boiling one must know the parameters at the Leidenfrost transition point, namely the heat flux, q_{MFB}, and the temperature, T_{MFB}. This section provides a description how those parameters are calculated.

Two correlations are available for calculating T_{MFB}: Simon and Berenson. The one giving a larger value of T_{MFB} is selected.

- *Simon correlation for minimum film boiling*

  The minimum film boiling temperature is calculated from the correlation of Simon et al. [149]:

$$\frac{T_{MFB}}{T_{crit}} = 0.13 \frac{p}{p_{crit}} + 0.86$$

  T_{crit} critical temperature, (K), for water: T_{crit} = 647.3 [31]
  p_{crit} critical pressure, (Pa), for water: $p_{crit} = 2.212 \cdot 10^7$ [31]

- *Berenson correlation for minimum film boiling*

  The minimum film boiling temperature is calculated from the correlation of Berenson [147]:

$$T_{MFB} = T_{sat} + C_{MFB} \cdot \frac{\rho_{vap} h_{l-v}}{k_{vap}} \cdot \left(\frac{g(\rho_{liq} - \rho_{vap})}{\rho_{liq} + \rho_{vap}} \right)^{2/3} \cdot \left(\frac{\sigma}{g(\rho_{liq} - \rho_{vap})} \right)^{0.5} \cdot \left(\frac{\mu_{vap}}{g(\rho_{liq} - \rho_{vap})} \right)^{1/3}$$

  T_{sat} saturation temperature, (K)
  ρ density, (kg/m^3)
  $h_{l\text{-}v}$ evaporation enthalpy, (J/kg)
  k thermal conductivity (W/m-K)
  μ dynamic viscosity (kg/m-s)
  g acceleration of gravity (m/s^2)
  vap, liq subscripts indicating vapor and liquid

C_{MFB} constant (user defined input parameter CMFBSC, default value 0.127)

The correlations assume that the transition point is solely determined by the properties of the boiling fluid. Experimental results indicate that the phenomenon is more complicated [55]. Therefore the current approach is considered to be a good first approximation, but not a generally recommended model for calculations in which Leidenfrost transition is of primary interest.

7.1.12 Boiling Hysteresis

The boiling curve hysteresis is defined following the observation of Ramilison and Lienhard [223], see Figure 7-25. If the hysteresis is used, then the transition boiling curve is different when the wall temperature increases and different when it decreases. With increasing wall temperature, the transition boiling curve leads from the Critical Heat Flux (CHF) point to the Minimum Film Boiling (MFB) point. When the temperature decreases the transition boiling curve leads from MFB to $q_{\max}$, defined as follows:

$$q_{\max} = Min(q_{CHF}, q_{MFB} \times C_{hyst})$$

Here C_{hyst} is a constant, defined by the user (Volume 2). The effect of C_{hyst} is shown in Figure 7-26. Calculations shown in the figures were performed using 1 bar pressure and different values of C_{hyst}. The wall temperature was increased and then decreased and the resulting heat flux was plotted as a function of wall temperature. As can be seen, for this test, the value of C_{hyst} at or above which both curves overlap is equal to 50. Therefore, in order to eliminate the boiling hysteresis at all conditions, the user should define a large value of C_{hyst}.

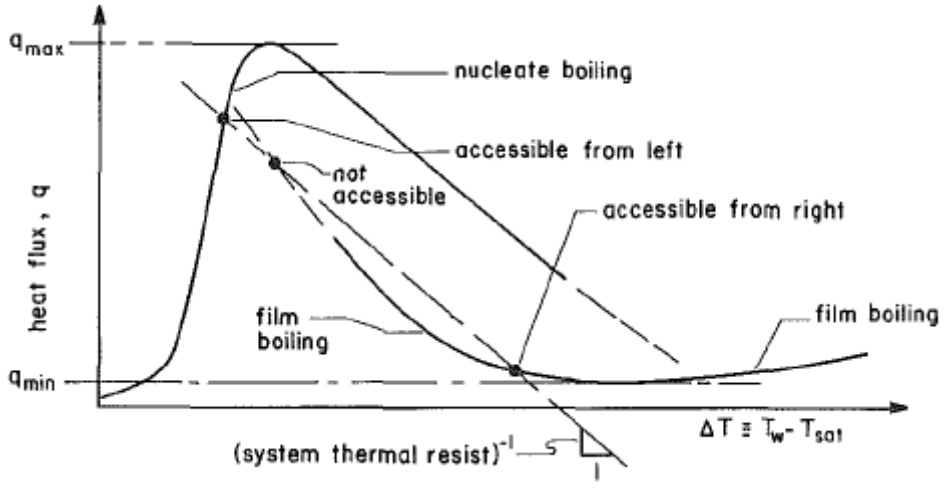


Figure 7-25 Boiling curve, Ramilison and Lienhard [223]

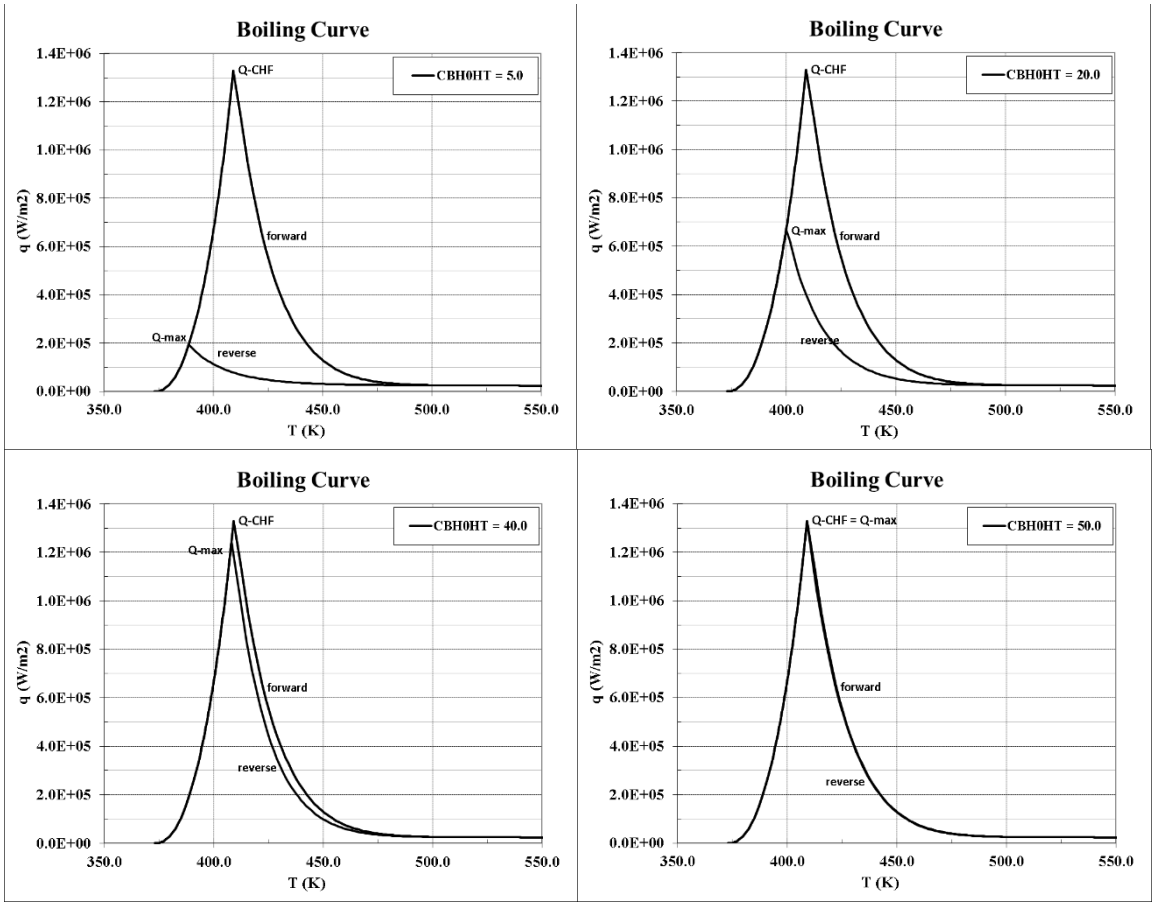


Figure 7-26 Boiling hysteresis, effect of C_{hyst}

7.1.13 Heat Transfer in Two-Phase Flow

The description of the heat transfer to two phase flow consists of two cases: bubbly flow, and mist flow.

- Bubbly flow

  The heat transfer rates in the two-phase, forced convective flow are typically correlated in the form [136]:

$$\frac{h_{c,TP}}{h_{c,liq}} = F(\chi_{tt})$$

  $h_{c,\,TP}$ convective two-phase heat transfer coefficient, (W/m^2/K)
  $h_{c,\,liq}$ convective heat transfer coefficient based on liquid component flow
  $F(\chi_{tt})$ multiplier, which depends on the Martinelli parameter

  The *F*-factor is calculated using the modified Collier-Pulling correlation, described in section 7.1.5. The Martinelli parameter, χ_{tt}, is defined as follows [136]:

$$\chi_{tt} = \left(\frac{\eta_{liq}}{\eta_{vap}}\right)^{0.1}\left(\frac{\rho_{vap}}{\rho_{liq}}\right)^{0.5}\left(\frac{1-X}{X}\right)^{0.9}$$

η_{liq} liquid viscosity, (kg/m/s)
η_{vap} vapor viscosity, (kg/m/s)
ρ_{liq} liquid density, (kg/m^3)
ρ_{vap} vapor density, (kg/m^3)
X quality, (-)

In the numerical implementation of the Martinelli parameter calculation, the following limits are imposed on the quality: $10^{-6} \le X \le 1.0 - 10^{-6}$.

The *F*-factor is called the two-phase Reynolds number factor, because in case of forced convection $h \sim Re^{0.8}$, and:

$$\frac{h_{c,TP}}{h_{c,liq}} = \left(\frac{\mathrm{Re}_{TP}}{\mathrm{Re}_{liq}}\right)^{0.8} = \left(\frac{\mathrm{Re}_{TP}}{G(1-X)D/\eta_{liq}}\right)^{0.8}$$

Re_{TP} two-phase flow Reynolds number, (-)
Re_{liq} pure liquid Reynolds number, (-)
G total (steam and water) mass flux, (kg/m^2/s)
X quality, (-)
D hydraulic diameter, (m)
η_{liq} viscosity of liquid, (kg/m/s)

Using the above relation the factor *F* may be related to the Reynolds numbers as follows [136]:

$$F(\chi_{tt}) = \left(\frac{\mathrm{Re}_{TP}}{\mathrm{Re}_{liq}}\right)^{0.8}$$

- Mist flow

In case of mist flow the Dougal-Rohsenow correlation [150] is applied:

$$\mathrm{Re} = \frac{\mathrm{Re}_{gas}}{X}\cdot\left(X + (1-X)\frac{\rho_{gas}}{\rho_{liq}}\right)$$

where ρ_{gas} is the density of the atmosphere gas, ρ_{liq} is the density of the droplets, and Re_{gas} is the Reynolds number for pure gas (X=1). The value of Re_{gas} has to be divided by X, because it is calculated in the program using the gas mass flux, while in the Dougal-Rohsenow correlation the total mass flux is needed (see [151], chapter 18, section 4.2).

7.1.14 Non-Equilibrium Mass Transfer

Nonequilibrium mass transfer occurs in two cases:

- If the temperature of water exceeds the saturation temperature at a given total pressure, then nonequilibrium boiling (flashing) occurs.
- If the temperature of gas is below the saturation temperature at steam partial pressure, then nonequilibrium condensation (fogging) occurs.

Very limited data exists on the mechanism of nonequilibrium mass transfer. Therefore a rather simple model is used. The model is based on the theoretical and experimental investigation from Friz [129]. The models of nonequilibrium boiling and condensation are described below.

- Nonequilibrium boiling (bulk boiling, flashing)

  The mass transfer rate during flashing is approximately proportional to the water superheat squared [129]. The volumetric mass transfer rate, Γ_{boil}, (kg/s/m^3), is calculated from:

$$\Gamma_{boil} = \begin{cases} C_{BB}\rho_{liq}(T_{liq} - T_{sat})^2 & if \quad T_{liq} > T_{sat} \\ 0.0 & if \quad T_{liq} \le T_{sat} \end{cases}$$

  ρ_{liq} density of liquid, (kg/m^3)
  C_{BB} bulk boiling constant, (1/s/K^2)

  Based on comparison with ISPRA experimental data of Friz [129], the recommended value of C_{BB} is 0.005. This is the default value of C_{BB}, but it may be redefined for each Control Volume via input data (see Volume 2).

  The enthalpy changes for steam bubbles and liquid, associated with the process, are as follows:
  - The steam (bubble) enthalpy source is associated with the created steam and is equal to: $\Gamma_{boil} \cdot h_{vap}$, where h_{vap} is the specific enthalpy of saturated vapor, (J/kg).
  - The water (pool) enthalpy sink is associated with the loss of liquid and with the evaporation enthalpy, which is assumed to be totally taken from the liquid pool. The enthalpy sink associated with the loss of liquid is equal to: $-\Gamma_{boil} \cdot h_{liq}$, The energy required for evaporation of water is equal to: $\Gamma_{boil} \cdot (h_{vap} - h_{liq})$. Total enthalpy loss for water is equal to: $-\Gamma_{boil} \cdot h_{liq} - \Gamma_{boil} \cdot (h_{vap} - h_{liq}) = -\Gamma_{boil} \cdot h_{vap}$, and its absolute value is identical to the enthalpy source for the bubbles.

- Nonequilibrium condensation (bulk condensation, fogging)

  The mass transfer rate during fogging, Γ_{cond}, (kg/s/m^3), is calculated from:

$$\Gamma_{cond} = \begin{cases} C_{BC}\rho_{vap}(T_{vap} - T_{sat})^2 & if \quad T_{vap} < T_{sat} \\ 0.0 & if \quad T_{vap} > T_{sat} \end{cases}$$

  ρ_{vap} density of vapor, (kg/m^3)

C_{BC} bulk condensation constant, ($1/s/K^2$)

The convention used in the nonequilibrium model is that the condensation process gives negative Γ, while the boiling process give positive Γ. The value of the bulk condensation constant, C_{BC}, is defined via input for each control volume. The default value is the same as the value of bulk boiling constant: 0.005 (see Volume 2).

The enthalpy changes for steam bubbles and liquid, associated with the process, are as follows:

- Water (droplets) enthalpy source is equal to: $-\Gamma_{cond} \cdot h_{liq}$ (note that the value is positive, since in this case $\Gamma<0$).
- Gas (atmosphere) enthalpy change is associated with the loss of vapor and with the enthalpy released during condensation, which is assumed to be totally deposited in the gas. The enthalpy sink associated with the loss of vapor is equal to: $\Gamma_{cond} \cdot h_{vap}$. The energy released during condensation is equal to: $-\Gamma_{cong} \cdot (h_{vap}-h_{liq})$. Total enthalpy change for gas is equal to: $+\Gamma_{cond} \cdot h_{vap} - \Gamma_{cond} \cdot (h_{vap}-h_{liq}) = \Gamma_{cond} \cdot h_{liq}$, and its absolute value is identical to the enthalpy gain for liquid droplets.

7.2 Heat and Mass Transfer Logic

This section describes the logic to calculate heat and mass (H&M) transfer in five different cases. The first two cases include H&M transfer from walls (SC/TC surfaces) to atmosphere and pool. Those cases are described in sections 7.2.1 and 7.2.2 respectively. The next section brings the description of H&M transfer from a pool surface. Finally, the last three sections provides the description of H&M transfer from the surface of dispersed components: droplets and bubbles.

7.2.1 Wall-Atmosphere Heat and Mass Transfer Logic

The wall-atmosphere heat and mass transfer model includes natural and forced convection, and condensation.

- Natural and forced convection

  The natural convection heat transfer coefficient is calculated using the natural convection model (section 7.1.1). The forced convection heat transfer coefficient is calculated using the forced convection model (section 7.1.2). If droplets are present then the Dougal-Rohsenow correlation is used (section 7.1.13). Selection between the natural and the forced convection is performed by choosing the mode that gives the larger heat transfer coefficient.

$$h = Max(natural\ convection,\ forced\ convection)$$

  The convective heat flux is calculated as:

$$q_{conv} = h_{conv}(T_{wall} - T_{fluid})$$

- Condensation

  If the surface temperature is below the saturation temperature, then the condensation model, described in section 7.1.4, is used. The overall heat flux is assumed to be the sum of the convective and condensation fluxes:

$$q_{wall} = h_{conv}(T_{wall} - T_{fluid}) + h_{cond}(T_{wall} - T_{sat})$$

  where h_{cond} is the condensation heat transfer coefficient, calculated as described in section 7.1.4. (Note that the general sign convention, assumed for wall heat transfer, requires the positive net flux to be from the wall to the fluid.)

  The mass transfer flux is calculated as the condensation heat flux divided by the difference between the enthalpy of steam and the enthalpy of the liquid. It is assumed that the condensate appears with the temperature equal to the average of the wall surface temperature, and the saturation temperature for the steam partial pressure:
  $T_{cond} = ½(T_{wall} + T_{sat}(p_{H2O}))$.

$$m_{cond} = \frac{h_{cond}(T_{sat} - T_{wall})}{h_{vap}(T_{gas}, p_{H_2O}) - h_{liq}(T_{cond})}$$

$h_{vap}(T_{gas},p_{H2O})$ vapor enthalpy at the temperature T_{gas} and steam partial pressure, p_{H2O}
$h_{liq}(T_{cond})$ liquid enthalpy at the condensate temperature T_{cond}

The net heat flux from atmosphere (the flux that changes the internal energy of gas) is calculated as:

$$q_{gas} = q_{wall} - q_{cond} = q_{wall} - m_{cond}(h_{vap} - h_{liq})$$

The above terms are used by the mass and energy balance for Control Volume (section 2.4).

Condensation correlations typically use the wall subcooling ($T_{sat} - T_{wall}$). The heat transfer coefficient needed as a boundary condition for the conduction equation is defined using the difference ($T_{wall} - T_{fluid}$). Therefore when the wall heat flux, q_{wall}, is calculated, the effective heat transfer coefficient is calculated as:

$$h_{wall} = \frac{q_{wall}}{(T_{wall} - T_{fluid})}$$

An example of wall to atmosphere heat and mass transfer is shown in Figure 7-27 and Figure 7-28. The heat transfer coefficient, and the mass transfer flux, are plotted versus wall temperature and fluid velocity, for the following atmosphere parameters:

- gas temperature: 330 K
- total pressure: 1.0×10^5 Pa
- steam partial pressure: 0.1×10^5 Pa, (T_{sat} ~319 K, humidity ~60%)

In case of low gas velocities the heat transfer is governed by natural convection (Figure 7-27), with the heat transfer coefficient proportional to the wall-gas temperature difference to the power of 1/3 (laminar boundary layer) or 1/4 (turbulent boundary layer) - see section 7.1.1.

When the wall temperature is below the saturation temperature at the steam partial pressure, then condensation occurs. Condensation is influenced by the gas velocity, through the shear factor (section 7.1.4). In the present case this influence is very small, and is almost invisible in Figure 7-27 and Figure 7-28. The value of the condensation mass flux for the wall temperature equal to 300 K, increases from 2.303×10^{-3} kg/m^2/s for gas velocity equal to zero, to 2.315×10^{-3} kg/m^2/s for gas velocity equal to 2.5 m/s.

It is seen in Figure 7-27 that the heat transfer coefficient during condensation decreases, when the wall temperature approaches the saturation temperature. This seems contradictory to the condensation model, which gives heat transfer coefficient proportional to $\Delta T_{sat}^{-1/4}$ (section 7.1.4). The behavior displayed in Figure 7-27 is a consequence of the fact that the effective heat transfer coefficient is defined as the ratio of the heat flux to the wall-fluid temperature difference, while internally in the condensation model it is equal to the ratio of heat flux to ΔT_{sat}.

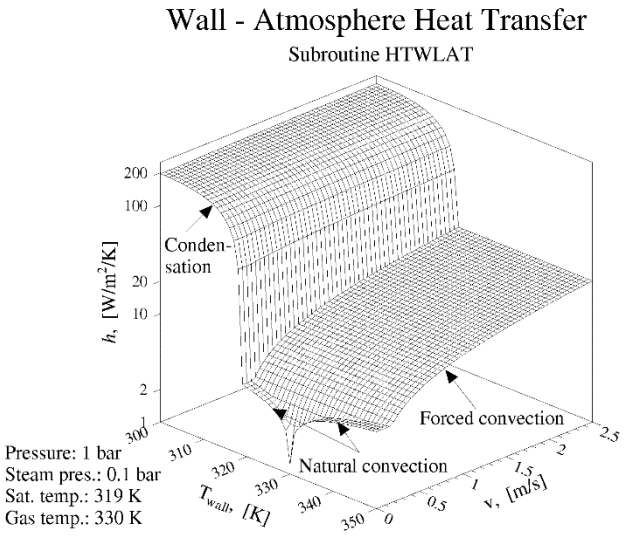


Figure 7-27 Wall-atmosphere heat transfer coefficient

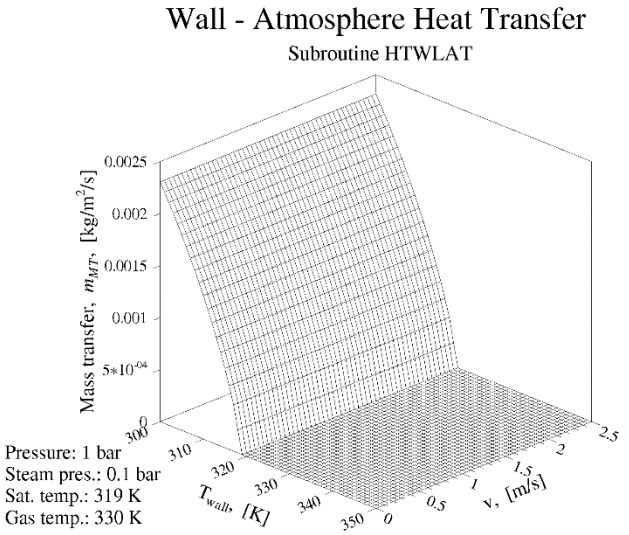


Figure 7-28 Wall-atmosphere mass flux

In the present case the saturation temperature is about 11 K lower than the gas temperature. If both saturation and gas temperatures are equal, then the proportionality of the heat transfer coefficient to $\Delta T_{sat}^{-1/4}$ is observed. This is seen in Figure 7-29, which shows results for gas temperature changed to 319 K.

The proportionality of the heat transfer coefficient to $\Delta T_{sat}^{-1/4}$ extends up to ΔT_{min} (assumed equal to 1.0 K). For ΔT_{sat} smaller than ΔT_{min} the condensation heat transfer coefficient is interpolated, as shown in Figure 7-30. This interpolation provides a continuous transition, from the fully developed condensation, to the non-condensing conditions.

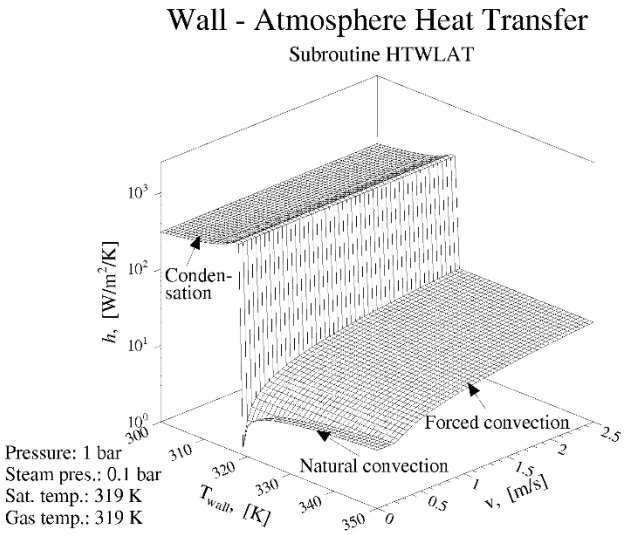


Figure 7-29 Wall-atmosphere heat flux

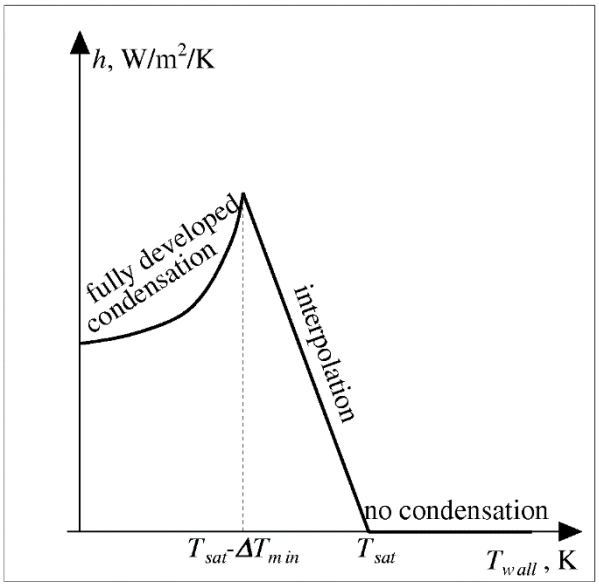


Figure 7-30 Interpolation of the condensation heat transfer coefficient

7.2.2 Wall-Pool Heat and Mass Transfer

The wall-pool heat and mass transfer model includes natural and forced convection, and boiling.

- Natural and forced convection

  The natural convection heat transfer coefficient is calculated using the natural convection model (section 7.1.1). The forced convection heat transfer coefficient is calculated using the forced convection model (section 7.1.2). If bubbles are present and the internal flow model is used, then the Reynolds number is multiplied by the two-phase Reynolds number factor, *F* (section 7.1.13). Selection between the natural and forced convection is performed by choosing the mode that gives a larger heat transfer coefficient.

$$h = Max(natural\ convection,\ forced\ convection)$$

  The convective heat flux is calculated as:

$$q_{conv} = h_{conv}(T_{wall} - T_{fluid})$$

- Boiling

  The boiling model consists of nucleate, transition, and film boiling, with the critical heat flux and minimum film boiling models. The calculation procedure is different in case of external and internal flow, and is described below.

  Nucleate boiling

  In case of external flow the Rohsenow correlation is used to calculate nucleate boiling. The total heat flux is defined as ([136], section 5.4):

$$q_{wall} = h_{conv}(T_{wall} - T_{fluid}) + q_{Roh}$$

  where q_{Roh} is the boiling heat flux, obtained from the Rohsenow correlation. In case of internal flow nucleate boiling is based on the Chen correlation. The heat flux is calculated from the following expression (see section 7.1.5):

$$q_{wall} = (hF)_{conv}(T_{wall} - T_{fluid}) + h_{F-Z} S \left(\frac{0.013}{C_{sf}} \right)^3 (T_{wall} - T_{fluid})$$

  Critical heat flux

  Critical heat flux (CHF) is calculated using the models described in section 7.1.6. In case of external flow the pool boiling model (Zuber correlation) is used, while in case of internal flow it is the combination of the Zuber correlation and the USSR Academy of Science tabulated data, or optionally the Biasi correlation. If the calculated wall heat flux is smaller than the critical heat flux, then nucleate boiling occurs and no further calculations are performed. If the critical heat flux, q_{CHF}, is exceeded, then the wall superheat at *CHF*, $\Delta T_{CHF}=(T_{CHF}-T_{sat})$, is calculated ($\Delta T_{CHF}$ is needed for the transition boiling calculation). It is calculated differently in case of external and internal flow, as shown below.

- CHF in external flow

In this case the Rohsenow correlation is used, where the wall heat flux is proportional to the wall superheat to the power of three (see section 7.1.5). Therefore:

$$q_{wall} \sim (T_{wall} - T_{sat})^3$$
$$q_{wall} \sim (T_{CHF} - T_{sat})^3$$

and ΔT_{CHF} is calculated as:

$$\Delta T_{CHF} = (T_{wall} - T_{sat})\left(\frac{q_{CHF}}{q_{wall}}\right)^3 = \Delta T_{wall}\left(\frac{q_{CHF}}{q_{wall}}\right)^3$$

- CHF in internal flow

In this case the Chen correlation is used, which consists of two parts, a convective part and a boiling part, the latter being approximately proportional to the wall superheat squared (section 7.1.5):

$$q_{wall} = h_{conv}(T_{wall} - T_{fluid}) + C(T_{wall} - T_{sat})^2$$
$$q_{CHF} = h_{conv}(T_{CHF} - T_{fluid}) + C(T_{CHF} - T_{sat})^2$$

When the constant C is calculated using the first equation and then substituted into the second equation, then a quadratic expression for $\Delta T_{CHF}=(T_{CHF}-T_{sat})$ is obtained and ΔT_{CHF} is calculated as:

$$\left(\frac{q_{boil}}{T_{wall}^2}\right)\Delta T_{CHF}^2 + (h_{conv})\Delta T_{CHF} - (q_{CHF} - h_{conv}(T_{sat} - T_{fluid})) = 0$$

where $\Delta T_{wall}=(T_{wall}-T_{sat})$. The above equation is solved for ΔT_{CHF}:

$$\Delta T_{CHF} = \frac{-h_{conv} + \left\{h_{conv}^2 + 4\frac{q_{boil}}{T_{wall}^2}\left[q_{CHF} - h_{conv}(T_{sat} - T_{fluid})\right]\right\}^{1/2}}{2\frac{q_{boil}}{T_{wall}^2}}$$

The above expression gives only an approximate value of ΔT_{CHF}, because the boiling heat transfer coefficient is not exactly proportional to ΔT_{wall}^2. When T_{wall} is close to T_{CHF}, the approximation is good. The accuracy decreases with increasing difference between T_{wall} and T_{CHF}. Since the value is only needed for the transition boiling calculation, the approximation is considered to be sufficient, in view of the fact that the other required parameter, ΔT_{MFB}, introduces a relatively large error due to uncertainties in the modelling of the Leidenfrost transition point. The important fact is that when $T_{wall} \rightarrow T_{CHF}$, then $q_{wall} \rightarrow q_{CHF}$, and thus there is no discontinuity in the boiling curve.

Film boiling

The film boiling heat flux, q_{boil}, is calculated using the film boiling model, described in section 7.1.9. In this heat transfer regime the convective flux is set to zero: q_{conv}=0.0.

Minimum film boiling

The minimum film boiling is calculated using two correlations: the Simon correlation and the Berenson correlation (section 7.1.11):

$$T_{MFB,1} = T_{MFB,Simon} \qquad q_{MFB,1} = h_{film}(T_{MFB} - T_{sat})$$
$$T_{MFB,2} = T_{MFB,Berenson} \qquad q_{MFB,2} = h_{film}(T_{MFB} - T_{sat})$$

$T_{MFB, Simon}$ minimum film boiling temperature from the Simon correlation
$T_{MFB, Berenson}$ minimum film boiling temperature from the Berenson correlation.

The minimum film boiling temperature, T_{MFB}, is obtained as the maximum of $T_{MFB,1}$ and $T_{MFB,2}$.

Transition boiling

If the wall temperature, T_{wall}, is below the minimum film boiling temperature, T_{MFB}, then the transition boiling heat flux is calculated using the transition boiling model, described in section 7.1.10. In this heat transfer regime the convective flux is set to zero: q_{conv}=0.0.

The mass transfer flux is calculated as the boiling heat flux, divided by the difference between the enthalpy of vapor and the enthalpy of water in the pool. It is assumed that the vapor appears with the temperature equal to the saturation temperature at the total pressure.

$$m_{boil} = \frac{q_{boil}}{h_{vap}(T_{sat}(p) - h_{liq}(T_{pool})}$$

where $h_{vap}(T_{sat}(p))$ is the saturated vapor enthalpy, $h_{liq}(T_{pool})$ is the liquid enthalpy at the pool temperature.

The net heat flux to liquid (flux that changes the internal energy of liquid) is calculated as:

$$q_{liq} = q_{wall} - q_{boil} = q_{wall} - m_{boil}(h_{vap} - h_{liq})$$

Boiling correlations typically use the wall superheat (T_{wall}–T_{sat}). The heat transfer coefficient needed as a boundary condition for the conduction equation is defined using (T_{wall}–T_{fluid}). Therefore when the wall heat flux, q_{wall}, is calculated, the effective corresponding heat transfer coefficient is calculated as:

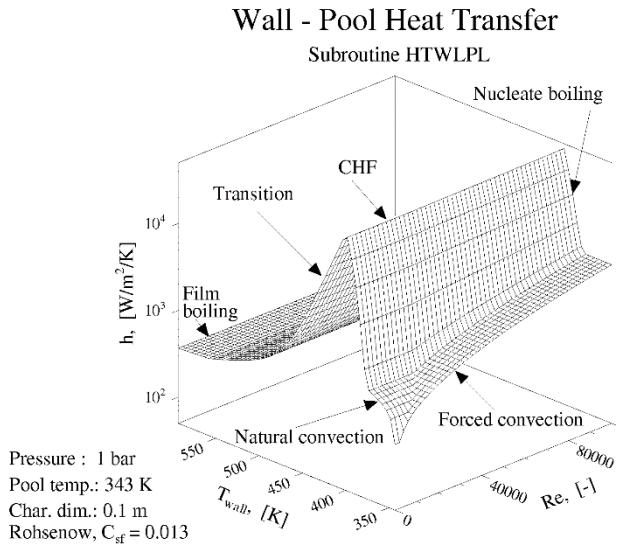


Figure 7-31 Wall-pool heat transfer coefficient versus *T* and *Re*, pool boiling

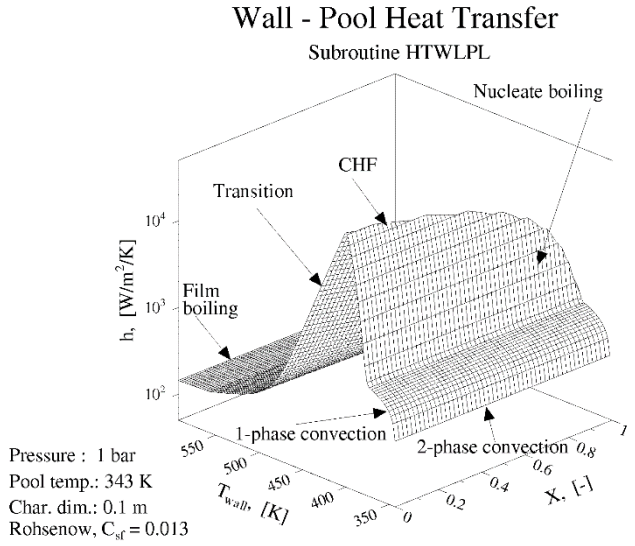


Figure 7-32 Wall-pool heat transfer coefficient versus *T* and *X*, pool boiling

$$h_{wall} = \frac{q_{wall}}{(T_{wall} - T_{fluid})}$$

Examples of wall to pool heat transfer are shown in Figure 7-31, Figure 7-32, Figure 7-33, and Figure 7-34. The heat transfer coefficient is plotted versus wall temperature and fluid velocity (through *Re* - Figure 7-31, Figure 7-33), as well as wall temperature and quality (Figure 7-32, Figure 7-34). The first two figures (Figure 7-31, Figure 7-32) give the heat transfer coefficient for the external flow ("pool boiling curve"), while the next two figures (Figure 7-33, and Figure 7-34) give the heat transfer coefficient for the internal flow ("convective boiling curve").

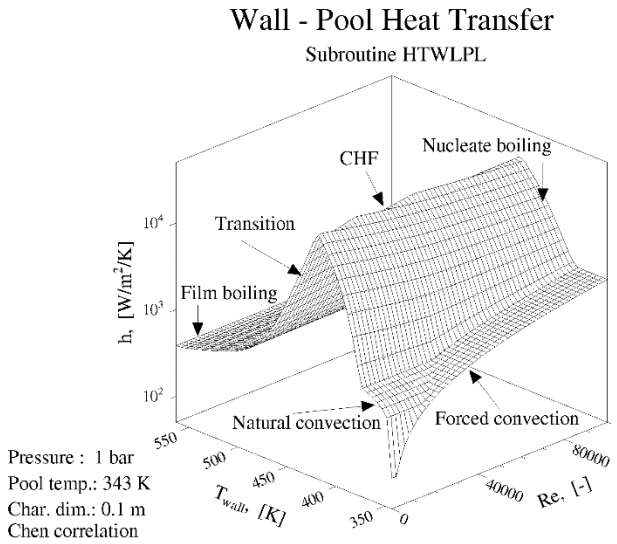


Figure 7-33 Wall-pool heat transfer coefficient versus *T* and *Re*, convective boiling

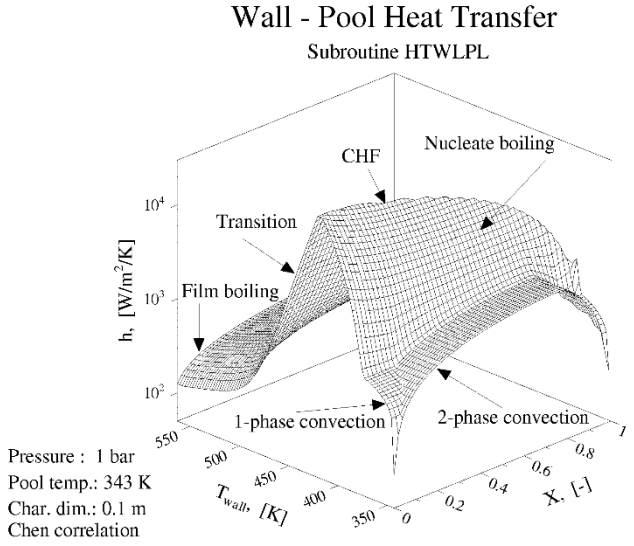


Figure 7-34 Wall-pool heat transfer coefficient versus *T* and *X*, convective boiling

The following pool parameters were assumed for calculations:

- Pool temperature: 343 K
- Total pressure: 1.0×10^5 Pa, (saturation temperature ~373 K)

Comparison of Figure 7-31 and Figure 7-33 shows that the fluid velocity has an influence on *CHF* only in case of convective boiling. This is because in case of the pool boiling model the Zuber *CHF* correlation is used, while in the convective boiling model a combination of Zuber and the flow dependent USSR Academy of Science or Biasi correlations, is used (section 7.1.6).

Comparison of Figure 7-32 and Figure 7-34 shows that the flow quality has an influence on the heat transfer coefficient only in case of convective boiling. This is because only in case of convective boiling the two phase Reynolds number factor, *F*, is used.

In both models *CHF* decreases with increasing quality. The decrease is linear in case of pool boiling, and approximately linear in case of convective boiling. This linear dependence is not clear in Figure 7-32 and Figure 7-34 , because the graphs are made in logarithmic scale.

7.2.3 Pool-Atmosphere Heat and Mass Transfer

The calculation of convective heat transfer and mass transfer (evaporation or condensation) from the pool surface to the atmosphere of a Control Volume is described below.

- Heat transfer

  The convective heat transfer coefficient is calculated similarly as in case of wall-to atmosphere heat transfer. The pool surface is treated as a horizontal flat surface. The convective heat transfer coefficient is calculated using the two models: the natural convection model (section 7.1.1), and the forced convection model (section 7.1.2). Selection between the natural and forced convection is performed by choosing the mode that gives larger heat transfer coefficient:

$$h = Max(natural\ convection, forced\ convection)$$

  The interphase velocity used to compute the forced convective correlation is taken as:

$$v_{\text{int}} = Max(\sqrt{v_{hor,pool}^2 + v_{hor,atms}^2}, v_{\text{min},pool-atms})$$

  Here $v_{hor,\ pool}$ and $v_{hor,\ atms}$ are the horizontal velocities of pool and atmosphere in a Control Volume. The $v_{min,\ pool\text{-}atms}$ is a minimum velocity for interphase heat transfer calculation (user-defined parameter VINTCV(IPLAT), with a default value of 0.0 - see Volume 2). The convective heat flux is calculated as:

$$q_{HT} = h_{conv}(T_{liq} - T_{gas})$$

- Mass transfer

  The mass flux is calculated using the analogy between heat and mass transfer ([21], section 9.4.4). The mass transfer coefficient is obtained using the same correlations as those for the heat transfer coefficient but with Prandtl number replaced by Schmidt number and Nusselt number replaced by Sherwood number:

$$Pr \rightarrow Sc$$
$$Nu \rightarrow Sh$$

  The mass transfer coefficient, K_M, (m/s), is obtained from the Sherwood number, similarly as the heat transfer coefficient from the Nusselt number:

  - Natural convection: $K_M = Sh_{NC} \cdot D_{C,f} / d_{NC}$

- Forced convection: $K_M = Sh_{FC} \cdot D_C / d_{FC}$.

D_C and $D_{C,f}$ are the diffusion coefficients for bulk and film temperature respectively, while d_{NC}, d_{FC}, are the characteristic dimensions for natural and forced convection respectively. Similarly as in case of heat transfer the correlation is chosen that gives larger mass transfer coefficient.

The mass transfer flux is obtained from the following formula ([136], section 10.3.4):

$$m_{MT} = \frac{K_M \rho_{vap}}{p_{am}} \left(p_{sat}(T_{liq}) - p_{vap} \right)$$

m_{MT} mass transfer flux, (kg/m^2/s)
ρ_{vap} saturated vapor density at total pressure, (kg/m^3)
$p_{sat}(T_{liq})$ steam pressure at the interphase, (Pa), equal to the saturation pressure at the liquid temperature
p_{vap} steam pressure in the bulk atmosphere, (Pa)
p_{am} log mean pressure, defined by:

$$p_{am} = \frac{(p - p_{sat}) - (p - p_{vap})}{\ln\left(\frac{p - p_{sat}(T_{liq})}{p - p_{vap}} \right)}$$

where p is the total pressure. Substitution of p_{am} definition into the equation for m_{MT} gives the final formula used by the program:

$$m_{MT} = K_M \rho_{vap} \ln\left(\frac{p - p_{vap}}{p - p_{sat}(T_{liq})} \right)$$

In the program a special procedure is provided to assure smooth curve and avoid floating exception when either the numerator or the denominator approaches zero. The convention assumed here is that positive mass transfer flux means evaporation of water from the pool to the atmosphere of a Control Volume.

The energy fluxes associated with the mass flux are calculated differently for the case of evaporation and condensation, as described below.

- Evaporation, $p_{vap} < p_{sat}(T_{liq})$; $m_{MT} > 0.0$

It is assumed that the vapor appears with the temperature equal to the pool surface temperature. The enthalpy of vapor is therefore equal to the saturated steam enthalpy at the pool temperature, $h_{sat.vap}(T_{pool})$. The enthalpy of disappearing water is $h_{liq}(T_{pool})$. The difference between the enthalpy gained by the gas, and lost by the liquid, gives the net heat effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{sat.vap}(T_{pool}) - h_{liq}(T_{pool})]$$

In the above formula T_{pool} is the pool surface temperature, which may be different from the average pool temperature if stratification models are active. The net energy flux, q_{MT}, is "consumed" during the evaporation process. This energy is assumed to be taken from the pool (upper layer of the pool when stratification models are active).

- Condensation, $p_{vap} > p_{sat}(T_{liq})$; $m_{MT} < 0.0$

It is assumed that the condensate appears with the temperature equal to the atmosphere gas temperature. The enthalpy of the condensate is therefore equal to the liquid enthalpy at the atmosphere temperature, $h_{liq}(T_{atms})$. The enthalpy of disappearing steam is equal to $h_{gas}(T_{atms}, p_{vap})$, where p_{vap} is steam partial pressure in the bulk atmosphere. The difference between the enthalpy lost by the gas, and gained by the liquid, gives the net effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{gas}(T_{atms}, p_{vap}) - h_{liq}(T_{atms})]$$

The value of q_{MT}, energy flux is "released" during the process. The energy is assumed to be deposited in the pool (upper layer of the pool when stratification models are active).

Examples of the pool-atmosphere heat and mass transfer are shown in Figure 7-35, Figure 7-36, and Figure 7-37. The convective heat flux, q_{HT}, the mass transfer flux, m_{MT}, and the energy flux accompanying the mass transfer, q_{MT}, are plotted versus pool temperature and gas velocity.

The following parameters were assumed for calculations:

- gas temperature: 330 K
- total pressure: 1.0×10^5 Pa
- steam partial pressure: 0.1×10^5 Pa, (saturation temperature ~319 K, humidity ~60%)

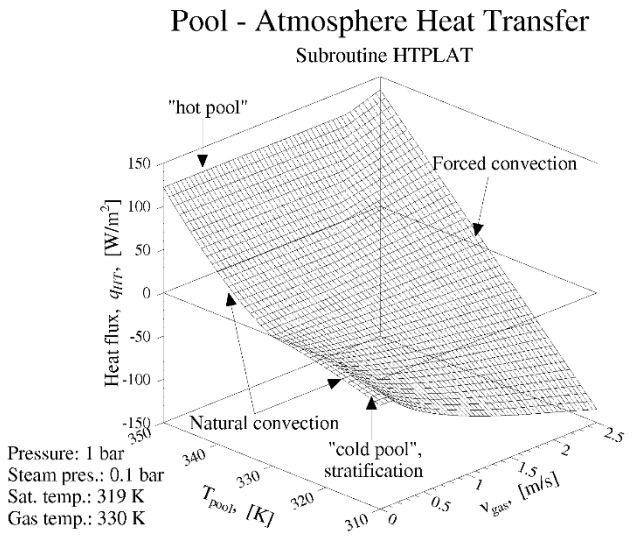


Figure 7-35 Pool-atmosphere convective heat flux, q_{HT}

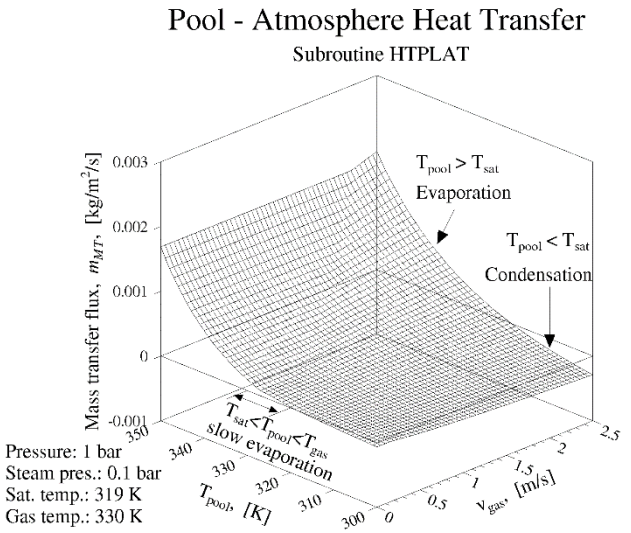


Figure 7-36 Pool-atmosphere mass transfer flux, m_{MT}

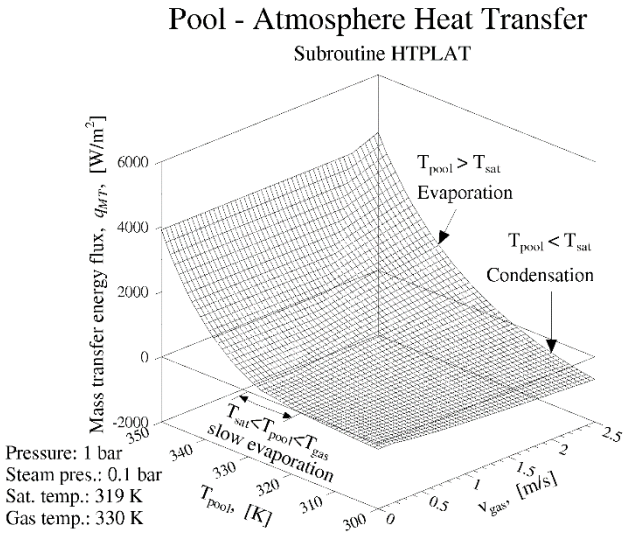


Figure 7-37 Pool-atmosphere mass transfer energy flux, q_{MT}

Figure 7-35 shows the convective heat flux. It is seen that in case of natural convection (small v_{gas}) the heat flux is clearly larger for warm pool ($T_{pool}>T_{gas}$=330 K), than the cold pool ($T_{pool}<T_{gas}$=330 K). If the pool is cold, a stratified cold layer of gas is created above the pool, and the heat transfer is low. In such case the code uses the correlation for horizontal plates "closed" for natural circulation (section 7.1.1) to calculate the convective heat transfer coefficient.

Figure 7-36 shows the calculated mass transfer flux. When the pool temperature is above the steam saturation temperature (319 K), the mass flux is positive - water evaporates from the pool surface. When the pool temperature is below the steam saturation temperature, the mass flux is negative - steam condenses on the pool surface. Note that evaporation occurs even when the pool temperature is below the gas temperature ($319 < T_{pool} < 330$). In this range the evaporation rate is small, especially for low pool-gas relative velocities.

Figure 7-37 shows the heat flux associated with the mass transfer. Since the energy flux is approximately equal to the mass flux times the latent heat of evaporation this plot is very similar to the mass transfer plot, only with different units at the vertical axis. Comparison of Figure 7-35 and Figure 7-37 shows that the mass transfer energy flux is in this case much larger than the convective heat flux.

7.2.4 Droplet-Atmosphere Heat and Mass Transfer

The heat and mass transfer from droplet surface to atmosphere is calculated based on the Nusselt number and the Sherwood number correlations of Ranz and Marshall [152], with correction factors introduced by Downing [153].

The Ranz and Marshall correlations for Nusselt and Sherwood number are:

$$Nu = 2.0 + 0.6 Re^{1/2} Pr^{1/3}$$
$$Sh = 2.0 + 0.6 Re^{1/2} Sc^{1/3}$$

It has been found out by Downing that when the gas temperature is significantly higher than the droplet temperature, the Ranz and Marshall correlations overpredict heat and mass transfer. Downing introduced correction factors, defined as follows:

$$L = \frac{\log(1+B)}{B}$$
$$M = 1 - 0.4\left(1 - \frac{T_{drop}}{T_{gas}}\right)$$
$$N = 1 - 0.4(1 - L)$$

where the value of B is calculated as:

$$B = \frac{c_{p,vap}(T_{gas} - T_{drop})}{h_{l-v}}$$

In this formula $c_{p,vap}$ is specific heat of vapor, and $h_{l\text{-}v}$ is the latent heat of vaporization. With the above correction factors the correlations are ([153], equations 16):

$$Nu = (2.0 + 0.6Re^{1/2}Pr^{1/3})LMN$$
$$Sh = (2.0 + 0.6Re^{1/2}Sc^{1/3})M$$

The correction factors have a significant influence on the results if T_{gas} is clearly higher than T_{drop} (see [153] figures 4, 5, 6).

Schwarz and Smolik [154] have recently reviewed a large number of correlations and compared them with experimental data for evaporation from droplets. Their conclusion was that the correlation of Downing gives generally good agreement with experimental data. Therefore the Downing correction factors are used in the SPECTRA model. The procedure to calculate heat and mass transfer is described below.

- Heat transfer

  The Nusselt number is calculated using the Ranz and Marshall correlation with the Downing correction factors:

  $$Nu = (2.0 + 0.6Re^{1/2}Pr^{1/3})LMN$$

  The correction factors, *L M N*, are used only if the gas temperature is above the droplet temperature ($T_{gas} > T_{drop}$). If the droplet temperature is larger than the gas temperature, then the correction factors are set to one: *L M N* = 1.0. The heat transfer coefficient is equal to:

  $$h = Nu\frac{k_{gas}}{D_{drop}}$$

  where k_{gas} is the thermal conductivity of gas at the bulk temperature, and D_{drop} is the droplet diameter. The interphase velocity used to compute the Reynolds number is taken as:

  $$v_{\text{int}} = Max(\sqrt{(v_{hor,drop} - v_{hor,atms})^2 + (v_{ver,drop} - v_{ver,atms})^2}, v_{\min,drop-atms})$$

  Here $v_{hor,\,drop}$, $v_{hor,\,atms}$, $v_{ver,\,drop}$, $v_{ver,\,atms}$, are the horizontal and vertical velocities of droplets and atmosphere in a Control Volume. The $v_{min,\,drop\text{-}atms}$ is a minimum velocity for interphase heat transfer calculation (user-defined parameter VINTCV(IDPAT), with a default value of 0.0 - see Volume 2). The convective heat flux is calculated as:

  $$q_{HT} = h(T_{drop} - T_{gas})$$

- Mass transfer

  The mass transfer Sherwood number is calculated using the Ranz and Marshall correlation with the Downing correction factor:

  $$Sh = (2.0 + 0.6Re^{1/2}Sc^{1/3})M$$

The correction factor M is used only if the gas temperature is above the droplet temperature ($T_{gas}>T_{drop}$). If the droplet temperature is higher than the gas temperature, then M is set to 1.0. The mass transfer coefficient, K_M, (m/s), is obtained from Sherwood number:

$$K_M = Sh\frac{D_C}{D_{drop}}$$

where D_C is the diffusion coefficient for steam in the bulk atmosphere, calculated as shown in section 3.5.2.3. The mass transfer flux is obtained from ([136], section 10.3.4):

$$m_{MT} = \frac{K_M \rho_{vap}}{p_{am}}(p_{sat}(T_{drop}) - p_{vap})$$

ρ_{vap}	saturated vapor density at total pressure, (kg/m^3)
$p_{sat}(T_{drop})$	steam pressure at the interphase, (Pa), equal to the saturation pressure at the droplet temperature
p_{vap}	steam pressure in the bulk atmosphere, (Pa)
p_{am}	log mean pressure, defined above in section 7.2.3.

Substitution of p_{am} definition into the equation for m_{MT} gives the formula used by the program:

$$m_{MT} = K_M \rho_{vap} \ln\left(\frac{p - p_{vap}}{p - p_{sat}(T_{drop})}\right)$$

The energy fluxes associated with the mass flux are calculated differently for the case of evaporation and condensation, as described below.

- Evaporation, $p_{vap} < p_{sat}(T_{drop})$; $m_{MT} > 0.0$
It is assumed that the vapor appears with the temperature equal to the droplet surface temperature. The enthalpy of the vapor is therefore equal to $h_{sat.vap}(T_{drop})$. The enthalpy of the disappearing water is $h_{liq}(T_{drop})$. The difference between the enthalpy flux gained by the gas, and lost by the droplet, gives the net heat effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{sat.vap}(T_{pool}) - h_{liq}(T_{pool})]$$

The value of q_{MT} represents the energy flux that is "consumed" during the process. This energy is assumed to be taken from the droplet.

- Condensation, $p_{vap} > p_{sat}(T_{drop})$; $m_{MT} < 0.0$
It is assumed that the condensate appears with the temperature equal to the atmosphere gas temperature. The enthalpy of the condensate is therefore equal to the liquid enthalpy at the atmosphere temperature, $h_{liq}(T_{atms})$. The enthalpy of disappearing steam is equal to $h_{gas}(T_{atms}, p_{vap})$, where p_{vap} is steam partial pressure in the bulk atmosphere. The difference between the enthalpy gained by the gas, and lost by the liquid, gives the net effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{gas}(T_{atms}, p_{vap}) - h_{liq}(T_{atms})]$$

The value of q_{MT} represents the energy flux that is "released" during the process. This energy is assumed to be deposited in the droplet.

An example of the droplet-atmosphere heat and mass transfer is shown in Figure 7-38, Figure 7-39, Figure 7-40. The convective heat flux, q_{HT}, the mass transfer flux, m_{MT}, and the energy flux accompanying the mass transfer, q_{MT}, are plotted versus droplet temperature and gas-droplet relative velocity.

The following parameters were assumed for calculations:

- gas temperature: 330 K
- total pressure: 1.0×10^5 Pa
- steam partial pressure: 0.1×10^5 Pa, (saturation temperature ~319 K, humidity ~60%)

The graphs are generally similar to the corresponding graphs for pool-atmosphere heat and mass transfer (Figure 7-35, Figure 7-36, and Figure 7-37), but in the present case there is no separate regime of natural convection. When the gas-droplet relative velocity approaches zero, the heat and mass transfer are governed by the limiting criterial numbers: Nu=2.0·L M N; Sh=2.0·M.

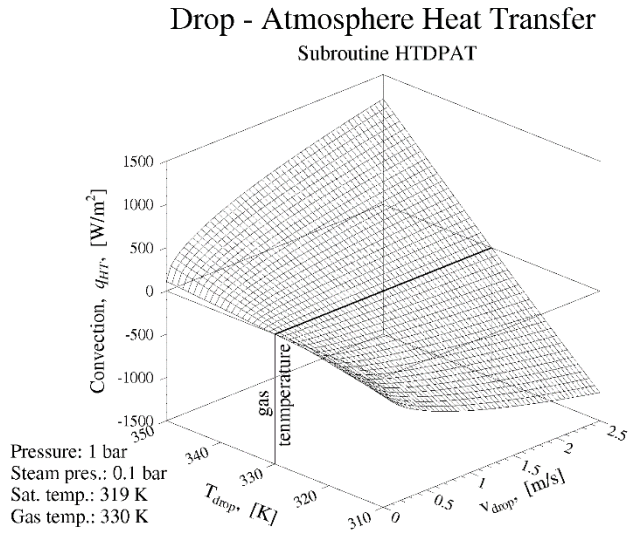


Figure 7-38 Droplet-atmosphere convective heat flux, q_{HT}

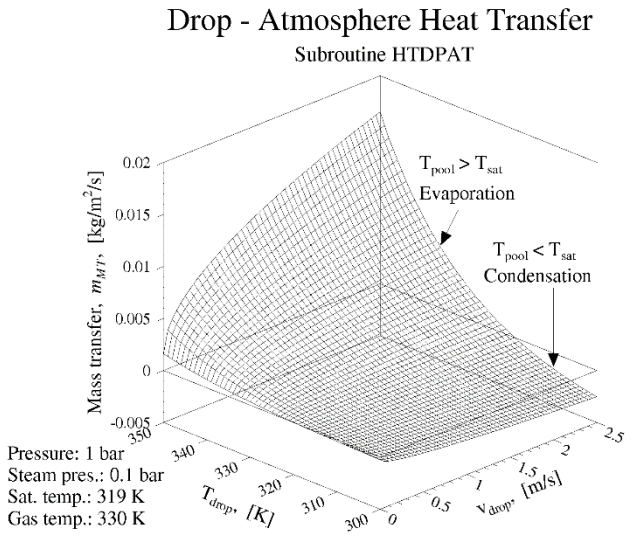


Figure 7-39 Droplet-atmosphere mass transfer flux, m_{MT}

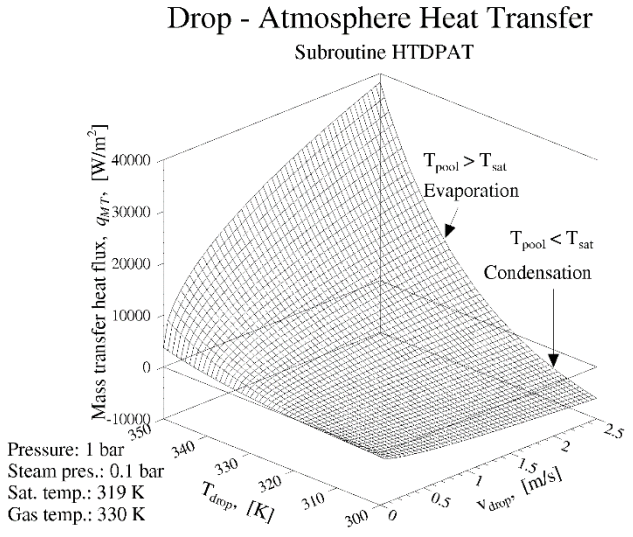


Figure 7-40 Droplet-atmosphere mass transfer energy flux, q_{MT}

7.2.5 Bubble-Pool Heat and Mass Transfer During Bubble Collapse

In case of bubbles two different periods are distinguished, in which different models of heat and mass transfer are used. The base model, used during the long term bubble rise phase, is described in the next section. This section provides a description of the so-called "bubble collapse" model. Bubble collapse occurs typically very quickly (time of order of tens of milliseconds) when a bubble enters the pool. During the collapse period the bubble parameters (temperature, steam concentration) equilibrate with the pool. The short time of bubble collapse is a consequence of the relatively small heat capacity of typical bubbles, and the relatively large fluxes involved in the processes of heat and mass transfer.

The bubble collapse model calculates heat and mass transfer fluxes, collapse time, etc., to determine the final bubble parameters (temperature, diameter, gas composition) at the end of the collapse period. The bubbles entering the pool are subjected to the collapse process, before they are actually added to the bubble component in a given Control Volume. The purpose of the bubble collapse model is to avoid creating very stiff numerical equations, by inserting bubbles which are far from equilibrium with a given pool. The bubble collapse model may be switched off via input data, resulting in a mechanistic calculation of the bubble collapse period using the equations described in the next section. As shown below, the numerical solver is capable of doing that, without excessive reduction of time step. However, in some cases switching off the bubble collapse model may result in significantly slower calculations, or even in convergence failure. Therefore the bubble collapse model should not be turned off, except for a separate effect calculations, like that presented below.

The bubble collapse model is applied in every situation which leads to creation of bubbles in the pool of a Control Volume. This includes the four following cases:

- *JN flow of atmosphere*
  Bubbles, which are created whenever atmosphere flowing through a junction enters the pool of a receiving Control Volume, are equilibrated with the pool by the bubble collapse model.

- *JN flow of bubbles*
  Bubbles flowing through a junction, from the pool of one Control Volume to the pool of another Control Volume, are equilibrated with the pool of the receiving Control Volume by the collapse model. This is necessary, since the pool temperatures may be very different in different Control Volumes.

- *SC boiling*
  Bubbles, which are created during boiling on the surface of a Solid Heat Conductor, are equilibrated with the pool by the collapse model. For any other conditions than a nearly saturated pool, this means a total collapse (full condensation), since a pure steam bubble cannot exist in a subcooled water environment. To provide a more realistic modelling, and also easier numerical solution, the bubble collapse model at the SC boiling surface is automatically switched off whenever the pool temperature is close to saturation.

- *Tabular mass sources*
  Bubbles, which are created when a tabular mass source of gas is present below the pool level, are equilibrated with the pool by the bubble collapse model.

Typically the term bubble collapse is used to signify condensation, which occurs when a steam-rich bubble enters the pool. The applicability of the model has been extended to dry bubbles, as described below, in the discussion of the collapse model.

The bubble collapse process is illustrated in Figure 7-41. It is assumed that a vertical bubble source is placed at a certain elevation in the liquid pool. The source injects a known quantity of gas bubbles (mass flow of $W_{gas,0}$), with the initial bubble diameter, $d_{bubb,0}$. The bubble collapse model calculates the bubble size, composition, temperature, at the requested distance from the source, Z_{det} (referred also as the "detector" elevation). It also calculates the change of the gas mass flow due to condensation (mass flows of the final bubbles, $W_{gas,1}$, and the condensate, $W_{liq,1}$), as well as the energy effects (enthalpy fluxes associated with the final gas and liquid flows, and the heat flux deposited in the pool, Q_{pool}).

The heat transfer during bubble collapse is calculated using a model based on the analysis of Moalem and Sideman [155]. The heat transfer coefficient is obtained using the following correlation ([155], formula 3):

$$Nu = \frac{2}{\sqrt{\pi}}(k_v Pe)^{1/2} \approx 1.13\sqrt{Pe}$$

Nu	Nusselt number, $Nu = D_{bubb,0} \cdot h_{BC} / k_{liq}$, (-)
$D_{bubb,0}$	initial bubble diameter, (m)
h_{BC}	heat transfer coefficient during bubble collapse, (W/m^2/K)
k_{liq}	thermal conductivity of liquid, (W/m/K)
k_v	equal to 1.0 for a single liquid component (water), (-)
Pe	Peclet number, $Pe = Re \cdot Pr$, (-)
Re	Reynolds number, $Re = D_{bubb,0} \cdot v \cdot \rho_{liq} / \eta_{liq}$, (-)
Pr	Prandtl number, $Pr = \eta_{liq} \cdot c_{p,liq} / k_{liq}$, (-)
v	bubble velocity, (m/s)
η_{liq}	viscosity of liquid, (kg/m/s)
ρ_{liq}	density of liquid, (kg/m^3)
$c_{p,liq}$	specific heat of liquid, (J/kg/K)

The heat flux from the collapsing bubble to the pool is obtained as:

$$q = h_{BC}(T_{sat}(p_{vap,0}) - T_{pool})$$

$T_{sat}(p_{vap,0})$ is the saturation temperature for the initial steam partial pressure in the bubble, $p_{vap,0}$.

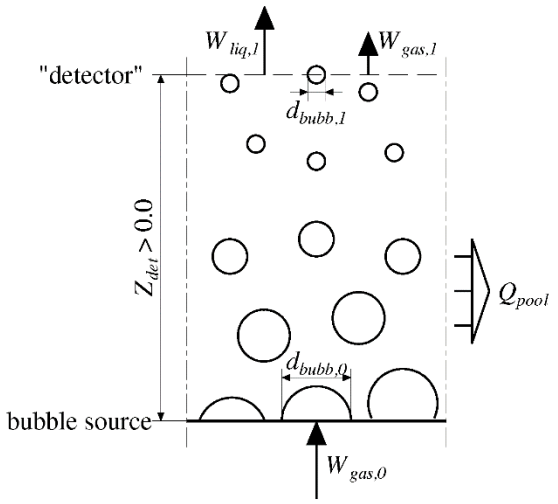


Figure 7-41 Bubble collapse model

The calculation of bubble collapse is described separately for pure steam, and steam-noncondensable bubbles.

- *Pure Steam Bubbles*

  For pure steam bubbles the bubble collapse time is calculated based on the equation derived by Moalem and Sideman for a variable bubble velocity ([155], equation 11):

$$\beta = \left[1 - \frac{5}{4}\left(\frac{k_v}{\pi}\right)^{1/2} \tau\right]^{4/5}$$

  β dimensionless bubble diameter, (-), equal to the current diameter divided by the initial diameter
  τ dimensionless time, equal to: $Ja \cdot Pe^{1/2} \cdot Fo$
  Ja Jacob number, (-), equal to: $\rho_{liq} \cdot c_{p,liq} \cdot \Delta T / (h_{l\text{-}v} \cdot \rho_{vap})$
  ΔT difference between the saturation temperature at bubble steam pressure and the pool temperature, (K)
  $h_{l\text{-}v}$ latent heat of vaporization, (J/kg)
  Pe Peclet number, (-), (see above)
  Fo Fourier number, (-), equal to: $k_{liq}/(\rho_{liq} \cdot c_{p,liq}) \cdot t/(d_{bubb,0}/2)^2$
  ρ_{vap} density of vapor, (kg/m^3)
  t time elapsed from bubble detachment, (s)

  The time for full bubble collapse ($t = t_{BC}$), is obtained by setting β=0 in the above formula. This leads to the following equation:

$$C\left(\frac{k_v}{\pi}\right)^{1/2} Ja \cdot Pe^{1/2} \cdot Fo = 1$$

where C is a constant, equal to 5/4. After rearrangements the following formula is obtained for the collapse time:

$$t_{BC} = \frac{Q_{cond}}{A_{eff} q} = \frac{V_{bubb,0} \rho_{vap} h_{l-v}}{A_{eff} h_{BC} (T_{sat}(p_{vap,0}) - T_{pool})}$$

where Q_{cond} is the enthalpy change associated with condensation of all steam, and A_{eff} is an effective area for heat transfer. The effective heat transfer area is equal to (5/12) of the initial bubble area: $A_{eff} = (5/12) \cdot \pi \cdot (d_{bubb,0})^2$. Note that in case of constant bubble velocity, the value of C is equal to 3/2 (see [155]), and A_{eff} is equal to one half of the initial bubble area, $A_{eff} = (1/2) \cdot \pi \cdot (d_{bubb,0})^2$, the same as recommended in [156] (equation 4).

The Moalem and Sideman equation for β is written in the following form:

$$\beta = (1 - t / t_{BC})^{4/5}$$

For times greater than the bubble collapse time, t_{BC}, the dimensionless bubble diameter is zero. Thus the formula used to calculate the relative bubble diameter at any given time is:

$$\beta = \begin{cases} (1 - t / t_{BC})^{4/5} & if \quad t < t_{BC} \\ 0 & if \quad t > t_{BC} \end{cases}$$

- *Steam-Noncondensable Bubbles*

For steam-noncondensable bubbles the relation between the dimensionless time, τ, and the dimensionless bubble diameter, β, recommended by Moalem and Sideman, is ([155], equation 14):

$$\tau = \tau_0(\beta) + \tau_1(\beta, \beta_f)$$

where τ_0 is given by the pure steam formula and τ_1 is the correction term for noncondensables, given by a relatively complex logarithmic formula ([155], equation 15). The value of β_f is the persistent relative bubble diameter (diameter at the end of bubble collapse), which depends on the initial fraction of noncondensables in the bubble.

Since the theoretical formula, recommended by Moalem and Sideman, requires an iterative calculation to obtain β, a simple fit curve, described below, was developed that approximates the theoretical formula. The discrepancies between the theoretical formula and the fit curve are very small (see Figure 7-43, Figure 7-44) and are not important from a practical point of view because of the limited accuracy of the theoretical model itself and the scatter of the experimental data.

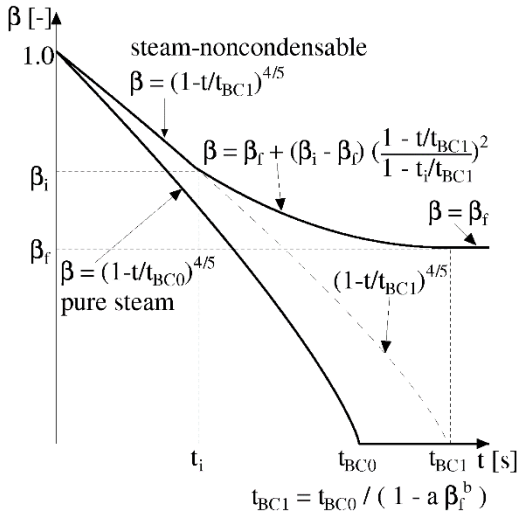


Figure 7-42 Calculation of relative bubble diameter, β

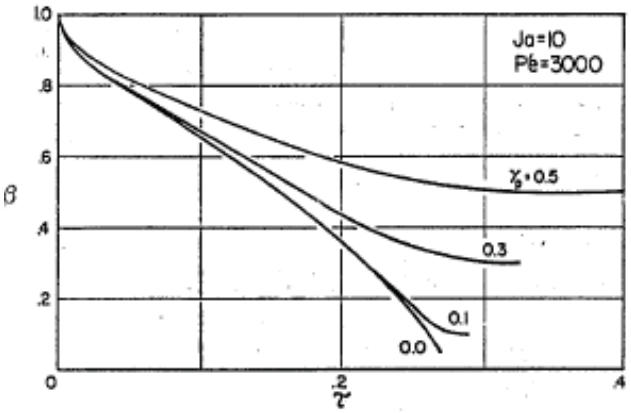


Figure 7-43 Relative bubble diameter versus dimenionless time, Wittke , [127] ($\gamma_p = \beta_f$)

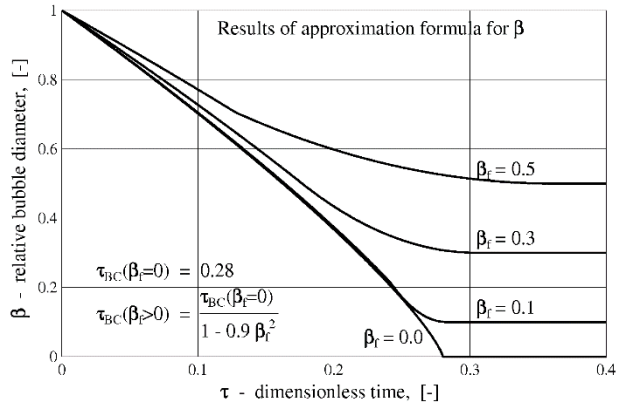


Figure 7-44 Relative bubble diameter versus dimenionless time, approximation function

The theoretical equation gives an infinite bubble collapse time because $\tau_1(\beta, \beta_f) \rightarrow \infty$ when $\beta \rightarrow \beta_f$. In practice however β becomes approximately equal to β_f at a finite time, somewhat larger than the time required for the pure steam bubble collapse (see Figure 7-43). The basic formula used in the present model is the same as that used for pure steam bubbles. The increased time of bubble collapse is taken into account by introducing a dimensionless factor, ψ, into the equation. The bubble collapse time is calculated from the formula:

$$t_{BC} = \frac{Q_{cond}}{A_{eff} q \psi}$$

where ψ is a dimensionless factor, introduced to take into account the increased collapse time in case of steam-noncondensable bubbles. The value of the factor ψ is calculated from the formula:

$$\psi = 1 - a\beta_f^b$$

The above formula was developed to provide a good fit to the results of the theoretical model shown in [127]. The values of a and b are: a=0.9, b=2.0. The formula used to calculate the relative bubble diameter at any given time is as follows:

$$\beta = \begin{cases} (1 - t/t_{BC})^{4/5} & if \quad t < t_i \\ \beta_f + (\beta_i - \beta_f)\left(\dfrac{1 - t/t_{BC}}{1 - t_i/t_{BC}}\right) & if \quad t_i < t < t_{BC} \\ \beta_f & if \quad t < t_{BC} \end{cases} \qquad \begin{cases} (1 - t/t_{BC})^{4/5} & if \quad t < t_{BC} \\ 0 & if \quad t > t_{BC} \end{cases}$$

t_i value of time at intersection of the first and the second equations (Figure 7-42), defined by: $t_i = t_{BC} \cdot (1 - \beta_f)^{3/2}$

β_i value of the dimensionless bubble diameter at time equal to t_i, equal to: $\beta_i = (1 - t_i/t_{BC})^{4/5}$

The calculation procedure to compute β for a pure steam bubble, and a steam-noncondensable bubble, is illustrated in Figure 7-42. The full collapse time for the pure steam and steam-noncondensable bubbles are denoted in the figure by t_{BC0} and t_{BC1} respectively. The value of t_{BC1} is related to the value of t_{BC0} by the formula: $t_{BC1} = t_{BC0}/\psi$.

The approximation formula for β has been verified by comparisons with a number of available experimental data. The comparison of the calculated results with Wittke and Chao experimental data [127] is presented in Figure 7-43 and Figure 7-44. A good agreement between the calculated results and the experimental data is observed.

The calculation procedure within the bubble collapse model is somewhat different in case of full bubble collapse ($t > t_{BC}$, $\beta = \beta_f$), and partial collapse ($t < t_{BC}$, $\beta > \beta_f$). The procedure applied for full bubble collapse is described below. Partial bubble collapse is less important, since bubble collapse is typically very fast and is completed within a very short distance from the source.

- Calculational procedure, full bubble collapse

In case of full bubble collapse, the final bubble is in equilibrium with the surrounding pool (see [155]). This means:

$$T_{bubb,1} = T_{pool}$$
$$p_{steam,1} = p_{sat}(T_{pool})$$

where $T_{bubb,\,1}$ is the final bubble temperature, and $p_{steam,\,1}$ is the final steam partial pressure. The final partial pressures of all gases in the bubble are therefore obtained from:

$$p_{k,1} = \begin{cases} p_{sat}(T_{pool}) & if \quad k = steam \\ p_{k,0}\,\dfrac{p - p_{sat}(T_{pool})}{p - p_{steam,0}} & if \quad k \neq steam \end{cases}$$

$p_{vap,\,1}$ partial pressure of vapor in the final bubble, (Pa)
$p_{vap,\,0}$ partial pressure vapor in the initial bubble, (Pa)

With all partial pressures and the bubble temperature known, the bubble gas composition (mass fractions) is calculated using the subroutine GASEQP (section 3.5.2). All physical properties of the final bubble, like density, enthalpy, etc., are also calculated by the Fluid Property package.

The fraction of condensed steam, F_{cond}, is calculated from the mass balance written using the initial and final steam mass fractions. Note that for pure steam bubbles the equilibrium condition requires that the whole bubble is collapsed (the saturation pressure at the pool temperature is smaller than the total pressure of the submerged bubble, thus collapse will proceed until the bubble disappears).

$$F_{cond} = \begin{cases} 1.0 & - \quad pure\ steam\ bubbles \\ \dfrac{c_{vap,0} - c_{vap,1}}{1.0 - c_{vap,1}} & - \quad steam - noncondensable\ bubbles \end{cases}$$

$c_{vap,\,1}$ mass fraction of vapor in the final bubble, (-)
$c_{vap,\,0}$ mass fraction of vapor in the initial bubble, (-)

The final mass fluxes of liquid (condensate), $W_{liq,\,1}$, and gas (bubbles), $W_{gas,\,1}$, are calculated as:

$$W_{liq,1} = W_{gas,0} \cdot F_{cond}$$
$$W_{gas,1} = W_{gas,0} \cdot (1 - F_{cond})$$

where $W_{gas,\,0}$ is the mass flow of bubble gas, (kg/s), at the start of the collapse process.

The difference between the enthalpy of the initial gas and the final enthalpies of the collapsed bubbles and liquid, gives the net heat effect of the bubble collapse process:

$$Q_{pool} = H_{gas,0} - H_{gas,1} - H_{liq,1} = W_{gas,0}h_{gas,0} - W_{gas,1}h_{gas,1} - W_{liq,0}h_{liq,0}$$

The heat Q_{pool} is released during the process. This heat is deposited in the pool (if stratification models are active, then the heat is either deposited at the bubble source elevation, or, if the plume model is active, it is deposited in the upper layer of the pool).

The procedure described above is appropriate for a typical collapse process, when the steam fraction is relatively large, and steam condensation occurs in the bubble upon entering the pool. Initially that was the only application of the collapse model. If dry bubbles were entering the pool, then the bubble collapse model was not used, that means the bubble parameters at the end of collapse were identical to the parameters of the initial bubble. It has been observed that in some cases the injection of very hot dry bubbles may cause numerical problems, and result in slowing down the calculation. To avoid those problems the bubble collapse model has been extended to dry bubbles, as described below.

If the bubbles are dry there is no "collapse" that means steam condensation. Instead, evaporation of steam into the bubbles occurs. The bubble temperature equilibrates with the pool temperature almost instantaneously (see the hot dry bubble test below). The evaporation process is somewhat slower.

The only modification needed to extend the bubble collapse model to the dry bubble case is the calculation of final steam partial pressure in the bubble. In the extended model, the final bubble parameters are defined as follows:

$$T_{bubb,1} = T_{pool}$$

$$p_{vap,1} = \begin{cases} p_{sat}(T_{pool}) & if \quad p_{vap,0} > p_{sat}(T_{pool}) \\ Max[\, p_{vap,0}, C \cdot p_{sat}(T_{pool})] & if \quad p_{vap,0} < p_{sat}(T_{pool}) \end{cases}$$

With the above formula, the relative humidity of the final bubble will never be smaller than C·100%. The value of C has been set to 0.3, based on separate calculations, described below.

A simple model has been set up to investigate the behavior of a hot dry bubble in a pool. The model consists of a nearly saturated pool at atmospheric pressure. At t=0.0 s hot dry bubbles are injected into the pool. The bubbles are composed of pure hydrogen, the initial bubble temperature is 1000 K. The bubble collapse model has been switched off, thus the bubble behavior was calculated mechanistically, using the equations shown in section 7.2.6, below. Results are shown in Figure 7-45 through Figure 7-50.

Figure 7-45 through Figure 7-47 show the bubble temperature, the energy fluxes due to convection and evaporation, and the bubble relative humidity, results obtained with a time step of 10^{-4} s. It is seen that the bubble temperature decreases very quickly (in about 0.001 s) to the pool temperature. In this period the bubble is intensively cooled by convection. The convective flux is of order of megawatts per square meter (Figure 7-46). The convective heat transfer coefficient, calculated from the LeClair-Hamielec correlation (see section 7.2.6) is about 8000 W/m^2/K. The large convective flux and very small heat capacity of bubble result in extremely fast cooling down of the bubble.

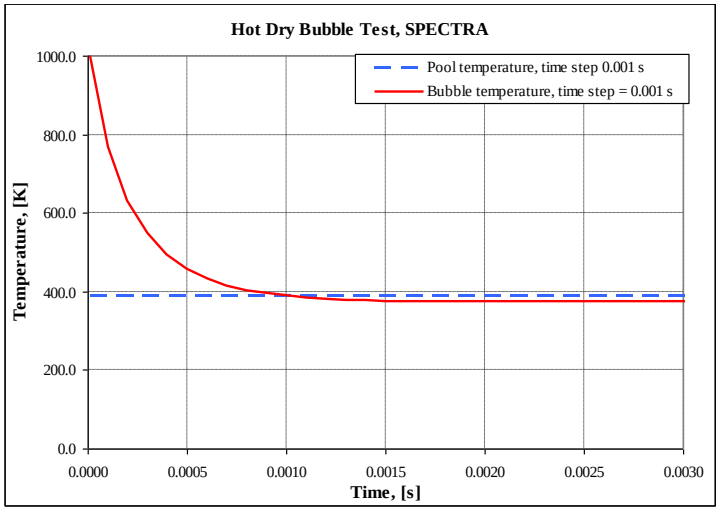


Figure 7-45 Bubble and pool temperature - hot dry bubble test

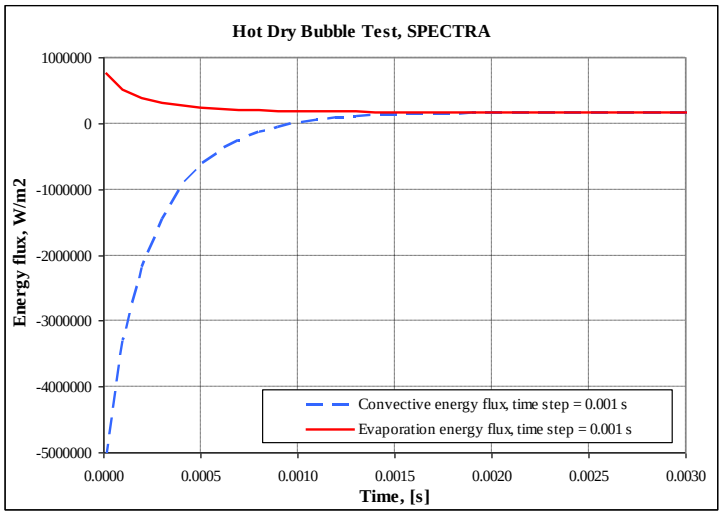


Figure 7-46 Convective heat flux and evaporation energy flux - hot dry bubble test

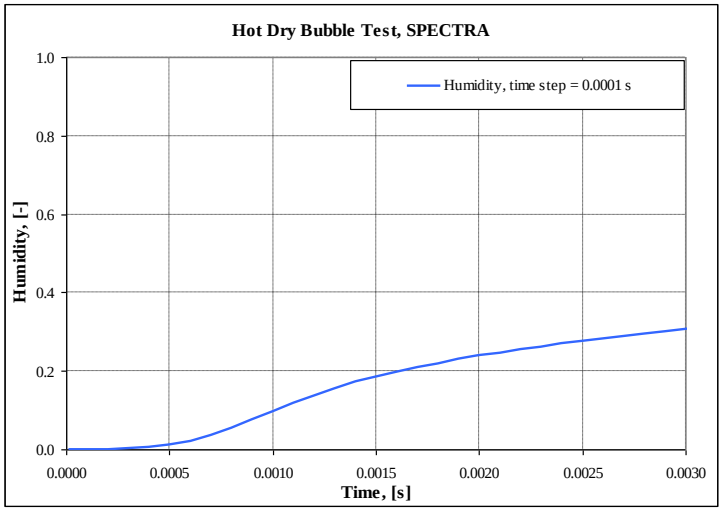


Figure 7-47 Relative humidity of the bubble gas - hot dry bubble test

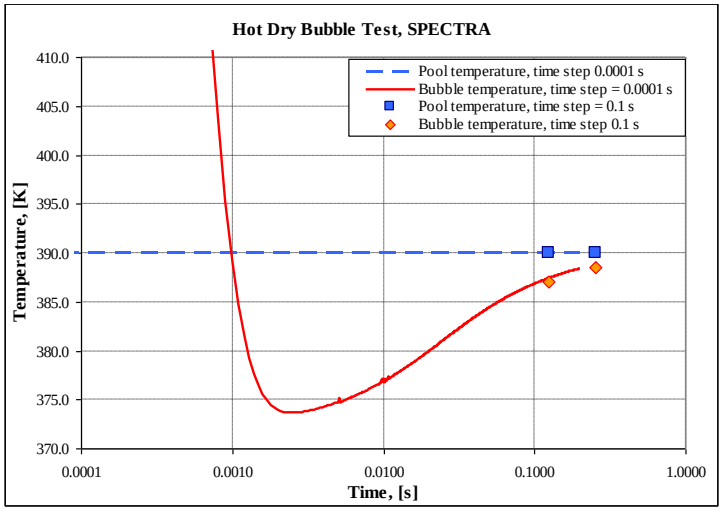


Figure 7-48 Bubble and pool temperature - hot dry bubble test

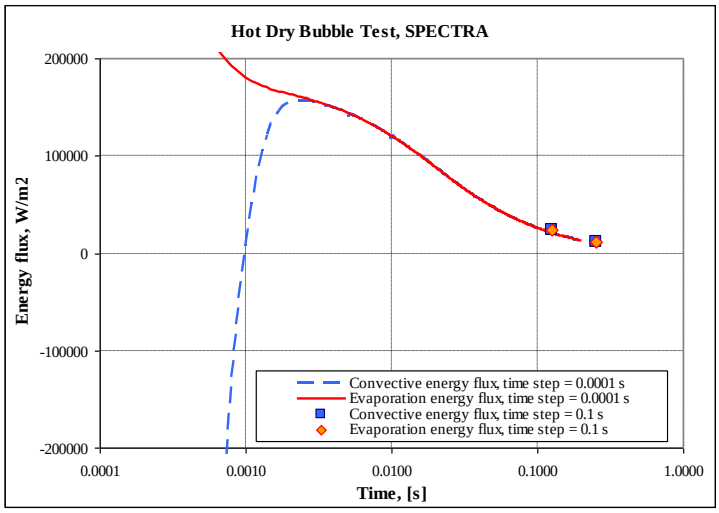


Figure 7-49 Convective heat flux and evaporation energy flux - hot dry bubble test

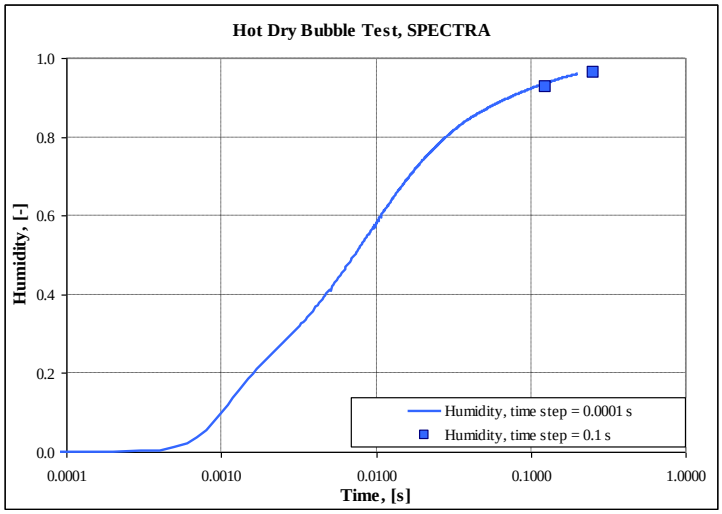


Figure 7-50 Relative humidity of the bubble gas - hot dry bubble test

During the next period (t>0.001 s) quasi stable conditions are established. The bubble temperature stabilizes at the a value little lower than the pool temperature (Figure 7-45). Relatively slow evaporation of steam into the bubble occurs. The evaporation is driven by the fact that the bubble is dry (at 0.001 s the relative humidity is ~10% - Figure 7-47). The evaporation process needs a continuous supply of energy. The energy stored in the bubble is very small, therefore energy has to be taken from the water. This is the reason why the bubble temperature stabilizes at the value a little lower than the pool temperature. With the established temperature difference the convective heat flux is able to supply energy needed for evaporation. It is seen in Figure 7-46 (as well as Figure 7-49), that after about 0.002 s both convective and evaporation energy fluxes become practically identical.

Comparison of the results of two runs is shown in Figure 7-48, Figure 7-49, Figure 7-50. The base run was performed using the time step of 10^{-4} s. Such a small time step was needed to show how the key parameters evolve in time. The comparison run was performed with the time step of 0.1 s. No internal reduction of time step was observed. It is seen that practically the same results are obtained. Thus, the numerical solver was capable of solving the present problem, involving rather "stiff" equations, using the time step of 0.1 s (further discussion of solving stiff equations is provided in section 16.6.3 and 19.1). The following conclusions are drawn from the present example problem:

- The bubble temperature equilibrates almost instantaneously with the pool temperature. Therefore, assuming the end of collapse temperature to be equal to equal to the pool temperature, $T_{bubb,1}$ is a good approximation.
- The evaporation of steam into the bubbles proceeds slower. Therefore the minimum relative humidity at the end of bubble collapse period was set at 30%, in the present example problem the value achieved after 0.003 s. The actual value is not very important, since the evaporation will proceed, calculated in a mechanistic way, once the bubble collapse is finished. The limit of 30% on the relative humidity allows to reduce somewhat the initial stiffness of the governing equations, and speed up calculations, as has been ascertained in several test runs.

One final remark has to be made about the bubble collapse model. The initial intention was to apply the model for the "detector elevation" (see Figure 7-41) being close to the bubble source elevation. With such approach an incomplete collapse may occur when the bubble flow time from the source to the detector is shorter than the bubble collapse time. This incomplete bubble collapse was observed to cause occasionally numerical problems. Incomplete collapse occurs very seldom. It may occur in some "pathological" cases, when $T_{sat}(p_{vap,\,0})$ is extremely close to the pool temperature. In such case the denominator of the equation for the bubble collapse time, t_{BC}, becomes very small, and consequently t_{BC} becomes very large. To avoid such situations, the present implementation of the model makes sure that the bubble collapse proceeds to the end in all circumstances, even when the model predicts partial collapse. This approach allows to prevent occasional numerical problems and it is considered to have no major impact on the results in practical cases.

7.2.6 Bubble-Pool Heat and Mass Transfer

When the bubble collapse period is finished (section 7.2.5), then the bubbles are placed in the pool of a Control Volume at the elevation of the bubble source, and begin their rise towards the pool surface (they can also be dragged downwards, if there is sufficiently strong downflow of liquid - see the description of the bubble velocity calculation, section 4.2.3). The equations used to calculate heat and mass exchange between bubble and pool in this period are discussed below.

- Heat Transfer

  The convective heat transfer coefficient is calculated using the correlations developed by LeClair and Hamielec for bubbles in swarms, with the creeping flow correlation, valid for low Reynolds numbers. The LeClair and Hamielec correlation [157] is:

$$Sh_{L-H} = \begin{cases} \dfrac{(0.65 + 0.06\,Re^{0.5})\,Pe^{0.5}}{\{(5 + 6\alpha^{1/3} + \alpha^2)/[5(1-\alpha)]\}^{0.5}} & for \quad Re < Re_1 \\ \dfrac{2.213\,Pe^{0.5}}{Re^{0.108}(1-\alpha)^{0.5}} & for \quad Re_2 < Re < Re_3 \\ \dfrac{1.13\,Pe^{0.5}}{(1-\alpha)^{0.5}} & for \quad Re > Re_3 \end{cases}$$

  where α is the void fraction (volumetric fraction of bubbles in the pool). The transition Reynolds numbers, Re_1, Re_2, and Re_3 are as follows:

  - $Re_1 = 10.0$ from [157]
  - $Re_2 = 20.0$ an interpolation range is defined to provide a smooth transition from the first to the second formula. For $Re_1<Re<Re_2$ the Sherwood number is obtained by linear interpolation between the values obtained from the first and the second formula.
  - $Re_3 = 504.435$ for this value of the Reynolds number the second and the third formula give the same Sherwood number, thus providing a smooth transition. The exact value is equal to: $(2.213/1.13)^{(1/0.108)}$.

  For large Reynolds numbers the correlation is:

$$Sh_{L-H} = \frac{1.13\,Pe^{0.5}}{(1-\alpha)^{0.5}}$$

  which is the same as the Nusselt number correlation for the bubble collapse model (section 7.2.5).

  The creeping flow correlation is ([158], section 3.III.2):

$$Sh_C = 1 + (1 + Pe)^{1/3}$$

  Selection of the correlation is made by the program by choosing the correlation that gives the larger Sherwood number:

$$Sh = Max(Sh_{L-H}, Sh_C)$$

In practical cases this means that the creeping flow correlation is used for Reynolds numbers up to about 1 ÷ 10 - Figure 7-51 (Re < ~1 for α = 0.7, and Re < ~10 for α = 1.0). This is in agreement with its range of validity (see [158], section 3.III). For higher Reynolds numbers the LeClair and Hamielec correlation is used.

For the heat transfer calculation a heat and mass transfer analogy is used, thus:

$$Sh(Pe) = Nu(Sc)$$

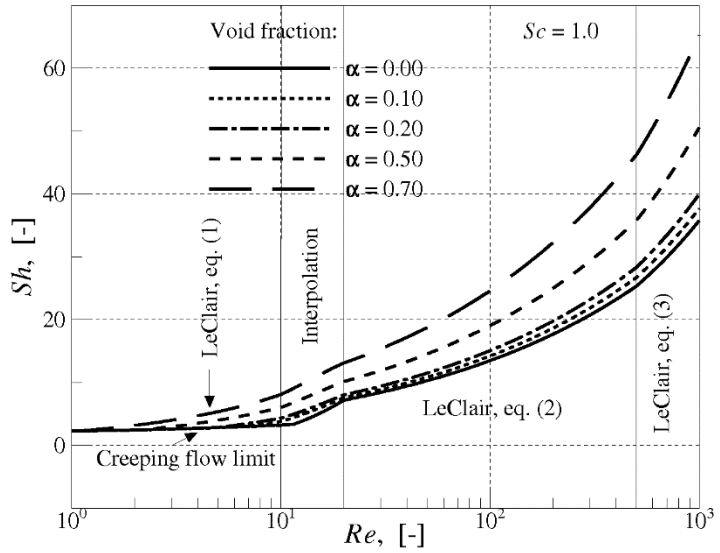


Figure 7-51 Pool-bubble heat transfer correlation

The Nusselt number, calculated for the heat transfer Peclet number ($Pe = Re \cdot Pr$), is equal to the Sherwood number, calculated for the mass transfer Peclet number ($Pe = Re \cdot Sc$) (see also [159], section IV.D). Note that the fluid properties for the criterial number are those of the liquid at pool temperature. When the Nusselt number is known the heat transfer coefficient is obtained as:

$$h = Nu \frac{k_{liq}}{D_{bubb}}$$

where k_{liq} is the thermal conductivity of liquid. The interphase velocity used to compute the Reynolds number is taken as:

$$v_{\text{int}} = Max(\sqrt{(v_{hor,bubb} - v_{hor,pool})^2 + (v_{ver,bubb} - v_{ver,pool})^2}, v_{\min,bubb-pool})$$

Here $v_{hor,\,bubb}$, $v_{hor,pools}$, $v_{ver,\,bubb}$, $v_{ver,\,pool}$, are the horizontal and vertical velocities of bubbles and pool in a Control Volume. The $v_{min,\,bubb\text{-}pool}$ is a minimum velocity for interphase heat transfer

calculation (user-defined parameter VINTCV(IBBPL), with a default value of 0.0 - see Volume 2). The heat flux is equal to:

$$q = h(T_{pool} - T_{bubb})$$

The convention used here is that the heat flux is positive when the heat is transported into the bubble, and negative when the heat is transported from the bubble to the pool.

- Mass transfer

The mass transfer is limited by an internal diffusion limit. The diffusion limit is calculated using the Newman correlation [160]:

$$Sh = 6.58$$

The mass transfer coefficient, K_M, (m/s), is obtained from Sherwood number:

$$K_M = \frac{Sh \cdot D_C}{D_{bubb}}$$

where D_C is the diffusion coefficient for steam in the bubble gas. The mass transfer flux is obtained similarly as in case of pool-atmosphere mass transfer (section 7.2.3), and droplet-atmosphere mass transfer (section 7.2.4):

$$m_{MT} = \frac{K_M \rho_{vap}}{p_{am}} (p_{sat}(T_{pool}) - p_{vap})$$

Substitution of the p_{am} definition into the equation for m_{MT} gives the formula used by the program:

$$m_{MT} = K_M \rho_{vap} \ln\left(\frac{p - p_{vap}}{p - p_{sat}(T_{pool})} \right)$$

The energy fluxes, associated with the mass flux, are calculated differently for the case evaporation and condensation, as described below.

- Evaporation, $p_{vap} < p_{sat}(T_{pool})$; $m_{MT} > 0.0$
It is assumed that the vapor appears at the temperature equal to the pool temperature. The enthalpy of the vapor is therefore $h_{sat.vap}(T_{pool})$. The enthalpy of the disappearing water is $h_{liq}(T_{pool})$. The difference between the enthalpy flux gained by the bubble, and lost by the pool, gives the net heat effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{sat.vap}(T_{pool}) - h_{liq}(T_{pool})]$$

The value of q_{MT} represents the energy flux is "consumed" during the process. This energy is assumed to be taken from the bubble.

- Condensation, $p_{vap} > p_{sat}(T_{drop})$; $m_{MT} < 0.0$

It is assumed that the condensate appears with the temperature equal to the bubble gas temperature. The enthalpy of the condensate is therefore equal to the liquid enthalpy at the atmosphere temperature, $h_{liq}(T_{bubb})$. The enthalpy of the disappearing steam is equal to $h_{gas}(T_{bubb}, p_{vap})$, where p_{vap} is steam partial pressure in the bubble. The difference between the enthalpy gained by the gas, and lost by the liquid, gives the net effect of the mass transfer process, q_{MT}:

$$q_{MT} = m_{MT}[h_{gas}(T_{bubb}, p_{vap}) - h_{liq}(T_{bubb})]$$

The value of q_{MT} represents the energy flux is "released" during the process. This energy is assumed to be deposited in the bubble.

Examples of the pool-bubble heat and mass transfer is shown in Figure 7-52, Figure 7-53, Figure 7-54, showing the convective heat flux, q_{HT}, the mass transfer flux, m_{MT}, and the energy flux accompanying the mass transfer q_{MT}. The values are plotted versus pool temperature, and pool-bubble relative velocity.

The following bubble parameters were assumed for calculations:

- temperature: 330 K
- total pressure: 1.0×10^5 Pa
- steam partial pressure: 0.1565×10^5 Pa, (saturation temperature: 328 K, humidity 90.5%)

The heat flux is governed by forced convection, with the heat transfer coefficient proportional to $Re^{1/2}$. The mass flux is driven by the steam pressures. The relative humidity of the bubbles (90.5%) was selected to obtain such steam partial pressure in the bubble, that the saturation temperature at this pressure is almost exactly equal to 328 K. It is seen in Figure 7-53, that if the pool temperature exceeds 328 K, then evaporation of steam into the bubble occurs. If, on the other hand, the pool temperature is lower than 328 K, then steam is condensed in the bubble. There is no influence of the bubble velocity on the mass transfer rates. This is a consequence of the fact that the Sherwood number is constant (Sh=6.58), in contrast to the droplet mass transfer, where the Sherwood number is a function of the Reynolds number (section 7.2.3), and the mass transfer rate depends on the droplet velocity (Figure 7-39).

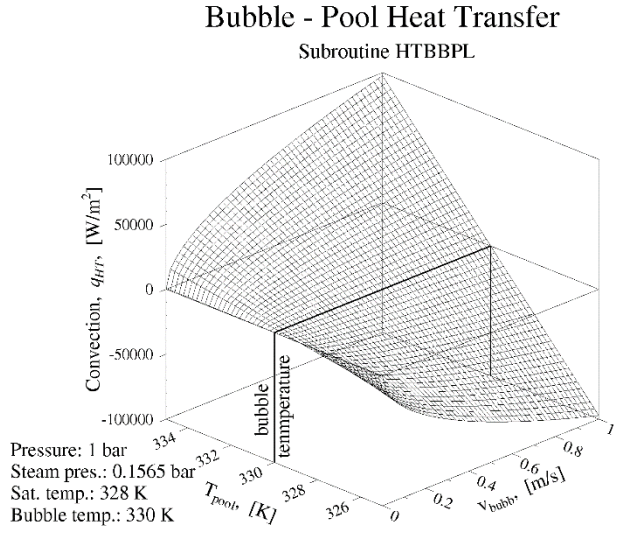


Figure 7-52 Pool-bubble convective heat flux, q_{HT}

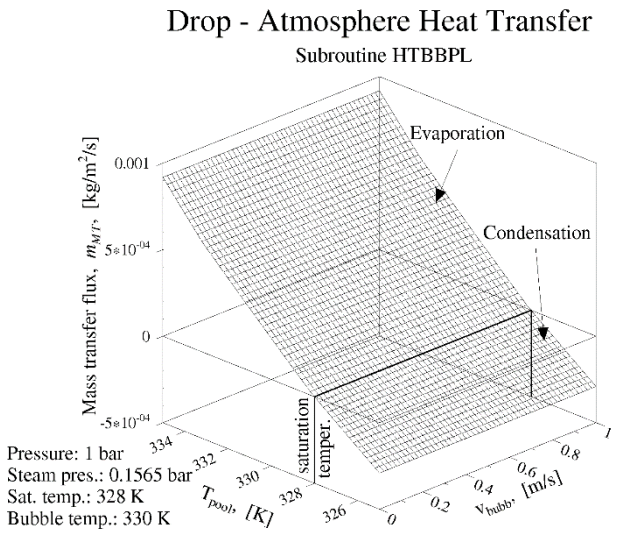


Figure 7-53 Pool-bubble mass transfer flux, m_{HT}

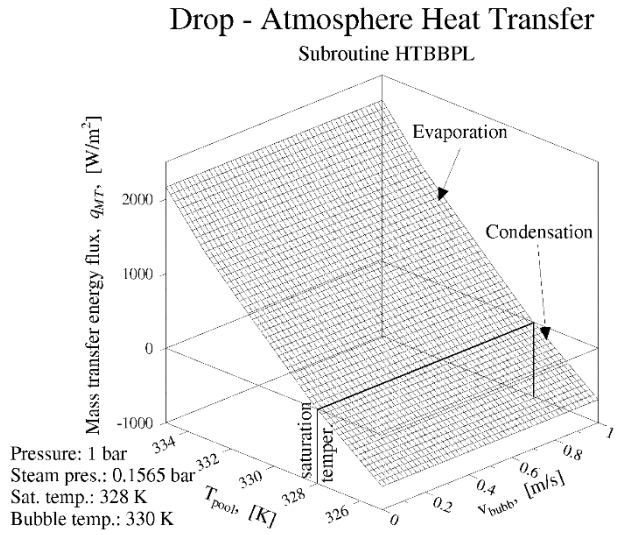


Figure 7-54 Pool-bubble mass transfer energy flux, q_{HT}

8 Thermal Radiation Package

8.1 Introduction

When thermal radiation is less important, then a simple radiation model, available inside the Solid Heat Conductor Package, may be used. In the simple radiation model only the radiative heat exchange between wall surface and gas is modeled. The gas is assumed to be opaque (gas emissivity and absorptivity are equal to one). For such cases the radiative heat flux is given by (see [113]):

$$q_r = \sigma \cdot \varepsilon_w(T_w) \cdot (T_w^4 - T_g^4)$$

where σ is the Stefan-Boltzmann constant (σ=5.67×10^{-8} W/m^2/K^4), $\varepsilon_w(T_w)$ is the wall emissivity (a function of temperature), T_w, T_g, are the wall and the gas temperatures respectively.

In case when radiation is more important, a detailed radiation model, based on the net enclosure with grey surfaces and non-grey gas (Hottel gas) approximations, is available. The model is available in the SPECTRA Thermal Radiation Package, and is shortly described below.

The Thermal Radiation Package allows to model up to nine independent radiating systems. Each system may consist of up to 50 radiating surfaces, and may use one of the two available models:

- Radiation in an enclosure with non-absorbing/non-emitting medium (section 8.2).
- Radiation in an enclosure with participating gas (section 8.3).

Each radiating system is characterized by a consistent set of view factors and, if the second model is used, by a set of mean beam lengths between surfaces. The characteristic features of a radiating system are described below.

A radiating system is defined by selecting surfaces of 1-D or 2-D Solid Heat Conductors, which will participate in the radiative heat transfer. Both left and right surfaces of a SC and any boundary cell of a TC may be selected. An option for radiation to a water pool instead of the structure may be selected for a surface representing a horizontal floor. The pool option is available only for 1-D Solid Heat Conductors. With the pool option the radiative flux is deposited at the SC surface in absence of a pool, or at the pool surface if a pool is present. Further discussion of the radiation calculations in presence of a water pool is given in section 8.4.

When all radiating surfaces of the given system are selected, a matrix defining view factors, F_{ij} (called also shape factors), must be supplied. The view factors must fulfill the reciprocity relation, and the enclosure relation. The reciprocity relation is ([21], section 6.3.1):

$$A_i F_{ij} = A_j F_{ji}$$

Here A_i, A_j are areas of the surfaces *i* and *j*. The enclosure relation is ([21], section 6.3.2):

$$\sum_{j=1}^{N} F_{ij} = 1.0$$

where *N* is the total number of radiating surfaces. The enclosure relation (called also the summation rule) expresses the fact that within a given radiating system all radiant energy leaving surface i must be intercepted by some surface of the enclosure (including *i* itself if it is concave). If the enclosure relation is not fulfilled, then a system will not be conserving energy. If $\Sigma F_{ij}<1.0$, then energy will be lost, that means the system will be radiating part of its energy to a "black hole". In case of $\Sigma F_{ij}>1.0$ the system will be gaining energy.

The SPECTRA program performs a total mass and energy check at the end of each time step. Typically the mass and energy errors are caused only by round-off errors of double precision arithmetics, which give relative errors of order of 10^{-15}. If the thermal radiation model is active then the energy error is governed by the accuracy with which view factors are entered.

The program will not accept view factors unless they fulfill the reciprocity relation and the enclosure relation, with the accuracy of at least 8 decimal places. While this requirement puts a significant burden on the user, it is deemed necessary to assure proper energy conservation during calculations. With the required accuracy of 8 decimal places, the expected relative energy error is of order of 10^{-8}, which is still a relatively large value, compared to the typical energy error of order of 10^{-15}, made by the program when thermal radiation model is inactive.

When the second radiation model is applied (radiation with participating gas), the user must supply the matrix of mean beam lengths, L_{ij}. The mean beam lengths are used to calculate gas emissivities and absorptivities. Since the emissivity correlations themselves have an accuracy of at best two decimal places, there is no need to enter the mean beam lengths with excessive accuracy (see Volume 2). The following relations must however be fulfilled when the beam length matrix is prepared:

$$L_{ij} = 0.0 \quad if \quad F_{ij} = 0.0$$
$$L_{ij} > 0.0 \quad if \quad F_{ij} > 0.0$$

A description of the two thermal radiation models is provided below. Section 8.2 presents the model for radiative heat exchange in a multi-surface enclosure, with a non-absorbing and non-emitting medium. Section 8.3 presents the model for radiative heat exchange in an enclosure with a participating gas.

8.2 Radiation Exchange in an Enclosure with a Non-absorbing Media

The theoretical basis of the model is described in more detail in [20], pages 15-31 through 15-35, and [113] sections 2, 3. The model is based on the following assumptions:

a) The enclosure can be divided into a finite number of isothermal surfaces.
b) Surfaces are gray body emitters, absorbers and reflectors.
c) The direction distribution of radiation leaving the surface obeys Lambert's Cosine Law.
d) Radiant energy leaving any surface is uniform over that surface.

At each surface of the enclosure an equation can be written expressing the fact that the flux of radiant energy leaving the surface is the sum of the emitted radiation plus the reflected radiation (see [113] section 2.1):

$$A_i H_i = A_i \varepsilon_i E_i + \rho_i A_i G_i$$

A_i area of the surface i, (m2)
H_i radiosity, radiant energy leaving surface i per unit area, (W/m^2)
E_i i'th surface black body emission power, equal to: σT_i^4, (W/m^2)
G_i radiation coming from other surfaces to surface i, per unit area of the surface i, (W/m^2)
ε_i emissivity of the surface i, (-)
ρ_i reflectivity of the surface i, (-)

The radiation coming at the surface i is equal to:

$$A_i G_i = \sum_{j=1}^{N} A_j H_j F_{ji}$$

F_{ij} view factor from the surface i to the surface j, (-)
N number of radiating surfaces

Thus the basic equation has the form:

$$A_i H_i = A_i \varepsilon_i E_i + \rho_i \sum_{j=1}^{N} A_j H_j F_{ji}$$

The surface blackbody emission power, E_i, is, according to Stefan-Boltzmann's law, equal to: $E_i=\sigma T_i^4$, where: σ is the Stefan-Boltzmann's constant, (W/m^2/K^4), (equal to 5.67×10^{-8} [32]), and T_i is the temperature of the surface i, (K).

Surface reflectivities of the gray surfaces, ρ_i, are equal to: $\rho_i=1-\varepsilon_i$. Taking into account the relation between ε_i and ρ_i, and the reciprocity law: $A_i\times F_{ij}=A_j\times F_{ji}$, the above equation may be written as:

$$H_i = \varepsilon_i E_i + (1-\varepsilon_i)\sum_{j=1}^{N} H_j F_{ij}$$

The above equation, written for each of the N radiating surfaces, defines a set of N linear equations with respect to H_j. This equation set may be written shortly in a matrix form:

$$CH = E$$

where the coefficients of the matrix $\boldsymbol{C}$, and the vector $\boldsymbol{E}$, are given by:

$$C_{ij} = \frac{\delta_{ij} - (1-\varepsilon_{ij})F_{ij}}{\varepsilon_{ij}}$$

$$E_i = \sigma T_i^4$$

δ_{ij} is the Kronecker delta, δ_{ij}=1 when i=j, and δ_{ij}=0 when $i \neq j$. The coefficients C_{ij} are known, since they depend only on the system geometry and physical properties. If the right hand sides of the equations are known (i.e. the surface temperatures, T_i, are determined) then the equation set can be solved to calculate the surface radiosities, H_i, using a standard procedure to solve linear equations.

When all radiosities, H_i, are determined, the heat flux lost from the surface i due to thermal radiation, can be calculated as a difference between the radiosity and the incoming radiation flux ([113] section 2.1):

$$A_i q_i = A_i H_i - A_i G_i$$

or:

$$q_i = H_i - \sum_{j=1}^{N} H_j F_{ij}$$

q_i is the heat flux lost from the surface i due to thermal radiation, (W/m^2). It is interesting to note that the sum of radiation heat fluxes from all surfaces is equal to:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N} A_i H_i - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i H_j F_{ij}$$

Using the reciprocity relation the above equation can be rewritten as:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N} A_i H_i - \sum_{i=1}^{N}\sum_{j=1}^{N} A_j H_j F_{ji} = \sum_{i=1}^{N} A_i H_i - \sum_{j=1}^{N} A_j H_j \sum_{i=1}^{N} F_{ji}$$

From the closure relation it follows that ΣF_{ji} = 1.0, and finally:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N} A_i H_i - \sum_{j=1}^{N} A_j H_j = 0.0$$

Thus the total radiative energy is conserved provided that the reciprocity and closure relations are fulfilled.

8.3 Radiation Exchange in an Enclosure with a Participating Gas

The theoretical basis of this model is described in more detail in [20], pages 15-54 through 15-64 or [113] section 7. The assumptions for the model are the same as for the first model (assumptions a) through d), section 8.2). Additionally it is assumed that:

e) Gas in the enclosure is assumed to be homogeneous and isothermal.
f) Gas is assumed to be nongray, and does not reflect the radiation.

The terms “gray” and “nongray” are sometimes confusing since they are not consistently applied in the literature. A gray gas (in general a gray body) is defined as such that the monochromatic emissivity ε_λ is independent of wavelength [24]:

$$\varepsilon_\lambda = \varepsilon = const.$$

For a non-gray body on the other hand, radiation properties are wavelength-dependent. Some authors [21], [114] put a further requirement on gray gas, that gas absorptivity is equal to its emissivity, $a=\varepsilon$. If absorptivity is different than emissivity, then such gas is called “nongray gas”, or “Hottel gas”. Thus:

$$gray\ gas: \quad \begin{cases} \varepsilon_\lambda = \varepsilon \\ a = \varepsilon \end{cases}$$

$$nongray\ gas: \quad \begin{cases} \varepsilon_\lambda = \varepsilon \\ a \neq \varepsilon \end{cases}$$

According to the above nomenclature the model present in the SPECTRA code is based on the nongray gas approximation. Since the term “nongray” may be confusing, the term “Hottel gas” is used below to signify the gas for which absorptivity and emissivity are wavelength independent, and $a \neq \varepsilon$.

The basic equation is similar to the case without participating gas. Again, for each surface of the enclosure the flux of radiant energy leaving the surface is the sum of the emitted radiation plus the reflected radiation:

$$A_i H_i = A_i \varepsilon_i E_i + \rho_i A_i G_i$$

The meaning of all symbols has been explained in section 8.2. In the present case the reflected radiation consists of two parts. The flux coming directly from other surfaces (due to some transmittance of gas), and the flux coming from the radiating gas itself. The equation is ([113], section 7.1):

$$A_i H_i = A_i \varepsilon_i E_i + \rho_i \left(\sum_{j=1}^{N} A_j H_j F_{ji} \tau_{g,ij} + \sum_{j=1}^{N} A_j E_j F_{ji} \varepsilon_{g,ij} \right)$$

$E_{g,ij}$ gas black body emission power of gas between surfaces *i* and *j*, equal to: $\sigma T_{g,ij}^4$, (W/m^2),
$\varepsilon_{g,ij}$ gas emissivity on the path between surfaces *i* and *j*, (-),
$\tau_{g,ij}$ gas transmittance on the path between surfaces *i* and *j*, (-).

In the above equation the gas emission, $\varepsilon_{g,ij} \times E_{g,ij}$, is written for each individual radiation path *i-j*, rather than using a single value, $\varepsilon_g \times E_g$, as in reference [113]. This is done to keep the formulation general, and applicable in case of several enclosures with different gases (different Control Volumes).

If only one enclosure is considered, then according to assumption e) the gas emission is the same on each path, and the term $\varepsilon_{g,ij} \times E_{g,ij}$ is constant ($=\varepsilon_g \times E_g$) and may be moved before the summation sign. In this case the sum gives 1.0 (which follows from the closure relation) and the term for gas emissivity becomes the same as in [113].

Taking into account the surface reflectivity: $\rho_i = 1 - \varepsilon_i$, the gas transmittance: $\tau_g = 1 - a_g$ and using the reciprocity relation, the basic equation can be written as:

$$H_i = \varepsilon_i E_i + (1-\varepsilon_i)\left(\sum_{j=1}^{N} H_j F_{ij} (1-a_{g,ji}) + \sum_{j=1}^{N} E_{g,ji} F_{ij} \varepsilon_{g,ji} \right)$$

The equation may be rearranged to obtain a set of linear equations, similarly as in the case of non-absorbing medium:

$$CH = E$$

The coefficients of the matrix ***C***, and the vector ***E***, are given by:

$$C_{ij} = \frac{\delta_{ij} - (1-\varepsilon_i) F_{ij} (1-a_{g,ji})}{\varepsilon_{ij}}$$

$$E_i = \sigma T_i^4 + \frac{1-\varepsilon_i}{\varepsilon_i} \sum_{i=1}^{N} F_{ij} \varepsilon_{g,ji} \sigma T_{g,ji}^4$$

A standard procedure to solve linear equations is used to solve the matrix, and calculate the surface radiosities, H_i.

When all radiosities, H_i, are determined, the heat flux lost from the surface *i* due to thermal radiation can be calculated as a difference between the radiosity and the incoming radiation flux ([113] section 2.1):

$$A_i q_i = A_i H_i - A_i G_i$$

which leads to:

$$q_i = H_i - \sum_{j=1}^{N} H_j F_{ij} (1-a_{g,ji}) - \sum_{j=1}^{N} E_{g,ji} F_{ij} \varepsilon_{g,ji}$$

As it was done in section 8.2, the sum of heat fluxes from all surfaces is calculated. This time the result should not give zero, but the energy absorbed in gas.

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N} A_i H_i - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i H_j F_{ij}(1-a_{g,ji}) - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i E_{g,ij} F_{ij} \varepsilon_{g,ji}$$

or:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N} A_i H_i - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i H_j F_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N} A_i H_j F_{ij} a_{g,ji} - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i E_{g,ij} F_{ij} \varepsilon_{g,ji}$$

Using the reciprocity and closure relations it may be easily shown that the first two terms on the right hand side are equal to zero (see section 8.2). Finally the sum of heat fluxes is equal to:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N}\sum_{j=1}^{N} A_i H_j F_{ij} a_{g,ji} - \sum_{i=1}^{N}\sum_{j=1}^{N} A_i E_{g,ij} F_{ij} \varepsilon_{g,ji}$$

or:

$$\sum_{i=1}^{N} A_i q_i = \sum_{i=1}^{N}\sum_{j=1}^{N} A_i F_{ij} (H_j a_{g,ji} - E_{g,ij} \varepsilon_{g,ji})$$

The right hand side gives the total net absorption of energy by gas, and is identical to the expression used in the program to calculate the radiative energy source for gas in Control Volumes. Thus the total energy is conserved provided that the reciprocity and closure relations are fulfilled.

Note that a single term of the above double sum represents the net absorption of energy by gas on the path from surface *i* to surface *j*:

$$Q_{g,ij} = A_i F_{ij} (H_j a_{g,ji} - E_{g,ij} \varepsilon_{g,ji})$$

To write the above formula the reciprocity relation, $A_i \times F_{ij} = A_j \times F_{ji}$, has been used, and then the indices *j* and *i* have been swapped for convenience. Since the emissivity of gas depends only on gas properties (temperature, optical length) the order of indexes does not matter, as $\varepsilon_{g,ij} = \varepsilon_{g,ji}$. Similarly $E_{g,ij} = E_{g,ji}$. Absorptivity however depends not only on gas properties but also on surface temperature. Therefore generally $a_{g,ij} \neq a_{g,ji}$ and the order of indexes should not be mixed.

If a radiation beam passes through a single Control Volume on a path between surfaces *i* and *j*, then the interface between the Thermal Radiation package and the Control Volume package is quite simple. The gas properties, $\varepsilon_{g,ij}$, $a_{g,ij}$, are calculated from the Control Volume conditions, as shown in section 8.5. Basically, they are functions of the CV temperatures, T_{CV}, gas partial pressures, p_{CV}, beam length, L_{ij}, as well as wall surface temperatures, T_w:

$$\varepsilon_{g,ij} = \varepsilon(T_{CV}, p_{CV}, L_{ij})$$
$$a_{g,ij} = a(T_{CV}, p_{CV}, L_{ij}, T_w)$$

The gas heat absorption, $Q_{g,ij}$, calculated by the Thermal Radiation package, is in such case taken directly by the Control Volume package as a heat source for CV gas space.

$$Q_{CV} = Q_{g,ij}$$

If a radiation beam passes through several Control Volumes on a path between surfaces *i* and *j*, then the calculation procedure is more complicated. Such case is shortly described below.

It is possible to build a model where a radiation beam passes through several Control Volumes, on its way between surfaces *i* and *j*. For each of these CV's the user must specify a path length, L_{ijk}. The sum of all individual path lengths must be equal to the overall beam length on the path between surfaces *i* and *j*:

$$\sum_k L_{ijk} = L_{ij}$$

The radiation model is valid for a homogeneous, isothermal enclosure - assumption e), above. Therefore to perform the calculation the Thermal Radiation package needs to have representative values for the gas temperature, T_g, and the gas properties, $\varepsilon_{g,ij}$, $a_{g,ij}$. These values are obtained from the following relations:

$$1-\varepsilon_{g,ij} = \prod_k (1-\varepsilon_{g,ijk}) = \prod_k [1-\varepsilon(T_{CV}, p_{CV}, L_{ijk})]$$

$$1-a_{g,ij} = \prod_k (1-a_{g,ijk}) = \prod_k [1-a(T_{CV}, p_{CV}, L_{ijk}, T_w)]$$

$$\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4) = \sum_k \varepsilon_{g,ijk} L_{ijk} (T_{CV(k)}^4 - T_w^4)$$

where $\varepsilon(T, pL)$ and $a(T, pL, T_w)$ are the values obtained from emissivity and absorptivity correlations, described in section 8.5.

The last relation is just one of many possible formulations that can be used to average gas temperatures and heat fluxes. With this relation the gas in Control Volume *k* does not emit/absorb radiation if:

- beam length in this CV is infinitesimally small, $L_{ijk} \to 0.0$
- gas emissivity (and absorptivity) is equal to zero, $\varepsilon_{g,ijk} = 0.0$
- gas temperature is equal to wall temperature, $T_{g,ijk} = T_w$

The condition of zero emission/absorption in these three cases must be preserved for a realistic averaging of gas properties in the case when a radiation beam passes through many Control Volumes. The above relations lead to the following formulae for average values:

$$\varepsilon_{g,ij} = 1 - \prod_k [1-\varepsilon(T_{CV}, p_{CV}, L_{ijk})]$$

$$a_{g,ij} = 1 - \prod_k [1-a(T_{CV}, p_{CV}, L_{ijk}, T_w)]$$

$$T_g = \left[T_w^4 + \frac{\sum_k \varepsilon_{g,ijk} L_{ijk} (T_{CV(k)}^4 - T_w^4)}{\varepsilon_{g,ji} L_{ij}} \right]^{1/4}$$

The gas heat absorption, $Q_{g,ij}$, calculated by the TR package, is distributed over all Control Volumes on the beam path between surfaces *i* and *j*, using the following formula:

$$\frac{Q_{CV(k)}}{Q_{g,ij}} = \frac{\varepsilon_{g,ijk} L_{ijk} (T_{CV(k)}^4 - T_w^4)}{\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4)}$$

This formula is consistent with the formula for the gas average temperature calculation, and it may easily be checked that the overall heat is conserved, that means: $\sum_k Q_{CV(k)} = Q_{g,ij}$

$$\sum_k Q_{CV(k)} = \sum_k Q_{g,ij} \cdot \frac{\varepsilon_{g,ijk} L_{ijk} (T_{CV(k)}^4 - T_w^4)}{\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4)} = \frac{Q_{g,ij}}{\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4)} \cdot \sum_k \varepsilon_{g,ijk} L_{ijk} (T_{CV(k)}^4 - T_w^4)$$

According to the definition of the representative gas temperature, T_g, given above, the last sum is equal to:

$$\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4)$$

Therefore:

$$\sum_k Q_{CV(k)} = \frac{Q_{g,ij}}{\varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4)} \cdot \varepsilon_{g,ji} L_{ij} (T_g^4 - T_w^4) = Q_{g,ij}$$

8.4 Radiation in to Water Pool

Typically the radiation model is applied for "dry" systems, meaning when there is no water pool in a Control Volume. If radiation to a pool needs to be taken into account, then the user may activate a "pool option" for one of the SC surfaces forming the radiation enclosure (input parameter IPLRAD - see Volume 2). The pool option is available only for 1-D Solid Heat Conductors. The structure with the pool option must be a floor of the Control Volume. With the pool option the radiative flux is deposited at the SC surface in absence of a pool, or at the pool surface if a pool is present. A transition zone is defined (0, 10^{-3} m), in which the radiative fluxes are interpolated between the SC surface and the pool surface. The emissivity of the pool surface is assumed to be constant, equal to 0.96.

If the pool option is not used, then the thermal radiation model will calculate radiation fluxes as long as there is at least one structure of the enclosure that is uncovered. If all surfaces are covered by pool, then the radiation model is turned off. In the previous versions of SPECTRA the radiation model was turned off when there was a single uncovered surface. This was not correct because in general self-radiation is possible.

8.5 Radiative Properties of Gases

For the calculation of radiative heat exchange with a participating gas the radiation properties of gases, such as emissivities, absorptivities and transmittances, must be known. The models to calculate the radiation properties of gases and gas mixtures are presented in this section. Recommended correlations to calculate individual emissivities of steam and carbon dioxide, are described in sections 8.5.1, 8.5.2, 8.5.3. Section 8.5.4 describes the method to calculate emissivities and absorptivities of a mixture of gases. The emissivities of steam and CO_2 may be either computed from the recommended correlations, or from a general equation which includes the effect of aerosols, shown in section 8.5.5.

8.5.1 H_2O Emissivity

The following correlation is used in SPECTRA to calculate the emissivity of steam:

$$\varepsilon_{H_2O} = Min(C_H \cdot \varepsilon_{H,K} \, , \varepsilon_{H,\infty})$$

$\varepsilon_{H,K}$ steam emissivity at atmospheric pressure and small steam partial pressure
C_H correction factor for different total pressures and steam partial pressures
$\varepsilon_{H,\infty}$ maximum emissivity, emissivity for infinitely large optical length

The three terms: $\varepsilon_{H,K}$, C_H, and $\varepsilon_{H,\infty}$, are described below.

- H_2O emissivity at atmospheric pressure and low H_2O partial pressure, $\varepsilon_{H,K}$

Steam emissivity at atmospheric pressure and low steam partial pressure is calculated from the correlation proposed in [113], referred here as the Kostowski correlation. This particular correlation has been selected after extensive review of available models and comparison of their results with graphs with Hottel data [115]. The Kostowski correlation was found to give the best representation of the Hottel data. The emissivity correlation has the general form:

$$\varepsilon_{H,K} = 1 - \exp[-(a + bT) \cdot (p_{H_2O} \cdot L)^c]$$

$p_{H2O}L$ optical length, (kPa m)
T temperature, (°C)

The values of coefficients *a, b, c,* are shown in Table 8-1. The ranges of application for the coefficients *a, b, c,* originally recommended by Kostowski, are shown in this table. Those recommended ranges are a little overlapping each other, meaning that in some regions the value may be obtained from different sets of coefficients and they should give similar values. For the purpose of numerical implementation only one equation must be selected in each case. Moreover, the obtained emissivity line should be smooth, thus discontinuities, even small, should be avoided.

To achieve this, the actual ranges of application of parameters (*a, b, c*) have been narrowed, leaving "gaps" between regions. In the "gaps" the emissivity is calculated by linear interpolation between the values obtained for the neighboring regions. The actual ranges are shown in Figure 8-1.

Results of the correlation are compared to the Hottel data in Figure 8-2. A good general agreement is observed. Differences are about 10% to 20%.

Table 8-1 Coefficients for Kostowski correlation for H_2O [113]

T range (°C)		*pL* range, (kPa m)				
		(1) 0.093 - 1.0	(2) 0.93 - 5.0	(3) 4.0 - 10.0	(4) 10.0 - 80.0	(5) 70.0 - 200.0
(1) 200 - 800	*a*	+0.06421	+0.05772	+0.06151	+0.07526	+0.11039
	$b \cdot 10^3$	–0.05438	–0.04022	–0.03602	–0.03358	–0.03959
	c	+0.797	+0.672	+0.580	+0.470	+0.367
(2) 800 - 1400	*a*	+0.04433	+0.03892	+0.04210	+0.05729	+0.09700
	$b \cdot 10^3$	–0.02552	–0.02027	–0.01979	–0.02375	–0.03809
	c	+0.945	+0.814	+0.692	+0.530	+0.395
(3) 1400 - 2000	*a*	+0.02510	+0.02197	+0.02273	+0.03677	+0.07379
	$b \cdot 10^3$	–0.01066	–0.00870	–0.00828	–0.01211	–0.02274
	c	+1.117	+0.947	+0.831	+0.588	+0.405

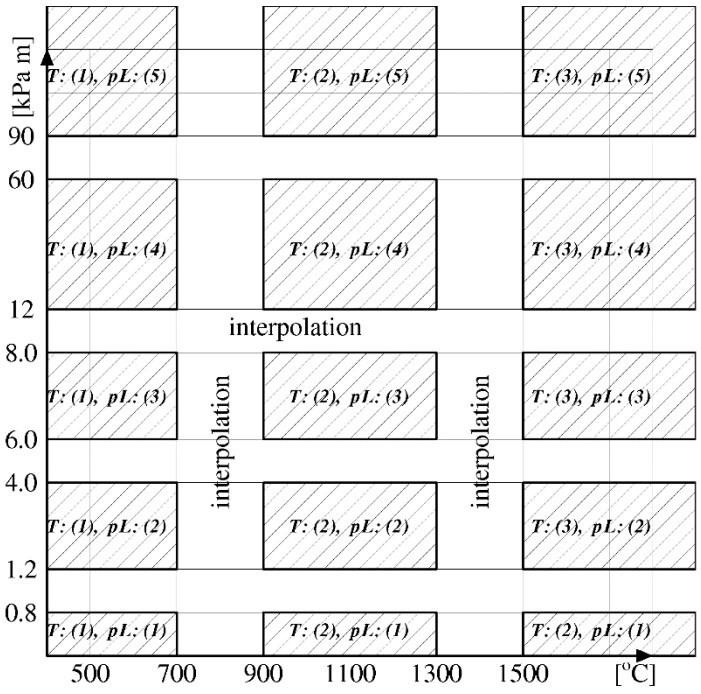


Figure 8-1 Applicability ranges for different sets of coefficients.

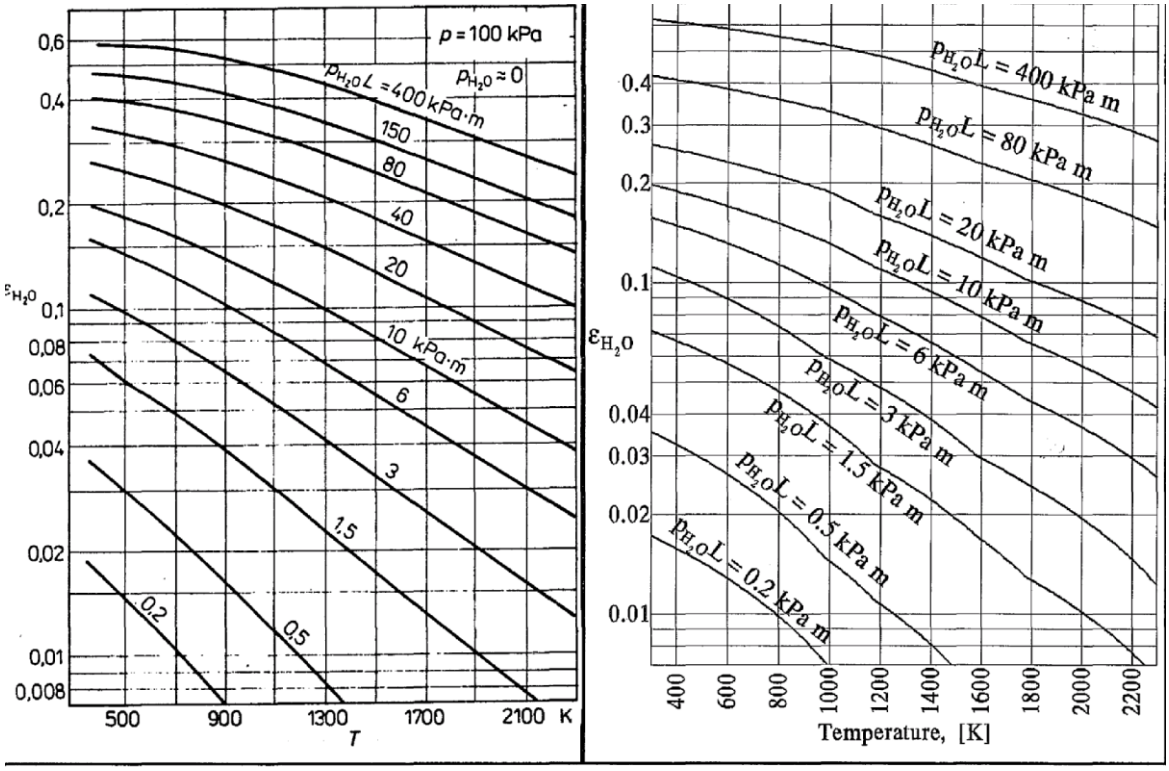


Figure 8-2 H2O emissivity, left - Hottel data, right - model applied in SPECTRA

- Correction for total pressure and steam partial pressure, C_H

The emissivity correlation described above is valid for atmospheric pressure and low steam partial pressures. For different conditions the correction factor, C_H, must be used. C_H depends on the optical length, $p_{H2O}L$, and $(p+p_{H2O})/2$. For computer calculation the values of C_H must either be tabulated or represented by a correlation. Reference [113] shows a correlation which allows to represent the correction factor C_H, but only in case when the total pressure, p, is equal to 10^5 Pa, and the steam partial pressure is within the range 0 - 4×10^3 Pa. This means that it covers only part of the available data. Therefore a broad-range expression was developed, to be used by the SPECTRA code, which is valid for the whole range of parameters shown in the Hottel graph. The expression is:

$$C_H = 1.0 + \left(2.8 - 0.4 \cdot \log_{10}(p_{H_2O}L)\right) \cdot \left(\beta - 9.45 \cdot \beta^2\right)$$

where:

$$\beta = \frac{(p + p_{H_2O})/2 - 0.5\times10^5}{10^5}$$

$p_{H2O}L$ optical length, (Pa m)
p_{H2O} steam partial pressure, (Pa)
p pressure, (Pa)
L radiation beam length, (m)

The following restrictions are imposed on the arguments:

$$1500 < p_{H2O}L < 1.0\times10^{7}$$

$$0.0 < \frac{p + p_{H2O}}{2} < 1.61\times10^{5}$$

With the above restrictions the applicability range of this correlation is extended into higher pressure ranges. Results of the correlation are compared to the Hottel data in Figure 8-3 and Figure 8-4. A good agreement is observed.

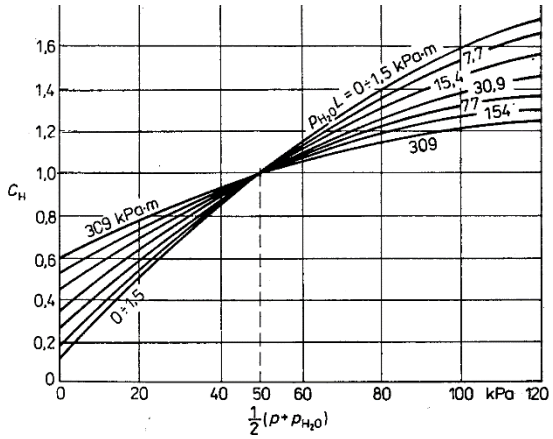


Figure 8-3 Pressure correction factor for H_2O - Hottel data

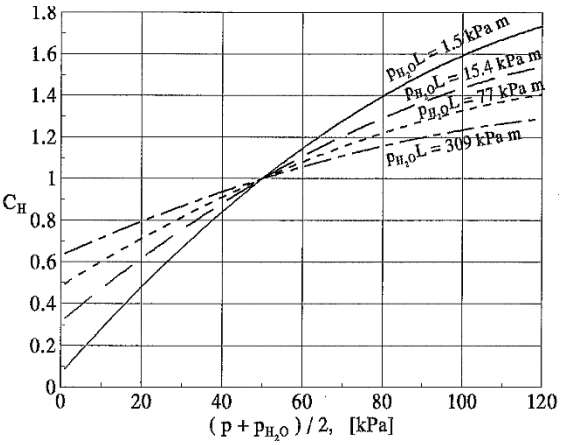


Figure 8-4 Pressure correction factor for H_2O - model applied in SPECTRA

- Maximum emissivity, $\varepsilon_{H,\infty}$

For a gray gas the emissivity (as well as the absorptivity) is equal to 1.0 for a very thick gas layer. In reality emission and absorption occurs only in certain spectral regions. When a radiation beam passes through a thick gas layer the absorbable wavelengths are filtered out, but for the remaining wavelengths the transmittance of the gas is 1.0. Thus in contrast to the gray gas model, in reality the absorptivity and emissivity of the gas are never equal to 1.0 (see [115], page 112, [113], page 127).

The values of the maximum emissivity, $\varepsilon_{H,\infty}$, for an infinitely large optical length: $p_{H2O}L \rightarrow \infty$ are shown in [113] (figure 6.15) as a function of gas temperature. For the purpose of numerical calculation the values were tabulated, as shown in Table 8-2. The program calculates $\varepsilon_{H,\infty}$ using linear interpolation of the tabular data. The values of $\varepsilon_{H,\infty}$, as calculated by the program, are shown in Figure 8-5 (as shown in Volume 3, Figure 8-5 is practically identical as the figure 6.15 in [113]).

Table 8-2 Maximum emissivity of H_2O as a function of temperature

T, (°C)	$\varepsilon_{H,\infty}$	T, (°C)	$\varepsilon_{H,\infty}$	T, (°C)	$\varepsilon_{H,\infty}$
0.0	0.760	800.0	0.677	1600.0	0.500
100.0	0.725	900.0	0.656	1700.0	0.470
200.0	0.705	1000.0	0.635	1800.0	0.440
300.0	0.700	1100.0	0.615	1900.0	0.410
400.0	0.700	1200.0	0.595	2000.0	0.380
500.0	0.700	1300.0	0.575	2100.0	0.350
600.0	0.700	1400.0	0.552	2200.0	0.320
700.0	0.700	1500.0	0.527		

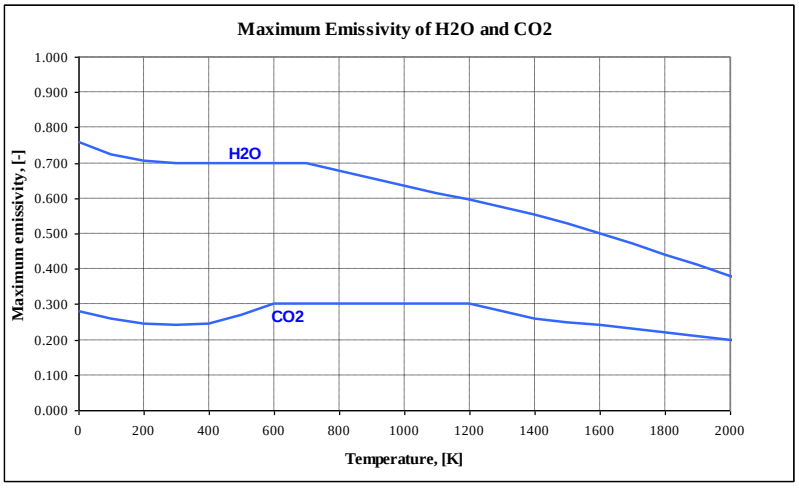


Figure 8-5 Maximum emissivity of H_2O and CO_2, as calculated by SPECTRA

8.5.2 CO_2 Emissivity

The following correlation is used in SPECTRA to calculate the emissivity of CO_2:

$$\varepsilon_{CO_2} = Min(C_C \cdot \varepsilon_{C,K} \, , \varepsilon_{C,\infty})$$

$\varepsilon_{C,K}$ CO_2 emissivity at atmospheric pressure
C_C correction factor for different total pressures
$\varepsilon_{C,\infty}$ maximum emissivity, emissivity for infinitely large optical length

The three terms: $\varepsilon_{C,K}$, C_C, and $\varepsilon_{C,\infty}$, are described below.

- CO_2 emissivity at atmospheric pressure, $\varepsilon_{C,K}$

The CO_2 emissivity at atmospheric pressure is calculated from the correlation proposed in [113], referred here as the Kostowski correlation. This particular correlation has been selected after extensive review of available models and comparison of their results with graphs containing Hottel data [115]. The Kostowski correlation was found to give the best representation of the Hottel data. The correlation has the general form:

$$\varepsilon_{H,K} = 1 - \exp[-(a + bT) \cdot (p_{CO_2} \cdot L)^c]$$

Here $p_{CO2}L$ is the optical length, (kPa m), and T is the temperature, (°C). The values of coefficients a, b, c, are shown in Table 8-3. The ranges of application for the coefficients a, b, c, originally recommended by Kostowski, are shown in this table. As in case of steam emissivity correlation, the overlapping ranges have been narrowed, leaving "gaps" in which linear interpolation is performed. The "map" of the actual ranges is shown in Figure 8-1. Results of the correlation are compared to the Hottel data in Figure 8-6. A good general agreement is observed. Differences are about 10% to 20%.

Table 8-3 Coefficients for Kostowski correlation for CO_2 [113]

T range (°C)		pL range, (kPa m)				
		(1) 0.093 - 1.0	(2) 0.93 - 5.0	(3) 4.0 - 10.0	(4) 10.0 - 80.0	(5) 70.0 - 200.0
(1) 200 - 800	a	+0.05320	+0.04596	+0.04608	+0.05000	+0.05070
	$b \cdot 10^3$	–0.00168	+0.01220	+0.01707	+0.02277	+0.03051
	c	+0.527	+0.345	+0.308	+0.262	+0.244
(2) 800 - 1400	a	+0.08697	+0.07814	+0.07613	+0.07791	+0.07350
	$b \cdot 10^3$	–0.04108	–0.03321	–0.03038	–0.02573	–0.02081
	c	+0.614	+0.391	+0.374	+0.314	+0.310
(3) 1400 - 2000	a	+0.06787	+0.06136	+0.06099	+0.06579	+0.06707
	$b \cdot 10^3$	–0.02619	–0.02245	–0.02146	–0.02228	–0.02193
	c	+0.672	+0.449	+0.414	+0.362	+0.346

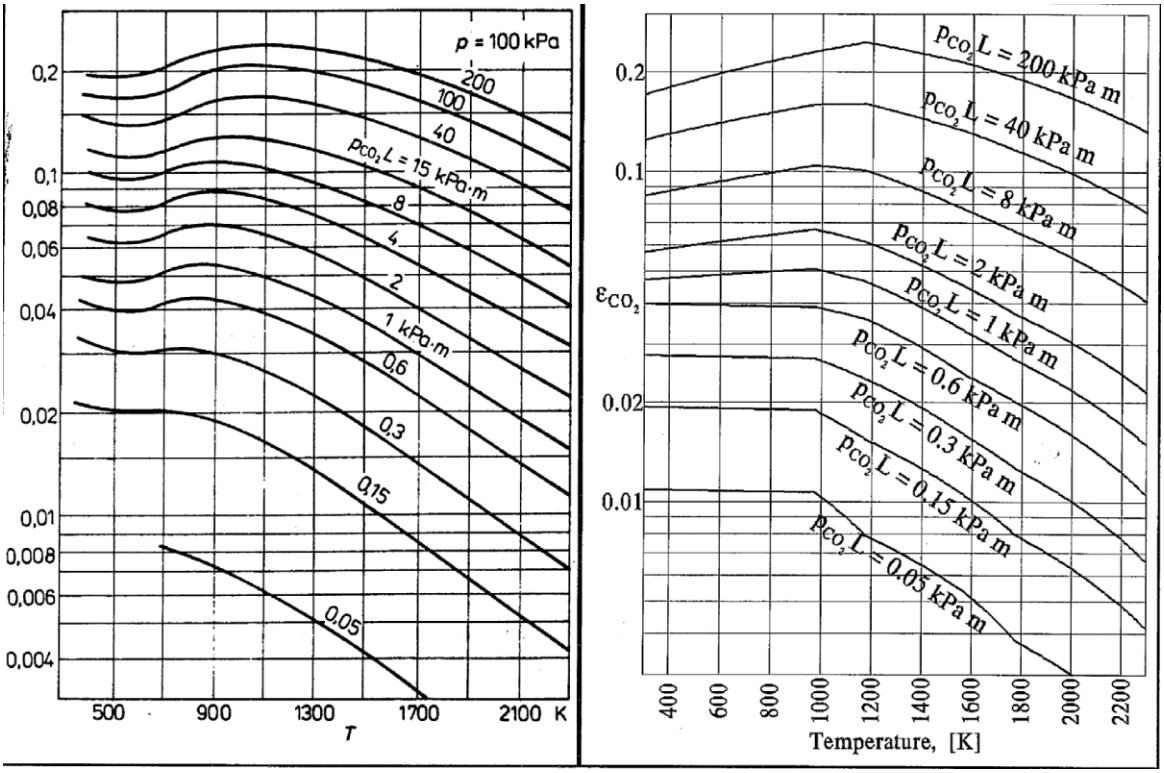


Figure 8-6 CO_2 emissivity, left - Hottel data, right - model applied in SPECTRA

- Correction for total pressure, C_C

The emissivity correlation described above is valid for atmospheric pressure. For different conditions the correction factor, C_C, must be used. C_C depends on the optical length, $p_{CO2}L$, and p. For computer calculation the values of C_C must either be tabulated or represented by a correlation. An expression was developed to be used by the SPECTRA code, which gives a good approximation of the Hottel data. The expression is:

$$\log_{10}(C_C) = \left(0.675 - 0.115 \cdot \log_{10}(p_{CO_2} L)\right) \cdot \left(\log_{10}(p/10^5) - 0.12 \cdot \{\log_{10}(p/10^5)\}^2\right)$$

Here $p_{CO2}L$ is the optical length, (Pa m), p_{CO2} is CO_2 partial pressure, (Pa), p is thew total pressure, (Pa) and L is the radiation beam length, (m). The following restrictions are imposed on the arguments:

$$0.6 \times 10^3 < p_{CO_2} L < 740 \times 10^3$$

$$0.0 < p < 1.0 \times 10^6$$

With the above restrictions the applicability range of this correlation is extended into higher pressure ranges. Results of the correlation are compared to the Hottel data in Figure 8-7 and Figure 8-8. A good agreement is observed.

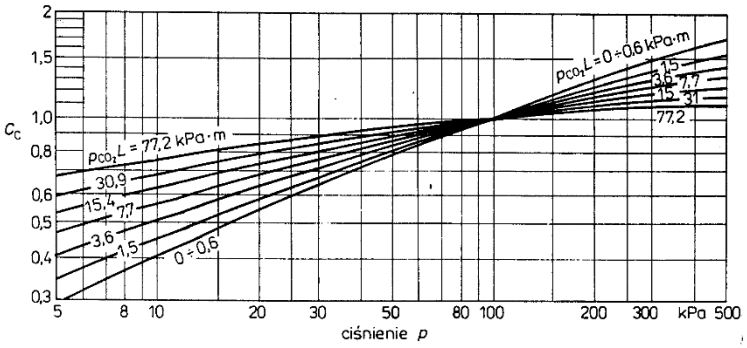


Figure 8-7 Pressure correction factor for CO_2 - Hottel data

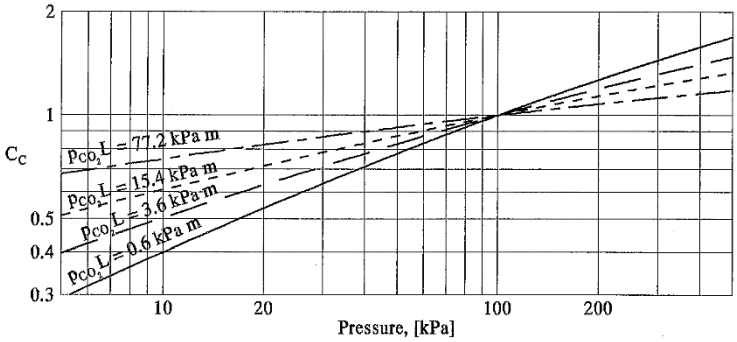


Figure 8-8 Pressure correction factor for CO_2 - model applied in SPECTRA

- Maximum emissivity, $\varepsilon_{C,\infty}$

Similarly as in case of steam, an upper limit is imposed on the CO_2 emissivity, $\varepsilon_{C,\infty}$. The values of the maximum emissivity, $\varepsilon_{C,\infty}$, are shown in [113] (figure 6.15). The program uses linear interpolation of the tabulated values, shown in Table 8-4. The values of $\varepsilon_{C,\infty}$, as calculated by the program, are shown in Figure 8-5.

Table 8-4 Maximum emissivity of CO_2 as a function of temperature

T, (°C)	$\varepsilon_{C,\infty}$	T, (°C)	$\varepsilon_{C,\infty}$	T, (°C)	$\varepsilon_{C,\infty}$
0.0	0.280	800.0	0.300	1600.0	0.240
100.0	0.260	900.0	0.300	1700.0	0.230
200.0	0.245	1000.0	0.300	1800.0	0.220
300.0	0.240	1100.0	0.300	1900.0	0.210
400.0	0.244	1200.0	0.300	2000.0	0.200
500.0	0.270	1300.0	0.280	2100.0	0.190
600.0	0.300	1400.0	0.260	2200.0	0.180
700.0	0.300	1500.0	0.250		

8.5.3 Emissivity Correction for H_2O and CO_2 Spectral Overlap

The emissivity correction for spectral overlap is shown in form of graphs in [115], [113]. The graphs shown in [113] are given in SI units, and are reproduced in Figure 8-9.

Typically the computer codes use the Leckner model [65] (MELCOR, CONTAIN) to calculate this correction factor. However, it was found that the results obtained with the Leckner correlation do not represent the Hottel data much better than the gray gas model (which gives the correction factor equal to the product of individual emissivities, $\Delta\varepsilon_g = \varepsilon_{H2O} \times \varepsilon_{CO2}$). An expression was developed to be used by the SPECTRA code, which gives a good approximation of the Hottel data. The expression is:

$$\Delta\varepsilon_g = \left(\sin\left[\pi \cdot \left(\frac{p_{H_2O}}{p_{H_2O} + p_{CO_2}} \right)^a \right] \right)^b \cdot \left(\frac{(p_{H_2O} + p_{CO_2})L - 5.99 \times 10^3}{1.48 \times 10^5} \right)^c \times$$

$$\times (1.01 \times 10^{-7} T^2 - 1.62 \times 10^{-4} T + 9.85 \times 10^{-2})$$

with:

$$a = 3.90 - 2.1 \times 10^{-3} T$$

$$b = 0.35 - 1.2 \times 10^{-4} T$$

$$c = 1.00 - 5.9 \times 10^{-4} T$$

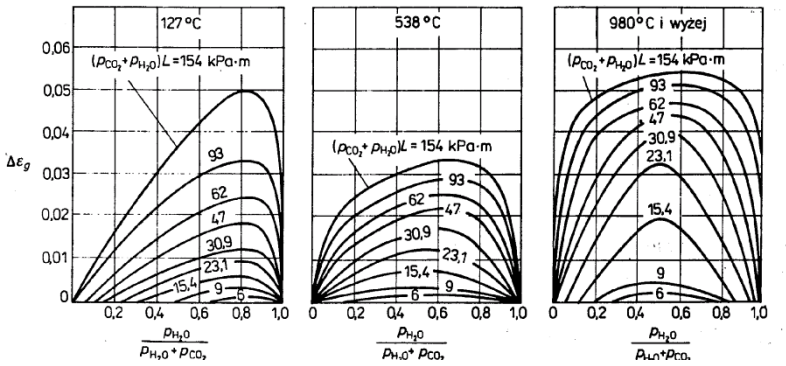


Figure 8-9 H_2O and CO_2 spectral overlap correction factor - Hottel data

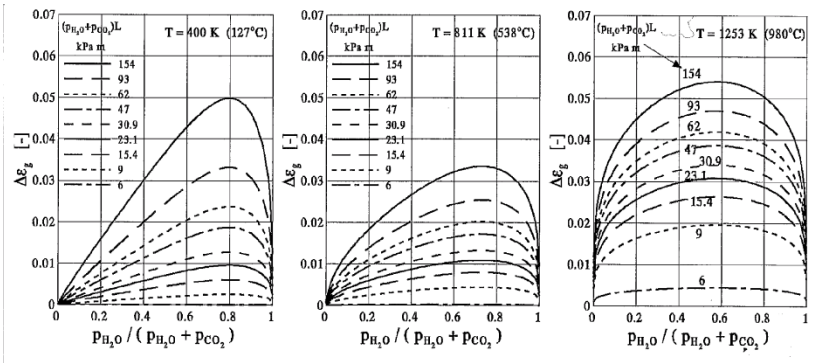


Figure 8-10 H_2O and CO_2 spectral overlap correction factor - model applied in SPECTRA

In the above formulae T is the temperature in (K), p_{H2O} and p_{CO2} are the partial pressures (Pa) of H_2O and CO_2, respectively, and L is the radiation beam length, (m). Results of the correlation are compared to the Hottel data in Figure 8-9 and Figure 8-10.

8.5.4 Emissivity and Absorptivity of a Gas Mixture

The general method, used to calculate the emissivity and absorptivity of any gas mixture, is described in this section. The discussion is divided into two parts. In the first part a method to calculate the gas mixture emissivity is described. The second part brings description of the mixture absorptivity calculation.

- Gas Emissivity

The emissivity of a mixture of gases is calculated in two steps. First, the emissivities of the individual gases are calculated. Individual emissivities are calculated using either the recommended correlations, described above in sections 8.5.1, 8.5.2, 8.5.3, or a general equation, described below, with user defined constants.

The recommended correlations for the calculation of the emissivity of gas k, ε_k, are:

- H_2O Kostowski correlation $\varepsilon_k = \text{Min}(C_H \varepsilon_{H,K}, \varepsilon_{H,\infty})$ - section 8.5.1
- CO_2 Kostowski correlation $\varepsilon_k = \text{Min}(C_C \varepsilon_{C,K}, \varepsilon_{C,\infty})$ - section 8.5.2
- Other no radiation $\varepsilon_k = 0.0$

The general equation has the form similar to that shown in [113]:

$$\varepsilon_k = \varepsilon_\infty \cdot \{1 - \exp[-(a_k + b_k T) \cdot (p_k \cdot L)^{c_k}]\}$$

ε_k	emissivity of the gas k, (-)
p_kL	optical length of the gas k, (Pa m)
T	temperature, (K)
$\varepsilon_{k,\infty}$	maximum emissivity, emissivity of the gas k, for $p_kL \to \infty$
a_k, b_k, c_k	user-defined constants

The general equation may be applied for all gases, also steam and carbon dioxide, thus overruling the default Kostowski correlations. The general correlation allows to obtain in a simple way any desired gas emissivity. For example, in the test cases presented in Volume 3, for which constant gas emissivity was desired ("screen" test, "Christiansen system" test, etc.) the general correlation was used with the following values: $a_k = 10^{10}$, $b_k = 0.0$, $c_k = 0.0$, and $\varepsilon_{k,\infty}$ equal to the desired emissivity value.

When the individual emissivities of all component gases are determined the emissivity of the mixture of gases is calculated using the gray gas approximation. The transmittance of the gas mixture is equal to the product of individual gas transmittances [113], section 5.3):

$$\tau_{mix} = \prod_{k=1}^{N_{gas}} \tau_k$$

τ_{mix} transmittance of gas mixture, for Hottel gas equal to $1 - \varepsilon_{mix}$, (-),
τ_k transmittance of gas k, for Hottel gas equal to $1 - \varepsilon_k$, (-),
N_{gas} number of gases in the mixture.

This leads to the following expression for the emissivity of the mixture:

$$\varepsilon_{mix} = 1 - \prod_{k=1}^{N_{gas}} (1 - \varepsilon_k)$$

where ε_{mix} is the emissivity of the gas mixture, while ε_k is the emissivity of the gas *k*.

The correlation equation for the spectral overlap of H_2O and CO_2 may be requested if the emissivity of at least one of the two gases: H_2O or CO_2, is calculated using the recommended (Kostowski) correlation. If it is requested, then the correlation for the H_2O and CO_2 overlap (section 8.5.3) is superimposed on the gray gas model as follows.

The general expression for a gas emissivity contains the product $\varepsilon_{H2O} \times \varepsilon_{CO2}$:

$$\varepsilon_{mix} = 1 - \prod_{k=1}^{N_{gas}} (1 - \varepsilon_k) = \ldots - \varepsilon_{H_2O} \cdot \varepsilon_{CO_2} + \ldots$$

This product represents the gray gas overlap correction factor. Thus if the correlation for $\Delta\varepsilon_g$ needs to be applied, then the product $\varepsilon_{H2O} \times \varepsilon_{CO2}$ should be replaced by $\Delta\varepsilon_g$. The formula for gas mixture emissivity becomes:

$$\varepsilon_{mix} = 1 - \prod_{k=1}^{N_{gas}} (1 - \varepsilon_k) + \varepsilon_{H_2O} \cdot \varepsilon_{CO_2} - \Delta\varepsilon_g$$

When only steam and carbon dioxide are present in the mixture, the mixture emissivity is:

Gray gas	$\varepsilon_{mix} = \varepsilon_{H_2O} + \varepsilon_{CO_2} - \varepsilon_{H_2O} \cdot \varepsilon_{CO_2}$
Overlap correction	$\varepsilon_{mix} = \varepsilon_{H_2O} + \varepsilon_{CO_2} - \Delta\varepsilon_g$

The gray gas model: $\Delta\varepsilon_g = \varepsilon_{H_2O} \cdot \varepsilon_{CO_2}$ is used if EMSMAX>0.0 (see Volume 2). The correlation for the overlap correction factor, $\Delta\varepsilon_g = f(T, p_{H_2O}, p_{CO_2})$, shown in Figure 8-10, is used if EMSMAX<0.0 (see Volume 2), with the limit set by the gray gas model: $\Delta\varepsilon_g \leq \varepsilon_{H_2O} \cdot \varepsilon_{CO_2}$. The gray gas limit is active at high temperatures (see Volume 3, "Emissivity Correction for Spectral Overlap of H_2O and CO_2").

In cases when not only steam and carbon dioxide are present in the mixture and the correlation for $\Delta\varepsilon_g$ is used, the method described above is not quite consistent, because it will not use $\Delta\varepsilon_g$ for the third (or higher) order terms. For example, in case of three radiating gases, the term ($\varepsilon_{H2O} \times \varepsilon_{CO2} \times \varepsilon_k$) will be used rather than ($\Delta\varepsilon_g \times \varepsilon_k$), where ε_k is the third gas emissivity. Those terms are typically very small and have negligible contribution to the overall emissivity. Thus there is no need to calculate them with very good accuracy.

An important fact in calculating the emissivity of the gas mixture is that physically the total emissivity can never be greater than 1.0. The gray gas equation, $1-\varepsilon_{mix} = \Pi(1-\varepsilon_k)$, will always yield ε_{mix} between 0.0 and 1.0, provided that the individual emissivities are also within this range. When the model for H_2O and CO_2 overlap correction, $\Delta\varepsilon_g$, is superimposed this may not necessarily be true. Therefore an additional check is made to ensure that the calculated value is in permitted range.

- Gas Absorptivity

The absorptivity of a mixture of gases is calculated in a similar way as the emissivity. First absorptivities of individual gases are calculated. Individual absorptivities are calculated using the relation (see [115] page 90):

$$a_g = \left(\frac{T_g}{T_w}\right)^n \cdot \varepsilon_g(T_w, pL\frac{T_w}{T_g})$$

T_g gas temperature, (K),
T_w wall temperature, (K), (in case of $a_{g,ij}$ this is the temperature of the surface *i*, while for $a_{g,ji}$ this is the temperature of the surface *j*)
ε_g gas emissivity, evaluated at wall temperature and optical length multiplied by (T_w/T_g), (-)

The exponent n is an input parameter. The default values are: for H_2O n=0.45; for CO_2 n=0.65; for other gases n=0.5.

The value of maximum absorptivity, a_∞, is determined similarly as the value of maximum emissivity, ε_∞. The argument for determining a_∞ is the wall temperature rather than the gas temperature, as in case of ε_∞.

When the individual absorptivities are determined the overall absorptivity is found using the gray gas assumption, which leads to the expression:

$$a_{mix} = 1 - \prod_{k=1}^{N_{gas}} (1 - a_k)$$

The correction term for the H_2O and CO_2 spectral overlap is calculated using the same equation which is valid for the emissivity correction factor. The argument is the wall temperature ([115] page 90):

$$\Delta a_g = \Delta\varepsilon_g\left(T_w, (p_{H_2O} + p_{CO_2})L\frac{T_w}{T_g}, \frac{p_{H_2O}}{p_{H_2O} + p_{CO_2}}\right)$$

To be consistent with the expression for gas individual absorptivity, the correction factor should be proportional to the term $(T_g/T_w)^n$.

If the default values for n are used, then the correction factor is proportional to: $\Delta a_g \sim a_{CO2} \times a_{H2O} \sim (T_g/T_w)^{0.45+0.65}$, and finally:

$$\Delta a_g = \left(\frac{T_g}{T_w} \right)^{1.10} \cdot \Delta \varepsilon_g (T_w , (p_{H_2O} + p_{CO_2}) L \frac{T_w}{T_g} , \frac{p_{H_2O}}{p_{H_2O} + p_{CO_2}})$$

The correction term for the H_2O and CO_2 spectral overlap is superimposed on the gray gas model, similarly as in case of the emissivity calculation.

8.5.5 Gas Emissivity and Absorptivity in Presence of Aerosols

The presence of aerosol particles in the gas changes the emissivity and absorptivity of the medium. This section describes how the emissivity and absorptivity are calculated in presence of aerosols.

- Emissivity of Gas and Aerosols

The emissivity of a gas with aerosol particles is calculated from:

$$\varepsilon_{mix} = 1 - (1 - \varepsilon_{aer}) \cdot \prod_{k=1}^{N_{gas}} (1 - \varepsilon_k)$$

Here ε_{aer} is the emissivity of aerosol particles while ε_k is the emissivity of the gas *k*. The aerosol emissivity is calculated from ([113], equation 5.8):

$$\varepsilon_{aer} = 1 - \exp\left[- L \cdot \sum_{i=1}^{N_{size}} n_i \cdot \sigma_i \right]$$

ε_{aer}	emissivity of aerosols, (-)
L	radiation beam length, (m)
n_i	density of aerosol size section *i*, ($1/m^3$) (number of particles per unit volume)
N_{size}	number of aerosol size sections
σ_i	cross section for radiation absorption, (m^2), assumed to be temperature-dependent:

$$\sigma_i = \sigma_{i,0} \cdot (a_{aer} + b_{aer} T^{c_{aer}})$$

Here $\sigma_{i,0}$ is the geometrical cross section of the aerosol particle, equal to ($\pi D_i^2 / 4$). Therefore:

$$\sigma_i = \left(\frac{\pi D_i^2}{4} \right) \cdot (a_{aer} + b_{aer} T^{c_{aer}})$$

D_i	diameter of aerosol size section *i*, (m)
T	temperature, (K)
a_{aer}, b_{aer}, c_{aer}	user-defined constants

The final expression for the aerosol emissivity is:

$$\varepsilon_{aer} = 1 - \exp\left[- L \cdot \sum_{i=1}^{N_{size}} n_i \cdot \left(\frac{\pi D_i^2}{4} \right) \cdot (a_{aer} + b_{aer} T^{c_{aer}}) \right]$$

The model constants a_{aer}, b_{aer}, c_{aer} characterize the aerosol particles and in general should be found in thermal radiation handbooks. In absence of a more detailed data, a simple way to take the aerosol emissivity into account is to use the following values of the model constants:

- $a_{aer} = 1.0$ $\quad b_{aer} = 0.0$ $\quad c_{aer} = 0.0$

In such case the aerosol emissivity is equal to:

$$\varepsilon_{aer} = 1 - \exp\left[- L \cdot \sum_{i=1}^{N_{size}} n_i \cdot \left(\frac{\pi D_i^2}{4} \right) \right]$$

- Absorptivity of Gas and Aerosols

The absorptivity of a gas with aerosol particles is calculated from:

$$a_{mix} = 1 - (1 - a_{aer}) \cdot \prod_{k=1}^{N_{gas}} (1 - a_k)$$

Here a_{aer} is the absorptivity of aerosol particles. The aerosol absorptivity is assumed to be equal to the emissivity:

$$a_{aer} = \varepsilon_{aer}$$

Therefore:

$$a_{aer} = 1 - \exp\left[- L \cdot \sum_{i=1}^{N_{size}} n_i \cdot \left(\frac{\pi D_i^2}{4} \right) \cdot (a_{aer} + b_{aer} T^{c_{aer}}) \right]$$

8.6 Radiative Properties of Solid Materials

Surface emissivity data is given in literature either as total emissivity, or spectral emissivity. For the radiation models presented in sections 8.2 and 8.3 the total emissivities must be used. If spectral emissivity data is available the total emissivity may easily be calculated using the function TSEMIS, described here.

Using the band approximation the total emissivity is given by:

$$\varepsilon(T) = \frac{1}{\sigma T^4} \cdot \sum_{i=1}^{N} \varepsilon_i \int_{\lambda_{i-1}}^{\lambda_i} \frac{C_1 \lambda^{-5}}{\exp(C_2 / \lambda T) - 1} \, d\lambda$$

C_1 constant: 3.7418×10^{-16} [113] page 25, (W m^2)
C_2 constant: 1.4388×10^{-2} [113] page 25, (m K)
λ wavelength, (m)
T temperature, (K)
σ Stefan-Boltzmann constant (equal to: $C_1/15\times(\pi/C_2)^4 = 5.67\times10^{-8}$) [113] page 28, (W/m^2/K^4)
λ_i upper boundary wavelength for band i, (m)
ε_i emissivity, for band i, (-)

The following expression:

$$F(\lambda,T) = \frac{1}{\sigma T^4}\cdot\int_0^{\lambda}\frac{C_1\lambda^{-5}}{\exp(C_2/\lambda T)-1}\,d\lambda$$

may be calculated using Wiebelt approximation formulae, given in [116] (as well as [113], section 1.4). The formulae are:

- For $x < 2$:

$$F(\lambda,T) = 1-\frac{15}{\pi^4}\cdot\left(\frac{1}{3}-\frac{x}{8}+\frac{x^2}{60}-\frac{x^3}{5{,}040}+\frac{x^4}{272{,}160}+\frac{x^8}{13{,}305{,}600}\right)$$

- For $x > 2$:

$$F(\lambda,T) = \frac{15}{\pi^4}\cdot\sum_{n=1}^{\infty}\frac{\exp(-nx)}{n^4}\{[(nx+3)\cdot nx+6]\cdot nx+6\}$$

Here $x = C_2(\lambda T)$. It is sufficient to use the first 3 - 4 terms of the sum in the formula for $x > 2$. This is because for $n>5$ and $x>2$ the value of $\exp(-nx)/n^4$ is smaller than 10^{-8}, and thus those terms have negligible influence on the result. Subroutine TSEMIS always uses the first four terms of the sum.

When the values of $F(\lambda,T)$ are calculated, the total emissivity, which is of course a function of temperature, T, is obtained as:

$$\varepsilon(T) = \sum_{i=1}^{N}\varepsilon_i\cdot[F(\lambda_i T)-F(\lambda_{i-1}T)]$$

Note that the subroutine TSEMIS, although it is a part of the Thermal Radiation Package in SPECTRA, is not used during SPECTRA calculations. In SPECTRA the total (temperature-dependent) emissivities , $\varepsilon(T)$, are used for solid surfaces. Volume 3 provides an example of how to use the subroutine TSEMIS to obtain a total emissivity, $\varepsilon(T)$ based on spectral emissivities, ε_i, for gold.

In practical cases there is no need to use the function TSEMIS to calculate the total emissivity because the total emissivities, $\varepsilon(T)$, obtained from experiments are available in literature for many materials. Spectral emissivities, ε_i, are in fact much more difficult to find.

9 Reactor Kinetics Package

9.1 Introduction

The Reactor Kinetics Package in SPECTRA consists of the following parts (models):

- Point Reactor Kinetics Model,
- Isotope Transformation Model,
- Reactor Kinetics Model for Circulating Fuel
- Nodal Point Kinetics Model.

These models allow calculating the power behavior of a nuclear reactor, including the immediate fission power and the power from decay of fission products. The effect of delayed neutrons is taken into account. Reactivity feedback from fuel temperature, moderator temperature and void fraction, as well as changes in core composition, are taken into account. The core composition changes caused by fuel burn-up, production of poisons (such as Xe-135, etc.), fuel reload, are calculated by the Isotope Transformation Model. The power resulting from decay of fission products is also calculated by the Isotope Transformation Model.

The models are described subsequently in the following sections. The Point Kinetics Model is described in section 9.2. The Isotope Transformation Model is described in section 9.3. The Reactor Kinetics Model for Circulating Fuel is described in section 9.4. The Nodal Point Kinetics Model is described in section 9.5.

9.2 Point Kinetics

9.2.1 Point Kinetics Equations

The derivation of the point kinetics equations may be found in [66] or [67]. The equations are ([67], equations 7.3.13):

$$\frac{dn(t)}{dt} = \frac{\rho(t) - \beta}{\Lambda} n(t) + \sum_i \lambda_i C_i(t) + S_{V,ext}(t)$$

$$\frac{dC_i(t)}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t)$$

n neutron concentration, (1/m^3),
C_i concentration of delayed neutron precursor group i, (1/m^3),
ρ reactivity, (-),
Λ prompt neutron generation time, (s), (constant or calculated by a Control Function)
λ_i decay constant of delayed neutron precursor group i, (1/s),
β_i yield fraction of delayed neutron precursor group i, (-),
β sum of delayed neutron fractions, $\beta = \Sigma \beta_i$, (-),
$S_{V,ext}$ external neutron source per unit volume per unit time, (1/m^3/s),
t time, (s).

Reactivity is frequently expressed in dollars. A reactivity of one dollar means that reactor is prompt-critical, that means the absolute reactivity ρ is equal to β. The relation between the reactivity in dollars (R), and the dimensionless reactivity (ρ) is therefore: $R(\$) = \rho/\beta$.

Using the reactivity in dollars, the point kinetics equations are written as:

$$\frac{dn(t)}{dt} = \frac{\beta(R(t)-1)}{\Lambda} n(t) + \sum_i \lambda_i C_i(t) + S_{V,ext}(t)$$
$$\frac{dC_i(t)}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t)$$

The reactor fission power, Q_f (W), is calculated based on the neutron concentration, n, from:

$$Q_f(t) = V_R \, \Sigma_f(t) \, \Phi(t) \, q_f = V_R \, \Sigma_f \, n(t) \, v_n \, q_f$$

where V_R is the reactor volume (m^3), Σ_f is the macroscopic fission cross section (1/m), Φ is the neutron flux, ($1/m^2/s$), q_f is the energy generated per fission (W/fission), v_n is the thermal neutron velocity. The thermal neutron velocity is calculated from $v_n=C_nT^{1/2}$. The initial value of the fission cross section, $\Sigma_f(t_0)$, is a user defined input parameter. The user must also specify the initial fission power, $Q_f(t_0)$. These values are used by the code to calculate the initial neutron density, $n(t_0)$.

Most of the model parameters have their default values, which are used if no value is entered, or zero is entered in the input deck. The default values are listed below.

- Λ the default value is calculated from: $\Lambda = (v \Sigma_f \, v_n)^{-1}$. This relation is obtained from: $\Lambda = (k_\infty v_n \Sigma_a)^{-1}$ ([68], section 5.3.1) and the one speed diffusion model: $\Sigma_a = (v \Sigma_f / k_\infty)$, ((Dude, 1976), equation 5-217).
- q_f $3.09 \cdot 10^{-11}$ (J/fission) (=192.9 MeV/fission), based on [66] table 2-5,
- C_n 128.0 ($m/s \cdot K^{-1/2}$), = $(2k/m)^{1/2}$ [66], equation 9-10 (the formula follows from $E = kT$)
- v number of neutrons per fission, equal to 2.5, based on [68], sections 1.5.1, 1.5.2,
- λ_i, β_i, delayed neutron precursor data, discussed in section 9.2.2.

A limit is imposed in the numerical solution scheme to prevent the neutron density, $n(t)$, from decreasing to zero. The applied scheme results in a minimum neutron density of about 1000 (m^{-3}) (1 neutron per litre) in case of a reactivity equal to –1.0 \$, and being inversely proportional to the reactivity. For example, at the reactivity of –10.0 \$ the minimum neutron flux is roughly equal to 100 (m^{-3}).

9.2.2 Delayed Neutron Precursors

In the past, six-groups of delayed neutron precursors (DNP) were often used (e.g.: [67], [224]). Currently, eight-group data is recommended [224]. An important fact, that needs to be remembered, is that the yield fractions of DNP depend on a number of factors, among others the energy of the neutron causing the fission and the kind of nucleus that undergoes the fission process. The data for thermal neutrons and three fissile isotopes, obtained from [224], is shown in Table 9-1. In the case of U-235, the total fission yield of all DNP groups is 0.0065, while for U-233 and Pu-239 it is much smaller, 0.0026 and 0.0021 respectively.

Table 9-1 Delayed neutron precursors data, six groups and eight-group data [224]

U^{235}

Group	Decay constant λ_i (sec^{-1})	Yield (neutrons per fission)	Fraction β_i
1	0.0124	0.00052	0.000215
2	0.0305	0.00346	0.001424
3	0.111	0.00310	0.001274
4	0.301	0.00624	0.002568
5	1.14	0.00182	0.000748
6	3.01	0.00066	0.000273

Total yield: 0.0158
Total delayed fraction (β): 0.0065

Pu^{239}

Group	Decay constant λ_i (sec^{-1})	Yield (neutrons per fission)	Fraction β_i
1	0.0128	0.00021	0.000073
2	0.0301	0.00182	0.000626
3	0.124	0.00129	0.000443
4	0.325	0.00199	0.000685
5	1.12	0.00052	0.000181
6	2.69	0.00027	0.000092

Total yield: 0.0061
Total delayed fraction (β): 0.0021

U^{233}

Group	Decay constant λ_i (sec^{-1})	Yield (neutrons per fission)	Fraction β_i
1	0.0126	0.00057	0.000224
2	0.0337	0.00197	0.000777
3	0.139	0.00166	0.000655
4	0.325	0.00184	0.000723
5	1.13	0.00034	0.000133
6	2.50	0.00022	0.000088

Total yield: 0.0066
Total delayed fraction (β): 0.0026

Group	Parameters of Eight-Group model (thermal fission of 235U)		
i	Mean energy (MeV)	Half-life (s)	Rel.Yield
1	0.211	55.6	0.033
2	0.612	24.5	0.154
3	0.269	16.3	0.091
4	0.441	5.21	0.197
5	0.516	2.37	0.331
6	0.512	1.040	0.090
7	0.616	0.424	0.081
8	0.619	0.195	0.023

Table 9-2 Delayed neutron precursors, eight group data

Group i	Decay constant λ_i, (1/s)	Yield fractions, β_i		
		U-235	U-233	Pu-239
1	1.24667E-02	2.145E-04	8.580E-05	6.930E-05
2	2.82917E-02	1.001E-03	4.004E-04	3.234E-04
3	4.25244E-02	5.915E-04	2.366E-04	1.911E-04
4	1.33042E-01	1.281E-03	5.122E-04	4.137E-04
5	2.92467E-01	2.151E-03	8.606E-04	6.951E-04
6	6.66488E-01	5.850E-04	2.340E-04	1.890E-04
7	1.63478E+00	5.265E-04	2.106E-04	1.701E-04
8	3.55460E+00	1.495E-04	5.980E-05	4.830E-05
	sum:	6.500E-03	2.600E-03	2.100E-03

In SPECTRA, the user may define up to 10 DNP groups. The default DNP data is the eight-group data for U-235, obtained from [224]. The values are shown in Table 9-2 (see also Volume 2). Table 9-2 also shows the yield fractions appropriate for U-233 and Pu-239, estimated as described in Volume 3 (test cases DNP-1).

- **Individual fission yields for DNP**

The user may define individual DNP yields for different fissile isotopes, where an individual yield fraction of DNP group *i* from fission of isotope *j*, $\beta_{j\to i}$, is defined in the input deck (BIDNRK - see Volume 2) for each of the fissile isotopes present in the model. In such case the current fission yields are calculated from.

$$\beta_i(t) = \frac{\sum_{j\in fis} \sigma_{f,j}\ N_j(t)\ \beta_{j\to i}}{\sum_{j\in fis} \sigma_{f,j}\ N_j(t)}$$

Here: $N_j(t)$ current concentration ($1/m^3$) of the fissile isotope *j*
$\sigma_{f,j}$ microscopic fission cross section, (m^2), of the fissile isotope *j*
$\beta_{j\to i}$ individual yield fraction of the DNP group *i* from fission of the isotope *j*
$\beta_i(t)$ current yield fraction of the DNP group *i*

The isotope concentrations are discussed in section 9.3. The definition of individual yields and calculation of current yield fraction for different and time-varying concentrations of fissile isotopes are discussed in Volume 3, see tests DNP-1, DNP-2.

The individual yields are very useful in case of Molten Salt Reactor (MSR) long term analyses (e.g. plant lifetime), for example MSRE analyses to calculate noble metals and noble gases behavior, discussed in Volume 3 and publications referenced there.

When using the concept of individual yields, care must be taken when defining the following user-defined reactivity values:

- "Control rod" (or tabular) reactivities (input parameter IREARK, see Volume 2)
- Reactivity feedback tables (input parameter RXTBRK, see Volume 2)

These values were always given in dollars. With the current concept of individual yields, the value of a "dollar" changes in time. It would be most convenient to define the above mentioned reactivities using absolute values (or pcm). However, to preserve compatibility with old input decks, it was decided to keep using dollars for those input parameters. It should be remembered that the definition of dollars is related here to the global DNP yields (input parameters BTDNRK) and not the individual DNP yields (BIDNRK). This is illustrated in Volume 3, test cases DNP-1, DNP-2.

9.2.3 Reactivity Calculation

The reactivity, R, is a function of time, and it consists of three parts:

- "Control rod" (or tabular) reactivities, $R_{i,CR}(t)$, defined using either Tabular or Control Functions. Up to nine independent tabular or control functions may be used. The total "control rod" reactivity is obtained as a sum of the independent reactivities, $R_{CR} = \Sigma R_{i,CR}$.
- Thermal-hydraulic feedback, that means reactivity feedback from fuel temperature, moderator temperature and void fraction, $R_{TF}(T_{fuel}, T_{mod}, \alpha)$.
- Feedback from isotope composition changes, $R_{IC}(N_i)$, which is a function of isotope concentrations, N_i.

The overall reactivity is obtained from:

$$R(t) = \sum_j R_{j,CR}(t) + R_{TF}(T_{fuel}, T_{mod}, \alpha) + R_{IC}(N_i) + R_{bias}$$

where R_{bias} is the initial reactivity bias, calculated by the code in order to obtain the desired initial reactivity (user input). Optionally one can use the moderator density, ρ_{mod}, (kg/m^3), instead of the moderator temperature. This option is foreseen mainly for gas cooled fast reactors (the only moderating material is gas). Each of the three reactivity terms is briefly described below.

- ***"Control rod" reactivity***

  The "control rod" reactivities are defined by Tabular or Control Functions. Simple control rod operations, for example sudden control rod insertion or withdrawal, may be easily modeled by Tabular Functions, specifying reactivity in dollars as a function of time. More complex control systems may be modeled using Control Functions, where the reactivity (in dollars) may be specified as a desired function of any parameters of the analyzed system.

  The use of multiple independent control rod reactivity tables allows easy modeling of elaborate systems, consisting of multiple groups of control rods, with groups controlled by different systems, for example by a reactor power level control system and by an emergency shutdown system. With the Control/Tabular Functions one may model not only the control rod movement, but also other phenomena, for example injection of a control poison, such as boron, into the core, etc.

- ***Thermal-hydraulic feedback***

  The feedback from fuel temperature (Doppler effect and fuel expansion), moderator temperature, and void fraction is calculated by the code with user defined reactivity feedback tables and weighting factors. The three parameters:
  - fuel temperature, T_{fuel}, (K),
  - moderator temperature, T_{mod}, (K), or moderator density, ρ_{mod}, (kg/m^3),
  - void fraction, α,

  are calculated by the code according to the user prescribed weighting factors, W_i:

$$T_{fuel} = \sum_i W_i T_{f,i}$$

$$T_{mod} = \sum_j W_j T_{m,j} \quad or \quad \rho_{mod} = \sum_j W_j \rho_{m,j}$$

$$\alpha = \sum_k W_k \alpha_k$$

where W_i, W_j, W_k are user defined weighting factors. The weighting factors are normalized to 1.0 if all entered values are positive. The local fuel temperatures, $T_{f,i}$, may be taken from SC, TC, or CV packages, as:

- Volume averaged cell temperature of a 1-D Solid Heat Conductor No. *i*.
- Volume averaged cell temperature of a 2-D Solid Heat Conductor No. *i*.
- Pool temperature of a Control Volume No. *i*.
- Atmosphere temperature of a Control Volume No. *i*.

The local moderator temperatures, $T_{m,j}$, or densities, $\rho_{m,j}$, may be taken from SC, TC, or CV packages:

- Volume averaged cell temperature of a 1-D Heat Conductor No. *m*, (available $T_{m,j}$ only).
- Volume averaged cell temperature of a 2-D Heat Conductor No. *m*. , (available $T_{m,j}$ only).
- Pool temperature of a Control Volume No. *m*, (available for $T_{m,j}$ and $\rho_{m,j}$).
- Atmosphere temperature of a Control Volume No. *m*, (available for $T_{m,j}$ and $\rho_{m,j}$).

Note that for the solid materials only the temperature is available for the moderator reactivity feedback calculations. In case of gas cooled thermal reactors the gas moderation is negligible and the reactivity feedback is specified for the solid moderator (e.g. graphite) as a function of temperature. The void reactivity is not applicable for this case and should not be specified. For the fluids (pool, atmosphere) both temperature and density may be used. In case of water-cooled reactors the user may specify alternatively reactivity versus water temperature or versus water density. Additionally the void reactivity feedback may be specified. In case of gas cooled fast reactors, where the only moderating material is the cooling gas, the feedback should be specified versus gas density (temperature would not be adequate in this case since the moderator density is depends not only on temperature but also on pressure). The void reactivity is not applicable for this case and should not be specified.

The local void fractions, α_k, may be taken from SC, TC, or CV packages:

- Local void fraction at the surface of a 1-D Conductor No. *k*, α_{SC}.
- Effective void fraction at the surface of a 1-D Conductor No. *k*, $\alpha_{SC} X_{pool,SC} + X_{atms,SC}$.
- Local void fraction at the surface of a 2-D Conductor No. *k*, α_{TC}.
- Effective void fraction at the surface of a 2-D Conductor No. *k*, $\alpha_{TC} X_{pool,TC} + X_{atms,TC}$.
- Average void fraction in the pool of a Control Volume No. *k*, α_{CV}.
- Effective void fraction in a Control Volume No. *k*, $\alpha_{CV} X_{pool,CV} + X_{atms,CV}$.

The symbols $X_{atms,SC}$, $X_{pool,SC}$ represent the fractions of the SC/TC surface covered by atmosphere and pool respectively, and $X_{atms,CV}$, $X_{pool,CV}$ the fractions of the CV volume occupied by atmosphere and pool respectively.

With the three thermal-hydraulic parameters, T_{fuel}, T_{mod}, α, the reactivity is calculated from a 3-dimensional reactivity feedback table, which defines reactivity versus these three parameters. The reactivity feedback table consists of three coordinate tables:

- fuel temperature table, $T_{f,T}(i)$, $i=1, 2, ..., N_f$,
- moderator temperature or density table, $T_{m,T}(j)$, or $\rho_{m,T}(j)$, $j=1, 2, ..., N_m$,
- void fraction table, $\alpha_T(k)$, $k=1, 2, ..., N_\alpha$,

and the reactivity feedback data table:

- reactivity table, $R_T(i, j, k) = R_T(T_{f,T}(i), T_{m,T}(j), \alpha_T(k))$.

The user may create a reactivity table, or use the built-in data, which contains values appropriate for most common LWR reactor types. The reactivity tables may be created in two ways:

- Independent reactivity table format:
  Three independent reactivity tables are created:

  1.) $R_{f,T}(i) = f (T_{f,T}(i))$,
  2.) $R_{m,T}(j) = g (T_{m,T}(j))$,
  3.) $R_{\alpha,T}(k) = h (\alpha_T(k))$.

  The full reactivity table is then created by the code, as a sum of the individual values:
  $R_T(i,j,k) = R_{f,T}(i) + R_{m,T}(j) + R_{\alpha,T}(k)$.

- Full reactivity table format:
  The full reactivity table is entered: $R(i,j,k)$, where $i = 1, ..., N_T$, $j = 1, ..., N_m$, $k = 1, ..., N_\alpha$.

  When the full reactivity table format is selected, then gaps may be left in the table, which means not all reactivity values have to be entered. The gaps are filled in by the program, using linear interpolation. The interpolation is performed in each of the three "dimensions", between the two closest defined points, bounding the gap from both sides (Figure 9-1). No extrapolation is performed, the defined points must be bounding the point being filled in from both sides. If such points cannot be found in a given direction, then interpolation is skipped in this direction.

  Therefore up to 3 interpolations may be performed to fill a gap. If three or two interpolations are performed, then an arithmetic average is calculated, and assigned to the gap - Figure 9-1 (a) and (b). If one interpolation is performed, the resulting value is assigned to the gap - Figure 9-1 (c). If no interpolation is performed, an error message is printed and the execution is stopped - Figure 9-1 (d). Only the user defined points are used for interpolation. The points, for which values have been obtained previously by interpolations, are not used in subsequent interpolations.

  The method of calculating the average value of independent interpolations is used to fill the gaps rather than a full 3-dimensional interpolation (see description of 3-d interpolation below), because in practice it would be difficult to find 8 defined points bounding the gap from all directions. Thus a full 3-d interpolation scheme would often fail because of lack of data. With the applied scheme the interpolation will fail for the first and the last points: $(i=1, j=1, k=1)$, $(i=N_T, j=N_m, k=N_\alpha)$, so these points must be filled in by the user. But other points, or whole rows of data may be left out and they will be filled in by interpolation.

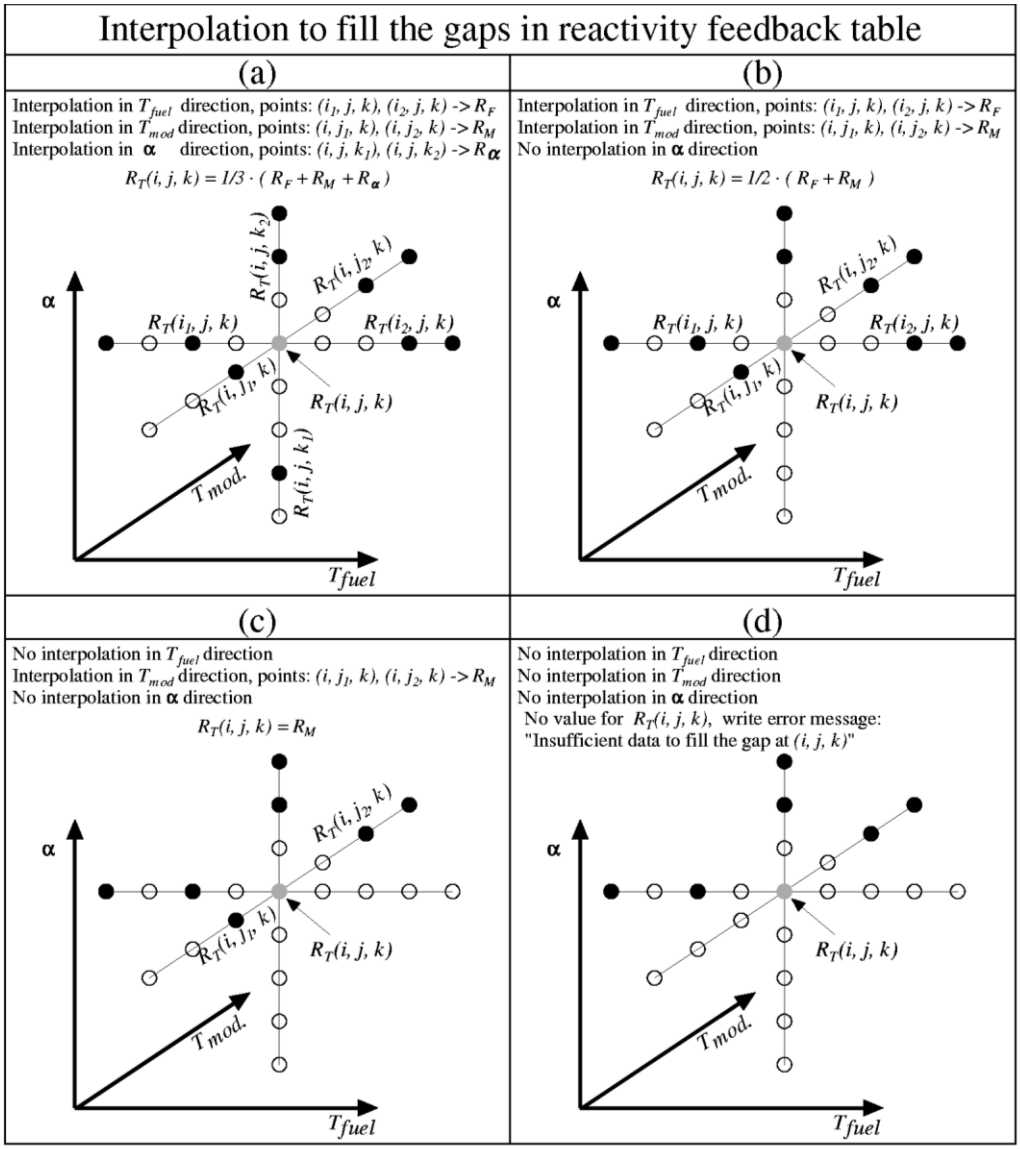


Figure 9-1 Interpolations to fill gaps in the reactivity feedback table.

- Default values of reactivity feedback

  Instead of entering the reactivity feedback table, a simple built-in table might be used. The built-in values are defined through independent reactivity feedback tables. These tables are based on reactivity coefficients shown in [66], table 14-2 (the selected values are those appropriate for light water reactors).

  - Fuel temperature: $\partial\rho/\partial T_{fuel} \approx -3.0 \cdot 10^{-5}$ (K^{-1}), $\partial R/\partial T_{fuel} \approx -5 \cdot 10^{-3}$ (\$/K),
  - Moderator temperature: $\partial\rho/\partial T_{mod} \approx -2.5 \cdot 10^{-4}$ (K^{-1}), $\partial R/\partial T_{mod} \approx -4 \cdot 10^{-2}$ (\$/K),
  - Void fraction: $\partial\rho/\partial\alpha \approx -1.5 \cdot 10^{-3}$ ($\%^{-1}$), $\partial R/\partial\alpha \approx -2 \cdot 10^{-1}$ (\$/%).

  The built-in tables are:

$T_{f,T}(1)$ = 0.0 K	$R_{f,T}(1)$ = 0.0 \$	$T_{f,T}(2)$ = 10000 K	$R_{f,T}(2)$ = –50.0 \$
$T_{m,T}(1)$ = 0.0 K	$R_{m,T}(1)$ = 0.0 \$	$T_{m,T}(2)$ = 10000 K	$R_{m,T}(2)$ =-400.0 \$
$\alpha_T(1)$ = 0.0	$R_{\alpha,T}(1)$ = 0.0 \$	$\alpha_T(2)$ = 1.0	$R_{\alpha,T}(2)$ = –20.0 \$

- Linear interpolation in 3-D space to obtain reactivity

Once the reactivity table is filled, the program can start calculations. During calculations the reactivity is calculated for actual values of the fuel temperature, T_{fuel}, the moderator temperature, T_{mod}, and the void fraction, α. This is done by performing a linear interpolation in a 3-dimensional "space", using eight tabulated points bounding the point: (T_{fuel}, $T_{mod.}$, α) from all sides - Figure 9-2. No extrapolation is performed. The end point values are kept outside the tabulated space.

The interpolations are performed in the following order (see Figure 9-2). First, four interpolations are performed for the fuel temperature direction. Next, two interpolations are performed in the moderator temperature direction. Finally, the last interpolation is performed in the void fraction direction. Since all interpolations are linear, the order of interpolations is meaningless. The same end value would be obtained by, for example, performing first four interpolations in $T_{mod.}$, next two interpolations in α, and finally one interpolation in T_{fuel}.

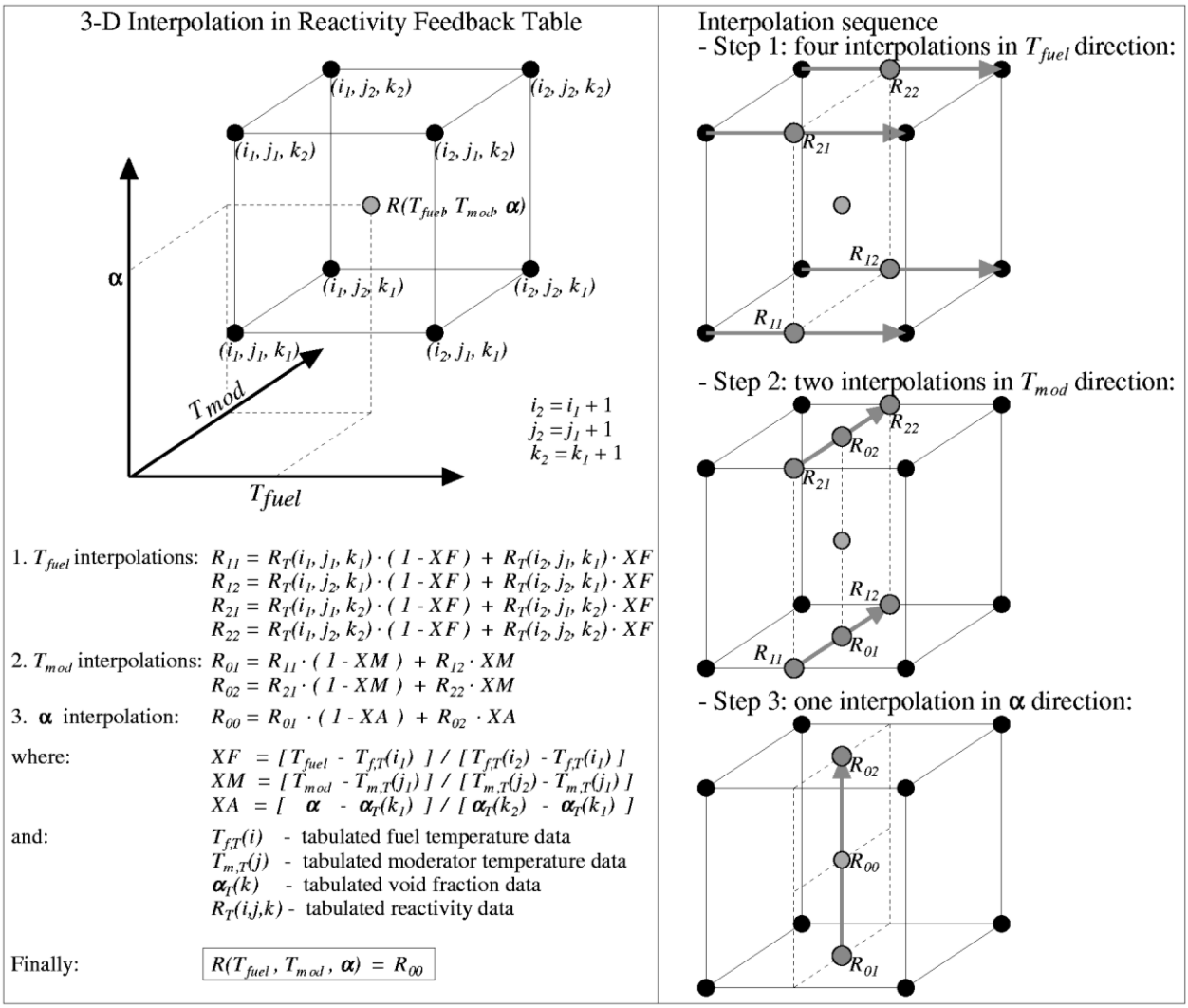


Figure 9-2 Calculating reactivity by 3-D linear interpolation in reactivity feedback table.

- ***Reactivity feedback from core composition changes***

The concentrations of isotopes change during reactor operation due to fuel burn-up, production of poisons, fuel reload, etc. The method of calculating isotope concentrations is described below, in section 9.3. The changes of isotope concentrations affect the reactivity. The reactivity effect is calculated by SPECTRA for all poisons (isotopes for which $\sigma_c > 0.0$), and fuels (isotopes for which $\sigma_f > 0.0$). The calculation is based on the one speed diffusion model ([66], chapter 5).

The relation between the reactivity, ρ, and the multiplication factor, *k*, is (see [66]), equation 15-3):

$$\rho = \frac{k-1}{k}$$

Using the one speed diffusion model, the multiplication factor is given by (see [66], equation 5-222):

$$k = \frac{\nu\Sigma_f}{\Sigma_a} P_{NL} = \frac{\nu\Sigma_f}{\Sigma_a} P_{FNL} P_{TNL}$$

where:
- ν - average number of neutrons generated per fission,
- Σ_f - macroscopic fission cross section, (1/m),
- Σ_a - macroscopic absorption cross section, (1/m),
- P_{NL} - non-leakage probability, (-),
- P_{FNL} - fast neutron non-leakage probability, (-),
- P_{TNL} - thermal neutron non-leakage probability, (-).

We need to calculate the reactivity change resulting from a change in one of the above mentioned parameters. The reactivity change from the initial value of ρ_0, to the final value of ρ, is given by:

$$\rho - \rho_0 = \frac{k-1}{k} - \frac{k_0 - 1}{k_0} = \frac{k - k_0}{k k_0}$$

If the one speed model expression for *k* is substituted into the above formula, we obtain a general formula for the reactivity change:

$$\rho - \rho_0 = \frac{\left(\frac{\nu\Sigma_f}{\Sigma_a} P_{NL}\right) - \left(\frac{\nu\Sigma_f}{\Sigma_a} P_{NL}\right)_0}{\left(\frac{\nu\Sigma_f}{\Sigma_a} P_{NL}\right) \cdot \left(\frac{\nu\Sigma_f}{\Sigma_a} P_{NL}\right)_0}$$

It is assumed that only the macroscopic cross sections, Σ_a, Σ_f, can change. The non-leakage probabilities are assumed to remain constant $P_{NL} = P_{NL,0}$. The reactivity change is calculated below for two cases:

- poison isotopes ($\sigma_f = 0.0$, $\sigma_c > 0.0$),
- fissile isotopes ($\sigma_f > 0.0$).

- Reactivity change due to varying concentrations of poison isotopes ($\sigma_c > 0.0$)

A change of concentration of a poison isotope is considered. This change of course does not affect Σ_f, so:

$$\Sigma_f = \Sigma_{f,0}$$

The absorption cross sections with and without poison are related by:

$$\Sigma_a = \Sigma_{a,0} + \sigma_c N$$

where: σ_c - microscopic cross section for neutron capture of the poison isotope, (m^2),
N - nuclide concentration of the poison isotope, ($1/m^3$).

When the above relations are introduced into the general formula for ρ-ρ_0 (keeping in mind that $P_{NL,0}=P_{NL}$), then one obtains:

$$\Delta\rho = \frac{\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right) - \left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)_0}{\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)\cdot\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)_0} = \frac{1-\frac{\Sigma_a}{\Sigma_{a,0}}}{\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}} = \frac{\Sigma_{a,0}-\Sigma_a}{\nu\Sigma_{f,0}P_{NL}} = -\frac{\sigma_c N}{\nu\Sigma_{f,0}P_{NL}}$$

In SPECTRA the reactivity expressed in dollars, R, is used, where $R=\rho/\beta$. Therefore the final formula, used by SPECTRA to compute the reactivity effect of any isotope i, for which $\sigma_{c,i} > 0.0$, is:

$$R_i = -\frac{C_{R,i}}{\beta}\cdot\frac{1}{\nu P_{FNL}P_{TNL}}\cdot\frac{\sigma_{c,i}}{\Sigma_{f,0}}\cdot N_i$$

SPECTRA calculates also the reactivity change, defined by the following formula:

$$\Delta R_i = -\frac{C_{R,i}}{\beta}\cdot\frac{1}{\nu P_{FNL}P_{TNL}}\cdot\frac{\sigma_{c,i}}{\Sigma_{f,0}}\cdot\left(N_i - N_{i,0}\right)$$

In the above formulae $\sigma_{c,i}$ is the microscopic cross section for neutron capture for isotope i (m^2), N_i is the current concentration of isotope i ($1/m^3$), $N_{i,0}$ is the initial concentration of isotope i ($1/m^3$), $\Sigma_{f,0}$ is the initial macroscopic fission cross section (1/m), and $C_{R,i}$ is a user defined multiplier, which accounts for reactivity worth in case of non-uniform neutron flux distribution. It may be shown using the perturbation theory that the reactivity worth is given by:

$$\Delta\rho \approx -\frac{\sum_V \Sigma_a(\bar{r})\,\Phi^2(\bar{r})\,d^3\bar{r}}{\nu\sum_V \Sigma_f(\bar{r})\,\Phi^2(\bar{r})\,d^3\bar{r}}$$

where $\Phi(r)$ is a local neutron flux at position r. With the point reactor kinetics the neutron flux is uniform, and the reactivity worth is given by:

$$\Delta\rho_{point} = -\frac{\Sigma_a}{\nu\Sigma_f}$$

The factor $C_{R,i}$ maybe therefore defined as the ratio of the true reactivity worth, given by the integral formula, and the space independent value, characteristic for the point kinetics model:

$$C_{R,i} = \frac{\Delta\rho}{\Delta\rho_{point}}$$

Reference [66] shows that for example, in case of an equilibrium poison concentration and a neutron flux distribution appropriate for a slab reactor, the ratio defined above is equal to 4/3.

The following parameters: P_{FNL}, P_{TNL}, ν, $C_{R,i}$, are defined in the input (see Volume 2). The default values are:
- P_{FNL} = 0.97, based on [66], chapter 3,
- P_{TNL} = 0.99, based on [66], chapter 3,
- ν = 2.5, based on [68], sections 1.5.1, 1.5.2.
- $C_{R,i}$ = 1.0 .

- Reactivity change due to varying concentrations of fuel isotopes ($\sigma_f > 0.0$)

A change of concentration of a fissile isotope is considered. In such case both Σ_f and Σ_a will change:

$$\Sigma_f = \Sigma_{f,0} + \sigma_f N$$
$$\Sigma_a = \Sigma_{a,0} + \sigma_f N + \sigma_c N$$

Again, we start with the general formula, and we keep in mind that $P_{NL,0}=P_{NL}$, but this time both fission and capture macroscopic cross sections will change:

$$\Delta\rho = \frac{\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right) - \left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)_0}{\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)\cdot\left(\frac{\nu\Sigma_f}{\Sigma_a}P_{NL}\right)_0} = \frac{1 - \frac{\Sigma_{f,0}}{\Sigma_{a,0}}\frac{\Sigma_a}{\Sigma_f}}{\frac{\nu\Sigma_{f,0}}{\Sigma_{a,0}}P_{NL}} = \frac{\Sigma_{a,0}\Sigma_f - \Sigma_a\Sigma_{f,0}}{\nu\Sigma_{f,0}\Sigma_f P_{NL}}$$

Now we substitute the relations for macroscopic cross sections:

$$\Delta\rho = \frac{(\Sigma_a - N\sigma_f - N\sigma_c)\Sigma_f - \Sigma_a(\Sigma_f - N\sigma_f)}{\nu\Sigma_{f,0}\Sigma_f P_{NL}}$$

Next, some rearrangements are performed to simplify the formula:

$$\Delta\rho = \frac{\Sigma_a N\sigma_f - \Sigma_f N\sigma_f - \Sigma_f N\sigma_c}{\nu\Sigma_{f,0}\Sigma_f P_{NL}}$$

Now a use is made of the multiplication factor definition in the one speed diffusion model ([66], chapter 5): $k = \nu\Sigma_f P_{NL}/\Sigma_a$, and the fact that in a critical reactor k=1.0. This leads to: $\Sigma_a = \nu\Sigma_f P_{NL}$, and:

$$\Delta\rho = \frac{\nu\Sigma_f P_{NL} N\sigma_f - \Sigma_f N\sigma_f - \Sigma_f N\sigma_c}{\nu\Sigma_{f,0}\Sigma_f P_{NL}} =$$

$$= \frac{(\nu P_{NL} - 1)N\sigma_f - N\sigma_c}{\nu\Sigma_{f,0} P_{NL}} = \left(1 - \frac{1}{\nu P_{NL}}\right)\frac{N\sigma_f}{\Sigma_{f,0}} - \frac{1}{\nu P_{NL}}\frac{N\sigma_c}{\Sigma_{f,0}}$$

The formula used in SPECTRA to compute reactivity effect of any isotope i, for which $\sigma_{f,i} >$ 0.0, is obtained again by expressing reactivity in dollars, and using the multiplier, $C_{R,i}$:

$$R_i = \frac{C_{R,i}}{\beta}\left[\left(1 - \frac{1}{\nu P_{FNL} P_{TNL}}\right)\cdot\frac{\sigma_{f,i}}{\Sigma_{f,0}} - \frac{1}{\nu P_{FNL} P_{TNL}}\cdot\frac{\sigma_{c,i}}{\Sigma_{f,0}}\right]\cdot N_i$$

Similarly as in case of poisons, the reactivity change is given by:

$$\Delta R_i = \frac{C_{R,i}}{\beta}\left[\left(1 - \frac{1}{\nu P_{FNL} P_{TNL}}\right)\cdot\frac{\sigma_{f,i}}{\Sigma_{f,0}} - \frac{1}{\nu P_{FNL} P_{TNL}}\cdot\frac{\sigma_{c,i}}{\Sigma_{f,0}}\right]\cdot(N_i - N_{i,0})$$

where N_i and $N_{i,0}$ are the current and the initial concentrations of fissile isotope i, while $\Sigma_{f,0}$ is the initial macroscopic fission cross section. Note that the second term (the term with σ_c) is exactly the same as in case of poison isotopes.

The SPECTRA code contains an option to include the resonance escape probability, p, and the fast fission factor, ε, in the calculation of reactivity effects. If this option is used, then the reactivity effect is inversely proportional to $p\varepsilon$ (see [92], equation 7.89), that means the ratio: $1/(\nu \cdot P_{FNL} \cdot P_{TNL})$ is replaced everywhere by: $1/(\nu \cdot p \cdot \varepsilon \cdot P_{FNL} \cdot P_{TNL})$.

In the above formulae only the initial fission cross section, $\Sigma_{f,0}$, is used, while the current value, Σ_f, is not needed. The current value of Σ_f is however needed for the reactor power calculation (see the description of the point kinetics equations above). The Reactor Kinetics Package is updating the macroscopic fission cross section, Σ_f, at every time step, based on the current values of fuel concentrations, N_i:

$$\Sigma_f = \sum_i \sigma_{f,i} N_i$$

9.3 Isotope Transformation Model

The Isotope Transformation Model is available in SPECTRA as a part of the Reactor Kinetics Package. It allows to calculate the core composition changes, caused by fuel burn-up, production of poisons (such as Xe-135, etc.), decay of isotopes, fuel reload (adding or removing fuel during the simulated transient), etc. The changes in isotope concentrations have an effect on the reactivity and the decay power production. The method of calculating reactivity effects is described above, in section 9.2.3. Effects related to decay heat production are described in this section.

9.3.1 Description of the Isotope Transformation Model

The reaction rate equations describing the concentrations of nuclei in the reactor core can be derived using simple balance ideas. Let N_i be a concentration ($1/m^3$) of nuclides of isotope *i*. In a more rigorous treatment N_i would be a function of position, $N_i(\mathbf{r})$. Since the isotope transformation model in SPECTRA works with the point reactor kinetics model, space dependencies cannot be taken into account. Therefore N_i is considered to be an average value for the whole core.

In general, a change of nuclide concentration of isotope *i* may be caused by the following eight reasons:

1.) Removal due to decay of isotope *i*. The rate of nuclide removal is equal to:

$$-\lambda_i N_i$$

where λ_i is a decay constant (1/s) of isotope *i*.

2.) Production due to decay of some other isotope *j*. The rate of change of *i* is equal to:

$$\sum_{j \neq i} \lambda_i N_i \gamma_{d,j \to i}$$

where $\gamma_{d,j \to i}$ is a yield fraction of isotope *i* from the decay of isotope *j*. In other words, $\gamma_{d,j \to i}$ is a probability of the fact that a decay of a nuclide of isotope *j* will result in production of a nuclide of isotope *i*.

3.) Removal due to neutron capture in isotope *i*. The rate of nuclide removal is equal to:

$$-\sigma_{c,i} N_i \Phi$$

where $\sigma_{c,i}$ is a microscopic cross section for neutron capture, (m^2), for isotope *i*, and Φ is the neutron flux ($1/m^2/s$). The neutron flux is equal to $n \cdot v_n$, with v_n being thermal neutron velocity (m/s), and *n* being the neutron density ($1/m^3$), calculated by the reactor kinetics model, as shown in section 9.2.

4.) Production due to neutron capture by some other isotope *j*. The rate of change of *i* is:

$$\sum_{j \neq i} \sigma_{c,j} \, N_j \, \Phi \, \gamma_{c,j \to i}$$

where $\gamma_{c,\, j \to i}$ is a yield fraction of isotope *i* from the neutron capture by isotope *j*. In other words, $\gamma_{c,\, j \to i}$ is a probability of the fact that a neutron capture by a nuclide of isotope *j* will result in production of a nuclide of isotope *i*.

5.) Removal due to fission of nuclide of isotope *i*. The rate of nuclide removal is equal to:

$$-\sigma_{f,i} N_i \Phi$$

where $\sigma_{f,\, i}$ is a microscopic cross section for fission, (m^2), for isotope *i*.

6.) Production due to fission of some other isotope *j*. The rate of change of *i* is equal to:

$$\sum_{j \neq i} \sigma_{f,j} \, N_j \, \Phi \, \gamma_{f,j \to i}$$

where $\gamma_{f,\, j \to i}$ is a yield fraction of isotope *i* from fission of isotope *j*. In other words, $\gamma_{f,j \to i}$ is a probability of the fact that nuclide of isotope *i* will be created from fission of a nuclide of isotope *j*.

7.) Removal due to removing of fuel elements from the core. The removal rate is:

$$-N_i C_i R^R$$

where R^R (1/s) is the relative rate of removal of fuel elements from the reactor core: $R^R = -(1/N_{fe}) \cdot (dN_{fe}/dt)$. The symbol N_{fe} is used here to denote the number of fuel elements in the core. R^R must be positive (or zero). C_i is a user-defined multiplier. It may be used to simulate removal of fuel with specified burn-up. For example, suppose the average enrichment in the core is 5%, and the fuel elements with an enrichment of 2% need to be removed from the core. The multiplier of $C_{U\text{-}235} = 2/5 = 0.4$ would then need to be imposed on U-235, while other isotopes would be removed according to their current concentrations $C_i = 1.0$.

8.) Increase due to inserting of fuel elements into the reactor core. The rate of change is:

$$\sum_k N_i^F(k) \, R^F(k)$$

where $R^F(k)$ (1/s) is the relative rate of inserting of fuel elements into the reactor core: $R^F(k) = (1/N_{fe}) \cdot (dN_{fe}(k)/dt)$. The symbol N_{fe} is used here to denote the number of fuel elements in the core. $N_i^F(k)$ is the concentration of isotope *i* in the "fresh" fuel elements, being loaded into the core. The symbol *k* is the loading function count. Several loading functions may be specified, each may have a different isotope composition $N_i^F(k)$. $R^F(k)$ may be both positive and negative. With negative values of $R^F(k)$ user may remove particular isotopes from the core, independently of the current core composition. This is in contrast to the removal function, R^R, which removes isotopes proportionally to the current core concentration.

Taking all the removal and production terms described above, a general balance of nuclide concentration of isotope *i*, is written as:

$$\begin{aligned}\frac{dN_i}{dt} = &-\lambda_i N_i + \sum_{j\neq i}\lambda_j N_j \gamma_{d,j\to i} - \\ &-\sigma_{c,i} N_i \Phi + \sum_{j\neq i}\sigma_{c,j} N_j \Phi \gamma_{c,j\to i} - \\ &-\sigma_{f,i} N_i \Phi + \sum_{j\neq i}\sigma_{f,j} N_j \Phi \gamma_{f,j\to i} - \\ &-N_i C_i R^R + \sum_k N_i^F(k) R^F(k)\end{aligned}$$

This equation can be written in a finite difference form, by replacing dN_i/dt by $(N_i - \overline{N_i})/\Delta t$, where $\overline{N_i}$ is the concentration of isotope *i* at the start of the time step. The finite difference form of the isotope balance equation is:

$$\begin{aligned}&N_i \frac{1}{\Delta t} + \lambda_i N_i - \sum_{j\neq i}\lambda_j N_j \gamma_{d,j\to i} + \\ &+\sigma_{c,i} N_i \Phi - \sum_{j\neq i}\sigma_{c,j} N_j \Phi \gamma_{c,j\to i} + \\ &+\sigma_{f,i} N_i \Phi - \sum_{j\neq i}\sigma_{f,j} N_j \Phi \gamma_{f,j\to i} + \\ &+N_i C_i R^R = \sum_k N_i^F(k) R^F(k) + \overline{N_i}\frac{1}{\Delta t}\end{aligned}$$

where the terms with actual (unknown) nuclide concentrations have been grouped on the left hand side, while other (known) terms were grouped at the right hand side of the equation. The above equation, written for each isotope, forms a set of linear equations that can be written shortly in a matrix form:

$$A \cdot N = B$$

where ***N*** is a vector of unknown nuclide concentrations, N_i, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. The elements of matrix ***A*** are equal to:

$$a_{i,j} = \begin{cases} 1/\Delta t + \lambda_i + \sigma_{c,i}\Phi + \sigma_{f,i}\Phi + C_i R^R & \text{if } i = j \\ -\lambda_j N_j \gamma_{d,j\to i} - \sigma_{c,j}\Phi\gamma_{c,j\to i} - \sigma_{f,j}\Phi\gamma_{f,j\to i} & \text{if } i \neq j \end{cases}$$

The elements of vector ***B*** are given by:

$$b_i = \overline{N_i}/\Delta t + \sum_k N_i^F(k)\, R^F(k)$$

Note that $N_i^F(k)$ are known quantities, since those are user defined concentrations of isotopes in fuel being loaded into the core during transient (see Volume 2).

This above matrix equation is solved by the isotope transformation model using one of the standard matrix solvers (see section 17.4). The current version of the program can accommodate up to 200 different isotopes.

In cases when an equilibrium initial core composition is requested (see Volume 2), then a steady-state version of the matrix equation is solved to determine the initial concentrations of isotopes. In the steady state version the terms with $1/\Delta t$ are set to zero: $1/\Delta t \rightarrow 0.0$. Isotopes for which there is no removal mechanism (λ_i=0, $\sigma_{c,i}$=0, $\sigma_{f,i}$=0) are excluded from the matrix (infinite equilibrium concentration). Such isotopes are assigned zero initial concentrations, unless the user specifies a different concentration in the input (see Volume 2).

All isotopes with a positive decay constant, $\lambda_i > 0.0$, contribute to the reactor decay heat. The decay heat is calculated from the following formula:

$$Q_d = V_R \sum_i \lambda_i N_i q_{d,i}$$

where V_R is the reactor volume (m^3), and $q_{c,i}$ is the heat per decay of isotope i, (W/decay). The total reactor power, Q_r, calculated by the reactor kinetics package, consists of the fission power, Q_f, and the decay power, Q_d, and is therefore given by the following formula:

$$Q_r = Q_f + Q_d = V_R \Sigma_f \Phi \, q_f + V_R \sum_i \lambda_i N_i q_{d,i}$$

where Σ_f is the macroscopic fission cross section (1/m), and q_f is the heat released per fission, (W/fission).

The parameters needed by the isotope transformation model, such as decay constants, heat released per decay, cross sections, rates of removal and filling of fuel elements, are user defined. For convenience SPECTRA contains a built-in library of most important isotopes. The isotopes present in the library are shortly described below.

9.3.2 Description of the Built-in Isotope Library

The built-in data may be requested by the user through an input parameter (see Volume 2). The average, one-group thermal cross sections are used, appropriate for the thermal reactor types. The following data is built-into the code:

- Isotope chain for ^{233}U / ^{235}U fuel (8 isotopes).
- Isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel (8 isotopes).
- Isotope chain for ^{135}Xe poison (3 isotopes).
- Isotope chain for ^{149}Sm poison (3 isotopes).
- Isotope chain for ^{157}Gd poison (2 isotopes).
- 11 groups of main isotopes responsible for the decay heat generation.

These isotope chains are described subsequently below.

- ***Isotope chain for ^{233}U / ^{235}U fuel (isotopes 101 through 108)***

The isotope chain for ^{233}U / ^{235}U fuel is shown in Figure 9-3 (reproduced from [67]). The chain implemented in the SPECTRA built-in isotope library is somewhat simplified - the isotopes behind ^{235}U are not taken into account. The isotope chain for ^{233}U / ^{235}U fuel, as implemented in SPECTRA, is shown in Figure 9-4. The decay constants were taken from [67]. The decay constant for ^{234}Pa was taken from reference [69]. Decay heats were taken from [70]. One-group cross-sections are used; the fission and the capture cross-sections were obtained from [71].

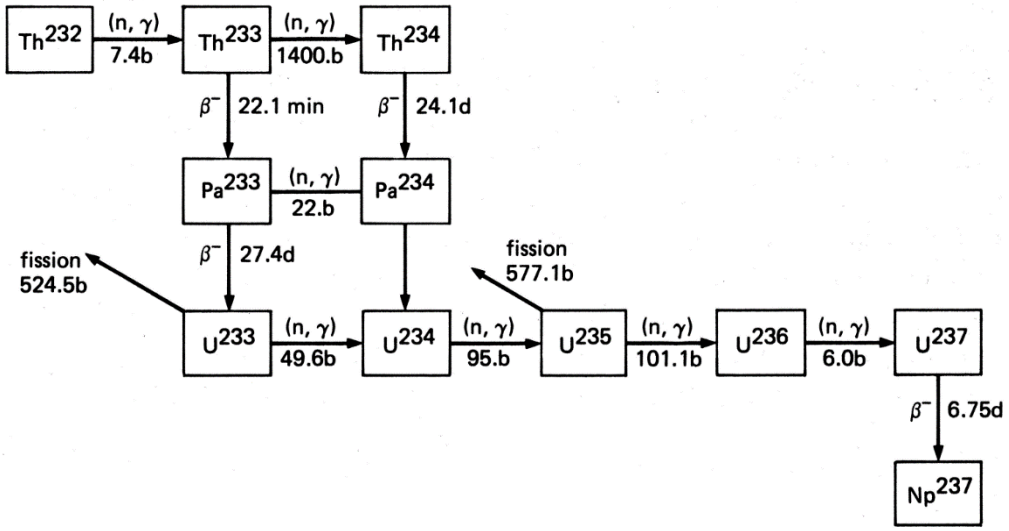


Figure 9-3 The isotope chain for ^{233}U / ^{235}U fuel, ([67], figure 6.2).

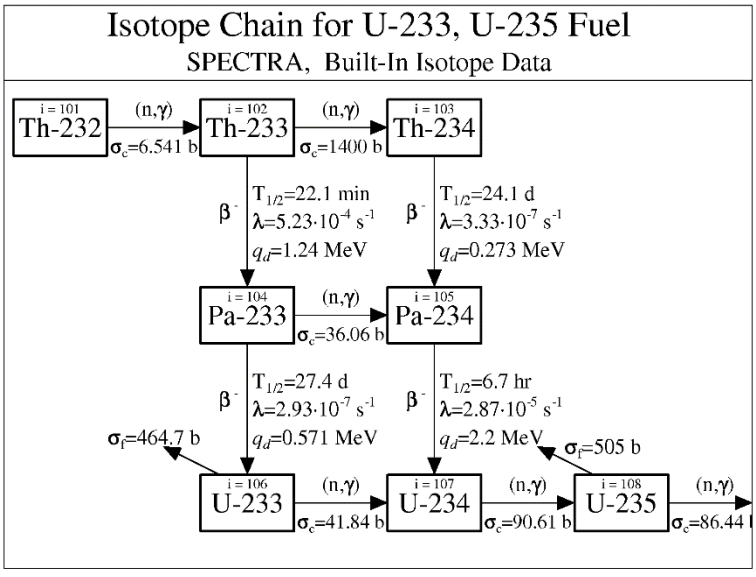


Figure 9-4 The isotope chain for ^{233}U / ^{235}U fuel, as implemented in SPECTRA.

- ***Isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel (isotopes 201 through 208)***

The isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel is shown in Figure 9-5 (reproduced from [67]). The chain implemented in the SPECTRA built-in isotope library is somewhat simplified - the isotopes beyond ^{241}Pu are not taken into account. The isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel, as implemented in SPECTRA, is shown in Figure 9-6. Decay data were taken from [67] and [70]. One-group cross-sections are used; the fission and the capture cross-sections were obtained from [71].

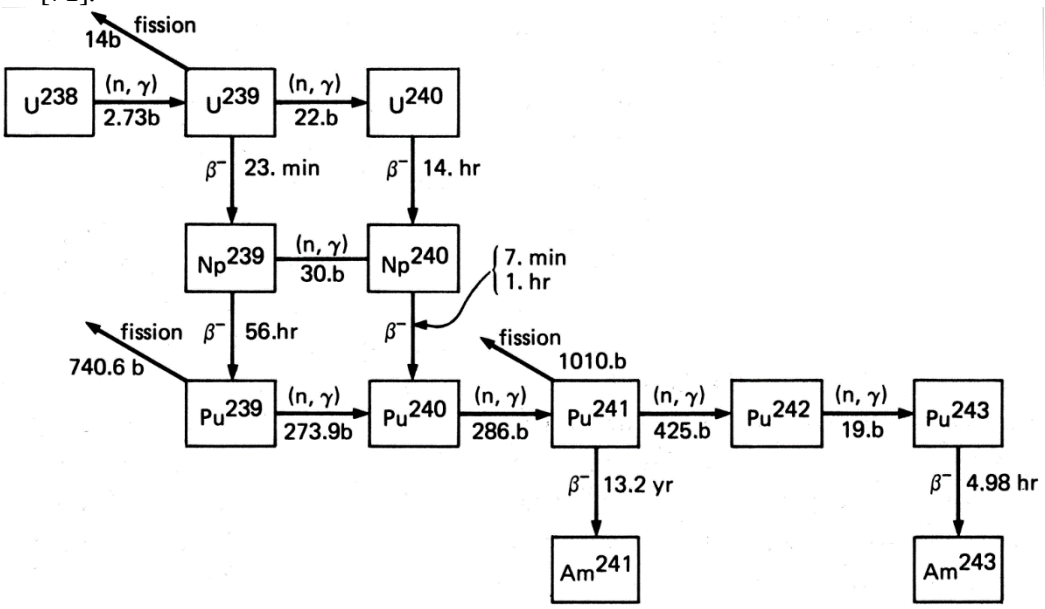


Figure 9-5 The isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel, ([67], figure 6.1).

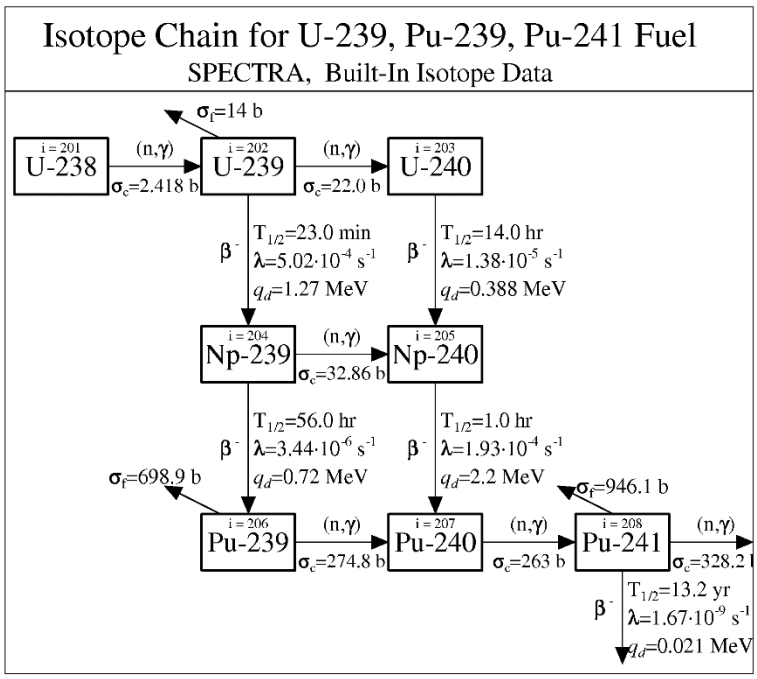


Figure 9-6 The isotope chain for ^{239}U / ^{239}Pu / ^{241}Pu fuel, as implemented in SPECTRA.

Correction for resonance capture by ^{232}Th and ^{238}U

In case of ^{232}Th and ^{238}U there is a significant neutron capture in resonances. To take that into account, the neutron capture cross sections for those two nuclides are modified. The "effective" cross sections are calculated from (compare [92], equation 7.119):

$$\sigma_{c,i} = \sigma_{c,i}^{true}\left(1+(1-p_i)\,\varepsilon\,\nu\,P_{FNL}\,\frac{\sum_j \sigma_{f,j}N_j}{\sigma_{c,i}^{true}N_i}\right)$$

where:
- $\sigma_{c,i}$ - effective cross section for isotope i, i=1 (^{232}Th) or =9 (^{238}U),
- p_i - probability of escaping resonance absorption in isotope i,
- ε - fast fission factor,
- ν - number of neutrons generated per fission,
- P_{FNL} - fast non-leakage probability,
- $\sigma_{f,j}$ - fission cross section for isotope j, (barn),
- N_i,N_j - concentrations of isotopes i, j, (1/m^3).

In the above equation the true capture cross section is given with the superscript "*true*", and the sum is taken over all fissile isotopes. The resonance escape probability is given by ([92], equation 6.119):

$$p_i = \exp(-c\,I_i\,N_i)$$

where I_i is the resonance integral for isotope i and c is a certain constant, the value of which varies for different reactors. The resonance integral for cylindrical fuel rods is given by ([92], equation 6.120): $I_i = A_i + C_i/(r\rho)^{1/2}$, where r is rod diameter (cm), ρ is fuel density (kg/cm^3), and A_i, C_i are constants, equal to ([92], table 6.5):

- i = ^{238}U (metal) A_i = 2.8 C_i = 38.3,
- i = $^{238}UO_2$ A_i = 3.0 C_i = 39.6,
- i = ^{232}Th (metal) A_i = 3.9 C_i = 20.9,
- i = $^{232}ThO_2$ A_i = 3.4 C_i = 24.5.

It is assumed that resonance absorption occurs only in ^{232}Th and ^{238}U. Therefore the individual resonance escape probabilities, p_i, are related to the overall resonance escape probability, p, by: $p_1 \cdot p_9 = p$. The value of p is a user input (see Volume 2). After easy transformations one obtains the following expressions for p_1, p_9:

$$p_1 = \exp\left(\ln(p)\frac{I_1N_1}{I_1N_1+I_9N_9}\right) \qquad p_9 = \exp\left(\ln(p)\frac{I_9N_9}{I_1N_1+I_9N_9}\right)$$

The following values are assumed for calculations: r=1.0, ρ=19.1, A_i, C_i for dioxides. These gives I_1 = 9.0, I_9 = 12.1. The effective cross sections are calculated based on initial concentrations, and are not modified during the transient.

It must be remembered that the effective cross sections are calculated as described above only if the built-in library is being used, and if the capture cross sections for ^{232}Th, ^{238}U are not modified

in the input data. Thus, the user can calculate his own resonance integrals, and enter the effective capture cross sections in the input data, overwriting the default values.

- ***Isotope chain for ^{135}Xe poison (isotopes 301 through 303)***

The isotope chain for ^{135}Xe fission product poison is shown in Figure 9-8. A simplified chain is used, and the short life isotopes, ^{135}Sb, ^{135}Te, are neglected. The yield fraction of ^{135}I is taken as a sum of the yield fractions for ^{135}Sb, ^{135}Te, and ^{135}I. For comparison, the full ^{135}Xe chain is shown in Figure 9-7. The decay constants were taken from reference [66] (figure 15-1). The decay heats were taken from [70]. The fission product yields were taken from reference [71] (table 8). One-group cross-sections are used; the fission and the capture cross-sections were obtained from [71].

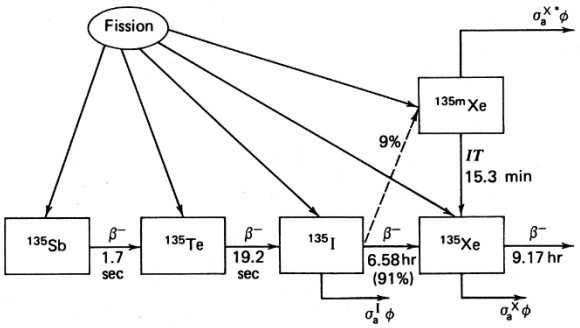


Figure 9-7 A full decay scheme for ^{135}Xe, [66], figure 15-1.

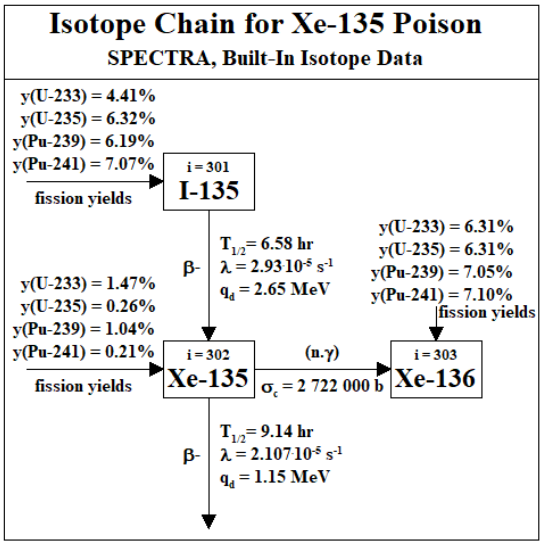


Figure 9-8 The isotope chain for ^{135}Xe, as implemented in SPECTRA.

- ***Isotope chain for ^{149}Sm poison (isotopes 401 through 403)***

The isotope chain for ^{149}Sm fission product poison is shown in Figure 9-9. The decay constants were taken from reference [67] (figure 6.3). The decay heats were taken from [70]. The fission product yields were taken from reference [71] (table 8). One-group cross-sections are used; the fission and the capture cross-sections were obtained from [71].

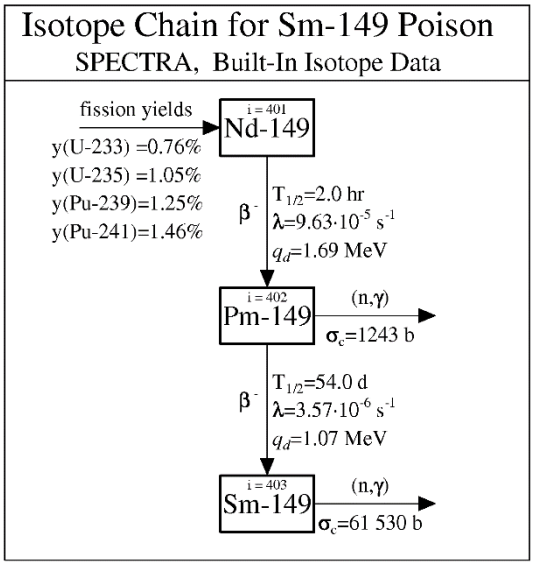


Figure 9-9 The isotope chain for ^{149}Sm, as implemented in SPECTRA.

- ***Isotope chain for ^{157}Gd poison (isotopes 501 through 502)***

The isotope chain for ^{157}Gd fission product poison is shown in Figure 9-10. The decay constants were taken from reference [71] (table 6). The decay heats were taken from [70]. The fission product yields were taken from reference [71] (table 8). One-group cross-sections are used; the fission and the capture cross-sections were obtained from [71].

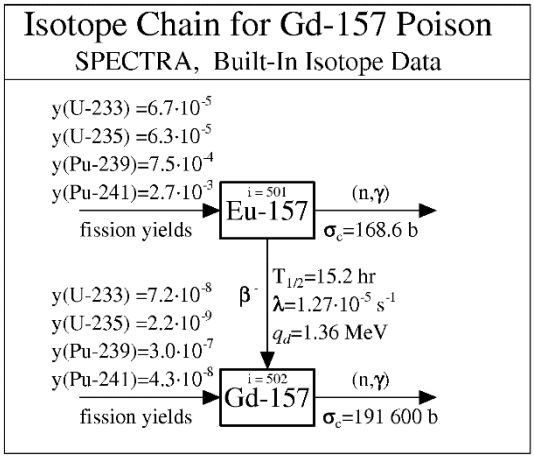


Figure 9-10 The isotope chain for ^{157}Gd, as implemented in SPECTRA.

- ***Isotope groups of main decay heat producers (isotopes 601 through 611)***

The decay heat producers are represented by 11 groups of "isotopes", similarly as in the RELAP code [72]. The decay constants and fission yields of these groups were established based on data from [72] and comparisons with ANS standard [73], as described below.

For all decay heat groups the values of the decay constant, λ_i, and the decay heat, $q_{d,i}$ are positive. The neutron absorption is neglected, thus the neutron capture cross section, $\sigma_{c,i}$, as well as fission cross section, $\sigma_{f,i}$, are equal to zero. Therefore, if the removal, R^R or refilling functions, $R^F(k)$, are not used, then the general isotope balance equation for decay heat groups reduces to:

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + \sum_{j \neq i} \sigma_{f,j} N_j \Phi \gamma_{f,j \to i}$$

where:
- N_i - concentration of nuclides from group i, ($1/m^3$),
- λ_i - decay constant of group i, (1/s),
- $\gamma_{d,j \to i}$ - decay yield fractions (probability that a decay of nuclide from group j will result in creation of nuclide in group i),
- $y_{f,j \to i}$ - fission yield fractions (probability that a fission of nuclide from a fissile group j will result in creation of nuclide in group i),
- Φ - neutron flux, ($1/m^2/s$).

The value of decay heat of each group is calculated as:

$$Q_{d,i} = \lambda N_i q_{d,i}$$

where:
- $Q_{d,i}$ - decay heat density, group i, (W/m^3),
- $q_{d,i}$ - heat release per decay, group i, (W/decay).

The model parameters include the decay constants, λ_i, the fission yield fractions, $y_{f,k \to i}$, the decay yield fractions, $y_{d,j \to i}$, as well as the energy per decay, $q_{d,i}$. The decay constants for all decay groups were taken from RELAP [72]. The values of the decay yield fractions, $y_{d,j \to i}$, as well as the energy per decay, $q_{d,i}$, could not be directly taken from RELAP, since RELAP works with energy fractions rather than the yield fractions and the energy per decay. The values applied in SPECTRA were calculated in order to obtain values, which provide exactly the same final result as the RELAP model. The transformation of RELAP data into SPECTRA model leaves some degree of freedom. Since one has to fit the product of the yield fractions and the energy per decay, the individual values may be arbitrary, as long as the product is in agreement with the value from RELAP. The values of the decay constants, λ_i, the fission yield fractions, $y_{f,k \to i}$, the decay yield fractions, $y_{d,j \to i}$, as well as the energy per decay, $q_{d,i}$, are shown in Table 9-3 and Figure 9-11.

Table 9-3 Decay heat group constants.

Group No.	Isotope i	Decay constant, λ_i, (1/s)	Decay yields $y_{d,\ i\rightarrow i+1}$, (-)	U fission yields $y_{f,\ U\rightarrow i}$, (-)	Pu fission yields $y_{f,\ Pu\rightarrow i}$, (-)
1	601	1.77	1.000	0.2392	0.1600
2	602	0.577	1.000	0.4208	0.3000
3	603	6.74×10^{-2}	1.000	0.5800	0.5800
4	604	6.21×10^{-3}	0.602	0.3080	0.3080
5	605	4.74×10^{-4}	0.554	0.0000	0.0000
6	606	4.81×10^{-5}	0.358	0.0000	0.0000
7	607	5.34×10^{-6}	0.710	0.0000	0.0000
8	608	5.73×10^{-7}	0.700	0.0000	0.0000
9	609	1.04×10^{-7}	0.506	0.0000	0.0000
10	610	2.96×10^{-8}	1.000	0.0000	0.0000
11	611	7.59×10^{-10}		0.0000	0.0000

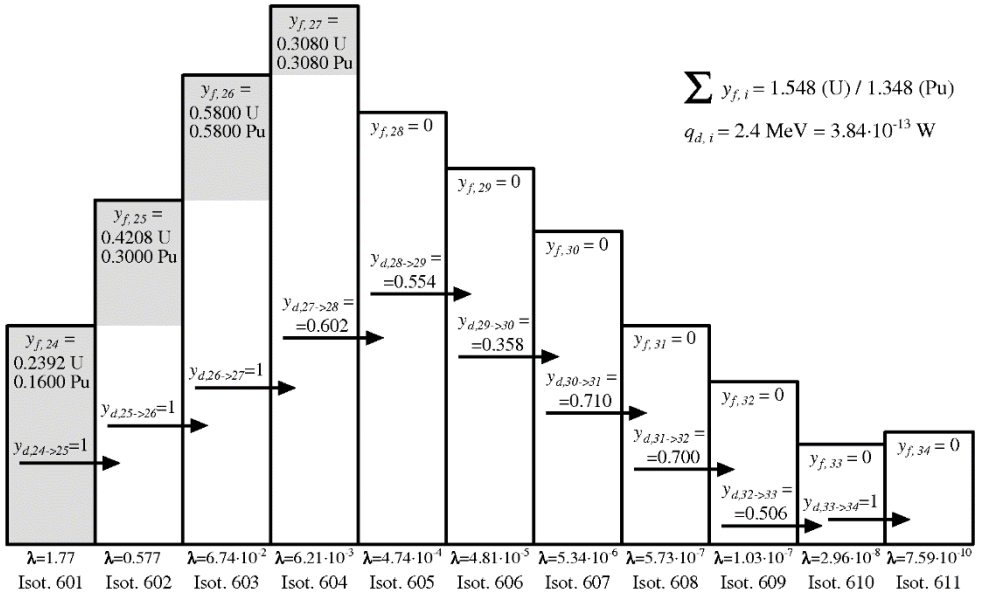


Figure 9-11 Decay heat group constants.

Comparison of the decay heat calculated by SPECTRA, with the decay heat curves obtained from the ANS standard [73], is shown in Figure 9-12 and Figure 9-13. The long-term behavior is very similar. Initially the SPECTRA calculated values are somewhat higher. This discrepancy is caused by the fission power, which did not decrease instantaneously to zero. In the model the reactor was shutdown by inserting a large negative reactivity. The fission power was produced for some time after the scram due to the delayed neutron precursors.

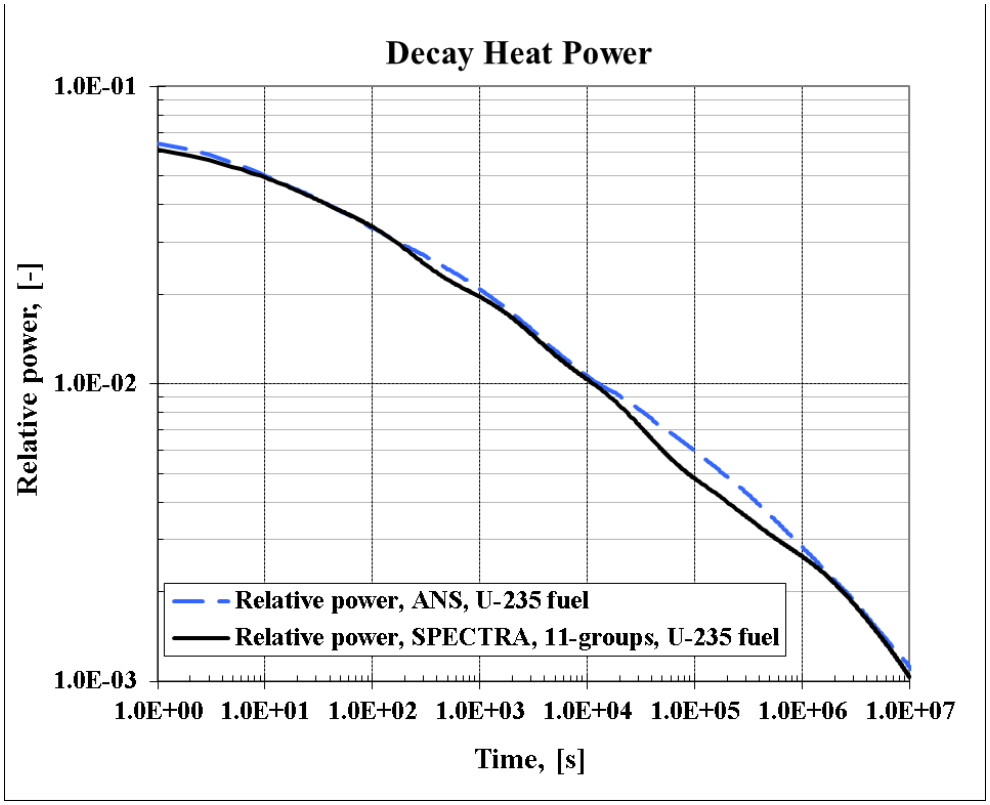


Figure 9-12 Comparison of decay heat for ^{235}U fuel, ANS and SPECTRA 11 groups.

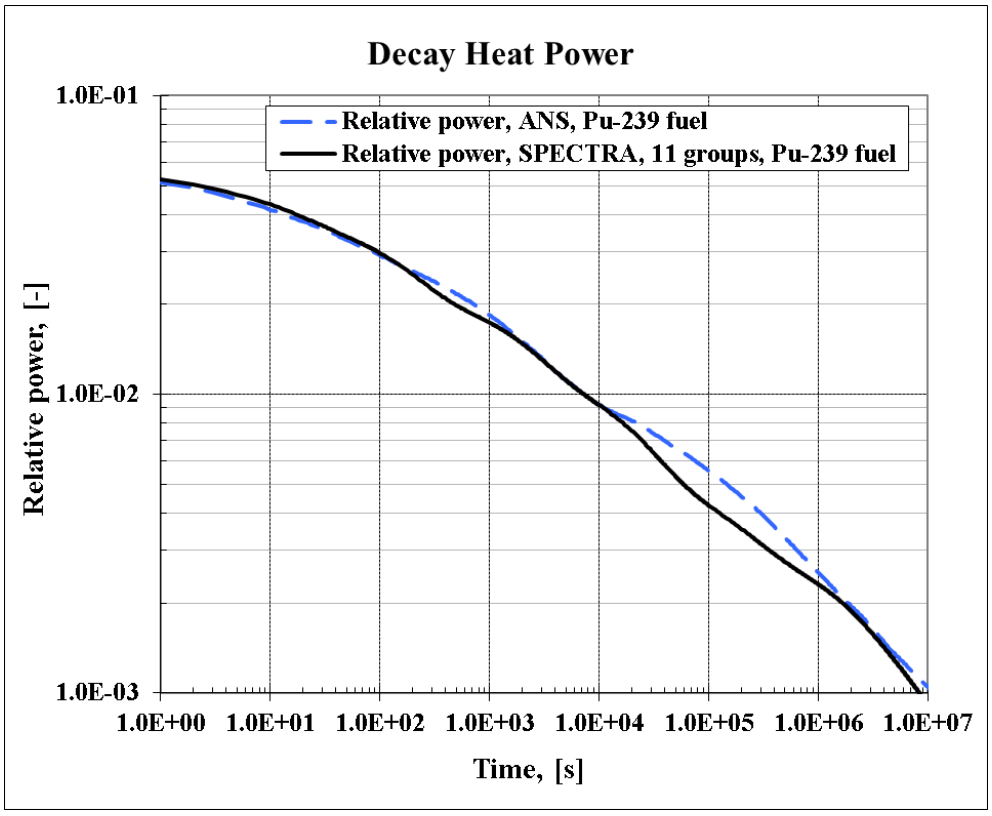


Figure 9-13 Comparison of decay heat for ^{239}Pu fuel, ANS and SPECTRA 11 groups.

9.4 Point Kinetics Model for Circulating Fuel

9.4.1 Introduction

The point kinetics model for circulating fuel is intended for applications such as molten salt reactors (MSR). In MSRs the fissile fuel is dissolved in a fluid (molten salt) which circulates in the primary reactor system. The point kinetics model for circulating fuel is invoked in the input deck by setting the fuel type, IFTORK, to 2 (Volume 2).

The main difference between the circulating fuel and the "normal", solid fuel point kinetics is the fact that the isotopes such as fission products, including the delayed neutron precursors, are circulating together with the carrier fluid The point kinetics model for circulating fuel is built using two packages:

1. The RK package, which calculates the reactor power using the point kinetics model. The main purpose of the RK Package is to compute the reactor power.
2. The RT package calculates transport of fission products in atmosphere and pool, including evaporation, condensation, sorption of vapors on metallic surfaces, etc. The main purpose of the RT package is to track the behavior of fission products released from the core.

In case of circulating fuel, the calculation procedure is as follows:

- The isotopes from the RK Package are "mapped" to the RT Package. This includes:
  - delayed neutron precursors,
  - isotopes such as poisons (Xe-135), main decay heat generators.

  The purpose is to create a set of isotopes in the RT Package that have identical properties as the isotopes that are used in the RK Package.

- The RK Package equations are modified, to calculate sources of isotopes (as described in detail in section 12.3). These sources are defined for the pool of core Control Volumes. Core Volumes are defined by positive fuel weighting factors. The source strength is assumed to be proportional to the weighting factor.

- The isotope concentrations in Control Volumes are calculated using the standard models (inter-volume transport with flowing fluid, filters, etc.)

- The masses of isotopes in the core volumes (positive fuel weighting factors) are passed back to the RK Package and are used to calculate the fission power as well as decay power, effects of poisons, etc.

In short, the existing RT Package in SPECTRA is used as a convenient "carrier" of isotopes in the fluid. Because of this fact a relatively small modification was sufficient to adapt the point kinetics model for the circulating fuel. This modification is described in detail in the following sections.

9.4.2 Point Kinetics Model Equations - Solid Fuel

The standard point kinetics equations, applied for solid fuel are (section 9.2.1):

$$\frac{dn(t)}{dt} = \frac{\beta(R(t)-1)}{\Lambda} n(t) + \sum_i \lambda_i C_i(t) + S_{V,ext}(t)$$

$$\frac{dC_i(t)}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t)$$

n neutron concentration, (1/m^3),
C_i concentration of delayed neutron precursor group i, (1/m^3),
R reactivity, ($),
Λ prompt neutron generation time, (s),
λ_i decay constant of delayed neutron precursor group i, (1/s),
β_i yield fraction of delayed neutron precursor group i, (-),
β sum of delayed neutron fractions, $\beta = \Sigma \beta_i$, (-),
$S_{V,ext}$ external neutron source per unit volume per unit time, (1/m^3/s),
t time, (s).

The reactor fission power, Q_f (W), is calculated based on the neutron concentration, n, from:

$$Q_f(t) = n(t) \cdot V_R \cdot \Sigma_f \cdot v_n \cdot q_f$$

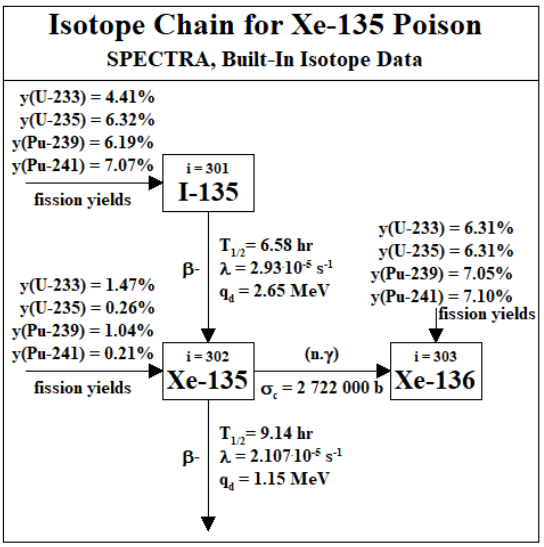
where V_R is the reactor volume (m^3) (user input), Σ_f is the macroscopic fission cross section (1/m), q_f is the energy generated per fission (W/fission), v_n is the thermal neutron velocity.

The isotope transformation model equations are (section 9.3.1).

$$\begin{aligned}\frac{dN_i}{dt} = &-\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} - \\ &- \sigma_{c,i} N_i \Phi + \sum_{j \neq i} \sigma_{c,j} N_j \Phi \gamma_{c,j \to i} - \\ &- \sigma_{f,i} N_i \Phi + \sum_{j \neq i} \sigma_{f,j} N_j \Phi \gamma_{f,j \to i} - \\ &- N_i C_i R^R + \sum_k N_i^F(k) R^F(k)\end{aligned}$$

N_i concentration of isotope i, (1/m^3),
λ_i decay constant of isotope i, (1/s),
$\sigma_{c,i}$ microscopic cross section for neutron capture of isotope i, (m^2),
$\sigma_{f,i}$ microscopic cross section for fission, (m^2), for isotope i ,
$\gamma_{d,j \to i}$ yield fraction of isotope i from the decay of isotope j ,
$\gamma_{c,j \to i}$ yield fraction of isotope i from the neutron capture by isotope j,
$\gamma_{f,j \to i}$ yield fraction of isotope i from fission of isotope j ,
Φ neutron flux (one-group) (1/m^2-s),
R^R user-defined relative rate of removal of fuel elements from the reactor core (1/s),
$R^F(k)$ user-defined relative rate of inserting fuel elements into the reactor core, (1/s),
$N_i^F(k)$ user-defined concentration of isotope i in the "fresh" fuel, being loaded into the core, (1/m^3).

The purpose of the isotope transformation model is to calculate decay heat, concentrations of poisons, such as Xe-135, Sm-149, as well as eventual changes of fuel concentrations due to burn-up, continuous reload (HTR/PBMR), etc. (section 9.3.2). For example, the isotope chain of the main poison, Xe-135, consists of two isotopes, I-135 and Xe-135, with the following data:



The sum of fission power and the decay power from all isotopes is available as the Control Function called the "reactor kinetics CF". Typically this CF is then used as a source of power for the solid structures (1-D or 2-D Solid Heat Conductors) that represent the fuel, with appropriate power fractions that take into account the axial and radial peaking factors.

9.4.3 Point Kinetics Model Equations - Circulating Fuel

For circulating fuel the RK Package calculates the isotope sources as follows:

- Delayed neutron precursors:

$$S_{V,i} = \frac{\beta_i}{\Lambda} n(t)$$

$S_{V,i}$ source of delayed neutron precursor *i* per unit volume, (1/m^3-s).

- Isotopes from the RK package (fission product, poisons, etc.):

$$S_{V,i} = -\sigma_{c,i} N_i \Phi + \sum_{j \neq i} \sigma_{c,j} N_j \Phi \gamma_{c,j \to i} -$$
$$- \sigma_{f,i} N_i \Phi + \sum_{j \neq i} \sigma_{f,j} N_j \Phi \gamma_{f,j \to i} -$$
$$- N_i C_i R^R + \sum_k N_i^F(k) R^F(k)$$

$S_{V,i}$ source of delayed neutron precursor *i* per unit volume, (1/m^3-s).

The source is distributed over all core volumes, according to the user-defined fuel weighting factors, W_K:

$$S_i(K) = S_{V,i} \cdot \frac{W_K}{\overline{W}} \cdot V_{pool}(K)$$

$S_i(K)$ source of delayed neutron precursor *i* in Control Volume *K*, (1/s),
W_K user-defined weighting factor for the core volume *K*, (-),
$\overline{W}$ average weighting factor, equal to: $\overline{W} = \sum_{K=1}^{N} W_K / N$
$V_{pool}(K)$ volume of pool (liquid) present in Control Volume *K*, (m^3),
N number of core volumes.

Finally, the mass source $S_{M,i}$, in kg/s for the Control Volume *K* is obtained from:

$$S_{M,i}(K) = S_i(K) \cdot \frac{N_{Av}}{M_i}$$

$S_{M,i}$ mass source of delayed neutron precursor *i* in Control Volume *K*, (kg/s),
N_{Av} Avogadro number, 6.022×10^{26} ($kmol^{-1}$),
M_i molar weight of the delayed neutron precursor *i*, (kg/kmol).

The sources calculated from the above equations are passed on to the RT Package. This Package calculates the isotope concentrations in Control Volumes using the standard models (inter-volume transport with flowing fluid, filters, etc.), as described in detail in Chapter 12 (Radioactive Particle Transport Package). For a single Control Volume, the equation used in the RT Package for the atmosphere or the pool of each Control Volume is:

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + S_{V,i}$$

N_i concentration of isotope *i*, ($1/m^3$),
λ_i decay constant of isotope *i*, (1/s),
$\gamma_{d,j \to i}$ yield fraction of isotope *i* from the decay of isotope *j* ,
$S_{V,i}$ external sources of isotope *i* per unit volume, ($1/m^3$-s), including eventual release of isotope from fuel, transport from neighboring volumes with flowing fluid, condensation or evaporation, sorption on surfaces, etc.

The masses of isotopes in the core volumes are passed back to the RK Package which finally calculates the neutron concentrations, power, reactivity effects of poisons, etc. The isotope average concentrations in the core are obtained from:

$$N_i = \sum_{K=1}^{N} N_i(K) \cdot \frac{V_{pool}(K)}{V_R} = \sum_{K=1}^{N} N_i(K) \cdot F_K$$

$V_{pool}(K)$ volume of pool (liquid) in the Control Volume *K*, (m^3),
$N_i(K)$ concentration of isotope *i* in the pool of Control Volume *K*, (-),
F_K = $V_{pool}(K) / V_R$,
V_R current reactor volume, (m^3), sum of pool volumes in all core Control Volumes:

$$V_R = \sum_{K=1}^{N} V_{pool}(K)$$

The final point kinetics equations of the applied to circulating fuel are

$$\frac{dn(t)}{dt} = \frac{\beta(R(t)-1)}{\Lambda} n(t) + \sum_i \lambda_i C_i(t) + S_{V,ext}(t)$$

$$C_i(t) = \sum_{K=1}^{N} C_i(K) \cdot F_K$$

The first equation is computed by the RK package, while the second is computed by the RK Package.

The isotope transformation model equations for circulating fuel are:

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + S_i$$

$$N_i = \sum_{K=1}^{N} N_i(K) \cdot F_K$$

Here again, the first equation is computed by the RK package, while the second is computed by the RK Package.

The reactor fission power, Q_f (W), is calculated from the same formula as in case of the solid fuel:

$$Q_f(t) = n(t) \cdot V_R \cdot \Sigma_f \cdot \nu_n \cdot q_f$$

However, in case of circulating fuel the reactor volume, V_R, may vary in time, as it is equal to the sum of pool volumes in all Control Volumes defined as core volumes. The fission power is available in the Control Function defined as the "reactor kinetics CF". Typically this CF is then used as a source of power for the pool of Control Volumes representing the core, with appropriate power fractions that take into account the axial and radial peaking factors.

In contrast to the solid fuel type, in case of liquid fuel only the fission power is included in the reactor kinetics CF. This is because the decay power is calculated by the RT Package and is automatically added to the pool of all Control Volumes.

9.4.4 Mapping of Isotopes Between the RK and the RT Packages

Mapping is required for the circulating fuel. At least one isotope chain must be mapped, which is the delayed neutron precursors. Additionally other chains may be mapped, such as the 11 isotopes of the main decay heat producers, the main poisons, etc.

During the mapping, new isotopes are created in the RT Package. If there are already isotopes present in the RT Package, the mapped isotopes are added as additional chains, below the chains already present. All properties of the mapped isotopes are taken from the corresponding isotopes in the RK Package. This includes: name, molar weight, fission yields, decay yields, absorption cross section, decay constants, as well as energy per decay. In case of delayed neutron precursors, the value of energy per decay is not used by the RK Package. The RT Package requires a positive heat for a decaying isotope. It is set arbitrarily to a relatively small value (0.1 MeV, typical energies are ~1 MeV). A small value is used to minimize discrepancies between the RK and the RT Package.

Below an example of mapping output is shown for the case where all isotopes from the RK Package are mapped to the RT Package and no other isotopes are present in the RT Package.

```
 =RT=  DETAILED MAPPING OF ISOTOPES FROM RK TO RT :

         RK Isotope          RT Isotope
         ----------          -------------
       - 101 Th-232   <->  FP-111 Th-232
       - 102 Th-233   <->  FP-112 Th-233
       - 103 Th-234   <->  FP-113 Th-234
       - 104 Pa-233   <->  FP-114 Pa-233
       - 105 Pa-234   <->  FP-115 Pa-234
       - 106  U-233   <->  FP-116  U-233
       - 107  U-234   <->  FP-117  U-234
       - 108  U-235   <->  FP-118  U-235
       - 201  U-238   <->  FP-121  U-238
       - 202  U-239   <->  FP-122  U-239
       - 203  U-240   <->  FP-123  U-240
       - 204 Np-239   <->  FP-124 Np-239
       - 205 Np-240   <->  FP-125 Np-240
       - 206 Pu-239   <->  FP-126 Pu-239
       - 207 Pu-240   <->  FP-127 Pu-240
       - 208 Pu-241   <->  FP-128 Pu-241
       - 301  I-135   <->  FP-131  I-135
       - 302 Xe-135   <->  FP-132 Xe-135
       - 401 Nd-149   <->  FP-141 Nd-149
       - 402 Pm-149   <->  FP-142 Pm-149
       - 403 Sm-149   <->  FP-143 Sm-149
       - 501 Eu-157   <->  FP-151 Eu-157
       - 502 Gd-157   <->  FP-152 Gd-157
       - 601 DHG-01   <->  FP-161 DHG-01
       - 602 DHG-02   <->  FP-162 DHG-02
       - 603 DHG-03   <->  FP-163 DHG-03
       - 604 DHG-04   <->  FP-164 DHG-04
       - 605 DHG-05   <->  FP-165 DHG-05
       - 606 DHG-06   <->  FP-166 DHG-06
       - 607 DHG-07   <->  FP-167 DHG-07
       - 608 DHG-08   <->  FP-168 DHG-08
       - 609 DHG-09   <->  FP-169 DHG-09
       - 610 DHG-10   <->  FP-16A DHG-10
       - 611 DHG-11   <->  FP-16B DHG-11
       - D.N.PREC.01  <->  FP-171 DNP-01
       - D.N.PREC.02  <->  FP-172 DNP-02
       - D.N.PREC.03  <->  FP-173 DNP-03
       - D.N.PREC.04  <->  FP-174 DNP-04
       - D.N.PREC.05  <->  FP-175 DNP-05
       - D.N.PREC.06  <->  FP-176 DNP-06
```

9.4.5 Additional Reactivity Effects

9.4.5.1 Non-Uniform Reactivity Effects

SPECTRA relies on the point kinetics model, so spatial effects are not taken into account. However some provisions have been made and tested against MSRE data, which allow introducing multi-dimensional effects in case of circulating fuel (molten salt reactors). The main problem lies in the difference between the neutron flux in graphite (where the thermal flux is generated) and the fuel salt (where it is mainly absorbed). Xenon exists both in the salt (where it is generated from decay of iodine) and in the graphite (due to migration from salt and diffusion into the graphite). Calculations show that the depth of xenon diffusion is of order of centimeters. Quite a significant amount of Xenon is therefore present in graphite, where the thermal neutron flux is relatively high. This is taken into account in a simplified way, described below. The theoretical background is based on reference [212].

In case of MSRE core with fuel salt and graphite moderator, the effective Xe concentration is ([212], sec. 7.1):

$$N_{Xe} = \frac{\int_{salt} N_s \Phi_2^* \Phi_2 dV_s + \int_{graphite} N_g \Phi_2^* \Phi_2 dV_g}{\int_{salt} \Phi_2^* \Phi_2 dV_s + \int_{graphite} \Phi_2^* \Phi_2 dV_g}$$

Here N_{Xe} is the importance-averaged concentration per unit reactor volume, N_s and N_g are the local concentrations per unit volumes of salt and graphite respectively, Φ_2 Φ_2^* are the thermal neutron flux and the adjoint thermal neutron flux respectively N_{Xe} is the uniform equilibrium concentration of xenon in the reactor, which produces the same reactivity effect as the actual distribution.

Effective Concentrations

This is approximated in SPECTRA by introducing the reactivity worth multipliers. The isotope concentrations in liquid (CV pools) are multiplied by RLITRK, while the concentrations in solids (SC/TC) are multiplied by RSITRK:

$$N_{eff} = \frac{\sum_{CV \in core} N_{CV} \times RLITRK \times V_{CV} + \sum_{SC \in core} N_{SC} \times RSITRK \times V_{SC}}{\sum_{CV \in core} RLITRK \times V_{CV} + \sum_{SC \in core} RSITRK \times V_{SC}}$$

Here N_{CV} and N_{SC} are the isotope concentrations in the core CV pools and core SCs respectively. The theoretical values of the reactivity worth multipliers are:

$$RLITRK = \int_{CV \in core} \Phi_{CV}^* \Phi_{CV} \, dV_{CV}$$

$$RSITRK = \int_{SC \in core} \Phi_{SC}^* \Phi_{SC} \, dV_{SC}$$

The values of RLITRK and RSITRK are defined by the user (input record 748000 - see Volume 2). The values are relative and may be scaled by any number. For example, let's assume that we define:

- RLITRK = 0.5, RSITRK = 1.0

Exactly the same results will be obtained when the values are defined as:

- RLITRK = 1.0, RSITRK = 2.0

Analyses performed for the MSRE showed that roughly speaking the ratio is 2.0: RSITRK/RLITRK ~ 2.0.

More detailed investigation showed that the ratio varies depending on the relative xenon concentration. In the initial situation, when graphite is "clean" (i.e. no xenon is present in graphite) the neutron flux in graphite is relatively large. In this case the ratio is larger, 3.0 to 3.5. When xenon has migrated into the graphite, the neutron flux in graphite is becoming smaller and the ratio RSITRK/RLITRK is also smaller. The value is approximately 1.8 for large concentrations in graphite. Values obtained based on the MSRE data [212] are shown in Figure 9-14. An approximation functions was developed to match the data:

$$f_2 = A + (B - A) \cdot \exp(-C \cdot x) = 1.77 + (3.5 - 1.77) \cdot \exp(-14x)$$

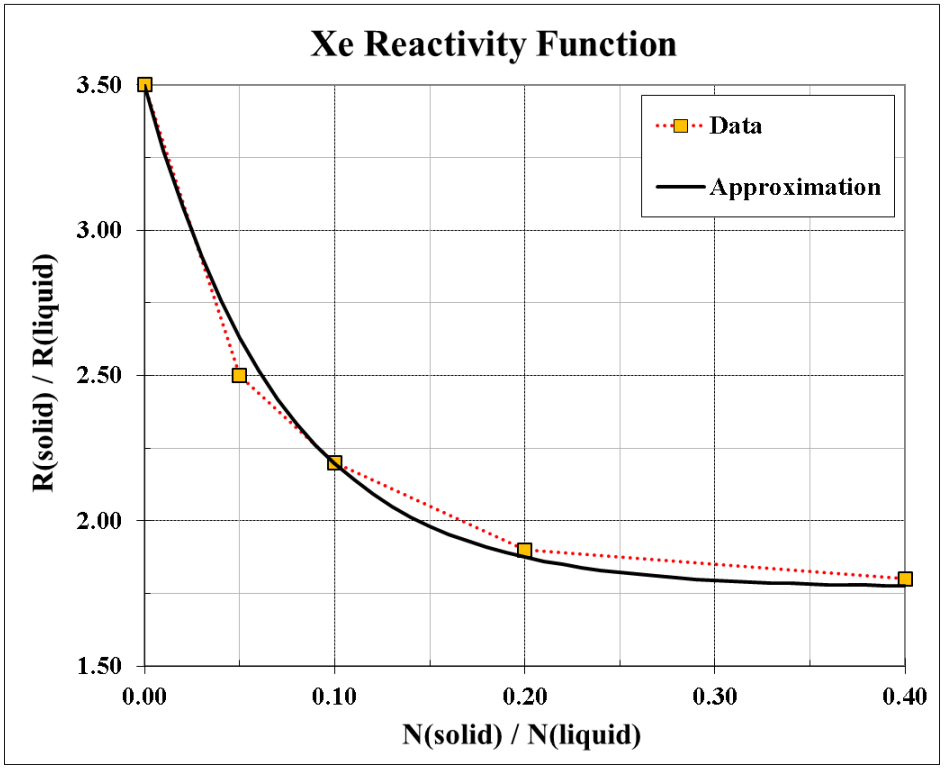


Figure 9-14 Ratio of (R_{solid}/R_{liquid}) versus (N_{solid}/N_{liquid})

The approximation functions are shown in Figure 9-14. The function may be applied in calculations when a positive value of IRSLRK is entered (see Volume 2). In such case, instead of using constant values of RSITRK, RLITRK, the ratio of RSITRK/RLITRK is defined by a Tabular Function number IRSLRK as a function of the ratio of the concentration in solid to the concentration in liquid of the user-defined isotope (for example Xenon-135) that is indicated by the input parameter JRSLRK. (The approximation functions must be tabulated to be used in calculations. An advantage of using tabulated functions is to obtain a relatively smooth function, defined by large number of data pairs, i.e. small intervals.) Note that the ratio RSITRK/RLITRK may be a function of only one isotope. The same value is applied for all isotopes in a problem.

Concentrations and Effective Concentrations in Solid/Liquid/Gas

Concentrations of isotopes in fuel and solids are calculated by the RT Package. The total number of atoms in liquid, solid, as well as an eventual gas phase, are available as plot parameters, defined as follows.

- RK-000-Nliq-XXXX (XXXX is the isotope number):

$$N_{liq} = \sum_{CV \in core} N_{CV,pool} \times V_{CV,pool} + \sum_{CV \in core} n_{CV,part}$$

$N_{CV,pool}$	volume concentration (atoms/m^3) in the pool of CV,
$V_{CV,pool}$	volume (m^3) of the pool of CV,
$n_{CV,part}$	number of atoms attached to the aerosol particles deposited in the CV pool.

- RK-000-Nsol-XXXX (XXXX is the isotope number):

$$N_{sol} = \sum_{SC \in core} N_{SC} \times A_{SC} + \sum_{SC \in core} n_{SC}$$

N_{SC}	surface concentration (atoms/m^2) deposited on SC adjacent to core CV,
A_{SC}	surface area (m^2) of SC,
n_{SC}	number of atoms attached to the aerosol particles deposited on SC,

- RK-000-Ngas-XXXX (XXXX is the isotope number):

$$N_{gas} = \sum_{CV \in core} N_{CV,atms} \times V_{CV,atms} + \sum_{CV \in core} n_{CV,atms}$$

$N_{CV,atms}$	number concentration (atoms/m^3) in the atmosphere of CV,
$V_{CV,atms}$	volume (m^3) of the atmosphere of CV,
$n_{CV,atms}$	number of atoms attached to the aerosol particles in the CV atmosphere.

The effective concentrations are defined as described below.

- RK-000-Xliq-XXXX (XXXX is the isotope number):

$$N_{liq,eff} = \frac{\sum_{CV \in core} (N_{CV,pool} \times V_{CV,pool} + n_{CV,part}) \times RLITRK}{RLITRK \times V_{liq} + RSITRK \times V_{sol} + RGITRG \times V_{gas}}$$

V_{liq}	volume of liquid in the core, obtained from:

$$V_{liq} = \sum_{CV \in core} V_{CV,pool}$$

V_{gas} volume of gas in the core, obtained from:

$$V_{liq} = \sum_{CV \in core} V_{CV,atms}$$

V_{sol} volume of solid structures in the core, obtained from:

$$V_{sol} = \frac{1 - RLTVRK}{RLTVRK} \cdot V_{liq} = \frac{1 - RLTVRK}{RLTVRK} \cdot \sum_{CV \in core} V_{CV,pool}$$

RLTVRK liquid volume fraction in the reactor, $V_{liq}/(V_{liq} + V_{sol})$, defined by the user.

Note: the volume of solid structures in the core is not necessarily equal to the volume of all structures (SC / TC) adjacent to the core Control Volumes (CVs with positive fuel weighting factors). Concentrations of isotopes are calculated by the code per unit surface area. An effective "depth" of solid structures is obtained as the ratio between the solid volume, V_{sol}, obtained as shown and the surface area:

$$D_{eff} = \frac{V_{sol}}{\sum_{SC \in core} A_{SC}}$$

The value of effective depth is calculated by the code and printed at the end of the RT Package edit:

```
=RT= PARAMETERS FOR REACTOR KINETICS WITH CIRCULATING FUEL
     - CORE FLUID VOLUME  = 4.72016E-01 m3
     - CORE SOLID VOLUME  = 2.01752E+00 m3
     - SOLID SURFACE AREA = 1.35871E+02 m2
     - EFFECTIVE DEPTH    = 1.48488E-02 m
```

- RK-000-Xsol-XXXX (XXXX is the isotope number):

$$N_{sol,eff} = \frac{\sum_{CV \in core} (N_{SC} \times A_{SC} + n_{SC}) \times RSITRK}{RLITRK \times V_{liq} + RSITRK \times V_{sol} + RGITRG \times V_{gas}}$$

- RK-000-Xgas-XXXX (XXXX is the isotope number):

$$N_{gas,eff} = \frac{\sum_{CV \in core} (N_{CV,atms} \times V_{CV,atms} + n_{CV,atms}) \times RGITRK}{RLITRK \times V_{liq} + RSITRK \times V_{sol} + RGITRG \times V_{gas}}$$

Xe Poisoning and Xe Reactivity

In relating the total reactivity change, it is convenient to define a third quantity, the effective thermal poison fraction, P_{Xe}. This is the number of neutrons absorbed in xenon per neutron absorbed in the fuel (U-235), weighted with respect to neutron importance ([212], eq. 7.2):

$$P_{Xe} = \frac{\sigma_{Xe} \int_{reactor} \Phi_2^* \Phi_2 \, dV}{\int_{reactor} (N_U \sigma_{U,1} \Phi_1^* \Phi_1 + N_U \sigma_{U,2} \Phi_2^* \Phi_2) \, dV} \cdot N_{Xe}$$

or:

$$P_{Xe} = \frac{absorption\,in\ Xe}{absorption\,in\ fuel}$$

N_U concentration of fuel (U-235) per unit volume,
$\sigma_{U,1}$ microscopic absorption cross section for fast neutrons,
$\sigma_{U,2}$ microscopic absorption cross section for thermal neutrons,
σ_{Xe} xenon thermal absorption cross section.

Recalling the formula for N_{Xe}, shown earlier this section, we have:

$$P_{Xe} = \sigma_{Xe} \frac{\int_{salt} N_{Xe,s} \Phi_2^* \Phi_2 dV_s + \int_{graphite} N_{Xe,g} \Phi_2^* \Phi_2 dV_g}{\int_{reactor} (N_U \sigma_{U,1} \Phi_1^* \Phi_1 + N_U \sigma_{U,2} \Phi_2^* \Phi_2) \, dV}$$

The relation between total xenon reactivity R_{Xe} and poisoning P_{Xe} is given by ([212], eq.7.3):

$$R_{Xe} = \left(\frac{\delta k}{k} \right)_{Xe} = - \frac{\int_{reactor} (N_U \sigma_{U,1} \Phi_1^* \Phi_1 + N_U \sigma_{U,2} \Phi_2^* \Phi_2) \, dV}{\int_{reactor} (\nu N_U \sigma_{f,1} \Phi_1^* \Phi_1 + \nu N_U \sigma_{f,2} \Phi_2^* \Phi_2) \, dV} \cdot P_{Xe}$$

or:

$$R_{Xe} = - \frac{absorption\,in\ fuel}{fission\,in\ fuel \times number\,of\ neutrons\,per\ fission} \cdot P_{Xe}$$

This can be written as:

$$R_{Xe} = - \frac{\Sigma_U}{\nu \Sigma_f} \cdot P_{Xe}$$

The proportionality coefficient is referred to as “poison fraction”. References [212] (Table 3.5) and [213] (page 49) give the values of poison fraction between 0.691 and 0.752.

9.4.5.2 Change of Liquid Volume

In case of solid fuel the core volume is constant (user-defined). In case of circulating fuel the core volume is, in general, time dependent. The liquid volume may change because of for example fuel dump, when the reactor eventually becomes completely dry, but also in case of bubbles or other materials are entering the core region. The total core volume is equal to the sum of liquid volumes of the Control Volumes that are defined as the core volumes (note that the core volumes are defined by a positive fuel weighting factor,). When the core volume changes, reactivity will generally change, for example due to different neutron leakage. This effect is taken into account by a user-defined Tabular Function. The function defines reactivity (in dollars) as a function of relative core volume:

$$R_V[\$] = f\left(\frac{V_R(t)}{V_{REF}}\right)$$

R_V reactivity effect caused by changes of core volume, ($),
$V_R(t)$ current core volume, m^3 ,
V_{REF} reference core volume, m^3 ,
f user-defined tabular function.

Typically the reactivity decreases with decreasing core volume. The values of this function must be obtained from external calculations.

9.4.6 Options

Three options are available, concerning the initial distribution of isotopes:

- ***Option* 1:** Isotope concentrations are initially set only in the core volumes, i.e. Control Volumes with a positive fuel weighting factor (WTFFRK>0). The concentrations are proportional to the fuel weighting factors, WTFFRK. Figure 9-15 shows the initial concentrations of delayed neutron precursors (six groups) and fuel (U-235) isotopes. The concentrations are proportional to the assumed power profile in the core and zero outside the core.
- ***Option* 2:** Isotope concentrations are initially set in all Control Volumes belonging to a user-defined CV group. The concentrations are not proportional to the fuel weighting factors, WTFFRK, in the core volumes. Figure 9-16 shows the initial concentrations of delayed neutron precursors and fuel. The concentrations constant in the entire loop.
- ***Option* 3:** Isotope concentrations of fuel (σ_f > 0) are initially set in all Control Volumes belonging to a user-defined CV group. The concentrations are not proportional to the fuel weighting factors, WTFFRK. All other isotopes (fission products, including delayed neutron precursors) are initially set only in the core volumes, i.e. Control Volumes with a positive fuel weighting factor (WTFFRK>0). The concentrations are proportional to the fuel weighting factors, WTFFRK, in the core volumes. Figure 9-17 shows the initial concentrations of delayed neutron precursors and fuel. The delayed neutron concentrations are proportional to the assumed power profile in the core and zero outside the core. The concentrations of fuel are constant in the entire loop. This option gives the most realistic initial conditions and is therefore selected as the default option.

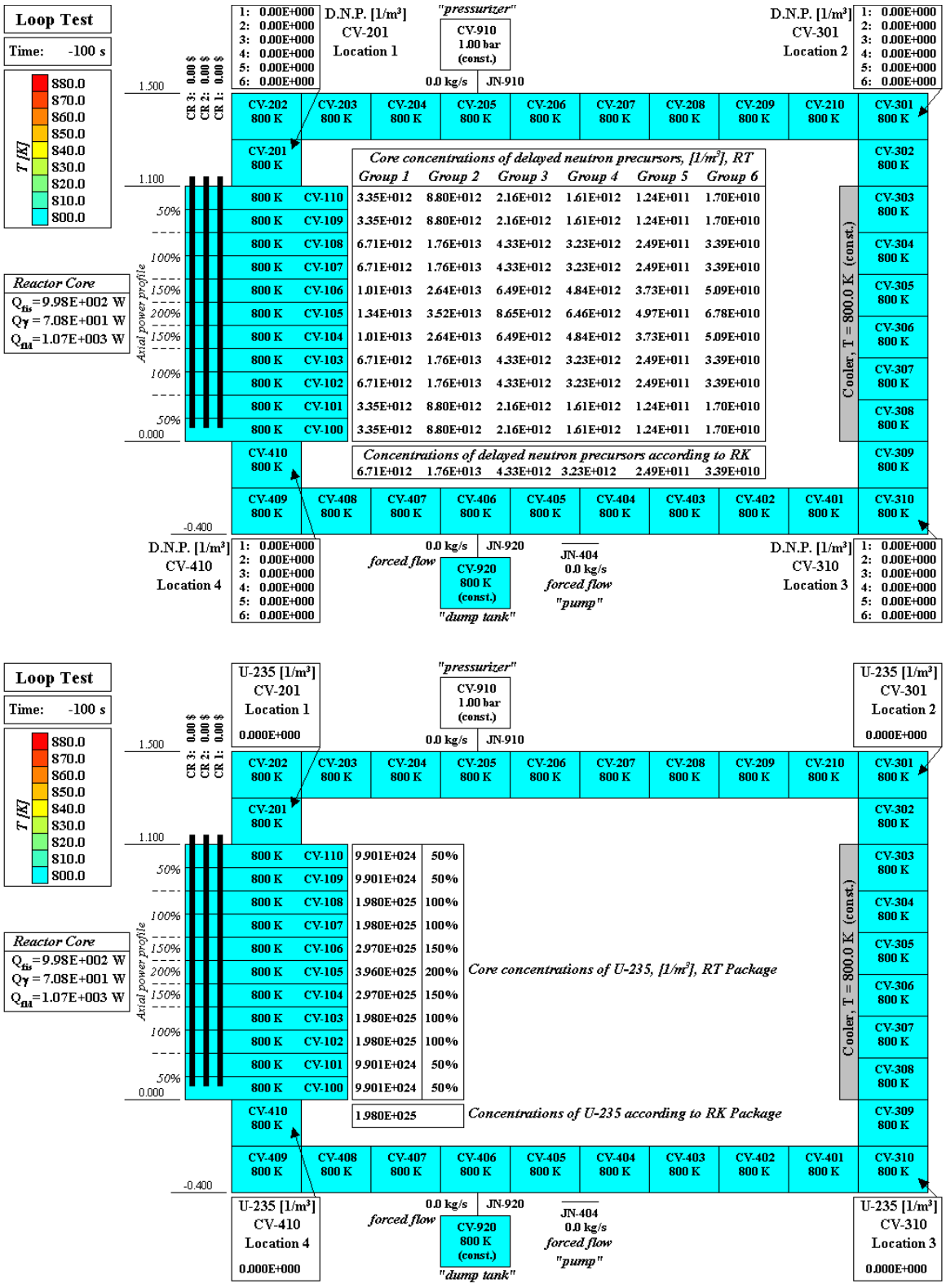


Figure 9-15 Initial distribution of isotopes - Option 1, (above) delayed neutron precursors, (below) fuel

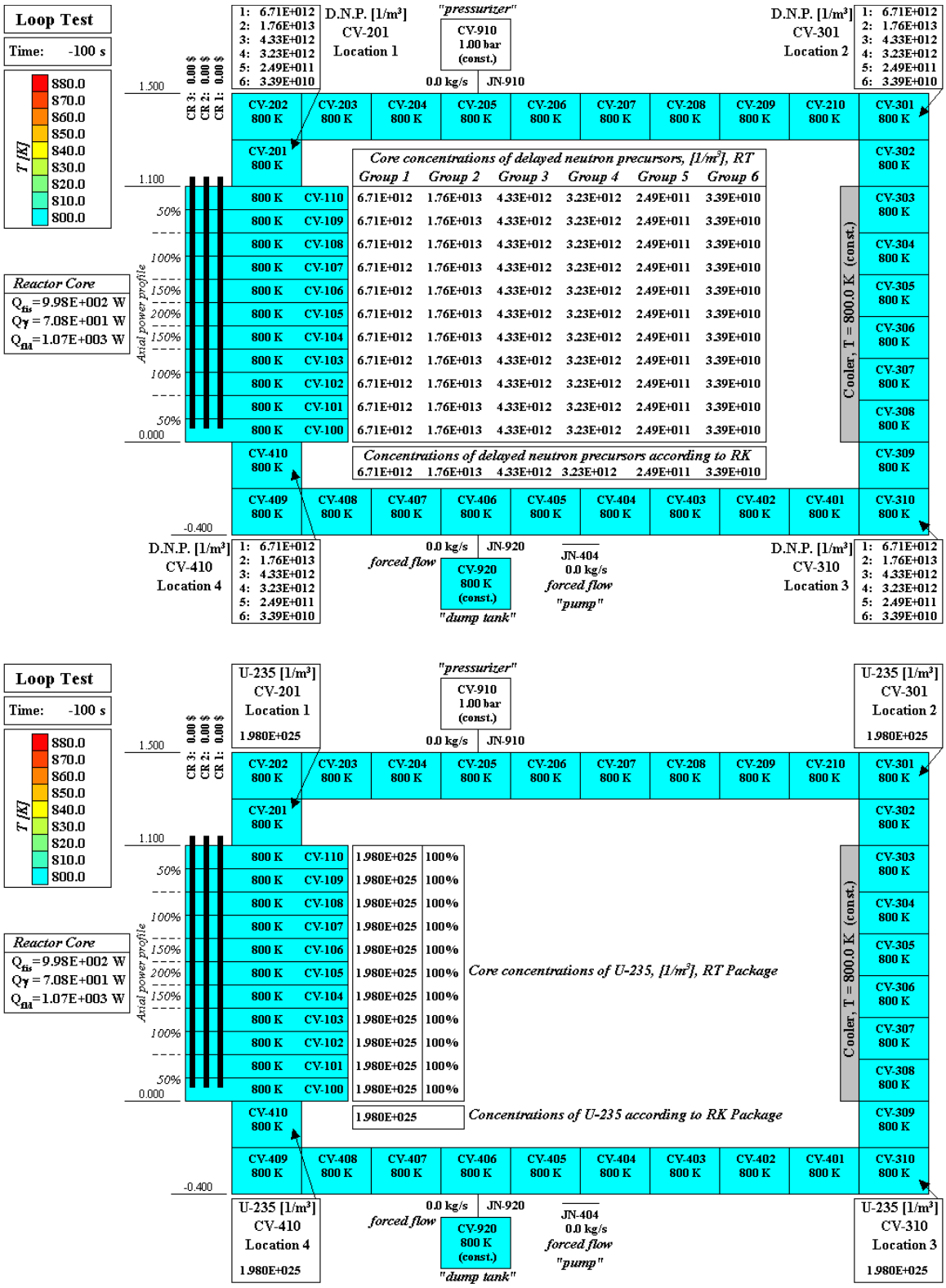


Figure 9-16 Initial distribution of isotopes - Option 2, (above) delayed neutron precursors, (below) fuel

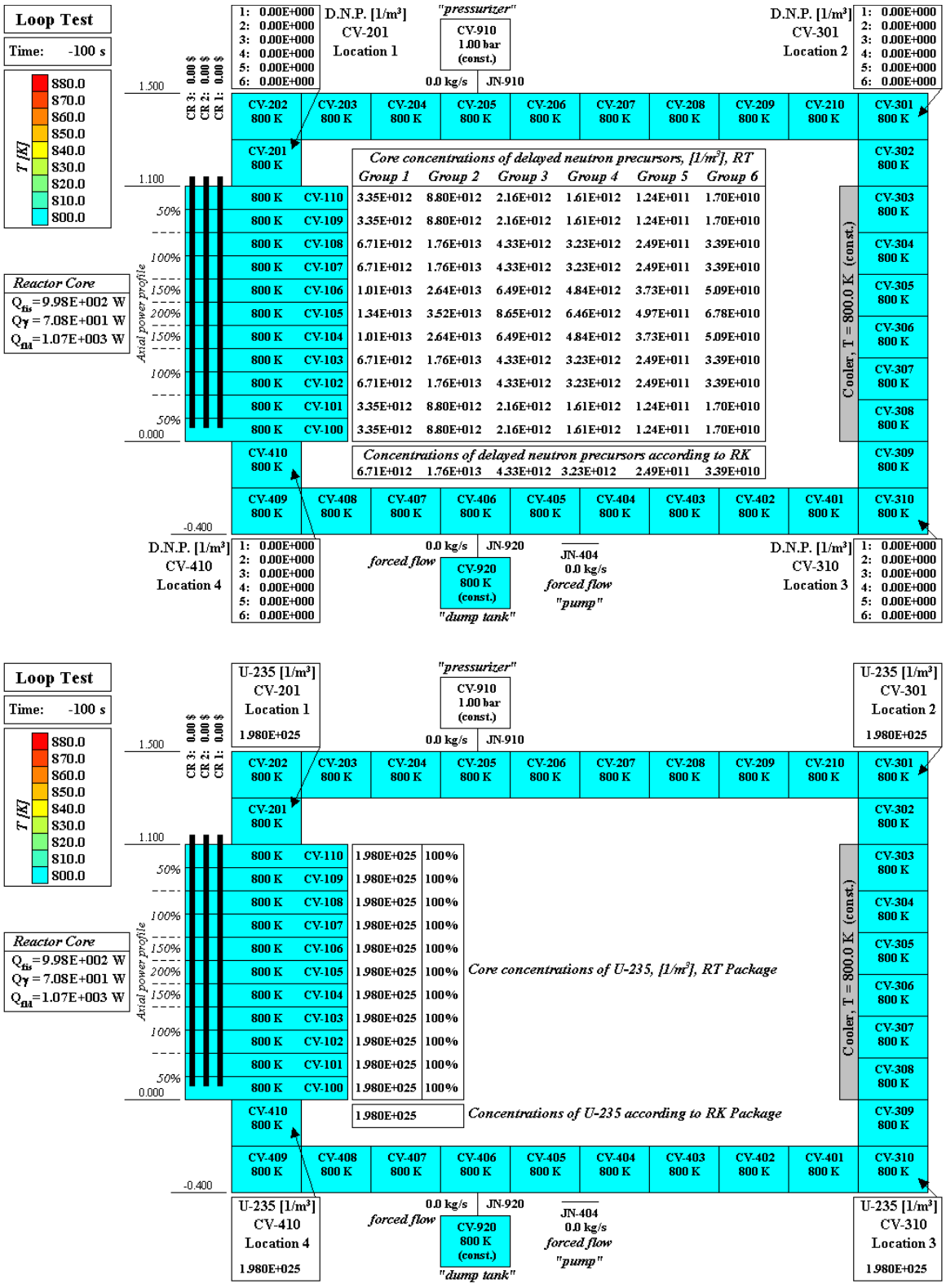


Figure 9-17 Initial distribution of isotopes - Option 3 (default), (above) delayed neutron precursors, (below) fuel

9.4.7 Analyzing Fission Product Behavior in Circulating Fuels

When a user wishes to analyze behavior of a fission product decay chain that is not a part of the RK Package, there are two alternative ways to do that.

1. The decay chain can be defined in the RK Package as a user-defined chain and then mapped to the RK Package. This way the calculation procedure will be exactly the same as used for the delayed neutron precursors and the default chains of the RK Package.

2. The user may activate the option to calculate the source of the isotopes within RT Package (non-mapped from the RK) on the same way as those mapped from the RK Package. The source is calculated by the code as:

$$S_i = \gamma_{f,i} \cdot \sum_{k \in fis} \sigma_{f,k} N_k \Phi - \sigma_{a,i} N_i \Phi + \sum_{j \neq i} \sigma_{a,j} N_j \Phi \gamma_{a,j \to i}$$

Here Φ is the neutron flux (1/m^2-s), N_k is the concentration (1/m^3) of isotope k, $\sigma_{f,k}$ is the fission cross section of isotope k, $\sigma_{a,j}$ is the neutron capture (non-fissile absorption) cross section of isotope j, $\gamma_{f,i}$ is the average yield fraction of isotope i from fission of all fissile isotopes, $\gamma_{a,j \to i}$ is the yield fraction of isotope i due to neutron capture by isotope j.

In such case the isotope chain is present only in the RT Package and the calculational procedure is somewhat different but the results are almost exactly the same as in method 1.

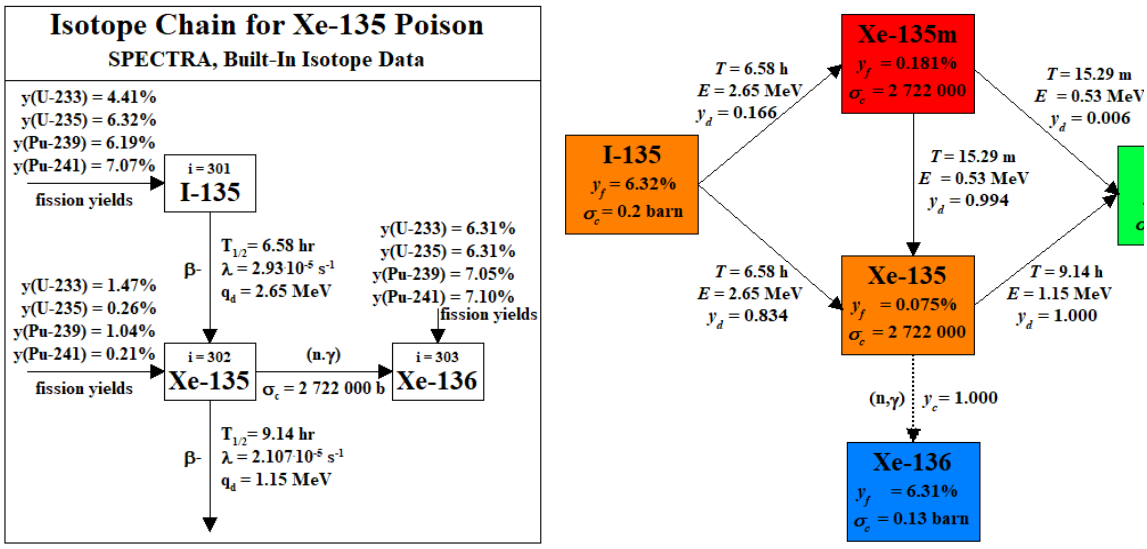


Figure 9-18 Decay chain M_W = 135 / 136,
left: RK Package, see: “Description of the Built-in Isotope Library”
right: RT Package, see: “Isotope Chains (Decay Chains)”

One difference is related to the fission yields. In the RK Package, the built-in fission yields are different for each fissile isotope, as the yield fraction is given as $\gamma_{f,k\to i}$: yield fraction of isotope *i* from fission of a nuclide of isotope *k*. In the RT Package uses by default only one yield fraction for each isotope. For example, the built-in Xe-135 chain in the RK Package is shown in Figure 9-18, left. The built-in values for the RT Package are shown in Figure 9-18, right. It is possible to define the individual fission yields for each fissile isotope, but this must be done by the user in the input file (examples are shown in Volume 3 test cases Kr-88-SOL, Kr-88-LIQ, Xe-136).

As an example, let's consider chain 137, which is one of the built-in chains for the RT Package. The chain is shown in Figure 9-19. The two methods are shortly described below. A more detailed description is provided in Volume 3.

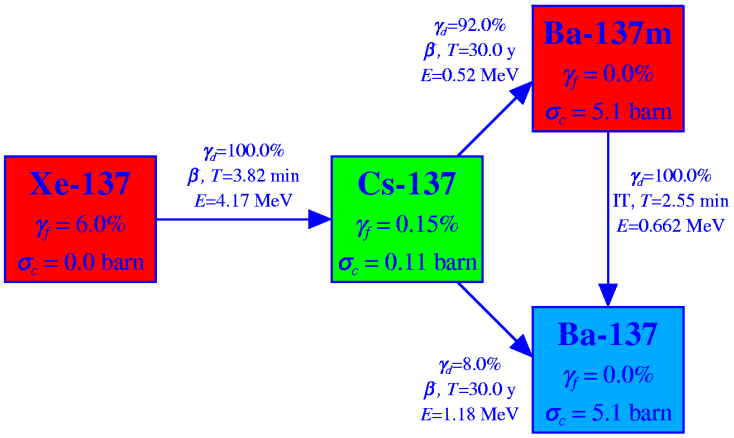


Figure 9-19 Decay chain, M_w = 137

1. Method 1
   - Step 1. The decay chain is defined in the RK Package as a user-defined chain, using records 741XXX, 742XXX, 743XXX. At the same time, the built-in chain 137 is NOT activated in the RT Package.
   - Step 2. The user-defined chain is mapped to the RT Package using IMAPRK ($h \neq 0$).
   - Step 3. The values of molar weight as well as the vapor classes, which are not a part of the RK, are defined within the RT Package using records 880XXY, 885XXY, 886XXY. Those values are not defined within the RK Package and during the mapping process they receive default values, molar weight of 235/2=117.5 and vapor class 9. (Note that this is only necessary in case of a user-defined RK chain. In case of the built-in chains, for example Xe-135 chain, these parameters are correctly defined during mapping.)
2. Method 2.
   - Step 1. The built-in chain 137 is activated in the RT Package.
   - Step 2. The option to calculate the RT isotope source in the circulating fuel is activated by setting IRTSRK to 2

Results of both methods are compared in Figure 9-20 and Figure 9-21, showing stationary state in a simple test loop. It is seen that the concentrations of Cs-137 as well as the source strengths in the core are practically identical.

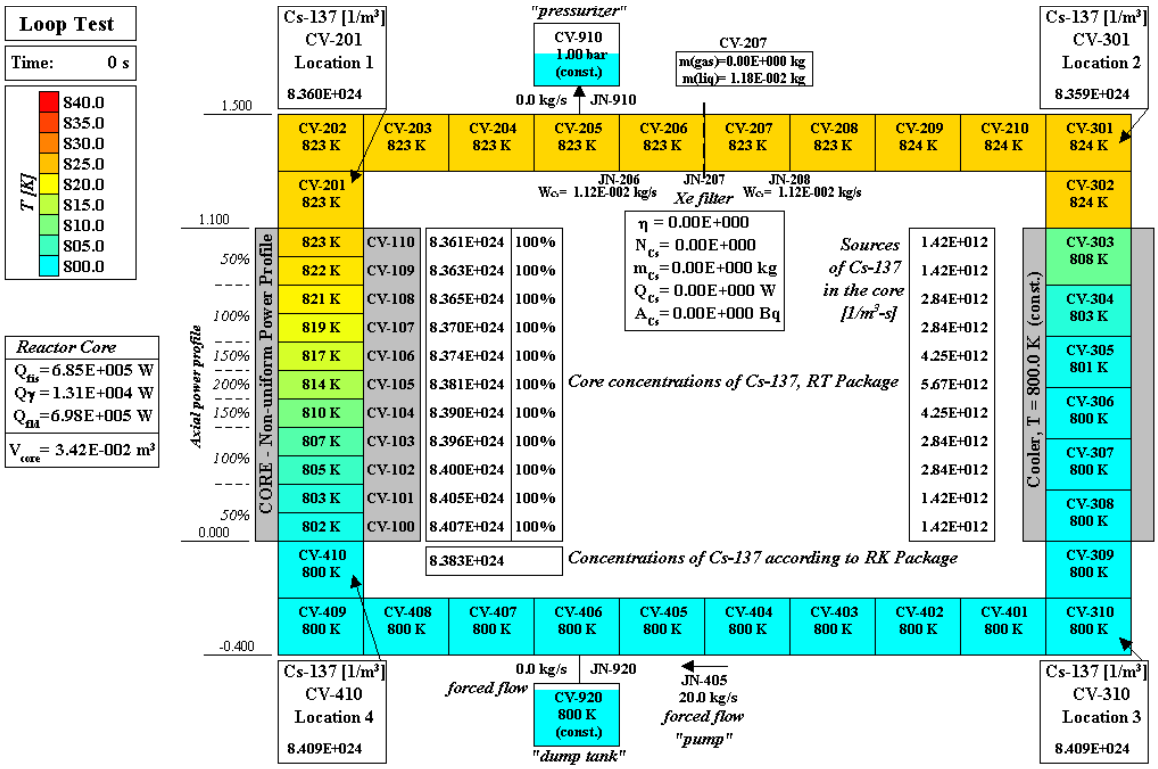


Figure 9-20 Analysis of Chain 137, results of method 1

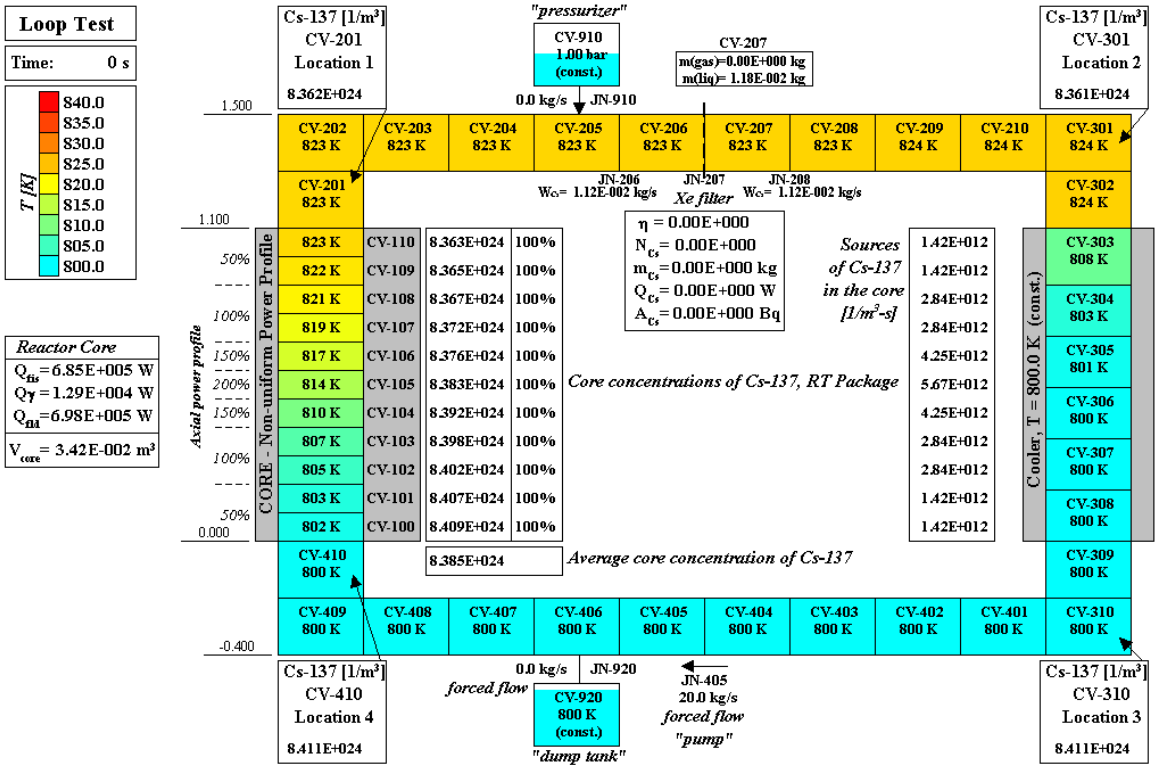


Figure 9-21 Analysis of Chain 137, results of method 2

9.4.8 Isotope Averaging Scheme

An averaging scheme, similar to the temperature averaging scheme (section 5.6), is available for isotopes in the circulating fuel. The concept is described in this section.

Without averaging, the concentration of isotopes, e.g. delayed neutron precursors (DNP), in the core is obtained by summing up concentrations is all Control Volumes belonging to the core (volumes with positive fuel temperature weighting factors). The formula is (see section 9.4.3):

$$N_i = \sum_{K=1}^{N} N_i(K) \cdot \frac{V_{pool}(K)}{V_R}$$

Here $N_i(K)$ is the concentration of isotope *i* in the pool of Control Volume *K*, (-). This situation is illustrated in Figure 9-22, showing DNP concentrations in the Standard MSR System test (see Volume 3). The concentrations of DNP increase as the salt flows through the core, as in this test case there is a continuous flow of salt. It may be argued that the average concentration in every core volume is better represented by an average of the concentration in the CV (which is equal to the outlet concentration, as transported through the outlet JN) and the inlet concentration. This argument is very similar to the argument leading to the temperature averaging (section 5.6).

When the isotope averaging scheme is applied (ITAISC>0, Volume 2), the average concentration in the core is obtained from:

$$N_i = \sum_{K=1}^{N} \overline{N_i(K)} \cdot \frac{V_{pool}(K)}{V_R}$$

The average concentration in the volume K is defined

$$\overline{N_i(K)} = \frac{0.5 \cdot N_i(K) + X \cdot N_i(K_{inlet})}{0.5 + X}$$

Here *X* is defined as:

$$X = \frac{0.5}{ITAISC}$$

For example, if ITAISC = 1, the average value is equal to:

$$\overline{N_i(K)} = 0.5 \cdot N_i(K) + 0.5 \cdot N_i(K_{inlet})$$

This means that the average concentration is obtained as the average value between the CV value and the inlet value. This situation is illustrated in Figure 9-23.

If ITAISC=3, then the average value is equal to:

$$\overline{N_i(K)} = \frac{0.5 \cdot N_i(K) + 0.5/3 \cdot N_i(K_{inlet})}{0.5 + 0.5/3} = 0.75 \cdot N_i(K) + 0.25 \cdot N_i(K_{inlet})$$

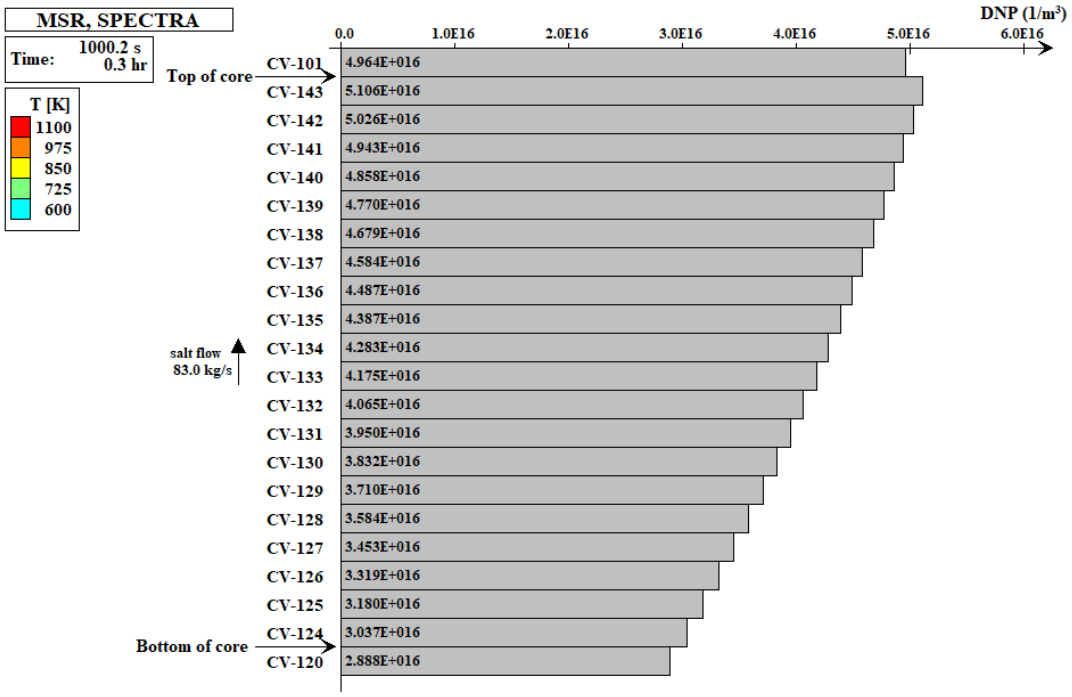


Figure 9-22 Delayed neutron concentrations, no isotope averaging

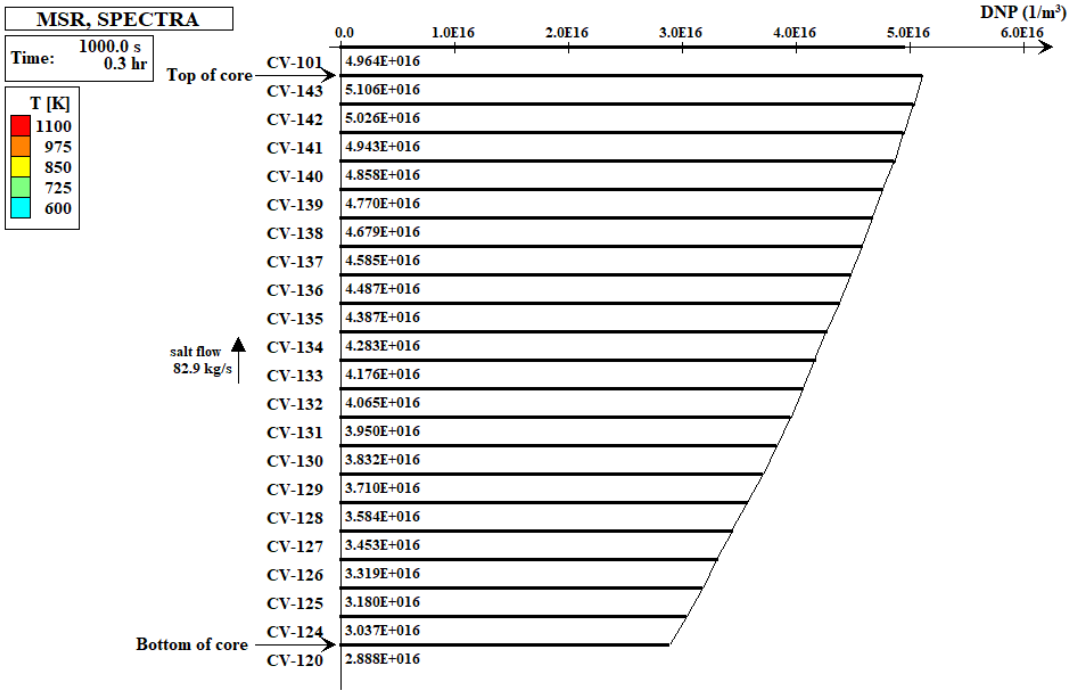


Figure 9-23 Delayed neutron concentrations, with isotope averaging

The averaging scheme for isotopes is linked to the Temperature Averaging (T-A) scheme (section 5.6). Therefore, in order to apply this scheme, the user must first activate the T-A scheme. All parameters valid for the T-A scheme also affect the isotope averaging, namely

- The limiting flows (W_{TA} in section 5.6, WTALSC / WTARSC, VLLMSC in Volume 2) below which the averaging is reduced. When the flow direction reverses, the averaging scheme is automatically adjusted. When there is no flow, the averaging is not applied and the value of isotope concentration (as well as temperature) is equal to the CV value.

9.5 Nodal Point Kinetics

9.5.1 Introduction

The point kinetics, described in section 9.2, provides the simplest, 0-D model to calculate a time-dependent reactor power behavior. The nodal point kinetics, described in this chapter, is a 1-D, 1-group model, which is using basically the same input as is required by the point kinetics. The advantage of this model is a possibility of extending the point kinetics model into a 1-D representation, which can be very useful in some cases (described in Volume 3) without the need of supplying a significant amount of neutron cross section data. The most accurate 3-D neutronics model may be used by an interactive coupling between SPECTRA and a 3D neutronic code. Generally such multi-physics coupling is realized using the External Data File Package (Chapter 18).

The nodal point kinetics is based on the references [219], [220], [221]. However, the formulation is somewhat different for the reasons explained in section 9.5.2.

The derivation presented below is based on one-group diffusion equation, with space discretization valid for arbitrary node sizes. It is assumed that the diffusion coefficient may, in general, be different in different nodes. The one-group diffusion equation is [220]:

$$\frac{1}{v}\frac{d\Phi}{dt} = (1-\beta)\nu\Sigma_f\Phi - \Sigma_a\Phi + \sum_k \lambda_k C_k + D\nabla^2\Phi$$

Φ neutron flux, (1/m^2-s)
v neutron velocity, (m/s)
β sum of delayed neutron fractions, $\beta = \Sigma\,\beta_k$, (-)
ν average number of neutrons per fission
Σ_f macroscopic fission cross section (1/m)
Σ_a macroscopic absorption cross section (1/m)
λ_k decay constant of the delayed neutron precursor group k, (1/s)
C_k concentration of the delayed neutron precursor group k, (1/m^3)
D diffusion coefficient, (m)

9.5.2 Diffusion Term

Neutron diffusion in 1-D situation, shown in Figure 9-24, is considered. The diffusion term applied here is somewhat different than in the references [219], [220], [221] for the following reasons.

In a system code, the user determines size of the nodes. The method must take into account the fact that the node size may be different. Consequently, in general:

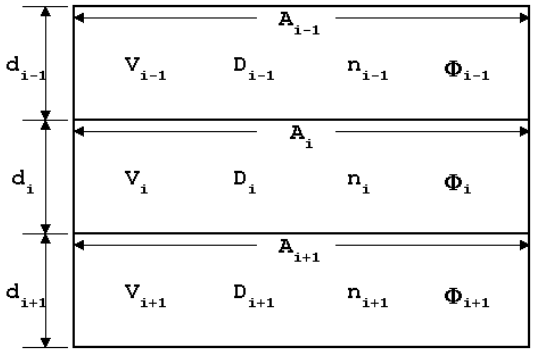


Figure 9-24 Numerical discretization for nodal kinetics

$$V_{i-1} \neq V_i \neq V_{i+1}$$
$$d_{i-1} \neq d_i \neq d_{i+1}$$
$$A_{i-1,i} \neq A_{i,i+1}$$

Also it is assumed that the diffusion coefficient may, in general, be different in different nodes:

$$D_{i-1} \neq D_i \neq D_{i+1}$$

In the formulation from [220] and [221] the diffusion coefficient D is constant therefore the equations presented there are not directly applicable. In the formulation from [219] the diffusion coefficient is node-dependent; however the approach presented there is not used here for a different reason. The diffusion term is in [219] proportional to:

$$\frac{D_{i-1}\Phi_{i-1} - D_i\Phi_i}{d_{ij}}$$

If $D_{i-1} \neq D_i$ non-physical results may be obtained. Suppose that $D_i \rightarrow 0$. For a given (constant) value of the neutron fluxes, $\Phi_{i-1} = \Phi_i$, the diffusion flux from the node i–1 to the node i will be maximum when $D_i \rightarrow 0$, which is against common sense. Furthermore, if $D_i \rightarrow 0$ and at the same time $\Phi_{i-1} < \Phi_i$, there will be a diffusion flux from the region with the smaller flux, Φ_{i-1}, to the region with the larger flux, Φ_i, simply because the product $D_i \cdot \Phi_i \rightarrow 0$.

In SPECTRA, the diffusion term is written analogously to the heat conduction term, where the heat conduction from one node (i) to another node (j), Q_{ij} (W), is given by:

$$Q_{ij} = \frac{A_{ij}}{R_{ij}}(T_i - T_j)$$

Here R_{ij} is the thermal resistance between the nodes i and j and is given by:

$$R_{ij} = \frac{d_i/2}{k_i} + \frac{d_j/2}{k_j}$$

Here k_i and k_j are the thermal conductivities in the nodes i and j.

The diffusion term is written in the same manner. The neutron transfer from node i–1 to node i (Figure 9-24), $\Gamma_{i-1,i}$ (1/s), is given by:

$$\Gamma_{i-1,i} = \frac{A_i}{X_{i-1,i}}(\Phi_{i-1} - \Phi_i)$$

Here $X_{i-1,i}$ is the diffusive resistance between the nodes i–1and i and is given by:

$$X_{i-1,i} = \frac{d_{i-1}/2}{D_{i-1}} + \frac{d_i/2}{D_i}$$

If one of the diffusion coefficients is becoming very small, e.g., $D_i \rightarrow 0.0$, then the diffusion term is zero, which is expected.

9.5.3 Nodal Kinetics Equations

The neutron balance equation shown in section 9.5.1 is written for a node i, which has a volume of V_i, using the diffusion term, shown in section 9.5.2:

$$\frac{1}{v}\frac{d\Phi_i}{dt} = (1-\beta)\nu\Sigma_f\Phi_i - \Sigma_a\Phi_i + \sum_k \lambda_k C_k + \frac{1}{V_i}\left[\frac{A_i}{X_{i-1,i}}(\Phi_{i-1} - \Phi_i) + \frac{A_{i+1}}{X_{i+1,i}}(\Phi_{i+1} - \Phi_i)\right]$$

The neutron flux is equal to:

$$\Phi_i = n_i v_i$$

Here n_i is the neutron density in the zone i (1/m^3) and v_i is the 1-group neutron velocity in the zone i (m/s). Therefore:

$$\frac{dn_i}{dt} = (1-\beta)\nu\Sigma_f v_i n_i - \Sigma_a v_i n_i + \sum_k \lambda_k C_k + \frac{1}{V_i}\left[\frac{A_i}{X_{i-1,i}}(v_{i-1}n_{i-1} - v_i n_i) + \frac{A_{i+1}}{X_{i+1,i}}(v_{i+1}n_{i+1} - v_i n_i)\right]$$

The following substitutions are made (see e.g. [219])

$$l = \frac{1}{\nu\Sigma_a} \qquad k = \frac{\nu\Sigma_f}{\Sigma_a} \qquad \Lambda = \frac{l}{k} \qquad \Rightarrow \qquad \nu = \frac{1}{\Lambda\nu\Sigma_f}$$

Here l is the average neutron lifetime (s), k is the multiplication factor (-), Λ is the neutron generation time (s). Furthermore it is assumed that $v_{i-1} \approx v_i \approx v_{i+1}$.

$$\frac{dn_i}{dt} = \frac{(1-\beta)}{\Lambda_i} n_i - \frac{1}{k_i \Lambda_i} n_i + \sum_k \lambda_k C_k + \frac{1}{\Lambda_i \nu\Sigma_{f,i}} \frac{1}{V_i} \left[\frac{A_i}{X_{i-1,i}} (n_{i-1} - n_i) + \frac{A_{i+1}}{X_{i+1,i}} (n_{i+1} - n_i) \right]$$

Substituting $\rho = (k-1)/k$, the following form is obtained:

$$\frac{dn_i}{dt} = \frac{(\rho_i - \beta)}{\Lambda_i} n_i + \sum_k \lambda_k C_k + \frac{1}{\Lambda_i \nu\Sigma_{f,i}} \frac{1}{V_i} \left[\frac{A_i}{X_{i-1,i}} (n_{i-1} - n_i) + \frac{A_{i+1}}{X_{i+1,i}} (n_{i+1} - n_i) \right]$$

As in the case of point kinetics, the reactivity is expressed in dollars: R_i (\$) = ρ_i/β, which leads to:

$$\frac{dn_i}{dt} = \frac{\beta(R_i - 1)}{\Lambda_i} n_i + \sum_k \lambda_k C_k + \frac{1}{\Lambda_i \nu\Sigma_{f,i}} \frac{1}{V_i} \left[\frac{A_i}{X_{i-1,i}} (n_{i-1} - n_i) + \frac{A_{i+1}}{X_{i+1,i}} (n_{i+1} - n_i) \right]$$

After substitutions and simple rearrangements, the equation becomes.

$$\begin{aligned} V_i \Lambda_i \nu\Sigma_{f,i} \frac{dn_i}{dt} = {} & V_i \nu\Sigma_{f,i} \beta (R_i - 1) n_i + \\ & + \frac{A_i}{X_{i,i-1}} (n_{i-1} - n_i) + \frac{A_{i+1}}{X_{i,i+1}} (n_{i+1} - n_i) + \\ & + V_i \Lambda_i \nu\Sigma_{f,i} \cdot \left(\sum_k \lambda_k C_{k,i} + S_{V,ext,i} \right) \end{aligned}$$

The external source of neutrons, $S_{V,ext,i}$ (1/m^3), was added to account for eventual presence of external sources as is done in the point kinetics model (section 9.2.1). The equation can be written as:

$$V_i \Lambda_i \nu\Sigma_{f,i} \frac{dn_i}{dt} = V_i \nu\Sigma_{f,i} \beta (R_i - 1) n_i + \frac{A_i}{X_{i,i-1}} (n_{i-1} - n_i) + \frac{A_{i+1}}{X_{i,i+1}} (n_{i+1} - n_i) + V_i \Lambda_i \nu\Sigma_{f,i} \cdot S_i$$

where the source S_i consists of the user-defined external source, $S_{V,ext,i}$ (1/m^3) and the term obtained from the decay of the delayed neutron precursors:

$$S_i = \sum_k \lambda_k C_{k,i} + S_{V,ext,i}$$

For numerical implementation, the equation is written for three cases: interior nodes ($1 < i < N$), first boundary node ($i = 1$), last boundary node ($i = N$).

- Interior nodes, $1 < i < N$

$$V_i \Lambda_i \nu\Sigma_{f,i} \frac{n_i - \overline{n_i}}{\Delta t} = V_i \nu\Sigma_{f,i} \beta (R_i - 1) n_i + S_i + \frac{A_i}{X_{i,i-1}} \left(n_{i-1} - n_i\right) + \frac{A_{i+1}}{X_{i,i+1}} \left(n_{i+1} - n_i\right)$$

with:

$$X_{i,i-1} = \frac{d_{i-1}/2}{D_{i-1}} + \frac{d_i/2}{D_i}$$

$$X_{i,i+1} = \frac{d_{i+1}/2}{D_{i+1}} + \frac{d_i/2}{D_i}$$

The $\overline{n_i}$ is the previous time step value. The equation above is very similar to the 1-D conduction equation, shown in section 5.2. The volumetric heat capacity, $\rho_i c_{pi}$, is replaced by $\Lambda_i \nu\Sigma_{f,i}$, the conduction resistance A_{ij}/R_{ij} is replaces by the diffusion resistance A_{ij}/X_{ij} and the heat source Q_i is replaced by the neutron source, S_i. The additional term $V_i \nu\Sigma_{f,i} \beta(R_i - 1)$ does not significantly change the form of the equation and the same solution scheme, based on tri-diagonal matrix, is used, as will be shown below.

- Boundary node, $i = 1$

Boundary conditions are needed for the boundary nodes. In contrast to e.g. [219], the boundary conditions are not defined from first principles. Physically, the boundary conditions represent the neutron loss in the upper and the lower part of the reactor core. These depend on the physical configuration of the reactor, i.e. the reflector thickness, the reflector material, eventual presence of control rods, etc. Thus the boundary conditions are different for different reactors. In the current model, user-defined reflection factors, F_{R1}, F_{RN} - Figure 9-25, are used. $F_{R1} = 0.0$ means no reflection (all neutrons are lost), $F_{R1} = 1.0$ means perfect reflection (no neutron loss). The values of F_{R1}, F_{RN} must be determined from 3D neutronics models (as shown in Volume 3). The diffusion term for the node 1 is given by:

$$\frac{A_1}{X_1} \left(n_1 - F_{R1} n_1\right) = \frac{A_1 D_1}{d_1/2} \left(n_1 - F_{R1} n_1\right)$$

The neutron balance for the node 1 is given by:

$$V_i \Lambda_i \nu\Sigma_{f,i} \frac{n_i - \overline{n_i}}{\Delta t} = V_i \nu\Sigma_{f,i} (R_i - 1) n_i + S_i + \frac{A_i}{X_{i,i-1}} \left(F_{R1} n_i - n_i\right) + \frac{A_{i+1}}{X_{i,i+1}} \left(n_{i+1} - n_i\right)$$

$$X_{i,i-1} = \frac{d_i/2}{D_i}$$

$$X_{i,i+1} = \frac{d_{i+1}/2}{D_{i+1}} + \frac{d_i/2}{D_i}$$

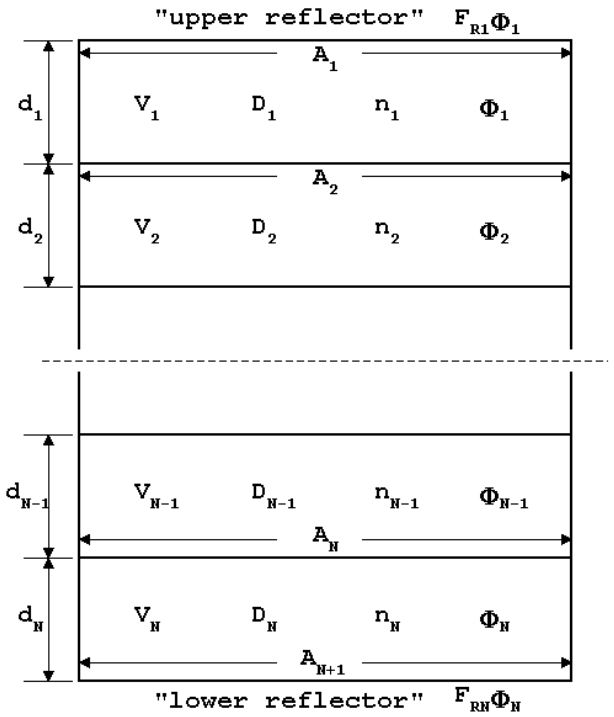


Figure 9-25 Boundary conditions for nodal kinetics

- Boundary node, $i = N$

  The diffusion term for the node N is given by:

$$\frac{A_{N+1}}{X_{N+1}}\left(n_N - F_{RN}n_N\right) = \frac{A_{N+1}D_N}{d_N/2}\left(n_N - F_{RN}n_N\right)$$

  The neutron balance for the node N is given by:

$$V_i\Lambda_i\nu\Sigma_{f,i}\frac{n_i - \overline{n_i}}{\Delta t} = V_i\nu\Sigma_{f,i}(R_i - 1)n_i + S_i + \frac{A_i}{X_{i,i-1}}\left(n_{i-1} - n_i\right) + \frac{A_{i+1}}{X_{i,i+1}}\left(F_{RN}n_i - n_i\right)$$

$$X_{i,i-1} = \frac{d_{i-1}/2}{D_{i-1}} + \frac{d_i/2}{D_i}$$

$$X_{i,i+1} = \frac{d_i/2}{D_i}$$

The above formulae present a set of N equations, with unknown neutron densities n_i. This equation set may be written shortly in a matrix form:

$$An = B$$

The matrix *A* is a tri-diagonal matrix, because the equations for internal nodes contain three unknown variables: n_{i-1}, n_i, and n_{i+1}. Therefore the matrix equation may be written as:

$$a_{i-1,i} \cdot n_{i-1} + a_{i,i} \cdot n_i + a_{i+1,i} \cdot n_{i+1} = b_i$$

- Interior nodes, $1 < i < N$

$$a_{i-1,i} = -\frac{A_i}{X_{i,i-1}}$$

$$a_{i+1,i} = -\frac{A_{i+1}}{X_{i,i+1}}$$

$$a_{i,i} = V_i \nu \Sigma_{f,i} \left(\frac{\Lambda_i}{\Delta t} - \beta (R_i - 1) \right) + \frac{A_i}{X_{i,i-1}} + \frac{A_{i+1}}{X_{i,i+1}}$$

$$b_i = V_i \nu \Sigma_{f,i} \Lambda_i \left(\frac{\overline{n_i}}{\Delta t} + S_i \right)$$

- Boundary node, $i = 1$

$$a_{i+1,i} = -\frac{A_{i+1}}{X_{i,i+1}}$$

$$a_{i,i} = V_i \nu \Sigma_{f,i} \left(\frac{\Lambda_i}{\Delta t} - \beta (R_i - 1) \right) + \frac{A_{i+1}}{X_{i,i+1}} + \frac{A_i}{X_{i,i-1}} F_{R1}$$

$$b_i = V_i \nu \Sigma_{f,i} \Lambda_i \left(\frac{\overline{n_i}}{\Delta t} + S_i \right)$$

- Boundary node, $i = N$

$$a_{i+1,i} = -\frac{A_i}{X_{i,i-1}}$$

$$a_{i,i} = V_i \nu \Sigma_{f,i} \left(\frac{\Lambda_i}{\Delta t} - \beta (R_i - 1) \right) + \frac{A_i}{X_{i,i-1}} + \frac{A_{i+1}}{X_{i,i+1}} F_{RN}$$

$$b_i = V_i \nu \Sigma_{f,i} \Lambda_i \left(\frac{\overline{n_i}}{\Delta t} + S_i \right)$$

The matrix elements are very similar to the 1-D conduction elements (section 5.2) and the matrix is solved using the same tri-diagonal matrix solver. The source due to the decay of delayed neutron precursors, which depends on the neutron densities, is solved in an implicit manner using the general SPECTRA implicit solver. The solution scheme is very similar to the one applied for the point kinetics and solution was proven by a number of tests described in Volume 3.

9.5.4 Verification - Nodal Kinetics versus Point Kinetics

This section shows that the nodal kinetics equation shown in the previous section reduces to the point kinetics equation if the node diffusion term is equal to 0.0. We start with the matrix elements written for the internal node and assume that $D_i = 0.0$, or $X_i \rightarrow \infty$. The matrix elements are:

$$a_{i-1,i} = a_{i+1,i} = 0$$

$$a_{i,i} = (V_i \nu\Sigma_{f,i})\left(\frac{\Lambda_i}{\Delta t} - \beta(R_i - 1)\right)$$

$$b_i = (V_i \nu\Sigma_{f,i})\Lambda_i\left(\frac{\bar{n}_i}{\Delta t} + S_i\right)_i$$

This means that for the (single) node *i*:

$$(V_i \nu\Sigma_{f,i})\left(\frac{\Lambda_i}{\Delta t} - \beta(R_i - 1)\right)\cdot n_i = (V_i \nu\Sigma_{f,i})\Lambda_i\left(\frac{\bar{n}_i}{\Delta t} + S_i\right)$$

This is transformed to:

$$\left(\frac{\Lambda_i}{\Delta t} - \beta(R_i - 1)\right)\cdot n_i = \Lambda_i\left(\frac{\bar{n}_i}{\Delta t} + S_i\right)$$

and

$$\frac{n_i - \bar{n}_i}{\Delta t} = \frac{\beta(R_i - 1)}{\Lambda_i}\cdot n_i + S_i$$

The source term is given by:

$$S_i = \sum_k \lambda_k C_{k,i} + S_{V,ext,i}$$

Finally, after dropping the node number *i*, the following form is obtained:

$$\frac{n - \bar{n}}{\Delta t} = \frac{\beta(R-1)}{\Lambda} + \sum_k \lambda_k C_k + S_{V,ext,i}$$

This is the point kinetics equation, described in section 9.2. Thus, it was shown that in the case when no diffusion from one node to another occurs, each node behaves exactly as a separate point kinetics model. This behavior is shown in several tests discussed in Volume 3.

9.5.5 Initialization of the Nodal Kinetics Model

Providing initial values for the nodal kinetics is more difficult that in case of point kinetics. The user must provide the initial power for every node and this would be very difficult and require a lot of iterations and external calculations. Therefore an initialization procedure is available which calculates the initial power distribution automatically. The user needs to do the following:

- Provide an initial guess for the power in every node. The recommended values here are the uniform power distribution, i.e. the same power in every node.
- Activate the initialization option (INONRK, Volume 2).
- Define the target total reactor power (POW0RK, Volume 2).

When the initialization option is used, the code automatically controls the power using a built-in PI controller. The proportional and integral constants, CPCVRK and CICVRK, were selected to provide quick convergence of power. The values may be redefined by the user as shown in Volume 2.

The code will calculate the neutron density and thus the power in every node. At the end of each time step the relative change of power is calculated from:

$$XMXNRK = Max\left(\frac{1}{n_i}\frac{dn_i}{dt}\right) = Max\left(\frac{1}{n_i}\frac{\Delta n_i}{\Delta t}\right)$$

Here Δn_i is the neutron density change ($1/m^3$) over time step and Δt is the time step size (s). Calculations are automatically terminated when the maximum value is smaller than the convergence criterion, set by XCNNRK (Volume 2).

$$Max\left(\frac{1}{n_i}\frac{\Delta n_i}{\Delta t}\right) < XCNNRK$$

Test calculations showed that XCNNRK = 10^{-7} usually provides a very good accuracy. The user may change the convergence criterion as shown in Volume 2. Applying a smaller value, e.g. 10^{-8}, will give better initial power distribution at the expense of a longer calculations.

At the end of the calculations all values, including the neutron densities, the delayed neutron precursor concentrations, the concentrations of decay heat groups, poisons such as Xe-135, etc. are stored in the Initial Conditions File (ICF). Thus, once the initialization run is finished, the ICF file from this run should be used as a starting point for the subsequent transient calculations.

Of course, if the initial conditions at different power level are needed, for example 105% power for conservative calculations, a separate initialization run should be performed for this power.

The initialization procedure is illustrated in Figure 9-26 and Figure 9-27 using the GEMINI+ model as an example case. Figure 9-26 shows the user-defined uniform power distribution applied as a starting point. Figure 9-27 shows the stationary state power distribution. More details about the GEMINI+ test are shown in Volume 3.

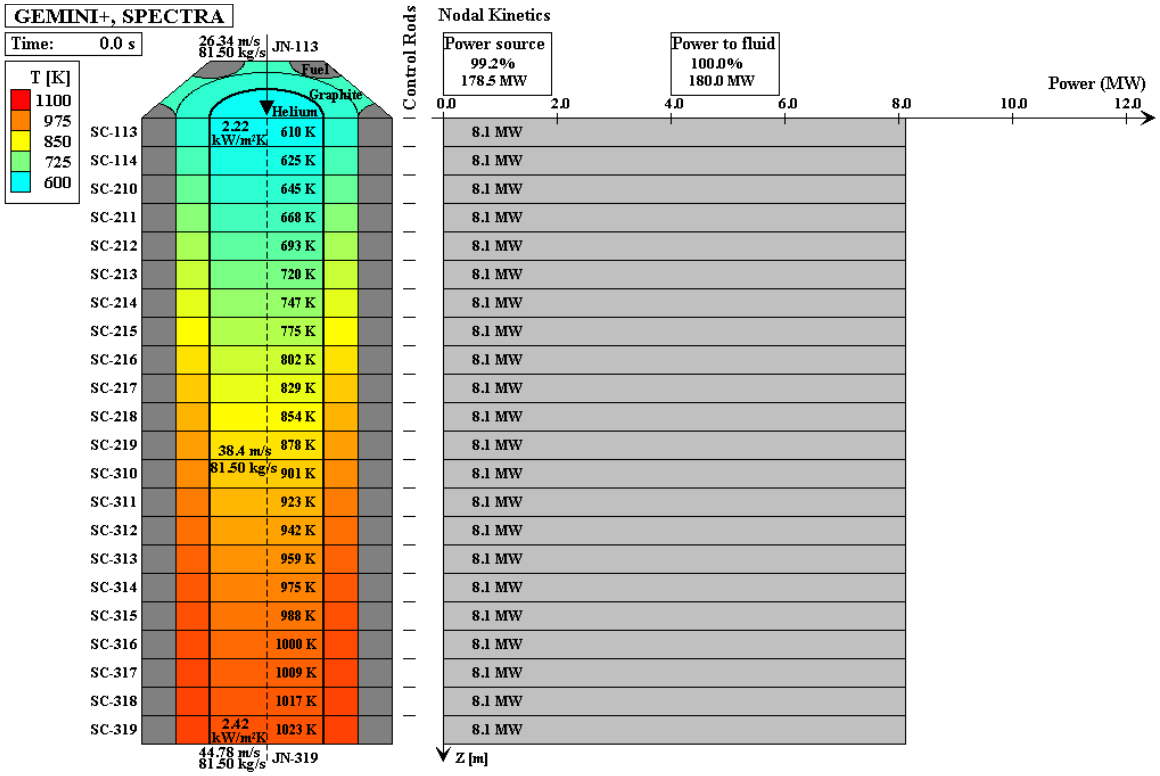


Figure 9-26 Initial power distribution, GEMINI+ test

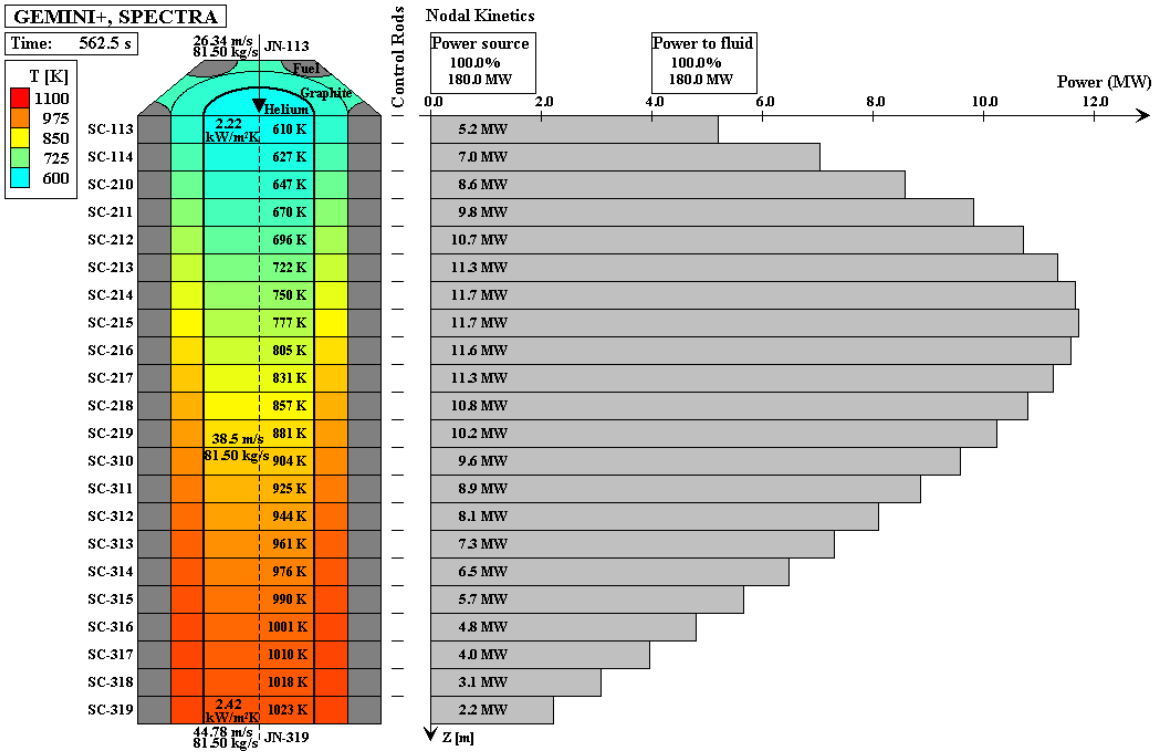


Figure 9-27 Final (stationary state) power distribution, GEMINI+ test

10 Material Oxidation Package

10.1 Introduction

A general material oxidation model is available for 1-D and 2-D Solid Heat Conductor surfaces. For every surface (left and right surface for SC, boundary cell surface for TC) the user may specify one or more oxidation reactions. If more than one reaction is specified (for example oxidation by steam and oxygen), two options are available:

- All reactions proceed simultaneously, according to their own reaction kinetics and availability of the oxidizing media (steam, oxygen). This option gives conservative results.
- Reactions proceed in sequence, defined by the user. If there is enough oxidant for the first reaction, only this reaction occurs. If not, second reaction starts, then third, etc. Note that according to [198], reaction with oxygen dominates over reactions with steam and nitrogen.

Several reactions are built-in and may be simply selected from the list. The built-in reactions are described in section 10.2. Other reactions may be built by specifying coefficients determining reaction kinetics, reaction heat, etc. The user-defined reactions are described in section 10.3.

10.2 Built-In Oxidation Reactions

The following built-in reactions are available:

- Oxidation by steam
  - Zircaloy + H_2O, model of Cathcart [95] and Urbanic-Heidrich
  - Zircaloy + H_2O, model of Urbanic-Heidrich [96]
  - Steel + H_2O, model of White [97]

- Oxidation by O_2
  - Zircaloy + O_2 model of Benjamin et al. [98]
  - Graphite + O_2 model of Roes [99]

The models are described in sections 10.2.1 through 10.2.5, below.

10.2.1 Zircaloy + H_2O Reaction, Model of Cathcart and Urbanic-Heidrich

The Zircaloy-steam reaction is:

$$Zr + 2\,H_2O \rightarrow ZrO_2 + 2\,H_2$$

The reaction proceeds according to the parabolic rate law:

$$\frac{dm^2}{dt} = K_T(T) = \begin{cases} A_1 \exp\left[-B_1/T\right] & for\ T < T_1 \\ A_2 \exp\left[-B_2/T\right] & for\ T > T_2 \end{cases}$$

where: m mass of metal reacted per unit surface area, (kg/m^2)
$K_T(T)$ temperature dependent reaction coefficient, ($kg^2/m^4/s$)

A reaction kinetics first constant, ($kg^2/m^4/s$)
B reaction kinetics second constant, (K)
T temperature, (K)

The reaction coefficients are taken from Cathcart [95] for temperatures below the crystal structure change from tetragonal to cubic (1853 K), and Urbanic-Heidrich [96] for temperatures above the structure change. This method is used following MATPRO [100]. In SPECTRA an interpolation zone is defined to ensure smooth transition between the models. The value of 1853 K is taken as the high temperature limit for the interpolation zone. The value from Prater-Courtright, 1783 K [103], is used as the low temperature limit.

The reaction kinetics constants are converted into SI units to give the kg of reacted metal (the original coefficients were specified to give weight gain, in other words, the oxygen uptake. The oxygen weight gain constants, shown in [100] (table 4-46 on page 4-210) are converted to give the mass of reacted Zr, using the ratio of molar weights of Zr and O_2

- For T < 1783.0 K, Cathcart model
  - $A_1 = (2\times16.8) \times (Mw_{Zr}/Mw_{O2})^2 = 273.0$ ($kg^2/m^4/s$)
  - $B_1 = 20{,}060.0$ (K)

- For T > 1853.0 K, Urbanic-Heidrich model
  - $A_1 = (2\times5.410) \times (Mw_{Zr}/Mw_{O2})^2 = 87.9$ ($kg^2/m^4/s$)
  - $B_1 = 16{,}610.0$ (K)

where: Mw_{Zr} molar weight of Zr, (kg/kmole), equal to 91.22 ([32], page 4-113)
Mw_{O2} molar weight of O_2, (kg/kmole), equal to 32.00 ([32], page 4-80)

The heat of reaction, Q_{ox}, is equal to 6.45×10^6 (J/kg) ([100], page 4-228, and [102], page 446). During calculations the heat generated due to reaction, q (W/m^2), is obtained from:

$$q = m \cdot Q_{ox}$$

q heat generated per unit surface area due to oxidation, (W/m^2)
Q_{ox} heat of reaction, (J/kg), expressed per kg of metal oxidized

This heat is deposited at the surface of the structure (in the first or the last cell for SC, or in the boundary cell for TC). The depth of oxidized material and the oxide layer thickness are obtained from:

$$d_{ox} = \frac{m}{\rho_{Zr}} \qquad t_{ox} = \frac{m}{\rho_{Zr}} \cdot \sigma_{ox}$$

where m mass of Zr oxidized per unit surface area, (kg/m^2)
ρ_{Zr} density of Zr, (kg/m^3), equal to 6490 ([100], page 4-230, and [102], page 446)
σ_{ox} volumetric growth upon oxidation, equal to: 1.5, ([102], page 446)

The oxide layer thickness is calculated although it is not further used in calculations. The values are only used for output; they are printed in the main output file and available for plotting. Because oxide thickness is not important for calculations, it is not obtained from separate correlations (as in the original model, where additional coefficient sets are provided for calculating the oxide layer and the alpha layer) but from the oxidized mass, m. Since σ_{ox} is just an estimation, the calculated oxide

thickness is not exactly the same as that obtained from the separate correlation. Differences are shown in Volume 3.

10.2.2 Zircaloy + H_2O Reaction, Model of Urbanic-Heidrich

The Zircaloy-steam reaction is:

$$Zr + 2\,H_2O \rightarrow ZrO_2 + 2\,H_2$$

The reaction proceeds according to the parabolic rate law:

$$\frac{dm^2}{dt} = K_T(T) = \begin{cases} A_1 \exp\left[-B_1/T\right] & for\ T < T_1 \\ A_2 \exp\left[-B_2/T\right] & for\ T > T_2 \end{cases}$$

where: m mass of metal reacted per unit surface area, (kg/m^2)
A reaction kinetics first constant, (kg^2/m^4/s)
B reaction kinetics second constant, (K)
T temperature, (K)

The reaction coefficients are taken from Urbanic-Heidrich [96]. An interpolation zone is defined to ensure smooth transition between the low- and the high-temperature formula. The value of 1853 K is taken as the high temperature limit for the interpolation zone. The value from Prater-Courtright, 1783 K [103], is used as the low temperature limit.

The reaction kinetics is calculated from (see [46], page COR-RM-43):

- For T < 1783.0 K
  - A_1 = 29.6 (kg^2/m^4/s)
  - B_1 = 16,820.0 (K)

- For T > 1853.0 K
  - A_1 = 87.9 (kg^2/m^4/s)
  - B_1 = 16,610.0 (K)

The values of heat of reaction, the volumetric growth upon oxidation, and the Zr density, are shown in the previous section. Again, the depth of oxidized material and the oxide layer thickness are calculated from:

$$d_{ox} = \frac{m}{\rho_{Zr}} \qquad t_{ox} = \frac{m}{\rho_{Zr}} \cdot \sigma_{ox}$$

The built-in model based on the Urbanic-Heidrich coefficients, is compared to the built-in model with the Cathcart coefficients (described in previous section) in Figure 10-1. Above the transition point (T≈1800 K, or 10,000/T≈5.5) Urbanic-Heidrich coefficients are used in both cases and the results are exactly the same. Below the transition point the reaction coefficients from Cathcart give somewhat lower reaction rates at low temperatures (T<1400 K, or 10,000/T>7.0), and somewhat higher rates at high temperatures.

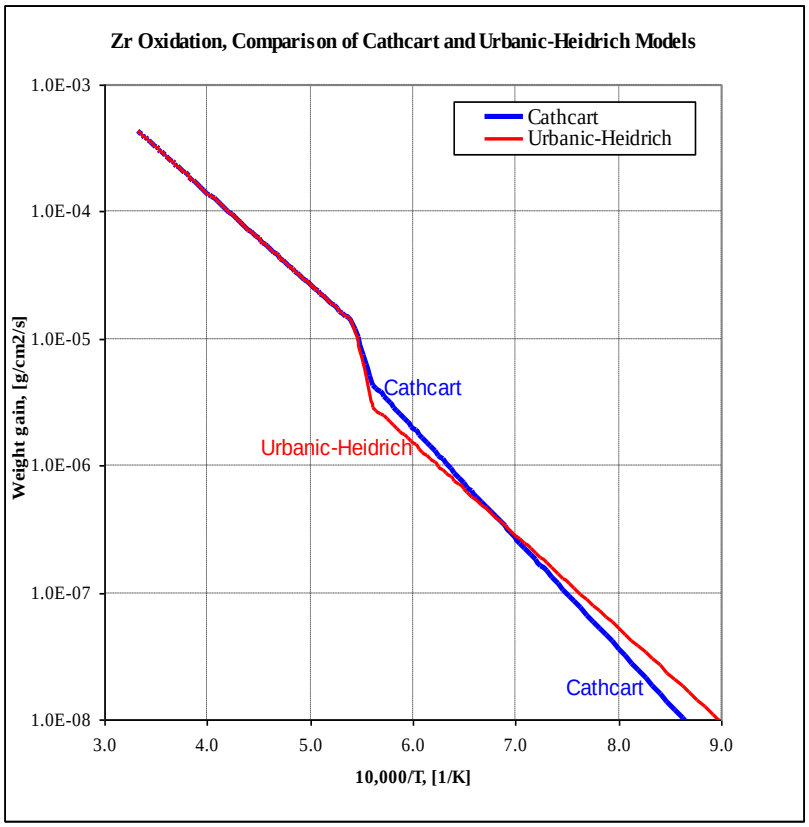


Figure 10-1 Comparison of the built-in Zr-H_2O oxidation models.

10.2.3 Steel + H_2O Reaction, Model of White

Oxidation of iron and chromium is considered. Oxidation of other elements present in steel, such as nickel and carbon, are neglected. Therefore the steel-steam reactions are:

$$Fe + H_2O \rightarrow FeO + H_2$$
$$2\,Cr + 3\,H_2O \rightarrow Cr_2O_3 + 3\,H_2$$

The reaction proceeds according to the parabolic rate law:

$$\frac{dm^2}{dt} = K_T(T) = A \exp\left[-B/T\right]$$

where: m mass of metal reacted per unit surface area, (kg/m^2)
A reaction kinetics first constant, ($kg^2/m^4/s$)
B reaction kinetics second constant, (K)
T temperature, (K)

The reaction coefficients are taken from White [97]. The reaction kinetics constants are converted into SI units to give the kg of reacted metal (the original coefficients were specified to give weight gain, in other words, the oxygen uptake. The oxygen weight gain constants, shown in [100] (table

6-1 on page 6-8) are converted to give the mass of reacted steel, using the molar weights of steel and O_2.

- $A = (2\times1.2E+8) \times (Mw_{St}/Mw_O)^2 = 2.51E+9$ (kg^2/m^4/s)
- $B = 42{,}428.0$ (K)

where: Mw_O molar weight of O, (kg/kmole), equal to ½·32.00 = 16.00 ([32], page 4-80)
Mw_{St} molar weight of steel per unit mole of steam reacted, (kg/kmole), calculated from:

$$Mw_{St} = \frac{x_{Fe}Mw_{Fe} + 2/3 \cdot x_{Cr}Mw_{Cr}}{x_{Fe} + x_{Cr}}$$

Mw_{Fe} molar weight of Fe, (kg/kmole), equal to 55.85 ([32], page 4-64)
Mw_{Cr} molar weight of Cr, (kg/kmole), equal to 52.00 ([32], page 4-53)
x_{Fe} molar fraction of Fe in steel, equal to 1.0 – 0.30 = 0.70 ([32], page 12-137)
x_{Cr} molar fraction of Cr in steel, equal to 0.17 ([32], page 12-137)

After substituting the data the value of Mw_{St} is obtained as: $\boldsymbol{Mw_{St} = 51.7}$. Note that the steel composition of the stainless steel SS-303 was taken for calculations. The conversion factor would be somewhat different for different types of steel. For example in case of carbon steel practically only Fe is reacting. Therefore for carbon steel

- $A = (2\times1.2\times10^8) \times (Mw_{Fe}/Mw_O)^2 = 2.92\times10^9$ (kg^2/m^4/s)

The effective reaction stoichiometry, i.e. number of moles of steam consumed per one mole of steel reacted and number of moles of hydrogen generated per one mole of steel reacted, are calculated using the molar fractions and the stoichiometry of both Fe and Cr reactions:

$$RR = \frac{x_{Fe} + x_{Cr} \cdot 3/2}{x_{Fe} + x_{Cr}}$$

The value is equal to 1.10. Therefore the effective reaction stoichiometry is:

$$St + 1.1\,H_2O \rightarrow StO_{1.1} + 1.1\,H_2$$

The heat of reaction, Q_{ox}, is taken from [100], page 6-8. The linear power of reaction (in W/m), presented in [100], is: $4.85\times10^6\times D\times(dm/dt)$. This is converted into W/kg of steel reacted dividing by the circumference, πD, and (Mw_{St}/Mw_{O2})

$$Q_{ox} = \frac{4.85\times10^6}{\pi \cdot (51.7/16.0)} = 0.477\times10^6$$

The volumetric growth upon oxidation, σ_{ox} , is estimated using the correlation for oxide thickness, shown in [100] (table 6-1). The oxide growth coefficient is equal to coefficient A_{ox}=(2×300.0) = 600.0 m^2/s. The thickness of consumed steel is $A^{½} / \rho_{St}$ = (2.51E+6) ^{½} / 8020 = 6.25. The thickness of oxide is $A_{ox}^{½}$ =24.5. The ratio is: σ_{ox} = 24.5 / 6.25 = 3.92. The depth of oxidized material and the oxide thickness are obtained from:

$$d_{ox} = \frac{m}{\rho_{St}} \qquad t_{ox} = \frac{m}{\rho_{St}} \cdot \sigma_{ox}$$

where m mass of steel oxidized per unit surface area, (kg/m^2)
ρ_{St} density of steel, (kg/m^3), equal to 8020 ([32], page 12-136)
σ_{ox} volumetric growth upon oxidation, equal to 3.92

10.2.4 Zircaloy + O_2 Reaction, Model of Benjamin

The Zircaloy-oxygen reaction is:

$$Zr + O_2 \rightarrow ZrO_2$$

The reaction proceeds according to the parabolic rate law:

$$\frac{dm^2}{dt} = K_T(T) = A \exp\left[-B/T\right]$$

where: m mass of metal reacted per unit surface area, (kg/m^2)
A reaction kinetics first constant, ($kg^2/m^4/s$)
B reaction kinetics second constant, (K)
T temperature, (K)

The reaction coefficients are taken from Benjamin et al.[104], see also [46], equation 2.4.13, page COR-RM-44):

- A = 50.4 ($kg^2/m^4/s$)
- B = 14,630.0 (K)

The remaining parameters of this reaction are:

Q_{ox}, heat of reaction, (J/kg), equal to 1.20×10^7 ([46], page COR-RM-43)
σ_{ox} volumetric growth upon oxidation, equal to σ_{ox} = 1.5 ([102], page 446)
ρ_{Zr} density of Zr, (kg/m^3), equal to 6490 ([100], page 4-230, and [102], page 446)
m mass of Zr oxidized per unit surface area, (kg/m^2)

The depth of oxidized material and the oxide layer thickness are calculated from:

$$d_{ox} = \frac{m}{\rho_{Zr}} \qquad t_{ox} = \frac{m}{\rho_{Zr}} \cdot \sigma_{ox}$$

A comparison of Zr oxidation by steam and oxygen is shown in Figure 10-2. The steam reaction is calculated from the Cathcart model, while the oxygen reaction is calculated from the Benjamin model. The oxygen reaction is roughly twice faster than the steam reaction.

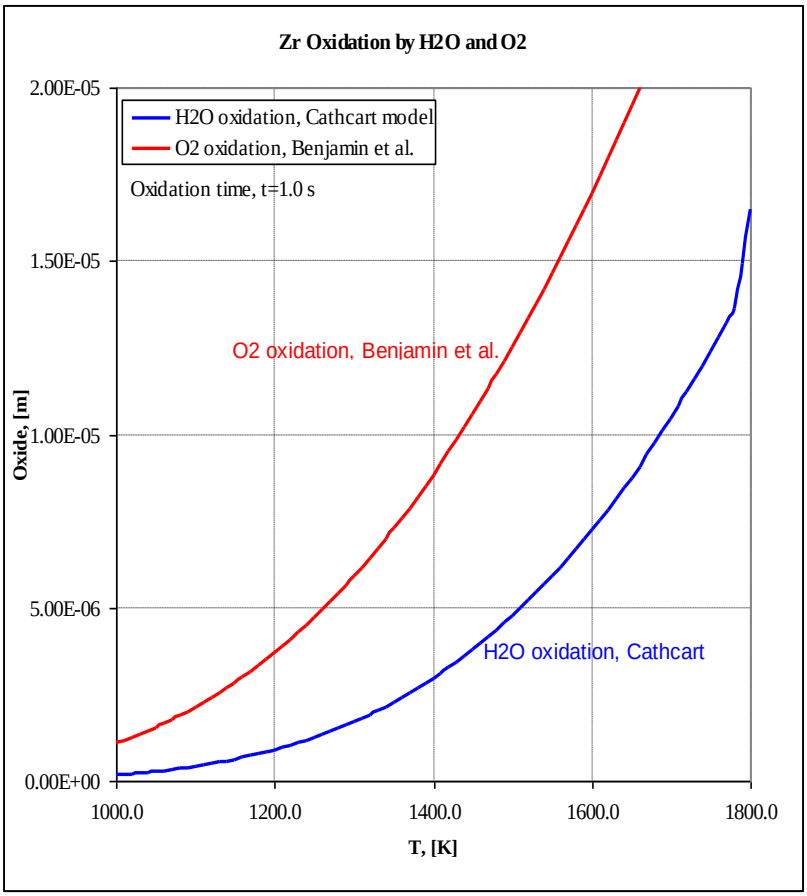


Figure 10-2 Comparison Zr-oxidation by O_2 and H_2O.

10.2.5 Graphite + O_2 Reaction, Model of Roes

The graphite oxidation reaction model is based on Roes work [99]. The reaction is:

$$C + O_2 \rightarrow CO_2$$

The reaction proceeds according to the following formula:

$$\frac{dm}{dt} = \left(\frac{1}{A \exp[-B/T]} + \frac{1}{C v^D T^E} \right)^{-1} \left(\frac{p}{p_{ref}} \right)$$

where: m mass of material reacted per unit surface area, (kg/m^2)
$A,B,..$ reaction constants, (kg^2/m^4/s)
T temperature, (K)
v gas velocity near the surface, (m/s)

p pressure, (Pa)
p_{ref} reference pressure (1.0 bar)

The reaction coefficients are taken from [99] page 62. The coefficients are

- A = 7.2E+9 (mg/cm^2/hr)
- B = 16,140.0 (K)
- C = 770.0 ((mg/cm^2/hr) / (m/s)^D / K^E)
- D = 0.65
- E = 0.34

The coefficients are not in SI units and they define oxygen uptake. Therefore they are converted to SI, to give mass of reacted graphite, using the conversion factor, equal to:
1.0E-6 (kg/mg) / 1.0E-4 (m^2/cm^2) / 3600 (s/hr) = 2.78E–6,
and the molar weight ratio, (Mw_C/Mw_{O2})

Mw_C molar weight of C, (kg/kmole), equal to 12.01 ([32], page 4-50)
Mw_{O2} molar weight of O_2, (kg/kmole), equal to 32.00 ([32], page 4-80)

The converted coefficients are:

- A = 7500 (kg/m^2/s)
- B = 16,140.0 (K)
- C =8.03×10^{-4} ((kg/m^2/s) / (m/s)^D / K^E)
- D = 0.65
- E = 0.34

The remaining parameters of this reaction are:

Q_{ox}, heat of reaction, (J/kg), equal to 3.935×10^8 J/kmol ([105], section 2) / Mw_C = 3.28×10^7 (J/kg)
σ_{ox} volumetric growth upon oxidation, assumed to be equal to 0.01. This means the oxidized layer is 1% of the initial layer. In fact all material is gone since it forms gaseous CO_2. The value of σ_{ox} must be positive; therefore a small number is used.
ρ_C density of graphite, (kg/m^3), equal to 2250 ([32], page 4-50)

The depth of oxidized material and the oxide layer thickness are calculated from:

$$d_{ox} = \frac{m}{\rho_C} \qquad t_{ox} = \frac{m}{\rho_C} \cdot \sigma_{ox}$$

Results of the model are illustrated in Figure 10-3. The reaction is strongly affected by the gas velocity. Since there is no build-up of the oxide layer, the reaction rate is linear, i.e. it does not slow down in time.

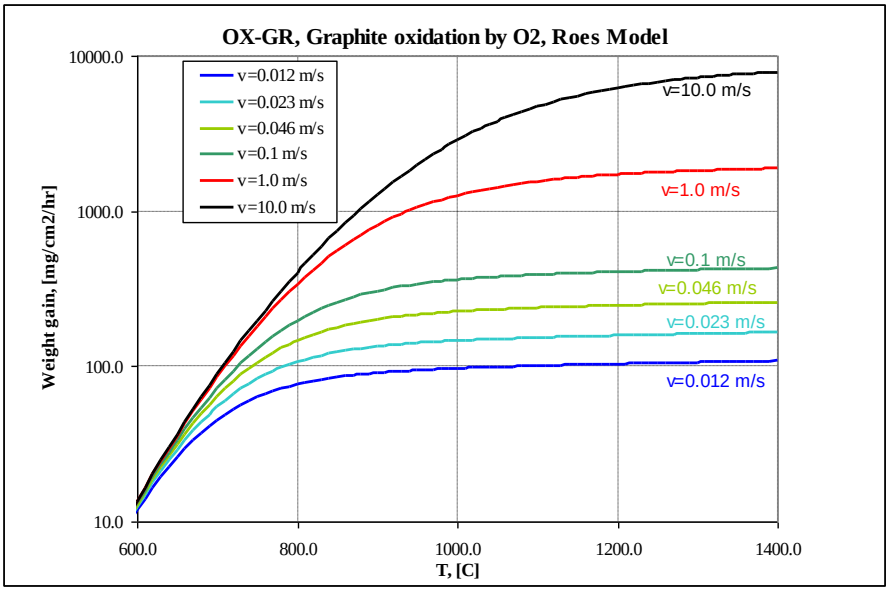


Figure 10-3 Graphite oxidation, Roes model

One concern for safety analysis is the ignition of the graphite that results in a flame. Theoretical analysis and experiments have been performed to address the phenomena for cylindrical graphite rods with steady gas flow towards the cylindrical surface [217]. This geometry is very convenient because the gas velocity field to a cylindrical surface can be determined analytically, thus enabling much theoretical analysis. Ignition was observed to be independent of the rod diameter in the diameter range tested of 0.5 to 2 cm, but dependent on the graphite surface temperature and flow velocities. There is considerable scatter in the data, and therefore [218] suggested a simple linear dependency of the ignition temperature with the logarithm of the free stream velocity gradient defined as 4V/d, where V is the free stream velocity and d is the rod diameter. From Fig. 8a of [217], the ignition temperature in air is 1400 K for 4V/d = 50 sec^{-1}, and 1975 K for 4V/d = 1000 sec^{-1}. The ignition temperature decreases as the concentration of oxygen increases. For nearly pure oxygen, from Fig. 8b of Makino and Law [217], the ignition temperature varies from 1350 K at 4V/d = 100 sec^{-1}, to 1700 K at 4V/d = 1000 sec^{-1}. Those data points were plotted in Figure 10-4. The following correlation was developed based on the data:

$$T_{ign} = (0.659 - 0.269 \cdot X_{O2}) \cdot (4V / d) + (1385 - 75 \cdot X_{O2})$$

Here X_{O2} is the oxygen volume fraction, V is the gas velocity, d is the diameter (taken as hydraulic diameter for forced convection). A limit on oxygen concentration is introduced as X_{lim} = 0.001. Below that limit, the ignition temperature is set to a large value (10^4 K). An interpolation zone is defined for $X_{lim} < X_{O2} < 2\,X_{lim}$, to ensure smooth transition.

If the Roes model is used, the code calculates graphite ignition based on the above formula. Calculations are performed only for 1-D Solid Conductors. The results may be plotted, including the ignition temperature (K) and the time (s) when it is reached - Volume 2, SC plot parameters 35 - 38. It should be noted that no calculation of flame is performed. The model is mainly intended as a user convenience to flag potentially dangerous conditions that may require separate analysis.

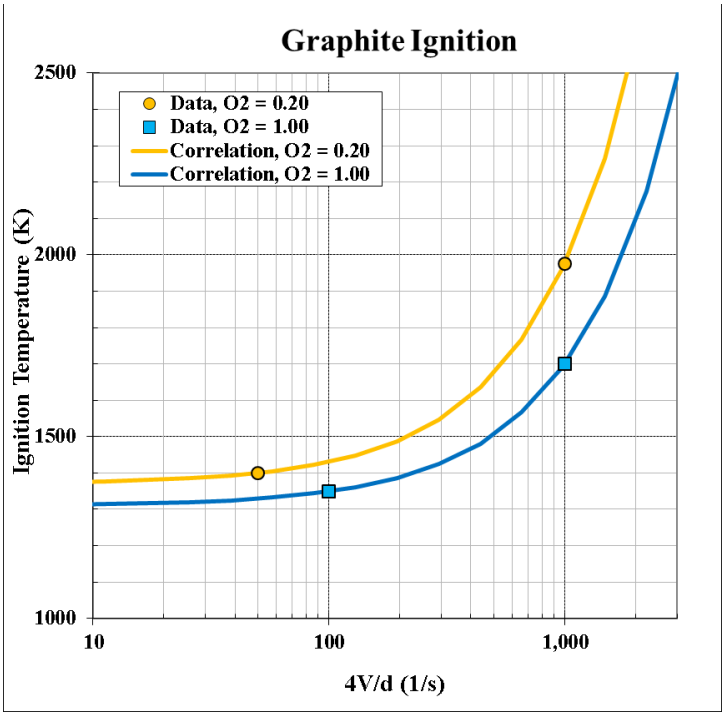


Figure 10-4 Graphite ignition temperature

10.3 User-Defined Oxidation Reactions

Any reaction may be modeled through a user-defined oxidation model. All coefficients for such reactions must be defined by the user. The oxidation reaction is assumed to have the form:

$$Mt + n_1\ Gs_1 \rightarrow MtOx + n_2\ Gs_2 + n_3\ Gs_3$$

Mt	material to be oxidized
MtOx	material oxide
Gs_1	gas 1 (oxidizing gas, for example steam, oxygen
Gs_2	gas 2 (product of reaction, for example hydrogen)
Gs_3	gas 3 (optional second product of reaction)
n_1	reaction ratio 1 - moles of gas Gs_1 per one mole of material to be oxidized
n_2	reaction ratio 2 - moles of gas Gs_2 per one mole of material to be oxidized
n_3	reaction ratio 3 - moles of gas Gs_3 per one mole of material to be oxidized

The oxidation reaction is assumed to have the following general form:

$$\frac{dm^x}{dt} = \left(\frac{1}{K_T(T) \cdot C_T(T, p_i)} + \frac{1}{K_v(v,T)} \right)^{-1} \cdot K_p(p_1) \cdot M(B,...)$$

m mass of material to be oxidized per unit surface area (kg/m^2)
x exponent, (-) (user-defined, x=2 for a parabolic oxidation rate)
t time (s)
K reaction rate ((kg/m^2)^x/s)
T temperature, (K)
v velocity of the oxidizing gas, (m/s)
p_i partial pressure of the gas i, (Pa)
$M(B,...)$ multiplier to account for the effect of burn-off

The individual terms, $K_T(T)$, $C_T(T, p_i)$, $K_v(v,T)$, $K_p(p)$, $M(B,...)$ are described below.

10.3.1 Temperature-Dependent Coefficient

The temperature-dependent coefficient is given by:

$$K_T(T) = \begin{cases} A_1 \exp\left[-B_1/T\right] & for: \quad T < T_1 \\ A_2 \exp\left[-B_2/T\right] & for: \quad T > T_2 \end{cases}$$

A_1 coefficient for reaction rate at low temperatures, ((kg/m^2)^x/s)
B_1 coefficient for reaction rate at low temperatures, (K)
T_1 maximum temperature limit to apply the low temperature equation, (K)
A_2 coefficient for reaction rate at high temperatures, ((kg/m^2)^x/s)
B_2 coefficient for reaction rate at high temperatures, (K)
T_2 minimum temperature limit to apply the high temperature kinetics equation, (K)
T current temperature, (K)

Linear interpolation is applied within the range $T_1<T<T_2$. Minimum and maximum temperature limits, T_{min}, T_{max}, are applied. The oxidation rate is set to zero for $T<T_{min}$. A linear interpolation from zero to a full value is performed between T_{min} and T_{min}+10.0 K. If $T>T_{max}$ then T_{max} is used to evaluate $K_T(T)$, as well as $K_v(v,T)$.

Breakaway
Additionally a post-breakaway model may be applied, where different coefficients are used, as well as a different value of the exponent x. The breakaway reaction is observed in case of air oxidation of Zircaloy, for example in case of loss of water in Spent Fuel Pool. In the post-breakaway regime the temperature-dependent coefficient, $K_T(T)$ is given by:

$$K_T(T) = \begin{cases} A_3 \exp\left[-B_3/T\right] & for: \quad T < T_3 \\ A_4 \exp\left[-B_4/T\right] & for: \quad T > T_4 \end{cases}$$

A_3 post-breakaway coefficient for reaction rate at low temperatures, ((kg/m^2)^x/s)
B_3 post-breakaway coefficient for reaction rate at low temperatures, (K)
T_3 maximum temperature limit to apply the low temperature equation, (K)
A_4 post-breakaway coefficient for reaction rate at high temperatures, ((kg/m^2)^x/s)
B_4 post-breakaway coefficient for reaction rate at high temperatures, (K)
T_4 minimum temperature limit to apply the high temperature kinetics equation, (K)

Linear interpolation is applied within the range $T_3<T<T_4$. The point of breakaway is calculated in one of two ways, using a critical time to breakaway or a critical oxidation depth at breakaway, both temperature-dependent.

- Option 1: critical time to breakaway, $\tau(T)$
  A time to breakaway function is tabulated versus temperature, $\tau(T)$. For a transient conditions a cumulative damage approach is used:

$$CD = \int_0^t \frac{dt'}{\tau(T)}$$

  Breakaway occurs when $CD = 1.0$. For isothermal conditions the integral gives: $CF = (1/\tau)\int dt' = t/\tau$. Therefore $t = \tau$; time to breakaway is correct. For $CD < 1.0$ pre-breakaway reaction coefficients are used. For $CD > 1.0$ + BINTOX (default=0.5), post-breakaway reaction coefficients are used. For $1.0 < CD < 1.0$ + BINTOX, a linear interpolation is performed between the pre- and the post-breakaway reaction kinetics.

- Option 2: critical oxidation depth at breakaway, $d_{OX}(T)$
  A critical oxidation depth at breakaway is tabulated versus temperature, $\delta_1(T)$. This is the upper limit of the pre-breakaway reaction. For $\delta < \delta_1(T)$ pre-breakaway reaction coefficients are used. For $\delta > \delta_1(T)\times(1.0 + \text{BINTOX})$, post-breakaway reaction coefficients are used. For $\delta_1(T) < \delta < \delta_1(T)\times(1.0 + \text{BINTOX})$, a linear interpolation is performed between the pre- and the post-breakaway reaction kinetics.

Literature shows that nitrogen plays an important role in air oxidation of zircaloy. Zirconium nitride (ZrN) increases porosity and breaks up coherent microstructure of the oxide scale and possibly causes breakaway [199]. In order to model the effect of nitrogen, a catalyst option may be used. The catalyst indicator (ICATOX) should be set to the gas number of nitrogen (4) and the limit of catalyst concentration (XCATOX) to a small positive number. If this is done, breakaway will only be possible when the nitrogen concentration exceeds the limit. The result will depend on which option is used for breakaway calculation.

- If critical time to breakaway, $\tau(T)$, is used, the "cumulative damage" function CD is calculated only when the catalyst (nitrogen) is present. The results are shown in Figure 10-5. Calculated values are compared to the measured data, KIT TG experiments [200]. A good agreement is observed.

- If critical oxidation depth, $d_{OX}(T)$, is used to calculate breakaway, the oxidation depth and the presence of catalyst are checked to determine occurrence of breakaway. If the depth satisfies the breakaway criterion, breakaway is assumed to occur as soon as the catalyst (nitrogen) becomes available. Results are shown in Figure 10-6.

The experimental data show that the real behavior is close to the values obtained with critical time to breakaway. Therefore this option should be used when the catalyst model is used. Use of breakaway option 2 with simultaneous use of the catalyst model is not allowed (see Volume 2, input parameters I1OX, ICATOX).

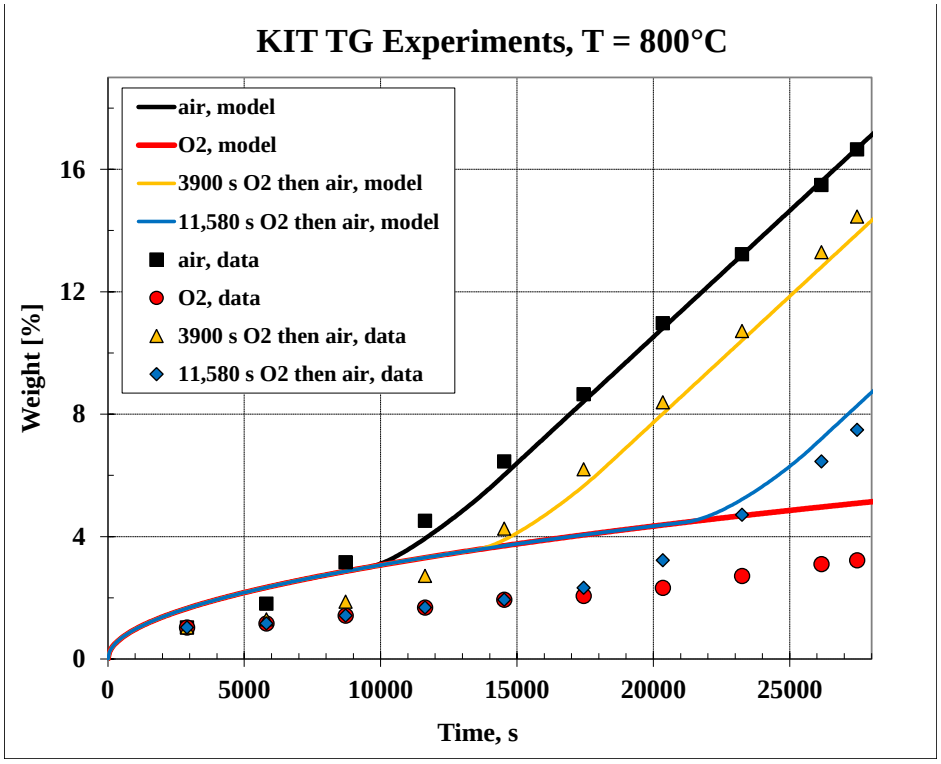


Figure 10-5 KIT TG tests, 800ºC, critical time to breakaway option

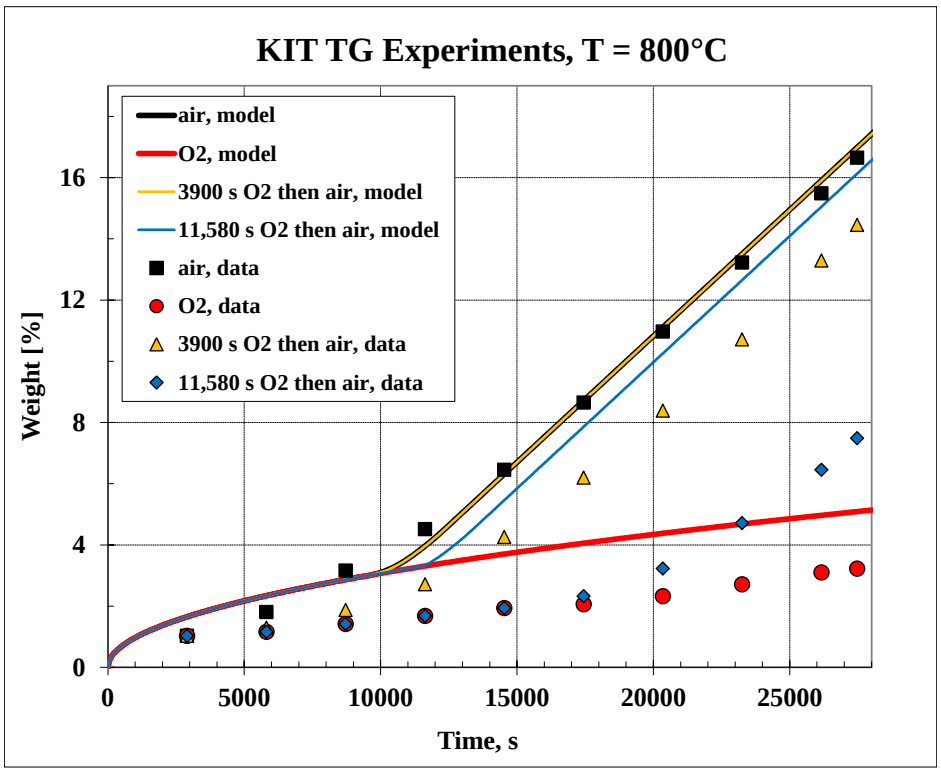


Figure 10-6 KIT TG tests, 800ºC, critical oxidation depth option

10.3.2 Correction to Temperature-Dependent Coefficient

The correction to temperature-dependent coefficient is given by:

$$C_T(T, p_1, p_2, p_3) = \frac{1}{1 + A_C \exp[B_C / T] \cdot p_1^{X_1} \cdot p_2^{X_2} \cdot p_3^{X_3}}$$

C_T	correction to the temperature-dependent reaction rate, (-),
A_C	first coefficient, (-),
B_C	second coefficient, (K),
p_1	partial pressure of the gas Gs_1 (oxidizing gas), (Pa),
p_2	partial pressure of the gas Gs_2 (first reaction product), (Pa),
p_3	partial pressure of the gas Gs_3 (second reaction product), (Pa),
X_1	exponent for the gas Gs_1
X_2	exponent for the gas Gs_2
X_3	exponent for the gas Gs_3

10.3.3 Velocity-Dependent Coefficient

The velocity-dependent coefficient is given by:

$$K_v(v,T) = C \cdot v^D \cdot T^E$$

v	gas velocity, (m/s)
T	temperature, (K)
C	user-defined coefficient determining the reaction rate, ($kg^x\ m^{-2x-D}\ s^{-1+D}\ K^{-E}$)
D	user-defined exponent, (-)
E	user-defined exponent, (-)

Minimum and maximum velocity limits, v_{min}, v_{max}, are applied. If $v<v_{min}$ then v_{min} is used to evaluate $K_v(v,T)$. If $v>v_{max}$ then the v_{max} is used to evaluate $K_v(v,T)$.

10.3.4 Pressure-Dependent Coefficient

The pressure-dependent coefficient is given by:

$$K_p(p_1) = \left(\frac{p_1}{p_{ref}} \right)^F$$

p_1	partial pressure of the oxidizing gas (Gs_1), (Pa)
p_{ref}	user-defined reference pressure, (Pa)
F	user-defined exponent, (-)

Minimum and maximum pressure limits, p_{min}, p_{max}, are applied. The oxidation rate is set to zero for $p_1<p_{min}$. A linear interpolation from zero to a full value is performed between p_{min} and $p_{min}\times 2.0$. If $p_1>p_{max}$ then p_{max} is used to evaluate $K_p(p_1)$.

10.3.5 Multiplier to Account for the Effect of Burn-off

Most oxidation reactions slow down when material is oxidized and a protective layer of oxide is built. This fact is taken into account in correlations by using exponent x (typically taken as 2 - parabolic reaction rate). For some reactions, for example graphite oxidation: $C+O_2 \rightarrow CO_2$ or $C+H_2O \rightarrow H_2+CO$, there is no oxide build-up. The reaction speeds up in time because of an increased surface area due to micro cavities created when graphite is burned off. This fact can be taken into account by using the burn-off function $M(B,...)$. No correlation is provided here, $M(B,...)$ must be defined by the user, via a Tabular Function, TF(d_{ox}) or a Control Function, CF. If TF is used, the argument is always the depth of oxidized material, d_{ox}.

10.3.6 Other Parameters

The following oxidation parameters must be defined by the user:

Q_{ox},	heat of reaction, (J/kg)
σ_{ox}	volumetric growth upon oxidation,
ρ_{Mt}	density of the oxidized material, (kg/m^3)
Mw_{Mt}	molar weight of the oxidized material, (kg/kmole)

The depth of oxidized material and the oxide layer thickness are calculated from:

$$d_{ox} = \frac{m}{\rho_{Mt}} \qquad t_{ox} = \frac{m}{\rho_{Mt}} \cdot \sigma_{ox}$$

It is seen that all the built-in reactions, described in sections 10.2.1 through 10.2.5, can be modeled by specifying appropriate coefficients for a user-defined reaction. In particular input records defining the Cathcart and Urbanic-Heidrich model for Zr-steam reaction, as well as Roes model for graphite-O_2 reaction, are shown in Volume 2. Values of all coefficients of the built-in reactions are shown in Table 10-1 and Table 10-2.

As an example, a model for Mn-Cr steel oxidation by O_2 is built with the user-defined model. Reaction kinetics data is taken from Nanni et al. [106]:

- T = 1273 K $k = 6.5\times10^{-9}$ g^2(O_2)/cm^4/s
- T = 1473 K $k = 8.2\times10^{-7}$ g^2(O_2)/cm^4/s

The reaction rate coefficients are converted to SI units using the conversion factor of 100.0 (kg^2/m^4)/(g^2/cm^4). For SPECTRA input the reaction rate needs to be expressed to give kg of steel reacted, rather than kg of oxygen uptake. Therefore the coefficients need to be multiplied by the ratio of molar weights of steel (average molar weight of 51.7 is used) and 1/2 O_2 squared: $(51.7/16.0)^2$. The conversion factor to SI is therefore: $100.0 \times (Mw_{Fe}/Mw_O)^2 = 1044.1$. Therefore:

- T = 1273 K $k = 6.8\times10^{-6}$ kg^2(steel)/m^4/s
- T = 1473 K $k = 8.6\times10^{-4}$ kg^2(steel)/m^4/s

These two data points are shown in Figure 10-7 with square markers. The reaction coefficients, A, B, were developed to fit these data points. The coefficients are: $A = 2.3\times10^{10}$, $B = 45350.0$. The reaction kinetics is therefore given by:

$$K_T(T) = 2.3\times10^{10}\ \exp\left[-45350\ /\ T\right]$$

The reaction kinetics calculated from the above equation is plotted in Figure 10-7 as a solid line. The reaction heat is equal to the enthalpy of creation of FeO, equal to 65 kcal/mol ([32], page 5-36), multiplied by the conversion factor of 4.184×10^6 (J/kmol)/(kcal/mol), divided by the average molar weight of steel, equal to 51.7. This gives 5.26×10^6 (J/kg).

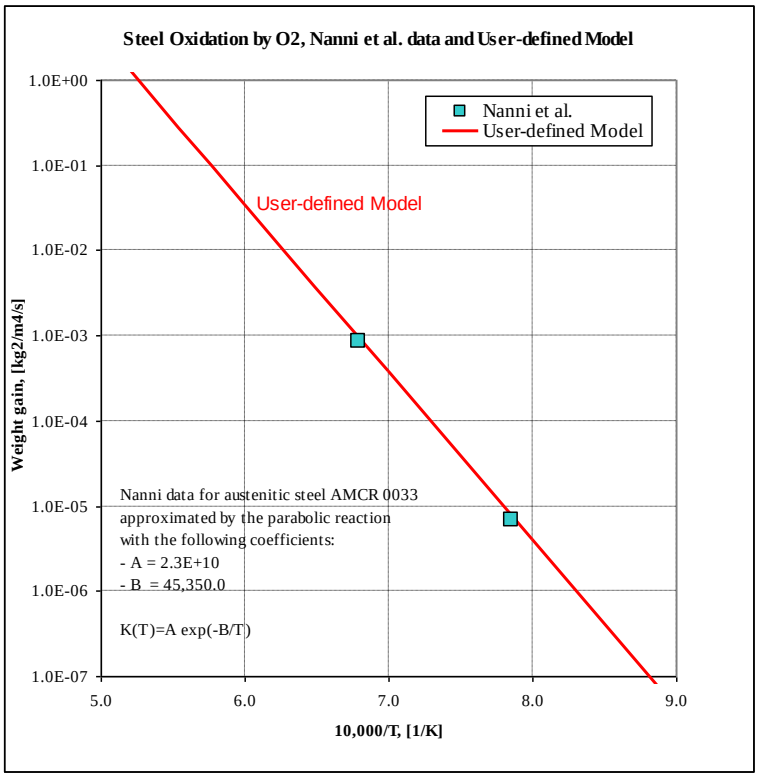


Figure 10-7 Steel oxidation by O_2 - Nanni et al. data and curve fit.

An input defining this reaction is shown in Volume 2 and Volume 3. Comparison of the calculated oxide layer thickness with a graph showing the measured values, taken from Nanni et al. [106], is given in Volume 3. Comparison of steel oxidation by steam and oxygen is shown in Figure 10-8. The steam reaction is calculated from the White model, while the oxygen reaction from a user-defined model, with reaction coefficients shown above. The oxygen reaction is roughly 1.3 times faster than the steam reaction.

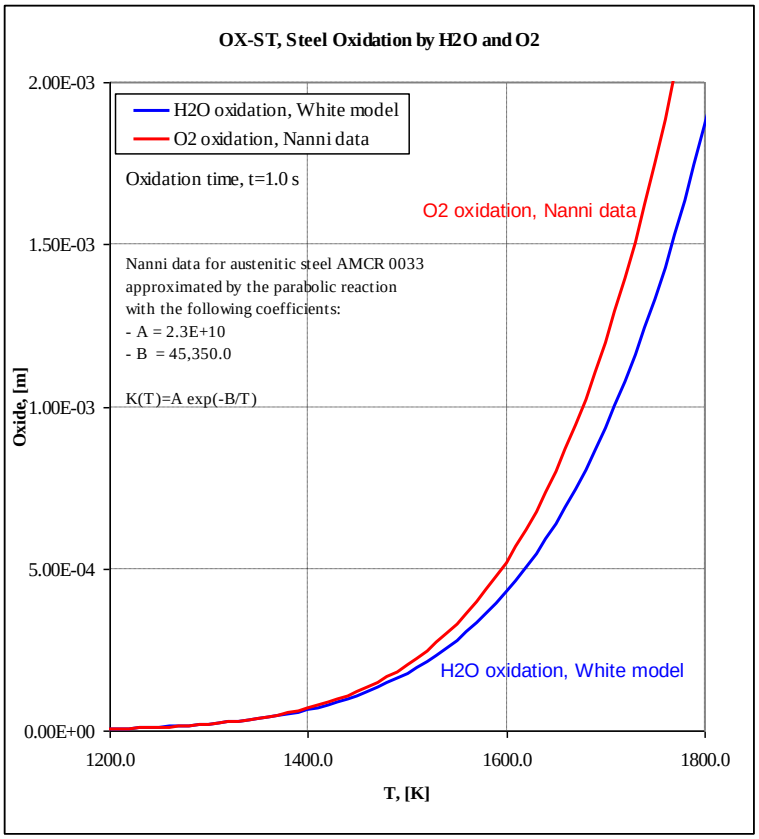


Figure 10-8 Comparison steel oxidation by O_2 and H_2O

10.4 Summary

Five built-in reactions and a general model, which allows the user to define any oxidation reaction, are available in SPECTRA. Several reactions may be activated for a given surface. For example, in case of Zr cladding, oxidation by O_2 and steam may be specified. If both gases are available, both oxidation reactions will proceed. The kinetics is calculated for both reactions using the oxide thickness at the beginning of a time step. The oxide thickness at the end of the time step is obtained by summing the oxide growth from the two reactions.

The arguments (T, v, p) for oxidation reactions are limited by minimum and maximum values. For temperatures below the minimum T the reaction is not calculated (reaction rate is set to zero). For all other limits the reaction rate is calculated by using the limiting value. All limits are shown in Table 10-1 and Table 10-2.

Table 10-1 Built-in oxidation reaction data for steam oxidation

Coefficient	Reaction		
	$Zr+2H_2O \rightarrow ZrO_2+2H_2$, Cathcart (low T), Urbanic-Hedrich (high T)	$Zr+2H_2O \rightarrow ZrO_2+2H_2$, Urbanic-Hedrich	$Fe+H_2O \rightarrow FeO+H_2$, $2Cr+3H_2O \rightarrow Cr_2O_3+3H_2$, White
n_1	2.0	2.0	1.1
n_2	2.0	2.0	1.1
x	2.0	2.0	2.0
Q_{ox}	6.45×10^6	6.45×10^6	0.477×10^6
σ_{ox}	1.5	1.5	3.92
ρ_{ox}	$\rho(Zr)$ = 6490.0	$\rho(Zr)$ = 6490.0	$\rho(Fe)$ = 8020.0
M_{ox}	$M(Zr)$ = 91.22	$M(Zr)$ = 91.22	$M(St)$ = 51.70
A_1	$2\times16.8\times(M_{Zr}/M_{O2})^2$ = 273.0	29.6	2.51×10^9
B_1	20,060.0	16,820.0	42,428.0
T_1	1783.0	1783.0	full temperature range
A_2	$2\times5.41\times(M_{Zr}/M_{O2})^2$ = 87.9	87.9	-
B_2	16,610.0	16,610.0	
T_2	1853.0	1853.0	
C	0.0	0.0	0.0
D	-	-	-
E	-	-	-
F	0.0	0.0	0.0
p_{ref}	-	-	-
T_{min} / T_{max}	600.0 / 10,000.0	600.0 / 10,000.0	600.0 / 10,000.0

Table 10-2 Built-in oxidation reaction data for oxidation by O_2.

Coefficient	Reaction	
	$Zr+O_2 \rightarrow ZrO_2$, Benjamin et al.	$C+O_2 \rightarrow CO_2$, Roes
n_1	1.0	1.0
n_2	0.0	1.0
x	2.0	1.0
Q_{ox}	12.1×10^6	32.8×10^6
σ_{ox}	1.5	0.01
ρ_{ox}	$\rho(Zr)$ = 6490.0	$\rho(C)$ = 2250.0
M_{ox}	$M(Zr)$ = 91.22	$M(C)$ = 12.01
A_1	50.4	7.5×10^3
B_1	14,630.0	16,140.0
T_1	full temperature range	full temperature range
A_2	-	-
B_2		
T_2		
C	0.0	8.03×10^{-4}
D	-	0.65
E	-	0.34
F	0.0	1.0
p_{ref}	-	1.0×10^5
T_{min} / T_{max}	600.0 / 10,000.0	600.0 / 10,000.0
v_{min} / v_{max}	-	0.01 / 1000.0
p_{min} / p_{max}	-	1.0×10^1 / 1.0×10^7

Each model has an internal limit, which stops oxidation when the oxidant volumetric fraction falls below a certain value (oxidant starvation limit). An interpolation is provided to interpolate the reaction rate between the oxidant starvation limit (OSL) and the full oxidation level. This is done by using the following multiplier on the reaction rate:

$$X_{OSL} = \begin{cases} 0.0 & if \quad x_{ox} < x_1 \\ (x_{ox} - x_1)/(x_2 - x_1) & if \quad x_1 < x_{ox} < x_2 \\ 1.0 & if \quad x_2 < x_{ox} \end{cases}$$

The applied limits are:

- $x_1 = 10^{-5}$
- $x_2 = 10^{-4}$

In case of multiple reactions, the dominant reaction is the only one occurring, as long as the fraction of oxidant exceeds the limit:

- $x_{\text{lim}} = 5 \times 10^{-5}$

The limit can be re-defined by the user (input parameters XLIMSC/XLIMTC). Default value was selected approximately in the middle of the interpolation zone, when the dominant reaction strength is reduced roughly by half.

11 Hydrogen Burn Package

11.1 Introduction

The hydrogen burn model is described in this chapter. The general approach to the modeling of burn in the SPECTRA code is similar to the approach taken in MELCOR [46], or HECTR 1.5 [6]. The model considers the effect of burning on a global basis, without tracking the actual flame front propagation within a Control Volume. The model implemented in SPECTRA is somewhat more elaborate and accurate than the models in MELCOR or HECTR. The influence of temperature on the flammability limits is taken into account. Data is obtained from extensive literature review. Results of recent investigation of hydrogen burn are taken into account, in particular the fast turbulent deflagration mode is separated from the slow deflagration mode, the expansion ratio (σ) criterion for fast turbulent deflagration, and the cell detonation size (λ) criterion for detonation, are available. An ignition model is available, with igniter temperature being a function of time (defined by a tabular or control function).

The flammability limits are expressed in SPECTRA in terms of a minimum hydrogen fraction (in literature referred to as the "lower flammability limit"), and a minimum oxygen fraction (the value of 1.0 – (minimum oxygen fraction)/(oxygen fraction in the atmosphere) is referred to as the "upper flammability limit"). The fact that SPECTRA uses the oxygen limit makes the model more general. The flammability limits are still appropriate when steam is replaced by another inert gas, for example helium - see Volume 3.

Generally the hydrogen burn model, when activated, calculates the flammability of the gas mixture in a Control Volume, checks ignition criteria and, upon ignition, calculates H_2 and O_2 consumption, as well as generation of steam and heat of reaction.

The burn model distinguishes three different modes:

- Slow deflagration
- Fast turbulent deflagration
- Detonation

In the first mode the flame propagates with subsonic velocity, in the latter two modes the flame propagates with supersonic velocity, and a shock wave is created. The shock wave pressure is calculated in a simplified way, using the ideal gas model.

The model is described in several sections. The description begins with the discussion of flammability limits, presented in section 11.2. Section 11.3 provides a description of the ignition model. Methods to calculate the burn rate, as well as the flame velocity during slow deflagrations, fast turbulent deflagrations, and detonations, are shown in section 11.4. Section 11.5 provides a description of the combustion completeness model. Shock wave calculation is described in section 11.6. Finally section 11.7 provides a description of flame propagation between neighboring Control Volumes.

11.2 Flammability Limits

11.2.1 Background

An extensive review of flammability limits may be found in [8], [9], [10]. Flammability limits are typically drawn on H_2 - H_2O diagrams, with hydrogen mole fraction on the vertical axis, and steam fraction on the horizontal axis Figure 11-1.

The lower, nearly horizontal part of the line is called the lower flammability limit, LFL. Below that line the hydrogen fraction is too small to burn. The upper line, inclined at about 45°, is called the upper flammability limit, UFL, and in fact it represents the oxygen limit. Above that line the oxygen fraction is too small to burn. With increasing steam fraction the gas mixture becomes more inert - the minimum hydrogen and oxygen fractions required for burn are increasing. This is visible in Figure 11-1, as the hydrogen limit increases slightly with increasing steam fraction, and the oxygen limit line is somewhat steeper than 45°.

In the MELCOR code the flammability limit line is approximated by using three numbers:

- Minimum hydrogen fraction, required to burn (default value: 7%)
- Minimum oxygen fraction, required to burn (default value: 5%)
- Maximum steam fraction, below which the mixture is burnable (default value: 55%)

The resulting flammability limit line is shown in Figure 11-2. The values of the limiting fractions may be redefined via input data, but the defined values are constant during analysis, and the same in all Control Volumes.

In reality the flammability limits depend on temperature. For example, in case of dry air the lower flammability limit decreases from ~4 to 5 vol% hydrogen at 20°C, to ~2 vol% at 500°C [8]. Thus the temperature effect is quite significant.

A method of constructing flammability limits using analytical functions has been proposed by Plys, Astleford, and Epstein [10]. The full flammability limit is represented by three functions, valid in different regions of hydrogen and steam fractions. As shown in [10] the resulting flammability curves are in excellent agreement with the published data.

The method used in the present model is based on the same approach, i.e. analytical functions are used to represent flammability limits, but the functions are different than those applied in [10]. The method presented below has the advantage of being somewhat simpler, while it still represents the actual data with similar accuracy as the method recommended by Plys et al.

The full flammability limit is represented by two functions, representing LFL and UFL respectively. The same general formula is used for both LFL and UFL. The formula is a combination of linear and exponential function:

$$x_{FL}(x_{inrt}, T) = A(T) + B \cdot x_{inrt} + \exp[-C \cdot (x_0 - x_{inrt})]$$

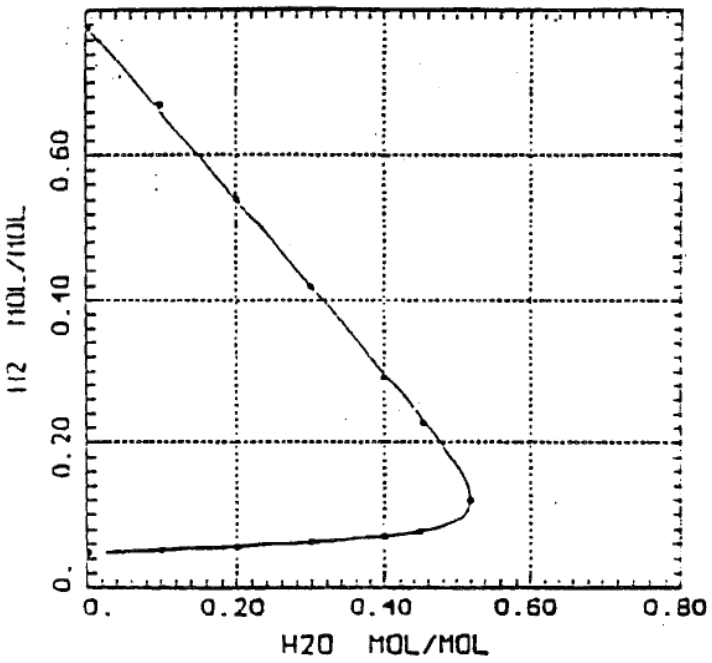


Figure 11-1 Typical flammability limits [10].

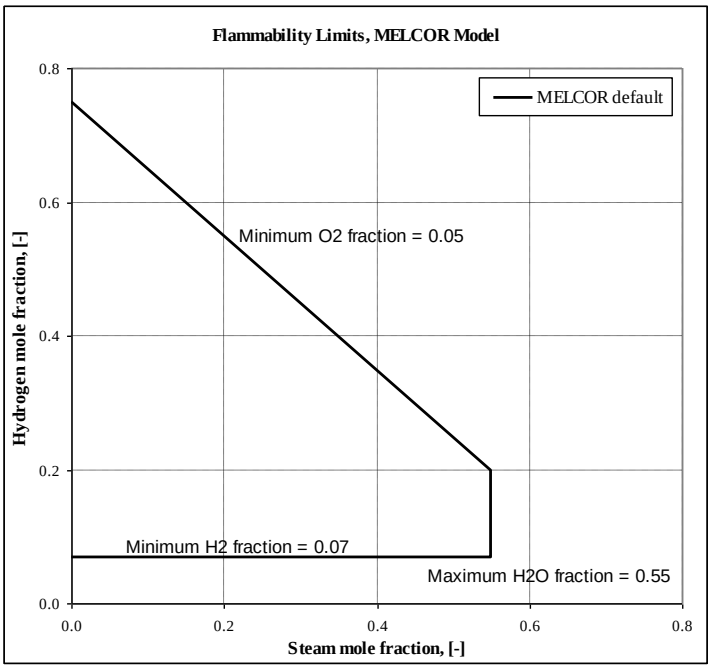


Figure 11-2 Flammability limits in MELCOR.

x_{FL} flammability limit, expressed as H_2 mole fraction in case of the lower limit ($x_{H2,LFL}$), and O_2 mole fraction in case of the upper limit ($x_{O2,UFL}$)

x_{inrt} mole fraction of inert gas (steam)

T temperature, K,

$A(T)$ the value of flammability limit for zero inert gas concentration. Temperature dependence is represented by an analytical function based on experimental data

B factor determining the influence of inert gas on the flammability limit for low inert gas concentrations

C factor determining the influence of inert gas on the flammability limit for high inert gas concentrations (close to x_0)

x_0 the value of inert gas concentration for which LFL and UFL lines would intersect if they were represented by straight lines (see Figure 11-3):

$$A_{LFL}(T) + B_{LFL} \cdot x_0 = A_{UFL}(T) + B_{UFL} \cdot x_0$$

An example of the flammability limit construction is shown in Figure 11-3. The LFL line is given straightforward by the formula for x_{FL}: LFL = $x_{H2,LFL}$. The UFL line is related to the formula for x_{FL} by: $x_{O2,UFL} = x_{O2,air} \times (1 - UFL)$, where x_{O2},air is the oxygen volume fraction in the air (0.20).

The limits of flammability are constructed using the method described above for the three burning modes:

- Slow deflagration
- Fast turbulent deflagration
- Detonation

The equations used for each of these modes are described below, in sections 11.2.2, 11.2.3, and 11.2.4. The functions defining flammability limits, and the values of constant coefficients were selected based on the flammability data collected in [8].

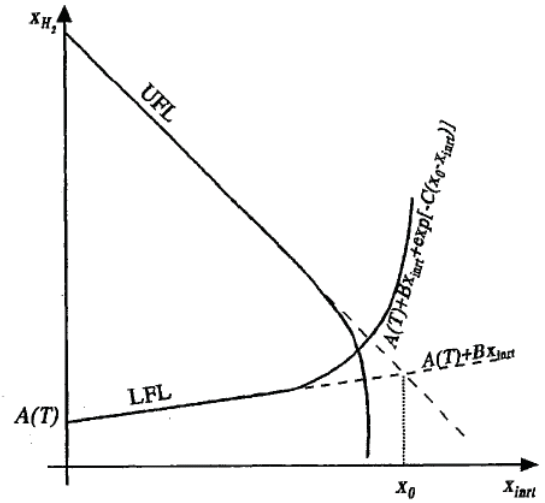


Figure 11-3 Construction of flammability line in SPECTRA

11.2.2 Slow Deflagration Limits

The functions defining slow deflagration limits are described below, for the lower and the upper flammability limits.

- Lower Flammability Limit, LFL

  The full lower flammability line is given by: LFL(x_{inrt}, T) = $x_{H2,\,LFL}$(x_{inrt}, T), where:

$$x_{H_2,LFL}(x_{inrt},T) = A(T) + B \cdot x_{inrt} + \exp[-C(x_0 - x_{inrt})]$$

  with the values of A, B, C, equal to:

$$A(T) = LFL(x_{inrt} = 0, T)$$
$$B = 0.02$$
$$C = 40.0$$

  where LFL(x_{inrt}=0, T) is the function determining the lower flammability limit for zero inert gas concentration. (Strictly speaking $x_{H2,\,LFL}$(x_{inrt}=0, T) is slightly larger than A(T), because of the exponential function, however, since x_0>0.5, then for x_{inrt}=0 the exponent is less than 10^{-20}, which equals zero with a very good accuracy).

  LFL(x_{inrt}=0, T) is calculated using two functions, valid in different temperature ranges. In the low temperature range a linear fit to the experimental data is used. For very high temperatures (above about 1000 K) the linear fit decreases below zero. To avoid negative values, in the high temperature range an exponential function is used, which asymptotically approaches zero (see Figure 11-4). The actual shape of the exponential function was selected to provide smooth extrapolation of the linear fit, that means the value of the exponential function and the value of its first derivative are equal to the corresponding values of the linear fit at the transition point from the linear function to the exponential asymptotic function.

  - Low temperature range ($T < T_1$ = 1000 K), linear fit to experimental data.

$$LFL(x_{inrt} = 0, T) = a \cdot [1 - b \cdot (T - T_0)]$$

    where T_0 is equal to 300 K, and the constants a, b, depend on the propagation direction.

    For the upwards propagation:
    a = 0.045
    b = 1.2×10^{-3}

    For the downwards propagation:
    a = 0.094
    b = 1.1×10^{-3}

- High temperature range ($T > T_1 = 1000$ K), asymptotic extrapolation.

$$LFL(x_{inrt} = 0, T) = c \cdot \exp[-d \cdot (T - T_1)]$$

where the constants *c*, *d*, are determined from the continuity of the function and its first derivative, and are equal to:

$$c = a \cdot [1 - b \cdot (T_1 - T_0)]$$
$$d = \frac{ab}{c}$$

- Upper Flammability Limit, UFL

The full upper flammability line is given by: $\mathrm{UFL}(x_{inrt}, T) = 1 - x_{\mathrm{O2,\,UFL}}(x_{inrt}, T)/x_{\mathrm{O2,\,air}}$, where:

$$x_{O_2,UFL}(x_{inrt}, T) = A(T) + B \cdot x_{inrt} + \exp[-C(x_0 - x_{inrt})]$$

with the values of *A, B, C*, equal to:

$$A(T) = [1.0 - UFL(x_{inrt} = 0, T)] \cdot x_{O2}$$
$$B = 0.005$$
$$C = 60.0$$

where $x_{\mathrm{O2,\,air}}$ is the oxygen volume fraction on the atmospheric air, $\mathrm{UFL}(x_{inrt}=0, T)$ is the function determining the upper flammability limit for zero inert gas concentration, calculated again using two functions, valid for different temperature ranges.

- Low temperature range ($T < T_1 = 1100$ K), linear fit to experimental data.

$$UFL(x_{inrt} = 0, T) = a \cdot [1 - b \cdot (T - T_0)]$$

where T_0 is equal to 300 K, and the constants *a*, *b*, are the same for the upward and downward propagation, and are equal to:

$a = 0.25$
$b = 1.1 \times 10^{-3}$

- High temperature range ($T > T_1 = 1100$ K), asymptotic extrapolation.

$$UFL(x_{inrt} = 0, T) = c \cdot \exp[-d \cdot (T - T_1)]$$

The constants *c*, *d*, are determined from the continuity of the function and its first derivative, and are expressed by the same formulae as shown above

The final flammability criterion, as calculated by the burn model in SPECTRA is:

$$x_{H_2} > x_{H_2,LFL}(x_{inrt}, T)$$
$$x_{O_2} > x_{O_2,UFL}(x_{inrt}, T)$$

Steam as well as helium and carbon dioxide are treated in the model as the inert gases, therefore the inert gas concentration is equal to: $x_{inrt} = x_{H2O} + x_{He} + x_{CO2}$. The temperature dependence of the lower and upper flammability limits for zero inert gas concentration are shown in Figure 11-4.

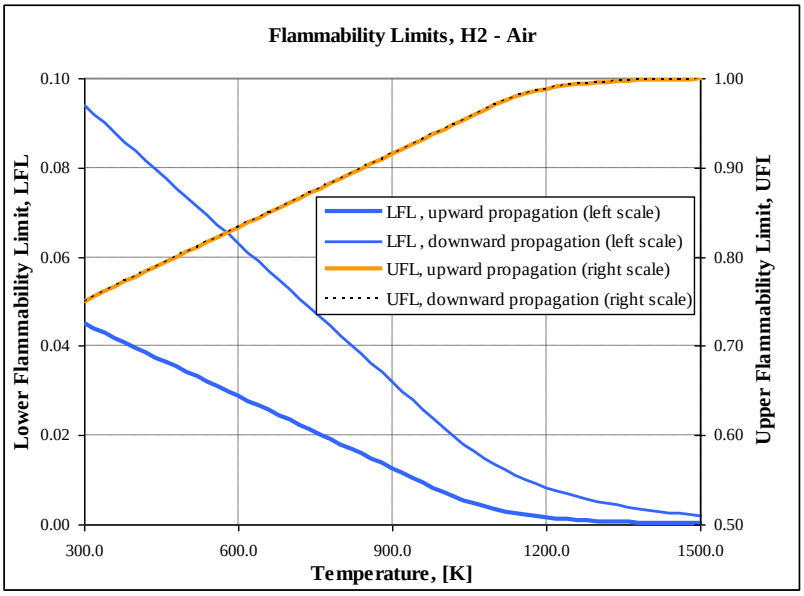


Figure 11-4 Flammability limits at zero inert gas concentration, burn model in SPECTRA

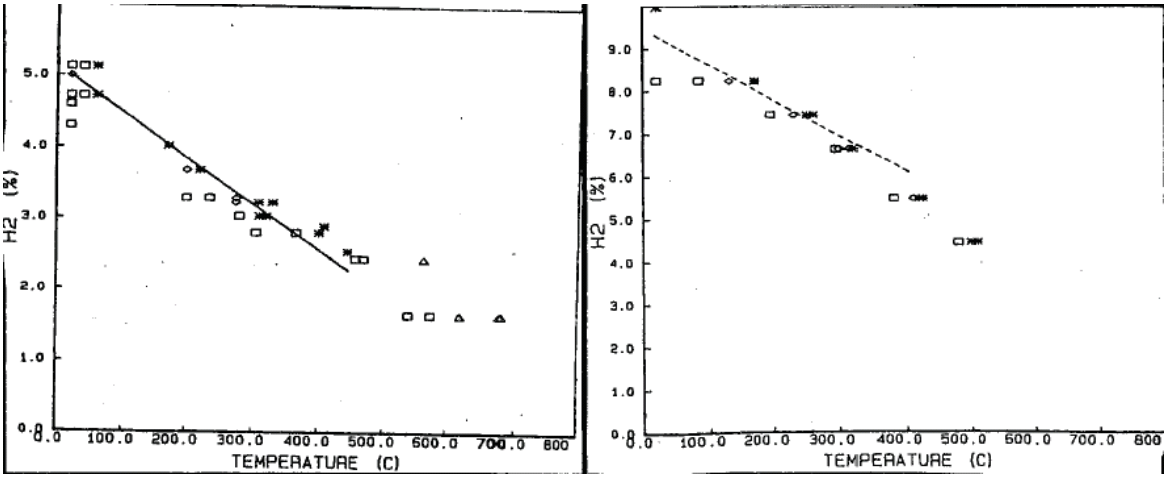


Figure 11-5 Upward (left) and downward (right) flammability limits [9]

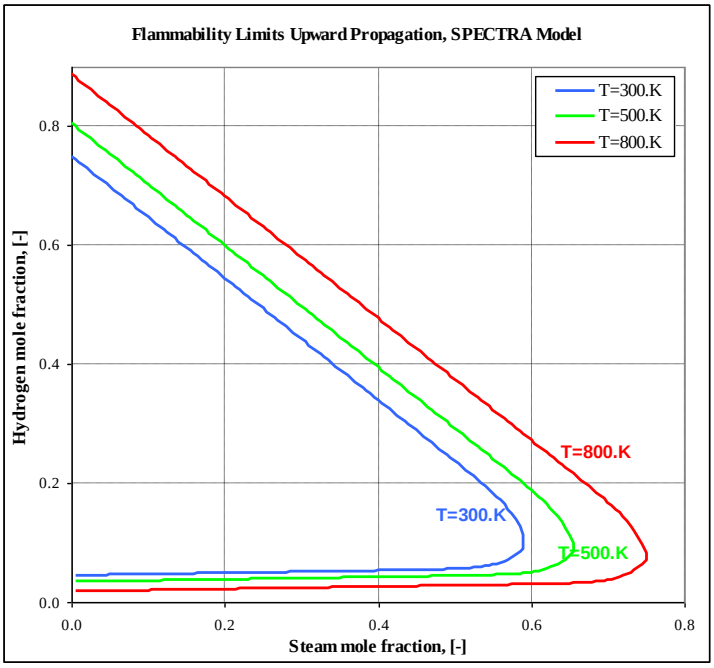


Figure 11-6 Flammability limits, upward propagation, burn model in SPECTRA

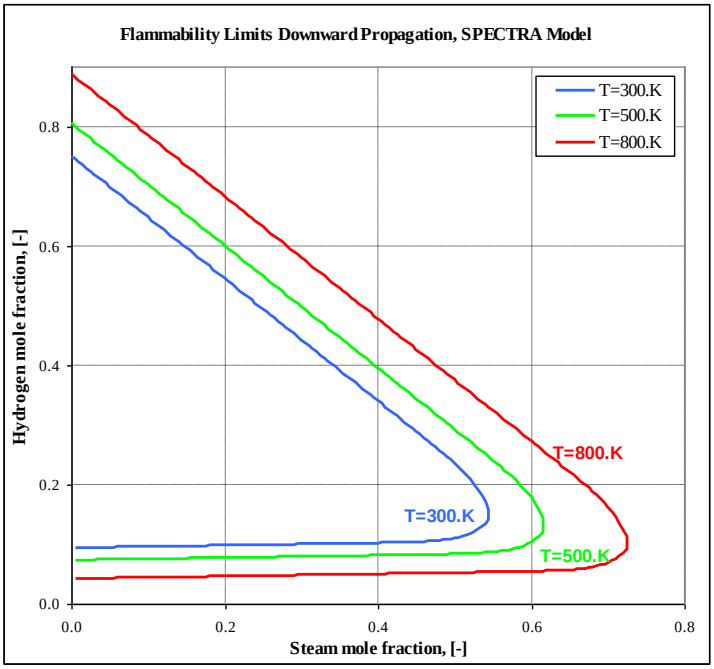


Figure 11-7 Flammability limits, downward propagation, burn model in SPECTRA

Experimental data may be found in for example [9]. Figure 11-5, reproduced from reference [9], show the lower flammability limits for upwards and downwards propagation. The experimental data indicates that at room temperature the flammability limit is 5 vol% H2. An overview of several experiments presented in [8] indicate that the value is between 4.5 and 5 vol%. In the present model the value of 4.5% is used, as the most conservative, as the flammability limit at 300 K.

Generally, the value of every constant in the SPECTRA burn model was chosen based on compilation of burn data presented in [8], to provide agreement with the most conservative results presented there.

The full flammability limits, as calculated by the model, are shown in Figure 11-6 and Figure 11-7. Figure 11-6 shows the flammability limits for upward propagation. These limits are always used to determine gas flammability status within a Control Volume. Figure 11-7 show the flammability limits for downward propagation. Those limits are used in case of down flame propagation from one CV to another, through a vertically oriented junction (see section 11.7).

11.2.3 Fast Turbulent Deflagration Limits

Two models are available to determine the limits of the fast turbulent deflagration, the default model and the alternative model. The default model is based on the same approximation functions, as those used to calculate the slow deflagration limits, but with different constants. The values of these constants were selected based on available experimental data to provide possibly conservative results, and generally the default model gives an earlier transition to the fast turbulent deflagration range than the alternative model. The alternative model is based on the expansion ratio criterion, σ (so called "σ criterion"). Both models are described below, in sections 11.2.3.1 and 11.2.3.2.

11.2.3.1 Default Model

The default model is based on the same functions as those used to calculate slow deflagration limits. Therefore all equations are identical to those shown in section 11.2.2, but the value of constants are different. The value of constants were selected based on data from [12], [14]. The values of the constants are listed below.

- Lower Fast Turbulent Deflagration Limit, LFTDL

  The values of constants are:

  $B = 0.03$
  $C = 40.0$
  $a = 0.094$
  $b = 1.1\times10^{-3}$
  $T_1 = 1000$

  There is no difference between the upwards and the downwards propagation, therefore the values of a, b, T_1, are valid for any propagation direction.

- Upper Fast Turbulent Deflagration Limit, UFTDL

  The values of constants are:

  $B = 0.05$
  $C = 60.0$
  $a = 0.25$
  $b = 1.1 \times 10^{-3}$
  $T_1 = 1100$

  There is no difference between the upwards and the downwards propagation, therefore the values of a, b, T_1, are valid for any propagation direction.

The temperature dependence of the lower and upper limits of fast turbulent deflagration for zero inert gas concentration are shown in Figure 11-8.

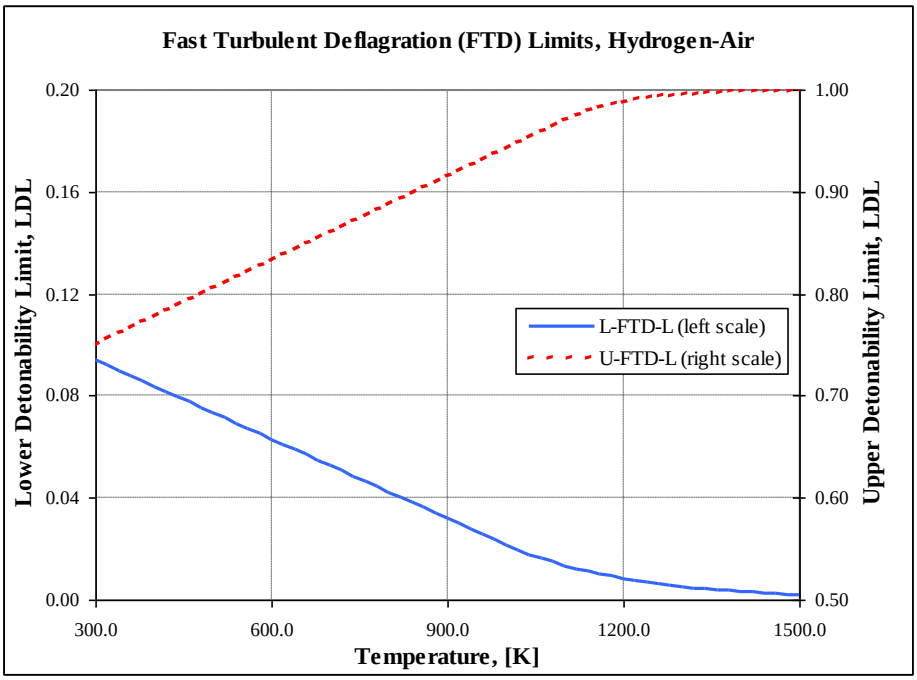


Figure 11-8 Flame acceleration (FTD) limits at zero inert gas concentration, SPECTRA.

11.2.3.2 Alternative Model - the "σ-criterion"

The σ criterion states that the value of the expansion ration, σ, defined as the ratio of specific volumes of burned to unburned gas at constant pressure, must be larger than a certain, temperature dependent critical value, $\sigma^*(T)$, in order to accelerate flame and get into the fast turbulent deflagration mode [13], [14]. The critical value of σ is expected to be equal to 3.5 - 4.0 at 300 K, and decrease with temperature. Reference [13] states that at 400 K the critical value is about 2.9. Reference [14] shows the temperature dependence in a graph, reproduced below, in Figure 11-9.

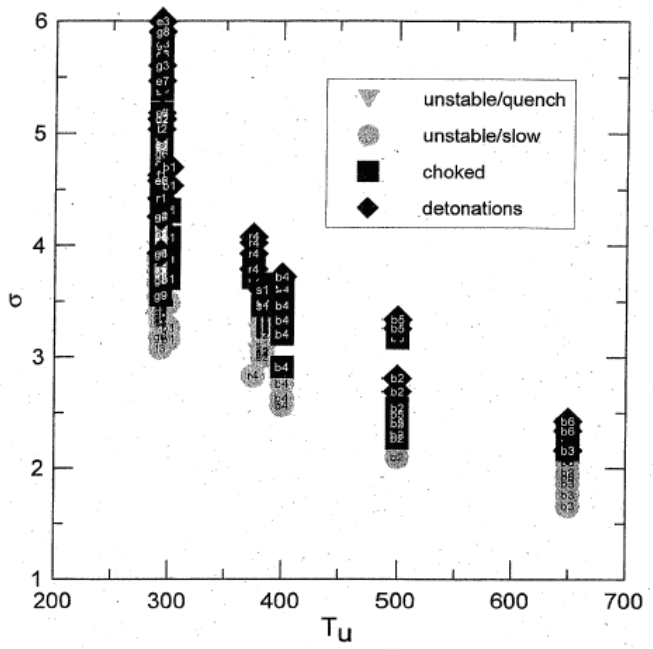


Figure 11-9 Expansion ratio, σ, versus initial temperature, T, [14]

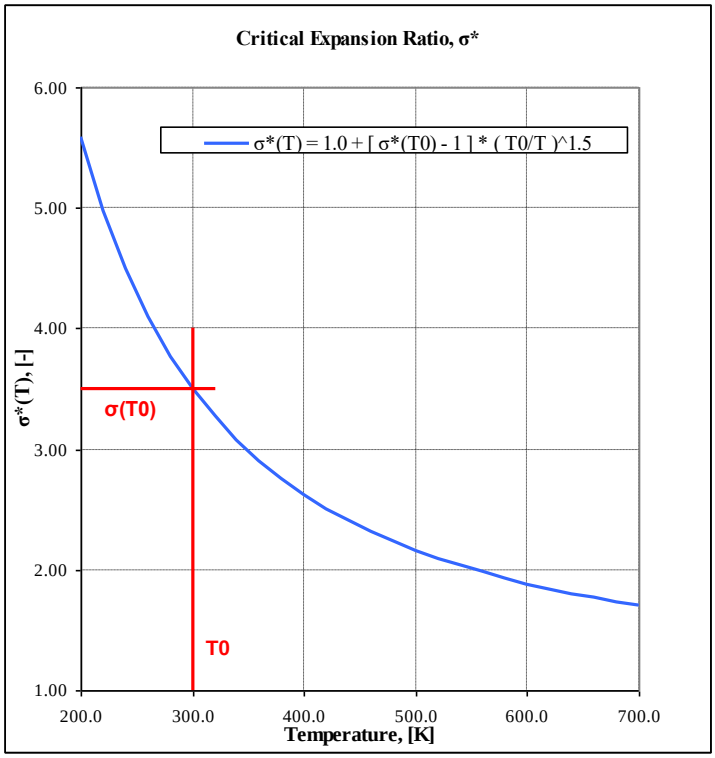


Figure 11-10 Expansion ratio, σ, versus initial temperature, correlation used in SPECTRA

In the SPECTRA burn model, the expansion ratio is calculated using the perfect gas assumption. This assumption is taken, because the σ-criterion is an empirical criterion which itself has a limited accuracy, and therefore there is no need to calculate σ with very good accuracy. With the perfect gas approach, the expansion ratio is calculated as:

$$\sigma = \frac{v_b}{v} = \frac{T_b}{T}$$

Here v is the specific volume, T is temperature. The subscript "b" refers to the burned gas, while the initial (unburned) gas values are given with no subscript. The temperature T_b is obtained from:

$$T_b = T + \frac{B \cdot q_{H_2} - [h(T, c_b) - h(T, c)]}{c_p(T)}$$

q_{H2} heat of burn per kilogram of burned H_2, J/kg_{H2}, (equal to: 1.206×10^8)

B mass of burned H_2 per one kilogram of burnable gas mixture, kg_{H2}/kg, equal to:

$$B = Min(c_{H_2}, c_{O_2}/8)$$

Here c_{H2}, c_{O2}, are the mass fractions of hydrogen and oxygen in the initial gas mixture. The factor 1/8 is the ratio of molar weights for the burn reaction: $2H_2 + O_2$.

$h(T, c)$ initial enthalpy, J/kg, of gas at initial temperature, T, and initial gas composition, c

$h(T, c_b)$ enthalpy of gas mixture at initial temperature, T and final (burned) gas composition, c_b

c_p specific heat at constant pressure, J/kg, calculated at the initial temperature T

In the above equation the value of specific heat, c_p, is taken at the initial temperature T. An exact formulation would require calculation of the integral:

$$\int_T^{T_b} c_p(T')dT'$$

In order to simplify the calculation the temperature dependence of c_p is neglected. Since c_p increases with temperature for all gases and all temperature range (at least for the gases and for the temperature range available in SPECTRA), this is a conservative assumption, that is it results in higher σ than would be obtained from the integral.

The final formula for the expansion ratio is therefore:

$$\sigma = \frac{T_b}{T} = 1.0 + \frac{B \cdot q_{H_2} - [h(T, c_b) - h(T, c)]}{c_p(T) \cdot T}$$

The critical value of the expansion ratio, σ^*, is calculated within the program by an approximation equation. The following equation has been selected:

$$\sigma^*(T) = 1.0 + [\sigma^*(T_0) - 1.0] \cdot \left(\frac{T_0}{T}\right)^{1.5}$$

where T_0 is equal to 300 K, and the value of critical expansion rate at the temperature T_0, $\sigma^*(T_0)$, is an input parameter for the code. The recommended value is 3.5 (see Volume 2), as the most conservative value recommended in [14]. The critical expansion ratio, $\sigma^*(T)$, as calculated by the code with this value of $\sigma^*(T_0)$, is shown in Figure 11-10. It is seen that the calculated line gives a conservative criterion for flame acceleration, as it is everywhere below, or equal to the "choked flame" values from Figure 11-9.

The alternative flame acceleration criterion is:

$$\sigma > \sigma^*(T)$$

The alternative criterion gives similar values as the default criterion. With the recommended value of $\sigma^*(T_0)$, equal to 3.5, the alternative criterion is still somewhat less conservative than the default criterion. Comparison of the default and the alternative criterion is provided in Volume 3.

11.2.4 Detonation Limits

Two models are available to determine the detonation limits. The default model is based on the same approximation functions, as used for the slow deflagration limits (section 11.2.2), and fast turbulent deflagration limits (section 11.2.3.1), but with different constants.

The values of the constants were selected based on available experimental data to provide possibly conservative results, and generally the default model gives an earlier transition to detonation than the alternative model. The alternative model is based on the cell detonation size, λ, criterion, (so called "λ-criterion"). Both models are described below, in sections 11.2.4.1 and 11.2.4.2. Results of both models are compared in Volume 3.

11.2.4.1 Default Model

The default model is based on the same functions as those used to calculate slow deflagration limits. Therefore all equations are identical to those shown in section 11.2.2, but the value of constants are different. The value of constants were selected based on data from [7], [8]. The values of the constants are listed below.

- Lower Detonability Limit, LDL

  The values of constants are:

  $B = 0.035$
  $C = 40.0$
  $a = 0.11$
  $b = 1.1\times10^{-3}$
  $T_1 = 1000$

  There is no difference between the upwards and the downwards propagation, therefore the values of a, b, T_1, are valid for any propagation direction.

- Upper Detonability Limit, UDL

  The values of constants are:

  $B = 0.09$
  $C = 60.0$
  $a = 0.25$
  $b = 1.1 \times 10^{-3}$
  $T_1 = 1100$

  There is no difference between the upwards and the downwards propagation, therefore the values of a, b, T_1, are valid for any propagation direction.

The temperature dependence of the lower and upper limits of fast turbulent deflagration for zero inert gas concentration are shown in Figure 11-8. The lower detonability limit at 300 K has been set to 11%. This is a conservative estimation based on data compilation in [8]. Reference [7] gives the limit of about 15% at 300 K.

The full detonability limits, as calculated by the SPECTRA model, are shown in Figure 11-12. For comparison, the detonability limits from the MELCOR code are shown in Figure 11-13. The detonability limit line is approximated in MELCOR by using three numbers:

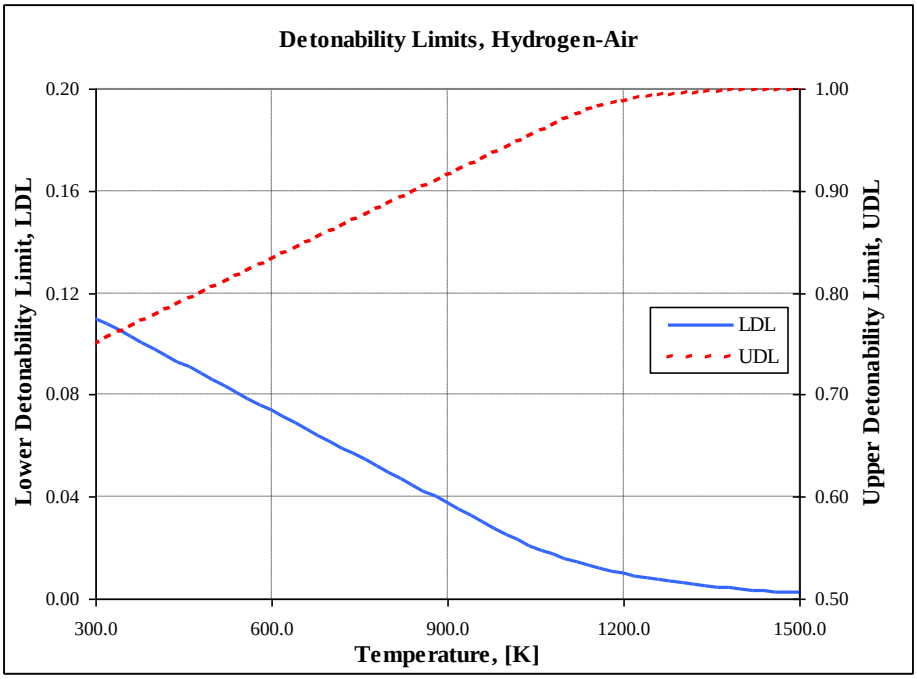


Figure 11-11 Detonability limits at zero inert gas concentration.

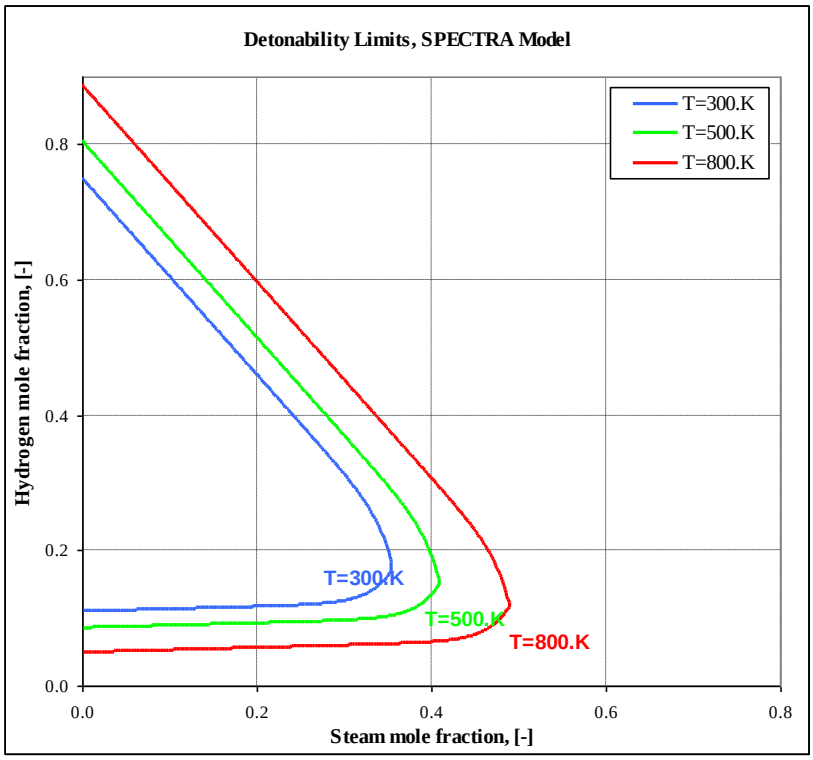


Figure 11-12 Detonability limits, SPECTRA H_2 burn model.

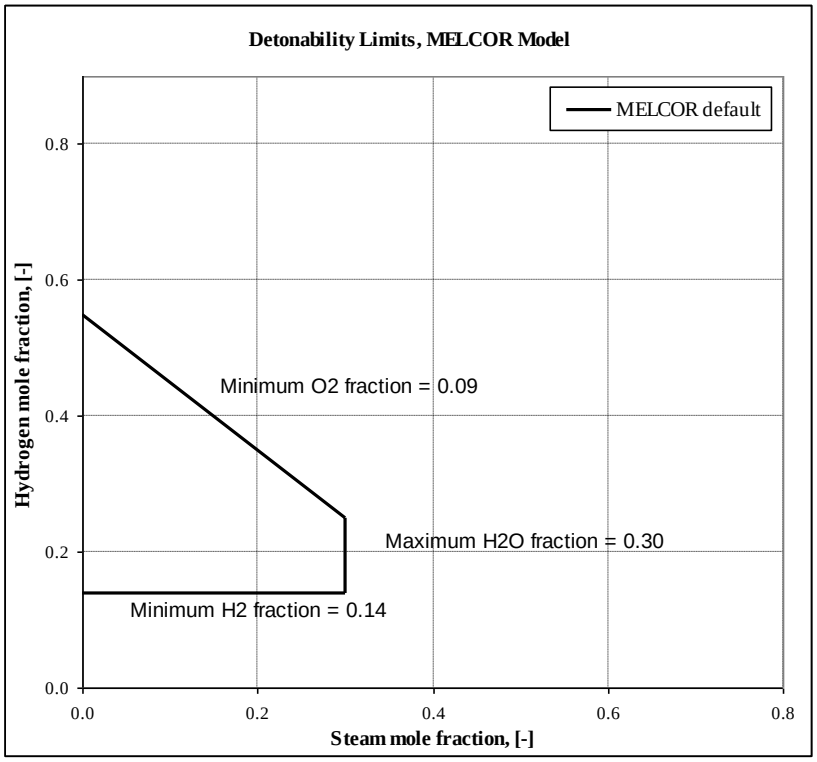


Figure 11-13 Detonability limits, MELCOR model.

- Minimum hydrogen fraction, required to burn (default value: 14%)
- Minimum oxygen fraction, required to burn (default value: 9%)
- Maximum steam fraction, below which the mixture is burnable (default value: 30%)

The detonability limits are temperature dependent, which is taken into account in the SPECTRA model. In the MELCOR model the detonability temperature dependence is neglected. At higher temperatures the detonability limits are much wider than those used by default in MELCOR.

11.2.4.2 Alternative Model - the "λ-criterion"

The λ criterion states that the ratio between the burn characteristic dimension, D, and the cell detonation size, λ, must be larger than a certain value, in order to obtain detonation [13], [14]. The critical value of D/λ is an input parameter for the code. It is expected to be equal to 6.0 - 7.0. Reference [13] recommends the value of 7.0, however, as can be seen in figure 4 in [13], detonations are sometimes possible for D/λ equal to about 6.0. Therefore the value of 6.0 is recommended in the SPECTRA User's Guide, to be used if the alternative criterion is selected (Volume 2).

To use the λ-criterion the burn characteristic dimension is needed. Reference [13] recommends the following values:

- Normal room, $L \approx W \approx H$: $D = (L + W + H)/3$
- Flat room, $L \approx W \gg H$: $D = (L + W)/2$
- Channels/tubes, diameter d: $D = 2.5\ d$

The value of detonation cell size, λ, is calculated by the program using the correlation shown in [14], Appendix D. The formula is:

$$\log_{10}(\lambda) = \left\{ a - t + \left[\frac{b}{(x_{H_2} - s/T)^f} + h \cdot (x_{H_2} - g \cdot T)^2 + q \cdot (x_{H_2} - g \cdot T) \right] \cdot \right.$$
$$\left. \cdot \left[1 + d \cdot x_{H_2O} + e \cdot T \cdot x_{H_2O}^2 \right] \cdot \frac{r}{T} \right\} \cdot (p - c) \cdot [v + u \cdot (p - w)] + t$$

T temperature, (K)
p pressure, (MPa)
x_{H2} hydrogen volume fraction, (-)
x_{H2O} steam volume fraction, (-)
λ detonation cell size, (cm)
a, b, c,...constants, given in Table 11-1

The values of the detonation cell size, calculated from the above equation, are shown in Figure 11-14. For comparison, Figure 11-15 shows the values of detonation cell size, as presented in reference [14]. Values obtained at different temperatures are shown in Figure 11-16 and Figure 11-17. At higher temperatures the detonability limits become broader.

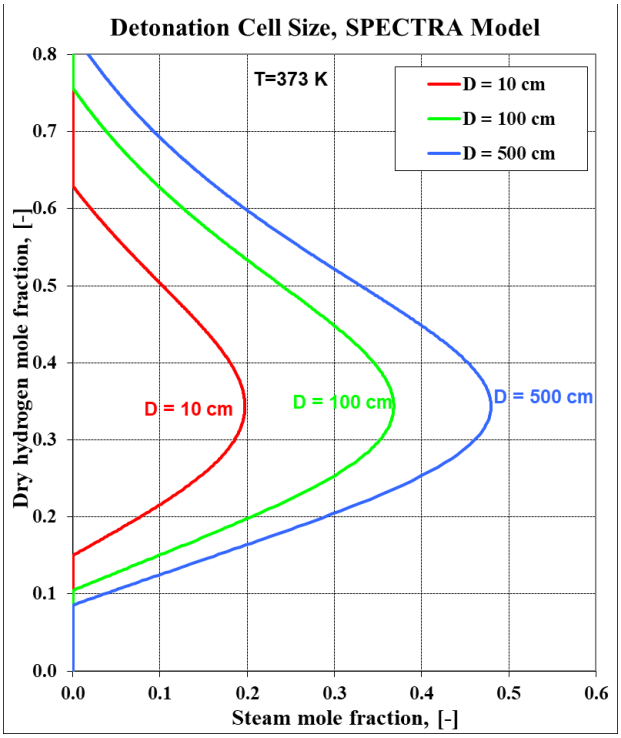


Figure 11-14 Detonation cell size as coded in SPECTRA

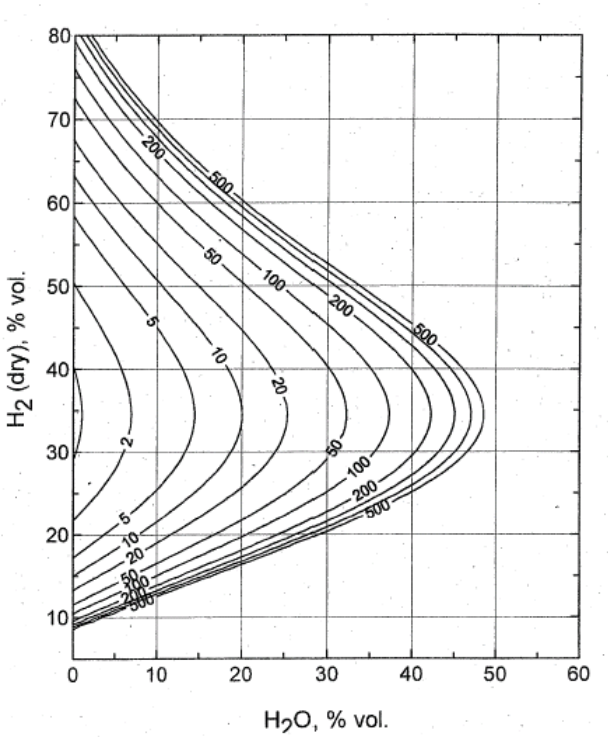


Figure 11-15 Detonation cell size [14]

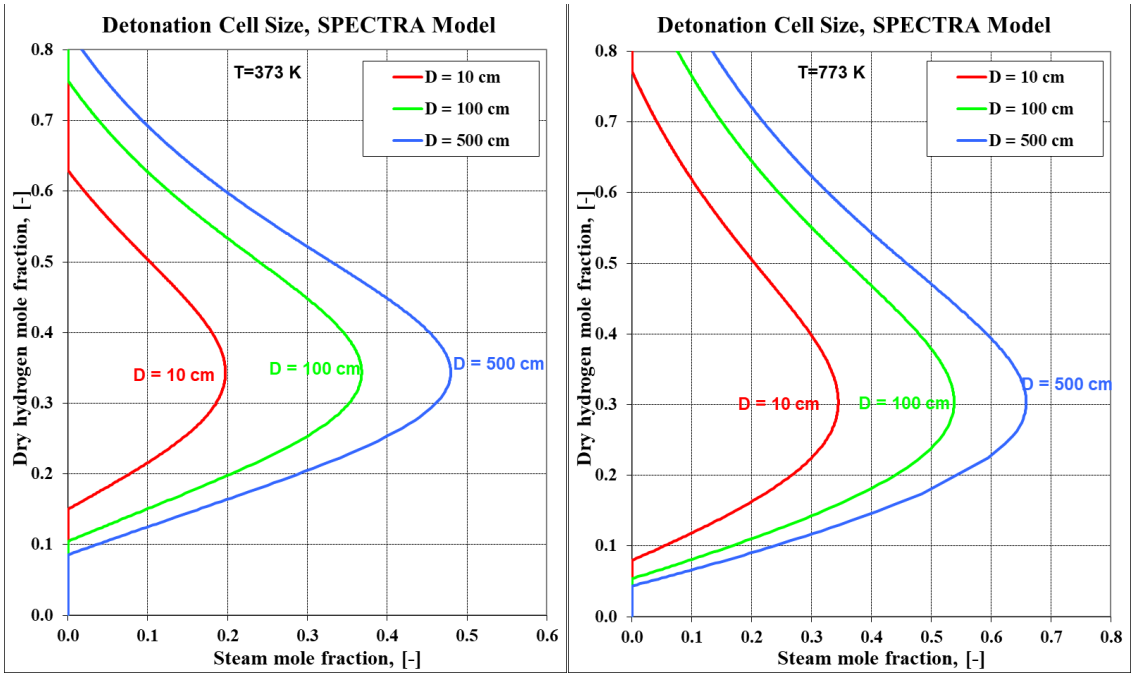


Figure 11-16 Detonation cell size, left: *T* = 100°C, right:500°C

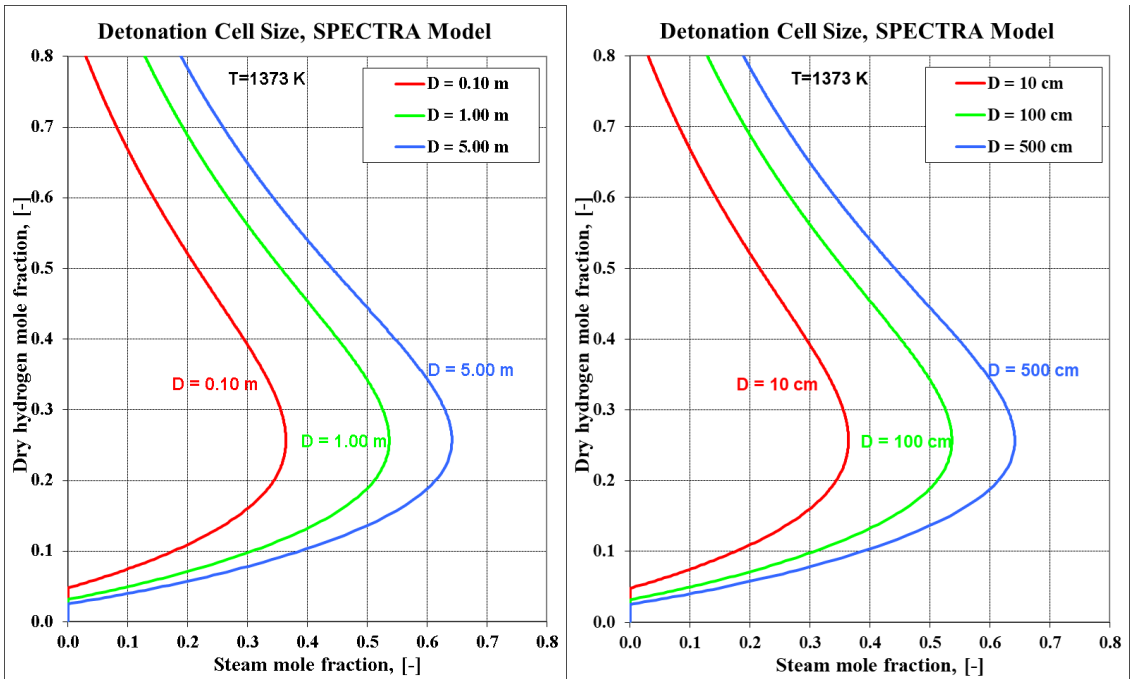


Figure 11-17 Detonation cell size, T = 1000°C

Table 11-1 Constants in the formula for detonation cell size.

Constant	Value	Constant	Value	Constant	Value
a	–1.1333	f	0.99747	s	–418.22
b	45.981	g	–0.026665	t	2.3897
c	–0.15765	h	8.75×10^{-4}	u	–8.4238
d	0.046543	q	–0.040764	$v=1/(0.1-c)$	3.8812
e	3.5962×10^{-7}	r	331.16	w	0.1

The alternative criterion gives similar values as the default criterion. For large steam fractions the alternative criterion is somewhat less conservative than the default criterion. Comparison of the default and the alternative criterion is provided in Volume 3.

11.2.5 Summary of the Flammability Model

The flammability model calculates gas flammability status within each Control Volume, where the hydrogen burn model has been activated. The model distinguishes three burning modes:

- Slow deflagrations
- Fast turbulent deflagrations
- Detonations

Each mode is given a code number. The plot variable MODEH2 allows to plot the flammability status within a Control Volume, using these code numbers (Volume 2). The meaning of the code numbers are:

- MODEH2=0 : inflammable gas mixture
- MODEH2=1 : flammable, slow deflagration expected upon ignition
- MODEH2=2 : flammable, fast turbulent deflagration expected upon ignition
- MODEH2=3 : flammable, detonation expected upon ignition

It should be noted that the variable MODEH2 gives only the gas flammability status, and not the burn status. It is possible that the gas mixture is flammable (MODEH2≥1), and burn does not occur. To initiate burn two criteria must be met: gas must be flammable, and additionally the ignition criterion must be met. The ignition criterion is discussed in the next section.

A comparison of the gas flammability models is presented in Volume 3, by plotting the mode indicator MODEH2 for different temperatures, hydrogen fractions, and steam fraction. Here only one graph is presented (Figure 11-18), which gives the limits of slow deflagration, fast turbulent deflagration (flame acceleration), and detonation. The graph was made using the default models for the fast turbulent deflagration limits and the detonation limits. For comparison, a similar graph, reproduced from [12], is shown in Figure 11-19.

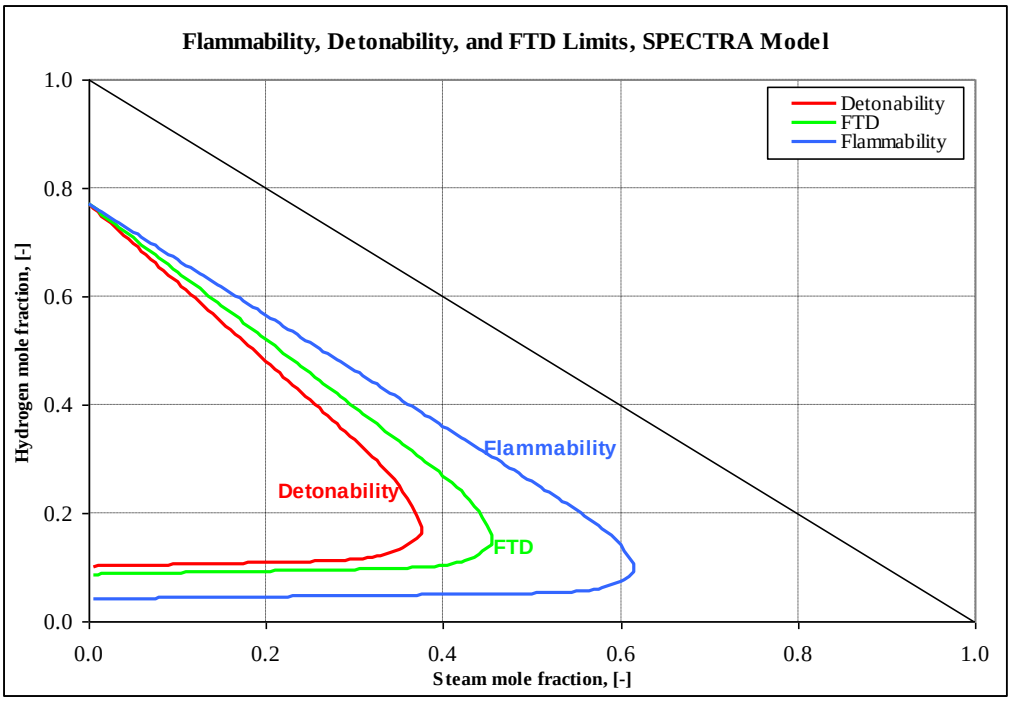


Figure 11-18 Flammability, detonability, and flame acceleration limits, *T*=373 K, SPECTRA.

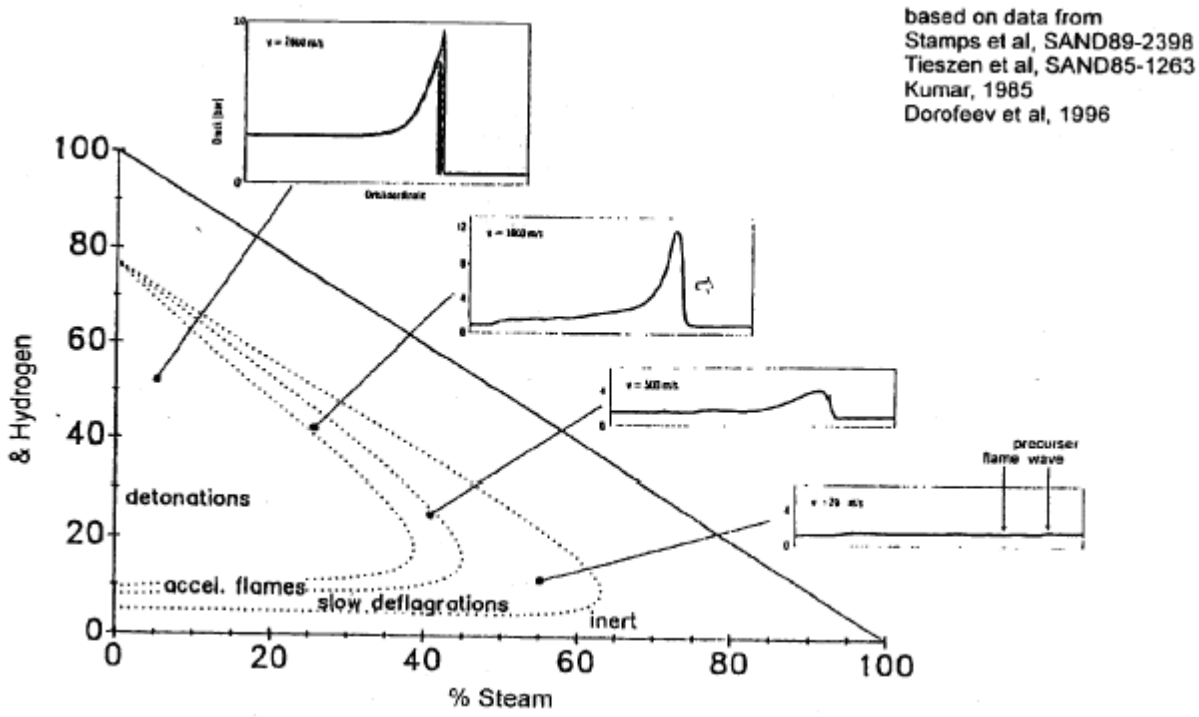


Figure 11-19 Flammability, detonability, and flame acceleration limits, *T*=373 K, [12].

11.3 Ignition Limits

In the previous section a method to calculate the gas flammability was described. If a gas mixture is flammable then it may be ignited. The ignition may be caused by external sources (igniters), such as burning candle, spark, etc. or it may occur without any external sources, a phenomenon which is called autoignition.

In the MELCOR code the approach to modelling ignition is as follows. The user determines if igniters are present in given Control Volume. If the igniters are present then ignition occurs as soon as the mixture becomes flammable (in fact the gas flammability limits are in MELCOR called "the ignition with igniters limits"). If there are no igniters, then autoignition will occur if the autoignition ("ignition without igniters") criteria are satisfied. The default autoignition criteria in MELCOR are:

- Minimum hydrogen fraction required for autoignition: 10%
- Minimum oxygen fraction required for autoignition: 5% (same as ignition with igniters)

With the above approach the ignition limits are quite similar to the flammability limits, only the constants are somewhat different. In fact the ignition and autoignition are not solely determined by gas concentrations, but also by parameters like temperature and pressure. In cold conditions one can obtained a detonable gas mixture with, say, 50% hydrogen, and the ignition would not occur.

In the SPECTRA code the following approach to ignition modelling is used. The autoignition limit is calculated using the method recommended in [15], as a function of temperature, pressure, gas composition, and the characteristic dimension. The eventual presence of igniters is taken into account by using the igniters temperature, specified by the user as a function of time through a tabular or a control function. Thus the ignition with igniters is calculated in the same way as the autoignition, except that the temperature is that of the igniters rather than that of the gas itself. Additionally the presence of any structures (1-D and 2-D Solid Heat Conductors) is taken into account by selecting the highest temperature of all surfaces present in a given Control Volume. The calculation procedure is described below.

The ignition limit is calculated from (see [15], chapter 2, equations 39, 42, 43, 44):

$$IL = A - \frac{1 + 0.5 \cdot C}{1.5 + 0.5 \cdot C + (2 + 0.5 \cdot C) \cdot B}$$

IL is the ignition limit. Ignition occurs when the value of *IL* > 0. The values of *A, B, C, D* are calculated from:

$$A = \frac{7.413}{x_{H_2} + 0.35 \cdot x_{O_2} + 0.43 \cdot x_{N_2} + 14.3 \cdot x_{H_2O}} \cdot \frac{T}{p} \cdot \exp\left[-10.6599 \cdot \left(\frac{803.0}{T} - 1 \right) \right]$$

$$B = 0.01305 \cdot \left(\frac{pD}{T} \right)^2 \cdot x_{H_2} \cdot \left(x_{H_2} + 6.88 \cdot x_{O_2} + 7.84 \cdot x_{N_2} \right) \cdot \exp\left[-15.0943 \cdot \left(\frac{803.0}{T} - 1 \right) \right]$$

$$C = \frac{0.0804}{x_{H_2} + 6.88 \cdot x_{O_2} + 7.84 \cdot x_{N_2} + 14.3 \cdot x_{H_2O}} \cdot \frac{T}{pD^2} \cdot \exp\left[-19.4387 \cdot \left(\frac{803.0}{T} - 1 \right) \right]$$

T temperature, K
p pressure, Pa
D characteristic dimension, m

The constants in the above equations have been converted to SI units. The original equation is written for pressure in mm of mercury, and dimension in cm. Correctness of the implementation of the explosion limit has been verified by comparing values calculated for a $2H_2+O_2$ mixture, with the graph presented in [15]. The calculated explosion limits are presented in Figure 11-20 (to make the plot the units of pressure and temperature were converted back to the same units as those used in [15]). The data from [15] is reproduced in Figure 11-21. A good agreement is observed.

It was observed that the above ignition criterion can sometimes predict ignition even if the gas mixture is inflammable. Therefore an additional criterion has been built into the model. The flammability limit for upwards propagation is calculated simultaneously with the above ignition criterion. The ignition is assumed to occur only if both criteria are satisfied. The final ignition limits, calculated for the burn characteristic dimensions between 1 cm and 10 m and hydrogen mole fractions between 0.0 and 1.0, are shown in Figure 11-22. The steep lines in the regions of low and high hydrogen fractions are a consequence of the additional criterion of gas flammability.

The ignition limits shown in Figure 11-22 are generally in agreement with the values presented in [8] (the autoignition limit is at about 550 - 600°C for D=3.9 cm). As mentioned above, in absence of igniters and hot structures in a Control Volume, the above limits represent the autoignition limits. If igniters and/or Solid Heat Conductors are present in given Control Volume, then the same function is used to calculate ignition limit, except that the temperature argument in this function is taken as a maximum of the gas temperature, the structure temperatures, and the igniter temperature:

$$T = Max(T_{gas}, T_{SC,i}, T_{TC,i}, T_{igniter})$$

T_{gas} gas temperature in Control Volume, (K)
$T_{SC,\,i}$ temperature of i-th 1-D Solid Heat Conductor surface present in the given CV, (K)
$T_{TC,\,i}$ temperature of i-th 2-D Solid Heat Conductor surface present in the given CV, (K)
$T_{igniter}$ igniter temperature, specified as a function of time through a Tabular or a Control Function.

With the present approach the user can for example simulate "sparks" occurring at certain time of accident, by specifying igniter temperature being low during the whole accident period except for a short period at the desired times. That way an ignition may be procured at the time when it is most dangerous, for example when the gas flammability status indicates detonability of the mixture. This may be desired to obtain the most conservative estimation of the containment loads during accident.

If, at any time of the calculation the ignition criteria are satisfied, then the hydrogen burn is initiated. The burn mode indicator, IBRNH2, becomes equal to the gas flammability at the moment of ignition (IBRNH2=MODEH2). The meaning of the variable IBRNH2 is therefore as follows:

- IBRNH2=0 : no burn
- IBRNH2=1 : slow deflagration in progress
- IBRNH2=2 : fast turbulent deflagration in progress
- IBRNH2=3 : detonation in progress

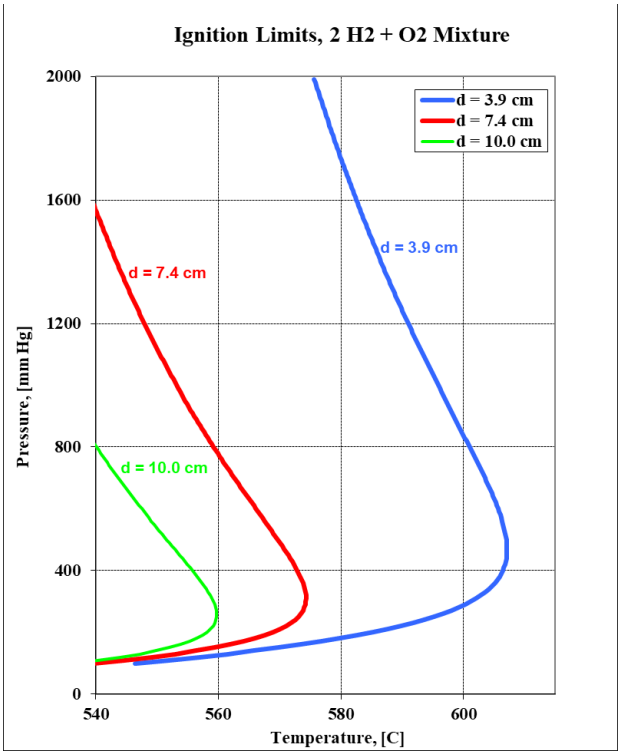


Figure 11-20 Ignition limits, $2H_2 + O_2$ mixture, SPECTRA model

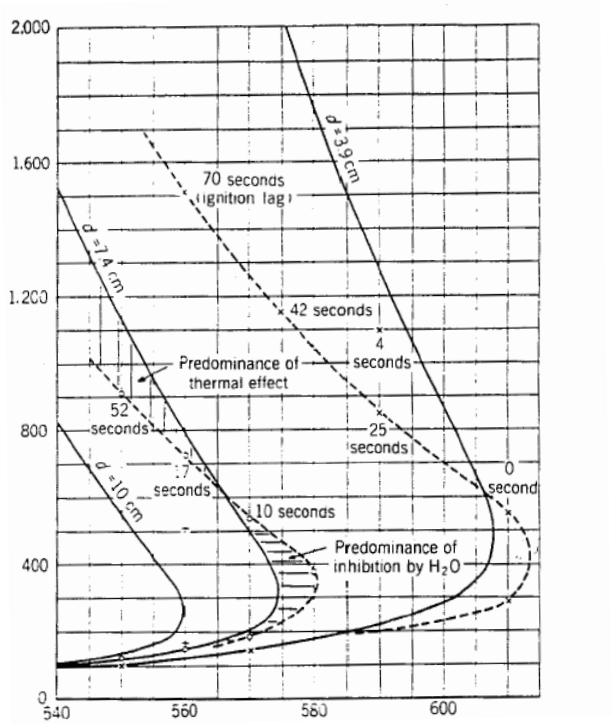


Figure 11-21 Ignition limits, $2H_2 + O_2$ mixture, [15]

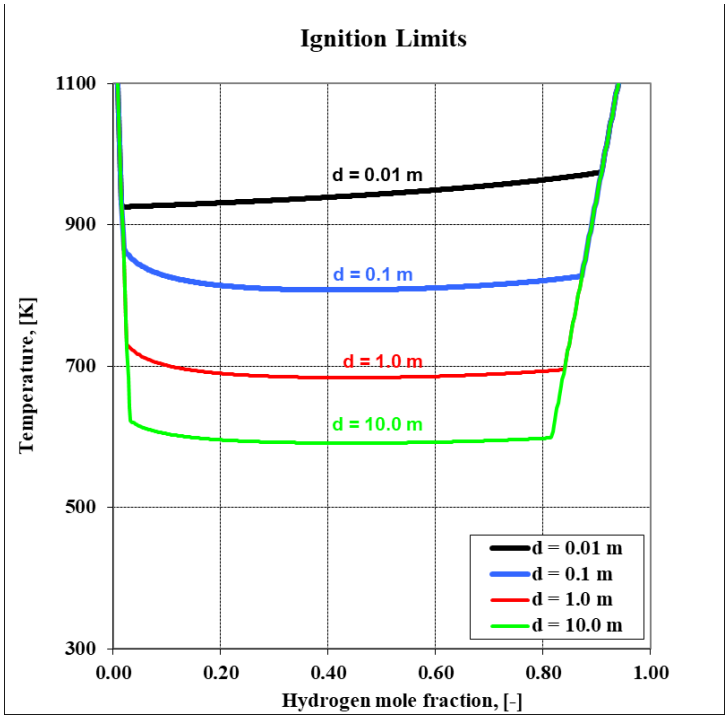


Figure 11-22 Ignition limits, hydrogen-air mixture, SPECTRA model

11.4 Reaction Rate and Flame Velocity

When the ignition criteria (see section 11.3) are satisfied, then hydrogen begins to burn. The reaction rate depends on the flame speed, and is calculated similarly as in the MELCOR code, as:

$$\frac{dm_{H_2}}{dt} = \frac{m^0_{H_2}}{\tau_{burn}}$$

dm_{H2}/dt	reaction rate, kg/s, mass of hydrogen consumed per second.
m_{H2}^0	mass of hydrogen in a Control Volume at the start of burn, kg
τ_{burn}	time of burn, s

The reaction is $2H_2+O_2 \rightarrow 2H_2O$. Therefore the O_2 consumption and H_2O production are equal to:

$$\frac{dm_{O_2}}{dt} = 8.0 \cdot \frac{dm_{H_2}}{dt} \qquad \frac{dm_{H_2O}}{dt} = 9.0 \cdot \frac{dm_{H_2}}{dt}$$

The influence of incomplete combustion is neglected in the above formula, in contrast to MELCOR where the combustion completeness is used (see [46], BUR Package Reference Manual, section 2.5). Neglecting incomplete combustion gives slightly larger reaction rates and thus more conservative loads due to burn. The use of combustion completeness is avoided here, because of uncertainties involved in the correlation (see section 11.5).

The burn time is calculated using the burn characteristic dimension, *D*, and the flame speed v_{flame}, as:

$$\tau_{burn} = \frac{D}{v_{flame}}$$

The value of *D* is an input parameter, with the default value equal to the radius of a sphere whose volume, *V*, is the same as the total volume of CV (see Volume 2):

$$D = \left(\frac{3V}{4\pi}\right)^{1/3}$$

The reaction rate is re-calculated at every time step during burn, and may change due to changing flame speed. For additional conservatism an assumption is made that flame speed can only increase during burn. Consequently the new time step flame velocity is used only if it is larger than the old time step velocity. If the calculated new time step velocity should be smaller than the old time step velocity then the old value is kept.

The flame velocity depends on the burning mode. The following three sections, 11.4.1, 11.4.2, 11.4.3, provide the description of flame velocity calculation for slow deflagrations, fast turbulent deflagrations, and detonations respectively.

11.4.1 Flame Velocity During Slow Deflagrations

The term slow deflagration includes here laminar and slow turbulent deflagrations, in which the flame velocity is from fraction of m/s to tens of m/s. The flame velocity is calculated in two steps. First the laminar flame speed is calculated. Then the influence of turbulence is taken into account, by using a multiplier on the laminar speed, to get the overall flame speed. Calculation of the laminar flame speed and the turbulent multiplier are described below.

- **Laminar Flame Speed**

  The laminar deflagration velocity is calculated using the correlation of Liu and MacFarlane [16]. The correlation is:

$$v_{lam} = B(x_{H_2}) \cdot T^{C(x_{H_2})} \cdot \exp[D(x_{H_2}) \cdot x_{H_2O}]$$

  T temperature, (K)
  x_{H2O} steam volume fraction, (-)

  The values of *B, C, D* are defined by:

$$B(x_{H_2}) = A_1 + A_2 \cdot (x_0 - x_{H_2}) + A_3 \cdot (x_0 - x_{H_2})^2$$
$$C(x_{H_2}) = A_4 + A_5 \cdot (x_0 - x_{H_2})$$
$$D(x_{H_2}) = A_6$$

Table 11-2 Constants in Liu McFarlane correlation

	$x_{H2} < x_0$	$x_{H2} > x_0$
A_1	4.644×10^{-4}	4.644×10^{-4}
A_2	-2.119×10^{-3}	9.898×10^{-4}
A_3	2.344×10^{-3}	-1.264×10^{-3}
A_4	1.571	1.571
A_5	3.839×10^{-1}	-2.476×10^{-1}
A_6	–2.21	–2.21

The value of x_0 is equal to 0.42, and the constants A_1 through A_6, are listed in Table 11-2. The value of A_6 for the high hydrogen concentrations ($x_{H2}>x_0$) has been changed from –2.24 (the value given in [16]) to –2.21. This was done to avoid discontinuity in the model. This modification results in slightly higher velocities when $x_{H2}>x_0$ and $x_{H2O}>0.0$. This fact is without any practical meaning, since with such high hydrogen fractions a detonation mode is typically expected. The original model has been modified in the low and the high hydrogen concentrations, based on comparison with the model applied in HECTR 1.5 [6] and MELCOR [46] codes. The modifications are as follows.

- Low hydrogen concentrations, $x_{H2} < x_{E1} = 0.15$

  In this region the flame velocity is calculated from:

$$v_{flame} = B(x_{E1}) \cdot T^{C(x_{E1})} \cdot \exp[D(x_{E1}) \cdot x_{H_2O}] \cdot \exp[E \cdot (x_{H_2} - x_{E1})]$$

  E is a constant. The value of E has been set to 9.0, based on comparison with HECTR 1.5 model (see Figure 11-29 and Figure 11-30 below).

- High hydrogen concentrations, $x_{H2} > x_{E2} = 0.65$

  In this region the flame velocity is calculated from:

$$v_{flame} = B(x_{E2}) \cdot T^{C(x_{E2})} \cdot \exp[D(x_{E2}) \cdot x_{H_2O}] \cdot \exp[E \cdot (x_{E2} - x_{H_2})]$$

  The value of E has been set to 9.0, the same as in case of low hydrogen fractions.

The flame speed values, as calculated by the appropriate subroutine in the SPECTRA code, are compared to the plots from the original paper of Liu, McFarlane in Figure 11-23 through Figure 11-28. Figure 11-23 and Figure 11-24 show the flame speeds for temperatures between 25°C and 250°C, for zero steam concentration. Figure 11-25 and Figure 11-26 show flame speeds for steam concentrations ranging from 0% to 12% (volume fractions), for 50°C. Figure 11-27 and Figure 11-28 show flame speeds for steam concentrations again from 0.0 to 12%, for 150°C. It is seen that the calculated values agree well with the original data. In the low and high hydrogen fraction regions the model coded in SPECTRA gives higher (thus more conservative) flame speeds than the original model.

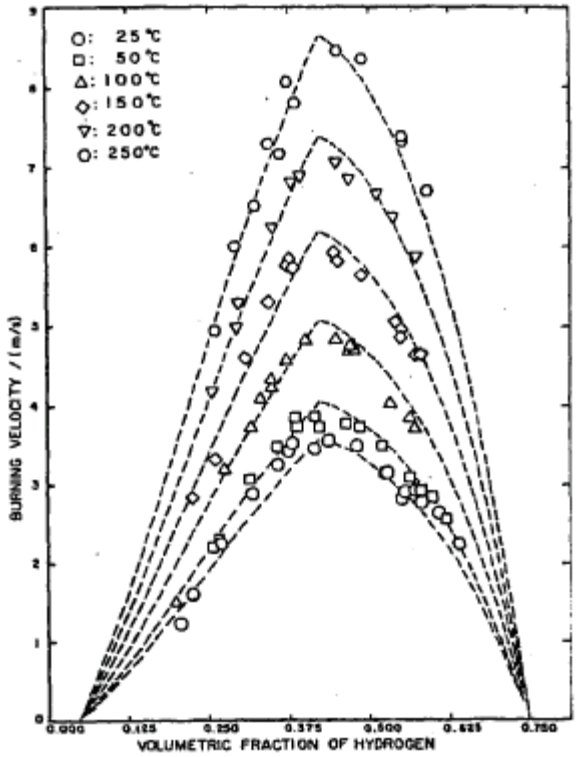


Figure 11-23 Laminar flame velocities, T = 25°C - 250°C, [16]

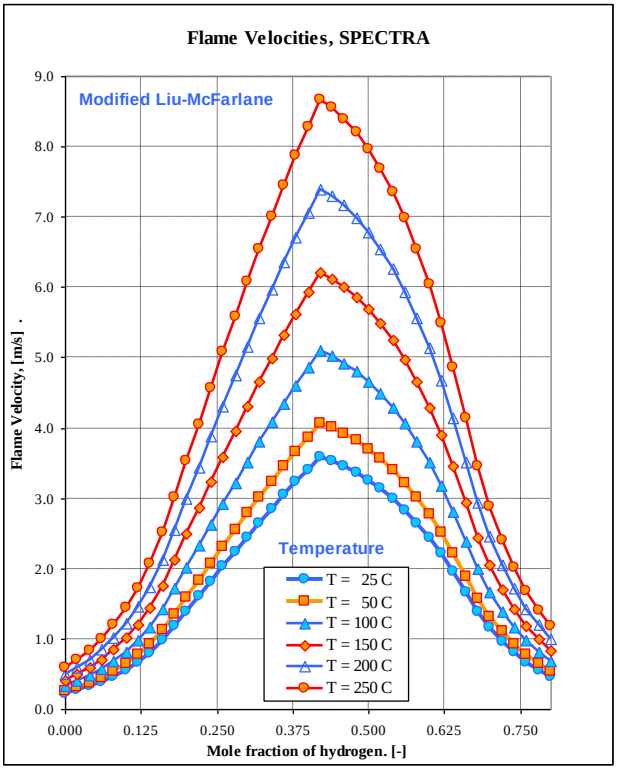


Figure 11-24 Laminar flame velocities, T = 25°C - 250°C, SPECTRA

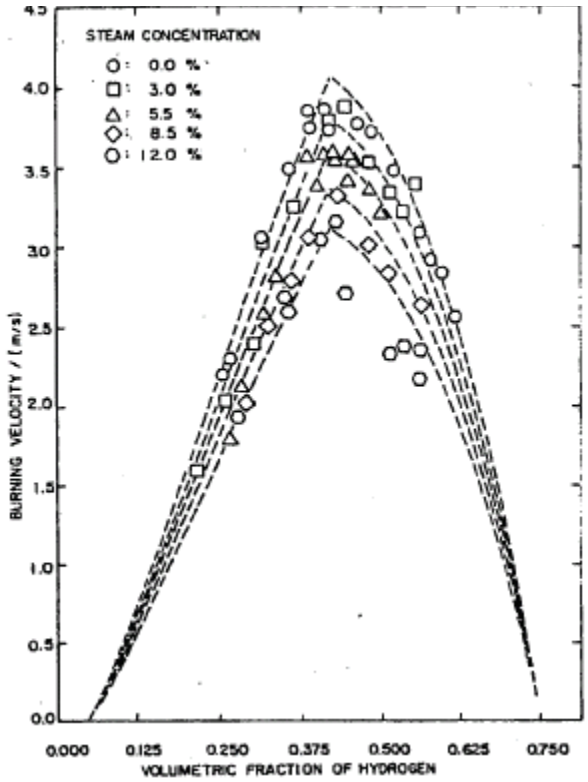


Figure 11-25 Laminar flame velocities, *T* = 50°C, [16]

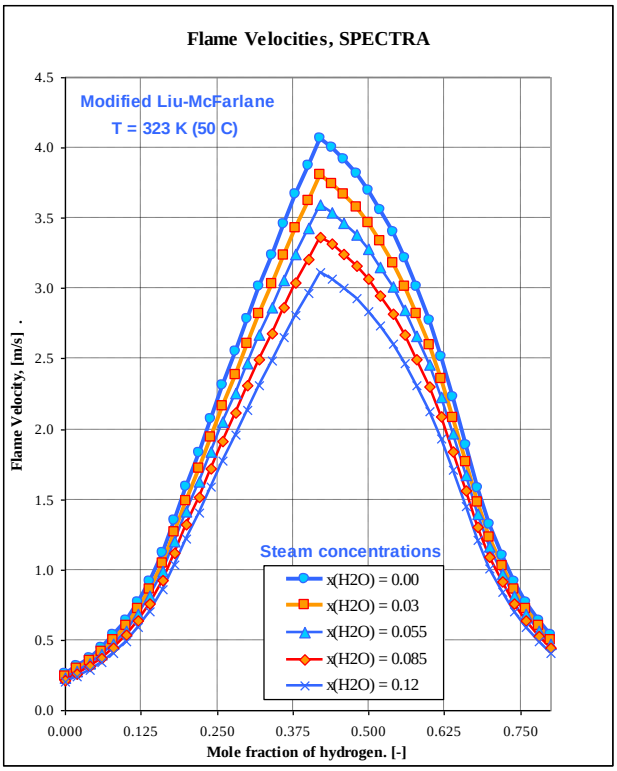


Figure 11-26 Laminar flame velocities, *T* = 50°C, SPECTRA

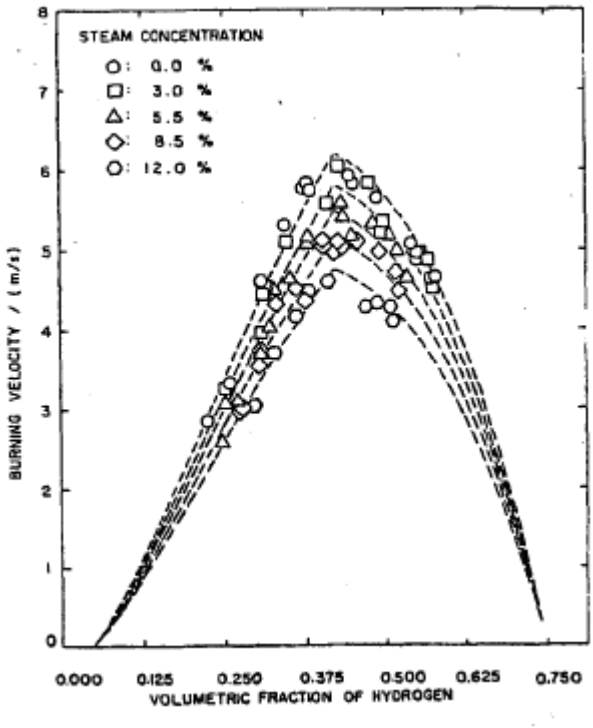


Figure 11-27 Laminar flame velocities, *T* = 150°C, [16]

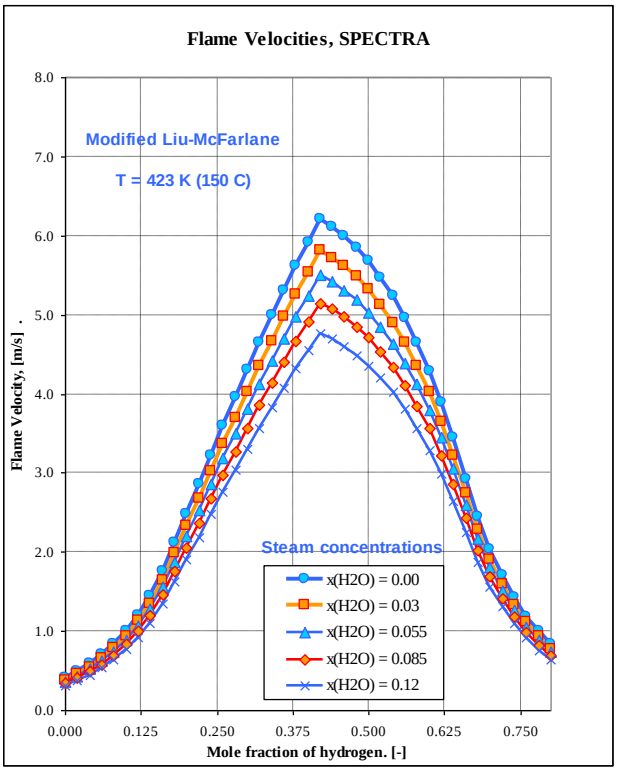


Figure 11-28 Laminar flame velocities, *T* = 150°C, SPECTRA

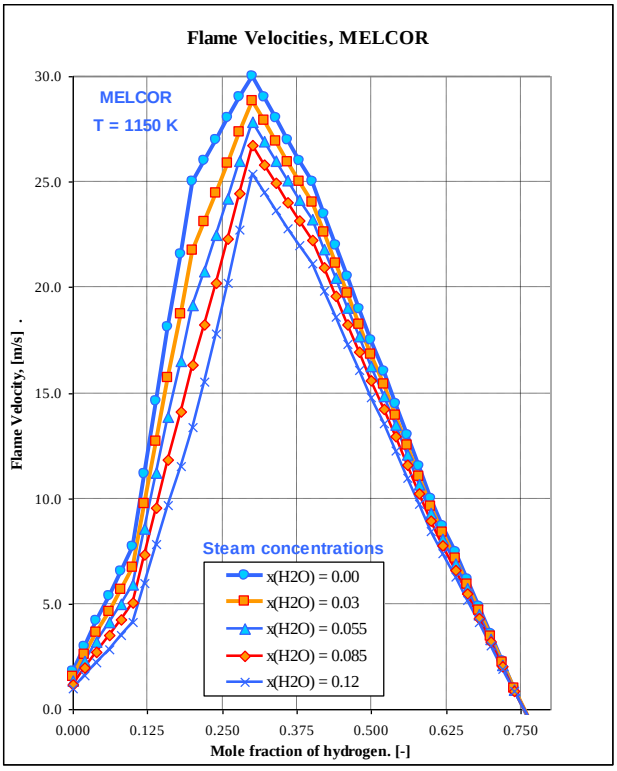


Figure 11-29 Laminar flame velocities, *T* = 1150 K, MELCOR

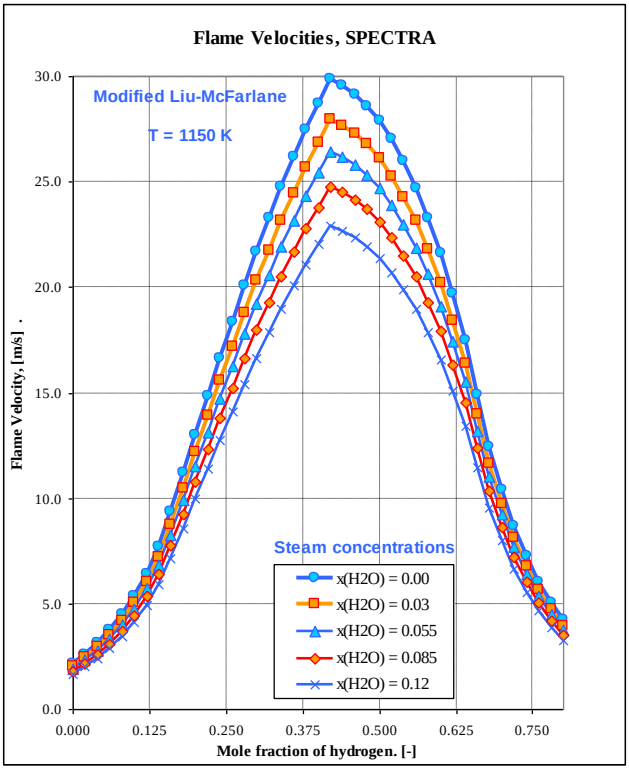


Figure 11-30 Laminar flame velocities, *T* = 1150 K, SPECTRA

Figure 11-29 and Figure 11-30 show comparison of the model coded in SPECTRA with the results obtained from the model applied in MELCOR (the HECTR 1.5 model). The model applied in SPECTRA gives results similar to the HECTR 1.5 model for the gas temperature equal to about 1150 K. It is seen that in the low hydrogen fraction region the modification to the original Liu-McFarlane model provides better agreement with the HECTR 1.5 model. (In the original Liu-McFarlane model the flame speed decreases to zero at approximately 6% hydrogen fraction - Figure 11-23, Figure 11-25, Figure 11-27.)

In SPECTRA one can use two options for the laminar flame velocity calculations; the current gas temperature in a Control Volume or a constant, user-defined temperature, may be used. The gas temperature in a Control Volume is an average over the whole volume, which may be very different from the local temperature during burn. Therefore the second option is used by default, with the gas temperature equal to 1150 K (see Volume 2). With this setting the SPECTRA model gives results similar to the MELCOR model, with MELCOR giving somewhat higher velocity for the practically interesting hydrogen fractions of 10 - 20% - see Figure 11-29 and Figure 11-30. As shown in Volume 3, this setting is more conservative, i.e. it gives higher flame speeds than using the CV gas temperature.

- **Turbulent Flame Speed**

When turbulence is relatively small, then the turbulent burning velocity may be estimated using a correction factor to the laminar burning velocity (see [17], section 10.3.4). The method is described below. It should be noted that this method is applied in SPECTRA only for slow deflagrations. In case of fast turbulent deflagrations a different method is used, described in section 11.4.2. The turbulent burning velocity, v_{tur}, may be related to the laminar burning velocity, v_{lam}, using one of the following expressions:

$$v_{tur} = v_{lam} \cdot f(q, v_{tur})$$
$$v_{tur} = v_{lam} \cdot g(q, v_{lam})$$

where $f(q, v_{tur})$, $g(q, v_{lam})$ are functions depending on turbulent kinetic energy density, q, and either turbulent or laminar flame velocities.

For isotropic turbulence the kinetic energy density, q, is given by the following expression [17]:

$$q = \frac{3}{2} \cdot (v')^2$$

where v' is the turbulent velocity, (m/s).

The function $g(q, v_{lam})$ is used in the present model, since it allows to obtain explicit expression for v_{tur}. Several expressions have been proposed for the function $g(q, v_{lam})$. Two correlations are used by the present model. The Klimov correlation [18] is:

$$g(q, v_{lam}) = C \cdot \left(\sqrt{q} / v_{lam}\right)^{0.7}$$

where C is a constant. The recommended value of C is 2.4 [17].

With the Klimov correlation the turbulent velocity approaches zero when the turbulent energy approaches zero. In reality the turbulent velocity should approach the laminar velocity in such case. This is achieved by using a second correlation, described below.

The second correlation built into the SPECTRA model has the following form [17]:

$$g(q, v_{lam}) = \left[\frac{1}{2} \cdot \left(1 + \sqrt{1 + (8/3) \cdot C \cdot (2q / v_{lam}^2)} \right) \right]^{1/2}$$

The constant C in the above formula has been set to the same value as that recommended for Klimov correlation. The value of this constant has little practical meaning since, as shown below, this correlation is applied for low turbulence, where the value of g is close to 1.0.

To obtain the turbulent flame speed the code selects the correlation which gives the larger value of v_{tur}. In practice this means that the Klimov correlation is used for large turbulent velocities, while the second correlation is used for small turbulent velocities. Figure 11-31 shows $g(q, v_{lam})$ for turbulent velocities from 0 to 5 m/s. The Klimov correlation is used for v' larger than approximately 0.2 m/s.

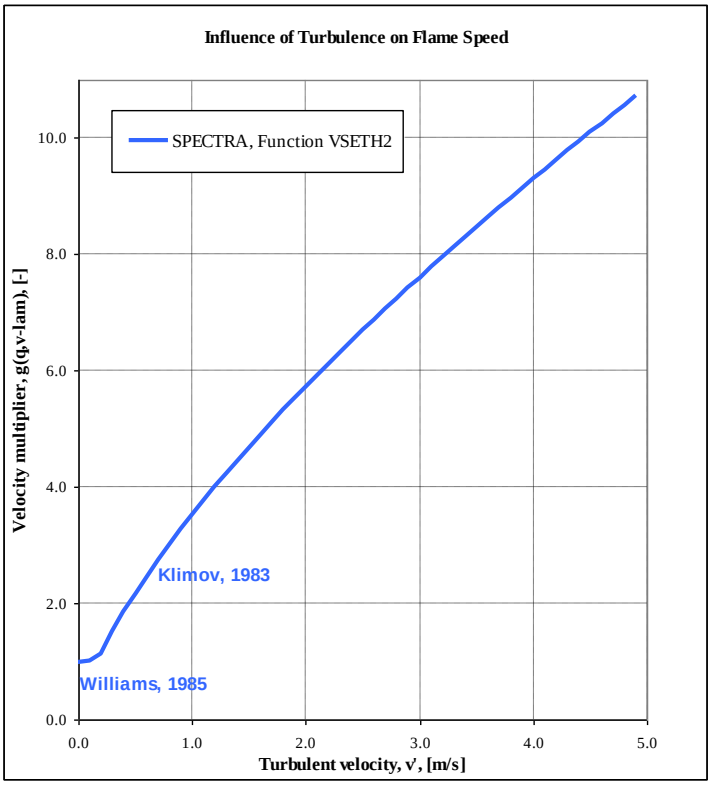


Figure 11-31 Influence of turbulence on flame speed

To apply the above correlations one needs to know the turbulence energy, *q*, which is determined by the turbulent velocity, *v'*. The Control Volume Package in SPECTRA code does not calculate the turbulent velocities. Only average CV velocities are available in the code. To obtain an estimation of turbulent velocities a series of CFD calculations have been performed with the STAR-CD 3.10 code, for a geometry representative for a containment compartment. The results of CFD calculations were plotted to correlate an average gas velocity in a compartment with the maximum local gas velocity, and the maximum local velocity with the maximum turbulent velocity. The main results are shown in Figure 11-32 and Figure 11-33.

Figure 11-32 shows local velocities for several different cases characterized by different average velocities and different gas temperatures. It is seen that the maximum local velocity can be approximated by a line:

$$v_{\max} = C_1 \cdot \bar{v}$$

where $v_{\max}$ is the maximum local velocity, $\bar{v}$ is the average velocity, and C_1 is a constant, approximately equal to 8.0.

Figure 11-33 shows turbulent velocities plotted versus maximum local velocities. The relation between these quantities can be approximated by a line:

$$v' = C_2 \cdot v_{\max}$$

where C_2 is a constant, approximately equal to 0.45.

The values of constants C_1, C_2, are entered in input data separately for each Control Volume (Volume 2). The default values are 8.0 and 0.45 for C_1 and C_2 respectively. The turbulent velocity in a Control Volume is obtained in the program from:

$$v' = C_1 \cdot C_2 \cdot \sqrt{v_{hor}^2 + v_{ver}^2}$$

v_{hor} horizontal velocity of atmosphere gas in a Control Volume, (m/s)
v_{ver} vertical velocity of atmosphere gas in a Control Volume, (m/s)

The method of calculating average CV velocities in horizontal and vertical directions are described in section 2.5.

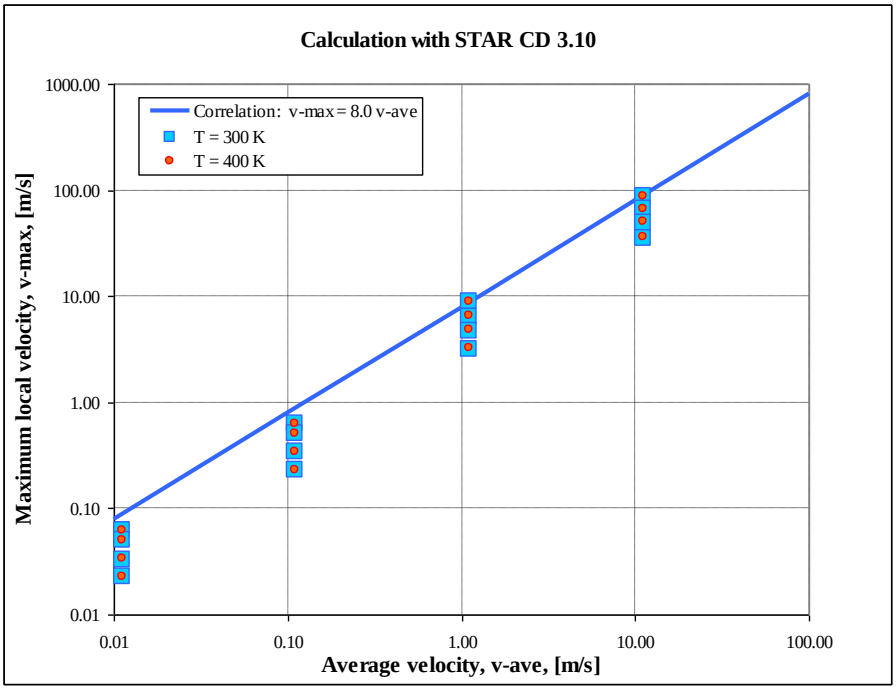


Figure 11-32 Maximum local velocity versus average velocity

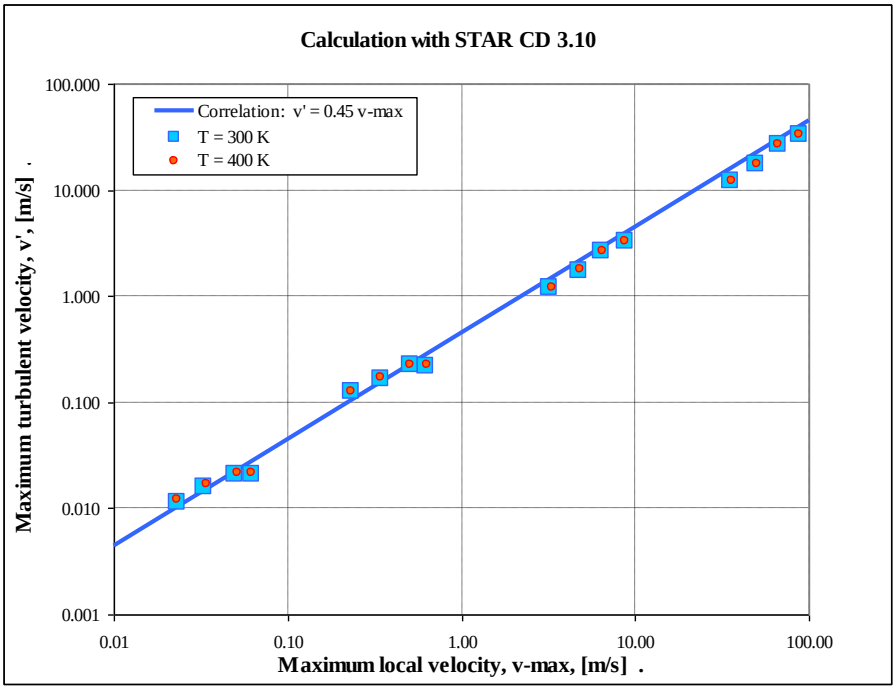


Figure 11-33 Maximum turbulent velocity versus maximum local velocity

11.4.2 Flame Velocity During Fast Turbulent Deflagrations

During fast turbulent deflagrations the flame propagates with supersonic velocity. The flame velocities are typically above 500 m/s [12]. The SPECTRA code calculates the flame velocity from a correlation developed based on data shown in [14], section 2. For conservatism the upper limit velocities were used. The correlation is:

$$v_{FTD} = A + B \cdot x_{H_2} + C \cdot x_{H_2}^2$$

where *A*, *B*, *C* are constants, equal to:

- *A* = +300.0
- *B* = +6500.0
- *C* = –7000.0

For additional conservatism a lower limit of 500.0 m/s is imposed on the values obtained from the above correlation. The calculated flame velocities are shown in Figure 11-34. The maximum values of flame velocities selected from the data reported in [14] are shown for comparison.

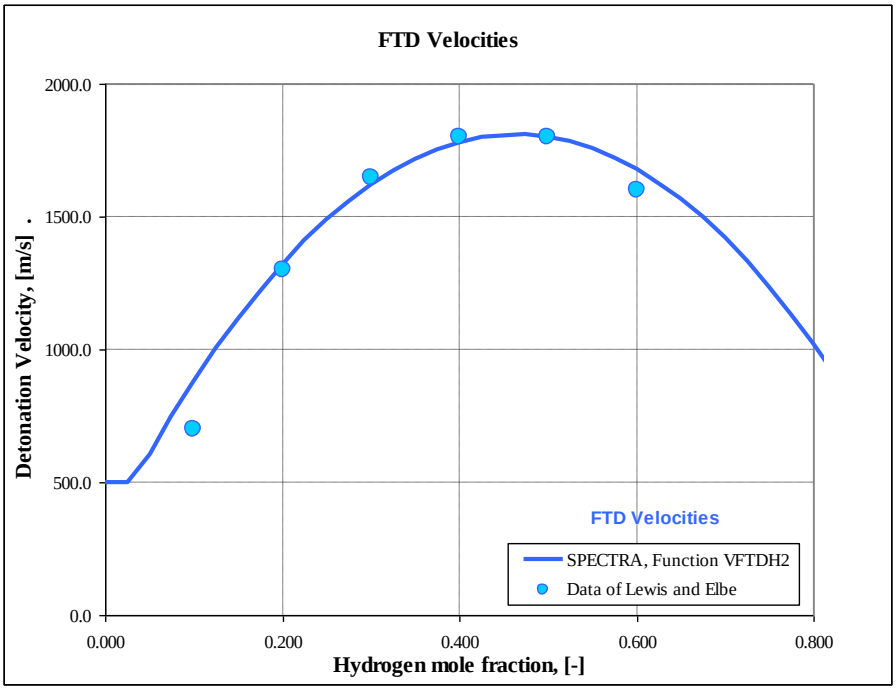


Figure 11-34 Flame velocity, fast turbulent deflagrations

11.4.3 Flame Velocity During Detonations

The detonation velocity is calculated in the SPECTRA code from a correlation developed based on data shown in [15], Chapter VIII, table 4. The correlation is:

$$v_{\text{det}} = v_0 \cdot f_p \cdot f_T \cdot f_{H_2}$$

in the above formula v_0 is a constant, equal to 2790.0 m/s, while f_p, f_T, f_{H2} are functions, taking into account the influence of pressure, temperature, and gas composition, respectively. The functions f_p, f_T, f_{H2} are given by:

$$f_p = 1 + 0.05 \cdot \left(\frac{p}{1.013 \times 10^5} \right) - 0.03 \cdot \left(\frac{p}{1.013 \times 10^5} \right)^2$$

$$f_T = 1 - 1.0 \times 10^{-4} \cdot (T - 273.0)$$

$$f_{H_2} = 1 + 0.87 \cdot (x_{H_2} - 0.667)$$

p pressure, (Pa)
T temperature, (K)
x_{H2} hydrogen mole fraction, (-)

The detonation velocities calculated from the above correlation are shown in Figure 11-35, Figure 11-36, and Figure 11-37. The figures show also measurement data, reproduced from [15]. The discrepancies between the developed correlation and the experimental data from [15] are within 1-2%, except for the region of 80-90% hydrogen fractions, where the discrepancies are 6-7%. This region is however not interesting from the practical standpoint.

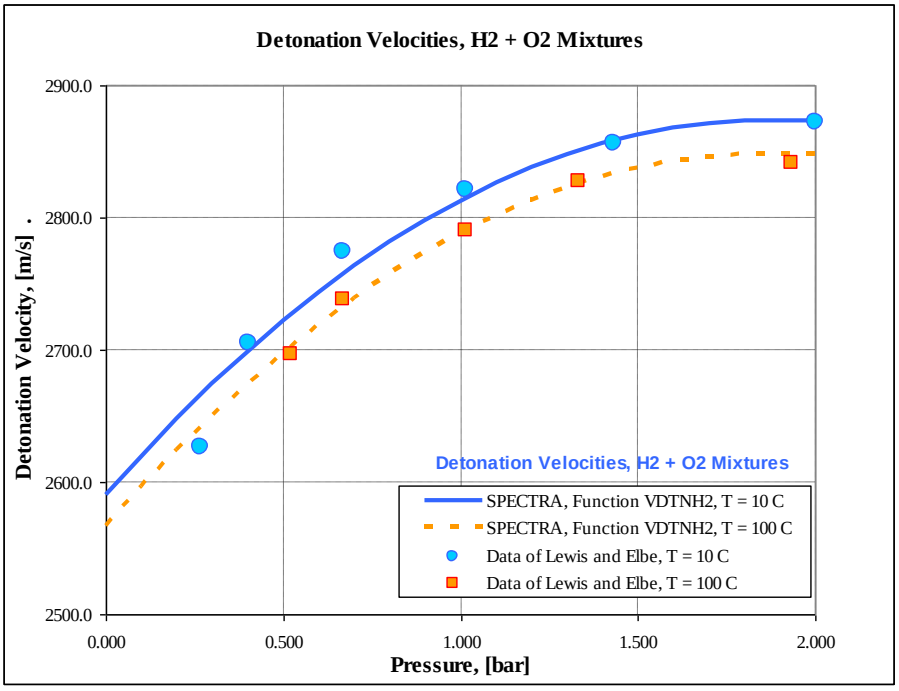


Figure 11-35 Detonation velocity, $2H_2 + O_2$ mixtures

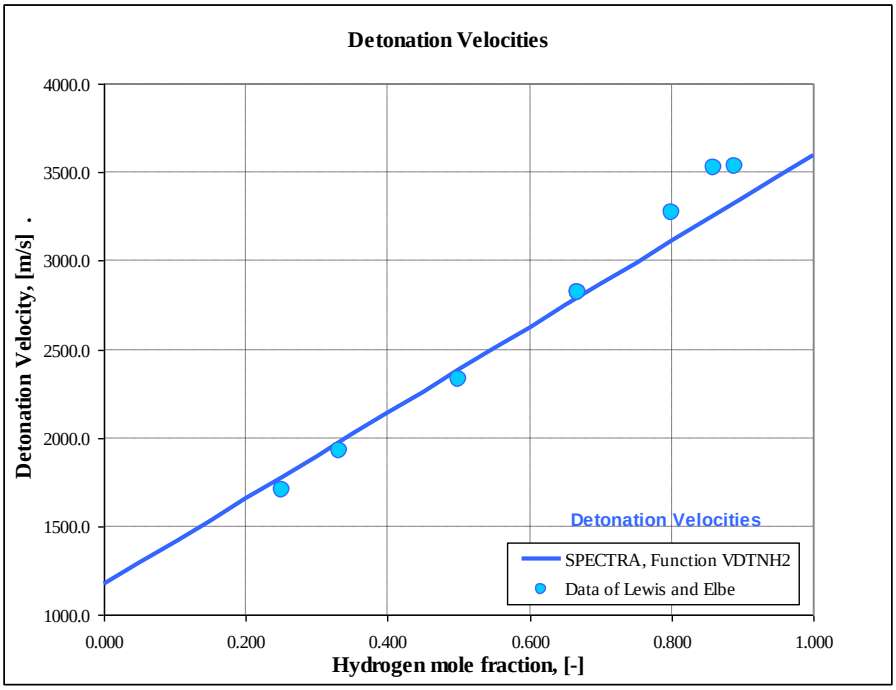


Figure 11-36 Detonation velocity, p = 1 bar, T = 373 K, influence of H_2

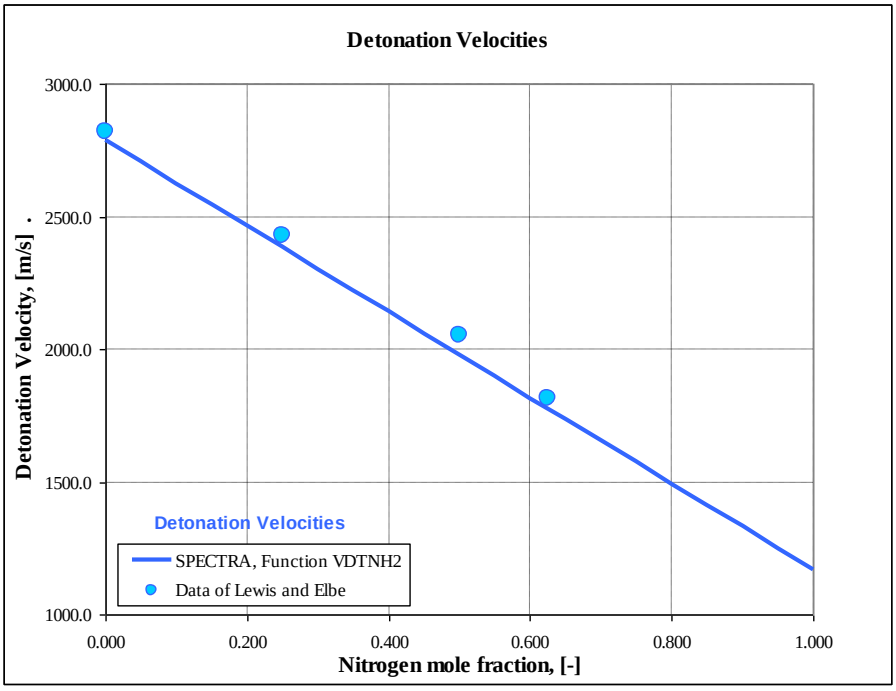


Figure 11-37 Detonation velocity, p = 1 bar, T = 373 K, influence of N_2

11.5 Combustion Completeness

The combustion process does not have to be complete, that is not all hydrogen has to be burned. The term combustion completeness is used to determine the amount of combustible gases at the end of combustion. With hydrogen being the only combustible gas, the combustion completeness, *CC*, is defined as follows.

$$CC = 1 - \frac{x_{H_2}^{EOB}}{x_{H_2}^{0}}$$

where $x_{H_2}^{EOB}$ is the hydrogen mole fraction at the end of burn, (-), and $x_{H_2}^{0}$ is the initial hydrogen mole fraction, (-). Two options are available to calculate combustion completeness. A correlation from HECTR 1.5 [6] correlation, which is also applied in the MELCOR code, gives combustion completeness as a function of initial hydrogen fraction, as:

$$CC = 23.4 \cdot (x_{H_2}^{0} - 0.0375)$$

The following lower and upper limits are imposed on the value obtained from the above correlation:

$$0.001 < CC < 0.999$$

The values of *CC* obtained from the above correlation are plotted in Figure 11-38.

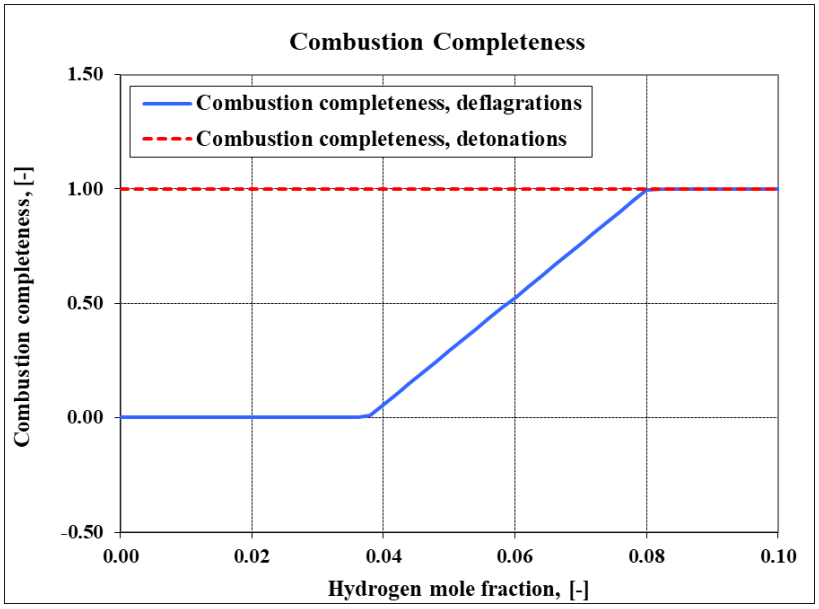


Figure 11-38 Combustion completeness.

As a second option, a constant value of *CC*, defined by the user, may be used. Default selection is a constant value of 0.99, thus nearly a complete combustion. This is done to obtain conservative loads during hydrogen combustion.

Hydrogen burn is finished when the hydrogen concentration falls below the value determined by the selected combustion completeness, namely:

$$x_{H_2} = x^0_{H_2}(1 - CC)$$

An additional check is made if the mixture is combustible. If the mixture is combustible, then the burn continues in spite of the value obtained from combustion completeness criterion. Inclusion of this additional check was motivated by an observation that in spite of the hydrogen depletion the gas mixture may still be flammable, because at high temperatures the flammability limits significantly change (see Figure 11-4). Termination of burn when the mixture is still flammable would result in burn starting again in the following time step. The burn would then go off and on in subsequent time steps. To avoid this un-physical behavior, the burn stops only when both combustion completeness criterion and inflammable gas criterion are satisfied - see Volume 3. Note that in codes like MELCOR such behavior is not a concern, because the temperature dependence of the flammability limit is not taken into account.

A weakness of the combustion completeness correlation is the fact that it is not temperature dependent. Therefore it is not used in the program by default, but as an optional model. For conservative analyses the default value of *CC* (0.99) should be used. For more realistic calculation a small value of *CC* may be selected (for example 0.1). In such case the burn will continue as long as the gas mixture is flammable.

11.6 Shock Waves During Fast Turbulent Deflagrations and Detonations

During fast turbulent deflagrations and detonations the flame propagates with supersonic velocities. In such case a shock wave is created, which poses a significant threat to containment integrity. Detailed analysis of shock waves is rather complex. CFD calculations are needed to calculate mechanistically the shock wave propagation and peak pressure. In the SPECTRA model the shock wave pressure is estimated in a simplified way, described below.

The shock wave pressure is calculated using the equation derived for a perfect gas mixture. The shock wave pressure, p_{SW}, is obtained from the following formula ([11], section 6.4):

$$p_{SW} = p \cdot \frac{\gamma}{\gamma + 1} \cdot \left(2C_M \cdot Ma^2 - \frac{\gamma - 1}{\gamma} \right)$$

γ	ratio of specific heats, c_p/c_v, (-)
Ma	Mach number, equal to the flame speed (calculated as described in sections 11.4.2, 11.4.3), divided by the speed of sound, (-)
p	pressure behind the shock wave, (Pa)
C_M	optional multiplier, (-)

The pressure behind the shock wave, p, is assumed to be equal to the current Control Volume pressure, calculated by the code. The speed of sound is calculated for the current gas conditions in a Control Volume.

The multiplier C_M is provided for additional conservatism. The default value is 1.4 (see Volume 2). With this value the shock wave pressure estimation is roughly 40% larger than would be obtained from the ideal gas equation. Use of this rather large multiplier is considered necessary, because of the relatively simple method used to estimate the shock wave pressure.

11.7 Flame Propagation Between Control Volumes

If a burn occurs in one Control Volume, the flame may propagate to the neighboring Control Volumes if certain conditions are satisfied. The method of calculating flame propagation between Control Volumes is described in this section.

For each junction connected to a given Control Volume the user specifies a propagation distance, L_{prop}. Default value of L_{prop} is equal to 80% of the burn characteristic dimension (see Volume 2). A flame may propagate to a neighboring Control Volume if the current flame propagation length, L_{flame}, is larger than L_{prop}:

$$L_{flame} = \int_{t_0}^{t} v_{flame}(t')dt' \geq L_{prop}$$

In the above equation $v_{flame}(t')$ is the flame velocity at time t'. The integration is performed from the time of the start of burn, t_0, to the current time, t.

If $L_{flame} \geq L_{prop}$ then the program checks the flammability limit in the neighboring Control Volume. The flammability limits are different in case of upward propagation and downward propagation (see section 11.2.2, Figure 11-4). In case of a vertical junction, the upward propagation limit is used if the flame has to propagate upwards to reach the neighboring Control Volume, and the downward propagation limit is used if the flame has to propagate downwards to reach the neighboring Control Volume. In case of horizontal junctions, the average value of the upward and the downward propagation limits is used.

12 Radioactive Particle Transport Package

12.1 Introduction

The Radioactive Particle Transport Package calculates transport of fission products, including:

- aerosol transport, deposition, and resuspension,
- fission product release, transport, condensation, sorption on surfaces, diffusion inside solid surfaces.

SPECTRA is basically a thermal-hydraulic tool, not a severe accident tool, such as for example MELCOR [46]. SPECTRA is foreseen for thermal-hydraulic analyses, including temporary core uncovery, leading to oxidation of cladding due to metal-steam reaction and limited release of fission products. For example, in some cases of LOCA in a LWR reactor the core can be uncovered for a sufficiently long time to heat up the fuel elements and release some of the most volatile fission products. SPECTRA is also applicable for analyzing release and transport of radioactive fission products during normal operation. For example, in case of HTR/PBMR reactors a significant amount of dust is generated due to pebble-to-pebble scratching. The radioactive fission product release, although small during normal operation, may eventually pose a threat to the environment because they will be attached to the dust particles and deposited throughout the primary system. In case of LOCA the large amount of radioactive dust can therefore be released to the environment.

Because SPECTRA is foreseen for different applications than MELCOR, the treatment of the radioactive isotopes is somewhat different.

- MELCOR, which is a severe accident code, models the whole core inventory of radioactive isotopes. They are divided into fission product classes, each class combining radioactive isotopes with similar properties. The radioactive decay of isotopes is not modeled (although the decay heat is taken into account). Thus for example if a certain total mass of the Iodine class is present initially in the core, this mass will be conserved (the iodine will not decay to form Xenon, etc.) This is a necessary simplification resulting from the huge amount of isotopes that need to be tracked in case of a severe accident.
- In SPECTRA only a limited number of radioactive isotopes is tracked. Since SPECTRA is not a severe accident code, there is no need to define the full core inventory. In case of normal operation or design accidents only a few isotopes are important. These isotopes can be tracked in SPECTRA in a more detail, including transformation of isotopes resulting from the radioactive decay. The isotope transformations are calculated using the isotope chains, such as those described in Chapter 9.

Compared to MELCOR, SPECTRA has a few advantages related to aerosol behavior, like:

- Decay of radioactive isotopes is tracked.
- Models exist applicable to circulating fuel (molten salt reactors).

Summarizing, concerning the radioactivity transport, MELCOR can do more (core melt-down, release of all radioactive isotopes) but with smaller accuracy, while SPECTRA can do less but with more accuracy (isotope decay chains).

12.2 Aerosol Dynamics

The aerosol particles, which in general are very different in size, are represented in the code using discrete size sections, described in section 12.2.1. The aerosol dynamics equation is presented in section 12.2.2, while the coagulation kernels are described in section 12.2.3. The aerosol dynamics equation is very similar to that from MELCOR, but in SPECTRA it is formulated in terms of particle density rather than particle mass. The aerosol dynamics calculations are performed using inter-volume aerosol transport, calculated as described in section 12.2.4, deposition of the aerosol particles on solid structures and water pools, described in section 12.2.5, and resuspension, described in section 12.2.7. Aerosol removal by pool scrubbing and by an aerosol filter is described in section 12.2.9 and 12.2.10 respectively.

12.2.1 Aerosol Size Distribution

The aerosol particles are discretized into size sections. Each size section is characterized by its characteristic (average) particle size, diameter, shape factors, etc. The maximum number of aerosol size sections is 20. By default 5 sections are used with the following average parameters:

Table 12-1 Default aerosol size sections.

Parameter	Size Section				
	1	2	3	4	5
Diameter, d_p, m	1.0×10^{-6}	3.0×10^{-6}	1.0×10^{-5}	3.0×10^{-5}	1.0×10^{-4}
Volume, V_p, m^3	5.2×10^{-19}	1.4×10^{-17}	5.2×10^{-16}	1.4×10^{-14}	5.2×10^{-13}
Dynamic shape factor, χ	1.0	1.0	1.0	1.0	1.0
Agglomeration shape factor, γ	1.0	1.0	1.0	1.0	1.0

Note that in contrast to MELCOR each size section is characterized by a single (representative) particle diameter. The aerosol dynamics parameters (agglomeration kernels, deposition velocities, resuspension, etc.) are calculated for each size section using these representative diameters. In MELCOR on the other hand the size sections are determined by specifying the minimum and the maximum diameters. When SPECTRA results are compared to MELCOR the user must therefore be aware how the minimum/maximum diameters from MELCOR translate into the representative values for SPECTRA. This issue is discussed in detail in Volume 3.

12.2.2 Aerosol Dynamics Within a Single Control Volume

For each Control Volume the aerosol dynamics equation is written in terms of aerosol particle density, expressed in particles per cubic meter. Before the aerosol dynamics equation is given, some basic definitions are presented.

Definitions

In the following discussion the density of particles of the size *i* is denoted as:

- **n_i number of aerosol particles of section *i* per cubic meter, (1/m^3).**

The coagulation kernel is expressed in the corresponding units:

- **K_{ij} coagulation kernel, (m^3/s).**

The coagulation kernel represents the collision frequency, i.e. the number of collisions per second and per cubic meter for the particle densities equal to one. The total number of collisions is equal to ([74], section 4.1):

- **$K_{ij} \cdot n_i \cdot n_j$ number of collisions per second, per unit volume, for particles of section *i* colliding with particles of section *j* ($i \neq j$), (1/m^3-s).**

The above formula is valid for distinguishable particles (that is if $i \neq j$). If the colliding particles are indistinguishable (that is if $i=j$), then every collision is counted twice and therefore the total number of collisions is given by:

- **$\frac{1}{2} \cdot K_{ii} \cdot n_i \cdot n_i$ number of collisions per second, per unit volume, for particles of section *i* colliding with particles of section *i*, (1/m^3-s).**

If every collision resulted in removing the particle from its section, then the rate of removal of particles due to collision would be given by the above expressions. This is however not the case for arbitrary size sections. For example, in the default settings the volume of the size section 2 is about 27 times larger than the volume of the size section 1 (see Table 12-1). Therefore one needs 26 collisions before a single particle of section 2 is created. This fact is introduced in the model by defining the section transfer probabilities

- **P_{ijk} probability of the fact that a collision of particles from section *i* with a particle from section *j* will result in creating a particle from section *k*.**

It may easily be shown that the amount of collisions to create a particle of a given size is always the same, independently of how the collisions proceed. For example, suppose there are two size sections, with the volumes of the section 1 and 2 equal to 1.0 and 4.0 respectively (in an arbitrary unit). The collisions may proceed as follows:

```
1                          or:   1
  > 2                            > 2
1                                1     > 3
        > 4                          1     > 4
1                                      1
  > 2
1
```

In each case exactly three collisions are needed to form a particle of the section 2. (A formal proof that the number of collisions is always the same may be given by considering mass balance of colliding particles.) In general, the number of collisions is given by the following formula:

$$N = \frac{V_k - V_{ave}}{V_{ave}}$$

where V_k is the volume of the size section k (the final particle), and V_{ave} is the average volume of the size sections i, j (the colliding particles), given by:

$$V_{ave} = \frac{1}{2}(V_i + V_j)$$

In the example presented above V_{ave} = 1.0, and N = (4.0 – 1.0) / 1.0 = 3.0.

The probability of transfer to the next section is of course equal to the inverse of the required number of collisions, so:

$$P_{ijk} = \frac{V_{ave}}{V_k - V_{ave}}$$

This formula is valid for transfer to a higher section, i.e. $k > \min(i, j)$. In order to calculate the probability of a particle remaining at the same section after collision one needs to take into account the fact that the final particle must be placed in some size section. This condition is written as:

$$\sum_k P_{ijk} = 1.0$$

The above equations allow to calculate all probabilities for all size sections. In the example presented above the probability of transfer from section 1 to 2 is equal to P_{112} = 1.0/(4.0–1.0) = 0.333, while the probability of remaining at the same section is P_{111} = 1.0 – 0.333 = 0.667. The code calculates internally all probabilities and prints them in the output file during input processing.

Using the section transfer probabilities the effective number of collisions that result in particle transfer to section k is written as:

- **$K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j$ number of particles of size section *k*, created per second, per unit volume, by collisions of particles from the sections *i* and *j*, (1/m^3-s).**

Agglomeration

First, let's calculate how many particles are removed from a given section due to collisions. Particles of section *k* collide with particles from all sections, including *k*. The total number of collisions resulting in removal of particles from section *k* is therefore equal to the sum:

$$\frac{1}{2} \cdot K_{kk} \cdot (1 - P_{kkk}) \cdot n_k \cdot n_k + \sum_{i \neq k} K_{ik} \cdot (1 - P_{ikk}) \cdot n_i \cdot n_k$$

Where P_{ikk} is the probability that after the collision of particles *i* and *k*, the resulting particle will remain in section *k*. The term with K_{kk} is written separately, because it must be multiplied by 1/2 (see the discussion of the coagulation kernel above).

The total number of particles of section *k* removed from this section is equal to:

- Number of collisions, in case of collisions with other particles, $i \neq k$.
- Twice the number of collisions, in case of collisions with the same particles, $i = k$.

Therefore the removal of particles from the size section *k* is given by:

$$Removal(k) = \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot n_i \cdot n_k$$

Next, let's calculate the source of particles in the section *k* due to all collisions. The particles of the size section *k* can be produced by collisions of all sections smaller than *k*, with particles smaller than *k* as well as *k* itself. Therefore it is obtained from the following double sum:

$$\frac{1}{2} K_{ii} \cdot P_{iik} \cdot n_i \cdot n_i + \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j$$

Particles *k* colliding with *k* cannot give source to *k*, so the first sum is restricted to *k*–1. Double counting of collisions is avoided by starting the second sum from *i*. The first term ($j = i$) is written separately before the double summation, since this term must be multiplied by ½ (see the discussion of coagulation kernel above).

In order to calculate the number of particles created in the section *k*, let's first consider again the example case with two size sections, with section volumes of 1.0 and 4.0. Suppose there are 100 collisions among the particles of the smaller section (1.0) leading to creation of particles of the higher section (4.0): $½ \cdot K_{11} \cdot P_{112} \cdot n_1 \cdot n_1 = 100$. The number of particles removed from the lower section is 200 (two particles are removed per singe collision). The total volume of the removed particles is 200·1.0 = 200. The particles of the higher section have the volume of 4.0; therefore the number of produced particles is equal to 200 / 4.0 = 50.

In general, the number of produced particles is obtained by multiplying the number of collisions by the ratio of volumes: $(V_i + V_j)/V_k$:

$$Source(k) = \frac{1}{2} K_{ii} \cdot P_{iik} \cdot n_i \cdot n_i \cdot \left(\frac{V_i + V_i}{V_k} \right) + \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j \cdot \left(\frac{V_i + V_j}{V_k} \right) \qquad j \neq k$$

The above formula is wrong for one special case, when *i* is colliding with *k* to form *k* (*j* = *k*). In such case using the volume ratio of $(V_i + V_k)/V_k$ would be wrong because the particles *k* were already in this section so they do not contribute to the source. Therefore for this particular case the ratio V_i/V_k must be taken instead of $(V_i + V_k)/V_k$. A general expression can be written as:

$$Source(k) = \frac{1}{2} K_{ii} \cdot P_{iik} \cdot n_i \cdot n_i \cdot \left(\frac{V_i + V_i}{V_k} \right) + \sum_{i=1}^{k-1} \sum_{j=1, j \neq i}^{k} K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j \cdot \left(\frac{V_i}{V_k} \right)$$

The summation over *j* is now performed from 1, so the collisions are counted twice (*i* with *j* and *j* with *i*) but each time only the volume of one of the colliding particles (particles *i*) contribute to the source. The first term (*ii*) may be written as: $K_{ii}\ P_{iik}\ n_i\ n_i\ (V_i/V_k)$, and inserted into the summation. The final expression for the agglomeration source for section *k* becomes:

$$Source(k) = \sum_{i=1}^{k-1} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j \cdot \left(\frac{V_i}{V_k} \right)$$

Correctness of the above expressions is easiest verified by checking if the total mass of aerosols is conserved during the coagulation process. SPECTRA prints out the total aerosol mass and the mass conservation was checked in multiple test runs, shown in Volume 3.

Aerosol Dynamics Equation

Within a single Control Volume the aerosol dynamics equation is written as:

$$\frac{dn_k}{dt} = \sum_{i=1}^{k-1} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot n_i \cdot n_j \cdot \left(\frac{V_i}{V_k} \right) - \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot n_i \cdot n_k + S_E + S_R - D + R + F$$

n_k	concentration of aerosols of the size section *k*, ($1/m^3$)
S_E	external source (user-defined tabular or control functions), ($1/m^3$-s)
S_R	source due to fission product release, ($1/(m^3$-s)
D	removal due to all deposition mechanisms, ($1/(m^3$-s)
R	source due to resuspension of deposited aerosols, ($1/(m^3$-s)
F	net source due to inter-volume flows (transport of particles with the gas flow), ($1/(m^3$-s)

The first and the second term on the right hand side of the above equation represent the source and the removal of particles due to agglomeration, described above.

The above equation is written in a finite difference form, by replacing the derivative by the difference:

$$\frac{dn_k}{dt} = \frac{n_k - n_k^0}{\Delta t}$$

n_k particle concentration, new value (1/m^3)
n^0_k particle concentration, old time step value (1/m^3)
Δt time step size, (s)

In order to solve the equation in an implicit way the right hand side terms need to be written using n_k, rather than the old (known) value, n^0_k. As a consequence one obtains a non-linear set of equations. In MELCOR this equation set is solved using the Runge-Kutta method. In SPECTRA a different approach is adopted. The non-linear coagulation terms are linearized. The linearized product of particle densities is written as:

$$n_i \otimes n_j \equiv \frac{1}{2}(n_i^0 \cdot n_j + n_i \cdot n_j^0)$$

As a result one obtains a linear set of equations, which is then easily solved by one of the standard solver (see section 17.4). A large amount of test runs performed showed that this method provides a fast and stable solution even with very large time steps.

After linearization the finite difference version of the aerosol dynamics equation is:

$$\frac{n_k}{\Delta t} - \frac{n_k^0}{\Delta t} =$$
$$= \sum_{i=1}^{k-1} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot \frac{1}{2}(n_i^0 n_j + n_i n_j^0) \cdot \left(\frac{V_i}{V_k}\right) - \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot \frac{1}{2}(n_i^0 n_k + n_i n_k^0) +$$
$$+ S_E + S_R - D + R + F$$

This equation is rearranged to give:

$$n_k \left(\frac{1}{\Delta t} + \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot \frac{1}{2} n_i^0 \right)$$
$$- \sum_{i=1}^{k-1} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot \frac{1}{2} n_i^0 n_j \cdot \left(\frac{V_i}{V_k}\right) - \sum_{i=1}^{k-1} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot \frac{1}{2} n_j^0 n_i \cdot \left(\frac{V_i}{V_k}\right)$$
$$+ \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot \frac{1}{2} n_k^0 n_i = \frac{n_k^0}{\Delta t} + S_E + S_R - D + R + F$$

The above set of equations can be written shortly in a matrix form:

$$A \cdot N = B$$

where ***N*** is a vector of unknown particle concentrations, n_i, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. The elements of matrix ***A*** are equal to:

$$a_{ki} = \left(\frac{1}{\Delta t} + \frac{1}{2} \sum_{i=1}^{N} K_{ik} \cdot (1 - P_{ikk}) \cdot n_i^0 \right) \delta_{ki} + \\ + \frac{1}{2} K_{ik} \cdot (1 - P_{ikk}) \cdot n_k^0 - X_1 - X_2$$

where δ_{ki} is the Kronecker delta:

$$\delta_{ki} = \begin{cases} 1.0 & if \quad k = i \\ 0.0 & if \quad k \neq i \end{cases}$$

and X_1 X_2 are the following sums:

$$X_1 = \begin{cases} \frac{1}{2} \sum_{j=1}^{k} K_{ij} \cdot P_{ijk} \cdot \frac{V_i}{V_k} \cdot n_j^0 & if \quad i < k \\ 0.0 & if \quad i \geq k \end{cases}$$

$$X_2 = \begin{cases} \frac{1}{2} \sum_{j=1}^{k-1} K_{ij} \cdot P_{ijk} \cdot \frac{V_j}{V_k} \cdot n_j^0 & if \quad i \leq k \\ 0.0 & if \quad i > k \end{cases}$$

The elements of vector ***B*** are equal to:

$$b_k = \frac{n_k^0}{\Delta t} + S_E + S_R - D + R + F$$

The aerosol dynamics equation derived for application in the SPECTRA code is somewhat different from the equation used by the MAEROS code, adopted in MELCOR [46]. The equation used in MELCOR is written for aerosol mass densities. In SPECTRA the aerosol number concentrations are used. Because of that, the coagulation kernels are given by formulae shown in the literature, without a need of using conversion factors (see section 12.2.3). Also section-to-section transfer is calculated in MELCOR using a somewhat different method than the presented above concept of the section transfer probabilities. It was verified that in spite of the differences in the equation formulation both codes give very similar results. A simple coagulation test is shown in Volume 3, where SPECTRA results are compared to MELCOR results as well as results of simple hand calculations, finding a good agreement for all calculations. Comparisons between SPECTRA and MELCOR were performed for a large number of more complex cases, and good agreement was found (see Volume 3).

12.2.3 Aerosol Coagulation

When two aerosol particles collide, they can combine to form a larger particle. This process is known as coagulation or agglomeration. The aerosol particles are usually not spherical, and the effective densities may be significantly smaller than the bulk density of the materials of which the aerosols are composed. In aerosol codes, these effects are taken into account by using a formalism based on fully dense spherical aerosols modified through the use of the coagulation shape factor, γ, and the dynamic shape factor, χ. Unit values of the shape factors correspond to dense aerosols of spherical shape, while porous spherical agglomerates lead, in theory, to values greater than unity. Highly irregular aerosols can have shape factors substantially greater than unity.

Some experimental work on irregularly shaped aerosols was carried out by Kunkel [77] and is reproduced in [74] (page 54). The data is shown in Table 12-2. The second and third columns present data reproduced from [74]. The dynamic and the agglomeration shape factors are presented in the next two columns. They were calculated from the data in the previous two columns, using the equations shown in [74]. The equations are not valid for short agglomerates; therefore the shape factors are not exactly 1.0 in the first case.

Table 12-2 Linear chain aerosols. Adopted from [74].

Arrangement	Equal volume radius, r_E, (mm)	Equal fall velocity radius, r_S, (mm)	Dynamic shape factor (-)	Agglomeration shape factor (-)
o	1.00	1.20	0.97	0.95
oo	1.51	1.40	1.16	1.01
ooo	1.72	1.50	1.31	1.30
oooo	1.95	1.56	1.56	1.60
ooooo	2.40	1.64	2.22	2.66

The shape factors are defined in SPECTRA separately for every aerosol size section (see Volume 2). This approach is deemed more appropriate than using the same value for all size sections, as it is done in MELCOR.

The coagulation kernel is calculated in SPECTRA taking into account three mechanisms:

- Gravitational coagulation, K_{grav}
- Brownian coagulation, K_{Brown}
- Turbulent coagulation, K_{turb}

The total coagulation kernel is calculated assuming that the three kernels may be added, similarly as in most aerosol codes:

$$K = K_{grav} + K_{Brown} + K_{turb}$$

The individual coagulation kernels are discussed below.

12.2.3.1 Gravitational Coagulation

The gravitational coagulation kernel is calculated from the following formula ([74], section 4.3):

$$K_{grav} = \varepsilon \cdot \frac{\pi}{4} \cdot \left(\gamma_1 d_1 + \gamma_2 d_2\right) \cdot \left|v_1 - v_2\right| =$$
$$= 0.785 \cdot \varepsilon \cdot \left(\gamma_1 d_1 + \gamma_2 d_2\right) \cdot \left|v_1 - v_2\right|$$

ε	collision efficiency, (-)
d_1	diameter of particle 1, (m)
d_2	diameter of particle 2, (m)
γ_1	coagulation shape factor, particle 1, (-)
γ_2	coagulation shape factor, particle 2, (-)
v_2	gravitational settling velocity, particle 1, (m/s)
v_2	gravitational settling velocity, particle 2, (m/s)
c_s	particle sticking coefficient, (-)

The formula presented in [74] is valid for spherical particles. Compared to this formula, the agglomeration shape factors, γ_1 and γ_2 , were introduced here, following [46]. The particle diameters, the shape factors, as well as the sticking coefficient are user-defined parameters (see Volume 2). The default values of particle diameters and shape factors are given in Table 12-1. The default value of the sticking coefficient is c_s = 1.0. The same value is used in MELCOR [46]. The default value is in agreement with the data shown in literature [78], [79].

The gravitational settling velocity calculation is described below (section 12.2.5). The collision efficiency is calculated in SPECTRA using one of three available models:

- Model of Fuchs, approximation for $d_1 \ll d_2$ ([74], section 4.3):

$$\varepsilon = 1.5 \cdot \left(\frac{d_1}{d_1 + d_2}\right)^2$$

- Model of Fuchs, exact formula ([74], section 4.3):

$$\varepsilon = 1.0 - 1.5 \cdot \left(\frac{d_1}{d_1 + d_2}\right) + 0.5 \cdot \left(\frac{d_1}{d_1 + d_2}\right)^3$$

- Pruppacher-Klett ([74], section 4.3):

$$\varepsilon = 0.5 \cdot \left(\frac{d_1}{d_1 + d_2}\right)^2$$

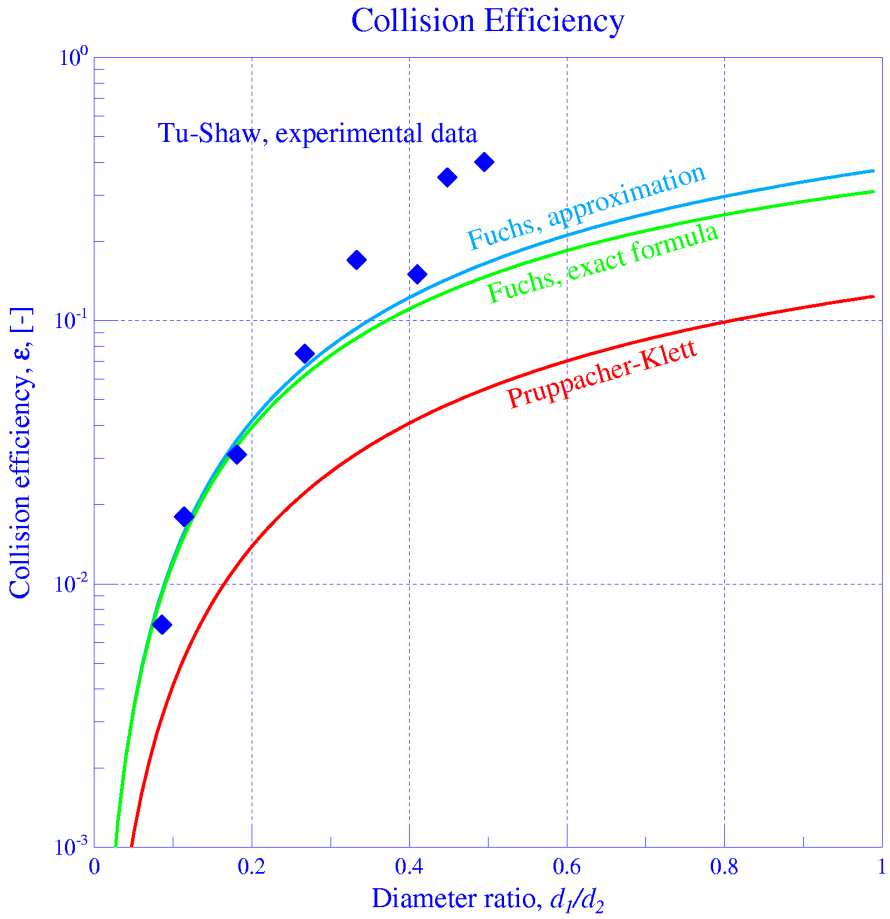


Figure 12-1 Comparison of the collision efficiency models with experimental data.

Results of all three models are shown in Figure 12-1, together with a number of experimental data points. The default model is 1 (see Volume 2). MELCOR uses the same model.

12.2.3.2 Brownian Coagulation

The Brownian coagulation kernel is calculated using one of three available models: the diffusion model, the slip flow model, and the Fuchs model. By default the slip model is used (see Volume 2). The three available models are shortly described below.

- Diffusion Model

The Brownian coagulation is given by the following formula ([74], section 4.2.1):

$$K_{Brown} = \frac{2k_B T}{3\mu_g} \cdot \left(\frac{1}{\gamma_1 d_1} + \frac{1}{\gamma_2 d_2} \right) \cdot \left(\gamma_1 d_1 + \gamma_2 d_2 \right) =$$
$$= 9.2 \times 10^{-24} \frac{T}{\mu_g} \cdot \left(\frac{1}{\gamma_1 d_1} + \frac{1}{\gamma_2 d_2} \right) \cdot \left(\gamma_1 d_1 + \gamma_2 d_2 \right)$$

k_B Boltzmann constant (= 1.38×10^{-23}), (J/K)
T temperature, (K)
μ_g gas viscosity, (kg/m/s)

- Slip Flow Model

The Brownian coagulation is given by the following formula ([74], section 4.2.3):

$$K_{Brown} = \frac{2k_B T}{3\mu_g} \cdot \left(\frac{C_{m1}}{\gamma_1 d_1} + \frac{C_{m2}}{\gamma_2 d_2} \right) \cdot \left(\gamma_1 d_1 + \gamma_2 d_2 \right) =$$
$$= 9.2 \times 10^{-24} \frac{T}{\mu_g} \cdot \left(\frac{C_{m1}}{\gamma_1 d_1} + \frac{C_{m2}}{\gamma_2 d_2} \right) \cdot \left(\gamma_1 d_1 + \gamma_2 d_2 \right)$$

C_{m1} Cunningham correction factor, particle 1, (-)
C_{m2} Cunningham correction factor, particle 2, (-)

The Cunningham correction factor is calculated from ([75], section 2.6, equation 2.57):

$$C_m = 1 + Kn \cdot \left(A_1 + A_2 \cdot \exp\left[-\frac{A_3}{Kn} \right] \right)$$

Where Kn is the Knudsen number, equal to twice the mean free path divided by the particle diameter ($Kn = 2\, l / d$), and A_i are user-defined constants, with the default values (see Volume 2):

$A_1 = 1.257$
$A_2 = 0.4$
$A_3 = 1.1$

The mean free path, needed for the Knudsen number calculation, is given by ([21], section 9.7.3):

$$l = \frac{\mu_g}{\rho_g} \cdot \sqrt{\frac{\pi M}{2 R_m T}} = 1.25 \cdot \frac{\mu_g}{\rho_g} \cdot \frac{1}{\sqrt{RT}}$$

μ_g gas viscosity, (kg/m/s)
ρ_g gas density, (kg/m^3)
R_m universal gas constant (=8314.51), (J/kmol/K)
R individual gas constant, (J/kg/K)
M gas molar weight, (kg/kmol)
T temperature, (K)

- Fuchs Model

The Brownian coagulation is given by the following formula ([74], section 4.2.4):

$$K_{Brown} = \frac{2\pi \cdot (\gamma_1 d_1 + \gamma_2 d_2) \cdot (D_{C,1} + D_{C,2})}{\frac{d_1 + d_2}{d_1 + d_2 + 2g_{12}} + \frac{8 \cdot (D_{C,1} + D_{C,2})}{(d_1 + d_2) \cdot v_{12}}}$$

where $D_{C,i}$ is the diffusion coefficient, given by ([74], section 3.1):

$$D_{C,i} = \frac{k_B T}{3\pi \mu_g d_i}$$

The formula presented in [74] is valid for spherical particles. Compared to this formula, the agglomeration shape factor, χ_i, and Cunningham correction factor, $C_{m,i}$, were introduced, following [46]. The final expression used in SPECTRA is:

$$D_{C,i} = 1.464 \times 10^{-24} \frac{T}{\mu_g \, d_i \chi_i} C_{m,i}$$

The particle relative velocity, v_{12}, is given by:

$$v_{12} = \sqrt{v_1^2 + v_2^2}$$

The individual velocities of particles 1 and 2 are equal to:

$$v_i = \sqrt{\frac{8 k_B T}{\pi m_i}} = 8.192 \times 10^{-12} \sqrt{\frac{T}{\rho_i d_i^3}}$$

k_B	Boltzmann constant (= 1.38×10^{-23}), (J/K)
T	temperature, (K)
m_i	mass of particle i, (kg) (= $\rho_i \, \pi/6 \, d_i^3$)
ρ_i	density of particle i, (kg/m^3)
d_i	diameter of particle i, (m)

The symbol g_{12}, represents the characteristic distance for coagulation of particles 1 and 2, and is given by:

$$g_{12} = \sqrt{g_1^2 + g_2^2}$$

The individual characteristic distances for particles 1 and 2 are determined by:

$$g_i = \frac{1}{3 d_i L_i} \left[(d_i + L_i)^3 + (d_i^2 + L_i^2)^{3/2} \right] - d_i$$

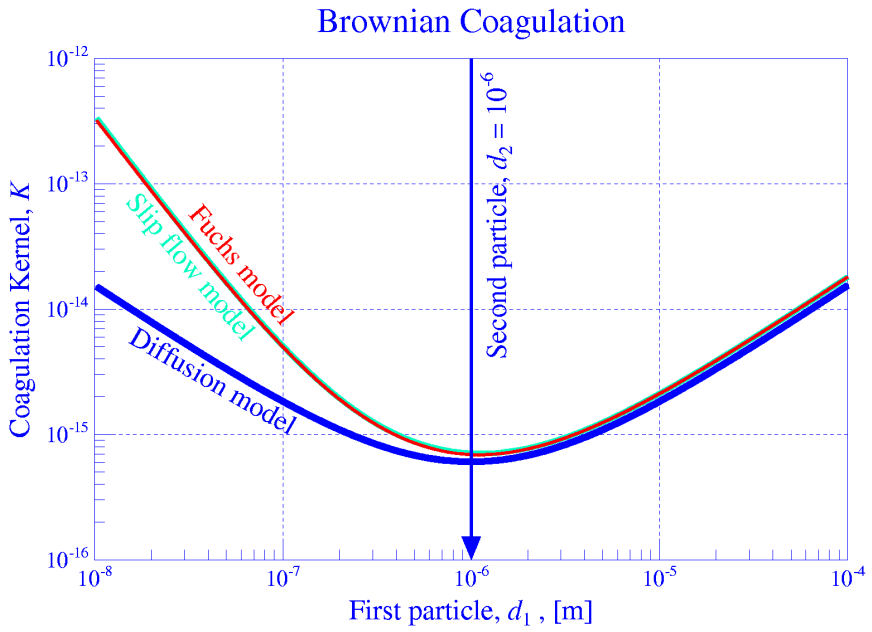


Figure 12-2 Brownian coagulation – comparison of models, $d_2 = 10^{-6}$ m

Finally, L is the characteristic travel distance, given by:

$$L_i = \frac{8D_{C,i}}{\pi v_i} = 2.546 \cdot \frac{D_{C,i}}{v_i}$$

Summarizing, there are three optional models for Brownian coagulation kernel calculation. These are the diffusion model, the slip flow model, and the Fuchs model. The Fuchs model is expected to be the most accurate one, and is therefore selected by default. The diffusion model is not valid for very small particles. This is clearly seen in Figure 12-2, comparing the three models. For particles smaller than 10^{-6} m the diffusion model predicts too low coagulation kernel.

The coagulation kernel, as calculated from one of the three available models, is multiplied by the sticking coefficient, c_s, to obtain the Brownian coagulation kernel. This is done to be consistent with the two other coagulation models. Therefore the user can deactivate the whole coagulation model by simply setting c_s to zero. Note that MELCOR uses the sticking coefficient differently, as a multiplier for the term $(d_1+d_2)\, v_{12}$ (see [46], RN Package Reference Manual, Appendix B). With this approach the user cannot set c_s to zero.

12.2.3.3 Turbulent Coagulation

The turbulent coagulation is calculated from ([46], RN-RM, Appendix B):

$$K_{turb} = c_s \sqrt{K_1^2 + K_2^2}$$

The term K_1 represents turbulent diffusion coagulation, and is given by ([74], sec. 4.6, eq. 4.114):

$$K_1 = \left(\frac{8\pi\rho_g \varepsilon_T}{15\mu_g} \right)^{1/2} \cdot (r_1 + r_2)^3$$

The coagulation shape factors are added, following [46], and the radius r is replaced by the diameter d:

$$K_1 = \left(\frac{\pi\rho_g \varepsilon_T}{120\mu_g} \right)^{1/2} \cdot (\gamma_1 d_1 + \gamma_2 d_2)^3 =$$

$$= 0.1618 \cdot \left(\frac{\rho_g \varepsilon_T}{\mu_g} \right)^{1/2} \cdot (\gamma_1 d_1 + \gamma_2 d_2)^3$$

Here ε_T is the turbulence dissipation rate, (m^2/s^3), other symbols were explained above. The turbulence dissipation density is, similarly as in MELCOR, a user-defined parameter, with the default value of 0.001. However, in contrast to MELCOR, in SPECTRA ε_T is not a constant number, but a function of Reynolds number. The *Re* dependence is shown in Figure 12-3.

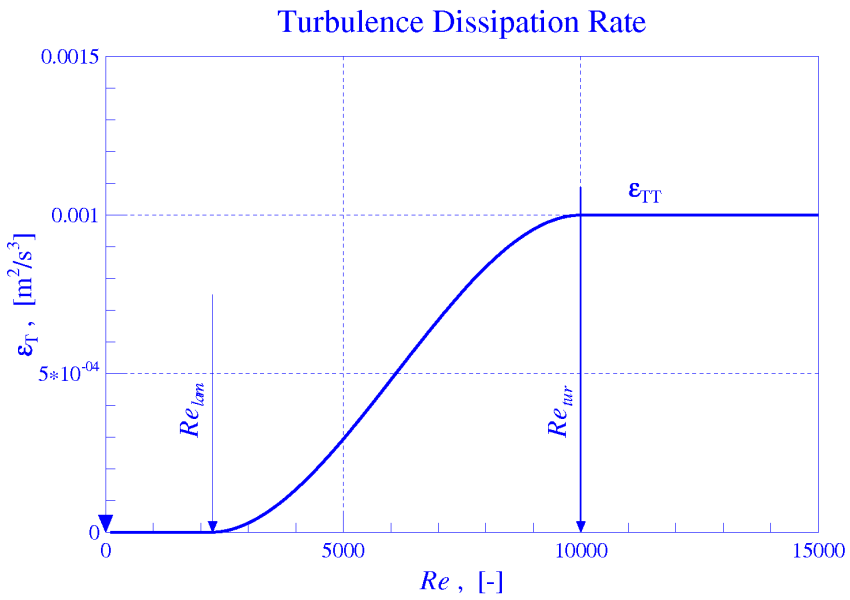


Figure 12-3 Turbulence dissipation rate

The limiting Reynolds numbers, Re_{lam}, Re_{tur}, are user-defined, with default values of 2200 and 10000 respectively. Thus the turbulent coagulation does not occur in stagnant atmosphere and in laminar flow.

The term K_2 represents inertial coagulation, and is given by ([74], sec. 4.11, eq. 4.253, 4.270):

$$K_2 = C \cdot \left(\frac{\rho_g^{1/4} \varepsilon_T^{3/4}}{\mu_g^{1/4}} \right) \cdot (r_1 + r_2)^2 \cdot |\tau_1 - \tau_2|$$

where C is equal to 6.2 (equation 4.253), or 5.51 (equation 4.270), and τ is given by [74]:

$$\tau = \frac{2 \cdot r^2 \cdot \rho_p}{9 \mu_g}$$

ρ_p	density of aerosol particles, (kg/m^3)
ρ_g	density of gas, (kg/m^3)
μ_g	viscosity of gas, (kg/m/s)
r	particle radius, (m)

The expression for τ is substituted into the formula for K_2. The radius r is replaced by the diameter d, and the coagulation shape factors are added, following [46]. The Cunningham factors and the dynamic shape factors are added in τ, following [46]. The resulting formula is:

$$K_2 = \frac{2C\rho_p}{9 \cdot 16} \cdot \left(\frac{\rho_g^{1/4} \varepsilon_T^{3/4}}{\mu_g^{5/4}} \right) \cdot (\gamma_1 d_1 + \gamma_2 d_2)^2 \cdot \left| \frac{C_{m,1} d_1^2}{\chi_1} - \frac{C_{m,2} d_2^2}{\chi_2} \right| =$$

$$= 0.086 \cdot \rho_p \cdot \left(\frac{\rho_g^{1/4} \varepsilon_T^{3/4}}{\mu_g^{5/4}} \right) \cdot (\gamma_1 d_1 + \gamma_2 d_2)^2 \cdot \left| \frac{C_{m,1} d_1^2}{\chi_1} - \frac{C_{m,2} d_2^2}{\chi_2} \right|$$

Where the constant 0.086 was calculated using C=6.2. Note that in MELCOR the constant is 0.04029 instead of 0.086 (compare [46], RN Package Reference Manual, Appendix B). Thus in MELCOR the multiplier on K_2 is about twice smaller than in SPECTRA. In spite of that all tests show that in MELCOR the coagulation proceeds somewhat faster than in SPECTRA – see Volume 3. The reason for this fact is unclear.

12.2.4 Aerosol Transport (Inter-Volume Flows of Aerosols)

In order to calculate the particle transport through Control Volumes an implicit formulation is needed. The aerosol package often needs to be executed with very large time steps: see Volume 3, Analysis of Dust Transport and Deposition in PBMR. The runs shown there were performed using stationary flow solution. The flow solution was frozen; the Courant limit was deactivated, and the time step was set at 1000 s. The CV's in the PBMR model are of the size ~1 m^2, (~5 kg of gas), while the stationary flow is ~200 kg/s. Therefore the mass of dust transported through a CV during a time step is extremely large compared to the mass of dust present within a CV. In stationary conditions the ratio of those two numbers (the same as the ratio of gas masses) is:

$$\frac{200\ kg/s \times 1000\ s}{5\ kg} = \frac{200{,}000}{5} = 40{,}000$$

The large time step (1000 s) was needed in order to run the simulation for a long time (1 year), since deposition of dust is very slow. In "normal" calculations the Courant limit restricts the time step to such a value that the quotient is ~1 (in the present example it would be ~5/200 = ~10^{-2} s). For the frozen flow solution the Courant limit does not exist of course, but still the development of the dust concentrations must be accurately calculated.

Therefore the following requirement was set for the aerosol flow solution – the aerosol flow solution must be able to find an accurate solution, independent and insensitive to the applied time step, for the time steps that exceed the Courant limit by several orders of magnitude.

The derivation of the aerosol balance equation, shown below, concerns a single size section. The same equation is solved for every size section. In order to limit the amount of subscripts in the discussion that follows, the subscript indicating the size section has not been used. It should be clear to the reader that all parameters introduced below are specific to a single size section.

Aerosol particles are transported between Control Volumes with fluid flows, including both atmosphere gas flow and pool flows. The aerosols are assumed to have velocities that are related to the fluid velocities by using slip factors.

Particle slip in atmosphere gas flow

The atmosphere gas slip factor is defined as the ratio of the aerosol particle velocity to that of the carrier gas. The particle transport through a junction is calculated using the particle velocity, $v_{p,l}$, defined as:

$$v_{p,l} = S_A \cdot v_g - J_{JN} \cdot v_{D,grav}$$

S_A	slip factor for particle flow with the gas atmosphere, (-)
v_g	gas velocity, (m/s)
$v_{D,grav}$	deposition velocity due to gravitational settling, (m/s)
J_{JN}	junction direction indicator, equal to 0 for horizontal junctions, +1 for vertical up, and –1 for vertical-down junctions (see Volume 2, record 200XXX, IVERJN):

The slip factors can be defined by the user, (separately for each size section), or calculated using the diffusiophoretic theory (see [46], RN RM section 2.4.2.2), as:

$$S_A = \sqrt{\frac{M_1}{M_2}} = \sqrt{\frac{R_2}{R_1}}$$

M_1 molecular weight of gas in the upstream CV, (kg/kmole)
M_2 molecular weight of gas in the downstream CV
R_1 gas constant in the upstream CV, (J/kg/K)
R_2 gas constant in the downstream CV

Including the gravitational deposition velocity allows the aerosol to settle through vertical junctions when there is no gas flow.

Particle slip and v_∞ in pool flow

The slip factor is defined as the ratio of the aerosol particle velocity to that of the carrier liquid. The particle transport through a junction is calculated using the particle velocity, $v_{p,l}$, defined as:

$$v_{p,l} = S_P \cdot v_l + J_{JN} \cdot v_\infty$$

S_P slip factor for the particle flow with pool, (-)
v_l liquid velocity, (m/s)
v_∞ vertical velocity (m/s) of a single particle in stagnant pool

Including the v_∞ velocity allows the aerosol to flow through vertical junctions when there is no liquid flow. A positive value of v_∞ means that the flow will be upwards (particle density smaller than the density of the liquid). A negative value of v_∞ means that the flow will be downwards (particle density larger than the density of the liquid). The user may define a constant value of v_∞ or use the following formula:

$$v_\infty = \left(\frac{4/3}{C_D} \cdot \frac{D_p g(\rho_f - \rho_p)}{\rho_f} \right)^{0.5}$$

g gravity constant, = 9.81 (m/s^2)
ρ_p particle density, (kg/m^3)
ρ_f fluid density, (kg/m^3)
D particle diameter, (m)
C_D drag coefficient, (-)

The formula comes from balancing the drag force, F_d and the buoyancy force, F_b:

$$F_d = C_D \cdot \frac{\rho_f v^2}{2} \cdot A = C_D \cdot \frac{\rho_f v^2}{2} \cdot \frac{\pi D^2}{4}$$

$$F_b = g \cdot (\rho_p - \rho_f) \cdot V = g \cdot (\rho_p - \rho_f) \cdot \frac{\pi D^3}{6}$$

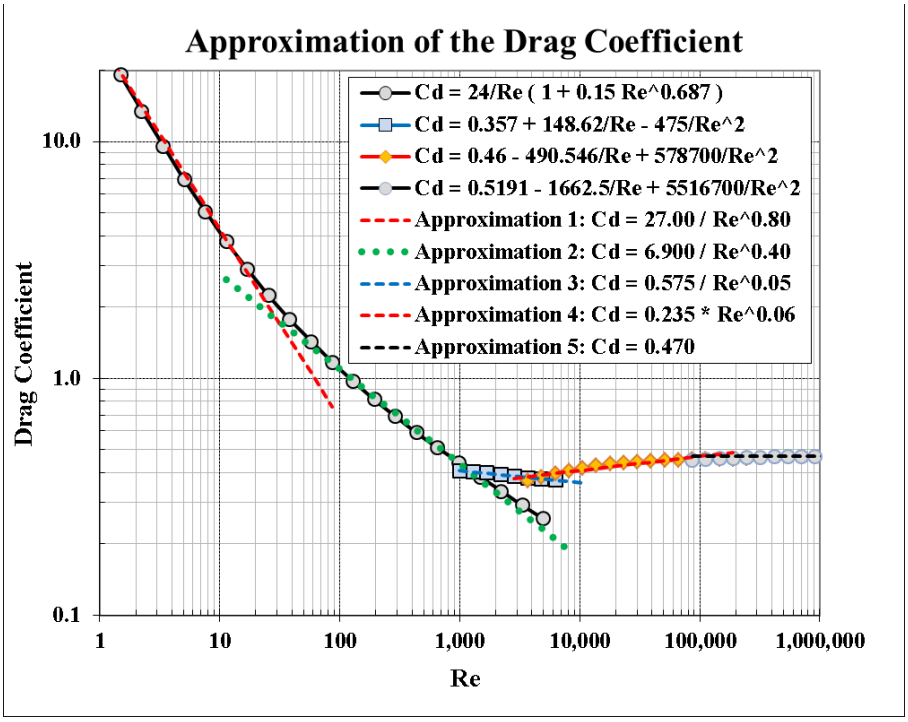


Figure 12-4 Drag coefficient for particles (also for droplets and bubbles - sec. 2.6.5)

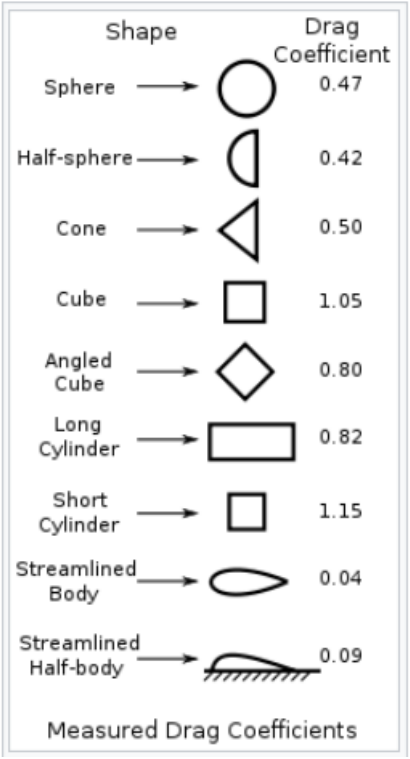


Figure 12-5 Measured drag coefficients $C_{D,\infty}$ [205].

The measured values of drag coefficient are shown in Figure 12-5. Note that the values are appropriate for $Re \rightarrow \infty$. The dependence of drag coefficient on Reynolds number in the whole range of Re is described in section 2.6.5. The values are shown in Figure 12-4. The user can define the value for $Re \rightarrow \infty$. by setting VINPRT > 100, VINFRT > 100. This is done for every size section. The drag coefficients are equal to:

- $C_{D,Re\rightarrow\infty}$= VINPRT – 100 (CV pool)
- $C_{D,Re\rightarrow\infty}$= VINFRT – 100 (pool flow through JN)

Alternatively the user may define a constant value of C_D (independent of Re) by setting VINPRT > 1000, VINFRT > 1000. The drag coefficients are equal to C_D = VINPRT – 1000, C_D = VINFRT – 1000.

- C_D = const. = VINPRT – 1000 (CV pool)
- C_D = const. = VINFRT – 1000 (pool flow through JN)

For spherical particles C_D = 0.47. In case of solid particles, created during coagulation, the shape will be very different from a sphere and this typically results in a larger C_D. In case of gas particles, the shape will be more like the streamlined body, resulting in a smaller C_D.

On top of the drag coefficient correlation, a correlation specific for very small particles may be used, as follows.

$$v_{\infty} = \frac{1}{18} \cdot \frac{D_p{}^2 g(\rho_f - \rho_p)}{\eta_f} \cdot C_{small}$$

Here C_{small} is the user-defined constant (CSMLCV). This correlation is applicable for very small particles, $D_p < \sim 10^{-4}$ m. The best estimate value of C_{small} is 1.0.

Particle flow through a junction

The number of particles transported through a junction per second is equal to:

$$v_{p,l} \cdot A_l \cdot n_i$$

$v_{p,l}$	velocity of particles in the junction l (in atmosphere or pool), (m/s)
A_l	fluid flow area (gas or liquid) for the junction l, (m^2)
n_i	particle concentration in the source volume i (in atmosphere or pool), ($1/m^3$).

The mass of particles transported through a junction per second is equal to:

$$v_{p,l} \cdot A_l \cdot \frac{m_i}{V_i}$$

$v_{p,l}$	velocity of particles in the junction l (in atmosphere or pool), (m/s)
A_l	fluid flow area (gas or liquid) for the junction l, (m^2)
V_i	fluid volume (gas or liquid) of the source volume i, (m^3)
m_i	mass of particles in the source control volume i (in atmosphere or pool), (kg).

Loss of particles due to flow out of a CV

The mass removal rate for a Control Volume is written for those junctions for which the flow is out of the control volume *i*. For this purpose an outgoing flow indicator is used:

O_{li} outgoing flow indicator
= 1: particle flow through the junction *l* is out of the control volume *i*.
= 0: otherwise.

Using this indicator, the mass removal rate for the volume *i* due to flow through junction *l* is written as:

$$-v_{p,l} \cdot A_l \cdot \frac{m_i}{V_i} \cdot O_{li}$$

The rate of change of particles due to flow out of the volume *i* is obtained by summing the junction flows for all junctions connected to the volume *i*:

$$\text{Flow of particles out of Control Volume } i = -\sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_i} O_{li} \right) \cdot m_i$$

Gain of particles due to flow into a CV

The mass source rate for a Control Volume is written for those junctions for which the flow is into the Control Volume *i*. For this purpose an incoming flow indicator is used:

I_{li} incoming flow indicator
= 1: particle flow through the junction *l* is into the volume *i* (from the volume *j*).
= 0: otherwise.

The source rate is given by:

$$+v_{p,l} \cdot A_l \cdot \frac{m_i}{V_j} \cdot I_{li} \cdot (1-\varepsilon_l)$$

Where the term $(1 - \varepsilon_l)$ takes into account eventual aerosol removal in the junction *l*. This removal may occur in three cases:

- A filter exists in the junction *l*; in this case ε_l is the filter efficiency (= ε_F, section 12.2.10)
- The stream of gas enters the pool in the receiving volume; in this case ε_l is the overall pool scrubbing efficiency (= ε_{PS}, section 12.2.9). This mass is removed from the atmosphere and added into the pool.
- An inertial impaction model is associated with this junction. In this case ε_l is the collection efficiency (= η, section 12.2.5.6). This mass is removed from the atmosphere and added into the pool.

The total source of particles for all flows into the control volume *i* is obtained by summing junction flows for all junctions connected to the volume *i*:

$$\text{Flow of particles into Control Volume } i = \sum_{l \in j} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j$$

Using the above terms, the total mass balance for a given CV takes into account aerosol sources and sinks:

$$\frac{dm_i}{dt} = \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j - \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_i} O_{li} \right) \cdot m_i + S_E + S_R - D + R + K$$

m_i	mass of particles in volume *i*, (kg)
S_E	external source (user-defined tabular or control functions), (kg/s)
S_R	source due to fission product release, (kg/s)
D	removal due to all deposition mechanisms, (kg/s)
R	source due to resuspension of deposited aerosols, (kg/s)
K	net source due to coagulation of all size sections, (kg/s)

Note that the above equation is similar to the equation for the aerosol dynamics within a single Control Volume, shown in section 12.2.2. Here the sum given by the first two terms on the right hand side of the equation determines the term *F*, for the equation in section 12.2.2. Conversely, the term *C*, representing here the source due to coagulation, is calculated as the sum of all coagulation sources, given by the equation in section 12.2.2.

The mass balance is written here using the particle masses, rather than the particle densities, as in section 12.2.2. SPECTRA is using and printing both the concentrations ($1/m^3$) and the total masses (kg) within a CV (in MELCOR for example only masses are available, which makes it sometimes difficult to analyze the results – see the Vent test case in Volume 3). The conversion from particle concentrations to particle masses is:

$$n = \frac{m}{V_{CV} \rho_p V_p}$$

V_{CV}	fluid volume within a CV (atmosphere or pool), (m^3),
ρ_p	particle density, (kg/m^3),
V_p	volume of a single particle (see Table 12-1), (m^3),

In order to solve the equation implicitly one needs to take into account that the deposition term is proportional to the particle concentrations, and consequently the particle masses. For a single deposition surface *k*, in the control volume *i*, the deposition rate (in kg/s) is given by:

$$A_{D,k} \cdot v_{D,k} \cdot \frac{m_i}{V_i} = \left(\frac{A_{D,k} \cdot v_{D,k}}{V_i} \right) \cdot m_i$$

$A_{D,k}$	deposition area, (m^2)
$v_{D,k}$	deposition velocity, calculated as shown in section 12.2.5, (m/s)

The total removal rate from volume *i* due to deposition is obtained by summing all the depositions in the volume *i*, including depositions on 1-D and 2-D Solid Heat Conductors, as well as depositions on the pool surface (if present).

Removal of particles due to deposition in Control Volume *i* = $$D = \sum_{k \in i} \left(\frac{A_{D,k} \cdot v_{D,k}}{V_i} \right) \cdot m_i$$

Since the deposition must be calculated using the end-of-time-step value of particle mass, m_i, the deposition term must be included implicitly in the balance equation. The mass balance equation becomes:

$$\frac{dm_i}{dt} = \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j - \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_i} O_{li} \right) \cdot m_i - \sum_{k \in i} \left(\frac{A_{D,k} v_{D,k}}{V_i} \right) \cdot m_i + S_E + S_R + R + K$$

The above equation is written in a finite difference form, by replacing the derivative by the difference:

$$\frac{dm_i}{dt} = \frac{m_i - m_i^0}{\Delta t}$$

m_i particle mass in volume *i*, new value (kg)
m_i^0 particle mass in volume *i*, old time step value (kg)
Δt time step size, (s)

The particle balance equation becomes:

$$\frac{m_i - m_i^0}{\Delta t} = \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) (1 - \varepsilon_l) m_j - \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_i} O_{li} \right) m_i - \sum_{k \in i} \left(\frac{A_{D,k} v_{D,k}}{V_i} \right) m_i + S_E + S_R + R + K$$

The above set of equations can be written shortly in a matrix form:

$$A \cdot M = B$$

where ***M*** is a vector of unknown particle masses, m_i, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. Elements of the matrix ***A*** are equal to:

- diagonal element ($i=j$):

$$a_{ii} = 1 + \Delta t \cdot \sum_{k \in i} \left(\frac{A_{D,k} \cdot v_{D,k}}{V_i} \right) + \Delta t \cdot \sum_{l \in i} \left(\frac{A_l \cdot v_{p,l}}{V_i} \cdot O_{li} \right)$$

- other elements ($i \neq j$):

$$a_{ij} = -\Delta t \cdot \sum_{l \in j} \left(\frac{A_l \cdot v_{p,l}}{V_i} \cdot I_{lj} \right) \cdot (1 - \varepsilon_l)$$

$A_{D,k}$ deposition area, deposition surface *k*, within Control Volume *i*, (m^2),
$v_{D,k}$ deposition velocity calculated for the deposition surface *k*, (m/s),
V_i fluid volume of the Control Volume *i*, (m^3),

A_l flow area, junction l, connected to Control Volume i and j, (m^2),
$v_{p,l}$ velocity of particle flowing with fluid, junction l, (m/s),
ε_l efficiency of filter (= ε_F, section 12.2.10), or overall pool scrubbing efficiency (= ε_{PS}, section 12.2.9) if present in the junction l, or inertial impaction collection efficiency (= η, section 12.2.5.6) if present in the junction l.

The first sum in the diagonal element is over all deposition surfaces k in the Control Volume i, the second sum is over all junctions l connected to or from the Control Volume i.

The elements of vector $\boldsymbol{B}$ are equal to:

$$b_i = m_i^0 + \Delta t \cdot \left(S_E + S_R + R + K\right)$$

The matrix equation is solved using one of the standard matrix solvers (see section 17.4). As a result the particle masses, m_i, in all Control Volumes are calculated.

The equation of inter-volume transport contains the coagulation term K, the value of which is calculated by the aerosol dynamics equation within a CV (section 12.2.2), and which is not known at the beginning of the time step. Similarly, the equation calculating coagulation, given in section 12.2.2 contains the term F, the value of which is determined by the above solution and which in turn is not known for the coagulation equation. The solution procedure is as follows:

- The value of K is guessed (the value from the old time step is used).
- The inter-volume transport equation is solved, resulting in the end-of time step values of all particle masses. These masses are called the "projected masses", since they represent the expected (projected) end-of time step masses, computed based on the estimated value of the coagulation term.
- The aerosol dynamics within a single CV (coagulation) is solved using the term F based on the projected masses. Those masses are called the "true masses" of particles, since they represent the exact (conservative) masses of particles within each Control Volume, for the inter-volume transport of particles as calculated by the projected masses.
- The value of K is updated for the next iteration. At least 3 iterations are made. The projected and true masses are typically very similar already after first iteration.

Because of the applied solution strategy the aerosol masses are conserved (within the accuracy of the double precision arithmetic – relative error ~10^{-15}). The projected masses, m_i, coming from the inter-volume transport equation are not conservative; their error (discrepancy between the conservative values, obtained from the single volume dynamics equation) is typically within 1%. Note that because of including the deposition terms in the implicit formulation, deposition rates are calculated based on the projected masses and not the true masses. This is considered unimportant because the accuracy of the correlations used to compute the deposition velocities (see section 12.2.5) is at best an order of magnitude worse (~10%) than the accuracy of the projected masses (~1%).

The stability of the aerosol flow solution, including the airborne aerosol transport with the atmosphere gas, and the transport of the aerosols deposited in the pool, was verified by performing time step sensitivity studies for the test cases shown in Volume 3. In all tests shown in Volume 3 the results (aerosol concentrations) were practically insensitive to the applied time step. In one case (Aerosol Loop Flow and Deposition case) sensitivity to the time step was found for the MELCOR code (solution results with Δt = 5 s were different from the solution with Δt = 0.5 – see Volume 3)

but not for SPECTRA. The PBMR cases were run with Δt = 1000 s, 100 s, and 10 s, and no significant differences were observed.

12.2.5 Aerosol Deposition

The deposition mechanisms in SPECTRA are:

- Gravitational settling, $v_{D,\,grav}$.
- Brownian deposition, $v_{D,\,Brown}$.
- Thermophoresis, $v_{D,\,ther}$.
- Diffusiophoresis, $v_{D,\,diff}$.
- Turbulent deposition, $v_{D,\,turb}$.
- Inertial impaction, $v_{D,\,inert}$.

For each of the mechanisms mentioned above a deposition velocity, $v_{D,\,i}$, is calculated using an appropriate correlation (described below). The total deposition velocity is calculated, as in most aerosol codes, as the sum of individual velocities:

$$v_D = v_{D,grav} + v_{D,Brown} + v_{D,ther} + v_{D,diff} + v_{D,turb} + v_{D,inert}$$

Note that compared to MELCOR, SPECTRA has two more deposition mechanism. The turbulent deposition and inertial impaction models are not present in MELCOR (according to [46] the inertial impaction is one of the most important deposition mechanisms missing in MELCOR).

The Brownian, turbulent, and inertial deposition mechanisms have always positive contribution to the overall deposition:

$$v_{D,Brown} > 0.0$$
$$v_{D,turb} > 0.0$$
$$v_{D,inert} > 0.0$$

The gravitational deposition, thermophoresis, and diffusiophoresis, can have both negative and positive contribution or no (zero) contribution:

$$v_{D,grav} < 0.0 \quad or \quad v_{D,grav} \geq 0.0$$
$$v_{D,ther} < 0.0 \quad or \quad v_{D,ther} \geq 0.0$$
$$v_{D,diff} < 0.0 \quad or \quad v_{D,diff} \geq 0.0$$

Gravitational deposition has negative contribution in case of horizontal, down-facing surfaces; thermophoresis has negative contribution if wall temperature is higher than the gas temperature; diffusiophoresis in case of evaporation (from pool surface).

Note that negative deposition velocity is something very different from resuspension! It only means that due to given mechanism the airborne particles have velocities away from the surface. The individual deposition velocities are coming into the overall velocity with plus or minus sign. Only if the overall deposition velocity is negative, it is then truncated to zero:

$$v_D \geq 0.0$$

12.2.5.1 Gravitational Deposition

The gravitational deposition velocity is given by ([74], section 7.2):

$$v_{D,grav} = \frac{1}{18}\frac{\rho_p d_p^2 g\, C_m}{\mu_g \chi} = 0.5448\,\frac{\rho_p d_p^2 C_m}{\mu_g \chi}$$

ρ_p	density of particle, (kg/m^3)
d_p	diameter of particle, (m)
g	gravity constant, (m/s^2) (= 9.80665, [32])
C_m	Cunningham correction factor (see section 12.2.3), (-)
μ_g	viscosity of gas, (kg/m/s)
χ	dynamic shape factor, (-)

Aerosols can deposit on the surfaces of 1-D and 2-D Solid Heat Conductors, as well as on the pool surface. The deposition depends on the vertical orientation of a surface. Each heat conductor surface has default gravity factors (δ_{grav}), which can be re-defined by the user through the input data (see Volume 2). The default settings are:

- Horizontal, up-facing surface: δ_{grav} = +1.0
- Vertical surface: δ_{grav} = 0.0
- Horizontal, down-facing surface: δ_{grav} = –1.0

In case of cylindrical and spherical geometries, as well as surfaces with fins or spines, the default values of the gravity factors are between 0.0 and 1.0 (see Volume 2 for detailed description of the default settings for all types of surfaces).

The effective gravitational deposition velocity is for each surface given by:

$$v_{D,grav} = 0.5448\,\frac{\rho_p d_p^2 C_m}{\mu_g \chi}\cdot\delta_{grav}$$

For calculation of the gravitational settling on a pool surface, the gravity factor is always equal to δ_{grav} = +1.0.

12.2.5.2 Brownian Deposition

The Brownian deposition velocity is given by ([80], section 5):

$$v_{D,Brown} = \frac{D_C}{\delta_{BL}}$$

D_C	diffusion coefficient, (m^2/s)
δ_{BL}	diffusion boundary layer thickness, (m)

The diffusion coefficient is given by ([74], sec. 3.1):

$$D_C = \frac{k_B T}{6\pi\mu_g r_p} = \frac{k_B T}{3\pi\mu_g d_p}$$

k_B	Boltzmann constant (= 1.38×10^{-23}), (J/K)
T	temperature, (K)
μ_g	viscosity of gas, (kg/m/s)
r_p	radius of particle, (m)
d_p	diameter of particle, (m)

The dynamic shape factor, χ, is introduced following [46]. The deposition velocity is equal to:

$$v_{D,Brown} = \frac{k_B T}{3\pi\mu_g d_p \chi \delta_{BL}} C_m = 1.464\times10^{-24}\frac{T}{\mu_g d_p \chi \delta_{BL}} C_m$$

Two options are available to determine the boundary layer thickness:

- User-defined constant value. The default value is (Volume 2):

$$\delta_{BL} = 1.0\times10^{-5}\ m$$

- Theoretical model, based on the heat and mass transfer analogy.
  It is assumed that the deposition velocity equals the mass transfer coefficient, $v_{D,Brown}=K_m$, (m/s), calculated for the particle Schmidt number:

$$K_m = \frac{Sh\cdot D_C}{D_{hyd}}$$

  The deposition velocity is equal to $v_{D,Brown}=D_C / \delta_{BL}$ (see for example [80], sec. 5, [74], page 8) and therefore:

$$\delta_{BL} = \frac{D_{hyd}}{Sh}$$

  The Sherwood number is calculated using the heat and mass transfer analogy ($Nu \rightarrow Sh$, $Pr \rightarrow Sc$), from the laminar and the turbulent flow correlations:

$$Sh = \begin{cases} C_{lam} & (laminar) \\ C_{tur} Re^a Sc^b & (turbulent) \end{cases}$$

  where *Re*, *Sc*, *Sh* are Reynolds number, particle Schmidt number, and Sherwood number respectively. The constants, C_{lam}, C_{tur}, a, b, are user-defined. The default values are (see Volume 2):

  C_{lam} = 3.656 a = 0.8
  C_{tur} = 0.023 b = 0.4

  The Reynolds number is taken from the Junction data (with a limit in the RT package the same as in the JN package: Re_{min}=100.0), while the *Sc* number is calculated from:

$$Sc = \frac{3\pi\ \mu_g^2\ d_p\ \chi}{k_B\ \rho_g\ T\ C_m}$$

12.2.5.3 Thermophoresis

The thermophoretic deposition results from the force exerted on aerosol particles by temperature gradient in the bulk gas. Difference in the momentum transfer by molecular impacts on opposite sides of the particle will tend to drive the particle into the direction of colder gas. Therefore, if the wall temperature is lower than the gas temperature then the thermophoretic deposition velocity will be towards the wall (positive), while it will be against the wall (negative) in case of hot surface and cold gas.

Three models are available in SPECTRA for calculation of thermophoresis:

- Brock correlation [82]
- He-Ahmadi correlation [83]
- Combination of Brock and He-Ahmadi, with the first used for large particles and the second used for small particles

These models are described below.

- Brock correlation

The Brock correlation is:

$$v_{D,therm} = \frac{2\mu_g C_m C_s B}{\chi \rho_g T (1+3F_s Kn)(1+2B)} \nabla T$$

C_m	Cunningham correction factor, (-)
C_s	thermal exchange coefficient, (-)
F_s	slip factor, (-)
T	temperature, (K)
μ_g	viscosity of gas, (kg/m/s)

The value of B is given by:

$$B = C_t Kn + \frac{k_g}{k_p}$$

C_t	thermal accommodation coefficient, (-)
Kn	Knudsen number, (-)
k_g	thermal conductivity of gas, (W/m/K)
k_p	thermal conductivity of particle, (W/m/K)

The dynamic shape factor, χ, has been added to the Brock correlation, based on [46]. Note that the formula applied in MELCOR is somewhat different than the one shown above. The formula applied in SPECTRA contains a multiplier of $2C_s$. In MELCOR, the constant 3/2 replaces $2C_s$ [46]. Based on work performed at NRG, the value of C_s recommended is 1.17. Therefore the Brock correlation, as implemented in MELCOR, gives the thermophoretic deposition velocities of about 64% (1.5/2.34) of the values obtained in SPECTRA.

For SPECTRA it was decided to use $2C_s$, based on comparison of Brock and He-Ahmadi (see below) correlations. With this coefficient the Brock correlation agrees very well with the He-Ahmadi correlation for the practically interesting particle range of $10^{-6} \div 10^{-5}$ m (Figure 12-6). For larger particles Brock already gives lower velocity than He-Ahmadi, so applying the factor 3/2 from MELCOR would enlarge further the discrepancy.

The values of coefficients in the Brock correlation are user-defined (see Volume 2). The default values are given below.

C_s = 1.17 (note that the Brock model may be deactivated by C_s = 0.0)
C_t = 2.50
F_s = 1.257 (equal to A_1 in the Cunningham correction factor)

The coefficients used in thermophoresis calculations by different aerosol codes were reviewed in [80]. The value of C_t was found to be between 2.0 (AEROSOLS, HAA4 codes) and 2.50 (HAARM, RETAIN codes). MELCOR uses 2.25 [46]. The value of 2.50 provides a very good agreement with the He-Ahmadi correlation for the practically interesting particle range of $10^{-6} \div 10^{-5}$ m (Figure 12-6). The value of F_s was found to be between 1.0 (AEROSIM, RETAIN, NAUA codes) and 1.37 (AEROSOLS, HAARM, HAA4 codes). MELCOR uses 1.257 [46].

Summarizing, the default coefficients in thermophoretic correlation are somewhat different in SPECTRA than in MELCOR. MELCOR correlation can be simulated in SPECTRA by setting C_t = 2.25, and C_s = (3/2)/2 = 0.75. Comparison of Brock correlation obtained using the MELCOR coefficients, with He-Ahmadi correlation is shown in Figure 12-7.

- He-Ahmadi correlation

The He-Ahmadi correlation is:

$$v_{D,therm} = \frac{1.15 \cdot Kn}{4\sqrt{2}\alpha(1+\pi_1 Kn/2)} \cdot \left[1-\exp\left(-\alpha/Kn\right)\right] \cdot \left(\frac{4}{3\pi}\varphi\pi_1 Kn\right)^{1/2} \frac{C_m k_B}{3\pi\mu_g D_m^2} d_p \cdot \nabla T$$

where π_1, α, φ, D_m, are given by rather complex formulae, shown below. Although the total formula is quite complex, it may be considerably simplified by substituting all constant numbers.

First, π_1 is equal to:

$$\pi_1 = 0.18\frac{36/\pi}{(2-S_n+S_t)\cdot 4/\pi + S_n} = 0.582$$

(since S_n and S_t are both equal to 1.0).

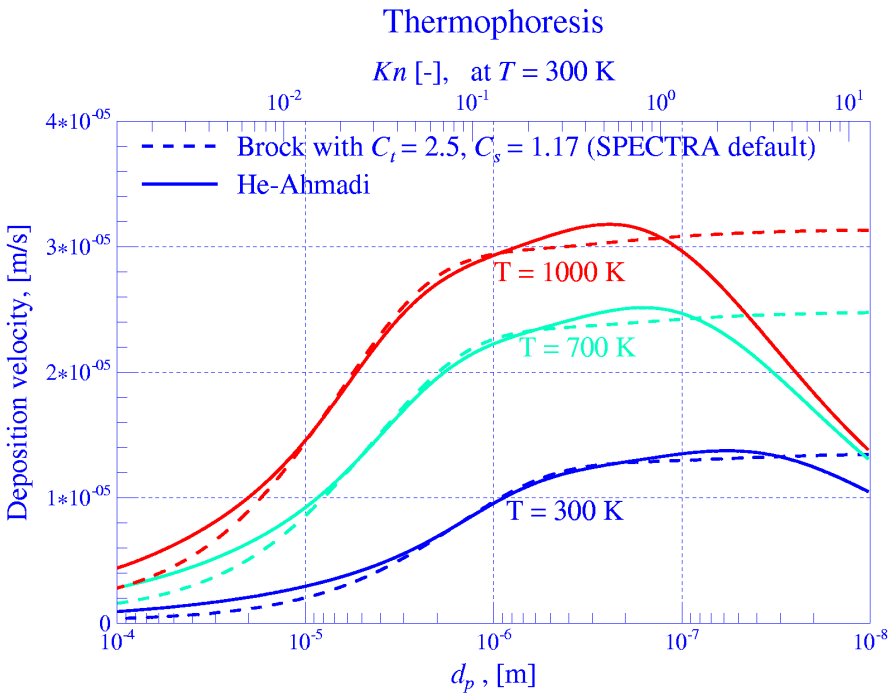


Figure 12-6 Comparison of Brock and He-Ahmadi correlations.

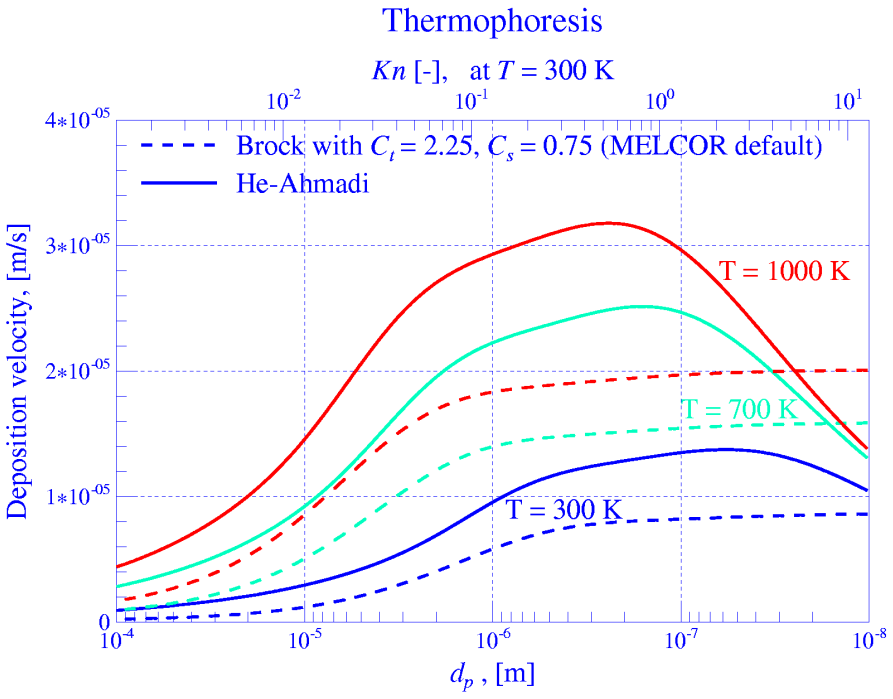


Figure 12-7 Comparison of Brock and He-Ahmadi correlations.

Next α is simplified as follows:

$$\alpha = 0.22\frac{\sqrt{\pi/6}\cdot\sqrt{\varphi}}{\sqrt{1+Kn\cdot\pi_1/2}} = \frac{0.1592\cdot\sqrt{\varphi}}{\sqrt{1+0.291\cdot Kn}}$$

$$\sqrt{\varphi} = \sqrt{0.25\cdot(9\kappa-5)\cdot\frac{c_v}{R}} = \sqrt{0.25\cdot(9\kappa-5)/(\kappa-1)}$$

where κ is the ratio of specific heats. While in theory φ depends on the kind of gas, it can be observed that the square root of φ does not change significantly over all possible values of κ: 1.33 ÷ 1.67:

$$\kappa = 1.33: \quad \sqrt{\varphi} = \sqrt{0.25\cdot(9\cdot1.33-5)/(1.33-1)} = 2.30$$
$$\kappa = 1.40: \quad \sqrt{\varphi} = \sqrt{0.25\cdot(9\cdot1.40-5)/(1.40-1)} = 2.18$$
$$\kappa = 1.67: \quad \sqrt{\varphi} = \sqrt{0.25\cdot(9\cdot1.67-5)/(1.67-1)} = 1.93$$

The average value of the square root of φ is about 2.1 and may differ by only up to about 10% for different gases. The difference of 10% is considered small, compared to the accuracy one can hope to achieve in practical application of the correlation. Therefore it is assumed that:

$$\sqrt{\varphi} = 2.1$$

and finally α is given by:

$$\alpha = \frac{0.33}{\sqrt{1+0.291\cdot Kn}}$$

The formula for D_m, reproduced from [81] is:

$$D_m^2 = \left(\frac{5}{16\mu_g}\right)\cdot\left(\frac{m_m k_B T}{\pi}\right)^{1/2} = \left(\frac{5}{16\mu_g}\right)\cdot\left(\frac{R_m k_B T}{\pi R N_A}\right)^{1/2}$$

(the molecule mass m_m was replaced using the Avogadro number, N_A, and the gas individual and universal gas constants, R, R_m). This formula can also be simplified by combining all constants:

$$D_m^2 = \left(\frac{5}{16}\right)\cdot\left(\frac{R_m k_B}{\pi N_A}\right)^{1/2}\cdot\frac{1}{\mu_g}\left(\frac{T}{R}\right)^{1/2} = \frac{2.43\times10^{-24}}{\mu_g}\sqrt{\frac{T}{R}}$$

After substituting all expressions and simplifying, the expression for the thermophoretic deposition velocity can be written in much simpler form:

$$v_{D,therm} = 1.146\cdot\alpha\cdot Kn^{3/2}\cdot\left[1-\exp(-\alpha/Kn)\right]\cdot C_m d_p\cdot\sqrt{\frac{R}{T}}\cdot\nabla T$$

where:

$$\alpha = \frac{0.33}{\sqrt{1 + 0.291 \cdot Kn}}$$

Kn	Knudsen number, (-)
C_m	Cunningham correction factor, (-)
d_p	particle diameter, (m)
R	individual gas constant, (J/kg/K)
T	temperature, (K)

3. Combination of Brock correlation and He-Ahmadi correlation

Several deposition correlations were compared in [86]. It was concluded that He-Ahmadi correlation gives the most accurate predictions for Knudsen numbers, $0.1 < Kn < 10$. The Brock correlation is valid for Kn up to about 0.1. The Kn number of 10 corresponds to the particle diameters of ~10^{-8} m, while the Kn number of 0.1 corresponds to the particle diameters of ~10^{-6} m.

The default SPECTRA model, consists of both correlations. The Brock correlation is used for $Kn < Kn_1 = 0.1$. The He-Ahmadi correlation is used for $Kn > Kn_2 = 0.2$. An interpolation zone, $Kn_1 < Kn < Kn_2$, is provided to ensure smooth transition from one correlation to another.

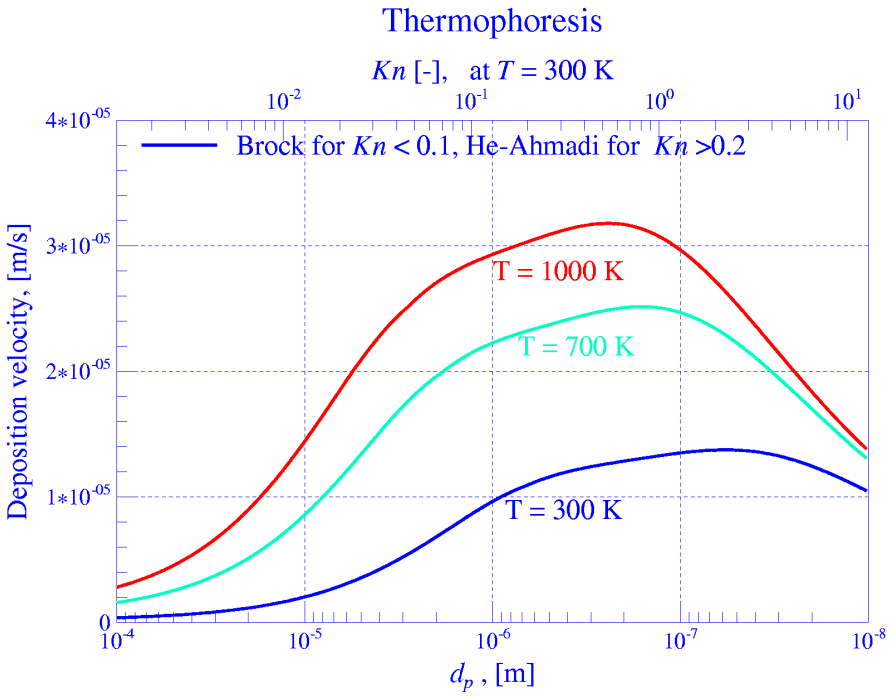


Figure 12-8 Default thermophoresis correlation in SPECTRA; Brock for $Kn < 0.1$, He-Ahmadi for $Kn > 0.2$.

In the interpolation zone the thermophoretic velocity is linearly interpolated between the values obtained from the two correlations for the same Knudsen number:

$$v_{D,therm}(Kn) = v_{Brock}(Kn) + \frac{Kn - Kn_1}{Kn_2 - Kn_1}\left[v_{H-A}(Kn) - v_{Brock}(Kn)\right]$$

When this combination of correlations is used, the He-Ahmadi correlation is multiplied by the ratio of C_s and the default value of C_s: (C_s/1.17). This is done to provide a common user-defined multiplier for the whole thermophoretic deposition model. Thus if the user wants for example to disable the thermophoresis it is enough to set C_s to a small number (exact zero should be avoided because the code will in such case assume the default value). Results obtained with the default correlation are shown in Figure 12-8.

Independently of which model is selected for the thermophoresis calculation, the temperature gradient is always calculated in the same way, described below. The temperature gradient is obtained as a ratio between the gas-to-wall temperature difference, and the thermal boundary layer thickness.

$$\nabla T = \frac{T_g - T_w}{\delta_{BL}}$$

Two options are available to determine the boundary layer thickness:

- User-defined constant value. The default value is (Volume 2):

$$\delta_{BL} = 2.0 \times 10^{-3}\ m$$

- Theoretical model, based on the heat convection through a boundary layer.
  It is assumed that the convected heat is transported through the boundary layer through gas conduction:

$$q = \frac{k_g}{\delta_{BL}}(T_g - T_w)$$

  Where k_g is the gas thermal conductivity. On the other hand, the heat flux may be obtained using the heat transfer coefficient, h:

$$q = h(T_g - T_w)$$

  Therefore:

$$\delta_{BL} = \frac{k_g}{h}$$

The default model for thermophoretic boundary layer thickness is model B (convection through the boundary layer). The same model is used by the MELCOR code.

12.2.5.4 Diffusiophoresis

When water condenses on (evaporates from) a structure surface or pool surface, composition gradients will exist in the adjacent gas, which will affect aerosol deposition on the surface [46]. Two related mechanisms produce these gradients. First, a net molar flux of gas toward (away from) the condensing (evaporating) surface will exist, and this net flux, commonly called the Stefan flow, will tend to move aerosol particles with it. Second, differences in the momentum transfer by molecular impacts on opposite sides of the particle will tend to drive the particle into the direction decreasing concentration of the heavier constituent. In SPECTRA, as in MELCOR, the term diffusiophoresis is used to represent the net result of both effects. Note that when the non-condensable gas is heavier than steam (for example air), the differential molecular impact effect opposes the Stefan flow (which dominates the net result); the effects are in the same direction if the non-condensable gas is lighter than steam (for example helium).

Two models are available in SPECTRA to calculate the diffusiophoretic deposition.

- Model 1 (default) - differential molecular impact effect included in case of both condensation and evaporation.

  Condensation [46]:

$$v_{D,diff} = -\frac{\sqrt{M_s}}{X_s\sqrt{M_s} + X_{NC}\sqrt{M_{NC}}} \cdot \frac{W}{\rho_g}$$

  Evaporation:

$$v_{D,diff} = -\frac{\sqrt{M_s}}{X_s\sqrt{M_s} + X_{NC}\sqrt{M_{NC}}} \cdot \frac{W}{\rho_{sat}}$$

  M_s molar weight of steam (=18.0153), (kg/kmole)
  M_{NC} molar weight of non-condensable gases, (kg/kmole)
  X_s mole fraction of steam in the bulk gas, (-)
  X_{NC} mole fraction of non-condensables in the bulk gas, (-) (= $1 - X_s$)
  ρ_g density of the bulk gas, (kg/m^3)
  ρ_{sat} saturated steam density, (kg/m^3)
  W evaporation/condensation flux, (kg/m^2/s) (positive in case of evaporation)

  The above expressions are not convenient for immediate use in SPECTRA. Firstly, the molecular weights are not stored in the SPECTRA Control Volume data block; secondly, the gas properties available in the Control Volume data block are given for the actual gas mixture and not for the non-condensable gases only. Therefore, the above formulae are somewhat transformed, as shown below.

  The first term is written as:

$$\frac{\sqrt{M_s}}{X_s\sqrt{M_s} + X_{NC}\sqrt{M_{NC}}} = \frac{1}{X_s + X_{NC}\sqrt{\frac{M_{NC}}{M_s}}} = \frac{1}{X_s + (1 - X_s)\sqrt{\frac{M_{NC}}{M_s}}}$$

In order to eliminate the unknown molecular weight of pure noncondensables, it is taken into account that the molecular weight of a mixture is equal to:

$$M = X_s M_s + (1 - X_s) M_{NC}$$

Therefore:

$$M_{NC} = \frac{M - X_s M_s}{1 - X_s}$$

Substituting M_{NC} into the formula, one obtains:

$$\frac{1}{X_s + (1 - X_s)\sqrt{\frac{M - X_s M_s}{(1 - X_s) M_s}}} = \frac{1}{X_s + \sqrt{(1 - X_s)}\sqrt{\frac{M}{M_s} - X_s}}$$

The molecular weights are not present in the SPECTRA Control Volume data block, but the individual gas constants are ($R=R_m / M$, - for steam $R_s = R_m / M_s$ = 8314.51 / 18.0153 = 461.5). Therefore:

$$\frac{1}{X_s + \sqrt{(1 - X_s)}\sqrt{\frac{M}{M_s} - X_s}} = \frac{1}{X_s + \sqrt{(1 - X_s)}\sqrt{\frac{R_s}{R} - X_s}} = \frac{1}{X_s + \sqrt{(1 - X_s)}\sqrt{\frac{461.5}{R} - X_s}}$$

Now the formula is expressed in terms of the steam volume fraction, X_s, and the gas constant of the gas mixture, R, both readily available in the Control Volume data block. The final equations used by SPECTRA are:

Condensation:

$$v_{D,diff} = -\frac{1}{X_s + \sqrt{(1 - X_s)} \cdot \sqrt{461.5 / R - X_s}} \cdot \frac{W}{\rho_g}$$

Evaporation:

$$v_{D,diff} = -\frac{1}{X_s + \sqrt{(1 - X_s)} \cdot \sqrt{461.5 / R - X_s}} \cdot \frac{W}{\rho_{sat}}$$

- Model 2 - differential molecular impact effect included only in case of condensation.

  In this model the differential molecular impact effect is included only in case of condensation. This formulation is the same as the one used in MELCOR ([46], RN Reference Manual, section 2.4.2.2).

Condensation:

$$v_{D,diff} = -\frac{\sqrt{M_s}}{X_s\sqrt{M_s} + X_{NC}\sqrt{M_{NC}}} \cdot \frac{W}{\rho_g}$$

Evaporation:

$$v_{D,diff} = -\frac{W}{\rho_{sat}}$$

Again, after transformation the final equations are

Condensation:

$$v_{D,diff} = -\frac{1}{X_s + \sqrt{(1-X_s)} \cdot \sqrt{461.5/R - X_s}} \cdot \frac{W}{\rho_g}$$

Evaporation:

$$v_{D,diff} = -\frac{W}{\rho_{sat}}$$

The results of both models are shown in Figure 12-9 and Figure 12-10. A few things may be observed in these figures:

- Steam-Helium (light gas) mixtures:
  In case of steam-helium mixtures the differential molecular impact effect enhances the Stefan flow and the deposition velocities are larger (with respect to the absolute value) compared to the steam-heavy gas mixture. This is quite expected since in this case the heavier molecules are those of steam.

- Steam-Air (heavy gas) mixtures:
  In case of steam-air mixtures the differential molecular impact effect opposes the Stefan flow and the deposition velocities are smaller (with respect to the absolute value) compared to the steam-light gas mixtures.

- Model 2:
  Model 2 neglects the differential molecular impact effect in case of evaporation, and therefore results of Steam-Helium and Steam-Air mixtures are the same when W>0.0 (Figure 12-10).

In SPECTRA the diffusiophoresis, as well as all other deposition mechanisms, is applied for walls and the pool surfaces. The evaporation regime is of course only possible in case of pool surface. In case of walls (1-D and 2-D Solid Heat Conductors) only the condensation regime may occur.

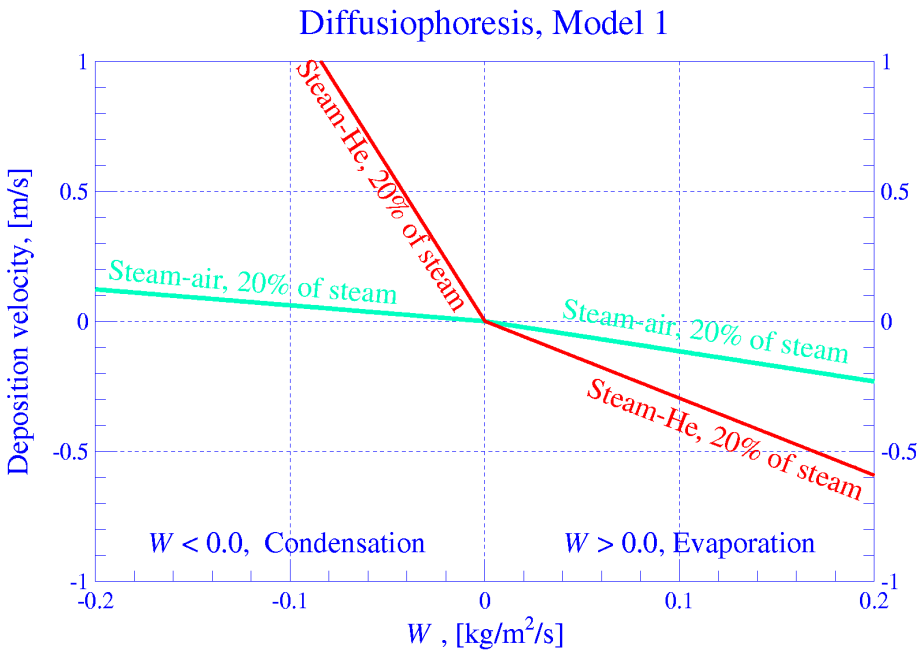


Figure 12-9 Diffusiophoresis, Model 1.

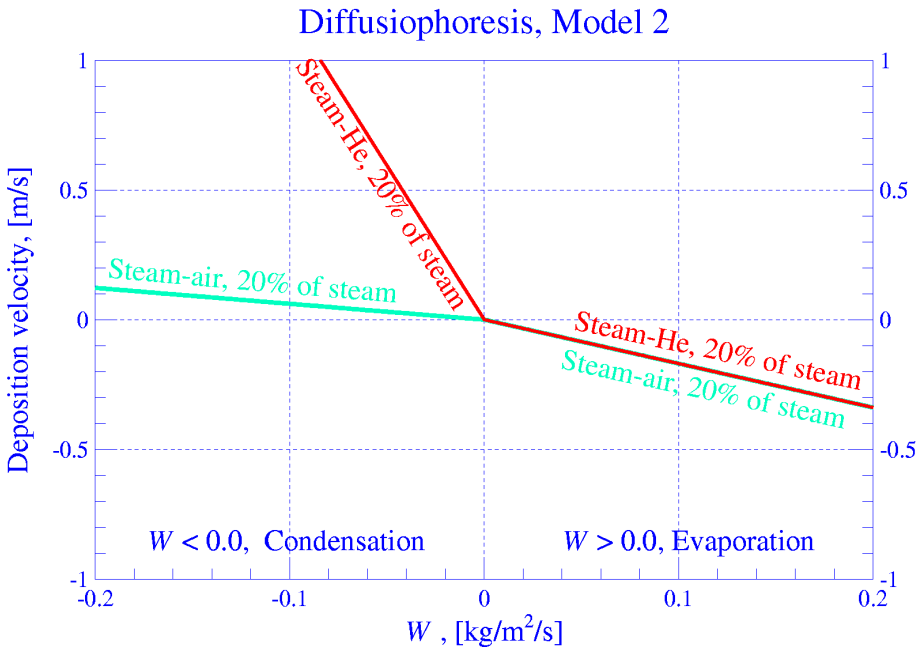


Figure 12-10 Diffusiophoresis, Model 2

12.2.5.5 Turbulent Deposition

The turbulent deposition model implemented in SPECTRA is based on the review presented in [84]. Three different ranges are distinguished:

- Diffusional deposition regime
- Diffusion-impaction regime, also called turbulent impaction regime
- Inertia-impaction regime

The dimensionless deposition velocity data is shown in Figure 12-11 versus dimensionless particle relaxation time, τ_p^+. The particle relaxation time is given by:

$$\tau_p^+ = \frac{\rho_p (d_p^+)^2 C_m}{18\rho_g} = \frac{\rho_g \rho_p d_p^2 C_m (u^*)^2}{18\mu_g^2} = \frac{\rho_g \rho_p d_p^2 C_m f v_g^2}{144\mu_g^2}$$

ρ_p	density of aerosol particle, (kg/m^3)
ρ_g	gas density, (kg/m^3)
d_p^+	dimensionless diameter, = $d_p\, u^* / \gamma_g = d_p\, u^* \rho_g / \mu_g$
d_p	particle diameter, (m)
μ_g	dynamic viscosity of gas, (kg/m-s)
f	friction factor
v_g	gas velocity, (m/s)
C_m	Cunningham correction factor, (-)

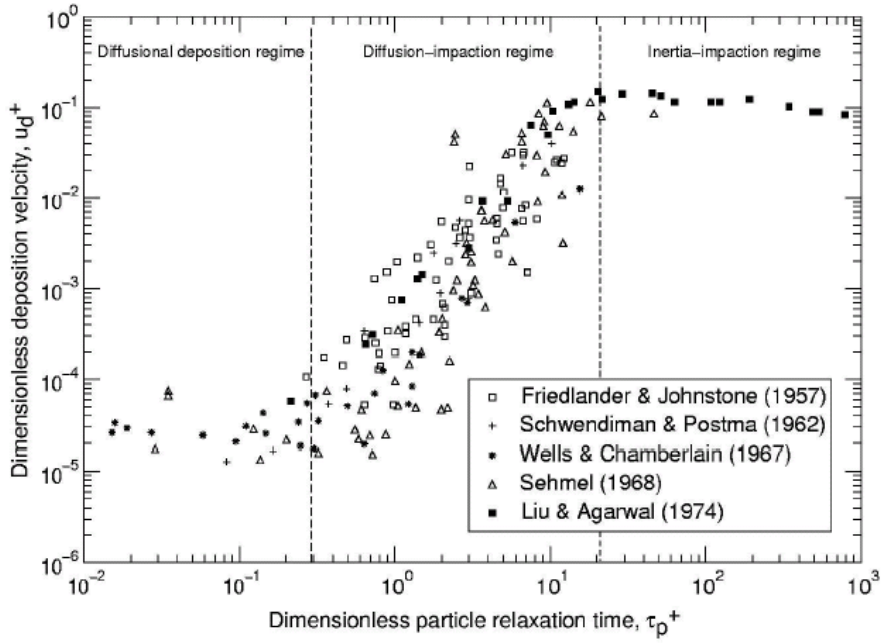


Figure 12-11 Dimensionless turbulent deposition velocities, [84].

The dimensionless deposition velocity is equal to:

$$v_{D,turb}^{+} = \frac{v_{D,turb}}{u^{*}}$$

where:

$$u^{*} = \left(\frac{\tau_w}{\rho_g}\right)^{1/2} = \left(\frac{f\rho_g v_g^2}{8\rho_g}\right)^{1/2} = 0.3536 \cdot \sqrt{f} \cdot v_g$$

τ_w wall shear stress, $= f \rho_g v_g^2$, (N/m^2)
f friction factor, (-)
v_g gas velocity, (m/s)

The deposition correlations applied for each regime are shown below.

- Diffusional deposition regime

  The correlation is based on particle Schmidt number, and is taken from [81]:

$$v_{D,1}^{+} = 0.057 \cdot Sc_p^{-2/3}$$

  The particle Schmidt number is given by:

$$Sc_p = \frac{3\pi\mu_g^2 d_p \chi}{k_B \rho_g T C_m} = 6.83 \times 10^{23} \cdot \frac{\mu_g^2 d_p \chi}{\rho_g T C_m}$$

  μ_g gas viscosity, (kg/m/s)
  T temperature, (K)

  The dynamic shape factor, χ, has been introduced, following [87] (section 2.9.1). Note that in [87] the multiplicative factor is equal to 0.065. The literature review shows values between 0.057 and 0.084 [81], [85].

- Diffusion impaction (turbulent impaction) regime

  The turbulent impaction correlation is taken from [87] (section 2.3.2.1). The correlation is:

$$v_{D,2}^{+} = A \cdot \left(\tau_p^{+}\right)^2$$

  The constant A is a user-defined parameter, with a default value of 4.5×10^{-4}, (selected based on review work performed at NRG).

The definition of τ_p^+ is, according to [87] (section 2.3.2.1):

$$\tau_p^+ = \frac{\rho_p \rho_g d_p^2 v_g^2 C_f}{36 \mu_g^2} = \frac{\rho_p \rho_g d_p^2 v_g^2 f}{36 \mu_g^2 \cdot 4} = \frac{\rho_p \rho_g d_p^2 f v_g^2}{144 \mu_g^2}$$

where f is the friction factor ([21], equation 4.15) and C_f is the skin friction coefficient ([21], equation 4.14), The skin friction coefficient is related to the friction factor by ([21], equation 4.16): $f = 4C_f$.

The above definition differs from the definition in [87] by the presence of the Cunningham correction factor. In [87] C_m is neglected. In SPECTRA a more general formulation, with C_m included, is used. Therefore:

$$\tau_p^+ = \frac{\rho_p \rho_g d_p^2 v_g^2 f}{144 \mu_g^2} C_m = 6.94 \times 10^{-3} \frac{\rho_p \rho_g d_p^2 v_g^2 f}{\mu_g^2} C_m$$

- Inertia impaction regime

For this range [87] (section 2.3.2.1) recommends a constant dimensionless velocity, equal to 0.2. Investigation of Figure 12-11 shows that 0.2 is a good value for the dimensionless particle relaxation times of 20 ÷ 50. For larger particle relaxation times the dimensionless velocity slowly decreases. At the relaxation time of 1000 the value is slightly below 0.1 (about 0.09). A simple correlation was developed to fit this behavior. The SPECTRA inertia impaction correlation is:

$$v_{D,3}^+ = 0.08 + 0.1 \cdot \exp\left(-0.002 \cdot \tau_p^+\right)$$

It can easily be checked that the correlation has the desired properties of:
$v^+_{D,3} = 0.18$ for $\tau_p^+ = 20$
$v^+_{D,3} = 0.09$ for $\tau_p^+ = 1000$

The qualitative shape of the correlation is seen in Figure 12-12. It is seen that the correlation provides a good match for the data presented in Figure 12-11.

The final deposition velocity is calculated from:

$$v_{D,turb}^+ = Max(v_{D,1}^+, Min(v_{D,2}^+, v_{D,3}^+))$$

The full turbulent deposition correlation is shown in Figure 12-12 and Figure 12-13. A constant friction factor of $f = 0.02$ was used to generate data presented in these figures. Normally, during SPECTRA calculations the friction factors are taken from junctions. Therefore the user must specify associations between the depositing surfaces and the junctions. This can be done automatically by the code. Two options are available for automatic associations:

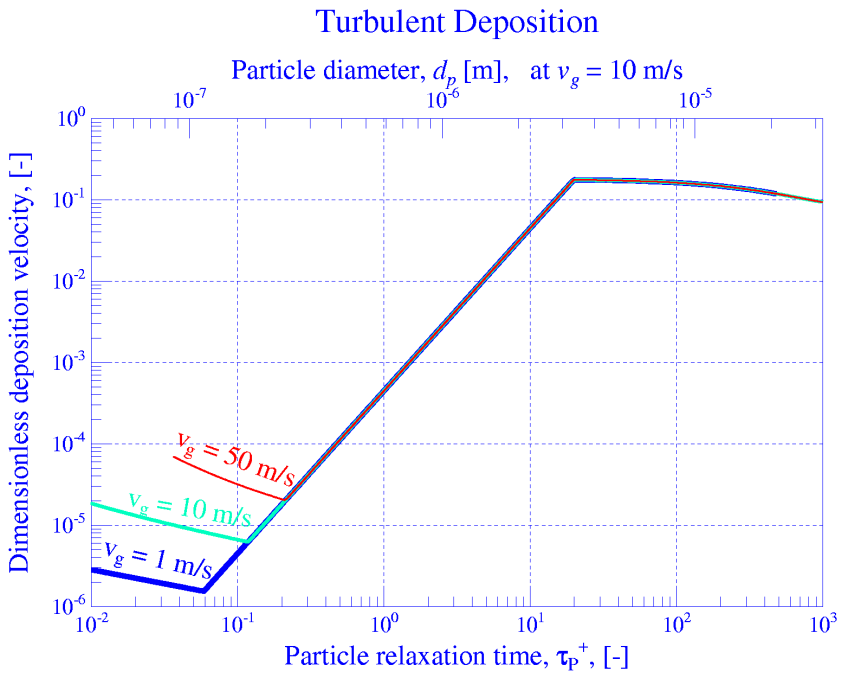


Figure 12-12 Dimensionless turbulent deposition velocities, v_g = 1, 10, 50 m/s.

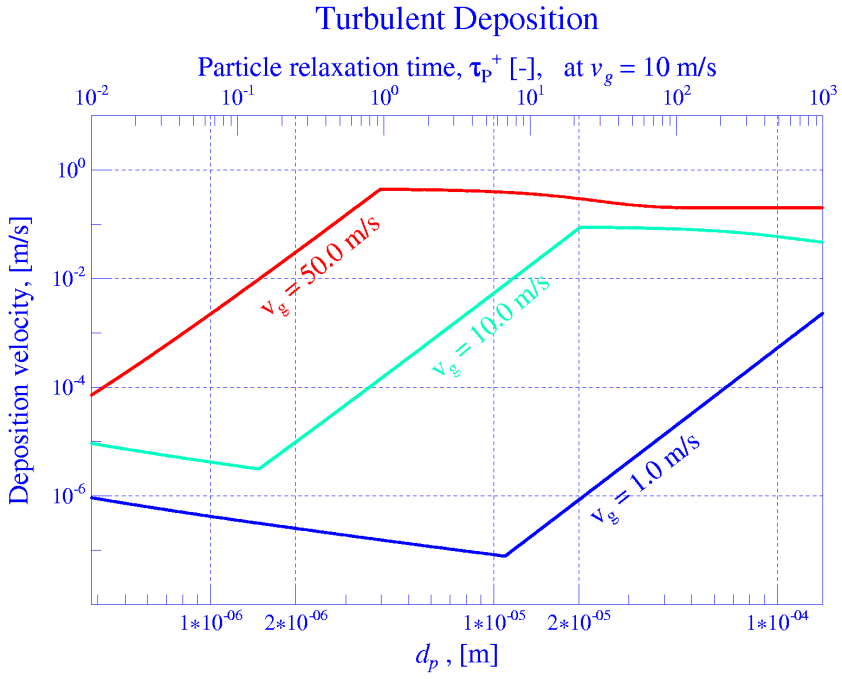


Figure 12-13 Turbulent deposition velocities, v_g = 1, 10, 50 m/s.

- The friction factor is taken from this junction connected to the boundary Control Volume of the depositing surface, which has a minimum flow area.
- The friction factor is taken from this junction connected to the boundary Control Volume of the depositing surface, which has a maximum flow area.

For the turbulent deposition on the pool surface the friction factor is a constant number, and is defined by the user. The default value is: $f = 0.1$ (Volume 2, record 171XXX).

Figure 12-12 shows the dimensionless deposition velocity versus dimensionless particle relaxation time. For comparison the particle diameter calculated for the gas velocity of $v = 10$ m/s is shown at the top axis. The deposition velocities were calculated by SPECTRA for three gas velocities: 1.0 m/s, 10.0 m/s, and 50.0 m/s. The dependence of the dimensionless deposition velocity on the gas velocity is visible only in the first (diffusional deposition) regime (see Figure 12-11). In the two other regimes the dimensionless deposition velocity is practically independent on the gas velocity. The results presented in Figure 12-12 well represent the experimental data shown in Figure 12-11.

Figure 12-13 shows the same data as Figure 12-12, but this time the actual (dimensional) deposition velocity and the particle diameters are used on the *x-y* axes. For comparison the dimensionless particle relaxation times for $v = 10$ m/s are shown at the top axis. When the dimensional parameters are used, it is clearly seen that for a given particle diameter the turbulent deposition strongly depends on the gas velocity. For example, for a particle of 10^{-5} m, the turbulent deposition velocity is clearly below 10^{-6} m/s for the gas velocity of 1.0 m/s, a little below 10^{-2} m/s for the gas velocity of 10.0 m/s, and a little below 1 m/s for the gas velocity of 50 m/s. Thus it can be roughly said that the turbulent deposition becomes important for the gas velocities above 10 m/s.

The above formulae for turbulent deposition are valid for turbulent flow. In case of laminar flow the turbulent deposition is zero. Therefore a Reynolds number dependent multiplier, E_t, is introduced in SPECTRA. The value of the multiplier is defined in the same way as the multiplier on the turbulence dissipation rate (section 12.2.3).

- $E_t = 1.0$ if $Re > Re_{tur}$,
- $E_t = 0.0$ if $Re > Re_{lam}$,
- Third order (smooth) interpolation if $Re_{lam} < Re < Re_{tur}$.

The multiplier is shown in Figure 12-14. The limiting Reynolds numbers, Re_{lam}, Re_{tur}, are user-defined, with default values of 2200 and 10,000 respectively (the same values are used for the turbulence dissipation rate – section 12.2.3). Therefore the turbulent deposition does not occur in stagnant atmosphere and in laminar flow.

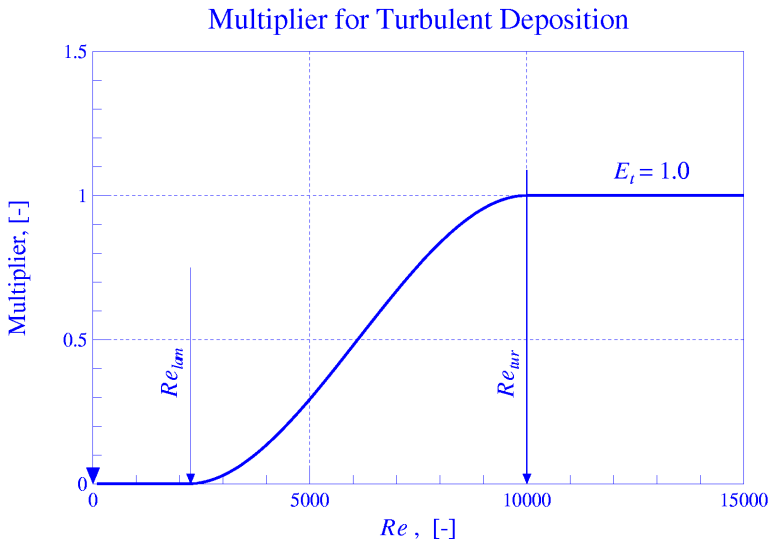


Figure 12-14 Multiplier on the turbulent deposition correlation.

12.2.5.6 Inertial Impaction Deposition

The inertial impaction deposition model implemented in SPECTRA is based on the review presented in [81]. Generally the inertial impaction model calculates the inertial impaction collection efficiency, defined as [180]:

$$\eta = 1 - \frac{m}{m_{in}} = \frac{m_{dep}}{m_{in}}$$

Here m_{in} is the incoming mass of aerosols, m is the mass passing through and m_{dep} is the deposited mass. The collection efficiency is calculated either through one of the built-in correlations or a user-defined correlation.

The following correlations are built-in:

- The Ciemat correlation [180], appropriate for tube bundle. The correlation was developed for steam generator tube rupture. The aerosols coming through the ruptured tube deposit on the surface of the neighboring tube or tubes - Figure 12-16 (a). The correlation is:

$$\eta = \frac{\eta_{\max}}{1 + c_1 \cdot \exp(-c_2 \cdot Stk^{c_3})}$$

Here $\eta_{\max}$, c_1, c_2, c_3 are user defined coefficients with the default values of 0.75, 29.31, 3.85, and 0.5 respectively, based on [180]. *Stk* is the Stokes number, defined as ([75], eq. 9.23):

$$Stk = \frac{C_m \cdot \rho_p \cdot d_p^2 \cdot v_g}{18 \cdot \mu_g \cdot D}$$

C_m	Cunningham correction factor, (-)
ρ_p	particle density, (kg/m^3)
d_p	particle diameter, (m)
v_g	gas velocity, (m/s)
μ_g	gas viscosity, (kg/m/s)
D	impactor dimension, (m)

- The collection efficiency is calculated from the correlation appropriate for a tube bend [181] - Figure 12-16 (c):

$$\eta = \eta_{\max} \cdot (1 - 10^{-c_1 \cdot Stk})$$

  Here $\eta_{\max}$, c_1, are user defined coefficients with the default values of 1.0 and 0.963 respectively, based on [181].

- The collection efficiency is calculated from a general correlation as a function of Stokes number:

$$\eta = f(Stk)$$

  The function f is defined using a Tabular Function. The independent variable for calculating the function is the Stokes number.

- The collection efficiency is calculated from a Control Function. With this option any function may be built-in. The program checks the value obtained from the Control Function and automatically resets the value to lie within the physical values:

$$0.0 \le \eta \le 1.0$$

Comparison of the built-in correlation and an arbitrary tabulated as a function of the Stokes number, are shown in Figure 12-15. The values of the tabulated function is shown in Table 12-3. Further discussion and comparisons with experimental data is shown in Volume 3.

The inertial impaction is implemented in SPECTRA in such a way that two options are available, each option calculates the deposition rate using different parameters.

- **Deposition surface associated with a junction**. In such case the deposition rate is calculated based on the mass flow of aerosols through the associated junction.
- **Deposition surface not associated with a junction**. In such case the deposition rate is calculated as a product of the deposition velocity and the aerosol concentration in the volume next to the structure surface.

These two options are discussed below.

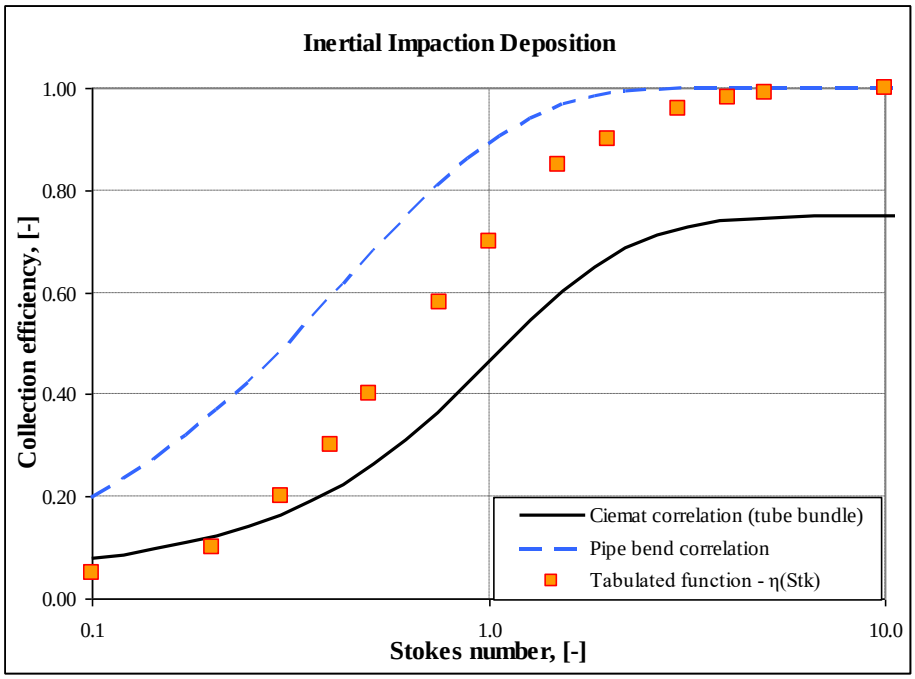


Figure 12-15 Comparison of the inertial impaction correlations available in SPECTRA.

Table 12-3 An arbitrary tabular function defining collection efficiency

Stk	$\eta = f(Stk)$
0.10	0.05
0.20	0.10
0.30	0.20
0.40	0.30
0.50	0.40
0.75	0.58
1.00	0.70
1.50	0.85
2.00	0.90
3.00	0.96
4.00	0.98
5.00	0.99
10.0	1.00

Deposition surface associated with a junction

In this case, the deposition rate is calculated based on the flow in the associated junction. An example shown in Figure 12-16 (b). The inertial impaction on SC-321 is calculated based on aerosol flow through JN-301.

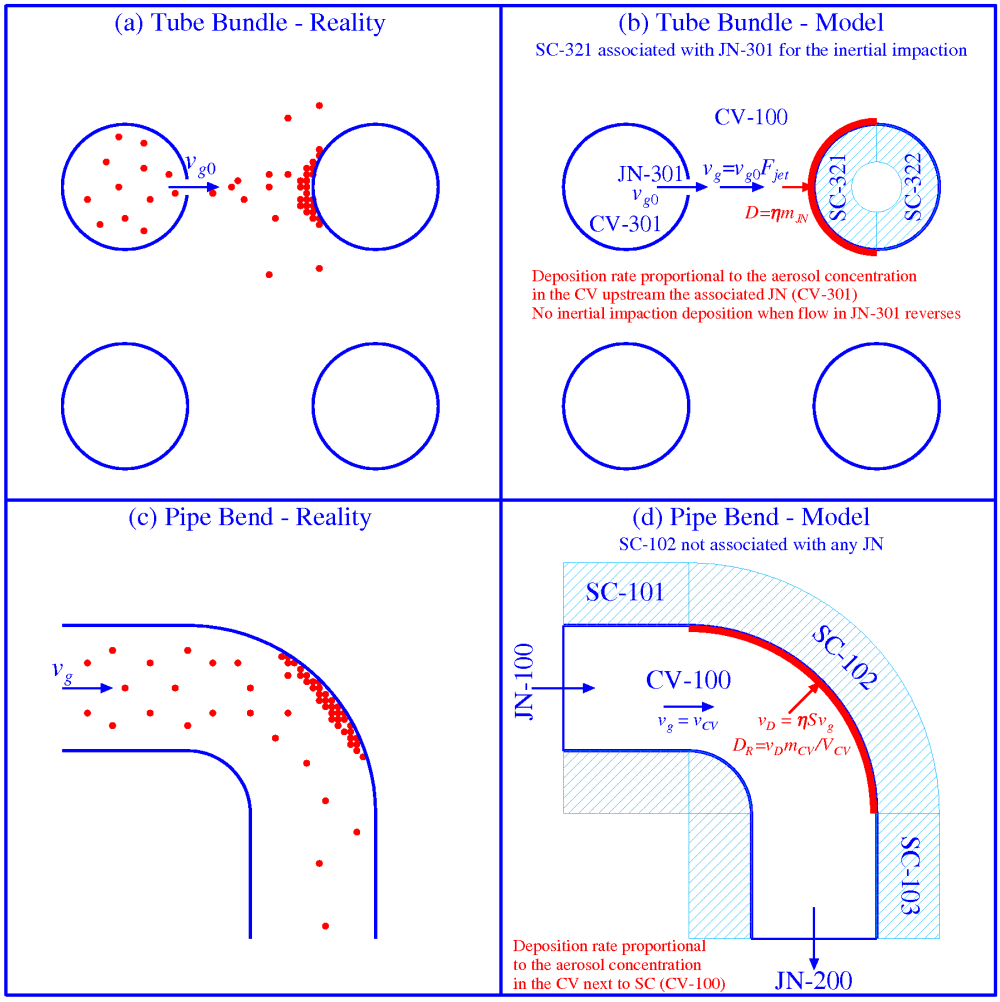


Figure 12-16 Inertial impaction models in SPECTRA

Suppose the gas velocity in JN-301 is v_{g0} (m/s), and the mass flow of aerosol particles through this junction is m_{JN} (kg/s). The code will use the gas velocity to calculate the Stokes number. However one may argue that the gas velocity in the junction does not necessarily apply for the conditions close to the deposition surface. The distance between the deposition surface (in this case SC-321) and the break jet (JN-301) is an important factor. If the distance is large, then the jet of gas created at the break will be dissipated and the gas velocity near the deposition surface will be reduced. This fact is taken into account in SPECTRA by introducing a parameter called the jet dissipation factor, F_{jet}. The gas velocity that is used in the Stokes number is equal to:

$$v_g = v_{g0} \cdot F_{jet}$$

Here v_{g0} is the gas velocity in the associated junction (representing the tube break). The deposition rate, D (kg/s), is obtained from:

$$D = \eta \cdot m_{JN}$$

Here m_{JN} (kg/s) is the mass flow of aerosol particles through the junction. Therefore with this option on, the deposition rate is proportional to the aerosol concentration in the Control Volume upstream the associated junction. In the present example this is CV-301. Note that the aerosol concentration in the volume next to the deposition surface (in this case CV-100) may be very low just because most of the aerosols coming through the break are deposited due to the inertial impaction. Therefore this kind of deposition requires different approach than that adopted for the other deposition mechanisms, where the deposition rate is simply a product of the deposition velocity and the aerosol concentration in the volume next to the structure surface.

The numerical scheme applied for the present option is very similar to the one applied for filter calculations, because this is what is really happening during the process - particles leaving one CV and never enter the other volume. Of course the deposition is expected to occur only if the gas velocity is towards the deposition surface. The deposition is set to zero when the gas velocity reverses.

The user may associate more than one surface with the same junction. When several surfaces are associated with the same junction, then the collection efficiency is calculated using fractions, $F_{R,i}$, defined by the user:

$$D_i = \eta \cdot m_{JN} \cdot F_{R,i}$$

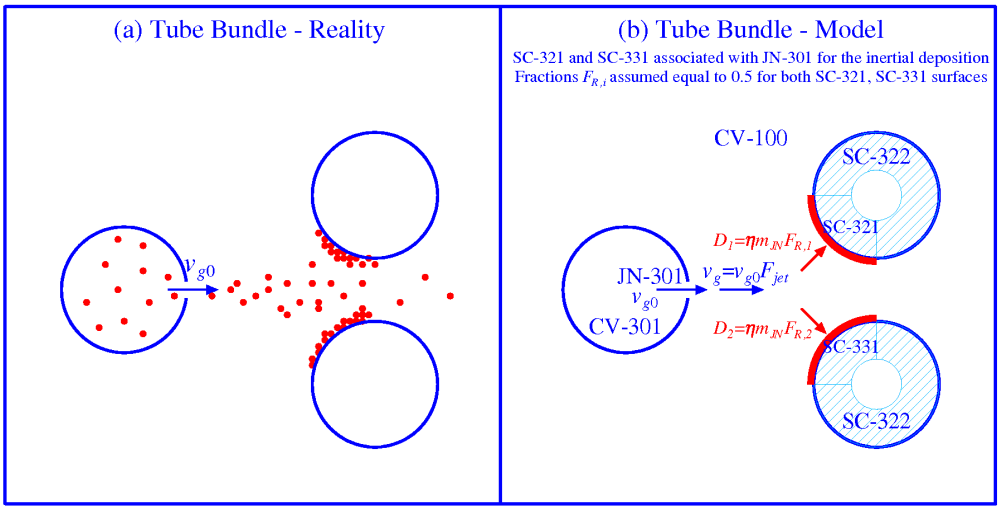


Figure 12-17 Inertial impaction with multiple deposition surfaces involved

The following condition must be fulfilled in defining $F_{R,i}$:

$$\sum_i F_{R,i} \leq 1.0$$

This condition must be fulfilled, otherwise the total collection efficiency would be higher than 1.0, in other words more particles would be collected than are coming in through the junction. A case when such modelling may be useful is shown in Figure 12-17. The inertial deposition occurs on two tubes neighboring to the break - Figure 12-17 (a). The model contains two structures on which the inertial deposition is active - Figure 12-17 (b). Both structures (SC-321, SC-331) are associated with the same junction (JN-301), and both fractions, $F_{R,i}$, are assumed to be equal to 0.5.

Deposition surface not associated with a junction

In this case the deposition rate is calculated as a product of the deposition velocity and the aerosol concentration in the volume next to the structure surface. This is a typical approach, used for all other deposition mechanisms. In order to perform calculations the deposition velocity in meters per second is needed. The collection efficiency must be therefore converted into the deposition velocity. This is done using the following reasoning.

In absence of the inertial impaction the aerosol density in a gas space is equal to the aerosol mass in a Control Volume, $m_{0,CV}$ (kg), divided by the gas volume in the CV, V_{CV} (m^3): (m_{CV}/V_{CV}). If the gas velocity is v_g, the particle velocity is ($S \cdot v_g$), where S is particle slip factor. The mass flux of incoming particles is equal to ($m_{0,CV}/V_{CV}$)·($S \cdot v_g$). The collection efficiency is η, therefore the mass flux of $\eta \cdot (m_{0,CV}/V_{CV}) \cdot (S \cdot v_g)$ is deposited. The total deposition rate is obtained by multiplying the deposition flux by the gas flow area, A_g. The deposition rate in kg/s is therefore equal to:

$$D = \eta \cdot A_g \cdot \frac{m_{0,CV}}{V_{CV}} \cdot S \cdot v_g$$

When the inertial deposition takes place the aerosol mass becomes smaller. The aerosol mass with the inertial impaction taken into account, m_{CV}, is related to the aerosol mass in absence of the impaction, $m_{0,CV}$, by: $m_{CV} = m_{0,CV}(1 - \eta)$. Therefore:

$$D = \frac{\eta}{1-\eta} \cdot A_g \cdot \frac{m_{CV}}{V_{CV}} \cdot S \cdot v_g$$

On the other hand, the deposition rate is obtained in SPECTRA from a general relation:

$$D = \left(\frac{A_D \cdot v_D}{V_{CV}} \right) \cdot m_{CV}$$

Here A_D is the deposition area (m^2), v_D is the deposition velocity (m/s), V_{CV} is the volume of the CV gas space (m^3), and m_{CV} is the aerosol mass (kg). Comparison of the above formula with the formula with the collection efficiency yields:

$$\frac{\eta}{1-\eta}\cdot A_g \cdot \frac{m_{CV}}{V_{CV}}\cdot S \cdot v_g = \left(\frac{A_D \cdot v_D}{V_{CV}}\right)\cdot m_{CV}$$

and:

$$v_D = \frac{\eta}{1-\eta}\cdot S \cdot v_g \cdot \frac{A_g}{A_D}$$

The total gas velocity in a CV consists of the horizontal, $v_{g0,H}$, and the vertical, $v_{g0,V}$, velocity. The corresponding vertical, $A_{g,V}$, and horizontal, $A_{g,H}$, flow areas must be used. Therefore:

$$v_D = \frac{\eta}{1-\eta}\cdot S \cdot v_g \cdot \frac{v_{g0,H}\cdot A_{g,V} + v_{g0,V}\cdot A_{g,H}}{v_{g0}\cdot A_D}$$

The gas velocity, v_g, is as before:

$$v_g = v_{g0}\cdot F_{jet}$$

This time the reference velocity, v_{g0}, is the velocity in the Control Volume next to the deposition surface. The final expression for the deposition velocity is:

$$v_D = \frac{\eta}{1-\eta}\cdot S \cdot F_{jet} \cdot \frac{v_{g0,H}\cdot A_{g,V} + v_{g0,V}\cdot A_{g,H}}{A_D}$$

An example of such case is shown in Figure 12-16 (d). The tube bend is represented by CV-100 and three structures, SC-101, 102, and 103. The inertial impaction is activated on SC-102. The impactor dimension, D_i, is in this case equal to the pipe radius (see [182]). The gas velocity is equal to the tube average velocity, therefore $v_g = v_{g0}$, therefore $F_{jet} = 1.0$.

12.2.5.7 Electrophoretic Deposition

Charged particles placed in an electrical field experience a force which is similar to the gravitational force. Hence the resulting velocity can be determined in the same way as the gravitational settling velocity ([76], section 2.2). The derivation of the equation for the electrophoretic deposition and analogy with gravitational field is shown in Table 12-4.

Table 12-4 Analogy between gravitational deposition and electrophoretic deposition

	Gravitational field	Electrical field
Particle velocity (m/s)	$v_{D,grav} = B \cdot F_{grav}$	$v_{D,e} = B \cdot F_e$
Particle mobility (m/Ns)	$B = C_m / (3\pi D_p \mu_g \chi)$	$B = C_m / (3\pi D_p \mu_g \chi)$
Force (N)	$F_{grav} = m_p g = (\pi D_p^3 \rho_p / 6) \cdot g$	$F_e = qE$
Final expression (m/s)	$v_{D,grav} = (C_m \rho_p D_p^2 g)/(18\mu_g \chi)$	$v_{D,e} = (C_m qE)/(3\pi D_p \mu_g \chi)$

The following symbols are used:

$v_{D,grav}$	deposition velocity due to gravity, (m/s)
$v_{D,e}$	deposition velocity due electrostatic field, (m/s)
F_{grav}	gravity force, (N)
F_e	electrostatic force, (N)
B	particle mobility, (m/Ns)

m_p	mass of particle, (kg)
ρ_p	density of particle, (kg/m^3)
D_p	diameter of particle, (m)
g	gravity acceleration, (m/s^2) (= 9.80665, [32])
C_m	Cunningham correction factor (see section 12.2.3), (-)
μ_g	viscosity of gas, (kg/m/s)
χ	dynamic shape factor, (-)
q	electric charge of particle, (C)
E	electric field strength, (V/m)

The expression for electrophoretic deposition calculation is:

$$v_{D,e} = \frac{1}{3\pi} \frac{C_m}{D_p \mu_g \chi} qE$$

If the electrophoretic deposition is to be taken into account in the calculations, the user must define two parameters: the electric field strength, *E*, and the net electric charge for the particle, *q*.

- **Electric field strength, *E***

  The electric field strength is defined independently for each deposition surface of 1-D and 2-D Solid Heat Conductors (see Volume 2, records 361XXX and 461XXX). It represents the field strength in the vicinity of this particular surface. The value must give only the part of the field strength vector that is normal (perpendicular) to the surface. A positive sign means the electric field is directed towards the surface and will have positive contribution to the deposition velocity. A negative sign means the electric field is directed out of the surface and will have negative contribution to the deposition velocity.

- **Net electric charge, *q***

  The particle charge is calculated in SPECTRA from the following formula:

$$q = C_q \cdot (D_p \times 10^6)^{x_q}$$

  Here D_p is the particle diameter (m), C_q and x_q are user-defined parameters (see Volume 2, records 86501, 86502).

  - Minimum charge. An equilibrium charge is given by ([75], equation 15.32, Figure 15.6): $n = 2.37 \times (D_p \times 10^6)^{1/2}$. Here n is the number of elementary charges. The charge is given by $q = ne$ ($e = 1.6 \times 10^{-19}$ C - [75], equation 15.7). Therefore the equilibrium charge is:

$$q = 3.8 \times 10^{-19} \cdot \sqrt{D_p \times 10^6}$$

    This is achieved by setting $C_q = 3.8 \times 10^{-19}$, $x_q = 0.5$.

- Maximum charge. A maximum charge of a negatively charged sphere is given by ([75], equation 15.6, Figure 15.6): $n = 156{,}000\times(D_p\times10^6)^2$. Therefore the maximum charge is:

$$q = 4.5\times10^{-14}\cdot(D_p\times10^6)^2$$

This is achieved by setting $C_q = 4.5\times10^{-14}$, $x_q = 2.0$.

12.2.6 Coagulation of Deposited Aerosols

Before the resuspension models are described it is important to describe what is happening with the deposited aerosol particles. The current SPECTRA version contains a very simple model, which allows to take into account eventual coagulation of particles upon their deposition on the surfaces of 1-D and 2-D Solid Heat Conductors.

The deposited particles are expected to remain separated in case of a mono-layer deposit; that is when the average deposition layer thickness is smaller than the diameter of the smallest deposited particles. If the layer thickness becomes larger, it is possible that the deposited particles stick together, to form larger agglomerates. An important consequence of this fact is that if the particles should become airborne at some later stage, then the size distribution of the resuspended particles may be very different than the size of the deposited particles.

In SPECTRA the deposited aerosol particles of a given size section are assumed to coagulate into a larger size section (characterized by its diameter, d_i) if the average layer thickness, t, exceeds the diameter, d_i, multiplied by a user-defined factor:

$t \geq d_i \times$ XDC1RT (for 1-D Solid Heat Conductors)
$t \geq d_i \times$ XDC2RT (for 2-D Solid Heat Conductors)

The default value of the parameters XCD1RT, XCD2RT is 1.0 (see Volume 2).

Use of the parameter XCD1RT (as well as XCD2RT) is illustrated in Figure 12-18. With the default value (1.0, Figure 12-18, a.), the particles of the smallest size section will be transformed into section 2 when the layer thickness becomes equal to (or larger than) the diameter of the section 2 (d_2). Similarly, the particles of section 2 will be transformed to the size section 3, as soon as the deposited thickness becomes equal or larger than d_3.

If the value of XCD1RT is larger than 1.0 (for example 1.2, Figure 12-18, b.), the particles will agglomerate later. In the present example the smallest section will be transformed into section 2 when the layer thickness becomes equal (or larger) to $1.2\times(d_2)$.

Finally, if the value is smaller than 1.0, then the particles will agglomerate faster. In the present examples (Figure 12-18, c.) particles of sections 1 and 2 will be transformed to the size section 3 when the layer thickness becomes (or larger) to $0.8\times(d_3)$.

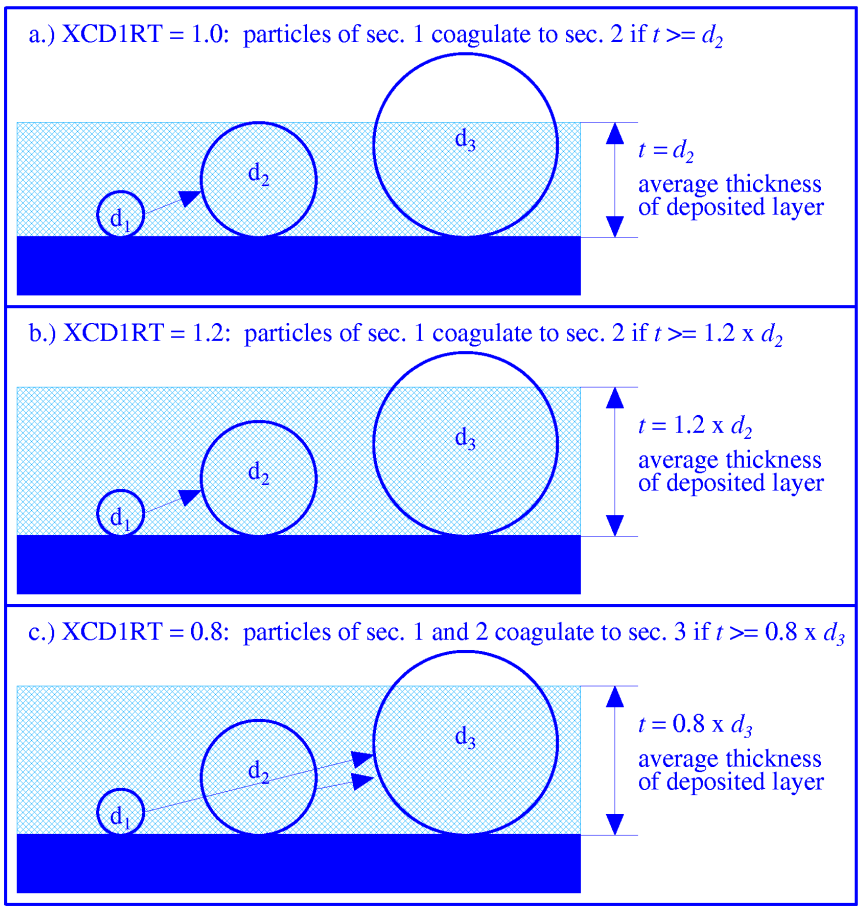


Figure 12-18 Coagulation of deposited aerosol particles.

12.2.7 Aerosol Resuspension - Parametric Model

Two resuspension models are available in SPECTRA: a parametric resuspension model and a mechanistic model. The parametric model is described in this section. The mechanistic model is described in the next section.

The parametric model contains a general polynomial function to calculate the resuspension rate. If the parametric resuspension model is used, then a change of the deposited mass due to resuspension (the resuspension term *R*, in the aerosol balance equation in section 12.2.4) is given by:

$$R = \left.\frac{dm(t)}{dt}\right|_{resusp} = -R_t(t) \cdot m(t)$$

$R_t(t)$ is the total resuspension rate per unit time (1/s), calculated from the parametric resuspension model, described below. The resuspension can be a result of:

- Large velocity of the gas flowing along the surface with deposited aerosols.
- Fast depressurization.
- A combination of both gas velocity and depressurization rate.

The resuspension rate is calculated for each process from a linear and power function:

- Velocity-dependent resuspension:

$$R_v(t) = C_1(v - v_0) + C_2(v - v_0)^x$$

R_v resuspension rate due to gas velocity, (1/s), $R_v = (1/m)(dm/dt)$, where m is the deposited mass (kg)
v fluid velocity parallel to the deposition surface, (m/s)
v_0 minimum fluid velocity for resuspension, (m/s)
C_i model coefficients (with an internal limit of: $0.0 \le C_i \le 1.0$)
x exponent ($0.1 < x < 10.0$)

The minimum velocity for resuspension and the coefficients are calculated from the following formulae:

$$v_0 = v_{00} + v_{0h}H + v_{0n}n + v_{0T}T$$
$$C_1 = C_{10} + C_{1h}H + C_{1n}n + C_{1T}T$$
$$C_2 = C_{20} + C_{2h}H + C_{2n}n + C_{2T}T$$

H relative humidity, (-) (further described below)
n concentration of airborne particles, ($1/m^3$)
T temperature, (K)

The coefficients, C_{ij}, may be specified for each size section independently, or simultaneously for all size sections (see Volume 2).

- Depressurization-dependent resuspension:

$$R_p(t) = C_1\left(\frac{dp}{dt} - \left(\frac{dp}{dt}\right)_0\right) + C_2\left(\frac{dp}{dt} - \left(\frac{dp}{dt}\right)_0\right)^x$$

R_p resuspension rate due to pressure change, (1/s), $R_p = (1/m)(dm/dt)$, where m is the deposited mass (kg).
dp/dt depressurization rate, (Pa/s) (positive when pressure decreases)
$(dp/dt)_0$ minimum depressurization rate for resuspension, (Pa/s)
C_i model coefficients (with an internal limit of: $0.0 \le C_i \le 1.0\times10^{-3}$)
x exponent ($0.1 < x < 10.0$)

The minimum depressurization rate for resuspension and the coefficients are calculated from the following formulae:

$$\left(\frac{dp}{dt}\right)_0 = \left(\frac{dp}{dt}\right)_{00} + \left(\frac{dp}{dt}\right)_{0h} H + \left(\frac{dp}{dt}\right)_{0n} n + \left(\frac{dp}{dt}\right)_{0T} T$$

$$C_1 = C_{10} + C_{1h}H + C_{1n}n + C_{1T}T$$
$$C_2 = C_{20} + C_{2h}H + C_{2n}n + C_{2T}T$$

H relative humidity, (-) (further described below)
n airborne particle concentration, ($1/m^3$)
T temperature, (K)

The coefficients, C_{ij}, may be specified for each size section independently, or simultaneously for all size sections (see Volume 2).

- Total resuspension rate

  The total resuspension rate is obtained from:

$$R_t(t) = 1.0 - (1.0 - R_v(t)) \cdot (1.0 - R_p(t))$$

- Influence of humidity

  The resuspension behavior depends on a long-term humidity rather than an instantaneous humidity at the moment of resuspension. The user can choose one of the following options:

  (1) *H* = current relative humidity in the Control Volume adjacent to the surface,
  (2) *H* = user-defined input entry,
  (3) *H* = maximum relative humidity recorded over the calculated time period.
  (4) *H* = maximum of (2) and (3).
  (5) *H* = time averaged relative humidity over the calculated time period.
  (6) *H* = maximum of (2) and (5).

The resuspension coefficients as well as the limiting velocity depend on the relative humidity, concentration of airborne particles, and temperature. The influence of humidity was observed in resuspension experiments [88]. Mono-layer particles subject to air currents can be resuspended by impaction of other particles [75]. Therefore the influence of airborne particles concentration, *n*, is present. The resuspension measurements indicate temperature dependency [75], *T*, which is therefore included.

A function of the exposure time to a given flow has been introduced based on the work of Fromentin [93]. The resuspension flux at a constant air speed was correlated as proportional to the time the deposit had been exposed to the flow, Δt, to a power of $(-b)$, where *b* is an empirical coefficient:

$$R \sim \Delta t^{-b}$$

Implementation of such relation into a computer code is not straightforward. Firstly, the gas velocity is typically not constant but varies in time. Therefore a reasonable formulation must be adopted; a formulation such that when the gas velocity becomes constant, approximates well the relation shown above. Secondly, a formulation should be reasonably insensitive to the applied time step. This is quite important, because in numerical calculations the formulation will depend on available values of gas velocity at the new and old time step (and not the total history of gas velocities). Since the function must be built based on only the new and the old time step value, as well as the time step size, one must make sure that when different time steps are applied the same solution is obtained.

In SPECTRA the equation of Fromentin is applied over a single time step and has been approximated by an exponential, rather than a power function:

$$R \sim \exp(-B\Delta t)$$

This formulation has the advantage of giving results that are practically insensitive to the time step, and may be applied with small time steps. Two models are available:

Model 1:

$$R = R_1 \cdot \exp(-B\Delta t) + R_2$$

In the above formula R_1 is an "old resuspension", equal to a minimum of the actual old time step resuspension rate and the new time step resuspension rate calculated from the velocity-dependent and pressure-dependent correlation, $R_0 = R_t(t)$. R_2 is a "new resuspension", equal to the difference (if positive) of the new R_0 and the old R_0 (i.e. it is the excess of the velocity- or pressure-induced resuspension, compared to the previous time step). Results of this model are shown in Figure 12-19. The resuspension function calculated from the gas velocity or pressure change, R_0, increases during the first 10 seconds, then remains constant for 10 seconds, and finally linearly decreases to zero during the next 10 s. The actual resuspension rates are shown for three values of the user-defined exponent B: 0.01, 0.1, and 1.0.

The results are not sensitive to the applied time step, which was verified by running the present test with the time steps of 0.1 s and 1.0 s. In case of B=1.0, it is clearly seen that the actual resuspension rate is always much smaller than that calculated from the time-independent, R_0. This is the case if the velocity increase is slow compared to the characteristic time for decay (~1/B).

Model 2:

$$R = Max\left[\ R_1 \cdot \exp(-B\Delta t),\ R_2\ \right]$$

This model is different from model 1 in case of increasing gas velocity (and thus increasing R_0). With model 2 the actual resuspension rate follows the calculated line, R=R_0, when R_0 is increasing. An increase rate of 10% per second has been defined as a minimum limit to consider the release rate as really increasing. This was done in order to filter out any small changes of velocity. The results are illustrated in Figure 12-20.

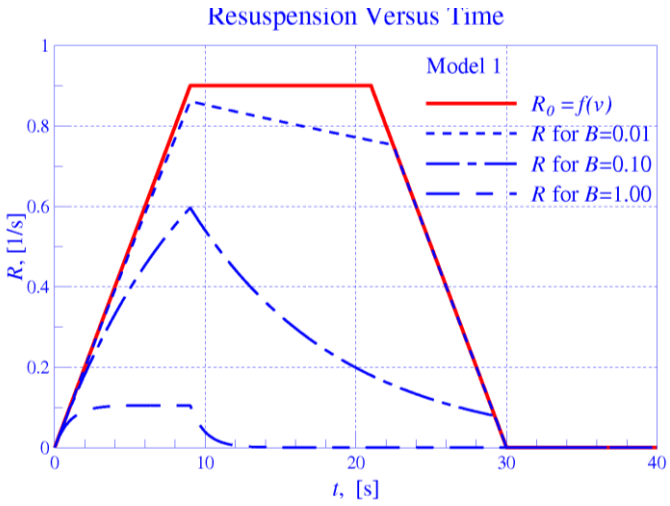


Figure 12-19 Resuspension versus exposure time – model 1.

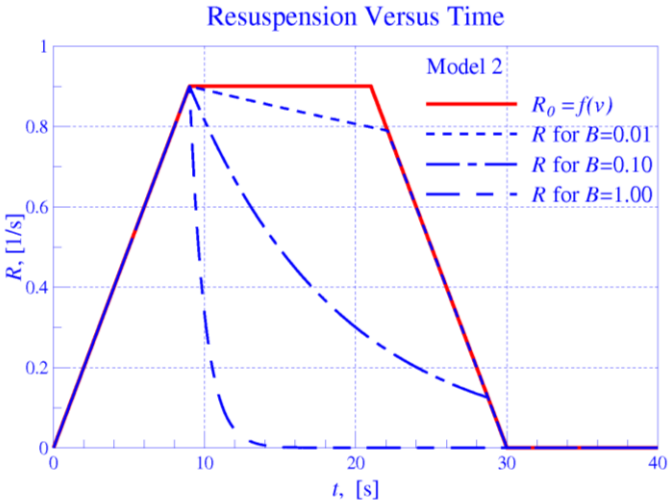


Figure 12-20 Resuspension versus exposure time – model 2.

The user must select model 1 or 2 and provide the value of the exponent *B* (see Volume 2). The recommended model is the Model 1 (default value of IRMTRT – see Volume 2). The recommended value of *B* is 0.1 (default value of BRMTRT, see Volume 2), based on the data of Paci et al. [94].

A disadvantage of the parametric model is illustrated by the "double" test below. In this test a double periods with the same large velocity are applied, separated by a short period without any gas flow - Figure 12-22. Deposited particles are resuspended for a short period (governed by the exponent *B*) after each velocity increase periods - Figure 12-23.

In reality the deposited particles have a distribution of adhesion forces, i.e. some particles are weakly bound to the surface and for some particles the bond is strong. After the first high velocity period the weakly bound particles are resuspended, while the strongly bound particles remain. During the second high velocity period the gas velocity is exactly the same as in the first step. Consequently the drag force acting on particles is the same and the particles that have survived it in the first step are also not resuspended in the second step.

This behavior is seen when a mechanistic resuspension model (Vainshtein or Rock'n Roll, described in the following sections) is used - see Figure 12-24, Figure 12-25. The initial distribution of adhesion forces, F_a, (called F_a-distribution) for the deposited particles is usually assumed to follow the lognormal distribution - Figure 12-21. After the first velocity step only the strongly bound (large F_a) particles remain - Figure 12-26. The same particles remain on the surface after the second step - Figure 12-27.

Because of that reason the parametric model is not recommended for general application. Only the mechanistic model, where the deposited particles are balanced for each F_a-section separately, is capable of calculating the physically correct behavior.

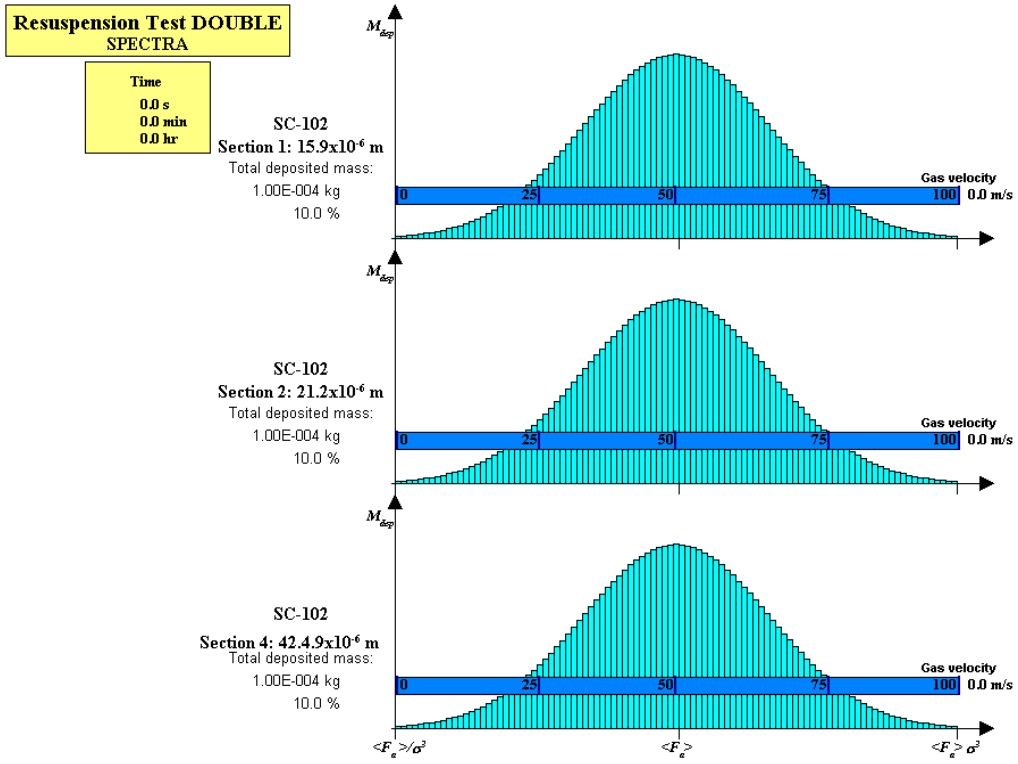


Figure 12-21 Resuspension test DOUBLE, F_a-distributions, t = 0.0 s, Vainshten.

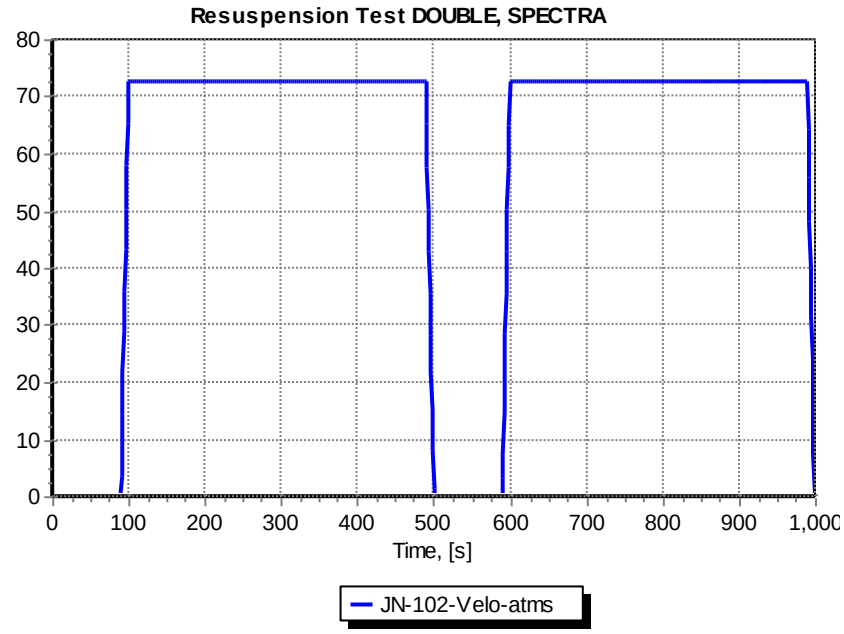


Figure 12-22 Resuspension test DOUBLE, gas velocities.

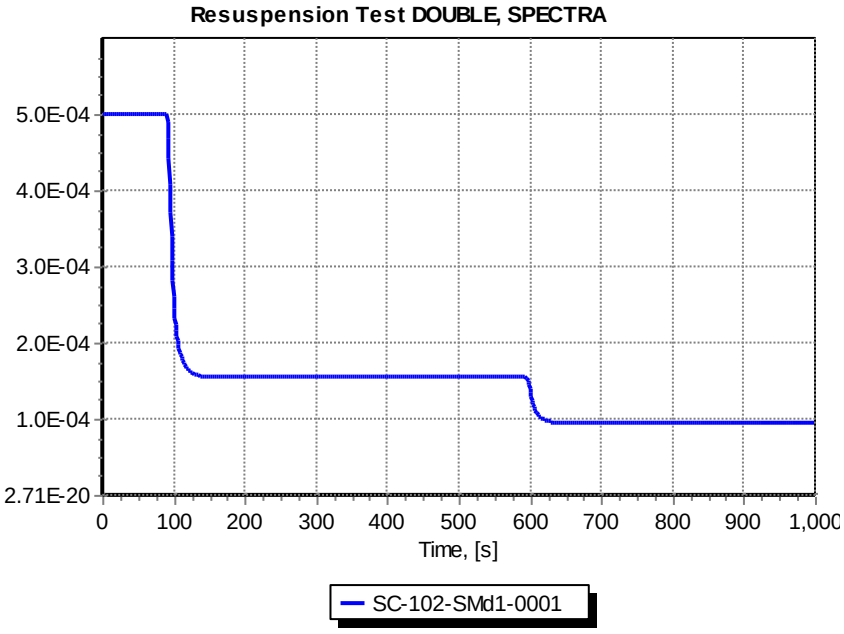


Figure 12-23 Resuspension test DOUBLE, deposited mass, parametric model.

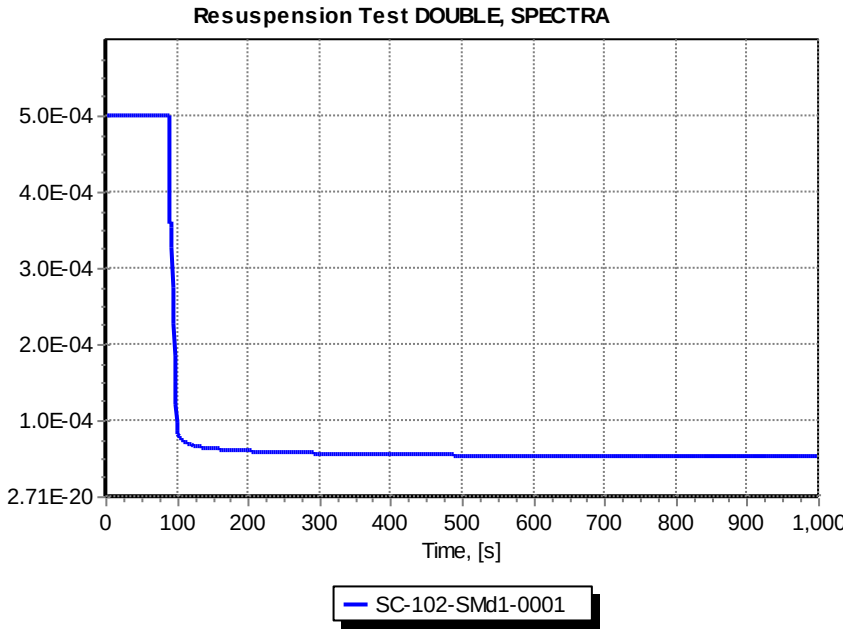


Figure 12-24 Resuspension test DOUBLE, deposited mass, Vainshtein model.

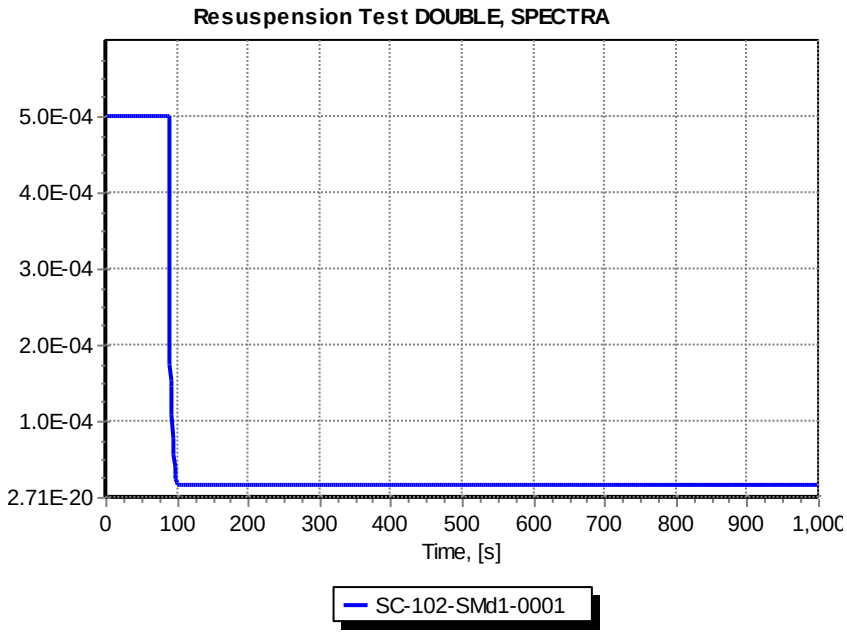


Figure 12-25 Resuspension test DOUBLE, deposited mass, Rock'n Roll model.

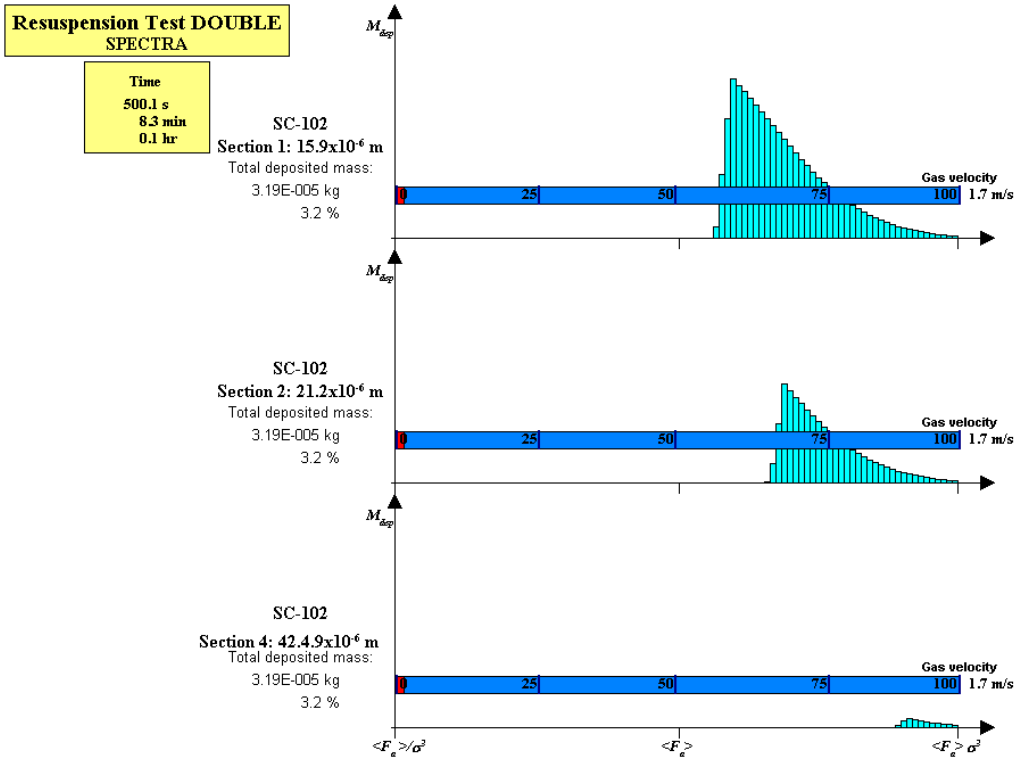


Figure 12-26 Resuspension test DOUBLE, F_a-distributions, t = 500.0 s, Vainshtein.

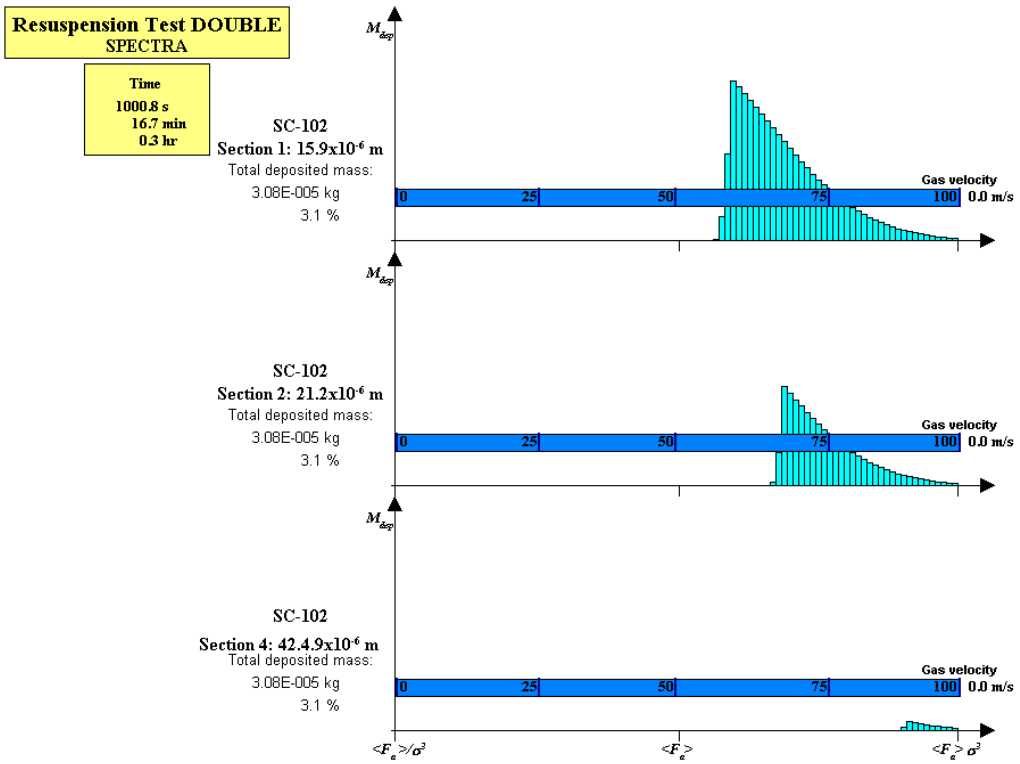


Figure 12-27 Resuspension test DOUBLE, F_a-distributions, t = 1000.0 s, Vainshtein.

12.2.8 Aerosol Resuspension - Mechanistic Model

Two resuspension models are available in SPECTRA: a parametric resuspension model and a mechanistic model. The parametric model is described in the previous section. The mechanistic model is described in this section.

If the mechanistic resuspension model is used, then a change of the deposited mass due to resuspension (the resuspension term *R*, in the aerosol balance equation in section 12.2.4) is given by:

$$R = \left.\frac{dm}{dt}\right|_{resusp} = -R_m \cdot m$$

R_m is the resuspension rate per unit time (1/s), calculated from the mechanistic resuspension model, described below.

Resuspension rate, R_m

The mechanistic resuspension model is based on the work of Vainshtein et al. [108], and Reeks et al. [109] and a review of models performed at NRG. The theory is based on the assumption that a particle is detached from a surface when it has accumulated enough potential energy to escape from the potential energy well. Such considerations lead to a formula for the resuspension rate:

$$R_m = f_0 \exp\left[-\frac{Q}{2\langle PE\rangle}\right]$$

f_0 is the typical frequency of vibration (1/s), Q is the height of the surface adhesion potential well, $\langle PE\rangle$ is the average potential energy of a particle in the well.

In the model of Vainshtein et al. a formula for the resuspension rate is obtained for surfaces where there is a spread of the adhesive forces due to surface roughness. In the model introduced in SPECTRA the continuous spread of the adhesion force is divided into a discrete number of sections, with constant adhesion force in each section. The SPECTRA model keeps track of the particles separately for each section of the adhesion force. During resuspension the adhesion force distribution changes, as the particles with low adhesion forces are resuspended easier (and therefore faster) than the particles with high adhesion forces.

The concept of the adhesion force distribution changes is further discussed below, and illustrated in Figure 12-31 through Figure 12-34. Influence of the distribution changes is shown and the results obtained by keeping track of particles in each section (default model) are compared to the results obtained with fixed adhesion force distribution (disabled particle tracking in the adhesion force sections).

Reference [108] shows that the potential energy, $\langle PE\rangle$, may be expressed in terms of the drag force, F_d, and the height of the potential well, Q, may be found as a function of the tangential pull-off force, $F_{a\tau}$, which in turn is determined by the adhesion force, F_a, as will be shown below.

The resuspension rate is given by [108]:

$$R_m = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_d}\right)^{4/3}\right]$$

In SPECTRA this formula is written as:

$$R_m = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_d}\right)^{x_F}\right]$$

where x_F is a user-defined parameter with the default value of 1.33. Calculations of the frequency of vibration, f_0, the tangential pull-off force, $F_{a\tau}$, the drag force, F_d, as well as the adhesion force, F_a (needed for $F_{a\tau}$ calculation), are described below.

For a given particle size and gas velocity, the drag force, F_d, is just a single number, however the adhesion force, F_a, and thus $F_{a\tau}$ spreads over a range of typically several orders of magnitude. This range is divided in SPECTRA into a finite number of sections, called the adhesion force sections, or shortly the F_a-sections. The above formula is applied individually for every F_a-section.

Typical Frequency of Vibration, f_0.

The typical frequency is given by [108]:

$$f_0 = \frac{\rho_g u_\tau^2}{300\,\mu_g}$$

ρ_g density of gas, (kg/m^3)
μ_g viscosity of gas, (kg/m/s)
u_τ friction velocity, (m/s), equal to:

$$u_\tau = \sqrt{f/8}\cdot V_g$$

f friction factor, (-)
V_g gas velocity, (m/s)

In SPECTRA the formula for f_0 is written as:

$$f_0 = C_{f0}\,\frac{f}{8}\,\frac{\rho_g V_g^2}{\mu_g}$$

where C_{f0} is a user-defined parameter with the default value of 1/300 = 3.33×10^{-3}. An option is available to make f_0 a constant value. In order to achieve that, a negative number should be entered for C_{f0}. In such case f_0 will be equal to the absolute value of C_{f0}:

$$f_0 = |C_{f0}| \qquad if\ C_{f0} < 0.0$$

Tangential Pull-off Force, F_{at}.

The relation between the tangential pull-off force, F_{at}, and the adhesion force, F_a, is derived based on the particle oscillation model described in reference [108]. The particle oscillations consider linear oscillations only using an idealized model of a particle on a spring. The spring stiffness, χ, is given by the Johnson, Kendall, and Roberts (JKR) model [110], as described in [109]:

$$\chi = \frac{9}{2} K^{2/3} r^{1/3} P_1^{1/3} \frac{P_1 + P_0}{5P_1 + P_0}$$

where r is the particle (or asperity) radius, while K is given by:

$$K = \frac{4}{3}\left(\frac{1-\nu_p^2}{E_p} + \frac{1-\nu_s^2}{E_s} \right)^{-1}$$

ν_i and E_i are Poisson's ratio and Young modulus respectively. The subscripts p and s stand for the particle material and the surface material respectively. The values may be found in material handbooks, such as [32]; the typical values are: $\nu_i \sim 0.3$, and $E_i \sim 10^{11}$ Pa (these are the default values - see Volume 2).

The forces P_0 and P_1 are defined in reference [109] (equation 55 and 56). At the point of particle detachment (maximum "spring" length):

$$P_0 = -F_a$$
$$P_1 = F_a$$

At the equilibrium point (minimum "spring" length):

$$P_0 = 0$$
$$P_1 = 4F_a$$

The resulting spring stiffness is equal to:

$$\chi = \begin{cases} 0.0 & \textit{at the particle detachment} \\ \dfrac{9}{10} K^{2/3} r^{1/3} (4F_a)^{1/3} & \textit{at the equilibrium point} \end{cases}$$

In reference [108], the value at the equilibrium point equilibrium point is used:

$$\chi_e = \frac{9}{10} K^{2/3} r^{1/3} (4F_a)^{1/3}$$

This equation is written in terms of the particle (or asperity) diameter, D, instead of the radius:

$$\chi_e = \frac{9 \cdot 4^{1/3}}{10 \cdot 2^{1/3}} \cdot K^{2/3} D^{1/3} F_a^{1/3} = 1.13 \cdot K^{2/3} D^{1/3} F_a^{1/3}$$

In SPECTRA this formula is written as:

$$\chi = C_\chi \cdot K^{2/3} \cdot D_{eff,0}^{1/3} \cdot F_a^{1/3}$$

where C_χ is a user-defined parameter with the default value of 1.13. The $D_{eff,0}$ represents effective particle diameter, which is proportional to the particle true diameter, D_p, if D_p « r_{as}, and is proportional to $2r_{as}$ if r_{as} « D_p (see the description of effective surface curvature, below). The $D_{eff,0}$ is calculated for each particle size section and each F_a-section from:

$$D_{eff,0} = \frac{1}{\frac{1}{x_{p,0} D_p} + \frac{1}{x_{as,0} 2r_{as}}}$$

$x_{p,0}$ and $x_{as,0}$ are user-defined multipliers, D_p is the particle diameter, (m), while r_{as} is the asperity radius, (m). The asperity radius, r_{as}, may be different for each F_a-section. Several options are available to define this parameter. Detailed descriptions of these options are provided below.

The approach applied in [109] is to use the maximum "spring" stiffness. This approach leads to the value of C_χ equal to 1.13. Other values may be applied; for example average spring stiffness may be used, which is achieved by setting C_χ to the value of about 0.56. The influence of the parameter C_χ on the calculated resuspension rates is shown in Volume 3.

The tangential pull-off force, $F_{a\tau}$, is equal to ([108], equation 11):

$$F_{a\tau} = \frac{2\chi}{D^2} x_B^3$$

with ([108], equation 10):

$$x_B = \sqrt{D y_B}$$

and ([108], equation 5):

$$y_B = \frac{F_a}{\chi}$$

Upon substitution the formula for the tangential pull-off force becomes:

$$F_{a\tau} = \frac{2\chi}{D^2} \cdot D^{3/2} \cdot \left(\frac{F_a}{\chi} \right)^{3/2} = 2 \cdot \frac{F_a^{3/2}}{D^{1/2} \cdot \chi^{1/2}}$$

In SPECTRA this formula is written as:

$$F_{a\tau} = C_{Fa} \cdot \frac{F_a^{3/2}}{D_{eff,0}^{1/2} \cdot \chi^{1/2}}$$

C_{Fa} is a user-defined parameter with the default value of 2.0. F_a, is the adhesion force, described below.

An option is available to use a scaled adhesion force, F_a, with the scaling factor equal to the absolute value of C_{Fa}:

$$F_{a\tau} = |C_{Fa}| \cdot F_a \qquad if\ C_{Fa} < 0.0$$

If the Vainshtein theory is applied to calculate the tangential pull-off force (i.e. $C_{Fa} > 0.0$) the ratio of the tangential force, $F_{a\tau}$, and the adhesion force, F_a, depends on the parameters such as C_{Fa}, $x_{p,0}$, $x_{as,0}$, and the option applied for r_{as} calculation. These are user-defined coefficients (CFARRT, XP0RRT, XA0RRT - see Volume 2, records 8700XX, 8791XX). Some combination of these parameters may give unrealistic ratio of $F_{a\tau}/F_a$ forces. In order to avoid extremely unrealistic values the limit is provided for the ratio of forces. The limit is defined by input parameters TMNRRT, TMXRRT (Volume 2, record 8791XX). The default values are 10^{-3} and 10^{-1}, therefore:

$$10^{-3} \le \left(\frac{F_{a\tau}}{F_a} \right) \le 1$$

Adhesion Force, F_a, Distribution

The principal assumption of the Vainshtein et al. resuspension model is existence of a spread of the adhesive forces due to surface roughness [108]. This spread is typically assumed for have the lognormal distribution (see for example reference [75], equation 4.42, [88], equation 11):

$$\varphi_a(F_a) = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{F_a \ln(\sigma_a)} \cdot \exp\left(-\frac{1}{2} \cdot \left[\frac{\ln(F_a / \langle F_a \rangle}{\ln(\sigma_a)} \right]^2 \right)$$

where $<F_a>$ is the mean value of the adhesive force and σ_a is the standard deviation of the log-normal distribution; in this case it is a measure of the spread of the adhesive forces, often referred to as the adhesive spread factor (see [88], page 6). Physically the value of $\varphi(F_a)$ gives the fraction of deposited particles that are attached to the surface with the adhesion force equal to F_a, per unit adhesion force, (1/N).

Examples of the lognormal distribution are shown in Figure 12-29. The figure shows three distributions. In each case the mean value is 1.0. The adhesive spread factors are equal to 1.2, 2.0, and 4.5. Note that with increasing adhesive spread factor the peak value is shifted towards smaller values. The mean value is nevertheless always the same. This is because the lines represent the fraction per unit adhesion force and in the logarithmic scale the unit force on the left-hand side is smaller than on the right-hand side.

The adhesive spread, σ_a, is defined by the user, and it may be different for every particle size section (default value of σ_a is 4.0). The mean value of the adhesive force, $<F_a>$, may be defined by the user, or alternatively calculated by the program from the formulae shown below. As the adhesive spread, the mean value is defined (or calculated) for every particle size section. Thus the adhesive mean forces and the adhesive spreads may be different for different particle sizes.

In SPECTRA the spread of the adhesion force is represented by dividing the adhesion force "space" into a discrete number of intervals, called the "adhesion force sections", or shortly the "F_a-sections". The total number of the F_a-sections is a user-defined parameter (see Volume 2).

The minimum and maximum values for the "F_a-sections" are determined internally by the code using the following reasoning. It is known that in case of the normal distribution practically all (>99%) of the function is located in the interval between $\langle F_a \rangle - 3\sigma_a$ and $\langle F_a \rangle + 3\sigma_a$. In case of lognormal the interval boundaries are:

$$A = \frac{\langle F_a \rangle}{\sigma_a^3}$$

$$B = \langle F_a \rangle \cdot \sigma_a^3$$

Mathematically it means with A and B selected as above, the following relation is fulfilled with a very good accuracy (error smaller that 1%):

$$\int_A^B \varphi_a(F_a, \langle F_a \rangle, \sigma_a) \cdot dF_a = 1.0$$

Therefore the values of A and B are used as the boundary values for the lognormal distribution. The F_a-sections are uniformly distributed in the logarithmic space, which means the section boundaries, $F_{a,\,sec}$, are given by:

$$F_{a,\mathrm{sec}}(i+1) = F_{a,\mathrm{sec}}(i) \cdot D_{\mathrm{sec}}$$

$F_{a,sec}(i)$ is the lower boundary value of the F_a-section number i, $F_{a,sec}(i+1)$ is the upper boundary value of the F_a-section number i (equal to the lower boundary value of the F_a-section number $i+1$), and D_{sec} is the F_a-section width in the logarithmic space, equal to:

$$D_{\mathrm{sec}} = \exp\left(\frac{\ln(B/A)}{N_{\mathrm{sec}}}\right)$$

N_{sec} is the total number of the F_a-sections. The fraction of particles in the F_a-section i, in other words the fraction of deposited particles of given size section that is attached to the surface by the adhesion force between $F_{a,sec}(i)$ and $F_{a,sec}(i+1)$, is obtained from:

$$\overline{\varphi}_a(i) = \int_{F_{a,\mathrm{sec}}(i)}^{F_{a,\mathrm{sec}}(i+1)} \varphi_a(F_a, \langle F_a \rangle, \sigma_a) \cdot dF_a$$

In SPECTRA the integral is evaluated numerically by dividing each sections into a finite number of intervals and performing a standard numerical integration (trapezoid method):

$$\int_{F_{a,\mathrm{sec}}(i)}^{F_{a,\mathrm{sec}}(i+1)} \varphi_a(F_a, \langle F_a \rangle, \sigma_a) \cdot dF_a = \sum_{j=1}^{Nj} \left(\frac{\varphi_{a,j}(j) + \varphi_{a,j}(j+1)}{2} \cdot [F_{a,j}(j+1) - F_{a,j}(j)] \right)$$

$F_{a,j}(j)$ and $F_{a,j}(j+1)$ are the integration points, $\varphi_{a,j}(j)$ and $\varphi_{a,j}(j+1)$ are the values of the distribution function φ in the points $F_{a,j}(j)$ and $F_{a,j}(j+1)$ respectively, and N_j is the number of integration intervals per F_a-section. N_j is a user-defined parameter (see Volume 2) with a default value of 1. The integration points are uniformly distributed in the logarithmic space, which means that:

$$F_{a,j}(j+1) = F_{a,j}(j) \cdot D_j$$

D_j is the integration interval in the logarithmic space, equal to:

$$D_j = \exp\left(\frac{\ln(B/A)}{N_{\text{sec}} \cdot N_j}\right)$$

The integration is illustrated in Figure 12-28. In the example shown in this figure the number of F_a-sections, N_{sec}, is equal to 6, the number of integration intervals per F_a-section, N_j, is equal to 2. Consequently the total number of integration intervals is $N_s \times N_j = 12$. For each F_a-section the representative adhesion force, $F_a(i)$, that is used to calculate the resuspension rate for this particular section is equal to the middle value in the logarithmic scale:

$$F_a(i) = \exp\left(\frac{\ln(F_{a,\text{sec}}(i) \cdot F_{a,\text{sec}}(i+1))}{N_{\text{sec}} \cdot N_j}\right) = \sqrt{F_{a,\text{sec}}(i) \cdot F_{a,\text{sec}}(i+1)}$$

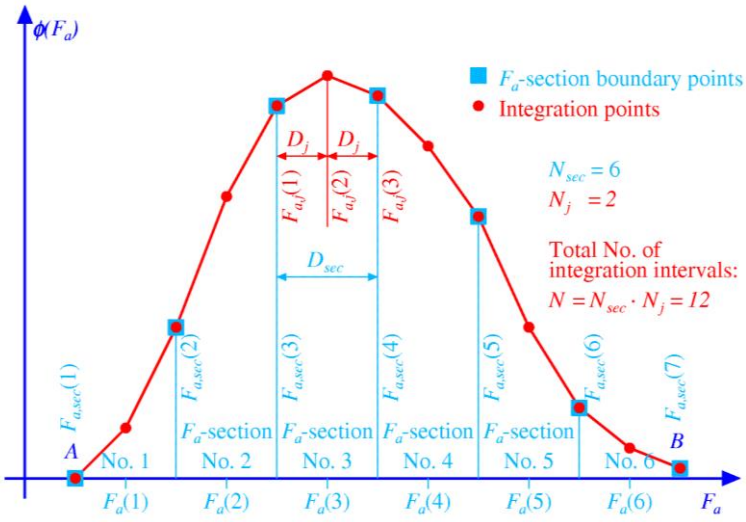


Figure 12-28 F_a-section boundaries and integration points.

It was found out in test calculations that for practically interesting values of adhesive spread $\sigma_a < 5$ (note that with $\sigma_a = 5$ the spread of adhesive forces is more than four orders of magnitude) quite accurate results are obtained using the total of 25 - 50 integration points.

- Intervals: 25 (N_s = 25, N_j=1), σ_a = 1.2, Integral = 0.997
  σ_a = 3.0, Integral = 1.009
  σ_a = 5.0, Integral = 1.019

- Intervals: 51 (N_s = 51, N_j=1), σ_a = 1.2, Integral = 0.997
  σ_a = 3.0, Integral = 1.000
  σ_a = 5.0, Integral = 1.003

The default value of F_a-sections is 51 and the number of integration intervals per F_a-section is 1. An advantage of having an uneven number of the F_a-sections is having a middle section with the adhesion force F_a exactly equal to the mean value $<F_a>$. With 51 F_a-sections the total integral is calculated with the error below about 0.3%. Considering the uncertainties involved in the resuspension model the accuracy of 0.3% in evaluating the integral is far more than sufficient for the practical purposes. The results of integration are normalized in SPECTRA, so that in the end the total sum is always exactly equal to 1.0. The maximum number of F_a-sections is 99. An influence of number of F_a sections on the results is discussed in Volume 3.

If the adhesive spread, σ_a, is large, a large number of F_a-sections is necessary to correctly represent section-by section resuspension. For example, with σ_a = 10, the min./max. range is 10^{-3} / 10^{3}. Thus the adhesive force spread is 6 orders of magnitude. Therefore a maximum number of 51 F_a-sections is selected by default only if $\sigma_a \leq 10$, while it is equal to 99 if $\sigma_a > 10$.

An example of the fractions calculated by SPECTRA for the lognormal distribution, $\overline{\varphi}$, is shown in Figure 12-30. The results are presented for the mean value of 1.0 and the adhesive spread factors of 1.2, 2.0, and 4.5. Note that the distribution expressed in particle fractions (relative number of particles) per single F_a-section is symmetrical around the mean value, independently of the adhesive spread σ_a.

Apart from the lognormal distribution the user may specify the adhesive force distribution using a tabular function. With this option any distribution may be defined. The adhesive force distribution needs to be obtained from measured data. If available measurements are not very accurate, the lognormal distribution is recommended, as the one that requires less user effort. If more detailed data is available, the user may wish to use the tabular distribution option in order to provide a closer match to the measured values.

The user input required for each option is summarized below.

- Input parameters for the lognormal distribution
  - N_{sec} and N_j for each particle size section
  - Values of $<F_a>$ (or methods of calculating them) for each F_a-section and each particle size section. Values of the adhesive spread, σ_a, each particle size section.

- Input parameters for the tabular distribution
  - N_{sec} and N_j for each particle size section. The total number of integration intervals is equal to $N_{sec} \times N_j$. The number of integration points is o $N_{sec} \times N_j + 1$.
  - Values of $<F_a>$ and σ_a, for each F_a-section and each particle size section. The grid points for distribution definition are defined as in case of lognormal distribution:

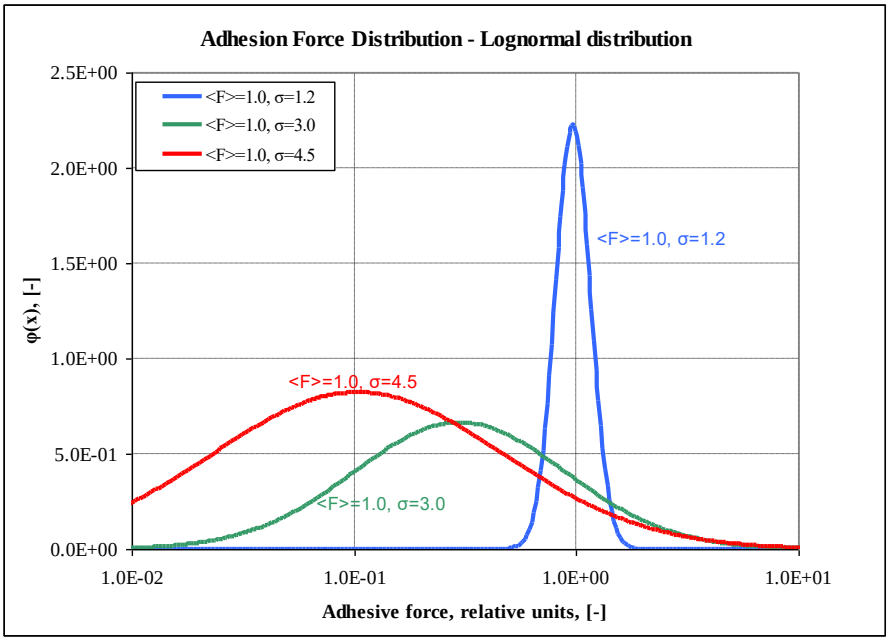


Figure 12-29 Fraction of particles per unit adhesion force, lognormal distribution.

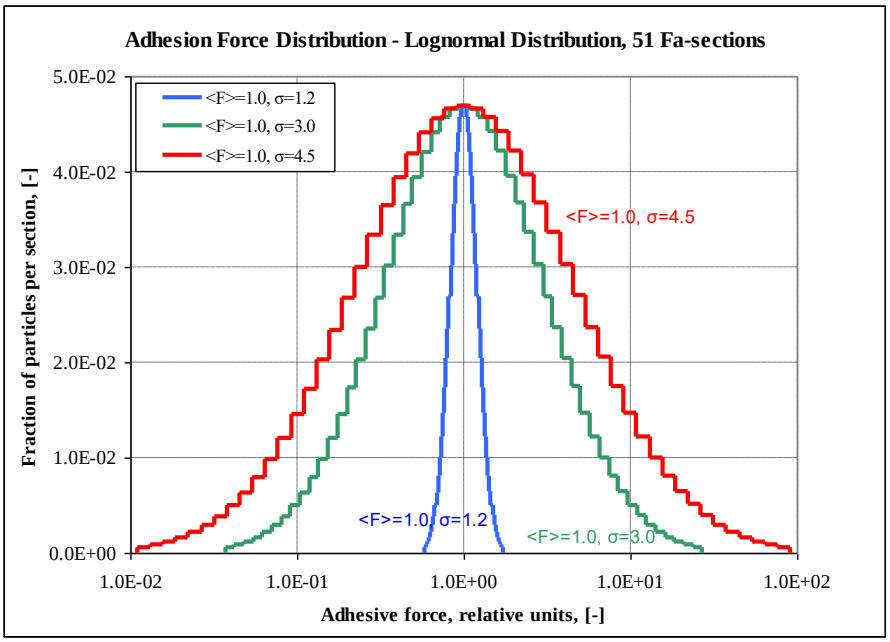


Figure 12-30 Fraction of particles per F_a-section, lognormal distribution.

$$F_{a,j}(1) = A = \frac{\langle F_a \rangle}{\sigma_a^3} \qquad F_{a,j}(N) = B = \langle F_a \rangle \cdot \sigma_a^3$$

$$F_{a,j}(j+1) = F_{a,j}(j) \cdot \exp\left(\frac{\ln(B/A)}{N_{\text{sec}} \cdot N_j} \right)$$

- Array defining the distribution function per unit adhesion force, φ_j, (1/N), for each integration point (see Volume 2). The particle fractions per F_a-section are calculated from the entered distribution function in exactly the same way as in case of the lognormal distribution, including normalization of the final values. Since the distribution is normalized, the entered values of φ_j may be scaled by any factor.

It is important to realize that with this option the user specifies the adhesion force distribution (by defining $<F_a>$ and σ_a) it is the tangential pull-off force, $F_{a\tau}$, which determines the resuspension rate. The $F_{a\tau}$ is related to F_a through the effective diameter:

$$D_{eff,0} = \frac{1}{\dfrac{1}{x_{p,0} D_p} + \dfrac{1}{x_{as,0} 2r_{as}}}$$

$x_{p,0}$ and $x_{as,0}$ are input parameters, while the asperity radius, r_{as}, may be calculated as follows::

- A single value of r_{as} may be used to calculate $D_{eff,0}$ for all F_a-sections, $r_{as}(i) = <r_{as}>$. The value of $<r_{as}>$ is defined by the input parameter RASRRT (see Volume 2, records 8791XX). Default value of RASRRT is 1.0×10^{-7} m, which is equal to $0.1\times D_p$ [108], for the particle diameter of $D_p=10^{-6}$ m.
- Distribution of r_{as} may be defined by defining the mean value, $<r_{as}>$ (input parameter RASRRT), and the spread factor σ_{as} (input parameter SASRRT - see Volume 2, records 8791XX). Default values of RASRRT and SASRRT are 1.0×10^{-7} m and 4.0 respectively. This is the default option (see Volume 2, description of records 8791XX). With this option the ratio of $F_{a\tau}$ to F_a is close to the value obtained from the Vainshtein theory. Identical results are obtained when single particle size, or when a single value for the mean asperity size $<r_{as}>$ is defined in the "Vainshtein option" (when the asperity size distributions, $\varphi_{as}(r_{as})$, are defined - see description below.

Two options are available in the program for the particle distribution calculation in each F_a-section:

- *Variable distribution*
  When this option is used the number of deposited particles is balanced separately for each F_a-section. It is important to do that because for a given (fixed) drag force, the particles that are weakly attached (low F_a-sections) will be resuspended quickly, while the particles that are strongly attached (high F_a-sections) will practically not be resuspended. Therefore the resuspension will soon stop and will only be resumed if the drag force (fluid velocity) is increased.

  With this option the user-defined distribution, $\varphi_a(F_a)$, is used only for deposition, to distribute the newly deposited particles in the appropriate F_a-sections. During resuspension a balance of particles is considered for each F_a-section, so the resuspension from a given

F_a-section stops when there are no particles left in this section. This is the default option in SPECTRA. When this option is used, the mass balance is written for each F_a-section:

$$\frac{dm(i)}{dt} = -R_m(i)\cdot m(i) + \bar{\varphi}_a(i)\cdot D$$

This equation is written using a finite difference scheme:

$$m(i) = m_0(i)\cdot\left(1 + R(i)\cdot\Delta t\right) + \bar{\varphi}_a(i)\cdot D\cdot\Delta t$$

$m(i)$	-	deposited mass, (kg), F_a-section number i, new time step value
$m_0(i)$	-	deposited mass, (kg), F_a-section number i, old time step value
$R_m(i)$	-	resuspension rate, (1/s), F_a-section number i
Δt	-	time step, (s)
D	-	deposition rate, (kg/s)

The particle mass in the section i, $m(i)$, is equal to the total deposited mass, m, multiplied by the current distribution function $\varphi(i)$. The above equation is rearranged to calculate the distribution function:

$$\bar{\varphi}(i) = \frac{m_0\cdot\bar{\varphi}_0(i)\cdot\left(1 + R(i)\cdot\Delta t\right) + \bar{\varphi}_a(i)\cdot D\cdot\Delta t}{m}$$

In the above formula $\varphi(i)$ is the current distribution, i.e. fraction of the particles in the F_a-section number i, $\varphi_0(i)$ is the previous time step value, m and m_0 are the total deposited masses (all F_a-sections) at the new and the old time step.

- *Fixed distribution*
  When this option is used the user-defined adhesion force distribution is kept constant throughout the transient. This means in practice that the particles are not tracked for each F_a-section individually. In other words, the code "does not know" which particles have already been resuspended and assumes that the particle distribution is the same and equal to the user-defined function, $\varphi_D(i)$, at every time step:

$$\bar{\varphi}(i) = \bar{\varphi}_a(i)$$

  This option is not recommended for general application. It is included mainly for test calculations, and comparisons with some analytical solutions.

Influence of balancing separately all size sections is illustrated in Figure 12-31 through Figure 12-34. In this test gas velocity linearly increases from V_g = 0.0 m/s at time t=0.0, to V_g ~ 88 m/s at time t = 50.0 The figures show calculations performed with the default option (Option 1) and keeping fixed distribution (Option 2). With the first option, only the particles from the "left" F_a-sections (i.e. the particles which are weakly attached to the surface) are resuspended. Once these weakly attached particles are removed from the surface, there is practically no more resuspension until the gas velocity and thus the drag force increases.

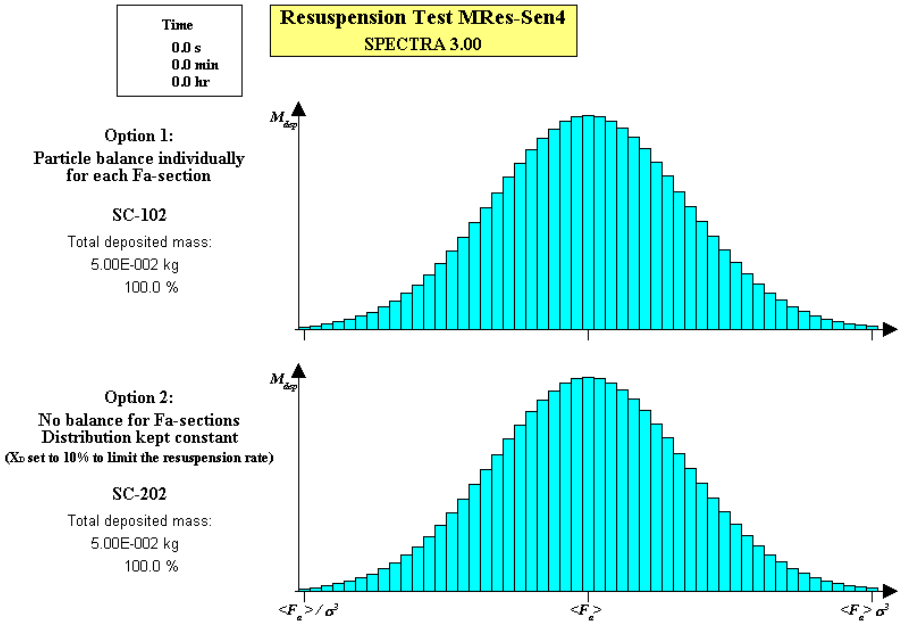


Figure 12-31 Resuspension test MRes4, influence of balancing each F_a-section, t = 0.0 s.

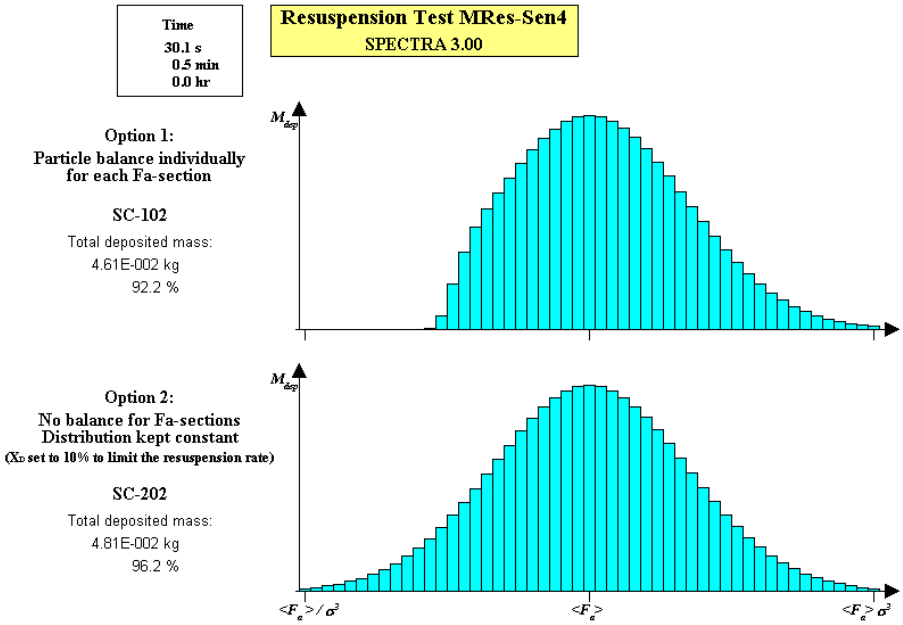


Figure 12-32 Resuspension test MRes4, influence of balancing each F_a-section, t = 30.0 s.

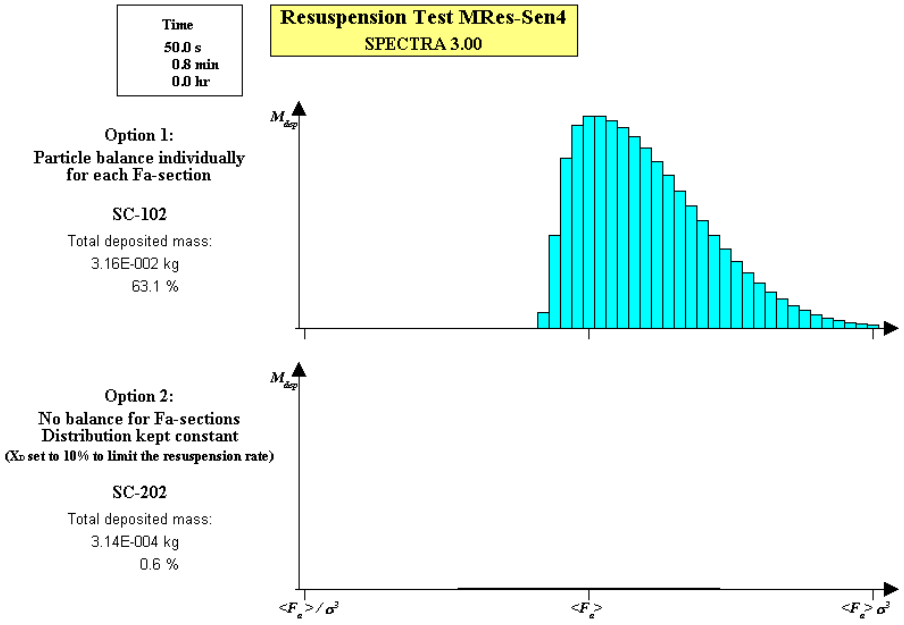


Figure 12-33 Resuspension test MRes4, influence of balancing each F_a-section, t = 50.0 s.

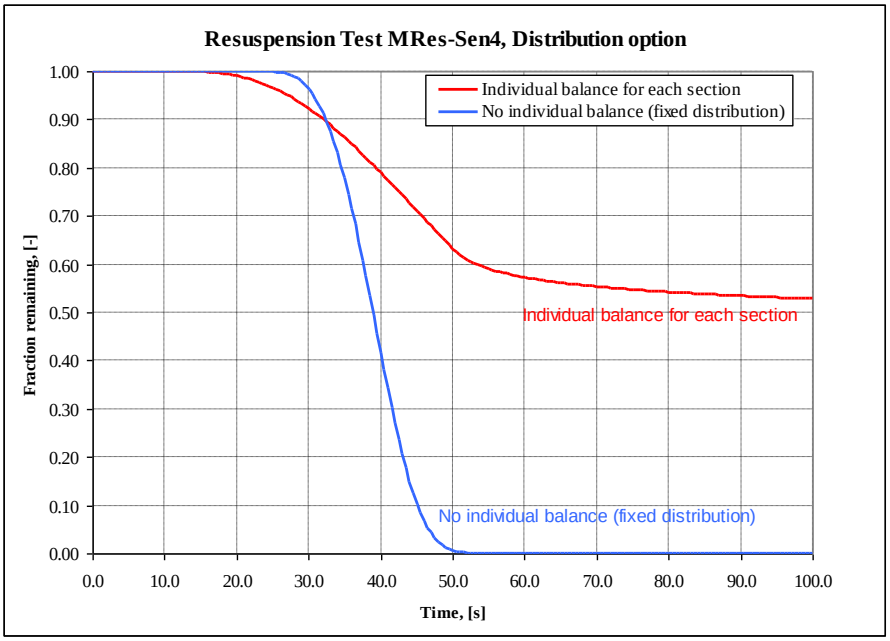


Figure 12-34 Resuspension test MRes4, influence of balancing each F_a-section.

On the other hand, if the Option 2 is selected, particle resuspension is much faster. Physically this option means that particles from all sections are resuspended, with the resuspension rate equal to the average for all F_a-sections.

The resuspension rate for the weakly bound ("left") particles is large, while for the strongly bound ("right") particles is practically zero. Nevertheless the average value is still quite significant and the particles are resuspended very fast. It should be noted that for the Option 2 the drag force multiplier, X_d, has been reduced by a factor of 10. Default value is 0.3 (see description of the drag force calculation, F_d, below). For the Option 2 X_d has been set to 0.03. Even with 10 times smaller drag force, the deposited mass is only 3×10^{-4} kg (0.6%) at 50 s, while with Option 1 it is about 32 g (63%) at the same time (see Figure 12-34). The resuspension rate is calculated for each F_a-section from the following formula:

$$R_m(i) = f_0 \exp\left[-\left(\frac{F_{a\tau}(i)}{F_d(i)} \right)^{x_F} \right]$$

This may be written as:

$$R_m(i) = f_0 \exp\left[-Y(i) \right]$$

where $Y(i)$ is an exponent, characteristic for the given F_a-section. In SPECTRA a limit is provided for the maximum value of $Y(i)$. This limit (an input parameter CUTRRT, see Volume 2, record 8700XX) is an important parameter, therefore a short discussion is provided here. Applying the limit, means that the resuspension rate is calculated from:

$$R_m(i) = \begin{cases} f_0 \exp\left[-Y(i) \right] & if \quad Y(i) < CUTRRT \\ 0.0 & if \quad Y(i) > CUTRRT \end{cases}$$

The cut-off values are between 5 and 20, with a default value of 10 (see Volume 2). Therefore the cut-off limits are:

- Minimum: CUTTRT = 10.0, $\exp(-Y(i)) > 4.5\times10^{-5}$ (-)
- Default: CUTTRT = 15.0, $\exp(-Y(i)) > 3.1\times10^{-7}$ (-)
- Maximum: CUTTRT = 20.0, $\exp(-Y(i)) > 2.1\times10^{-9}$ (-)

With CUTOFF = 20.0 the resuspension rates are set to zero only for those sections for which it is very low anyway ~10^{-9}. This means that there is practically no cut-off, and the results are representing the model of Vainshtein et al. This is shown as the first line in Figure 12-35 (this figure shows resuspension results for 3 size sections, 0.4, 1.5, 4.0 μm, for a constant gas velocity - detailed description of the test condition, and comparison with an analytical solution of Vainshtein model is shown in Volume 3).

With the cut-off limit of 20 most resuspension (~35%) occurs during the first second. During the slow resuspension that follows (1000 s) another ~10 % is resuspended. During this and later times the resuspension is governed by the size sections for which $\exp(-Y(i)) \sim 10^{-5}$ (resuspension time of order of 10^5 s. While in theory this resuspension rates are possible, the experimental evidence [112] shows that resuspension is typically much more rapid and occurs in a matter of seconds rather than thousands of seconds. Therefore it is surmised that applying the Vainshtein formula to the sections for which $\exp(-Y(i)) \sim 10^{-7}$ is simply applying the model beyond it's range of validity.

Therefore CUTRRT of 15 is applied as a default value. This cut-off value will remove all the lower rate resuspension terms. This means in practice that the resuspension will stop after the time of order of 100 s (Figure 12-35, second line).

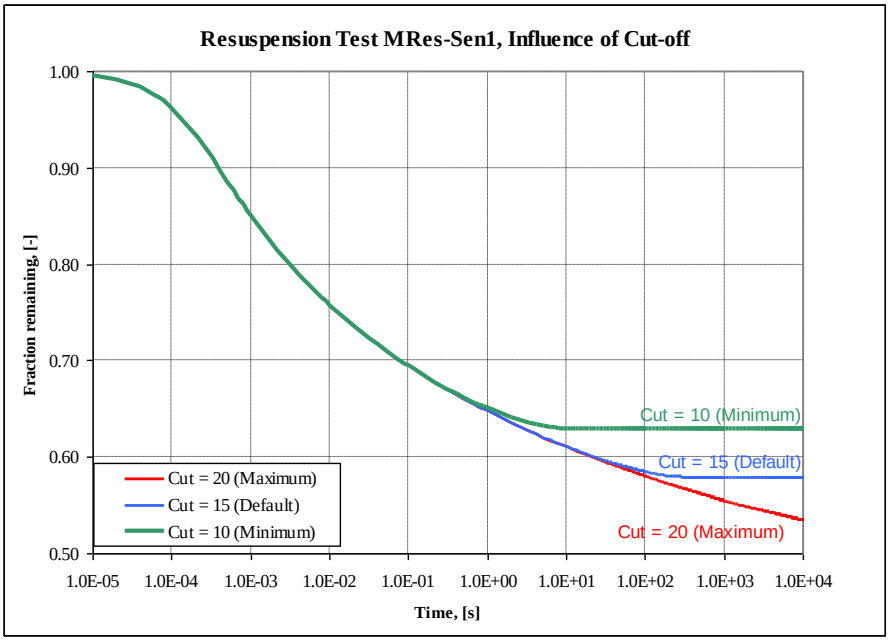


Figure 12-35 Influence of the cut-off limit on resuspension rates.

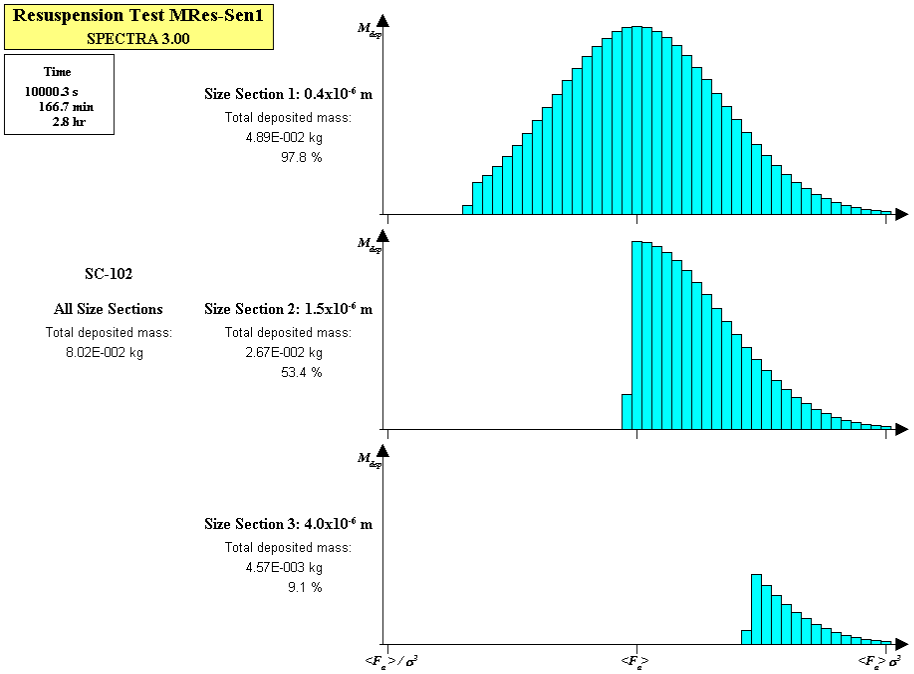


Figure 12-36 Influence of the cut-off limit - CUTRRT = 20 (maximum).

The minimum cut-off value is 10.0. This value will result in stopping the resuspension already after about 10 s (Figure 12-35, third line). The end-results (at 1000 s) for all three cases are shown in Figure 12-36 through Figure 12-38. Further discussion is provided in Volume 3.

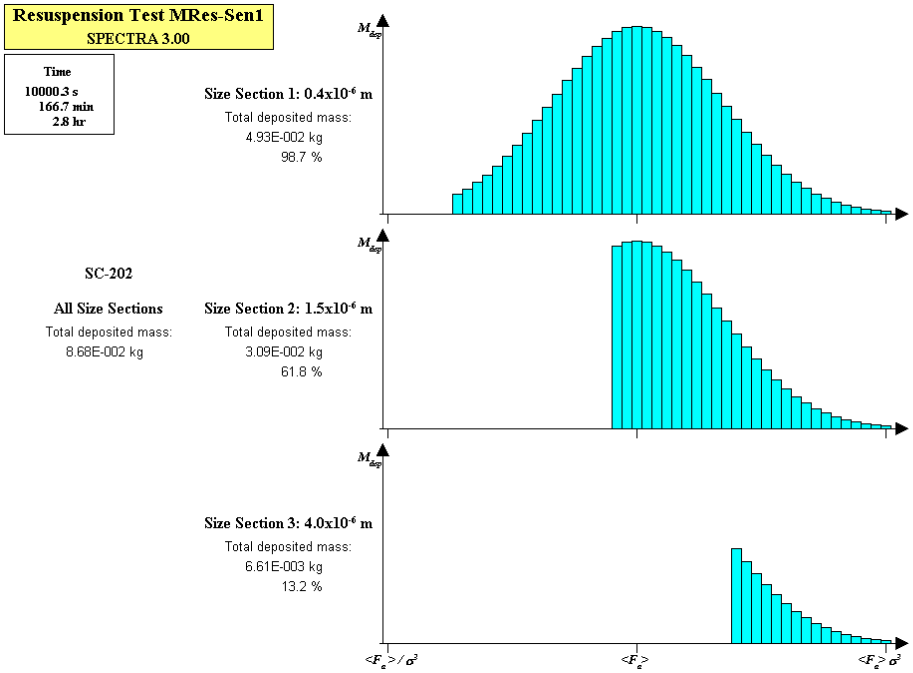


Figure 12-37 Influence of the cut-off limit - CUTRRT=15 (default).

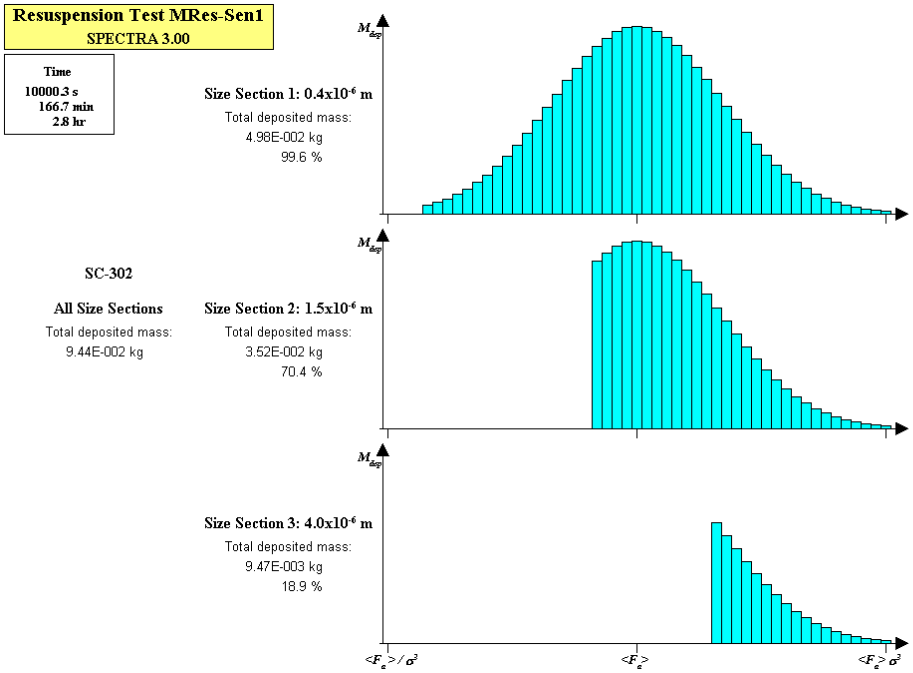


Figure 12-38 Influence of the cut-off limit - CUTRRT = 10 (minimum).

Calculation of the Adhesion Force Distribution from the Asperity Radius Distribution

The resuspension option presented so far was based on a direct knowledge of the adhesion force distribution, $\varphi_a(F_a)$, with the mean value $<F_a>$ and the adhesive spread, σ_a, being user-defined parameters. Such distribution is used for example in [88].

Quite often the literature another approach is made. A surface asperity, r_{as}, distribution is assumed, $\varphi_{as}(r_{as})$. The adhesion force is then expressed in as a function of the surface asperity radius, $F_{as}=f(r_{as})$, which in the end allows to obtain the adhesion force distribution itself. Such approach is used for example in references [108] and [109]. This approach seems a little of going around to tackle the problem, because of two reasons.

- Firstly, the adhesion forces are easier to measure (this is done in centrifuges) than the asperity radii,
- Secondly, the use of asperity radii requires certain assumptions (discussed below), which are difficult to justify.

This option is included in SPECTRA because it is applied in the article of Vainshtein [108], which is the principal source of the model applied in SPECTRA. To allow easier comparison with [108] and other literature data the option described below is included.

When the method based on the surface asperity distribution is used, the asperity distribution is defined instead of the adhesion force itself. This is again done using the lognormal distribution or the tabular distribution. The lognormal distribution is in this case written as:

$$\varphi_{as}(r_{as}) = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{r_{as} \ln(\sigma_{as})} \cdot \exp\left(-\frac{1}{2} \cdot \left[\frac{\ln(r_{as} / \langle r_{as} \rangle}{\ln(\sigma_{as})} \right]^2 \right)$$

where the subscript *as* is used for asperities. The values of $<r_{as}>$ and σ_{as} are of course input parameters, which need to be defined instead of the $<F_a>$ and σ_a used when the adhesion force distribution is entered directly. Those parameters are defined individually for each size section. Therefore the asperity size may be specified as different for each particle size. The asperity size sections, r_{as}-sections, are then defined in exactly the same way as the F_a-sections:

$$A = \frac{\langle r_{as} \rangle}{\sigma_{as}^3} \qquad B = \langle F_{as} \rangle \cdot \sigma_{as}^3$$

$$r_{as,\mathrm{sec}}(i+1) = r_{as,\mathrm{sec}}(i) \cdot D_{\mathrm{sec}}$$

$$D_{\mathrm{sec}} = \exp\left(\frac{\ln(B/A)}{N_{\mathrm{sec}}} \right)$$

$$r_{as}(i) = \sqrt{r_{as,\mathrm{sec}}(i) \cdot r_{as,\mathrm{sec}}(i+1)}$$

Next, a relation between the asperity radius and the corresponding adhesion force is needed, in order to convert the r_{as}-sections into the F_a-sections. Typically it is done using the following reasoning (see [108]). Adhesion force is written for a particle of a given diameter, D_p, deposited on a smooth surface. For the basic adhesion forces applicable for typical dust particles the adhesion force is proportional to the particle diameter:

$$F_{a,smooth} \sim D_p$$

In the Vainshtein approach it is stated that for rough surfaces a small surface asperity may play a role of a particle while a particle may play a role of a flat surface [108], in which case:

$$F_a \sim r_{as}$$

A "reduction factor" is then defined:

$$(F_a / F_{a,smooth}) = (r_{as} / R_p) = (2r_{as} / D_p)$$

This relation may be used to scale the smooth values into the rough values. This approach may be criticized for a number of reasons. The most important weak points are:

- Firstly, the particles are typically not smooth but have asperities of their own.
- Secondly, the assumptions imply that $r_{as} \ll D_p$, so it is not applicable for different particle sizes and therefore cannot be applied in SPECTRA calculations where the particle size sections may in general vary significantly (see section 12.2.1).
- Thirdly, a deposited particle will be in contact with several (typically 3) asperities, because a particle cannot rest on just one asperity.

The first weakness cannot be easily remedied at present. The next two weakness however can be to some extend remedied. This is done by defining the effective diameter, $D_{eff,i}$:

$$D_{eff,i} = \frac{1}{\frac{1}{x_{p,i} D_p} + \frac{1}{x_{as,i} 2r_{as}}}$$

$x_{p,i}$ and $x_{as,i}$ are user-defined multipliers, which in general may be different for different types of adhesive forces. The subscript i indicates the type of adhesion force, (i=1: van der Waals, i=2: adsorbed liquid). The values of $x_{p,i}$ and $x_{as,i}$ are discussed below. With the effective particle diameter defined as above the adhesion force is proportional to:

$$F_a \sim \begin{cases} r_{as} & if \quad r_{as} << D_p \\ D_p & if \quad r_{as} >> D_p \end{cases}$$

Therefore it has the desired property $F_a \sim r_{as}$ for $r_{as} \ll D_p$, but it has applicability range is extended to $r_{as} \gg D_p$. Moreover the effect of particle diameter may be easily removed by setting the $x_{p,i}$ to a large value. In this way the "classical" Vainshtein model can be obtained for comparisons with literature data (see Volume 3).

Finally, the fact that a particle rests typically on three asperities may be taken into account by setting $x_{as,i}$ to 3.0. If $r_{as} \ll D_p$, then $F_a \sim 3r_{as}$ - the total adhesion force resulting from asperity contact in the three resting points is assumed to be sum of the individual asperity forces.

When the asperity distribution is used and converted to the adhesive force distribution, then the effective diameter, $D_{eff,i}$, and finally the adhesion force $F_a(i)$ are calculated for each F_a-section using the average asperity radius, $r_{as}(i) = (r_{as,sec}(i) \times r_{as,sec}(i+1))^{1/2}$, for this section.

It should be remembered that when the asperity distribution is used and converted to the adhesive force distribution, one should calculate the adhesive force for a smooth surface (the Vainshtein model assumes that asperity plays the role of particle on a smooth surface), therefore the surface roughness should be entered as zero. If this is not the case, SPECTRA will print a warning message in the input diagnostics.

In the Vainshtein "prescription" the asperity radius is equal to:

$$\langle r_{as} \rangle = 0.1 \times (D_p / 2)$$

This is only possible to achieve using the option with the asperity size distribution, $\varphi_{as}(r_{as})$. Such inputs are shown in Volume 3. Here a short example is given, where $<r_{as}>$ is defined according to this prescription for five size sections:

```
*      <Ras>    sigma
870101 0.125E-7  4.0   * size section 1, Dp = 0.25E-06, f'=0.1 => <Ras> = 0.1*0.25E-6/2 = 0.125E-07
870201 0.250E-7  4.0   * size section 2, Dp = 0.50E-06, f'=0.1 => <Ras> = 0.1*0.50E-6/2 = 0.250E-07
870301 0.500E-7  4.0   * size section 3, Dp = 1.00E-06, f'=0.1 => <Ras> = 0.1*1.00E-6/2 = 0.500E-07
870401 1.000E-7  4.0   * size section 4, Dp = 2.00E-06, f'=0.1 => <Ras> = 0.1*2.00E-6/2 = 1.000E-07
870501 2.000E-7  4.0   * size section 5, Dp = 4.00E-06, f'=0.1 => <Ras> = 0.1*4.00E-6/2 = 2.000E-07
```

On the other hand, the input below illustrates the default option ($\varphi_a(F_a)$ distribution is defined), with the mean forces $<F_a>$ left to be calculated by the code (zeroes are entered), and the mean asperity radius equal to 1.0×10^{-7} m:

$$\langle r_{as} \rangle = 1.0 \times 10^{-7}$$

In this case the mean asperity size and the asperity spread factor are defined in the record 8791101. These values do not affect the adhesion force, F_a, but are used for conversion of the adhesion force into the tangential pull-off force, $F_{a\tau}$ (calculation of the effective diameter, $D_{eff,0}$). Entering zero means applying the default value of 10^{-7} m. Additionally $x_{p,1}$ is set in this case to a large number, to obtain $D_{eff,1}$ proportional to r_{as}, in order to compare the results with the next test case.

```
*     Adhesion force distribution
*
*      <Fa>   sigma
870101 0.0    4.0   * size section 1
870201 0.0    4.0   * size section 2
870301 0.0    4.0   * size section 3
870401 0.0    4.0   * size section 4
870501 0.0    4.0   * size section 5
*
*      <Ras>   sigma         xp0   xa0
879101 0.0    4.0           0.0   0.0

*        Adhesion force mean value <Fa> calculation - model constants
*        van der Waals
*        A1    x1   R    xp1    xa1
879201  0.0   0.0 -1.0  1.0e10  0.0
```

Exactly the same results will be obtained using the asperity size distribution, $\varphi_{as}(r_{as})$, option, if the mean asperity size is specified as the same as above, i.e. 1.0×10^{-7} m for all size sections (in this case the mean asperity radius defined in the record 8791101 is not used):

$$\langle r_{as} \rangle = 1.0 \times 10^{-7} \quad for\ all\ D_p$$

```
*
*      <Ras>    sigma
870101 1.000E-7  4.0   * size section 1, Dp = 0.25E-06
870201 1.000E-7  4.0   * size section 2, Dp = 0.50E-06
870301 1.000E-7  4.0   * size section 3, Dp = 1.00E-06
870401 1.000E-7  4.0   * size section 4, Dp = 2.00E-06
870501 1.000E-7  4.0   * size section 5, Dp = 4.00E-06
*
*      <Ras>   sigma         xp0   xa0
879101 0.0     4.0           0.0   0.0

*        Adhesion force mean value <Fa> calculation - model constants
*        van der Waals
*        A1    x1   R    xp1    xa1
879201  0.0   0.0 -1.0  1.0e10  0.0
```

The results will be identical provided that the adhesive force is proportional to the asperity size, $D_{eff,1} \propto r_{as}$ ($x_{p,1}$ is set to a large number). If this is not the case there will be some small differences in the calculated adhesion force. In contrast to $x_{p,1}$, $x_{p,0}$ does not need to be set to a large number. This value is used only for the spring stiffness calculation and identical results will be obtained if the same value is used consistently in both models (in the present examples zeroes are entered, which means that a default value, $x_{p,0} = 1.0$, is used).

Calculation of the Mean Adhesion Force, $<F_a>$

The mean adhesion force, $<F_a>$, needs to be calculated if the adhesion force distribution is used and the $<F_a>$ is not specified by the user. If the lognormal distribution is used, the mean adhesion force, $<F_a>$, has its usual meaning. In case of the tabulated distribution, the $<F_a>$ represents the nominal adhesion force. The x-coordinate in the distribution table is give the relative adhesion force: $x = F_a/<F_a>$.

For smooth surfaces the literature (see [108], equation 1) gives the following formulae for the adhesion:

$$F_a = \pi \cdot \Delta\gamma \cdot D_p \qquad \text{for small hard particles}$$

$$F_a = \frac{3}{4}\pi \cdot \Delta\gamma \cdot D_p \qquad \text{for large soft particles}$$

$\Delta\gamma$ is the adhesive surface energy, (J/m^2). The value of $\Delta\gamma$ used in [108] is 0.15 J/m^2. In general it may be observed that for smooth surfaces the adhesion force is proportional to the particle diameter:

$$F_a = C \cdot D_p$$

For the value of $\Delta\gamma$ from [108] the value of C is between about 0.3 (large soft particles) and 0.5 (small hard particles).

In reality the deposition surfaces are rough. For these surfaces the adhesion force is more difficult to calculate. Moreover in general one needs to consider several mechanisms of particle adhesion to a surface. For example the gravity force will have a small contribution and may therefore be neglected for small particles, but should not be neglected for large particles. Since in SPECTRA the particle size sections may vary significantly (see section 12.2.1) a general resuspension model must consider the effect of gravity. Discussion of the main adhesive forces and the method of calculation adapted in SPECTRA is discussed below.

The main adhesive forces are (see [75]):

- (1) The van der Waals force
- (2) Force arising from the surface tension of adsorbed liquid
- (3) The electrostatic force

On top of these three adhesion forces a gravity force is added, which is important for large particles:

- (4) The gravity force

All these forces are discussed below.

- (1) The van der Waals force

The van der Waals forces are determined by [75]:

$$F_a = A_{Ham} \frac{D_p}{x_{sep}^2}$$

In the above formula D_p is the particle diameter, (m); x_{sep} is the separation distance, (m), and A_{Ham} is the Hamaker constant, equal to $A_{Ham} = 6\times10^{-20} \div 150\times10^{-20}$ (J) [75].

The model applied in CÆSAR [107] predicts that the particle-surface adhesive force is proportional to the particle diameter and inversely proportional to the surface roughness. The adhesion force is shown in [107] and is reproduced in Figure 12-39.

In SPECTRA it is assumed that the adhesion may be correlated using the effective particle diameter, D_{eff}, and the surface roughness, R:

$$F_{a,1} = \frac{A_1}{R^{x_1}} D_{eff,1}$$

In the above formula D_{eff} is the effective particle diameter, (m); R is the surface roughness, while A_1 and x_1 are user-defined constants. An internal limit is applied in SPECTRA for the surface roughness: $R \geq 10^{-9}$. Therefore for the smooth surface: $F_{a,1} = 10^9 \times A_1 \times D_{eff,1}$. The effective particle diameter is defined differently if the adhesion force distribution is given directly and if it is given through the asperity distribution. When the asperity size is zero, the effective particle diameter is given by:

$$D_{eff,1} = x_{p,1} D_p$$

The x_p is a user-defined multiplier, which may be viewed as a shape factor allowing taking into account a non-spherical shape of particles. The value of x_p is defined separately for each particle size section, thus larger particles (formed from coagulation of smaller ones) may be given different (larger) values of the shape factor. When the asperity size (RASRRT, see Volume 2, records 8791XX) is positive, the effective diameter is (see the description of asperity distribution, above):

$$D_{eff,1} = \frac{1}{\frac{1}{x_{p,1} D_p} + \frac{1}{x_{as,1} 2r_{as}}}$$

where r_{as} is the asperity radius, while $x_{p,1}$ and $x_{as,1}$ are user-defined multipliers with the default values of 1.0 and 3.0 respectively.

The model applied in CÆSAR [107] predicts that the particle-surface adhesive force is proportional to the particle diameter and inversely proportional to the surface roughness, which means x_1=1.0. Results obtained with the formula applied in SPECTRA with $A_1 = 5\times10^{-10}$ and $x_1 = 1.0$ are shown in Figure 12-40. Results agree well with the values calculated by CÆSAR (Figure 12-39). Therefore the values of $A_1 = 5\times10^{-10}$ and $x_1 = 1.0$ are applied as the default values for calculation of the adhesive force.

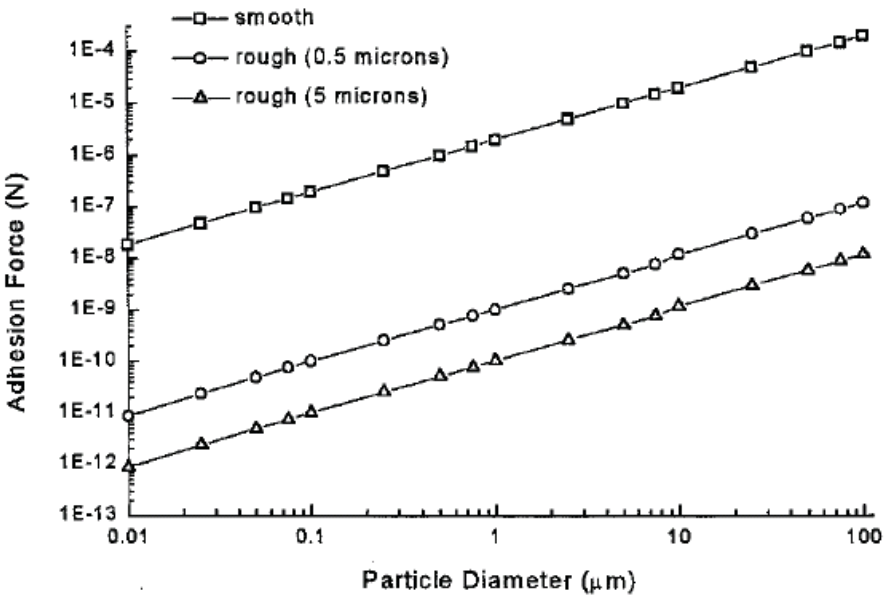


Figure 12-39 Adhesive force of SnO_2 particles to a steel surface, [107].

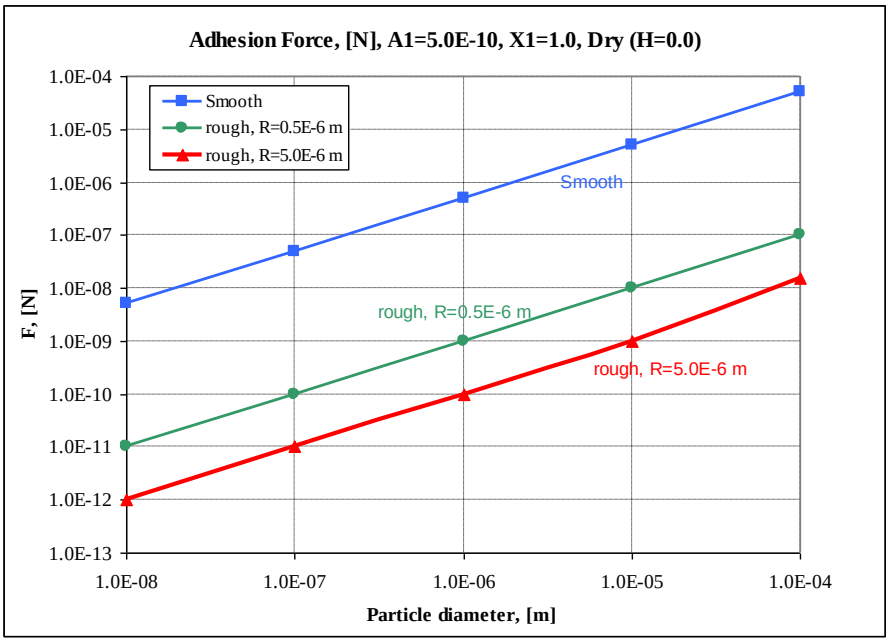


Figure 12-40 Adhesive force, dry atmosphere, $A_1 = 5.0\times10^{-10}$, $x_1 = 1.0$.

Note that in SPECTRA the value of R has an internal limit of a minimum of 10^{-9}. Thus the "smooth surface" line is obtained for $R=10^{-9}$. With the value of $A_1 = 5.0\times10^{-10}$ the proportionality coefficient in the adhesion force is equal to:

$$\frac{A_1}{R^{x_1}} = \frac{5\times10^{-10}}{(1\times10^{-9})^1} = 0.5$$

For smooth surfaces the literature (see [108], equation 1) gives the following formulae for the adhesion:

$$F_a = \pi \cdot \Delta\gamma \cdot D_p \qquad \text{for small hard particles}$$

$$F_a = \frac{3}{4}\pi \cdot \Delta\gamma \cdot D_p \qquad \text{for large soft particles}$$

$\Delta\gamma$ is the adhesive surface energy, (J/m^2). The value of $\Delta\gamma$ used in [108] is 0.15 J/m^2. Therefore:

$$F_a = 0.47 \cdot D_p \qquad \text{for small hard particles}$$

$$F_a = 0.35 \cdot D_p \qquad \text{for large soft particles}$$

To obtain exactly the same values in SPECTRA, the following values should be used:

- $A_1 = 4.7\times10^{-10}$, for small hard particles.
- $A_1 = 3.5\times10^{-10}$, for large soft particles.

The default value, $A_1 = 5.0\times10^{-10}$, is sufficiently close to those values for most practical applications.

The discussion above was limited to the cases when surface asperity did not affect the result. The effective diameter was simply equal to ythe particle diameter:

$$D_{eff,1} = D_p$$

Below the influence of surface asperities is shown. The effective diameter is given by:

$$D_{eff,1} = \frac{1}{\frac{1}{x_{p,1}D_p} + \frac{1}{x_{as,1}2r_{as}}}$$

$x_{p,1}$ and $x_{as,1}$ are user-defined multipliers. With this definition qualitatively good results are obtained for the asymptotic cases:

$$F_a \sim \begin{cases} r_{as} & if \quad r_{as} << D_p \\ D_p & if \quad r_{as} >> D_p \end{cases}$$

Results shown in Figure 12-41 show adhesion forces for the asperity radius of 10^{-7} m, and the default values of the user-defined multipliers (see Volume 2):

- Asperity radius: $r_{as} = 10^{-7}$ m, $x_{p,1} = 1.0$ and $x_{as,1} = 3.0$.

Figure 12-41 shows that when the particle diameter becomes large, the adhesion force does not depend on the particle diameter (lines become horizontal). In this region the adhesion force is governed by the asperity radius, $r_{as} = 10^{-7}$ m. This region is indicated in the figure with the comment "**asperity effect**". In the first case, $(A_1, x_1) = (5.0\times10^{-10}, 1.0)$, the gravity effect becomes visible on the right-hand side of the picture. In the no-asperity effect cases, the forces in this region were so large ($\sim10^{-8}$ Figure 12-40) that the gravity force was almost not visible. In the present case the gravity force ($\sim10^{-9}$ for the large particles) becomes visible in the right-hand side of the figure. This region is indicated in the figure with the comment "**gravity effect**".

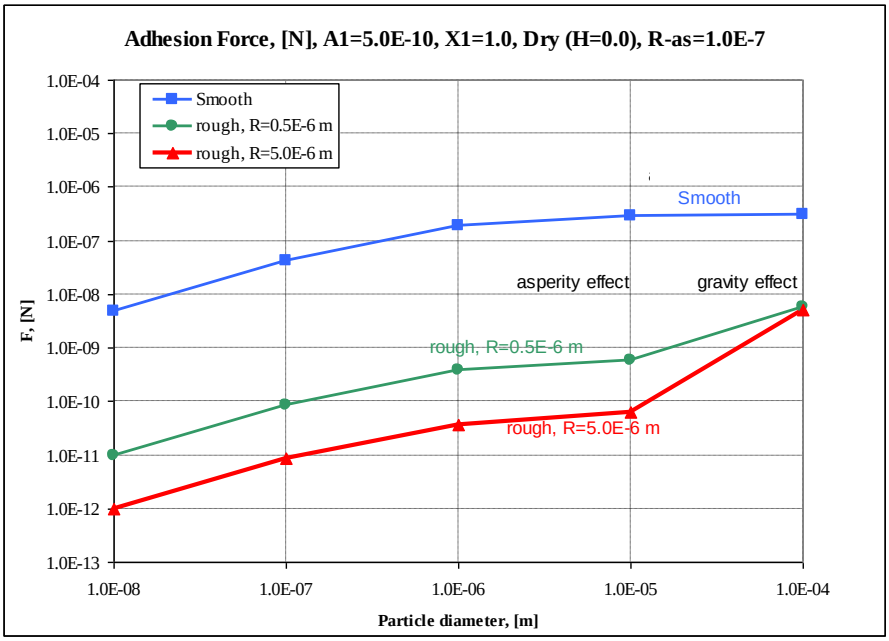


Figure 12-41 Adhesive force, $A_1 = 5.0\times10^{-10}$, $x_1 = 1.0$, $H = 0.0$, $\langle r_{as}\rangle = 1.0\times10^{-7}$.

- (2) Force arising from the surface tension of adsorbed liquid

In humid conditions most materials adsorb liquid molecules on their surface [75]. An attractive force between a particle and a surface is created by a surface tension of the liquid drawn into the capillary space at the point of contact. For relative humidities greater than 90% and smooth surface the force is (see reference [75], equation 6.3):

$$F_{a,2} = 2\pi \cdot \sigma \cdot D_p$$

In the above formula σ is the surface tension, (N/m). For real surfaces at lower relative humidities, the force depends on the curvature of asperities at the points of contact, not on the particle diameter. This curvature varies greatly from particle to particle, giving rise to a distribution of adhesive forces for the same size particles ([75], section 6.1, page 143). In the present implementation it is assumed that the adhesive force on a rough surface is proportional to the effective curvature, resulting from the curvature of the particle itself and the surface asperities. The effective curvature is given by:

$$D_{eff} = \frac{1}{\frac{1}{D_p} + \frac{1}{2r_{as}}}$$

In the implementation it is written as:

$$D_{eff,2} = \frac{1}{\frac{1}{x_{p,2} D_p} + \frac{1}{x_{as,2} 2r_{as}}}$$

where r_{as} is the asperity radius, while $x_{p,2}$ and $x_{as,2}$ are user-defined multipliers, both having the default value of 1.0. If the force distribution is entered, then a single value, representing the average asperity radius (user input), is used. If the asperity distribution is entered, then a different value is used for each r_{as}-section, equal to the r_{as}-section middle value in the logarithmic space:

$$r_{as}(i) = \sqrt{r_{as,\text{sec}}(i) \cdot r_{as,\text{sec}}(i+1)}$$

where $r_{as,\,sec}(i)$ and $r_{as,\,sec}(i+1)$ are the r_{as}-section number i left and right boundaries.

The effect of relative humidity is taken into account using the humidity-dependent function, $f(H)$, defined in the following way. If the humidity is equal to, or higher than 90% then $f(H)$ is equal to 1.0. If the humidity is equal to 0% then $f(H)$ is equal to 0.0. A third order, smooth interpolation is performed in the range between 0 and 90%. Thus the $f(H)$ is defined as follows:

$$f(H) = \begin{cases} 0.0 & if \quad H = 0.0 \\ (3-2X)X^2 & if \quad 0.0 < H < 0.9 \\ 1.0 & if \quad H \geq 0.9 \end{cases}$$

where $X = H/0.9$. Finally, the adhesion force, caused by the humidity is given by (see reference [75], equation 6.3):

$$F_{a,2} = A_2 \cdot f(H) \cdot \sigma \cdot D_{eff,2}$$

where A_2 is a user-defined constant, with a default value of 6.28 (=2π). The adhesive force for a humid atmosphere (H=1.0) is shown in Figure 12-42. The values shown in this figure are the total force coming from the van der Waals force (computed with $A_1 = 5.0 \times 10^{-10}$ and $x_1 = 1.0$) and the humidity (computed with $A_2 = 6.28$).

- (3) The electrostatic force

The electrostatic force is given by (see reference [75], equation 6.2):

$$F_{a,3} = K_E \cdot \frac{q^2}{x_q^2}$$

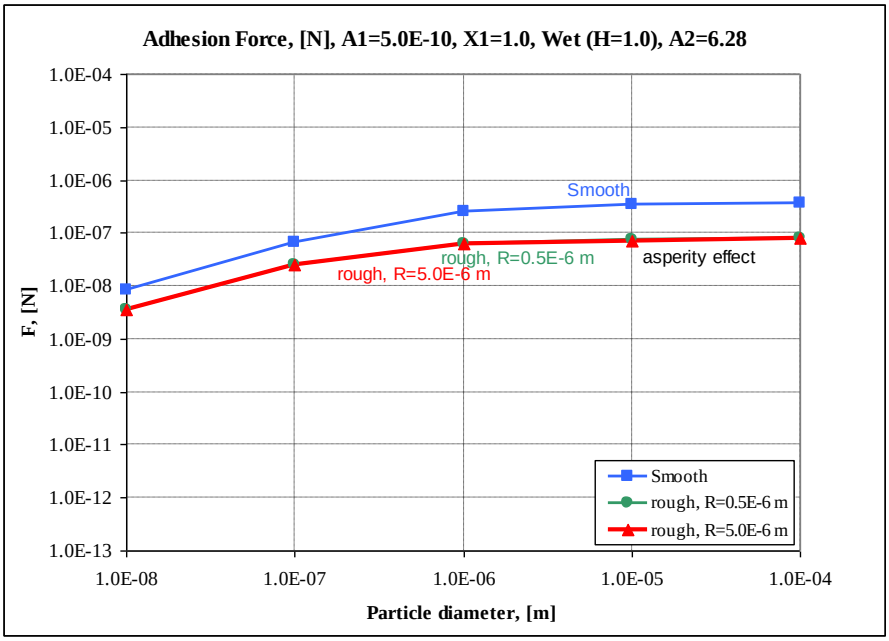


Figure 12-42 Adhesive force, humid atmosphere, $A_1 = 5.0\times10^{-10}$, $x_1 = 1.0$, $A_2 = 6.28$.

The K_E is the Coulomb constant; equal to 9.0×10^9 Nm^2/C^2 ([75], equation 15.2), and x_q is the separation distance of opposite charges. The formula is based on an assumption that the particle charge q induces an equal and opposite charge on the surface. In general, if the induced charge is denoted as q_i, we have:

$$F_{a,3} = K_E \cdot \frac{q q_i}{x_q^2}$$

The induced charge depends on the material and is proportional to q: $q_i = A_3 q$. For non-conducting materials it is close to zero (A_3=0.0), for conductors it is close to the charge q (A_3=1.0).

It is assumed that the separation distance is proportional to the surface roughness, with the proportionality factor, x_E (default value: $x_E = 0.5$). If the particle diameter is smaller than $x_E R$, the diameter is used. The adhesion force caused by the electrostatic forces is given by:

$$F_{a,3} = K_E \cdot \frac{A_3 q^2}{[\min(x_E R, D_p)]^2}$$

A_3, x_E are user-defined constants. $K_E = 9.0\times10^9$, and q is the net electric charge of the particle. The electric charge of particles is calculated from:

$$q = C_q \cdot (D_p \times 10^6)^{x_q}$$

C_q and x_q are input parameters, discused in section 12.2.5.7.

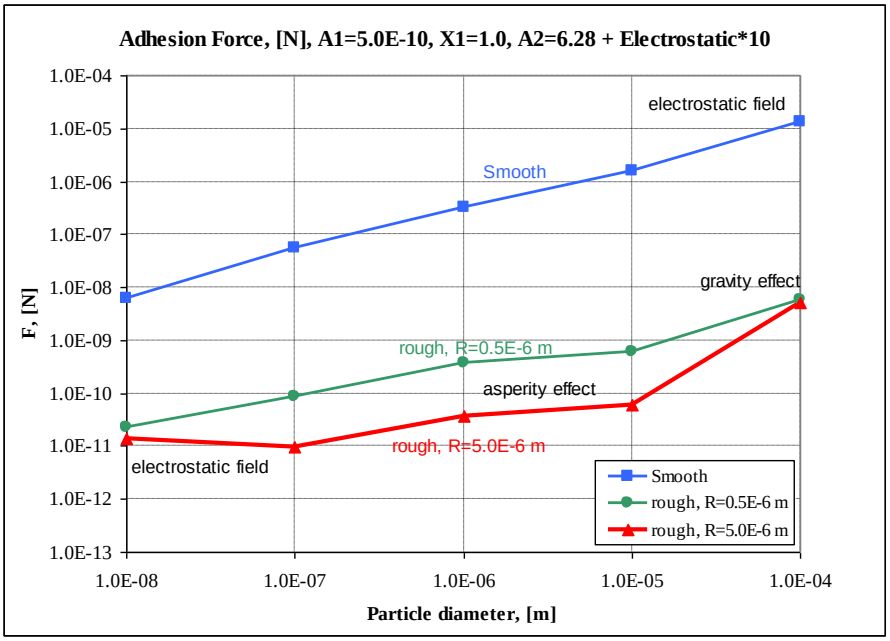


Figure 12-43 Adhesive force, $A_1 = 5.0\times10^{-10}$, $x_1 = 1.0$, $\langle r_{as}\rangle = 1.0\times10^{-7}$, electric charge.

- Minimum charge. An equilibrium (minimum) charge is given by:

$$q = 3.8\times10^{-19}\cdot\sqrt{D_p\times10^6}$$

- A maximum charge of a negatively charged sphere is given by:

$$q = 4.5\times10^{-14}\cdot(D_p\times10^6)^2$$

Figure 12-43 shows adhesion forces calculated for a 10-fold equilibrium charge, which means C_q=3.8×10^{-18}. Influence of the electrostatic field is visible by comparing this figure to Figure 12-41. The effect is visible for large particles on a smooth surface and small particles on a rough surface.

- (4) The gravity force

The gravity force acting on a particle with density ρ_p and volume V_p is:

$$F_{a,4} = g\cdot\rho_p\cdot V_p = \frac{\pi}{6}\cdot g\cdot\rho_p\cdot D_p^3$$

where g is the gravity constant. The gravity force depends on the vertical orientation of a surface. Each heat conductor surface has default gravity factors (δ_{grav}), which can be re-defined by the user through the input data (see Volume 2). The default settings are:

- Horizontal, up-facing surface: $\delta_{grav} = +1.0$
- Vertical surface: $\delta_{grav} = 0.0$
- Horizontal, down-facing surface: $\delta_{grav} = -1.0$

In case of cylindrical and spherical geometries, as well as surfaces with fins or spines, the default values of the gravity factors are between 0.0 and 1.0 (see Volume 2 for detailed description of the default settings for all types of surfaces).

The final formula for the gravity force applied in SPECTRA includes the gravity factor and is:

$$F_{a,4} = C_4 \cdot \rho_p \cdot D_p^3 \cdot \delta_{grav}$$

C is a constant, equal to 5.14 ($=\pi/6\times g$). Influence of gravity force is typically very small or negligible. It may be visible in case of dry surfaces and large particles - see Figure 12-41.

- Total adhesion force

The total adhesion force is calculated as the sum of all forces:

$$\langle F_a \rangle = F_{a,1} + F_{a,2} + F_{a,3} + F_{a,4}$$

The mean adhesion forces are calculated for each particle size section during the input processing and stored for the calculations. The values are printed in the output file. The values of adhesion forces do not change during calculations (are time-independent).

- Practical example of adhesion force definition

It is instructive to show how an empirical expression can be built through the input coefficients. A useful empirical expression based on direct measurement of glass and quartz particles (Corn data) is given by ([75], equation 15.7):

$$F_a = 0.063 \cdot D_p \cdot \left[1 + 0.009\left(\%H\right)\right] = 063 \cdot D_p \cdot \left(1 + 0.9H\right)$$

where D_p is particle diameter in m, $\%H$ is the relative humidity in %, and H is the non-dimensional relative humidity. This formula may be written as:

$$F_a = 0.063 \cdot D_p + 063 \cdot 0.9 \cdot H \cdot D_p = F_{a,1} + F_{a,2}$$

Therefore:

$$F_{a,1} = 0.063 \cdot D_p$$

$$F_{a,2} = (0.063 \cdot 0.9 / \sigma) \cdot \sigma \cdot H \cdot D_p$$

Finally the formula can be approximated in SPECTRA by entering the following values of the user-defined coefficients:

$$A_1 = 0.063$$
$$A_2 = 0.063 \cdot 0.9 / \sigma$$
$$x_1 = 0.0$$

Additionally the asperity radius, r_{as}, should be set to zero (by entering a small positive number), or alternatively setting $x_{a,i}$ to a large value. Either way the effective diameter will be equal to $D_{eff,i}=D_p$. The electrostatic and the gravitational force should be set to zero (default values may be used). The values for $\sigma = 0.059$, $A_2 = 0.96$, are shown in Figure 12-44. For comparison the results obtained directly from the Corn formula are shown using the empty markers.

The SPECTRA-calculated values are identical to the values obtained from the Corn formula for the relative humidities of 0.0 and 1.0. For the relative humidity of 0.5 the SPECTRA values are slightly different because of the third order humidity interpolation applied in the code.

More examples, including how the adhesive force distributions may be defined through the asperity size distributions, are shown in Volume 3.

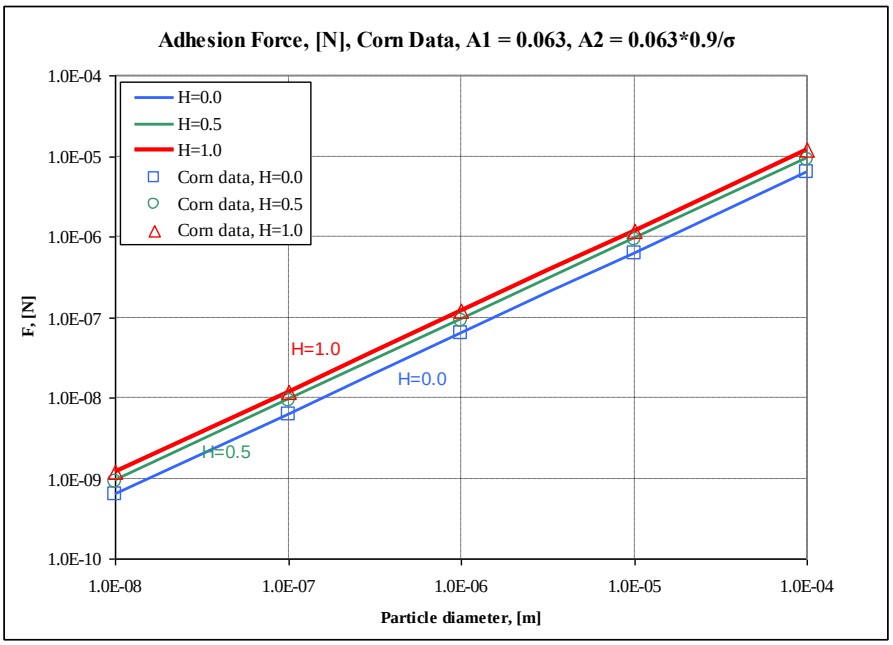


Figure 12-44 Adhesive force, Corn data, $A_1 = 0.063$, $x_1 = 0.0$, $A_2 = 0.96$.

Drag Force, F_d

For the particle in a bulk flow the drag force is calculated from ([75], equation 3.4):

$$F_{d,b} = C_D \cdot \frac{\pi}{8} \cdot \rho_g \left(D_p V_g\right)^2$$

ρ_g - density of gas, (kg/m^3)
D_p - particle diameter, (m)
V_g - gas velocity, (m/s)
C_D - drag coefficient, given by:

$$C_D = \begin{cases} \frac{24}{\mathrm{Re}}(1+0.15\,\mathrm{Re}^{0.687}) & if \quad \mathrm{Re} < 998.947 \\ 0.44 & if \quad \mathrm{Re} > 998.947 \end{cases}$$

The value of the boundary Reynolds number was selected at the cross point of the two lines, i.e. when $24/\mathrm{Re}(1+0.15\mathrm{Re}^{0.687})=0.44$. The Reynolds number is defined as ([75], equation 3.10):

$$\mathrm{Re} = \frac{\rho_g D_p V_g}{\mu_g}$$

μ_g - viscosity of gas, (kg/m/s)

For the resuspension calculation the value of $F_{d,b}$ is not useful, and is presented here only for comparison. For particles deposited on a wall the drag force is given by ([88], eq. 21):

$$F_d^{'} = 8.0 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^2$$

In SPECTRA a user-defined multiplier, X_d, is added. The drag force, as coded within SPECTRA, is given by:

$$F_d = X_d \cdot 8.0 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^2$$

With X_d of one the expression represents the drag force acting on a particle at a wall. However, in the Vainshtein model the mean value of the fluctuating component should be used. The mean value of the fluctuating component is about one third of the full value. This means applying the multiplier of 0.3 in the expression for the drag force ([88], eq. 16). Therefore the default value of the multiplier X_d is 0.3.

D_p^+ is a dimensionless particle diameter, given by:

$$D_p^+ = \sqrt{\frac{f}{8}} \cdot \frac{\rho_g}{\mu_g} \cdot D_p \cdot \left|V_g\right| \cdot \chi$$

χ is the dynamic shape factor, applied for non-spherical particles (see [75], eq. 3.23).

The drag forces, calculated by SPECTRA as described above, are shown in Figure 12-45 and Figure 12-46. The first figure presents the drag force for particle in a bulk flow. The second figure shows the drag force for particle at a wall. This is the force F_d used in the resuspension model. It is seen that the force seen by particle at a wall is much smaller than that experienced by the particle in the bulk flow for the same gas bulk velocity.

For comparison, the approximate data presented in reference [75] (in Table 6.1) is reproduced here in Table 12-5. The adhesion forces presented in this table correspond to 50% humidity and should therefore be compared with the values shown in Figure 12-44. The forces from air current are comparable to the drag forces, $F_{d,b}$, presented in Figure 12-45. It is seen that the values have the same order of magnitude, the values from SPECTRA being calculated being somewhat larger than the value presented in the table.

Table 12-5 Adhesion and removal forces ([75] Table 6.1).

	Force, (N)	
Diameter, (m)	Adhesion	Air current at 10 m/s
1.0×10^{-7}	10^{-8}	2×10^{-10}
1.0×10^{-6}	10^{-7}	2×10^{-9}
1.0×10^{-5}	10^{-6}	3×10^{-8}
1.0×10^{-4}	10^{-5}	6×10^{-7}

Lift force, F_L

The lift force, F_L, experienced by a particle at a wall. The lift forces is calculated from:

- Expression of Soltani [111], valid for $D^+ < 1.8$:

$$F'_{L,Soltani} = 0.975 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^{3.0}$$

- Expression of Hall ([88], eq. 20), valid for $1.8 < D^+ < 70$:

$$F'_{L,Hall} = 4.215 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^{2.31}$$

where the constant is equal to 20.9 divided by $2^{2.31}$.

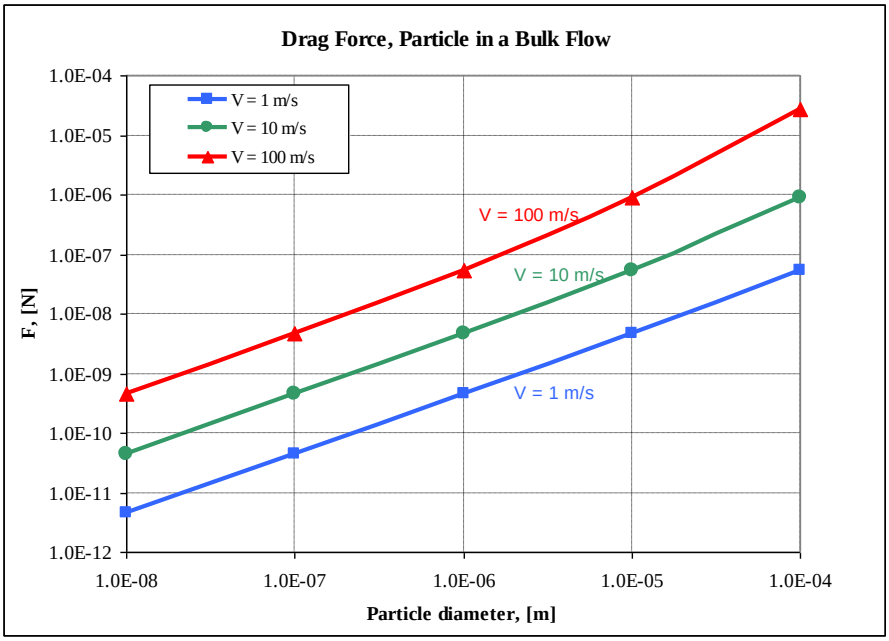


Figure 12-45 Drag force for particle at bulk flow, $F_{d,b}$.

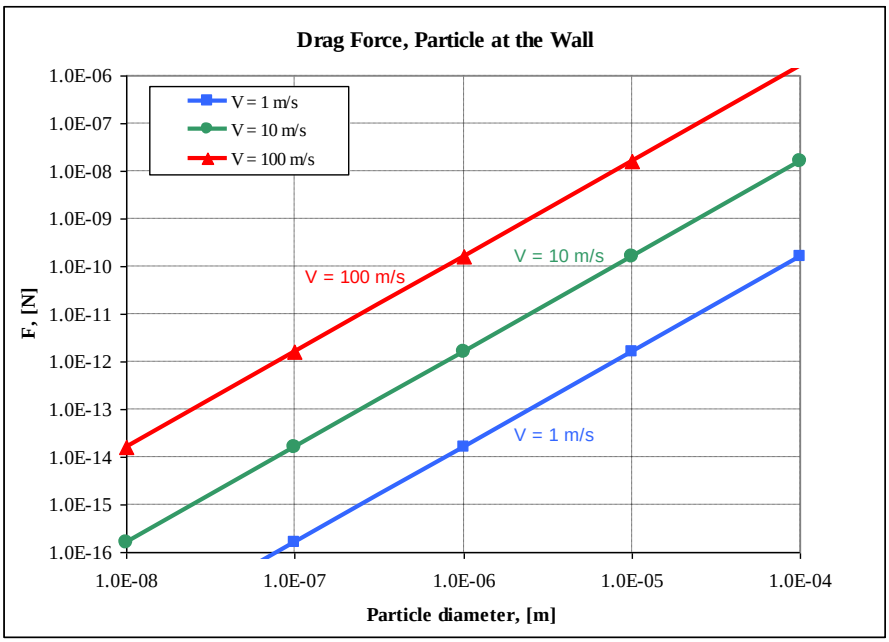


Figure 12-46 Drag force for particle at a wall, F_d, calculated within SPECTRA

In SPECTRA a user-defined multiplier, X_L, is added. In order to obtain smooth transition from one formula to the other, the transition point has been chosen at the crossing point of the Soltani's and the Hall's formulae. The exact value is equal to $(4.215 / 0.975)^{1/(3-2.31)} = 8.34525$. The lift force, as coded within SPECTRA, is given by

- Expression of Soltani, applied for $D^+ < 8.34525$:

$$F_{L,Soltani} = X_L \cdot 0.975 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^{3.0}$$

- Expression of Hall, applied for $D^+ > 8.34525$:

$$F_{L,Hall} = X_L \cdot 4.215 \cdot \frac{\mu_g^2}{\rho_g} \cdot \left(D_p^+\right)^{2.31}$$

As in case of the drag force, the dimensionless particle diameter is given by:

$$D_p^+ = \sqrt{\frac{f}{8}} \cdot \frac{\rho_g}{\mu_g} \cdot D_p \cdot \left|V_g\right|$$

X_L	-	user-defined multiplier (default = 1.0)
f	-	friction factor, (-)
ρ_g	-	density of gas, (kg/m^3)
D_p	-	particle diameter, (m)
V_g	-	gas velocity, (m/s)

Comparison of the expressions of Hall and Soltani is shown in Figure 12-47. Lines calculated with both expressions cross at D^+ of about 10 (the exact value is 8.34525). Results of the model, as coded in SPECTRA are shown in Figure 12-48. The lift forces, F_L, are typically much smaller than the drag forces, F_d.

The lift force is not used in the Vainshtein resuspension model and is not used by default in SPECTRA. However, an option is available to use the lift force or a combination of the lift and the drag force. Recall that the resuspension rate is calculated as:

$$R_m(t) = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_d}\right)^{x_F}\right]$$

In the default option F_d is the drag force, calculated using the multiplier X_d. The lift force, F_L, calculated using the multiplier X_L may substitute the drag force.

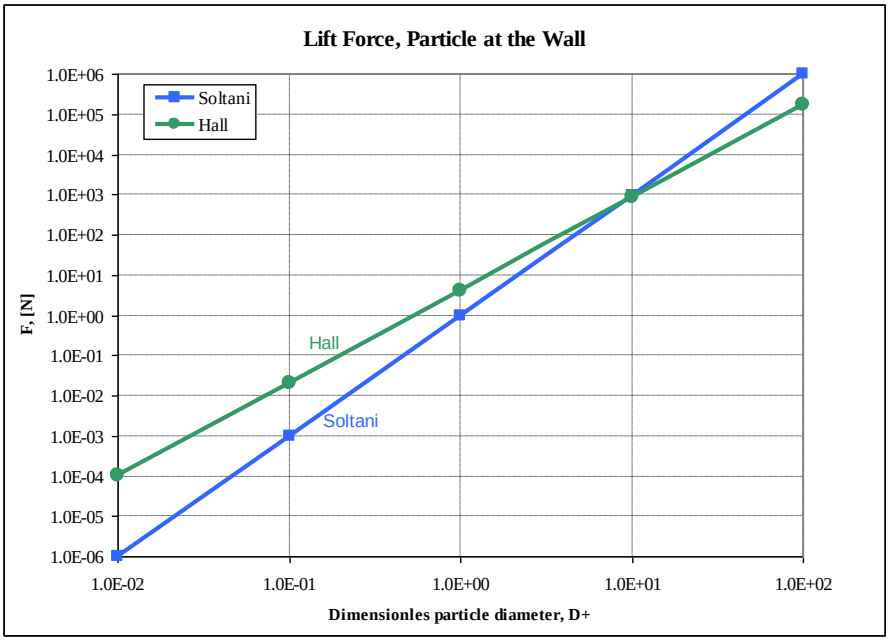


Figure 12-47 Lift force - comparison of the expressions of Hall and Soltani.

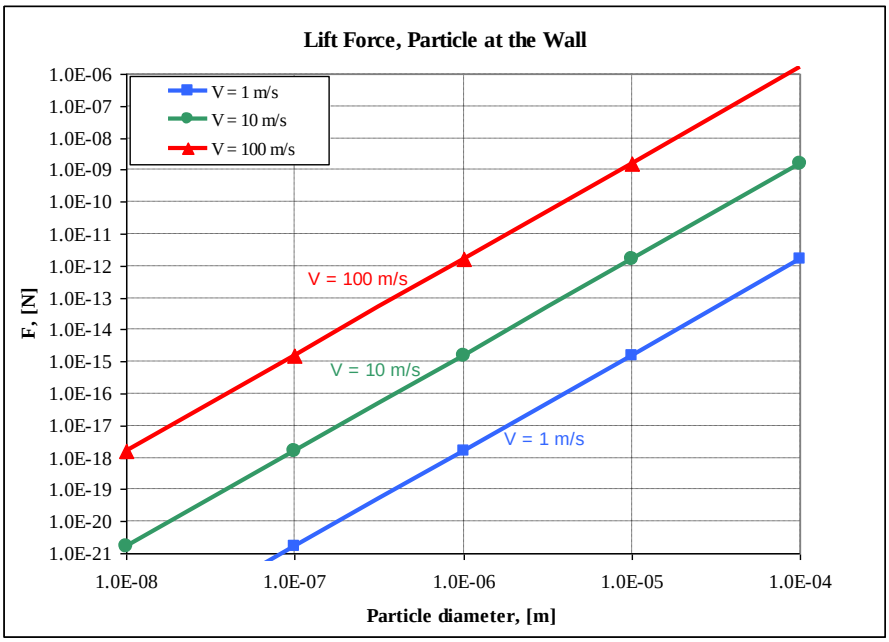


Figure 12-48 Lift force, F_L, calculated within SPECTRA.

In such case the resuspension will be calculated from:

$$R_m(t) = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_L}\right)^{x_F}\right]$$

Finally, the sum may be used. In such case:

$$R_m(t) = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_d + F_L}\right)^{x_F}\right]$$

Note the is the drag force is calculated with the user-defined multiplier X_d, while the lift force, F_L, is calculated with the user-defined multiplier X_L:

$$R_m(t) = f_0 \exp\left[-\left(\frac{F_{a\tau}}{\left(X_d F_d^{'}\right) + \left(X_L F_L^{'}\right)}\right)^{x_F}\right]$$

Since both X_d and X_L are user defined parameters, any linear combination of drag and lift forces may be achieved in the program.

Effective Surface Curvature

An effective surface curvature is used in SPECTRA to define the effective particle diameter:

$$D_{eff,i} = \frac{1}{\frac{1}{x_{p,i} D_p} + \frac{1}{x_{as,i} 2r_{as}}}$$

The formula is based on an effective curvature of two surfaces, a particle, with diameter of D_p and a surface asperity, with the asperity radius of r_{as} (Figure 12-49). First, consider a particle on a smooth surface (Figure 12-49 A). At the distance l from the contact point, the distance between the particle and the surface is H (l may for example represent the length of the adsorbed liquid - see [75], figure 6.3). For a small angle, φ, the following proportion may be written:

$$\frac{H}{l} = \frac{l}{D_p / 2}$$

Next, consider a case when the surface has a curvature with its radius of r_{as} (Figure 12-49 B). For small angles, φ_1 and φ_2 the following proportions are written

$$\frac{H_1}{l} = \frac{l}{D_p / 2} \qquad \frac{H_2}{l} = \frac{l}{r_{as}}$$

or:

$$H_1 = \frac{l^2}{D_p / 2} \qquad H_2 = \frac{l^2}{r_{as}}$$

An effective curvature will have the diameter D_{eff} and the corresponding effective distance H_{eff}. The relation between these two parameters will be the same as that for a particle on a smooth surface, which means:

$$H_{eff} = \frac{l^2}{D_{eff} / 2}$$

The effective distance, H_{eff}, for a situation depicted in Figure 12-49 (B), is equal to:

$$H_{eff} = H_1 + H_2$$

This leads to:

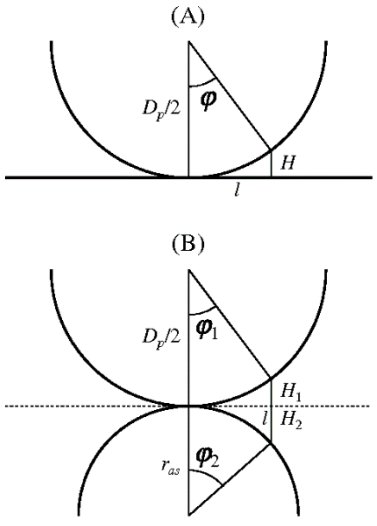
$$\frac{l^2}{D_{eff} / 2} = \frac{l^2}{D_p / 2} + \frac{l^2}{r_{as}}$$

This means:

$$\frac{1}{D_{eff}} = \frac{1}{D_p} + \frac{1}{2r_{as}}$$

Finally:

$$D_{eff} = \frac{1}{\frac{1}{D_p} + \frac{1}{2r_{as}}}$$



In the formula applied in SPECTRA two user-defined multipliers $x_{p,i}$ and $x_{as,i}$ are added to make it more general.

The effective diameter calculated from the above formula provides a smooth transition from two extreme cases, when $D_p \ll r_{as}$, and when $r_{as} \ll D_p$. In those cases the effective diameter is equal to the diameter of the more curved surface:

$$D_{eff} = \begin{cases} 2r_{as} & if \quad r_{as} << D_p \\ D_p & if \quad r_{as} >> D_p \end{cases}$$

Figure 12-49 Effective curvature

Extended Mechanistic Model

The extended resuspension model offers a little more general formula for the resuspension rate calculation. Instead of the formula described above:

$$R_m = f_0 \exp\left[-\left(\frac{F_{a\tau}}{F_d} \right)^{x_F} \right]$$

the following formula is used:

$$R_m = f_0 \exp\left[-C_1 \left(\frac{F_{a\tau} - C_2 F_d}{F_d} \right)^{x_F} \right]$$

C_1 and C_2 are the user-defined parameters (default values of 1.0 and 0.0 respectively). A limit is imposed on the difference, such that it never becomes negative ($F_{a\tau} - C_2\, F_d \geq 0.0$). Therefore if $C_2\, F_d \geq F_{a\tau}$ the result is independent on the forces and is always equal to: $f_0 \exp(0) = f_0$.

Using the extended formulation the "Rock'n Roll" model may be built. Recall that:

$$F_{a\tau} = |C_{Fa}| \cdot F_a \qquad if\ C_{Fa} < 0.0$$
$$F_d = \left(X_d F_d^{'} \right) + \left(X_L F_L^{'} \right)$$

In the "Rock'n Roll" model [88]:

$$R_m = f_0 \exp\left[-\frac{1}{2} \left(\frac{F_a - F_{aero}}{0.2 \cdot F_{aero}} \right)^2 \right]$$

The aerodynamic force, F_{aero}, is given by:

$$F_{aero} = \left(\frac{r}{a} F_d^{'} \right) + \left(\frac{1}{2} F_L^{'} \right)$$

where $(r/a) \sim 10^2$. The frequency of vibration:

$$f_0 = 6.58 \times 10^{-3} \frac{f}{8} \frac{\rho_g V_g^2}{\mu_g}$$

In the "Rock'n Roll" model the R_m is divided by a term with error function:

$$\frac{1}{2} \left[1 + errf \left(\frac{F_a - F_{aero}}{\sqrt{2 \cdot (0.2 \cdot F_{aero})}} \right)^2 \right]$$

This term has been neglected here. This term gives the values between 0.5 and 1.0, which can be accommodated by dividing f_0 by 1/2 (this is achieved by dividing C_{f0}) and performing sensitivity calculations. Results of such sensitivity calculations, shown in Volume 3, prove that the error introduced by neglecting this term is practically negligible. Therefore the following values need to be applied in order to obtain the "Rock'n Roll" model:

$C_1 = 12.5 = (1/2) / (0.2)^2$	$X_d = 10^2 = (r/a)$
$C_2 = 1.0$	$X_L = 0.5$
$C_{Fa} = -1.0$	$x_F = 2.0$
$C_{f0} = 6.58\times10^{-3} \div 13.2\times10^{-3}$	*IDRRT* = 3 (option to allow use of X_d and X_L)

Conclusions

Conclusions on the mechanistic resuspension model in SPECTRA are given below.

- The mechanistic model is based on the work of Vainshtein et al. [108].

- The model as coded is very flexible and different variations may be easily achieved. In particular:
  - The adhesion force distribution may be defined by the user if measured data exist.
  - The adhesion force distribution may be calculated by the code from the built-in correlations. In this case the user must enter the parameters such as surface roughness, relative humidity, etc.
  - The adhesion forces may be determined through the asperity distribution. With this option and the appropriate values of the mean asperity size, $<r_{as}>$, and the spread factor, σ_{as}, the results are identical to the results of the original model. This has been verified in several calculations (see Volume 3).
  - The default number of F_a-sections is 51 (an advantage of having an uneven number of the F_a-sections is having a middle section with the adhesion force F_a exactly equal to the mean value $<F_a>$). It was found out that approximately this number of sections in necessary in order to obtain an accurate solution for typical adhesive spreads $\sigma_a \leq 5$. If the adhesive spread, σ_a, is large, a large number of F_a-sections is necessary to correctly represent section-by section resuspension. For example, with $\sigma_a = 7$, the (min./max.) range is $[\exp(-7) / \exp(+7)] \sim (10^{-3} / 10^{3})$. Thus the adhesive force spread is 6 orders of magnitude. Therefore a maximum number of 51 F_a-sections is selected by default only if $\sigma_a \leq 5$, while it is equal to 99 if $\sigma_a > 5$.
  - In contrast to the applied number of F_a-sections, the number of integration intervals per single F_a-section does not have an important effect on the solution (At least with the default number of F_a-sections. It would have a meaning if very few F_a-sections were used). Therefore the default setting is 1 integration interval per F_a-section (which may be increased by the user to a maximum of 5).
  - If the user-defined distribution option is used, then the user must of course define the distribution function for all integration points (equal to the number of F_a-sections multiplied by the number of integration intervals, plus one). This may be tedious but is necessary to obtain an accurate and consistent (i.e. independent to some small changes of the number of sections) solution. The user-defined fractions are internally normalized by the code, so the values may be scaled by any factor, which makes it easier to prepare the input data.

- An extended option is available. Using this option, the user may easily build other resuspension model, for example the "Rock'n Roll" model.

- The model has been validated against available test data - see Volume 3.

12.2.9 Pool Scrubbing

When a stream of gas enters a pool region of the receiving Control Volume, then it forms bubbles in the pool. The gas is assumed to quickly reach equilibrium at the entrance to the pool (the equilibrium conditions are calculated by the Bubble Collapse Model – see Chapter 2). At the end of the bubble collapse, the mass flow of gas entering the pool, W_B, is divided into the mass flow of gas, W_G, and condensed liquid, W_L. The flow W_G forms bubbles in the pool, while the condensed steam, W_L, remains in the pool. The aerosols entering the pool with the stream of gas are divided as follows:

- The part W_L / W_B remains in the pool.
- The part W_G / W_B is transported partly to the atmosphere and partly remains in the pool, depending on the pool scrubbing efficiency, E_{PS}.

The division of aerosol particles is:

- Fraction remaining in the pool: $\frac{W_G}{W_B} E_{PS} + \frac{W_L}{W_B}$
- Fraction reaching the atmosphere: $\frac{W_G}{W_B}(1 - E_{PS})$

The overall pool scrubbing efficiency, ε_{PS}, is defined as the fraction of aerosols remaining in the pool:

$$\varepsilon_{PS} = \frac{W_G}{W_B} E_{PS} + \frac{W_L}{W_B}$$

When the pool scrubbing efficiency is equal to 1.0, then all aerosols remain in the pool and the overall pool scrubbing efficiency is also 1.0:

$$\varepsilon_{PS} = \frac{W_G}{W_B} \cdot 1.0 + \frac{W_L}{W_B} = \frac{W_B}{W_B} = 1.0 = \varepsilon_{PS}(\max)$$

When the pool scrubbing efficiency is equal to 0.0, then the overall pool scrubbing efficiency is equal to:

$$\varepsilon_{PS} = \frac{W_G}{W_B} \cdot 0.0 + \frac{W_L}{W_B} = \frac{W_L}{W_B} = \varepsilon_{PS}(\min)$$

The pool scrubbing efficiency, E_{PS}, is calculated using one of the two models, described below

- Model 1 – user-defined pool scrubbing efficiency.

The pool scrubbing efficiencies, E_{PS}, are defined for each size section in the input data using either a Tabular or a Control Function.

- Model 2 – correlation.

A correlation is available in SPECTRA to calculate the pool scrubbing efficiency. The correlation is:

$$E_{PS} = A \cdot (1 - \exp[-B \cdot D_p])$$

In the formula the A, B are user-defined coefficients, and D_p is the particle diameter, (m). The default values of the user-defined coefficients are $A = 0.8$ and $B = 0.5 \times 10^6$ (m^{-1}) (see Volume 2).

The pool scrubbing efficiencies, calculated from the above correlation, are shown and compared to the numerical results obtained with SPARC and BUSCA codes in Figure 12-50 and in Volume 3.

The default value of the coefficient A is conservatively taken as 0.8, while the best estimate value is 1.0, as is seen in Figure 12-50. The conservatism in efficiency calculation was applied because pool scrubbing is not calculated in a mechanistic way, but a very simple correlation is used instead.

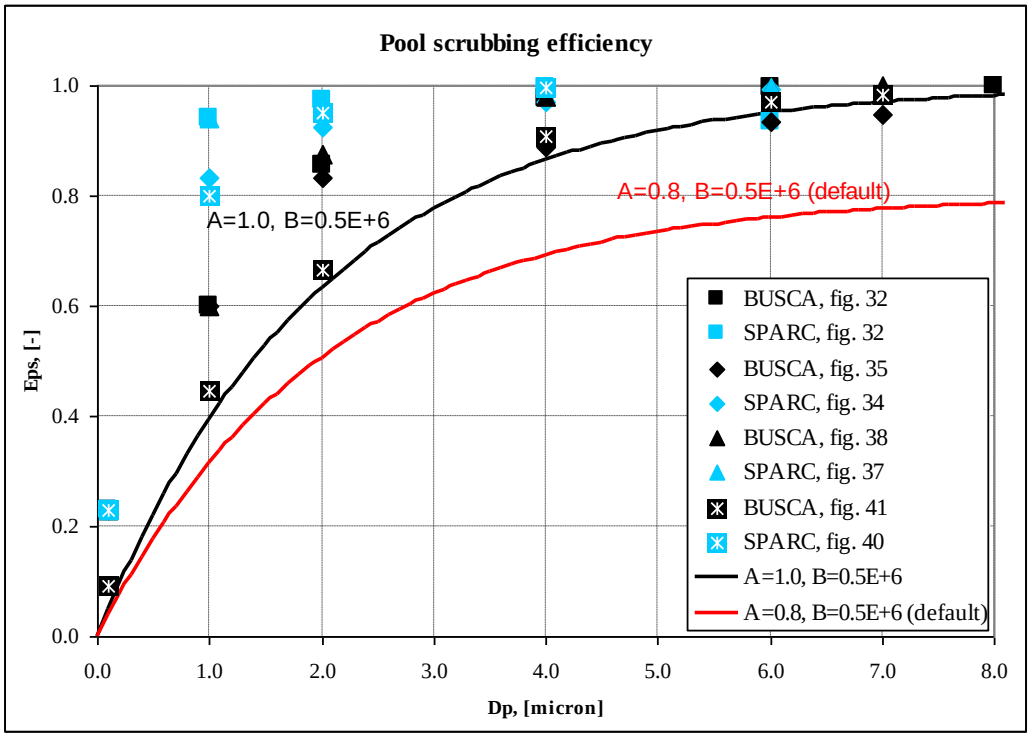


Figure 12-50 Pool scrubbing efficiency correlation.

12.2.10 Aerosol Filters

Two types of aerosol filters are available in SPECTRA.

Filter type 1 – user-defined filter efficiency.

For this type of filter the user defines filter efficiency, separately for each size section. The filter efficiency, ε_F, is defined as a fraction of the incoming aerosols that is removed by the filter:

$$\varepsilon_F = \frac{W_{rem}}{W_{in}}$$

Of course the filter efficiency must be within the range:

$$0.0 \le \varepsilon_F \le 1.0$$

Note that other codes (see for example [46]) often use the decontamination factor, *DF*, rather than the filter efficiency, as an input parameter. The decontamination factor is defined as a ratio of aerosols coming into the filter to the aerosols leaving the filter:

$$DF_F = \frac{W_{in}}{W_{out}}$$

Taking into account that $W_{rem} = W_{in} - W_{out}$, it can be easily shown that the relation between the filter efficiency and the decontamination factor is:

$$DF_F = \frac{1}{1-\varepsilon_F} \qquad \varepsilon_F = \frac{DF_F - 1}{DF_F}$$

The range of the decontamination factor is:

$$1.0 \le DF_F \le \infty$$

Filter type 2 – glass fiber filter.

The glass fiber filter model is implemented based on the data from [75] (section 9). The calculation procedure is as follows:

The single fiber efficiency is calculated using five deposition mechanisms:

- interception, ε_R,
- inertial impaction, ε_I,
- Brownian diffusion, ε_D,
- diffusion-interception, ε_{DR},
- gravitational settling, ε_G,

The single fiber efficiency for interception is given by ([75], equation 9.21):

$$\varepsilon_R = \frac{(1-\alpha)R^2}{Ku(1+R)}$$

where *Ku* is the Kubawara hydrodynamic factor, given by:

$$Ku = -\frac{\ln\alpha}{2} - \frac{3}{4} + \alpha - \frac{\alpha}{4}$$

α	volume fraction of fibers, or packing density, or solidity, (-)
R	ratio of particle diameter to a single fiber diameter d_p / d_f , (-)

The single fibre efficiency for inertial impaction is given by ([75], equation 9.24):

$$\varepsilon_I = \frac{Stk\ J}{2\ Ku}$$

where:

$$J = (29.6 - 28\alpha^{0.62})R^2 - 27.5R^{2.8} \qquad R < 0.4$$

$$Stk = \frac{\rho_p d_p^2 C_m |U_0|}{18\mu_g d_f}$$

ρ_p	density of particle, (kg/m^3)
μ_g	viscosity of gas, (kg/m/s)
d_p	particle diameter, (m)
d_f	diameter of a single fibre, (m)
C_m	Cunningham correction factor, (-)
U_0	face velocity (gas velocity in absence of filter material = Junction velocity), (m/s)

The single fiber efficiency due to diffusion is given by ([75], equation 9.27):

$$\varepsilon_D = 2Pe^{-2/3}$$

where:

$$Pe = \frac{d_f |U_0|}{D_C}$$

The diffusion coefficient is given by ([74], section 3.1):

$$D_C = \frac{k_B T}{3\pi\mu_g\, d_p \chi} C_m = 1.464\times10^{-24} \cdot \frac{T}{\mu\, d_p \chi} C_m$$

k_B	Boltzmann constant (= 1.38×10^{-23}), (J/K).

The Cunningham correction factor, C_m, and the dynamic shape factor, χ, were added following [46] (RN RM section 2.4.2.2)

The single fiber efficiency due to diffusion-interception is given by ([75], equation 9.28):

$$\varepsilon_{DR} = \frac{1.24 R^{2/3}}{(Ku\ Pe)^{1/2}}$$

The single fiber efficiency due to gravitational settling is given by ([75], equation 9.30):

$$\varepsilon_G = G \cdot (1 + R)$$

where:

$$G = \frac{\rho_g d_p^2 C_m g}{18 \mu_g |U_0|}$$

The single fiber efficiency is obtained by assuming that all mechanisms described above act independently. The overall efficiency is equal to ([75], equation 9.33a):

$$\varepsilon_\Sigma = 1 - \prod_{i=1}^{5} (1 - \varepsilon_i)$$

or:

$$\varepsilon_\Sigma = 1 - (1 - \varepsilon_R) \cdot (1 - \varepsilon_I) \cdot (1 - \varepsilon_D) \cdot (1 - \varepsilon_{DR}) \cdot (1 - \varepsilon_G)$$

When the single fiber efficiency, ε_Σ, is known, the total filter efficiency is calculated from ([75], equation 9.19):

$$\varepsilon_F = 1 - \exp\left(-\frac{4 \alpha \varepsilon_\Sigma t}{\pi d_f} \right) = 1 - \exp\left(-1.27 \frac{\alpha \varepsilon_\Sigma t}{d_f} \right)$$

where t is the thickness of the filter. Results obtained with fiber filter test calculations are shown in Figure 12-51 (see also Volume 3).

The filter efficiency is calculated using the single fiber diameter, d_f. Quite often this value is not known for a particular filter. The value however can be deduced from the pressure drop over the filter. The two parameters are related by ([75], equation 9.36)

$$\Delta p = \frac{\mu_g\ t_f\ U_0\ f(\alpha)}{d_f^2}$$

μ_g	gas viscosity (equal to the air viscosity: 1.8×10^{-5} kg/m/s),
T	gas temperature (equal to 300 K),
U_0	face velocity (equal to 2.7×10^{-1} m/s),

$$f(\alpha) = 64 \alpha^{1.5}\ (1 + 56 \alpha^3) \quad for: 0.006 < \alpha < 0.3$$

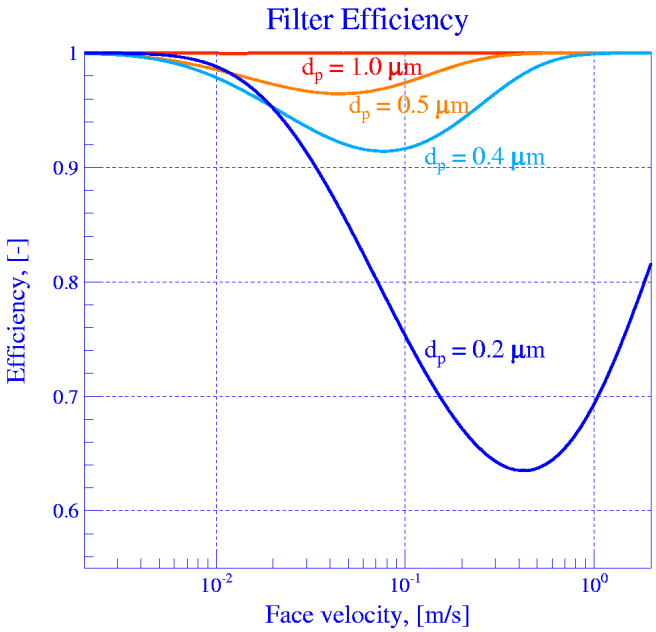


Figure 12-51 Glass fiber filter efficiencies.

The user can enter either the fiber diameter, or the pressure drop over the filter, or both. If only one of the two parameters is entered, then the other parameter is obtained using the above relation (see Volume 2). The code checks if the forward and reverse loss coefficients for the junction where the filter is located are consistent with the filter data. The minimum loss factor is calculated from:

$$K_{\min} = \frac{2\Delta p}{\rho v^2} = \frac{2\Delta p}{1.2 \cdot 0.27^2}$$

where 1.2 is the air density at normal conditions. If the value of the loss coefficient is smaller than $K_{\min}$, then a warning message is printed and the loss coefficient is reset to the value of $K_{\min}$.

12.3 Fission Products

12.3.1 Isotope Chains (Decay Chains)

The concentrations of radioactive fission products are calculated from an isotope balance equation, similar to that used to calculate concentrations of nuclides in the reactor core, described in section 9.3. There are three important differences between the isotopes considered is by the Isotope Transformation Model from the Reactor Kinetics (RK) Package, described in section 9.3, and the decay chains, discussed here.

- The isotopes from the RK Package always reside in the core. There is no release of these isotopes. In contrast to that, the isotopes from the RT Package can be released from the core and transported as vapors or aerosols (section 12.2).
- The isotopes from the RK Package are continuously produced and burned in the core due to fission and capture processes. The isotopes from the RT package have constant mass, equal to the initial mass specified by the user. The initial masses of isotopes are calculated outside SPECTRA (for example by the ORIGEN code) and entered in the input deck (optionally the equilibrium concentrations may be used to calculate the initial masses – this option is only available if the reactor kinetics model is used). The initial masses are kept constant as long as the isotopes reside in the core. Once they are released from the core, the transformation of these isotopes due to decay is calculated as described below.
- The isotopes from the RK package affect the reactivity. The isotopes from the RT package have no effect on reactivity.

The isotope balance is calculated from the equation similar to that used is by the Isotope Transformation Model, described in section 9.3. However, since the RT Package deals only with isotopes released from the core, the terms involving neutron capture and fission, as well as fuel removal and reload, are not included. The only remaining terms are those describing the radioactive decay. Therefore this model is referred to as the "decay chains".

A change of concentration of isotope *i* may be caused by the following reasons:

- Removal due to decay of isotope *i*. The rate of nuclide removal is equal to:

$$-\lambda_i N_i$$

  λ_i is the decay constant (s^{-1}), N_i is the number concentration of the isotope *i* (m^{-3}).

- Production due to decay of some other isotope *j*, from the same chain. The rate of change of *i* is equal to:

$$\sum_{j \neq i} \lambda_i N_i \gamma_{d,j \to i}$$

  $\gamma_{d,j \to i}$ is a yield fraction of isotope *i* from the decay of isotope *j*. In other words, $\gamma_{d,j \to i}$ is a probability of the fact that a decay of a nuclide of isotope *j* will result in production of a nuclide of isotope *i*.

- Production or removal due to processes other than decay (for example, source from fission yields, from release of fission products, from sorption of fission product vapors, etc. The source rate of *i* is generally written as S_i ($s^{-1}m^{-3}$).

Taking both the removal and the production terms, described above, a general balance of nuclide concentration of isotope *i* is written as:

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + S_i$$

This equation can be written in a finite difference form, by replacing dN_i/dt by $(N_i - \overline{N_i})/\Delta t$, where $\overline{N_i}$ is the value of isotope *i* concentration at the start of time step. The finite difference form of the isotope balance equation is:

$$N_i \frac{1}{\Delta t} + \lambda_i N_i - \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} = \overline{N_i} \frac{1}{\Delta t} + S_i$$

where the terms with actual (unknown) nuclide concentrations have been grouped on the left hand side, while other (known) terms were grouped at the right hand side of the equation. The above equation, written for each isotope of a given chain, forms a set of linear equations that can be written shortly in a matrix form:

$$A \cdot N = B$$

where ***N*** is a vector of unknown nuclide concentrations, N_i, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. The elements of matrix ***A*** are equal to:

$$a_{i,j} = \begin{cases} 1/\Delta t + \lambda_i & \text{if } i = j \\ -\lambda_j \gamma_{d,j \to i} & \text{if } i \neq j \end{cases}$$

The elements of vector ***B*** are given by:

$$b_i = \overline{N_i} / \Delta t + S_i$$

This above matrix equation is solved using one of the standard matrix solvers (see section 17.4). A single isotope chain may consists of up to 11 different isotopes. The maximum number of independent decay chains is 20.

All isotopes with positive decay constant, $\lambda_i > 0.0$, contribute to the decay heat. The decay heat is calculated from the following formula:

$$Q_d = V_{chain} \sum_i \lambda_i N_i q_{d,i}$$

Examples of decay chains are shown in Figure 12-53 through Figure 12-57. Isotopes are shown in colors depending on their decay half-life time, *T*. The red color is used for the short-life isotopes (*T* of order of minutes). The stable isotopes are shown in blue. The applied color scale is shown in Figure 12-52.

The data for those isotopes was obtained from [70], [71], and [89]. The decay chains were defined following [89]. The fission yields, γ_f, were obtained from [71] and [89]. Whenever available, the more recent data from [71] was used. The neutron capture cross sections, σ_c, were obtained from [70] and [71]. For most isotopes the data can be found in both references, and the values are very similar. In one case (I-131) the data was significantly different (σ_c = 80.0 barn according to [70], and σ_c = 0.621 barn according to [71]). It was decided to use the data from [71]. The decay yields, γ_d, were obtained from [89]. The half-life times, *T*, were obtained from [71] and [89]. Whenever available, the more recent data from [71] was used. The decay energy data, *E*, was obtained from [70].

The decay chains shown in Figure 12-53 through Figure 12-57 are built into the code and may be requested through input (see Volume 2). If these chains are requested then any parameter can be redefined by the user. The user can also define different chains. The maximum number of decay chains is 24, with up to 11 isotopes per chain.

Initial masses of fission products

The initial masses of isotopes in the code can be entered by the user, or equilibrium concentrations may be calculated by the code. The second option is available only if the Reactor Kinetics Package is used in the calculations. If the equilibrium concentrations are requested, then the concentrations in the reactor are calculated from a simplified isotope balance equation (compare section 9.3.1):

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j\neq i} \lambda_j N_j \gamma_{d,j\to i} - \sigma_{c,i} N_i \Phi + \sum_{j\neq i} \sigma_{c,j} N_j \gamma_{c,j\to i} + \gamma_{f,i} \cdot \sum_{j\in fis} \sigma_{f,j} N_j \Phi =$$
$$= -\lambda_i N_i + \sum_{j\neq i} \lambda_j N_j \gamma_{d,j\to i} - \sigma_{c,i} N_i \Phi + \sum_{j\neq i} \sigma_{c,j} N_j \gamma_{c,j\to i} + \gamma_{f,i} \cdot S$$

Where *S* is the nuclear source and is calculated as a sum over all fissile nuclides present in the Reactor Kinetics (RK) Package. Since the information about all isotopes present in the RK Package is available, the value of *S* is known. The isotope concentrations are calculated by transforming the above equation into a final difference form (similarly as the decay chain equation, shown above):

$$N_i \frac{1}{\Delta t} + \lambda_i N_i - \sum_{j\neq i} \lambda_j N_j \gamma_{d,j\to i} + \sigma_{c,i} N_i \Phi - \sum_{j\neq i} \sigma_{c,j} N_j \Phi \gamma_{c,j\to i} = \overline{N_i} \frac{1}{\Delta t} + \gamma_{f,i} S$$

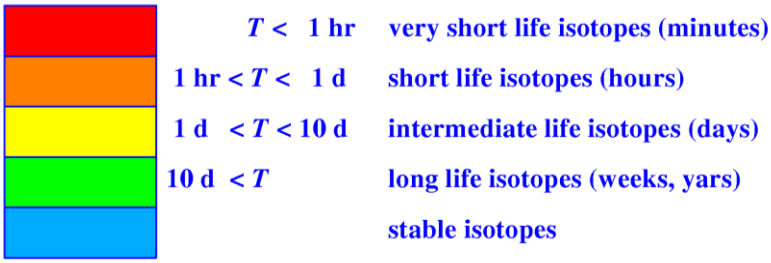


Figure 12-52 Decay chains, colour legend.

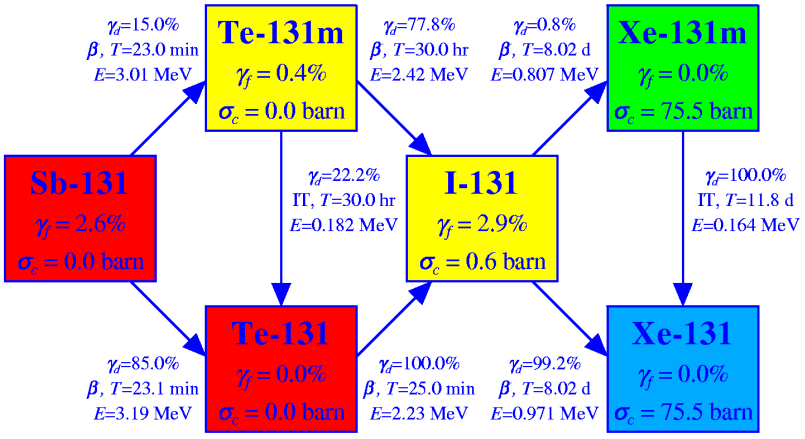


Figure 12-53 Decay chain, M_w = 131.

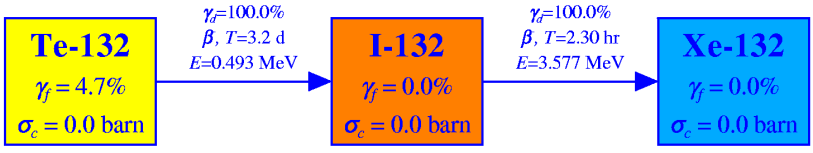


Figure 12-54 Decay chain, M_w = 132.

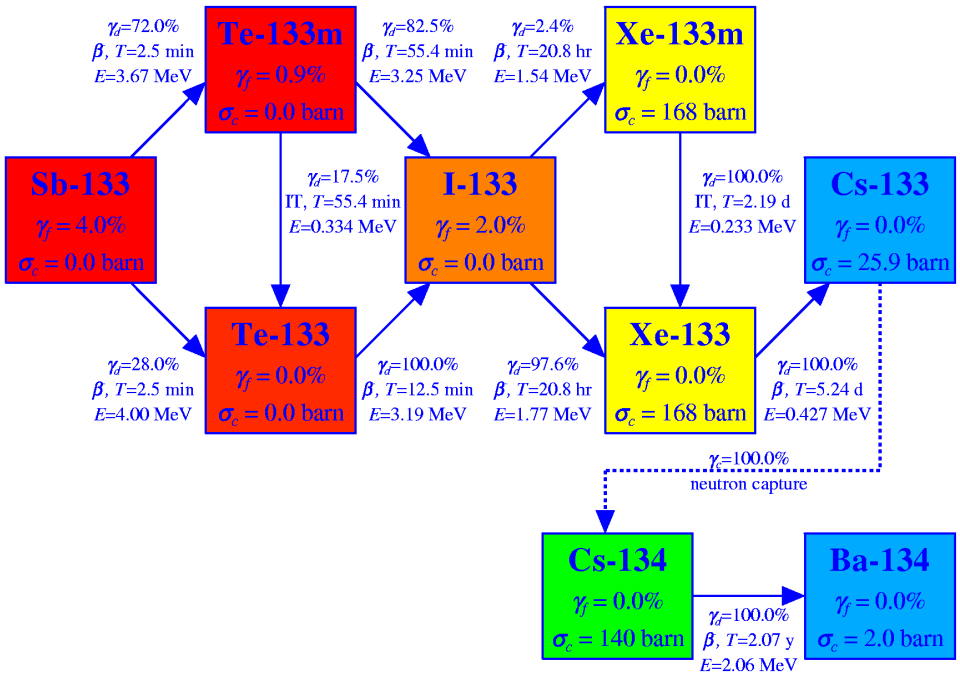


Figure 12-55 Decay chain, M_w = 133 / 134.

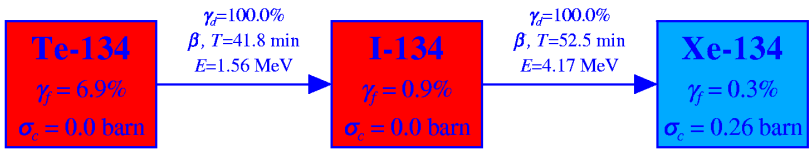


Figure 12-56 Decay chain, M_w = 134.

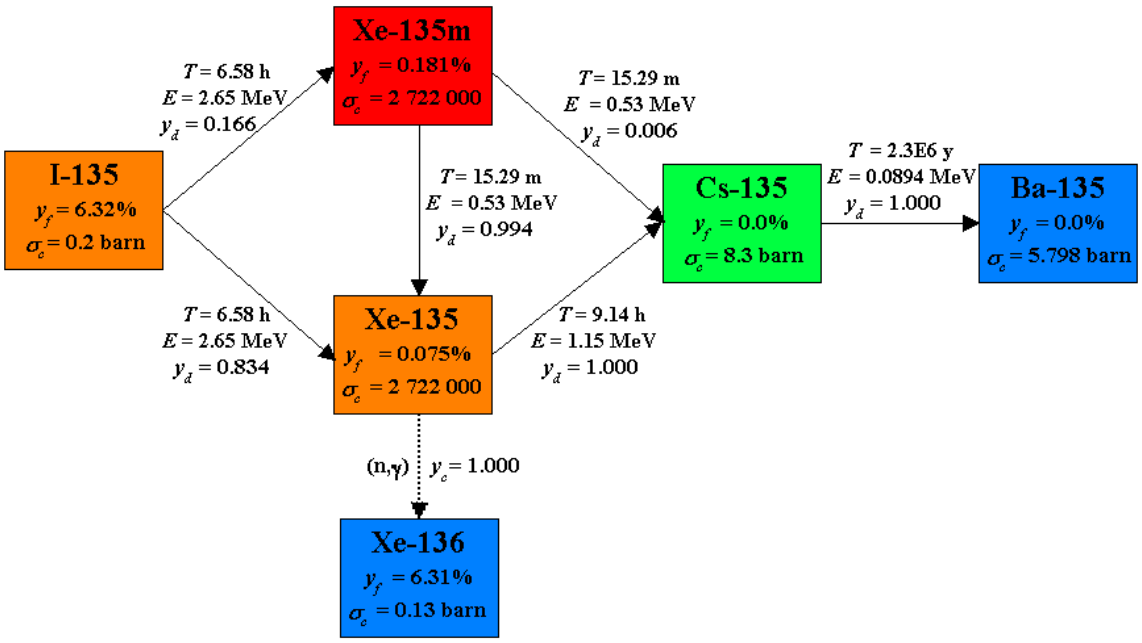


Figure 12-57 Decay chain, M_w = 135 / 136.

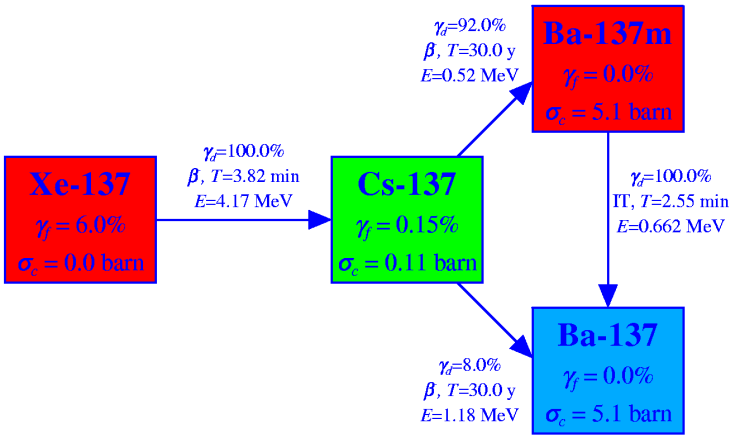


Figure 12-58 Decay chain, M_w = 137.

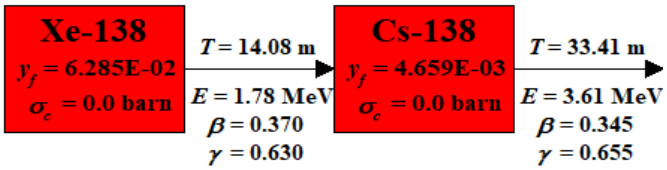


Figure 12-59 Decay chain, M_w = 138.

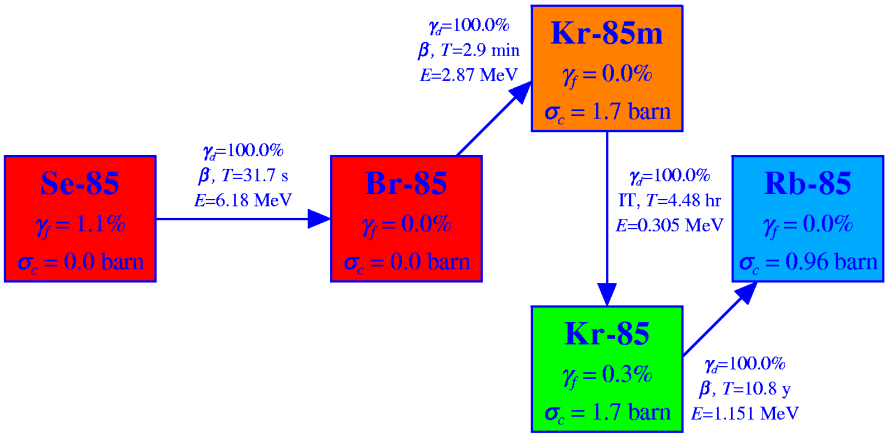


Figure 12-60 Decay chain, M_W = 85.

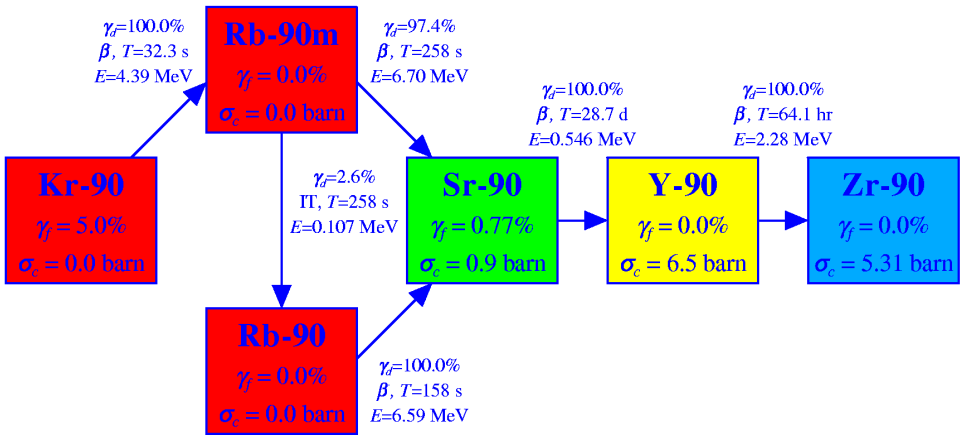


Figure 12-61 Decay chain, M_W = 90.

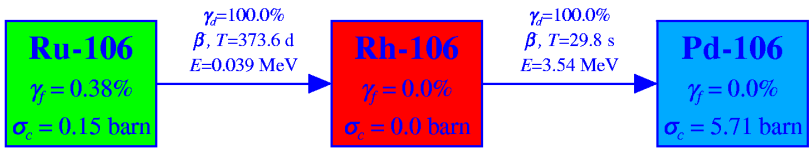


Figure 12-62 Decay chain, M_W = 106.

The above equation, written for each isotope from a given chain, forms a set of linear equations that can be written shortly in a matrix form:

$$A \cdot N = B$$

where $\boldsymbol{N}$ is a vector of unknown nuclide concentrations, N_i, $\boldsymbol{A}$ is a square matrix, and $\boldsymbol{B}$ is a vector of right hand side quantities. The elements of matrix $\boldsymbol{A}$ are equal to:

$$a_{i,j} = \begin{cases} 1/\Delta t + \lambda_i + \sigma_{c,i}\Phi & if\ i = j \\ -\lambda_j \gamma_{d,j\to i} - \sigma_{c,j}\Phi\gamma_{c,j\to i} & if\ i \neq j \end{cases}$$

The elements of vector $\boldsymbol{B}$ are given by:

$$b_i = \overline{N_i} / \Delta t + \gamma_{f,i} S$$

This above matrix equation is solved using one of the standard matrix solvers (see section 17.4).

The time step, and the total integration time are selected based on the user-defined average fuel burn-up. The total time of fuel burn, t_B, is equal to:

$$t_B = \frac{m_{fuel}}{e} \frac{B \cdot 8.64 \times 10^{10}}{Q_R}$$

m_{fuel}	total mass of fuel (fissile isotopes) in the core, (kg), calculated based on the data from the RK Package, equal to: $(V_R \cdot M_{w,f} / A_v) \cdot \Sigma(N_{f,j}\ \sigma_{f,j})$
e	fuel enrichment, (-)
V_R	reactor volume, (m^3) (from the RK Package)
M_V	average molar weight of the fuel (default = 235.0)
A_v	Avogadro number (=6.022×10^{26} - [32]), (1/kmol)
$\sigma_{f,j}$	fission cross section for fissile material j (from the RK Package), (barn)
$N_{f,j}$	concentration of fissile material j (from the RK Package), (m^{-3})
B	fuel burn-up, user-defined input parameter BNUPRT with the default value of 80.0, (MWd/kg). The factor 8.64×10^{10} is a conversion factor from MWd/kg to Ws/kg, = 10^6 (W/MW) × 8.64×10^4 (s/day).
Q_R	reactor fission power, (W)

Note: there is an internal limit on t_B = 1.0 s. Therefore reducing B below the values corresponding to t_B = 1.0 s will not have any effect on the results. The value of t_B (BTIMRT) is printed in the SPECTRA output file below the value of B (BNUPRT).

During the initial state calculation, the total burn-up time, t_B, is divided into N_{step} subintervals, and the decay and neutron capture equations are solved with the time step of:

$$\Delta t = \frac{t_B}{N_{step}} = \frac{t_B}{100}$$

The number of steps was selected based on sensitivity calculations. It was observed that the end masses were clearly different is 1 step and 10 steps were applied. There was practically no difference

between the end masses calculated using 10 steps and 100 steps. Therefore 100 steps were selected to be used.

When the N_{step} steps are performed, the concentrations, N_i, (m^{-3}), of all isotopes from a given chain are calculated for the burn-up B. When the initial concentrations are known, the initial masses, m_i, (kg), are calculated from:

$$m_i = V_R \frac{N_i}{A_v} M_{w,i}$$

V_R total volume of the reactor, (m^3)
$M_{w,i}$ molar weight of the isotope i, (kg/kmol)
Φ neutron flux, (m^{-2} s^{-1})

Individual fission yields

In the above discussion, the fission source is:

$$\gamma_{f,i} \cdot \sum_{j \in fis} \sigma_{f,j} N_j \Phi$$

where $\gamma_{f,i}$ is the average fission yield of isotope i. The values of $\gamma_{f,i}$ for the built-in isotopes are shown in Figure 12-53 through Figure 12-62.

The user may define individual isotope yields for different fissile isotopes, where an individual yield fraction of isotope i from fission of isotope j, $\beta_{j \to i}$, is defined in the input deck (YIFPRT - see Volume 2) for each of the fissile isotopes present in the model. In such case the current fission yields are calculated from.

$$\gamma_i(t) = \frac{\sum_{j \in fis} \sigma_{f,j} \; N_j(t) \, \gamma_{f,j \to i}}{\sum_{j \in fis} \sigma_{f,j} \; N_j(t)}$$

Here: $N_j(t)$ current concentration ($1/m^3$) of the fissile isotope j
$\sigma_{f,j}$ microscopic fission cross section, (m^2), of the fissile isotope j
$\gamma_{j \to i}$ individual yield fraction of the isotope i from fission of the isotope j
$\gamma_i(t)$ current yield fraction of the isotope i

The definition of individual yields and calculation of current yield fraction for different and time-varying concentrations of fissile isotopes are discussed in Volume 3, see tests Kr-88-SOL, Kr-88-LIQ, Xe-136.

12.3.2 Fission Product Release

For the purpose of fission product release, each isotope present in the core is associated with one of the fission product release classes. Several default release classes are built-into the SPECTRA code. New classes can be added and the coefficients of the built-in classes may be modified through input (see Volume 2).

The default release classes were specified following [46]. The same numbering scheme was preserved, but the built-in classes include only the first 12 classes from [46]. The default release classes, and the member elements are shown in Table 12-6.

Table 12-6 SPECTRA default fission product classes.

Release class	Member elements from reference [46]	Default member elements in SPECTRA
1	**Xe**, He, Ne, Ar, Kr, Rn, H, N	Xe, Kr
2	**Cs**, Li, Na, K, Rb, Fr, Cu	Cs, Rb
3	**Ba**, Be, Mg, Ca, Sr, Ra, Es, Fm	Ba, Sr
4	**I**, F, Cl, Br, At	I, Br
5	**Te**, O, S, Se, Po	Te, Se
6	**Ru**, Rh, Pd, Re, Os, Ir, Pt, Au, Ni	Ru, Rh, Pd
7	**Mo**, V, Cr, Fe, Co, Mn, Nb, Tc, Ta, W	-
8	**Ce**, Ti, Zr, Hf, Th, Pa, Np, Pu, C	Zr
9	**La**, Al, Sc, Y, Ac, Pr, Nd, Pm, Sm, Eu, Gd, Tb,	Y
	Dy, Ho, Er, Tm, Yb, Lu, Am, Cm, Bk, Cf	-
10	**U**	-
11	**Cd**, Hg, Zn, As, Sb, Pb, Tl, Bi	Sb
12	**Sn**, Ga, Ge, In, Ag	-

Each of the built-in isotopes, shown in section 12.3.1, has a default assignment to one of the above classes. For example, in case of chain 131, the assignments are:

Sb-131 - class 11
Te-131m, Te-131 - class 5
I-131 - class 4
Xe-131m, Xe-131 - class 1

Each isotope, which can be released from the fuel and whose release is not determined by a user-defined Control Function, needs to be assigned to one of the release classes, as well as one of the vapor classes (see section 12.3.3). The release classes are used to determine the release rates of fission products, while the vapor classes are used to calculate condensation/vaporization of the released fission products. The vapor classes and the release classes are independent, so the same isotope may be associated with release class and vapor class with different numbers. It is however recommended to keep the release class and vapor class numbering the same, to avoid confusion. The built-in vapor classes are the same as the release classes (see section 12.3.3).

Three built-in fission product release models are available:

- CORSOR-M model [90].
- ARSAP model [91].
- HTR-FPR.

The release coefficients are built-in for each vapor class but can be redefined in input. Alternatively the user may define a different release model using a Control Function.

The user may additionally specify a minimum limit on the fission product release rate. In this case the release rate calculated from the one of the above mentioned models would be used if the obtained value is greater than the user-specified minimum release rate for this class, and the class minimum release rate will be used otherwise. The minimum release rate option is useful for taking into account fuel manufacturing defects, which may cause a small release during normal plant operation. The release models are discussed subsequently below.

12.3.2.1 CORSOR-M Model

The CORSOR-M model correlates the release data using an Arrhenius equation form [90]:

$$R = k_0 \exp\left(-\frac{Q}{RT}\right)$$

where: R - release rate (mass fraction released per minute), (1/min)
k_0 - fractional release rate coefficient, (1/min)
Q - release constant, (kcal/mol)
R - gas constant, equal to 1.987×10^{-3} (kcal/mol/K)

This equation is converted to the SI units, as follows:

$$R = A \cdot \exp\left(-\frac{B}{T}\right)$$

where: R - release rate, (1/s)
A - fractional release rate coefficient, (1/s)
B - release constant, (K)

The above equation is used by SPECTRA. Conversion of the coefficients to the SI units is quite simple: $A = k_0 / 60.0$, $B = Q / R = Q / 1.987\times10^{-3}$. The original release coefficients from [90] (also applied in MELCOR 1.8.3 - [46]) and the conversion to the SI units is shown in Table 12-7.

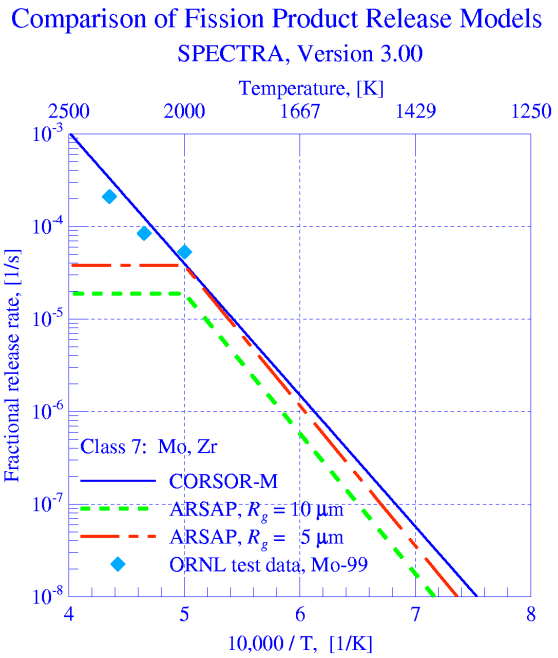
Table 12-7 Release coefficients, original CORSOR-M data [90], [46].

Release class	Member Elements	***A***, (s^{-1}), ($=k_0$ / 60.0)	***B***, (K), ($=Q$ / R)
1	Xe, Kr	2.00×10^{5} / 60.0 = 3.333×10^{3}	63.8 / R = 32108.7
2	Cs, Rb	2.00×10^{5} / 60.0 = 3.333×10^{3}	63.8 / R = 32108.7
3	Ba, Sr	2.95×10^{5} / 60.0 = 4.917×10^{3}	100.2 / R = 50427.8
4	I, Br	2.00×10^{5} / 60.0 = 3.333×10^{3}	63.8 / R = 32108.7
5	Te, Se	2.00×10^{5} / 60.0 = 3.333×10^{3}	63.8 / R = 32108.7
6	Ru, Rh, Pd	1.62×10^{6} / 60.0 = 2.700×10^{4}	152.8 / R = 76899.8
7	(Mo)	0.0	0.0
8	Zr	2.67×10^{8} / 60.0 = 4.450×10^{6}	188.2 / R = 94715.7
9	Y	0.0	0.0
10	(U)	1.46×10^{7} / 60.0 = 2.433×10^{5}	143.1 / R = 72018.1
11	Sb	0.0	0.0
12	(Sn)	5.95×10^{3} / 60.0 = 9.917×10^{1}	70.8 / R = 35631.6

The original CORSOR-M coefficients contain a few deficiencies, discussed below

Class 7, 9, and 11 – no release coefficients.

The release coefficients for the classes 7 (Mo), 9 (La), and 11 (U), are omitted "due to lack of radiological significance" [90]. The same approach was taken in MELCOR 1.8.3 [46]. Recently these coefficients were modified in MELCOR 1.8.5. For the Mo class the coefficients were developed based on fit to test data. For class 9 the data from the classes 10 was applied. For class 11 the data from the classes 12 was applied. In SPECTRA a similar approach was adopted.



For the class 7 (Mo) a data fit has been developed (note that the coefficients were developed independently of MELCOR 1.8.5, so the values are different):

$$A = 3\times10^{4} / 60 = 500.0$$

$$B = 65.0 / R = 3.271\times10^{4}$$

The class 7 release coefficients were checked by comparing results of the CORSOR-M model with the ORNL test data and with the alternative release model – ARSAP. The results are shown in Figure 12-63.

Figure 12-63 New release coefficients for class 7, comparison with ARSAP and ORNL data.

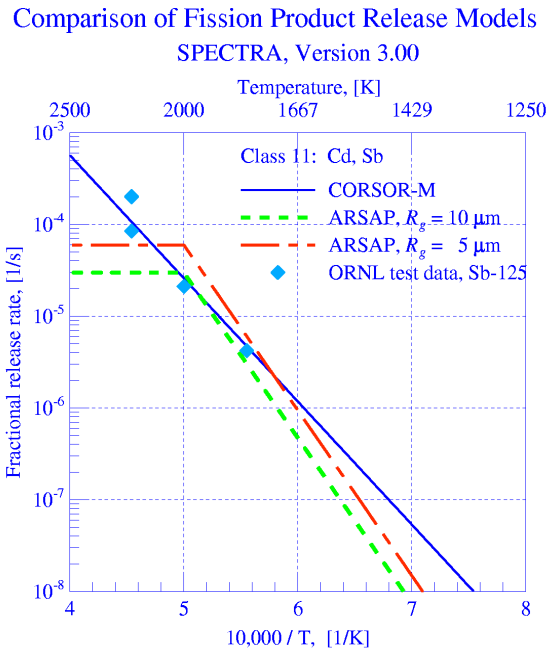


Figure 12-64 New release coefficients for class 11, comparison with ARSAP and ORNL data.

For the class 11 (Sb) the data from class 12 was used. The class 11 release coefficients were checked by comparing results of the CORSOR-M model with the ORNL test data and with the alternative release model – ARSAP. The results are shown in Figure 12-64.

For the class 9 (Y) the data from class 10 was used. The release rates of these classes are very small. At the temperature of 2500 K the release rate is of order of 10^{-8} (1/s).

Class 3 and 6 – inconsistency between CORSOR and CORSOR-M

During comparison of results of the CORSOR-M model with the CORSOR model [90], it was found out that for two classes: 3 (Ba), and 6 (Ru), the results of CORSOR-M with the original coefficients from [90] are very different from the results of CORSOR with the coefficients from the same reference (the CORSOR-M coefficients were derived based on the CORSOR data). Comparison of the CORSOR and CORSOR-M results is shown in Figure 12-65. In case of class 3 the discrepancy is 2 – 3 orders of magnitude. In case of class 6 the discrepancy is 3 – 6 orders of magnitude! Comparison of the CORSOR-M results with measurement data confirmed that the original coefficient give too small release rates for these classes (Figure 12-66).

It was decided to modify the release coefficient k_0 for those two classes, to provide better agreement with the CORSOR model, the ARSAP model, and the ORNL test data. The release constant, Q, was left unchanged.

Class 3: $k_0 = 9.5\times10^{6}$ (instead of 2.95×10^{5})
Class 6: $k_0 = 2.0\times10^{10}$ (instead of 1.62×10^{6})

Results are shown in Figure 12-66. It is seen that with the modified coefficients the CORSOR-M results are in better agreement with the ARSAP model as well as the ORNL test data (not to mention that they are in much better agreement with the original CORSOR model).

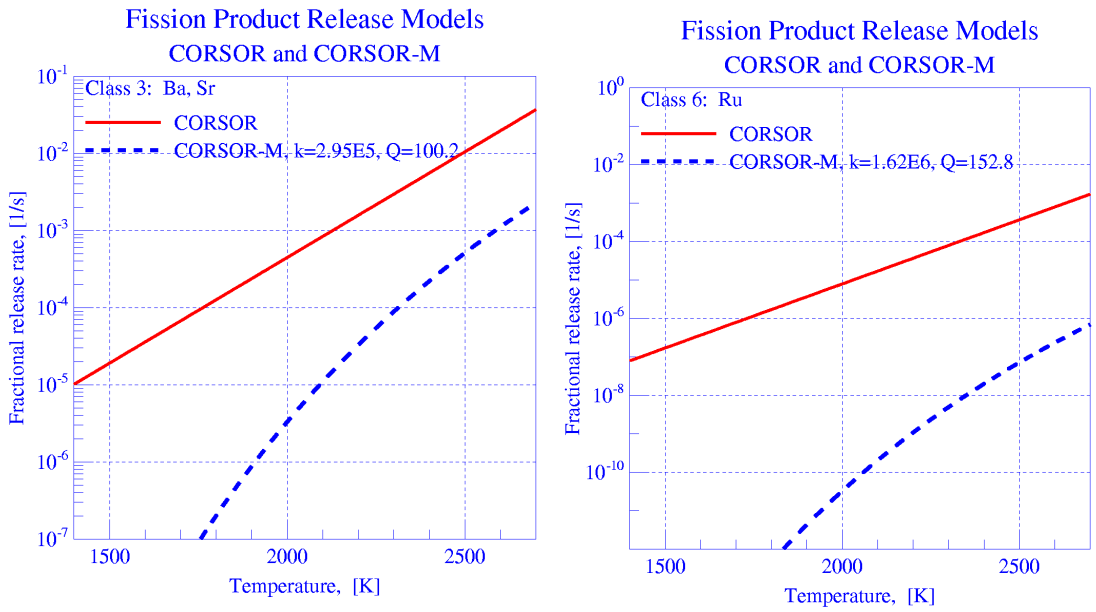


Figure 12-65 Class 3 (Ba) and 6 (Ru), comparison of CORSOR and CORSOR-M.

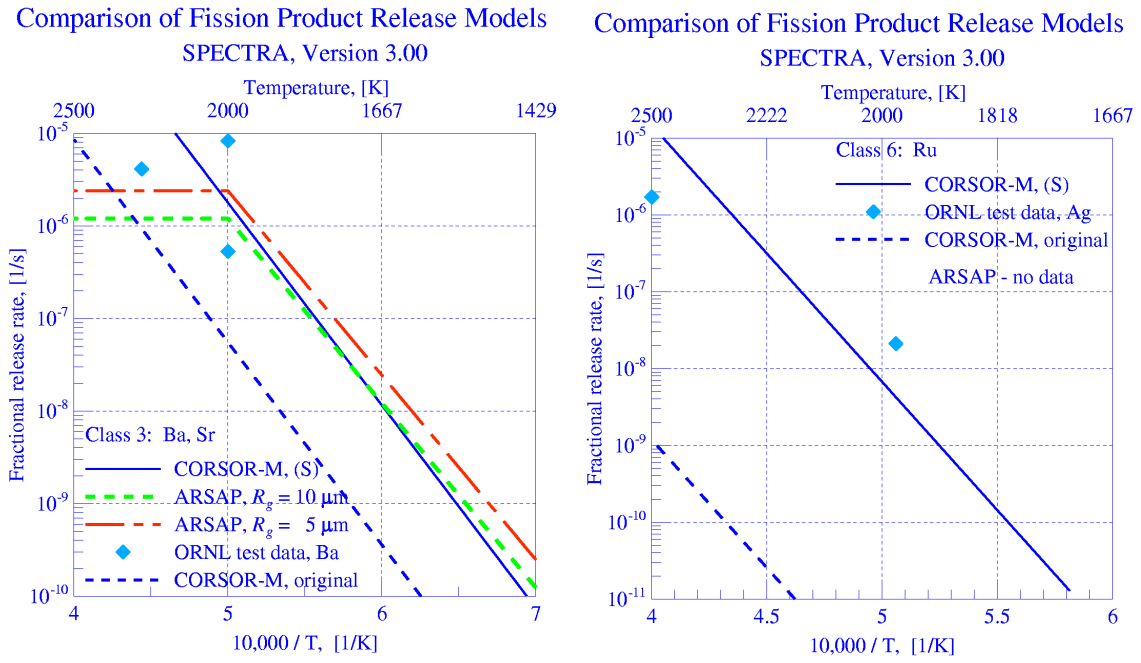
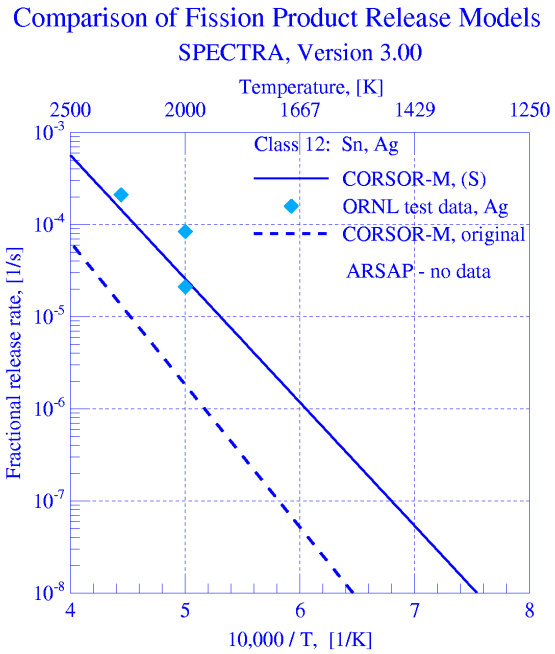


Figure 12-66 Fission product release models, Class 3 (Ba) and 6 (Ru).

Class 12 – Sn or Ag



CORSOR Manual shows separately the release coefficients for Sn and Ag. The values are [90]:

Sn: $k_0 = 5.95\times10^3$ $Q = 70.8$
Ag: $k_0 = 7.90\times10^3$ $Q = 61.4$

The coefficients for Ag provide about an order of magnitude higher release rates. In MELCOR 1.8.3 both Ag and Sn are members of the class 12 [46]. Since the release of Ag is much larger than Sn it was decided to modify the release rate coefficients for the class 12. The release rates are compared to the ORNL test data for Ag in Figure 12-67.

Figure 12-67 Fission product release models, Class 12 (Ag).

The final CORSOR-M default release coefficients, as used by SPECTRA are shown in Table 12-8. The release rate equations, as applied in SPECTRA (SI units) for the 12 release classes, defined following [46], are shown in Table 12-9. A comparison of the CORSOR-M model with the ARSAP model and with ORNL test data for all isotopes is shown in Volume 3.

Table 12-8 Release coefficients, CORSOR-M, modified

Release class	Member Elements	***A***, (s^{-1}), ($=k_0$ / 60.0)	***B***, (K), ($=Q$ / R)
1	Xe, Kr	2.00×10^5 / 60.0 = 3.333×10^3	63.8 / R = 32108.7
2	Cs, Rb	2.00×10^5 / 60.0 = 3.333×10^3	63.8 / R = 32108.7
3	Ba, Sr	9.50×10^6 / 60.0 = 1.583×10^5	100.2 / R = 50427.8
4	I, Br	2.00×10^5 / 60.0 = 3.333×10^3	63.8 / R = 32108.7
5	Te, Se	2.00×10^5 / 60.0 = 3.333×10^3	63.8 / R = 32108.7
6	Ru, Rh, Pd	2.00×10^{10}/ 60.0 = 3.333×10^8	152.8 / R = 76899.8
7	(Mo)	3.00×10^4 / 60.0 = 5.000×10^2	65.0 / R = 32712.6
8	Zr	2.67×10^8 / 60.0 = 4.450×10^6	188.2 / R = 94715.7
9	Y	1.46×10^7 / 60.0 = 2.433×10^5	143.1 / R = 72018.1
10	(U)-	1.46×10^7 / 60.0 = 2.433×10^5	143.1 / R = 72018.1
11	Sb	7.90×10^3 / 60.0 = 1.317×10^2	61.4 / R = 30900.9
12	(Ag)	7.90×10^3 / 60.0 = 1.317×10^2	61.4 / R = 30900.9

Table 12-9 Release rate formulae, CORSOR-M model, SI units.

Class	A	B
1:	R = 3.333E+03 exp (- 32108.7 / T)	
2:	R = 3.333E+03 exp (- 32108.7 / T)	
3:	R = 1.583E+05 exp (- 50427.8 / T)	
4:	R = 3.333E+03 exp (- 32108.7 / T)	
5:	R = 3.333E+03 exp (- 32108.7 / T)	
6:	R = 3.333E+08 exp (- 76899.8 / T)	
7:	R = 5.000E+02 exp (- 32712.6 / T)	
8:	R = 4.450E+06 exp (- 94715.7 / T)	
9:	R = 2.433E+05 exp (- 72018.1 / T)	
10:	R = 2.433E+05 exp (- 72018.1 / T)	
11:	R = 1.317E+02 exp (- 30900.9 / T)	
12:	R = 1.317E+02 exp (- 30900.9 / T)	

The effect of surface-to-volume ratio is taken into account as follows:

$$R = R_{CORSOR\text{-}M} \times (S/V) \,/\, (S/V)_{CORSOR\text{-}M}$$

Here $(S/V)_{CORSOR\text{-}M}$ is the reference surface-to-volume ratio for the CORSOR-M model, equal to 422.5 [46].

12.3.2.2 ARSAP Model

The ARSAP model [91] correlates the release data using the following equation form:

$$R = \frac{1}{R_g} \cdot \exp\left(-\frac{c_1}{T} + c_2 \right)$$

where:
- R - release rate, (1/s)
- c_1 ,c_2 - coefficients
- R_g - grain size, (m)
- T - minimum of the current fuel temperature the FCI temperature, (K)
- T_{FCI} - fuel-cladding interaction (FCI) temperature, (K)

In order to be consistent with the formulation of the CORSOR-M model, the above equation has been transformed in SPECTRA to the following form:

$$R = \frac{C}{R_g} \cdot \exp\left(-\frac{c_1}{T} \right)$$

The transformation is simple: $C = \exp(c_2)$. The release coefficients from [91] and the conversion to the SI units is shown in Table 12-10. The class numbering scheme from the CORSOR-M model (from [46]) was preserved here to avoid confusion. Therefore the elements shown in Table 12-10 are not in the same order as they are appearing in reference [91]. The release rate equations, as applied in SPECTRA for the 12 release classes, are shown in Table 12-11.

The principal differences between the ARSAP and CORSOR-M models are (apart from different values of coefficients): the use of grain size and the FCI temperature in ARSAP. Comparison of CORSOR-M and ARSAP model results are shown for some release classes in Figure 12-63, Figure 12-64, and Figure 12-67. Comparison for all release classes is given in Volume 3.

Table 12-10 Release coefficients, ARSAP [91].

Release class	Elements, [91]	C, (s^{-1}), (=exp(c_2))	c_1, (K)
1	Krypton, Kr	exp(-1.49D+00) = 2.254E-01	37300.0
2	Cesium, Cs	exp(1.34D+00) = 3.819E+00	43000.0
3	Barium, Ba	exp(-2.20D+00) = 1.108E-01	45900.0
4	Iodine, I	exp(9.36D-01) = 2.550E+00	42100.0
5	Tellurium, Te	exp(1.19D+00) = 3.287E+00	44500.0
6	-	0.0	0.0
7	Molybdenum, Mo	exp(-4.94D+00) = 7.155E-03	34900.0
8	Cerium, Ce	exp(20.20D+00) = 5.926E+08	119000.0
9	Neodymium, Nd	exp(24.00D+00) = 2.649E+10	130000.0
10	-	0.0	0.0
11	Antimony, Sb	exp(-1.19D+00) = 3.042E-01	41500.0
12	-	0.0	0.0

Table 12-11 Release rate formulae, ARSAP model, T = Min(T_{fuel}, T_{MCI}).

Class	C	c1
1:	R = (2.254E-01 / Rg) exp (- 37300.0 / T)	
2:	R = (3.819E+00 / Rg) exp (- 43000.0 / T)	
3:	R = (1.108E-01 / Rg) exp (- 45900.0 / T)	
4:	R = (2.550E+00 / Rg) exp (- 42100.0 / T)	
5:	R = (3.287E+00 / Rg) exp (- 44500.0 / T)	
6:	R = (0.000E+00 / Rg) exp (- 0.0 / T)	
7:	R = (7.155E-03 / Rg) exp (- 34900.0 / T)	
8:	R = (5.926E+08 / Rg) exp (-119000.0 / T)	
9:	R = (2.649E+10 / Rg) exp (-130000.0 / T)	
10:	R = (0.000E+00 / Rg) exp (- 0.0 / T)	
11:	R = (3.042E-01 / Rg) exp (- 41500.0 / T)	
12:	R = (0.000E+00 / Rg) exp (- 0.0 / T)	

The effect of surface-to-volume ratio is taken into account as follows:

$$R = R_{ARSAP} \times (S/V) \,/\, (S/V)_{ARSAP}$$

Here $(S/V)_{CORSOR\text{-}M}$ is the reference surface-to-volume ratio for the ARSAP model, equal to 422.5.

12.3.2.3 HTR Fission Product Release Model (HTR-FPR)

The model is specific to TRISO particles, applied in High Temperature Reactors. The model consists of two parts: particle failure and diffusion through the coating. These are discussed subsequently below.

- **Particle failure**

The particle failure model determines the fraction of TRISO particles that has failed during the transient. The failure fraction is assumed to be a function of temperature and relative fuel burn-up. Two options are available: (a) built-in correlation, (b) user-defined 2-D table. The 2-D table defines the failure fraction as a function of temperature and burn-up. Using this option any function may be easily tabulated. An example is shown in Volume 2. The built-in correlation is described below.

The correlation has been developed to match the data of AREVA [214], Figure 12-68. The correlation is:

$$\log_{10}(F) = A_0 + A_1 \cdot \theta + A_2 \cdot \exp(-n_2 \cdot \theta)$$

$$\theta = \frac{T - 1373}{1000}$$

Here F is the failure fraction (-), T is the temperature, A_0, A_1, A_2, n_2, are burn-up dependent coefficients:

$$A_0 = -3.977 + 0.115 \cdot B$$

$$A_1 = 1.5$$

$$A_2 = -0.73 + 1.71 \cdot B - 0.70 \cdot B^2$$

$$n_2 = 6.0 + 2.0 \cdot B$$

Here B is dimensionless burn-up. B=0.0 means fresh fuel, while B=1.0 means maximum burn-up (the relative burn-up of 1.0 corresponds to 80,000 MWd/t). The correlation is compared to the source data in Figure 12-69. The agreement with the source data is very good. The main advantages of using a correlation rather than tabulated data are:

(a) user convenience (no need to define tables),
(b) values for intermediate values of burn-up are easily obtained,
(c) correlation may be extended beyond the range of the tabulated values.

- *Very high temperatures*

The applicability range of the correlation was extended to very high temperatures as follows. In the very high temperature range the coating will be damaged. Reference [215] shows that at very high temperatures (2500 - 2800°C) a steep increase in the release fraction is observed - Figure 12-70. The correlation provides an approximately linear (in the logarithmic scale) dependency of failure fraction and temperature. In order to obtain a similar shape as shown in Figure 12-70, an additional term was added: $A_3 \times \exp(\theta^{n3})$. The full correlation is:

$$\log_{10}(F) = A_0 + A_1 \cdot \theta + A_2 \cdot \exp(-n_2 \cdot \theta) + A_3 \cdot \exp(\cdot \theta^{n_3})$$
$$A_0 = -3.977 + 0.115 \cdot B$$
$$A_1 = 1.5$$
$$A_2 = -0.73 + 1.71 \cdot B - 0.70 \cdot B^2$$
$$n_2 = 6.0 + 2.0 \cdot B$$
$$A_3 = 10^{-2}$$
$$n_3 = 4.0$$

The term with A_3 is only meaningful for temperatures higher than 2400°C; in this region the curves bend upwards - Figure 12-71. The full correlation is shown in Figure 12-72.

- *Very high burn-ups*

The correlation was extended to burn-ups higher than 1.0 (the relative burn-up of 1.0 corresponds to 80,000 MWd/t). It was checked that the correlation provides reasonable values up to B = 2.0. This is the maximum value that may be applied (Volume 2, input parameter BTRIRT). Results obtained with B = 2.0 are shown in Figure 12-73.

The mass of fission products released during a single time step from TRISO (fuel region) to graphite (SC) due to failure of particles, is calculated from:

$$\left(\frac{dM_{FP}}{dt}\right)_{fail} = -\left(\frac{M_{FP} \cdot \Delta F}{(1-F_0)} + F_0 \cdot \Delta M_{FP,f}\right) \cdot \frac{1}{\Delta t}$$

M_{FP} mass of fission product in fuel, (kg)
ΔF increase of the failure fraction, end of time step, (-), $\Delta F = F - F_0$
F failure fraction, end of time step, (-)
F_0 failure fraction, start of time step, (-)
$\Delta M_{FP,f}$ increase of the mass of fission product in fuel due to fission during time step, (kg)
Δt time step size, (s)

The first term, $M_{FP} \cdot \Delta F/(1-F_0)$, represents the mass that is released from TRISO particles due to increase of the failure fraction during a single time step. The second term, $F_0 \cdot \Delta M_{FP}$, represents the release from particles that were already failed at the start of the time step. This term is positive if there is mass source of a given isotope in the core due to fission (isotopes with positive fission yield). The source due to decay of a parent isotope is not contributing to this source. Volume 3 gives examples and explanations of the cases when there is a source due to fission and due to decay. In short, fission is creating isotopes in intact and failed particles at the same time. Decay is calculated separately for fission products that are staying in the fuel and separately for the released fission products.

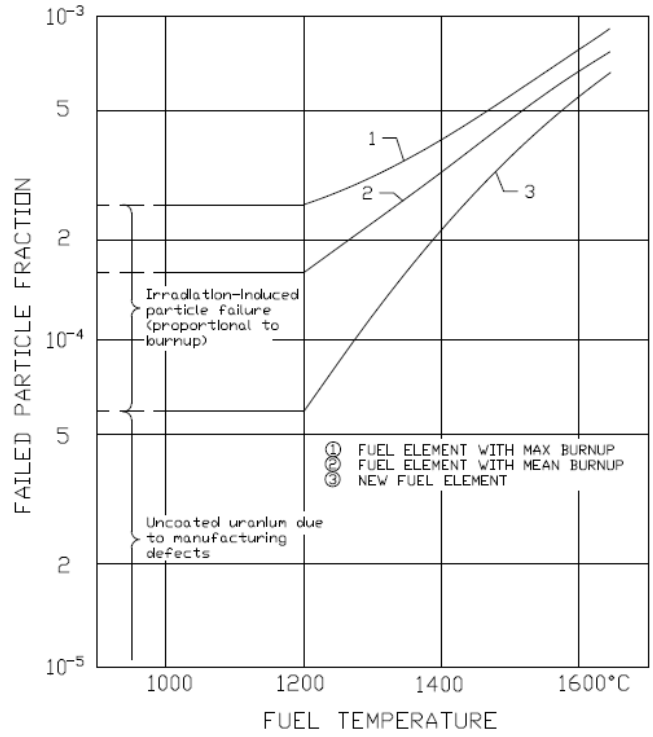


Figure 12-68 Failure fractions of TRISO particles - [214]

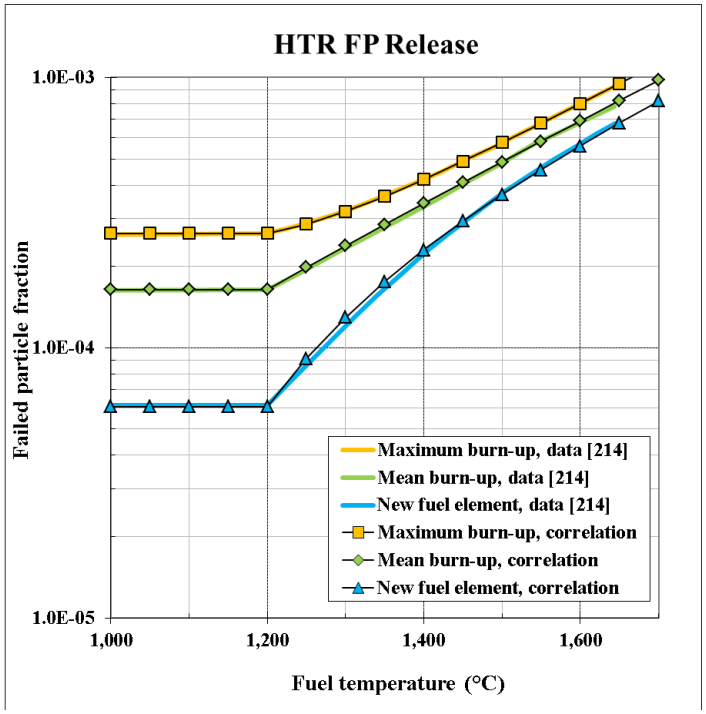


Figure 12-69 Failure fractions of TRISO particles - correlation

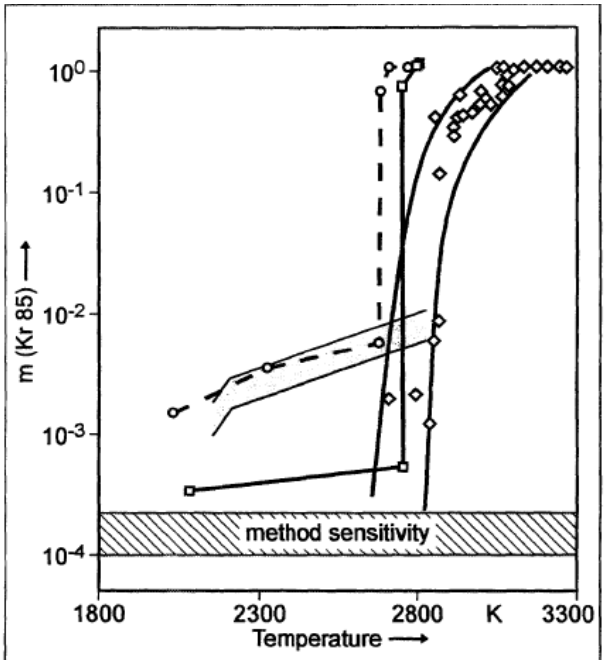


Figure 12-70 Failure fractions of TRISO particles - very high temperature data [215]

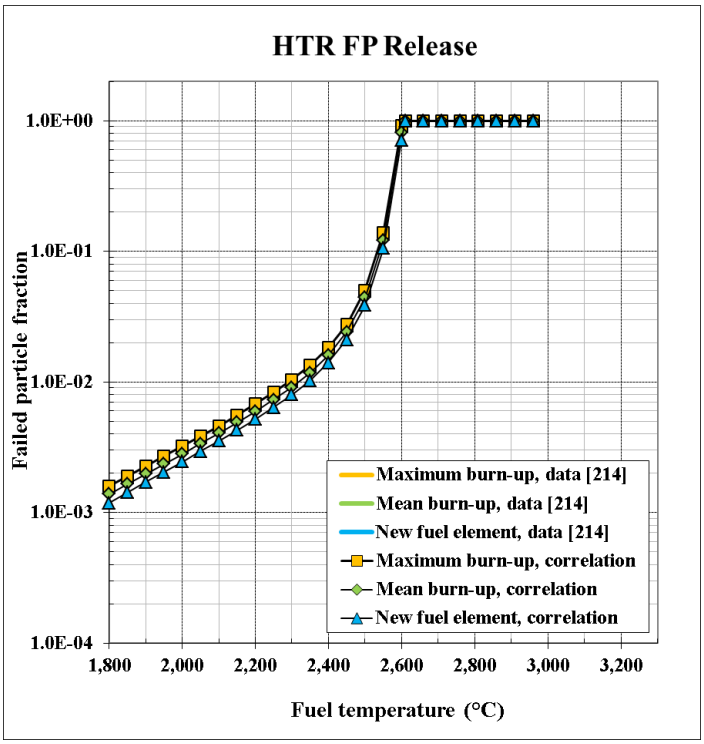


Figure 12-71 Failure fractions of TRISO particles - extension to very high temperatures

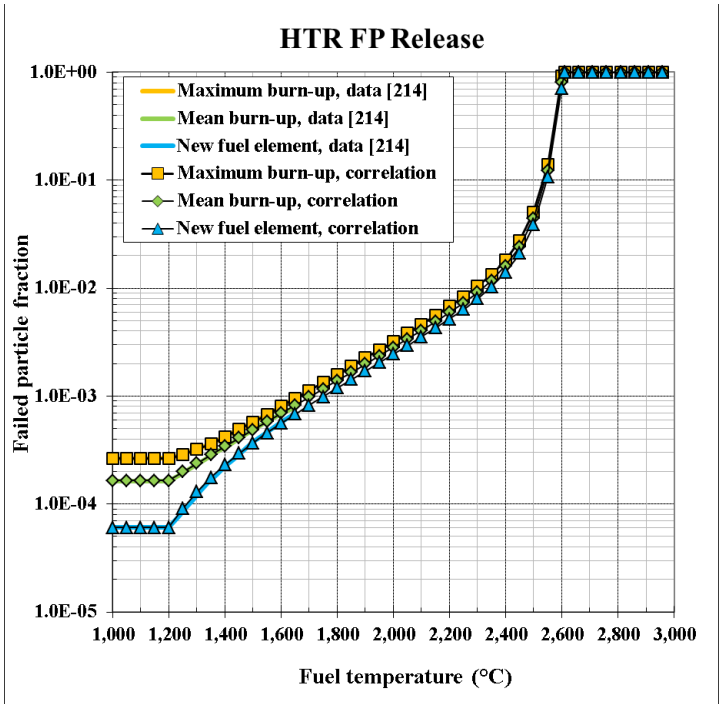


Figure 12-72 Failure fractions of TRISO particles - full temperature range

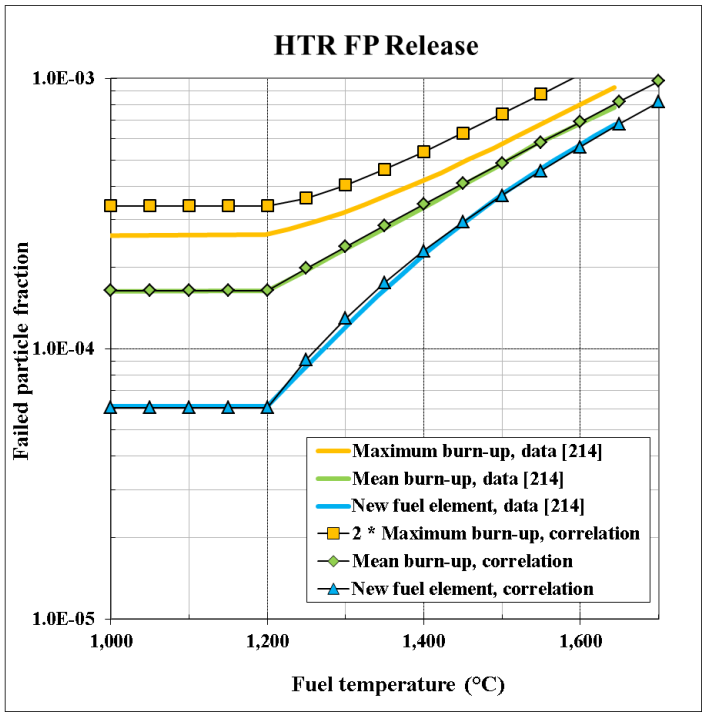


Figure 12-73 Failure fractions of TRISO particles - extension to very high burn-ups

- **Diffusion through the coating**

The second mechanism of fission product release is diffusion through the coating layers. A diffusion equation may be applied directly for TRISO particles. Examples are shown in Volume 3. However this would require to model a representative TRISO particle for each region of the core. In case of large models this would be impractical and too time-consuming. Therefore a simplified approach is taken. The release rate is calculated from the following equation:

$$\left(\frac{dM_{FP}}{dt}\right)_{diff} = -A_{TRISO} \cdot \frac{C_{FP}}{R_{diff}} = -\frac{A_{TRISO}}{V_{TRISO}} \cdot \frac{M_{FP}}{R_{diff}} = -\frac{3}{R_{TRISO}} \cdot \frac{M_{FP}}{R_{diff}}$$

C_{FP} concentration of fission product isotope in the kernel of TRISO particles, (kg/m^3)
M_{FP} mass of fission product isotope in the kernel of TRISO particles, (kg)
A_{TRISO} surface area of the kernel of all TRISO particles, (m^2)
V_{TRISO} volume of the kernel of all TRISO particles, (m^3)
R_{TRISO} radius of the kernel of a TRISO particle, (m) (input parameter RTRIRT, Volume 2)
R_{diff} effective resistance for diffusion of coating layers, (s/m). The value is calculated from:

$$R_{diff} = \frac{X_{\text{ker}}}{D_{\text{ker}}} + \frac{X_{SiC}}{D_{SiC}} + \frac{X_{PyC}}{D_{PyC}}$$

D_{ker} diffusion coefficient in the fuel kernel, (m^2/s)
X_{ker} effective thickness of the kernel, (m) (equal to ¼ of the kernel radius RTRIRT, Volume 2)
D_{SiC} diffusion coefficient in the SiC layer, (m^2/s)
X_{SiC} thickness of the SiC layer, (m) (input parameter TTRIRT(1), Volume 2)
D_{PyC} diffusion coefficient in the PyC layer, (m^2/s)
X_{PyC} thickness of the PyC layer, (m) (input parameter TTRIRT(2), Volume 2)

The effective thickness of kernel, $X_{ker,}$ is equal to ¼ of the kernel radius, for the following reason. For any radius r, within the kernel, the distance from the kernel surface is equal to $R_{TRISO} - r$ - Figure 12-74.

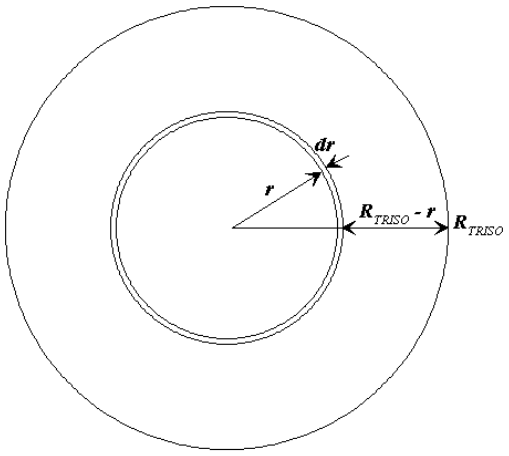


Figure 12-74 Calculation of effective diffusion distance in the fuel kernel

The average distance is given by:

$$\bar{r} = \frac{\int_0^{R_{TRISO}} (R_{TRISO} - r) 4\pi r^2 dr}{\int_0^{R_{TRISO}} 4\pi r^2 dr} = \frac{[(1/3)R_{TRISO} r^3 - (1/4) r^4]\Big|_0^R}{(1/3) r^3 \Big|_0^R} = \frac{1}{4} R_{TRISO}$$

The correctness of this formula is checked by comparing the results obtained with this formula with the results obtained with the exact diffusion model. Such comparisons are shown in Volume 3.

The theoretical formula uses the constant 3, which is appropriate for spherical geometry (see [215], eq. 4-9). Test calculations, discussed in Volume 3 (TRISO-Cs test), showed that better results may be obtained by using slightly different constant. Therefore the formula used in the code is:

$$\left(\frac{dM_{FP}}{dt} \right)_{diff} = -\frac{C_{FPR}}{R_{TRISO}} \cdot \frac{M_{FP}}{R_{diff}}$$

Here C_{FPR} is a constant (input parameter CFPRRT) with default value of 3.0.

The fission product isotopes that are released from TRISO particles (fuel regions) are placed in the corresponding structure (SC or TC), proportional to the power density. As an example a pebble model, consisting of graphite/fuel matrix (5 cm radius) and pure graphite (1 cm) is discussed in Volume 3 (HTR-FPR failure tests 1 and 2 and diffusion test). The fission products will subsequently diffuse through the structure and be released from the surface using a sorption/desorption model (section 12.3.5). The presence of a sorption model is required by the code if the HTR-FPR model is used.

The diffusion coefficients need to be defined by the user. The coefficients for main materials and fission products are discussed below. The coefficients were obtained from [216]. The coefficients are defined as:

$$D = D_1 \cdot \exp\left(-Q_1 / RT\right)$$

D diffusion coefficient (m^2/s),
T temperature (K),
R gas constant, equal to 8.3145 (J/mol-K),
D_1 coefficient (m^2/s),
Q_1 activation energy (J/mol),

In SPECTRA, the diffusion coefficients are defined by:

$$D = D_1 \cdot \exp\left(-B_1 / T\right)$$

B_1 coefficient (K)

With one exception, the coefficients, including the values of $B_1 = Q_1 / R$, were obtained as the recommended coefficients from [216]. The exception is Kr in SiC; in this case a correlation covering all experimental data was developed: $D = 5.0\times10^{-5}\times\exp(-50{,}514.0/T)$.

The values are available for the following materials:

- UO_2 as fuel kernel (kernel material in a given fuel region, MTRIRT(0) - Volume 2),
- SiC (first coating material in a given fuel region, MTRIRT(1) - Volume 2),
- PyC (second coating material in a given fuel region, MTRIRT(2) - Volume 2),
- matrix,
- pure graphite.

The values are available for four vapor classes:

- Kr vapor class 1
- Cs vapor class 2
- Sr vapor class 3
- Ag vapor class 12

All the coefficients are listed below.

UO_2 fuel kernel:

	D_1 (m^2/s)	Q_1 (J/mol)	B_1 (K)
• Kr	1.3×10^{-12}	$1.26\times10^{+5}$	15,154.0
• Cs	5.6×10^{-8}	$2.09\times10^{+5}$	25,137.0
• Sr	2.2×10^{-3}	$4.88\times10^{+5}$	58,693.0
• Ag	6.7×10^{-9}	$1.65\times10^{+5}$	19,845.0

SiC:

	D_1 (m^2/s)	Q_1 (J/mol)	B_1 (K)
• Kr	5.0×10^{-5}	$4.20\times10^{+5}$	50,514.0
• Cs	2.8×10^{-4}	$4.20\times10^{+5}$	50,514.0
• Sr	1.2×10^{-9}	$2.05\times10^{+5}$	24,656.0
• Ag	3.6×10^{-9}	$2.15\times10^{+5}$	25,858.0

PyC:

	D_1 (m^2/s)	Q_1 (J/mol)	B_1 (K)
• Kr	2.9×10^{-8}	$2.91\times10^{+5}$	34,999.0
• Cs	6.3×10^{-8}	$2.22\times10^{+5}$	26,700.0
• Sr	2.3×10^{-6}	$1.97\times10^{+5}$	23,694.0
• Ag	5.3×10^{-9}	$1.54\times10^{+5}$	18,522.0

Matrix:

	D_1 (m^2/s)	Q_1 (J/mol)	B_1 (K)
• Kr	2.8×10^{-3}	$2.05\times10^{+5}$	24,656.0
• Cs	3.6×10^{-4}	$1.89\times10^{+5}$	22,731.0
• Sr	1.7×10^{-2}	$2.68\times10^{+5}$	32,233.0
• Ag	1.6	$2.58\times10^{+5}$	31,030.0

Graphite:

	D_1 (m^2/s)	Q_1 (J/mol)	B_1 (K)
• Kr	2.8×10^{-3}	$2.05\times10^{+5}$	24,656.0
• Cs	1.7×10^{-6}	$1.49\times10^{+5}$	17,921.0
• Sr	1.7×10^{-2}	$2.68\times10^{+5}$	32,233.0
• Ag	1.6	$2.58\times10^{+5}$	31,030.0

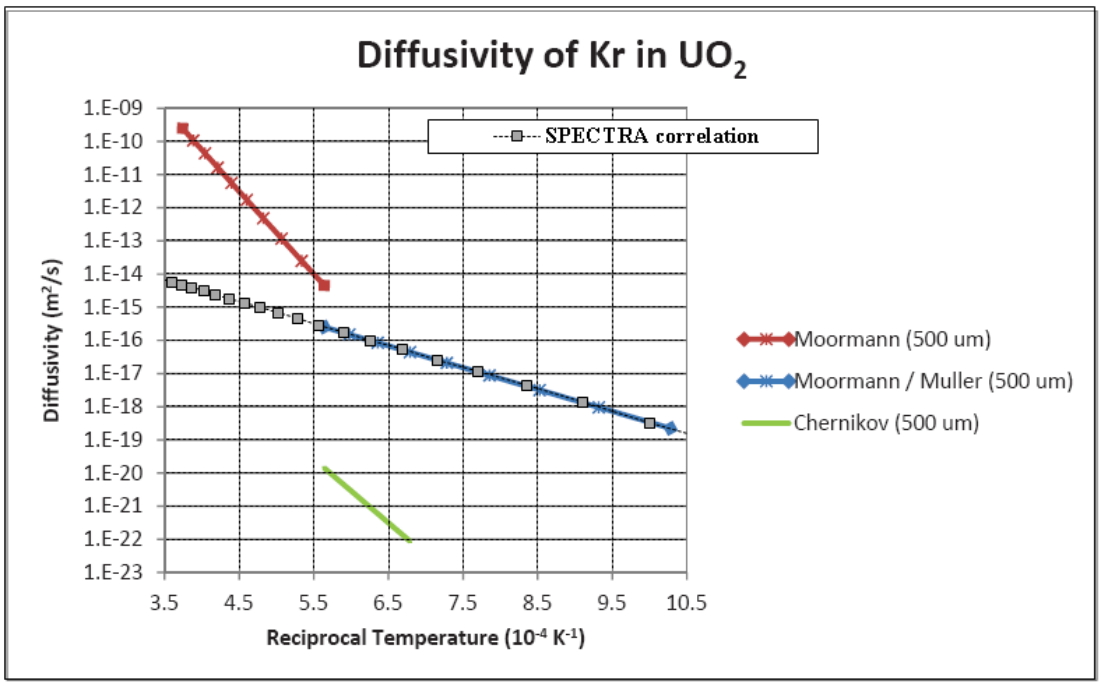
The following graphs are presented below:

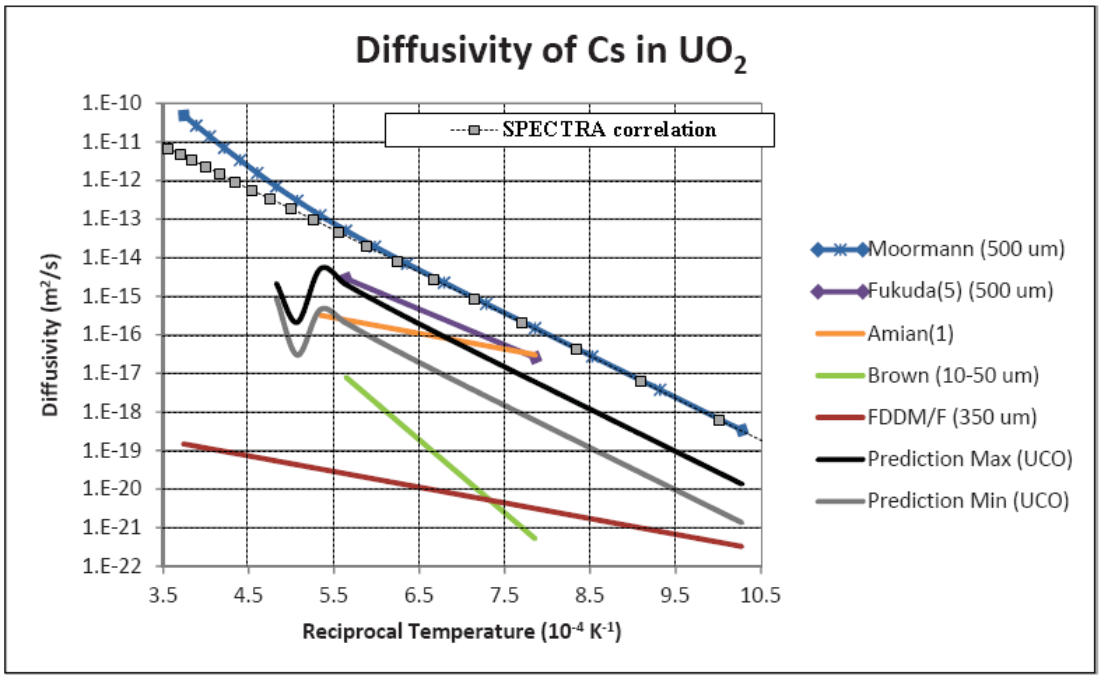
- Diffusion coefficients in UO_2 kernel
- Diffusion coefficients in SiC
- Diffusion coefficients in PyC
- Diffusion coefficients in matrix
- Diffusion coefficients in graphite

Figures below show the diffusion coefficients graphs copied from [216], with the lines obtained with SPECTRA correlation superimposed on the original graphs.

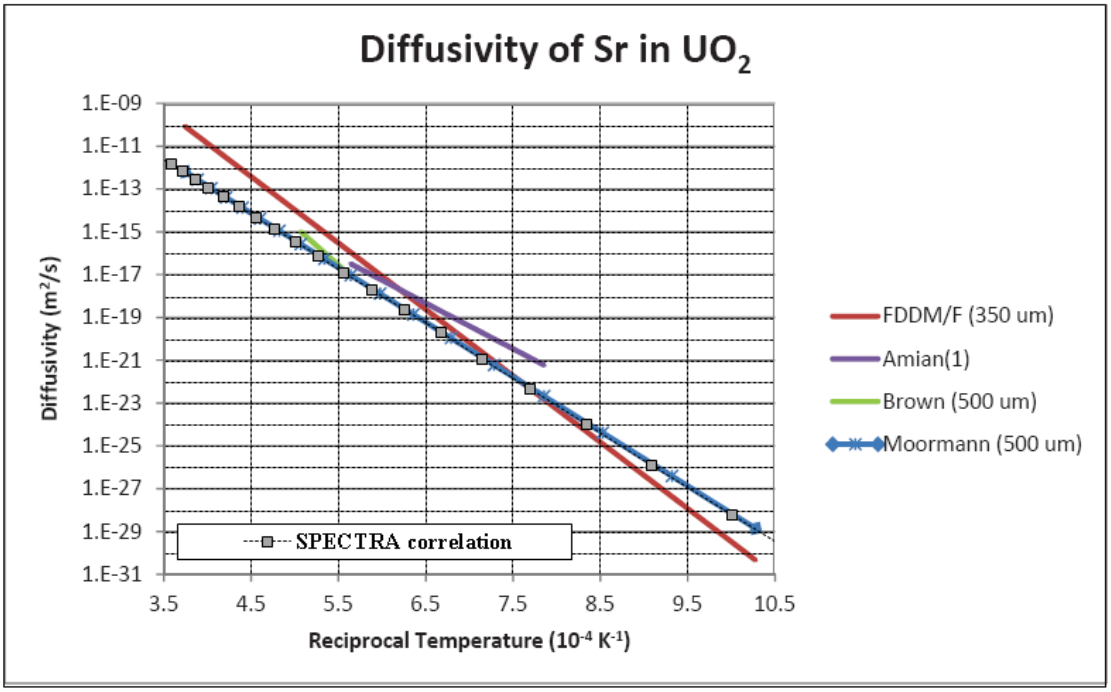
It should be noted that Iodine (vapor class 4) is practically not diffusing. Measurements showed that "I-131 measurements were barely above detection limit" [215], sec. 4.1.1.3.2. Observed released fractions from the graphite is below 10–8 even at temperatures as high as 1600 - 1800°C - [215], sec. 4.1.1.3.2.

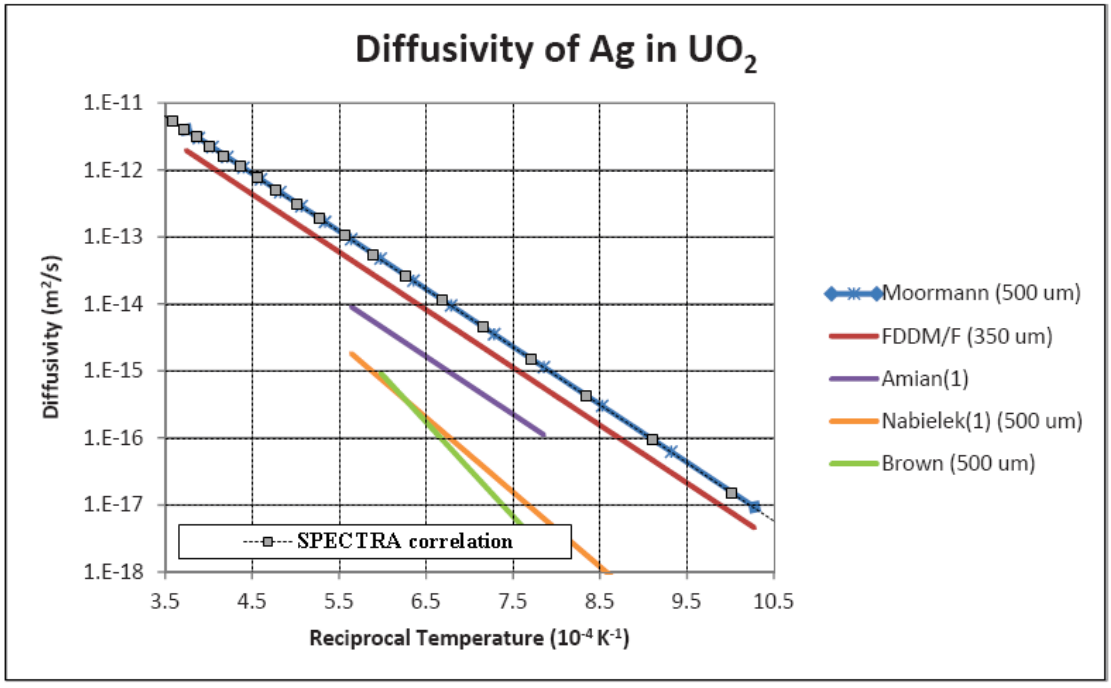
Diffusion coefficients in UO_2 kernel



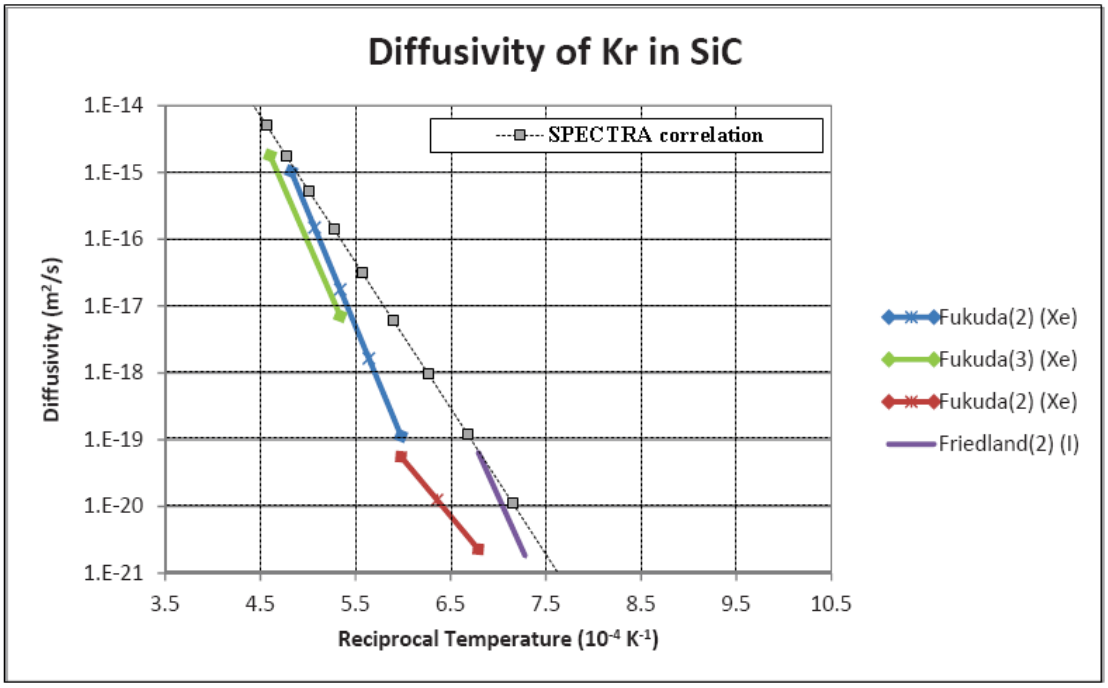


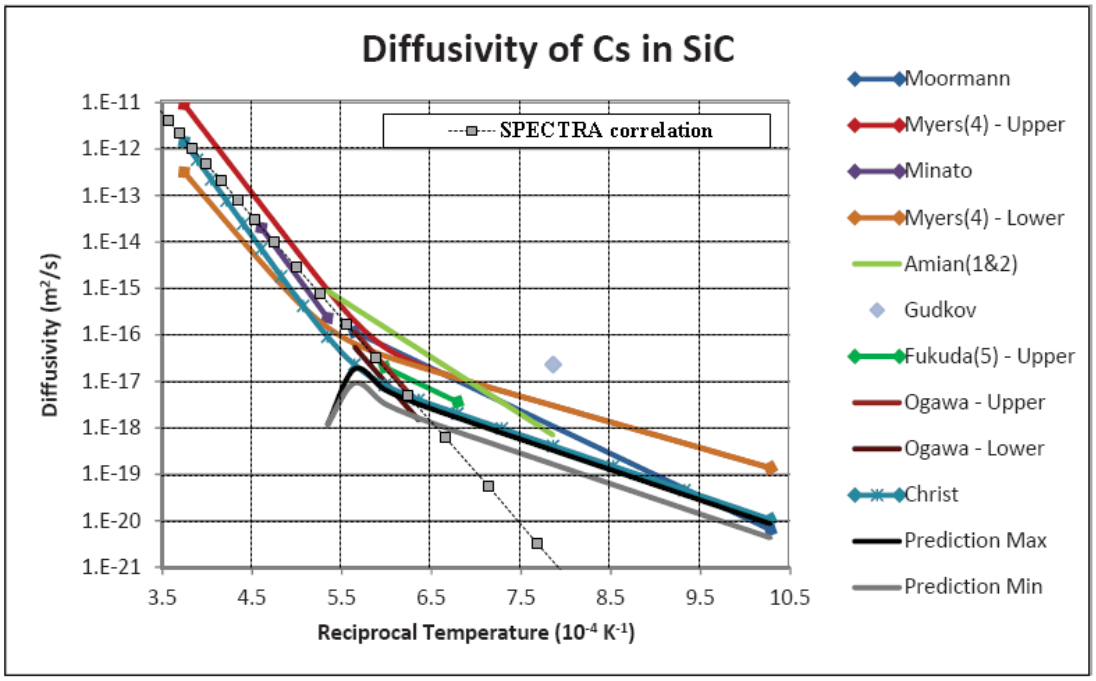
Diffusion coefficients in UO_2 kernel (continued)



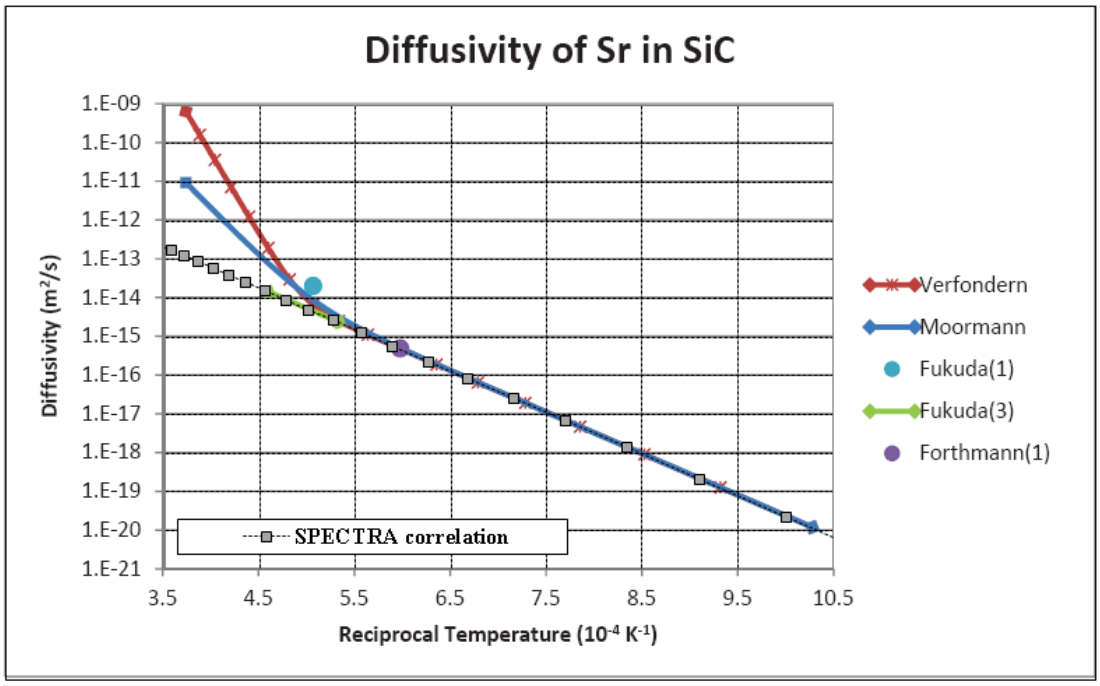


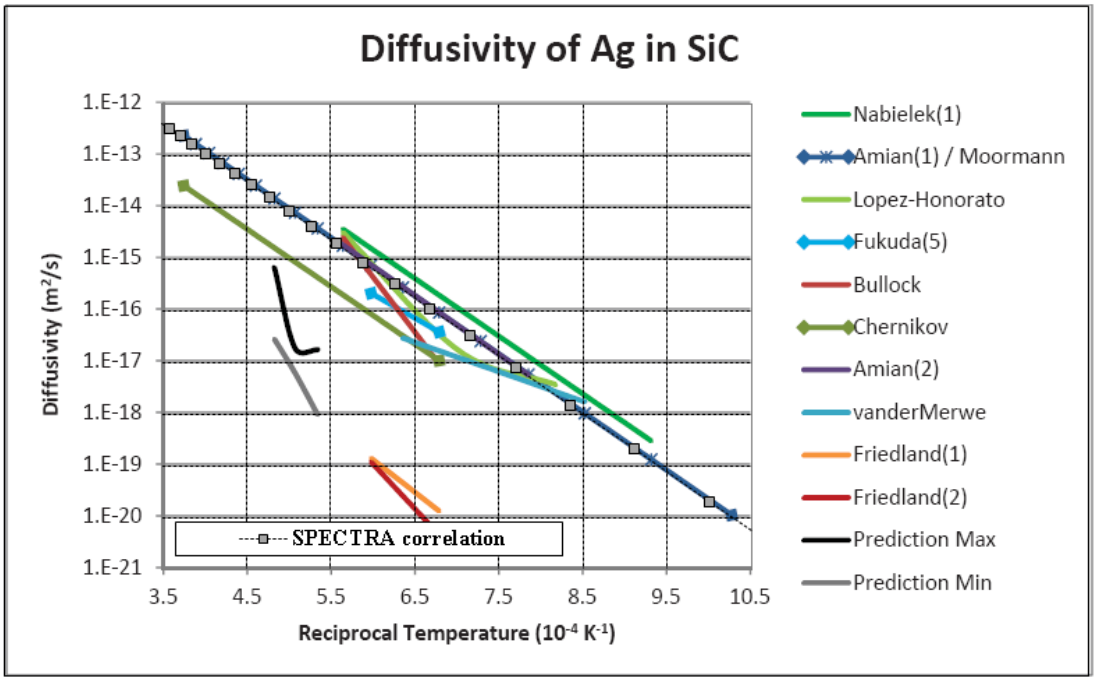
Diffusion coefficients in SiC



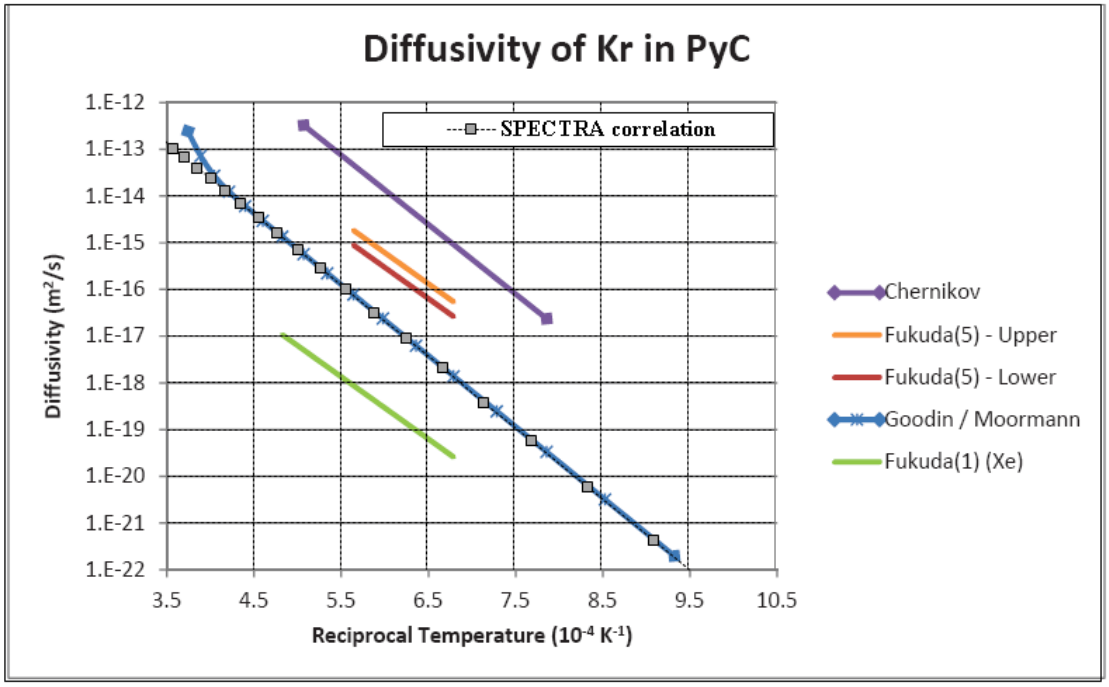


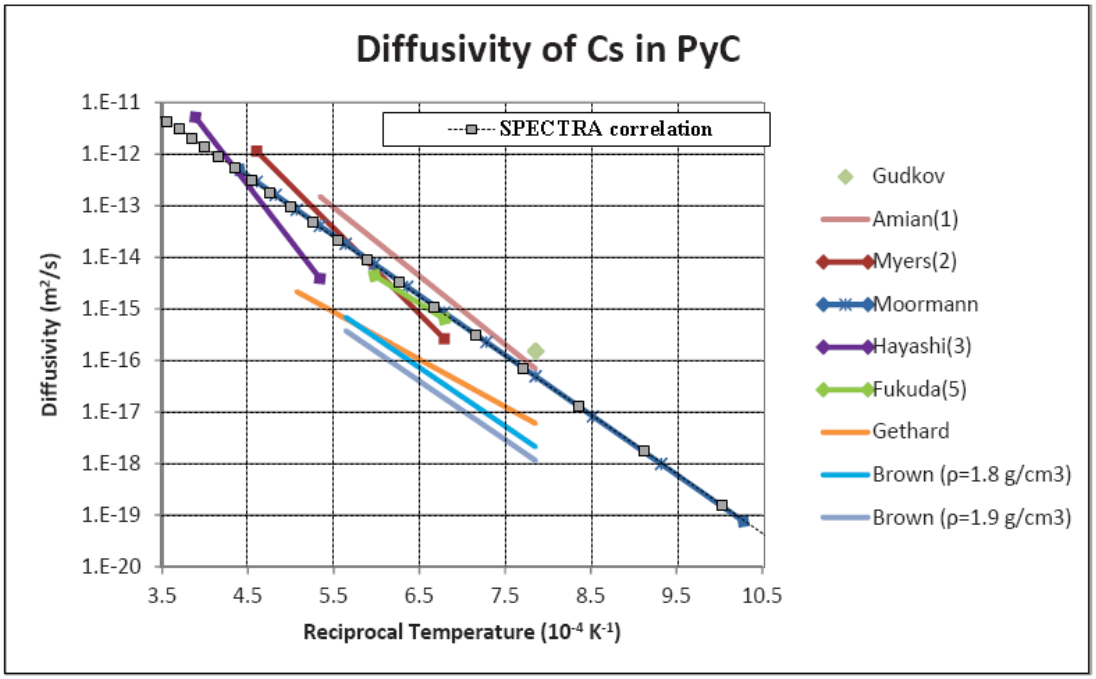
Diffusion coefficients in SiC (continued)



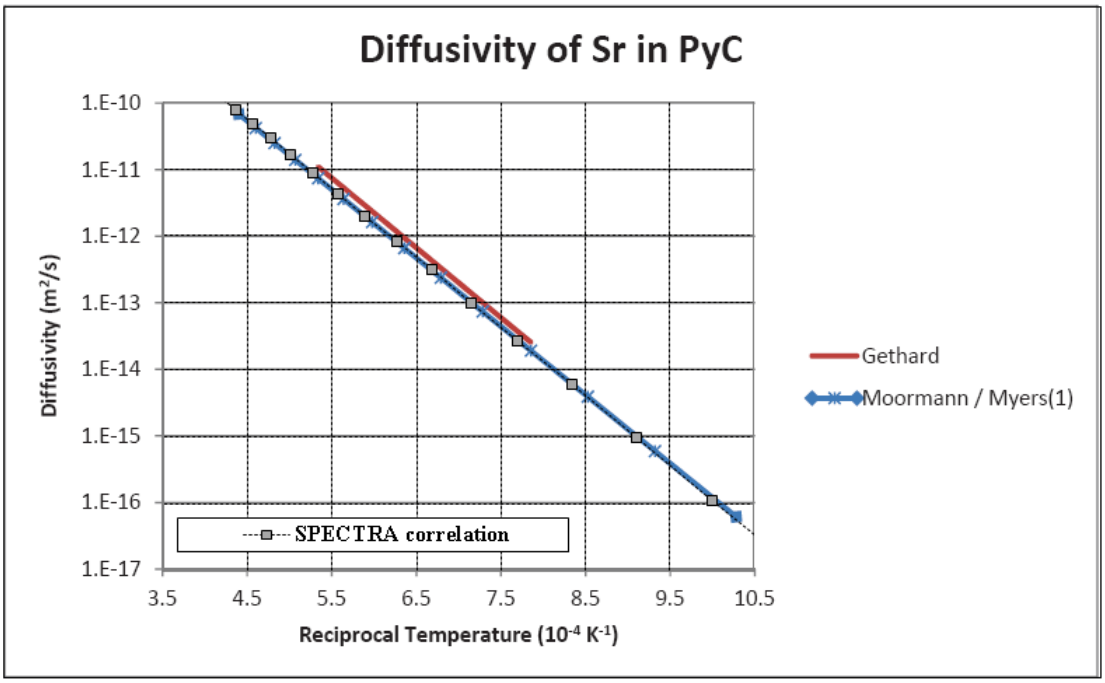


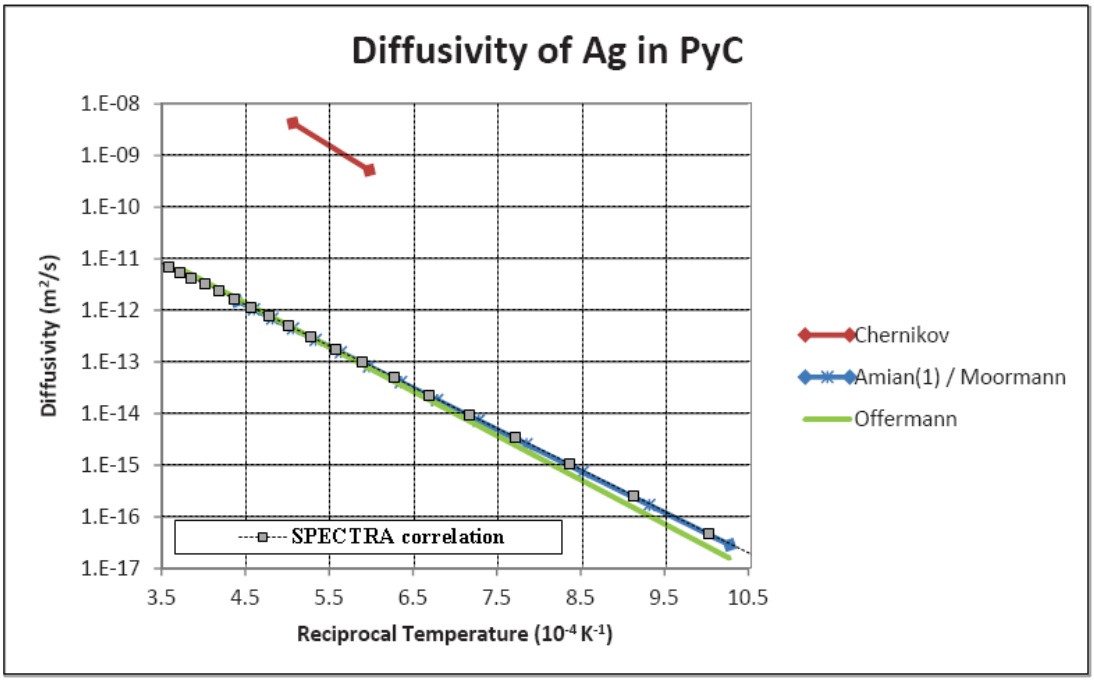
Diffusion coefficients in PyC



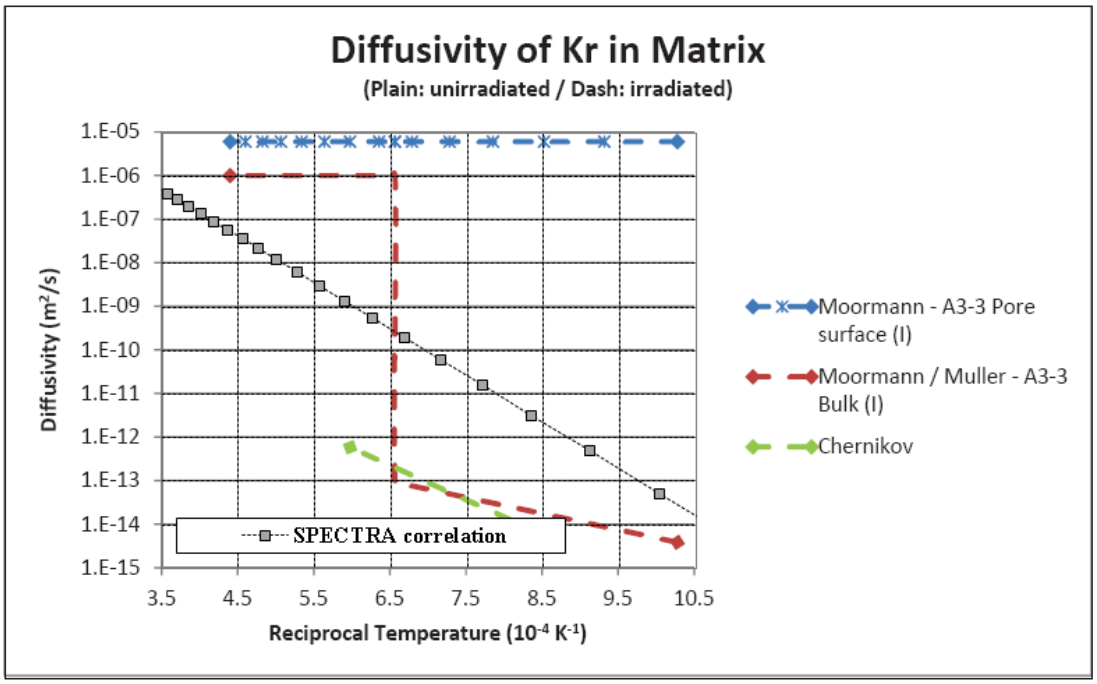


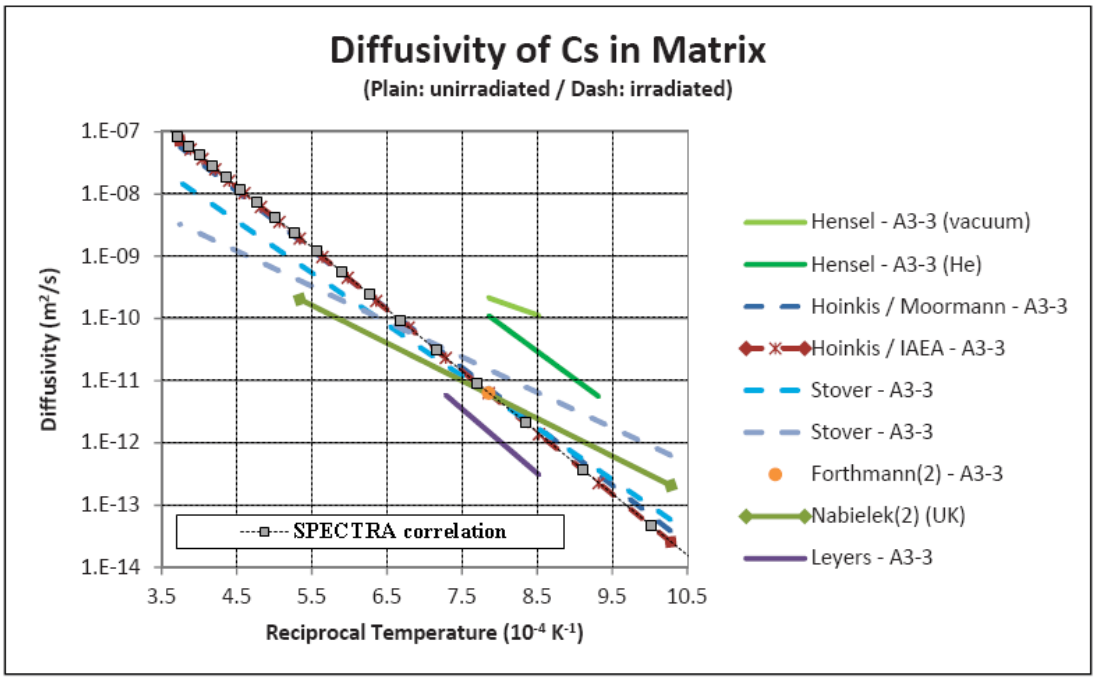
Diffusion coefficients in PyC (continued)



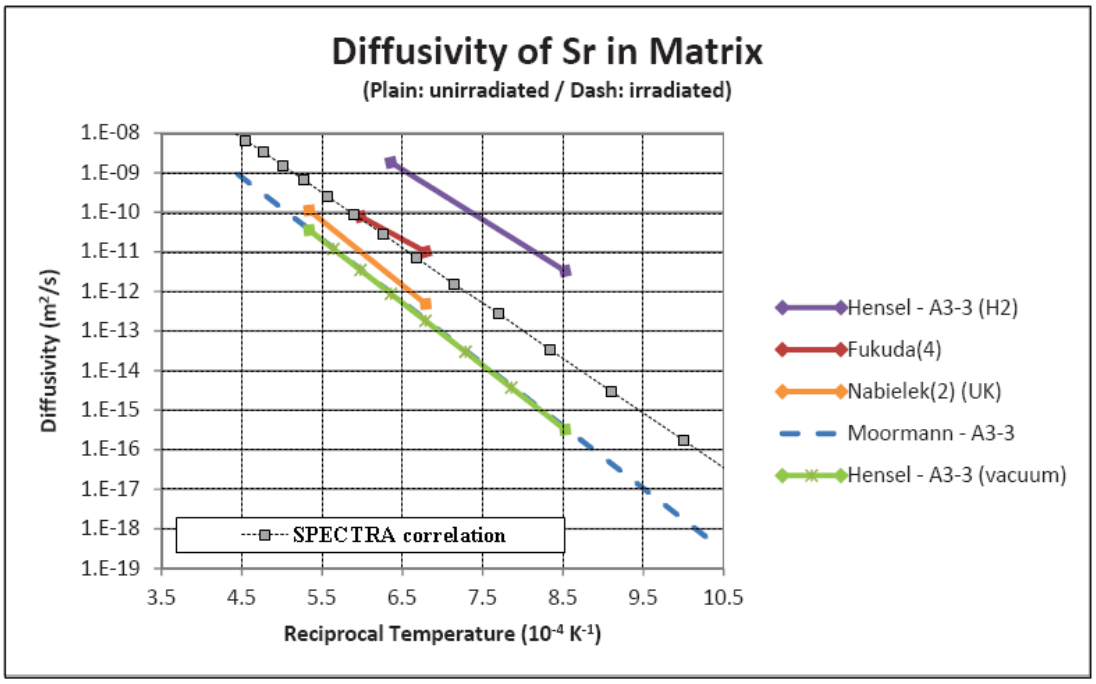


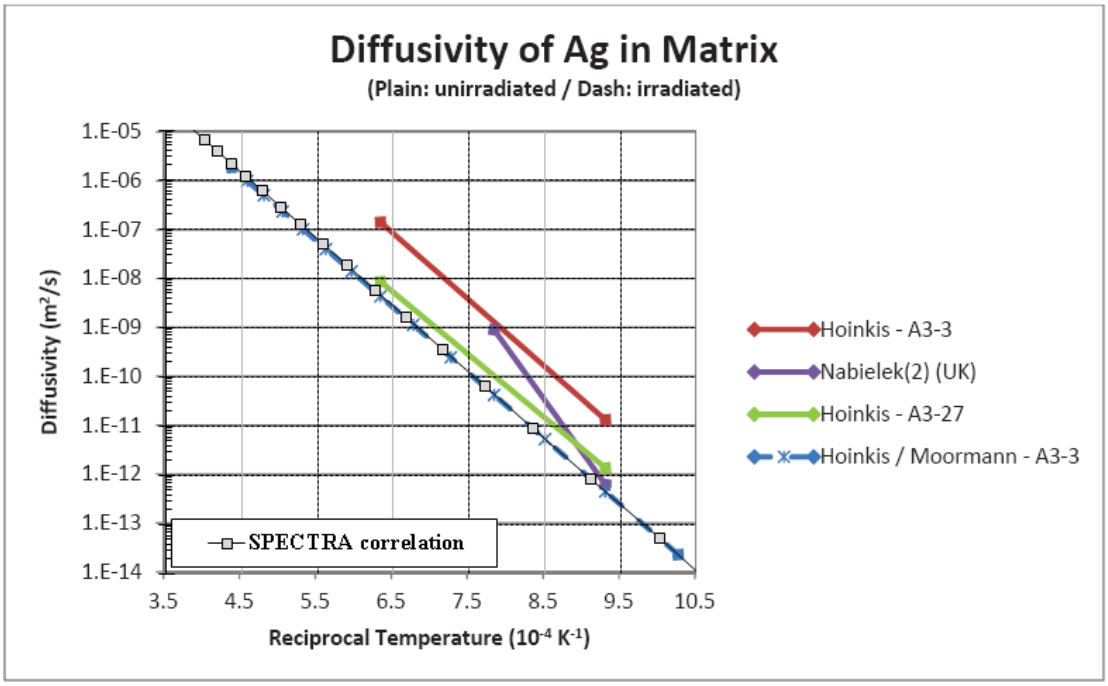
Diffusion coefficients in Matrix



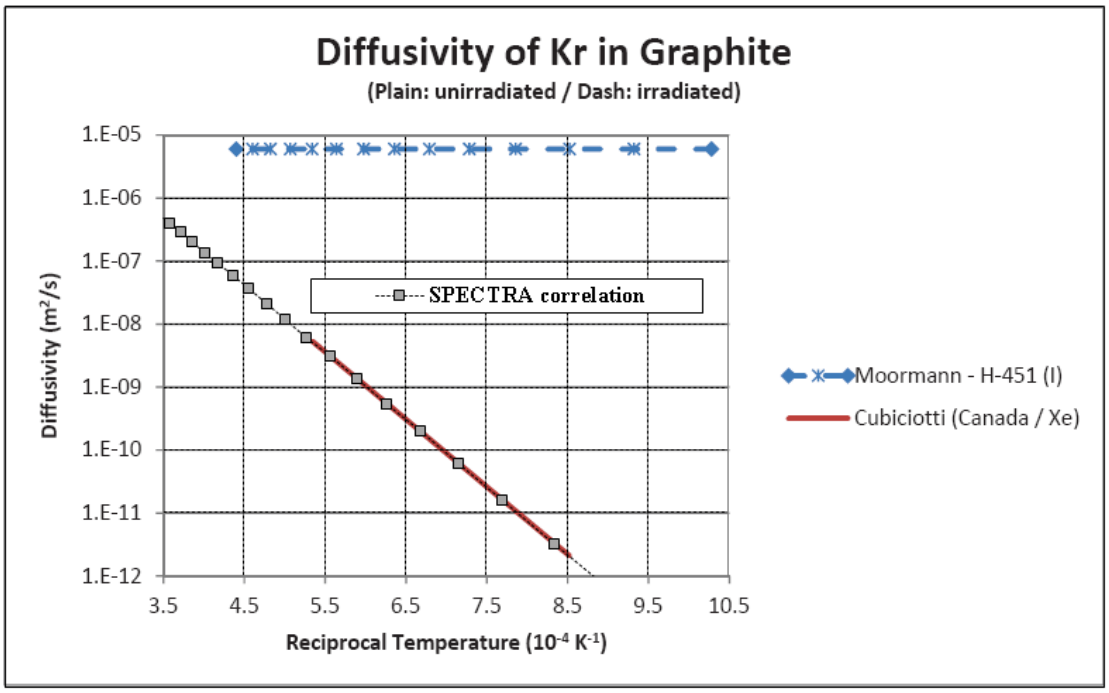


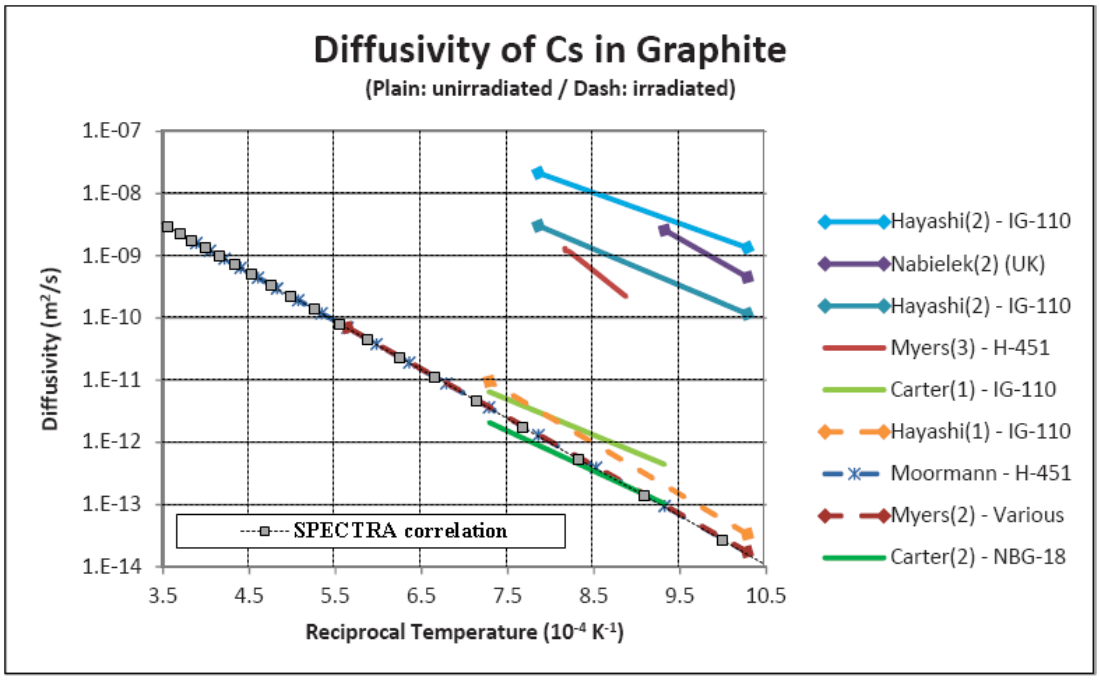
Diffusion coefficients in Matrix (continued)



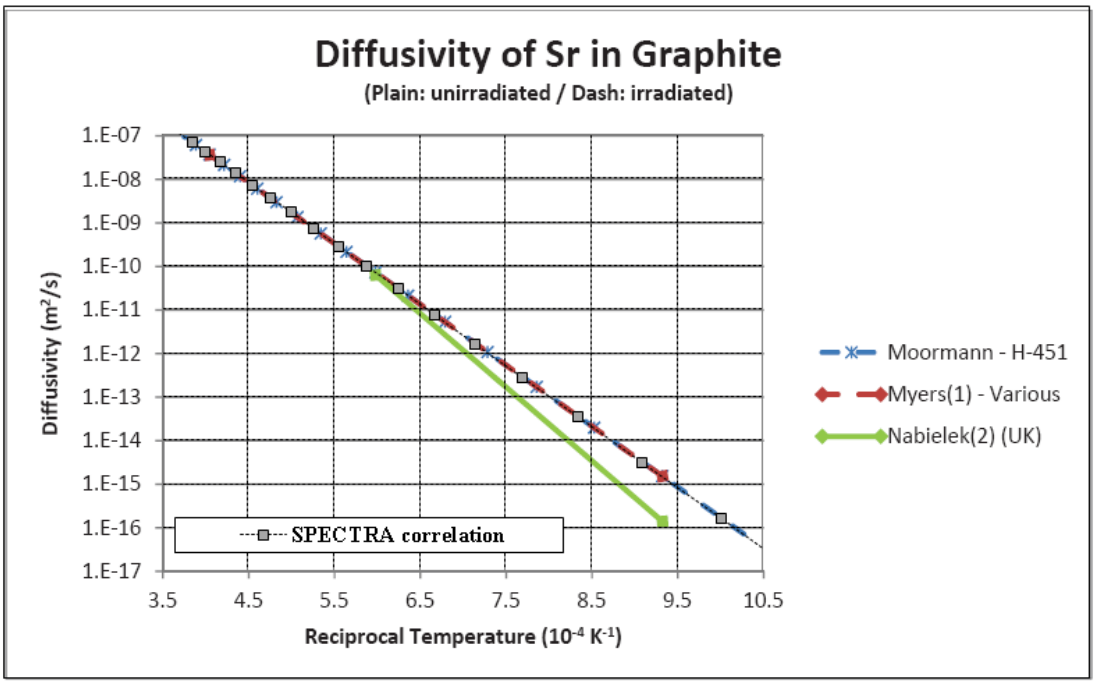


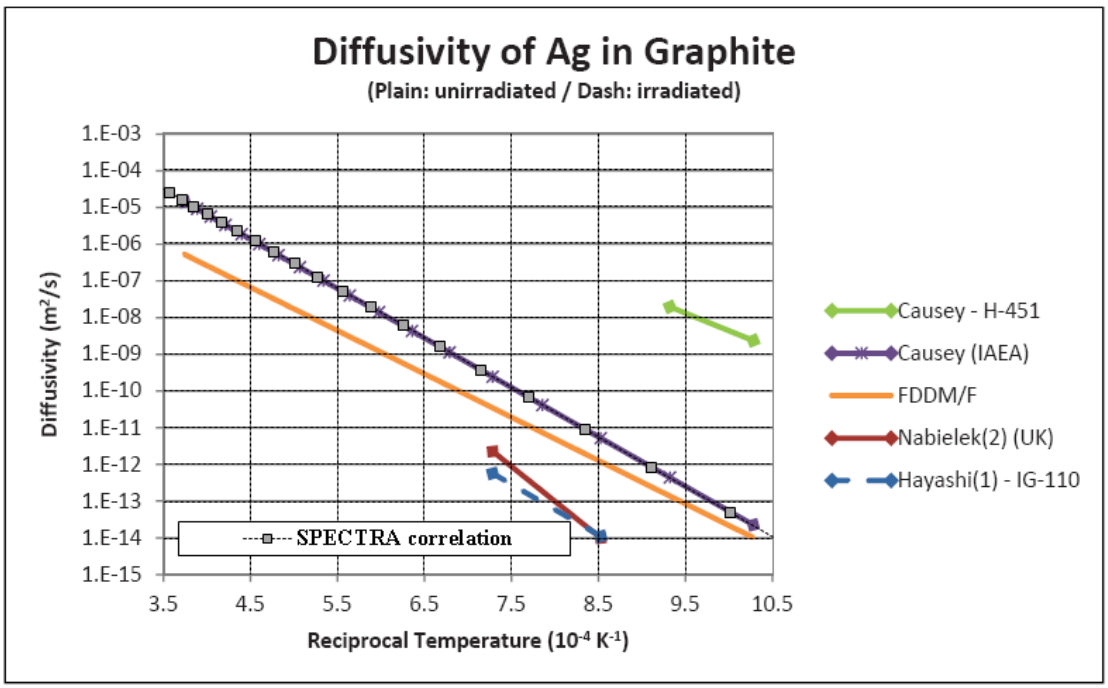
Diffusion coefficients in Graphite





Diffusion coefficients in Graphite (continued)





12.3.3 Fission Product Vapors

Each fission product must be associated with one of the vapor class. The vapor classes and the release classes are independent, and therefore the same isotope may be associated with a vapor class and a release class with a different number (or no release class at all, if no release is calculated or the release rate is defined using a Control Function). The fission product classes are shown in Table 12-6. The coefficients shown in Table 12-7 and Table 12-12 are specified based on reference [46]. The class numbering from reference [46] was preserved; therefore the release class and the vapor class for a given isotope have the same number. Compared to the release classes one more vapor class is added: class 13, representing CsI. Upon release all Cs and I that are simultaneously released during one time step are assumed to be combined into CsI.

The general formula used to compute the fission product vapor pressure is [46]:

$$\log_{10}(P'[mmHg\ ]) = -\frac{A'}{T} + B' + C'\log_{10}(T)$$

where: P - saturation pressure, (mm Hg)
A', B', C' coefficients,
T - temperature, (K)

In SPECTRA the SI units are used; therefore the formula is converted to:

$$\log_{10}(P) = -\frac{A}{T} + B + C\log_{10}(T)$$

where: P - saturation pressure, (Pa)
A, B, C coefficients,
T - temperature, (K)

Conversion to the SI units is quite simple: $B = B' + \log_{10}(10^5/750) = B' + 2.125$ (A and C remain unchanged: $A=A'$, $C=C'$). The fission product vapor equations, as applied in SPECTRA (SI units) for 13 vapor classes (the first 12 classes the same as the release classes, and the 13-th class of CsI), are shown in Table 12-12. The coefficients for all classes except for the Class 1 were obtained from [46]. Class 1 (noble gases) is always a vapor; therefore the coefficients were set to give a constant vapor pressure of 10^{10} Pa. (B=10.0, A=C=0.0).

The limiting temperatures for the first sets were modified compared to the data from [46]. They were set to the values for which $P = 10^{-10}$ Pa. This was done to minimize discontinuities at this temperature (see Volume 3). The original limiting temperatures from reference [46] are printed in Table 12-12 in brackets, behind the actually used temperatures.

The vapor pressure is defined using up to three sets of coefficients; each set has its lower limit temperature. For the class 2, 4, and 13 multiple sets are used. The calculation of the vapor pressure is performed as follows:

Table 12-12 Fission product vapor pressures, SI units.

Class		A		B		C		T-lim	
1:	log(P) = -	0.0 / T	+	10.00	-	0.00 log(T) ,	T >	273.0	
2:	log(P) = -	9400.0 / T	+	23.71	-	3.75 log(T) ,	T >	390.0	(600.0)
	log(P) = -	6870.8 / T	+	10.12	-	0.00 log(T) ,	T >	1553.0	
3:	log(P) = -	7836.0 / T	+	8.57	-	0.00 log(T) ,	T >	422.0	(1000.0)
4:	log(P) = -	3578.0 / T	+	19.84	-	2.51 log(T) ,	T >	273.0	(298.0)
	log(P) = -	3205.0 / T	+	25.79	-	5.18 log(T) ,	T >	387.0	
	log(P) = -	2176.9 / T	+	9.77	-	0.00 log(T) ,	T >	457.0	
5:	log(P) = -	13940.0 / T	+	25.63	-	3.52 log(T) ,	T >	534.0	(298.0)
6:	log(P) = -	33200.0 / T	+	12.73	-	0.00 log(T) ,	T >	1460.0	(1500.0)
7:	log(P) = -	32800.0 / T	+	11.80	-	0.00 log(T) ,	T >	1504.0	(1500.0)
8:	log(P) = -	21570.0 / T	+	10.87	-	0.00 log(T) ,	T >	1032.0	(1500.0)
9:	log(P) = -	21800.0 / T	+	10.80	-	0.00 log(T) ,	T >	1046.0	(1000.0)
10:	log(P) = -	32110.0 / T	+	14.00	-	0.00 log(T) ,	T >	1338.0	(1500.0)
11:	log(P) = -	13730.0 / T	+	10.55	-	0.00 log(T) ,	T >	666.0	(1000.0)
12:	log(P) = -	15400.0 / T	+	10.28	-	0.00 log(T) ,	T >	758.0	(1000.0)
13:	log(P) = -	10420.0 / T	+	21.82	-	3.02 log(T) ,	T >	436.0	(600.0)
	log(P) = -	9678.0 / T	+	22.48	-	3.52 log(T) ,	T >	894.0	
	log(P) = -	7303.9 / T	+	9.71	-	0.00 log(T) ,	T >	1553.0	

For temperatures below the limiting temperature for the first set zero vapor pressure is used. For temperatures between the limit for the first set and the second set, the first set of coefficients is used, etc.

- $T < T_{lim}(1)$ $P(T) = 0.0$
- $T_{lim}(1) < T < T_{lim}(2)$ Use first set
- $T_{lim}(2) < T < T_{lim}(3)$ Use second set
- $T > T_{lim}(3)$ Use third set

The temperature limits for the default classes with multiple sets (class 2, 4, and 12 – see Table 12-12) were selected in order to provide continuous *P(T)* [46]. This means a limiting temperature between data sets is defined at the points where the lines defined by the two sets are crossing.

In such case, i.e. if the limiting temperatures are selected to provide a smooth *P(T)*, there is no need to perform any interpolation between the ranges of applicability of different sets. Therefore, if the built-in data was to be used solely, then the computational scheme would not need any interpolations. However, since coefficients of the existing sets may be re-defined by the user, and new sets may be defined, there is in general a need to provide a safe procedure to ensure continuity of *P(T)*. Therefore an interpolation scheme has been introduced as follows:

- $T_{lim}(1)–10.0 < T < T_{lim}(1)+10.0$ interpolation between zero and the first set.
- $T_{lim}(2)–10.0 < T < T_{lim}(2)+10.0$ interpolation between the first set and the second set.
- $T_{lim}(3)–10.0 < T < T_{lim}(3)+10.0$ interpolation between the second set and the third set.

The interpolated range was set at ±10.0 K. Note that a requirement is put on the limiting temperatures such that they are at least 50.0 K apart (see Volume 2). In the interpolation zones a linear interpolation is performed for $X = A/T + B + C \log_{10}(T)$.

As an example of the interpolating scheme a simple function has been built, using three sets that provide a constant value within each set:

- Set 1: $A = 0.0$, $B = 1.0$, $C = 0.0$, $T_{lim} = 350.0$ $P(T)$=10.0
- Set 2: $A = 0.0$, $B = 1.3$, $C = 0.0$, $T_{lim} = 400.0$ $P(T)$=19.9
- Set 3: $A = 0.0$, $B = 1.5$, $C = 0.0$, $T_{lim} = 450.0$ $P(T)$=31.6

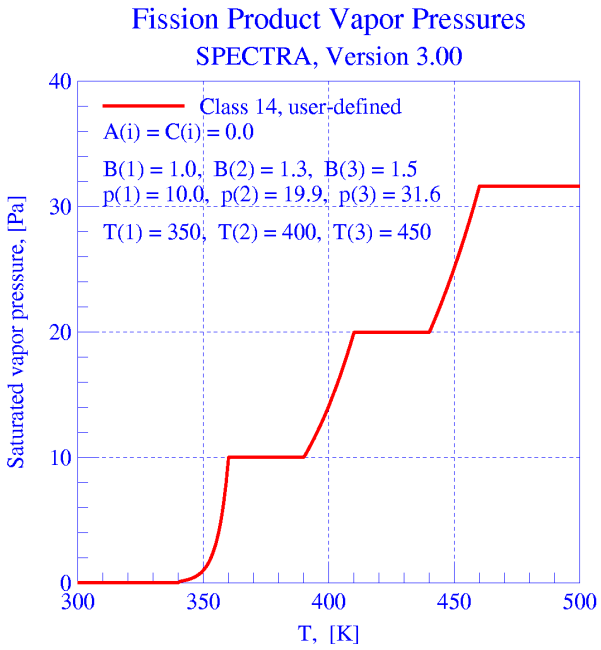


Figure 12-75 Example of a user-defined class, an illustration of the interpolation scheme.

Resulting pressure is shown in Figure 12-75. The interpolation is linear with respect to the exponent, X, so the actual vapor pressure, which is equal to 10^X, is not linear. The function $P(T)$ is however continuous. Summarizing, the applied interpolation scheme assures that the vapor pressure function is always continues, independently of the value of the coefficients and the limiting temperatures, which are entered by the user.

The vapor pressures for all built-in classes, except for the class 1 are shown in Figure 12-76 and Figure 12-77. Class 1 (Xe, Kr) is assumed to be always a vapor (the saturation pressure is infinitely large). The only other class that has a large vapor pressure at the room temperature is class 4 (I, Br) - Figure 12-76. All other classes have their saturation pressures equal to zero at the temperatures below ~400 K - Figure 12-76.

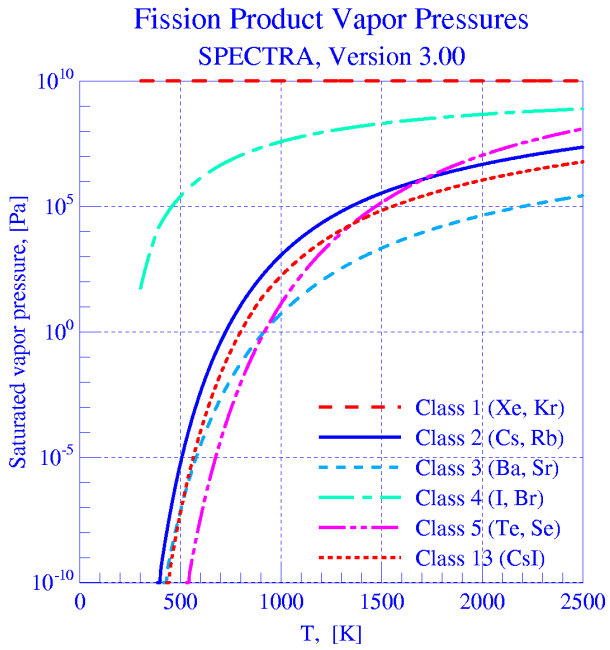


Figure 12-76 Saturation pressures for built-in vapor classes 2 – 5, and 13.

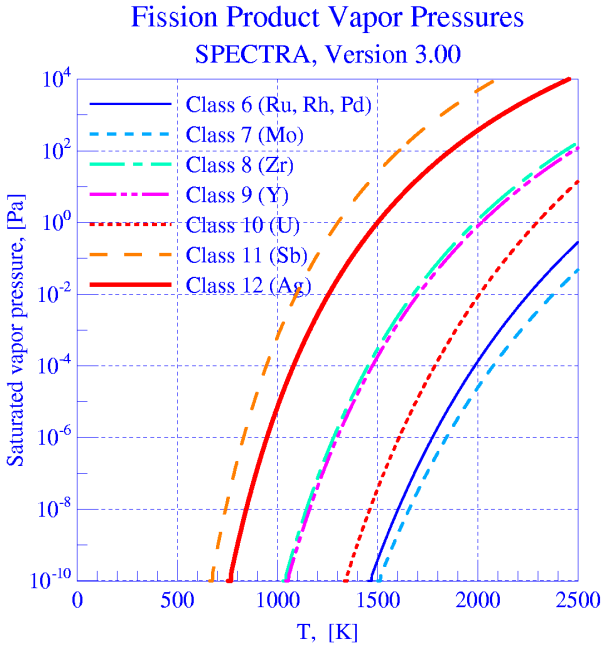


Figure 12-77 Saturation pressures for built-in vapor classes 6 – 12.

12.3.4 Fission Product Chemistry

Fission product chemistry is not implemented in the current code version. The equations shown below will be implemented in a future version.

12.3.4.1 Chemical Reaction

A simplified model is considered, with a chemical reaction of two elements:

$$nA + mB \leftrightarrow A_n B_m$$

The elements *A* and *B* are reacting to form the compound A_nB_m. All isotopes of the element *A* and *B* should be taken into account in defining the reaction.

The reaction kinetics is given by:

$$\frac{d[A_n B_m]}{dt} = R_{for}(T)\cdot[A]\cdot[B] - R_{rev}(T)\cdot[A_n B_m]$$

[*A*] concentration of the element *A*, (kmol/m^3)
[*B*] concentration of the element *B*, (kmol/m^3)
[A_nB_m] concentration of the compound A_nB_m, (kmol/m^3)
R_{for} forward reaction rate, (m^3/kmol-s)
R_{rev} reverse reaction rate, (m^3/kmol-s)

In general the forward reaction term should be written as $R_{for} \cdot (A)^x \cdot (B)^y$. Here a simplified version is considered, where it is assumed that x = y = 1.0. The reaction rates of the elements follow from the reaction stoichiometry: $d(A)/dt = -n\, d(A_nB_m)/dt$ and $d(B)/dt = -m\, d(A_nB_m)/dt$. Therefore:

$$\frac{1}{n}\cdot\frac{d[A]}{dt} = -R_{for}(T)\cdot[A]\cdot[B] + R_{rev}(T)\cdot[A_n B_m]$$
$$\frac{1}{m}\cdot\frac{d[B]}{dt} = -R_{for}(T)\cdot[A]\cdot[B] + R_{rev}(T)\cdot[A_n B_m]$$

The reaction rates for the forward and the reverse reactions are given by:

$$R_{for}(T) = R_{0,\,for}\cdot\exp\left(-\frac{A_{R,\,for}}{T}\right)$$
$$R_{rev}(T) = R_{0,rev}\cdot\exp\left(-\frac{A_{R,rev}}{T}\right)$$

$R_{0,\,for}$ forward reaction first coefficient, (m^3/kmol-s)
$A_{R,\,for}$ forward reaction second coefficient, (K) (activation energy divided by the gas constant)
$R_{0,\,rev}$ reverse reaction first coefficient, (m^3/kmol-s)
$A_{R,\,rev}$ reverse reaction second coefficient, (K) (activation energy divided by the gas constant)

The equations are solved in the following way. The non-linear reaction terms are linearized. The linearized product of reactant concentrations are written as:

$$[A]\otimes[B] \equiv \frac{1}{2}([A]^0 \cdot [B] + [A]\cdot[B]^0)$$

Here $[A]^0$ and $[B]^0$ are the beginning of time step values. As a result one obtains a linear set of equations, which is then easily solved by one of the standard solver (see section 17.4). A large amount of test runs performed showed that this method provides a fast and stable solution even with very large time steps.

After linearization the finite difference version of the reaction kinetics equation is:

$$\frac{[A_nB_m]}{\Delta t} - \frac{[A_nB_m]^0}{\Delta t} = \frac{R_{for}}{2}\cdot[A]^0\cdot[B] + \frac{R_{for}}{2}\cdot[A]\cdot[B]^0 - R_{rev}\cdot[A_nB_m]$$

$$\frac{[A]}{n\Delta t} - \frac{[A]^0}{n\Delta t} = -\frac{R_{for}}{2}\cdot[A]^0\cdot[B] - \frac{R_{for}}{2}\cdot[A]\cdot[B]^0 + R_{rev}\cdot[A_nB_m]$$

$$\frac{[B]}{m\Delta t} - \frac{[B]^0}{m\Delta t} = -\frac{R_{for}}{2}\cdot[A]^0\cdot[B] - \frac{R_{for}}{2}\cdot[A]\cdot[B]^0 + R_{rev}\cdot[A_nB_m]$$

After easy transformations, one obtains a matrix equation of the form:

$$\begin{vmatrix} 1+\frac{R_{for}n\Delta t}{2}[B]^0 & \frac{R_{for}n\Delta t}{2}[A]^0 & -R_{rev}n\Delta t \\ \frac{R_{for}m\Delta t}{2}[B]^0 & 1+\frac{R_{for}m\Delta t}{2}[A]^0 & -R_{rev}m\Delta t \\ -\frac{R_{for}\Delta t}{2}[B]^0 & -\frac{R_{for}\Delta t}{2}[A]^0 & 1+R_{rev}\Delta t \end{vmatrix} \times \begin{vmatrix} [A] \\ [B] \\ [A_nB_m] \end{vmatrix} = \begin{vmatrix} [A]^0 \\ [B]^0 \\ [A_nB_m]^0 \end{vmatrix}$$

The above matrix equation is solved using Gauss-Jordan elimination method, to find the end-of-time step concentrations [A], [B], and [A_nB_m].

It should be noted that in equilibrium the concentrations fulfill the following relation:

$$\frac{[A]\cdot[B]}{[A_nB_m]} = \frac{R_{rev}(T)}{R_{for}(T)}$$

This relation may serve as an easy check of the calculated values for equilibrium conditions - see next section.

12.3.4.2 Example of a Chemical Reaction - CsI

As an example the following chemical reaction is considered:

$$Cs + I \leftrightarrow CsI$$

The elements *Cs* and *I* are reacting to form the compound *CsI*. All built-in isotopes of the element *Cs* and *I* are taken into account in defining the reaction. That means :

- Cs-133, Cs-134, Cs-135, Cs-136, Cs-137
- I-131, I-132, I-133, I-134, I-135

The reaction kinetics is given by:

$$\frac{d[CsI]}{dt} = R_{for}(T)\cdot[Cs]\cdot[I] - R_{rev}(T)\cdot[CsI]$$

[*Cs*] concentration of *Cs*, (kmol/m^3)
[*I*] concentration of *I*, (kmol/m^3)
[*CsI*] concentration of the compound *CsI*, (kmol/m^3)
R_{for} forward reaction rate, (m^3/kmol-s)
R_{rev} reverse reaction rate, (1/s)

The reaction rates of the elements follow from the reaction stoichiometry: $d[Cs]/dt = n\, d[CsI]/dt$ and $d[I]/dt = m\, d[CsI]/dt$. Therefore:

$$\frac{1}{n}\cdot\frac{d[Cs]}{dt} = -R_{for}(T)\cdot[Cs]\cdot[I] + R_{rev}(T)\cdot[CsI]$$
$$\frac{1}{m}\cdot\frac{d[I]}{dt} = -R_{for}(T)\cdot[Cs]\cdot[I] + R_{rev}(T)\cdot[CsI]$$

The reaction rates for the forward and the reverse reactions are given by:

$$R_{for}(T) = R_{0,for}\cdot\exp\left(-\frac{A_{R,for}}{T}\right)$$
$$R_{rev}(T) = R_{0,rev}\cdot\exp\left(-\frac{A_{R,rev}}{T}\right)$$

$R_{0,for}$ forward reaction first coefficient, (m^3/kmol-s)
$A_{R,for}$ forward reaction second coefficient, (K) (activation energy divided by the gas constant)
$R_{0,rev}$ reverse reaction first coefficient, (m^3/kmol-s)
$A_{R,rev}$ reverse reaction second coefficient, (K) (activation energy divided by the gas constant)

12.3.5 Sorption of Fission Product Vapors on Surfaces

Discussion of the sorption models available in SPECTRA is preceded by a short overview of the sorption phenomena, given in section 12.3.5.1. An overview of the sorption models available in the SPECTRA code is given in section 12.3.5.2. Detailed description of the available models is given in sections 12.3.5.3, 12.3.5.4, and 12.3.5.5. The sorption models may be used in combination with diffusion models. In such case the diffusion models calculate diffusion of the adsorbed material throughout the solid structures. Diffusion in the 1-D and the 2-D structures is described in section 12.3.5.6. Finally, section 12.3.5.7 presents a summary of the sorption models.

12.3.5.1 Overview of Sorption Phenomena

A short description of the sorption phenomena, is given below. Broadly speaking adsorption can be described as the accumulation of gas molecules on a surface. Absorption, on the other hand, is a term usually reserved for accumulation of gas molecules inside a solid material. In practice, however, we usually find no sharp boundary between adsorption and absorption and the term sorption has been introduced in 1909 by W. McBain, to cover phenomena that fall in between these two extremes or contain elements of both.

A typical example of a sorption phenomenon is when molecules impinge on a surface layer with pores interconnected into the interior. Gas molecules can diffuse down the pores and adsorb onto the pore walls inside the bulk of the material. So, even though this strictly speaking is a pure adsorption process it also conforms to the definition of absorption in as much as molecules are absorbed into the material like water into a sponge. The adsorption process is slowed down by the time bottle-neck imposed by diffusion. The process appears to be much slower than plane surface adsorption and the number of molecules adsorbed per unit area of the adsorbent is also much higher. The nature of the attractive interaction between the adsorbed molecules and the adsorbent, however, to a first approximation, remains the same as for a smooth surface. This simple example shows that, although the activation energy for adsorption may be the same for the porous layer and the smooth surface, effective timescales to achieve adsorption equilibrium may differ appreciably.

Apart from physical or chemical adsorption onto a surface, molecules may also dissolve in the alloy or oxide layer at the surface. The result is a solid solution, defined as a state where dissolved atoms are randomly distributed through the matrix of the host material. They may be present as interstitial atoms or may take up spaces where other atoms were displaced. Atoms in solid solution are free to diffuse through the host crystal and can evaporate from the solid at high enough temperature and come in equilibrium with its own vapor. The situation is analogous to a gas in equilibrium with its dissolved phase in a liquid. When the absorbed atoms are incorporated into a regular array rather than a random distribution inside the host crystal the final state resembles a new chemical compound more closely than a solid solution. Exchange between the trapped atoms and its free gas is then described by different characteristic evaporation energy per mole as for solid solution.

For adsorption of gas molecules onto a solid surface some or all of the above phenomena may play a role. The mass of gas absorbed in a liquid (or dissolved in it) at low pressure is directly proportional to the pressure.

This well-known relationship is known as Henry's law and may be expected to hold at low gas pressures also for certain adsorbent-adsorbate systems where the adsorbent is a solid surface and the adsorbate is a gas. In practice though, most gas-solid systems show non-linear relationship between the amount of gas adsorbed and the gas pressure. A relationship, or plot, of the amount of gas adsorbed versus the gas pressure is known as an absorption isotherm because it is defined at a constant temperature. At a different temperature a new isotherm needs to be established.

The reason why most gas-solid systems show non-Henrian behavior is most probably due to the non-uniformity of their surfaces. Gas molecules are caught preferentially in cracks, pits, along ledges, distorted crystal areas and regions containing unsaturated or "dangling" bonds. There exists a whole spectrum of these sites, each with its own characteristic bonding energy. The first gas molecules are caught by the stronger attractors. This presumably happens because weakly bound molecules can easily diffuse across the surface until they become trapped in a well and also because almost all molecules that impinge near such sites become trapped. So, the deep wells get filled first. After this the spots with less attraction are filled, and so on. A solid surface therefore behaves like a fast collection of "different liquids", each with Henrian behavior. The isotherms of these inherently different trap site populations differ quantitatively and qualitatively.

For instance, suppose we have a uniformly distributed population of strong attracting sites that accumulate gas effectively and allow very few incoming molecules to escape. We say these sites have a high capture cross-section or a large sticking factor (a number between 0 and 1). Their presence will lead to a strong initial upward slope on a graph of adsorbed mass versus gas pressure. As the gas pressure increases these sites start to saturate and the next population of sites, with a lower sticking factor starts to fill at a less rapid rate and the slope of the curve drops. The result is non-Henrian behavior.

From the above theory of adsorption it is conceivable that, for a given adsorbent and adsorbate, a uniform distribution of strong sites may dominate the adsorption behavior at very low pressure and very low surface concentration, leading to Henrian behavior. Smooth surfaces with low porosity and uniform surface structure should also promote Henrian behavior at low pressure and surface concentration. In fact, a theoretical isotherm has been derived for gaseous cesium sorption on particulate nuclear graphite that fits the experimental data very well and indeed reduces to a Henrian isotherm at low concentration and very low pressure but becomes of the exponential or so-called Freundlich type at higher pressure. The Freundlich isotherm has the form:

$$\frac{x}{m} = k \cdot C_V^{1/n}$$

Here x/m is the adsorbed mass of the adsorbate divided by the mass of the adsorbent. Both k and $1/n$ are empirical constants, which depend on the nature of the adsorbate-adsorbent system. The form $1/n$ is used explicitly because the exponent is often less than one. However, isotherms with an exponent greater than one are also sometimes called Freundlich. For the special case where the exponent is one, the Freundlich isotherm becomes Henry's law.

It is easy to visualise how the potential depth of a trap site can influence its sticking factor. Let the trapping energy be E^*, then only those gas molecules with kinetic energy less than or equal to E^* can be trapped. So for a very deep potential well almost all molecules, impinging onto it, are caught and the sticking factor becomes close to one.

Shallow wells, on the other hand, only manage to catch a small fraction of incoming molecules and therefore display low sticking factors. The latter sites are emptied first during desorption as the surface temperature is raised. At very high surface temperature only the very deep trap sites – and of course some subsurface reservoirs – remain filled with the adsorbate.

Consider a distribution of trap sites with a continuous range of adsorption energy. For simplicity we assume that the energy ranges from zero to infinity. The latter of course cannot be true in practice but the example suffices to show the essentials of the arguments involved. Let us assume an exponential fall-off of the density of trapping sites with energy. A distribution that suites this description is:

$$n(E) = \frac{1}{E}\exp\left(-\frac{E}{\langle E\rangle}\right)$$

Here <*E*> is the mean site energy and the formula constitutes a proper distribution as:

$$\int_0^\infty n(E)dE = 1.0$$

The incoming flux (molecules per unit area per unit time) is given by the formula:

$$\phi_0 = \frac{p}{\sqrt{2\pi m k_B T}}$$

Here p is the partial pressure of the radioactive nuclide, m it's mass, k_B is the Boltzmann constant, and T the absolute temperature. The incoming molecules have a Maxwell distribution in velocity so that only a small fraction of molecules have velocity much smaller or much larger than the mean value. If we stick to classical dynamics a molecule can only be trapped if its kinetic energy is less than the adsorption energy. So the fraction of the available flux that can fill up sites with energy E is:

$$\phi(E) = \phi_0 \int_0^{u(E)} f(u)du = \alpha(E)\phi_0$$

The integration limit is defined as:

$$u(E) = \sqrt{\frac{2E}{m}}$$

The parameter $\alpha(E)$ is the so-called sticking factor. It is the fraction of the available total flux that can adsorb to sites of energy E. It is further assumed that the sites of energy E undergo desorption at a rate:

$$\frac{dC(E)}{dt} = -C(E)\vartheta_0 \exp\left(-\frac{E}{RT}\right)$$

Here $C(E)$ is the density of filled sites per unit area of energy E. We further assume that the probability of adsorption, of sites with energy E, depends linearly on the availability of such sites. Let C_{max} be the total number of sites per unit area, then the maximum density of sites per unit area of energy E is:

$$C_{\max}(E) = \frac{C_{\max}}{E} \exp\left(-\frac{E}{\langle E \rangle} \right)$$

The filling rate is proportional to the available empty sites and the flux.

$$\frac{dC(E)}{dt} = \alpha(E)\phi_0 \left(1 - \frac{C(E)}{C_{\max}(E)} \right)$$

Combining the rate of adsorption and desorption we get the overall population rate for sites of energy E.

$$\frac{dC(E)}{dt} = \alpha(E)\phi_0 \left(1 - \frac{C(E)}{C_{\max}(E)} \right) - C(E)\vartheta_0 \exp\left(-\frac{E}{k_B T} \right)$$

We simplify by dropping the explicit reference to energy dependence, remembering that the final equation holds for single site energy only, and we use:

$$\vartheta = \vartheta_0 \exp\left(-\frac{E}{k_B T} \right)$$

then:

$$\frac{dC}{dt} + \left(\frac{\alpha\phi_0}{C_{\max}} + \vartheta \right) C = \alpha\phi_0$$

The incoming flux, Φ_0, depends in general on the thickness of the δ-sublayer [121], equal to the mean free path. The mean free path is given by ([21], section 9.7.3):

$$\delta = \frac{\mu_g}{\rho_g} \cdot \sqrt{\frac{\pi M_w}{2 R_m T_g}} = 0.0137 \cdot \frac{\mu_g}{\rho_g} \cdot \sqrt{\frac{M_w}{T_g}}$$

μ_g	gas viscosity, (kg/m/s)
ρ_g	gas density, (kg/m^3)
R_m	universal gas constant (=8314.51), (J/kmol/K)
M_w	gas molar weight, (kg/kmol)
T_g	gas temperature, (K)

The δ-layer thickness is extremely small. For example, in case of dry air at 300 K and 1.0×10^5 Pa:

$\delta = 0.0137 \times 1.8\times10^{-5} / 1.16 \times (29. / 300.)^{1/2} = 6.6\times10^{-8}$ m

In case of helium at 700 K and 9.0×10^6 Pa:

$\delta = 0.0137 \times 3.2\times10^{-5} / 6.07 \times (4.0 / 700.)^{1/2} = 5.5\times10^{-9}$ m

12.3.5.2 Sorption Models

Three models are available, these are:

- User-defined (Control Function) model, described in section 12.3.5.3
- Sorption model 1 (SPECTRA model), described in section 12.3.5.4
- Sorption model 2 (PATRAS/SPATRA model), described in section 12.3.5.5

The models 1 and 2 are of course the preferred for most calculations. The user-defined model is included mainly for test calculations (for example when a test requires constant sorption flux, constant surface concentration, etc - see Vapor Diffusion Tests, described in Volume 3).

12.3.5.3 User-Defined (CF) Model

In the user-defined model the user simply defines the total sorption mass transfer rate $(dC_S/dt)_{total}$ (kg/m^2-s) using a Control Function:

$$\left(\frac{dC_S}{dt}\right)_{total} = CF$$

CF is the Control Function value. The following limits are imposed internally on the value obtained from the Control Function:

$$-10^{-3} \le CF \le 10^{-3}$$

$$if \;\; C < C_{\min} = 10^{-15}, \quad then \quad (dC_S / dt) = CF \cdot (C / C_{\min})$$

C is the vapor concentration, kg/m^3, in the gas space (if *CF*>0) or inside the wall (if *CF*<0). The second limit (referred to as a "starvation limit") prevents sorption when there is no material to be sorbed. A user-defined multiplier, X1SRRT (for 1-D structures) and X2SRRT (for 2-D structures) is applied for the value obtained from the Control Function.

12.3.5.4 Sorption Model 1 (SPECTRA Model)

The Sorption Model 1 has been elaborated specifically for the SPECTRA code. It is simpler than the Model 2 (PATRAS/SPATRA Model - described in the next section), but it is more practical for use in view of frequent lack of sufficient data to determine all coefficients for the Model 2. Quite often all available data is given in terms of equilibrium isotherms. Such data is relatively easily converted to the Model 1 coefficients (see Volume 3) but is insufficient to deduce all coefficients required by the Model 2.

The Sorption Model 1 is based on Freundlich isotherm, which (see 12.3.5.1) has the following form: $x/m = k \cdot C_V^{1/n}$. The behavior described above may be represented by the differential equation:

$$\left(\frac{dC_S}{dt}\right)_{total} = A_S(T) \cdot C_V^{\,x_A} - B_S(T) \cdot C_d$$

$(dC_S/dt)_{total}$	total sorption mass transfer rate, (kg/m^2-s)
$A_S(T)$	adsorption coefficient, (m/s), dependent on wall temperature, T
C_V	concentration of the vapor in the gas space (kg/m^3). An upper limit on C_V is imposed: $C_V \le C_{sat}(T)$ based on experimental data (see Volume 3)
x_A	adsorption exponent, (-)
$B_S(T)$	desorption coefficient, (m/s), or (1/s) depending on interpretation of C_d
C_d	concentration of the vapor on the surface (=C_S) (kg/m^2), or in the material (=C_S/d, where d is the material thickness) (kg/m^3), depending on the user choice (see input parameter IDSRRT, Volume 2)

An asymptotic solution of the above equation is in agreement with the Freundlich isotherm. Assume that $x_A = 1/n$ and there is a constant vapor concentration, C_V, and that $C_d = C_S$ or C_S/d, where d is the material thickness, (m). The latter assumption means that the diffusion inside the material is instantaneous, or at least fast compared to the adsorption rate. With these assumptions the solution of the above differential equation gives:

$$C_d = A_S(T) \cdot d \cdot C_V^{1/n} \cdot \left(1 - \exp[-B_S t]\right)$$

The asymptotic value of C_d is:

$$C_d(t \to \infty) = A_S(T) \cdot d \cdot C_V^{1/n}$$

This is identical to the Freundlich isotherm shown above, if $A_S(T) = k/d$, since the adsorbed concentration is equal to $C_d = (x/m)$. The coefficients A_S, B_S are obtained from the relation applied in [46]:

$$A_S(T) = A_0 \cdot \exp\left(-\frac{A_A}{T}\right)$$

$$B_S(T) = B_0 \cdot \exp\left(-\frac{A_B}{T}\right)$$

Substituting the above formulae into the equation defining the total sorption rate we obtain:

$$\left(\frac{dC_S}{dt}\right)_{total} = A_0 \cdot \exp\left(-\frac{A_A}{T}\right) \cdot C_V^{\,x_A} - B_0 \cdot \exp\left(-\frac{A_B}{T}\right) \cdot C_d$$

In the implementation a user-defined multiplier, X1SRRT (for 1-D structures) and X2SRRT (for 2-D structures), is applied for the value obtained from the above formula and the following limits are imposed (the values may be redefined by the user - see Volume 2):

$$A_{\min} \le A_S(T) \le A_{\max}$$

$$B_{\min} \le B_S(T) \le B_{\max}$$

Note that a similar model is adopted in the MELCOR code. In MELCOR the sorption rate is calculated from [46]:

$$\frac{dC_S}{dt} = a_{ij} \cdot \exp\left(-\frac{E_{ij}}{RT}\right) \cdot C_V$$

A corresponding model for SPECTRA may be built by applying the following coefficients:

$$A_0 = a_{ij} \quad A_A = \left(\frac{E_{ij}}{R}\right) \quad x_A = 1.0$$
$$B_0 = 0.0 \quad A_B = 0.0$$

In MELCOR the desorption constant is not used ($B_0 = 0.0$). In SPECTRA a positive value may be used. If a positive B_0 is applied, then two options are available (see input parameter IDSRRT, Volume 2):

- Both direction sorption
  With this option negative sorption is allowed. When the vapor concentration, C_V, becomes small, then:
  $$A_S \cdot C_V^{x_A} - B_S \cdot C_d < 0.0$$
  In such case:
  $$\left(\frac{dC_S}{dt}\right)_{total} < 0.0$$

- Only positive sorption allowed.
  A limit may be imposed that will prevent desorption of the adsorbed isotopes in case then C_V becomes small. With this option the total sorption flux is: $(dC_S/dt)_{total} \geq 0$.

Based on the experimental observations a limit is imposed on the value of C_V in the adsorption term, $C_{sat}(T)$. It has been observed (see Figure 12-78) that for vapor concentrations higher than a certain limit (temperature-dependent) the sorption is slowed down or stopped completely (the surface becomes saturated). In the SPECTRA model this is achieved by using the reduced exponent, x_{red}, (XRSRRT - Volume 2). If the value of x_{red} is zero, then the adsorption term remains constant for $C_V > C_{sat}(T)$ (see Figure 12-79):

$$A_0 \cdot \exp\left(-\frac{A_A}{T}\right) \cdot C_{sat}^{x_A}(T) \qquad if \ x_{red} = 0.0$$

If a positive value is used then the adsorption term slowly increases with C_V (see Figure 12-80):

$$A_0 \cdot \exp\left(-\frac{A_A}{T}\right) \cdot C_V^{x_{red}} \cdot \left(\frac{C_{sat}^{x_A}(T)}{C_{sat}^{x_{red}}(T)}\right) \qquad if \ x_{red} > 0.0$$

The last term ensures continuity of the function when $C_V = C_{sat}(T)$. The value of $C_{sat}(T)$ may be a function of wall temperature, T, according to the equation similar to that applied for $A_S(T)$ and $B_S(T)$:

$$C_{sat}(T) = C_0 \cdot \exp\left(-\frac{A_C}{T}\right)$$

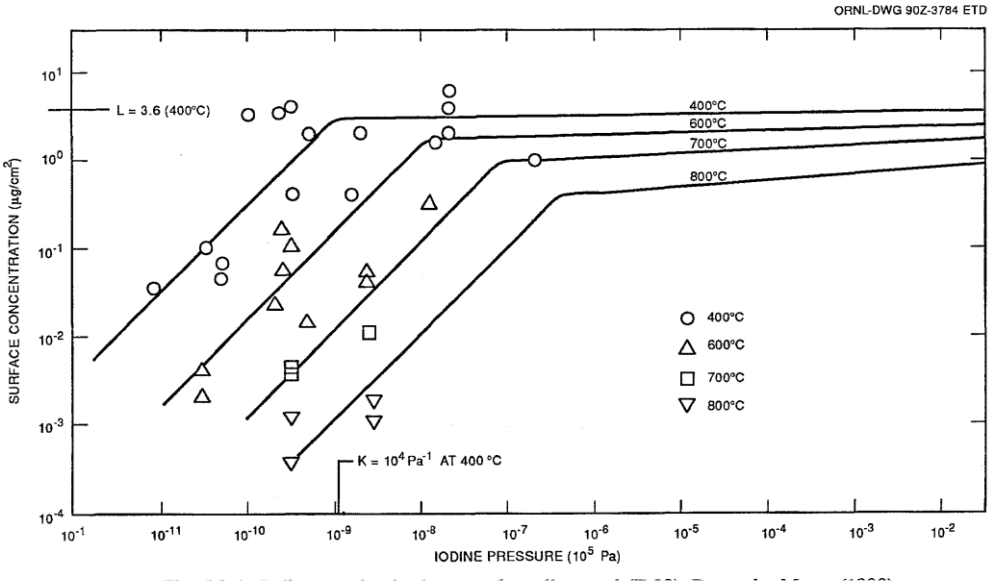


Figure 12-78 Sorption on steel - data and Langmuir isotherms from Wichner [183].

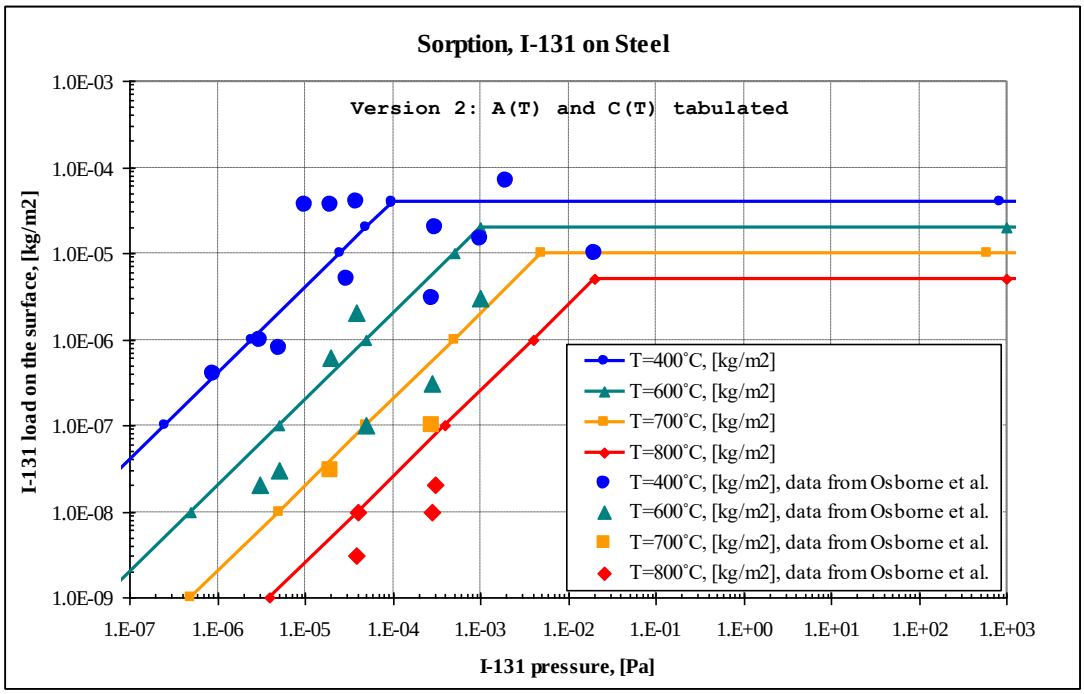


Figure 12-79 Sorption on steel - data and SPECTRA model with x_{red} = 0.0

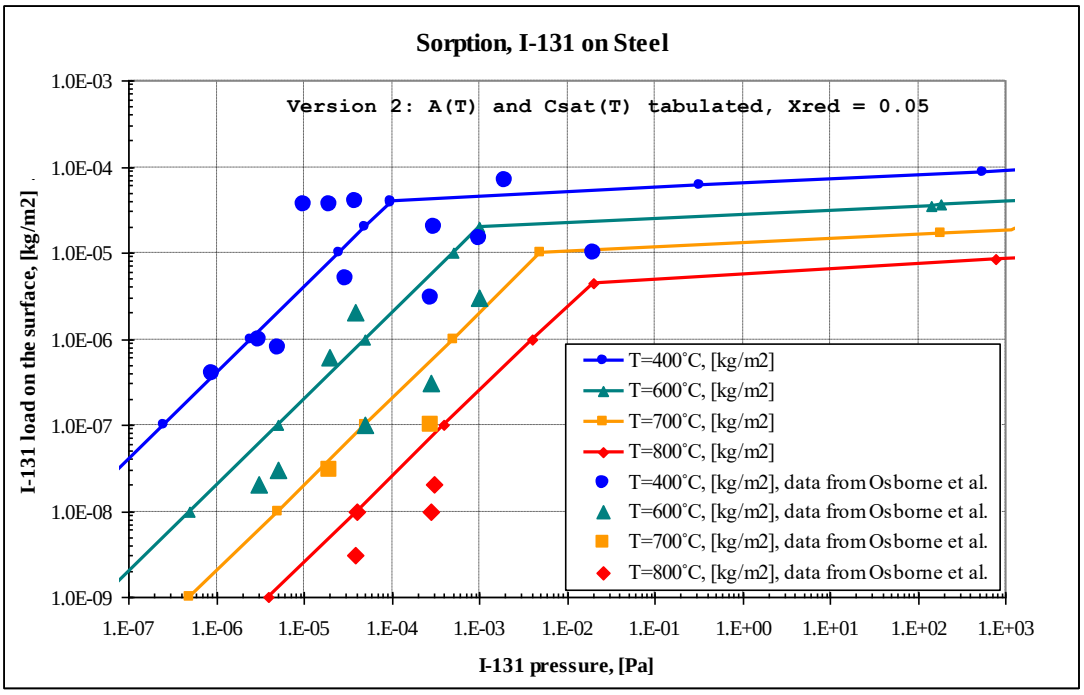


Figure 12-80 Sorption on steel - data and SPECTRA model with x_{red} = 0.05

Alternatively any or all of the three coefficients, $A_S(T)$, $B_S(T)$, and $C_{sat}(T)$, may be tabulated versus temperature using the general purpose Tabular Function utility available in SPECTRA (see Volume 2).

Summarizing, the adsorption is calculated as:

- If $C_V < C_{sat}$:

$$A_S(T) \cdot C_V^{x_A}$$

- If $C_V > C_{sat}$:

$$A_S(T) \cdot C_V^{x_{red}} \cdot \left(\frac{C_{sat}^{x_A}(T)}{C_{sat}^{x_{red}}(T)} \right)$$

Finally, the sorption model equation is:

$$\left(\frac{dC_S}{dt} \right)_{total} = \begin{cases} A_S(T) \cdot C_V{}^{x_A} - B_S(T) \cdot C_d & for\ C_V \le C_{sat}(T) \\ A_S(T) \cdot C_V{}^{x_A} \cdot \left(\frac{C_{sat}^{x_A}}{C_{sat}^{x_{red}}} \right) - B_S(T) \cdot C_d & for\ C_V > C_{sat}(T) \end{cases}$$

The model may be applied for either gas-covered or liquid-covered surfaces, depending on the user-defined selector (ISRLRT, see Volume 2).

If the sorption model is applied for liquids, then the adsorption flux may be calculated using the heat and mass transfer analogy. In order to do that, the user must define the diffusion coefficient in liquids, D_C, (m^2/s) (DCSRRT - Volume 2).

- If the diffusion coefficient D_C (DCSRRT, Volume 2) is not defined, then the adsorption coefficient is obtained from the user-defined constants A_0, A_A and the temperature-dependent formula:

$$A_S = A_0 \cdot \exp(-A_A / T)$$

- If the diffusion coefficient D_C (DCSRRT) is defined, then the adsorption flux is computed using the heat and mass transfer analogy, where: $Nu \rightarrow Sh$, $Pr \rightarrow Sc$. The Sherwood number is defined as:

$$Sh = \frac{A_S \cdot D_{FC}}{D_C}$$

  Here A_S is the adsorption mass transfer coefficient (m/s), D_{FC} is the hydraulic diameter (characteristic dimension for forced convection), D_C, is the diffusion coefficient in liquids.

  The mass transfer correlation has the following general form:

$$Sh = A_{Sh} + B_{Sh} \frac{Re^{C_{Sh}}\ Sc^{D_{Sh}}}{1 + E_{Sh}\ Re^{F_{Sh}}\ Sc^{G_{Sh}}}$$

  For example, in case of internal forced convection, Dittus-Boelter correlation is:

$$Sh = 0.023\ Re^{0.8}\ Sc^{0.4}$$

  To apply this correlation, one needs to set $B_{Sh} = 0.023$, $C_{Sh} = 0.8$, $D_{Sh} = 0.4$, other coefficients being equal to zero.

- If the diffusion coefficient D_C (DCSRRT) and, additionally, the diffusion layer boundary thickness δ_{BL} (TBL2RT) and the constant C_{LB}.(CLB2RT) in the Langmuir-Blodgett correlation for inertial impaction are defined, then the adsorption flux is computed using Brownian diffusion and inertial impaction. The model is available only if fission products are transported as particles (DPFPRT>0.0, RHFPRT>0.0) and transport on liquid-covered surfaces is active (ISRLRT=2). Aerosol particles represent gas bubbles in this case.

  - *Brownian diffusion*
    The Brownian diffusion velocity is obtained from:

$$v_{Brown} = \frac{D_C}{\delta_{BL}}$$

    v_{Brown} Brownian deposition velocity, (m/s)
    D_C diffusion coefficient, (m^2/s)
    δ_{BL} diffusion boundary layer thickness, (m)

  - *Inertial impaction*

The inertial impaction velocity is obtained from:

$$v_{inertial} = v_{\infty} \cdot \eta \cdot A_{h,bubb} / A_{bubb} = v_{\infty} \cdot \eta / 4$$

$v_{inertial}$	inertial impaction deposition velocity, (m/s)
$A_{h,bubb}$	horizontal cross section area of a "bubble", (m^2), $= \pi \times D_{bubb}^2 / 4$
A_{bubb}	total surface area of a "bubble", (m^2), $= \pi \times D_{bubb}^2$
v_{∞}	fission product particle-to-"bubble" relative velocity (m/s)
η	collection efficiency, calculated from one of three correlations, described below.

Langmuir and Blodgett (L-B) correlation

The L-B correlation was developed for a single sphere. The correlation is:

$$\eta = \eta_{\max} \cdot \frac{Stk^2}{(Stk + C_{LB})^2}$$

C_{LB}	constant (=0.25, see [206], eq. 3)
η_{max}	maximum value of collection efficiency (=1.0)
Stk	Stokes number, (-), defined as:

$$Stk = \frac{\rho_p \cdot d_p^2 \cdot v_{\infty}}{9 \cdot \mu_f \cdot D_{bubb}}$$

ρ_p	fission product particle density, (kg/m^3) = RHFPRT
d_p	fission product particle diameter, (m) = DPFPRT
v_{∞}	fission product particle-to-"bubble" relative velocity, (m/s)
μ_f	liquid viscosity, (kg/m/s)
D_{bubb}	"bubble" diameter, (m) (diameter of aerosol particle)

Collection efficiency obtained from the correlation is shown in Figure 12-81. The effect of the constant C_{LB} on the collection efficiency is shown. The value of η_{max} determines the maximum value of collection efficiency (η_{max} = 1.0 in Figure 12-81). The values are defined by the user in the input deck (C_{LB} = CLB2RT, η_{max} = ELB2RT, Volume 2).

Modified Langmuir and Blodgett correlation

The modified correlation is defined as follows:

$$\eta = \eta_{\max} \cdot \begin{cases} \dfrac{0.25}{C_{LB}} \cdot Stk & for \quad Stk < C_{LB} \\ \dfrac{Stk^2}{(Stk + C_{LB})^2} & for \quad Stk > C_{LB} \end{cases}$$

As seen in Figure 12-81, for Stk = 0.25 the line $\eta = Stk$ is tangential to the L-B correlation. In general, the line $\eta = (0.25/C_{LB}) \times Stk$ is tangential for any value of C_{LB}. In the modified correlation this line is used for $Stk < C_{LB}$, which gives somewhat higher collection

efficiency in this region. Comparison of both correlations is shown in Figure 12-82. The modified L-B line is very similar to the L-B line; the difference is smaller than the scatter of the source data ([206], figure 1). The difference is important only for small values of the Stokes number. For example, for *Stk*=0.03, the L-B correlation gives η=0.01 and the modified correlation η=0.03, so the relative difference is quite large in this region. The modified correlation gives much better agreement with the experimental data for the Static Column experiments, as shown in Volume 3.

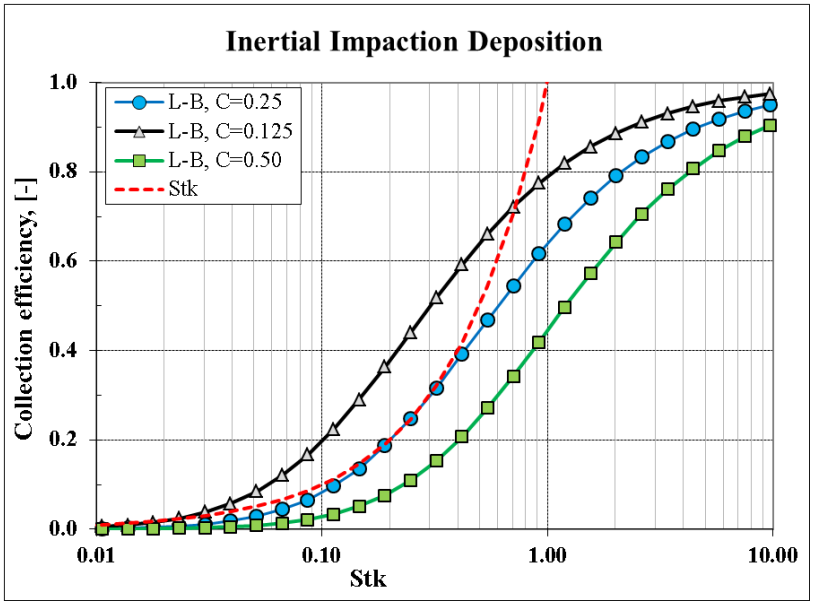


Figure 12-81 Langmuir and Blodgett correlation

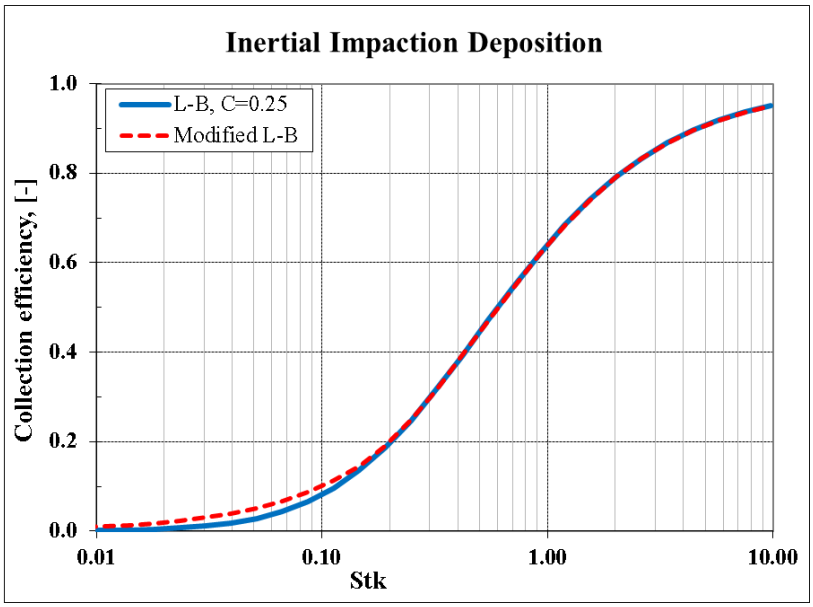


Figure 12-82 Modified versus original Langmuir and Blodgett correlation

Correlation developed based on data of Yoon & Lutrell and Afruns & Kitchener
The following correlation has been developed specifically for use in the SPECTRA code, based on data of Yoon & Lutrell [207] for coal particles and Afruns & Kitchener [208] for quartz particles. The data, (copied from [209], figure 9.5) is shown in Figure 12-83. The correlation is:

$$\eta = Min\left(\eta_{\max}, A \cdot \frac{d_p^{\,2}}{D_b}\right)$$

Performed calculations showed that best agreement is obtained with A = 13,000. The lines obtained with A = 13,000 are shown in Figure 12-83 (red lines, yellow markers). It was found that due to scatter of data the that values between 9,000 ≤ A ≤ 17,000 can be justified.

NOTE: The correlations described above are applied for migration of fission product isotopes to aerosol particles in a liquid. Exactly the same correlations are applied for migration to bubbles, as described in section 12.3.7.2.

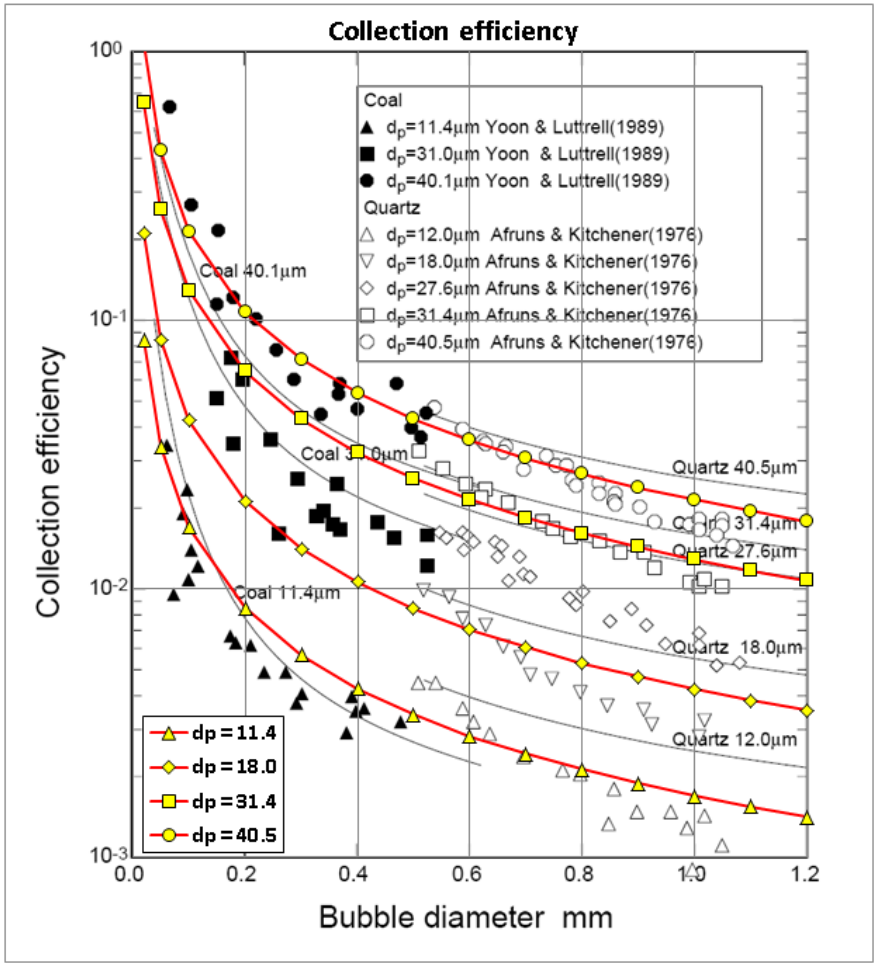


Figure 12-83 Data of Yoon & Lutrell and Afruns & Kitchener and correlation

12.3.5.5 Sorption Model 2 (PATRAS/SPATRA Model)

The Sorption Model 2 is based on the PATRAS/SPATRA model (published by Kress & Neil model and Iniotakis [121] takes into account the following phenomena Figure 12-84:

- Transport of the vapor from the bulk gas to the δ-sublayer (mean free path).
- Adsorption of the vapor from the δ-sublayer to the wall material, where the part β remains on the surface as reversibly bound molecules, while the part $(1-\beta)$ penetrates into the material.
- Desorption of the molecules that are reversibly bound at the surface back to the δ-sublayer.
- Eventual evaporation of the molecules that have penetrated into the material from the surface node into the δ-sublayer.

The model is applicable for the gas-covered surfaces only. In case of liquid-covered surfaces, only desorption of the reversibly-bound particles is calculated. The transport rates for each of these phenomena are discussed subsequently below.

- **Transport to the δ-sublayer**

  The transport to the δ-sublayer is given by:

$$h_\delta \cdot (C_{CV} - C_\delta)$$

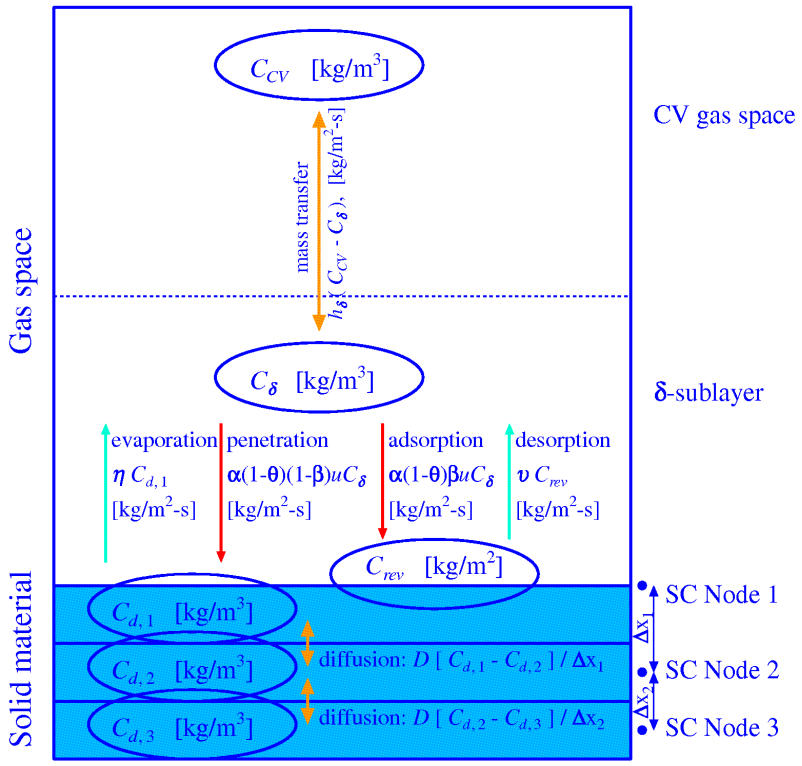


Figure 12-84 Sorption/penetration model in SPECTRA.

h_δ mass transfer coefficient, (m/s)
C_{CV} vapor concentration in the gas space of a Control Volume CV, (kg/m^3)
C_δ vapor concentration in the δ-sublayer, (kg/m^3)

The mass transfer coefficient, h_δ, is obtained using the heat and mass transfer analogy from the same correlations as those used to calculate convective heat transfer coefficient at the wall surface - see sections 7.1.1 and 7.1.2. In the heat and mass transfer analogy; the Nusselt number is replaced by the Sherwood number, while the Prandtl number is replaced by the Schmidt number. For example, in case of internal flow, and turbulent forced convection the heat transfer coefficient obtained from the Dittus-Boelter correlation [22] (section 7.1.2):

$$Nu = 0.023\ Re^{0.8}\ Pr^{0.4}$$

In such case the mass transfer coefficient is obtained from:

$$Sh = 0.023\ Re^{0.8}\ Sc^{0.4}$$

Sh Sherwood number, equal to: $Sh = h_\delta \cdot D_{FC} / D_C$, (-)
D_{FC} characteristic dimension, (m), for forced convective heat transfer
D_C diffusion coefficient, (m^2/s)
Sc Schmidt number, equal to: $Sc = \eta_g / (\rho_g \cdot D_C)$, (-)
η_g gas viscosity, (kg/m-s)
ρ_g gas density, (kg/m^3)

As an alternative a user-defined Sherwood number correlation may be used. In such case the Sherwood number is calculated from the following correlation:

$$Sh = A_{Sh} + B_{Sh}\frac{Re^{C_{Sh}}\ Sc^{D_{Sh}}}{1 + E_{Sh}\ Re^{F_{Sh}}\ Sc^{G_{Sh}}}$$

A_{Sh}, B_{Sh}, C_{Sh}, D_{Sh}, E_{Sh}, F_{Sh}, G_{Sh}, are user-defined coefficients. In order to use the above formula either A_{Sh} or B_{Sh} must be defined as positive (see Volume 2). As a second alternative the Gnielinski correlation may be used. The Gnielinski correlation [122] is:

$$Sh = \frac{r \cdot (Re - 1000) \cdot Sc}{1 + 12.7 \cdot \sqrt{r} \cdot (Sc^{2/3} - 1)} \cdot \left(1 + \left(\frac{d}{x}\right)^{2/3}\right) \cdot \left(\frac{Sc_{gas}}{Sc_{wall}}\right)^{0.11}$$

where:

$$r = (1/8) \cdot (1.82 \cdot \log_{10}(Re) - 1.64)^{-2}$$

Sc_g and Sc_w are the Schmidt numbers at the gas temperature, T_{gas}, and the wall temperature, T_{wall} , respectively, d is the hydraulic diameter, and x is the distance from tube entrance. For practical purposes half of the flow length can be used. In the implementation used in RADAX the ratio of Schmidt numbers is replaced by the temperature ratio [121]. The same approach is taken in SPECTRA:

$$Sh = \frac{r \cdot (Re - 1000) \cdot Sc}{1 + 12.7 \cdot \sqrt{r} \cdot (Sc^{2/3} - 1)} \cdot \left(1 + \left(\frac{d}{x}\right)^{2/3}\right) \cdot \left(\frac{T_{gas}}{T_{wall}}\right)^{y}$$

In the RADAX code the value of y is taken as 0.45 [121]. Comparison of the results of $(Sc_g/Sc_w)^{0.11}$ and $(T_g/T_w)^{y}$ obtained with the value of y = 0.45 is shown in Figure 12-85. It was found out (see Volume 3) that a better representation is obtained using y = 0.015 - Figure 12-86. Therefore this value is chosen as a default value in SPECTRA (see Volume 2, Word 28, record 895YXX).

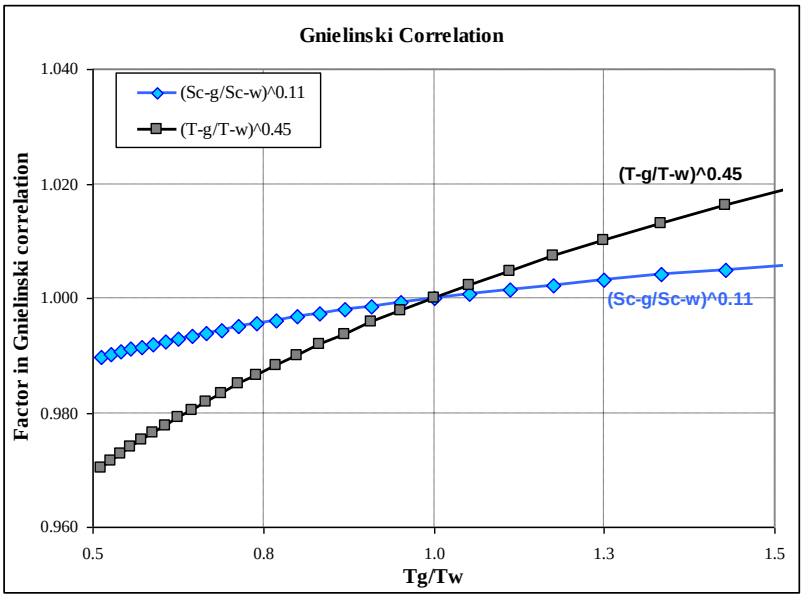


Figure 12-85 Comparison of correction factors $(Sc_g/Sc_w)^{0.11}$ and $(T_g/T_w)^{0.45}$

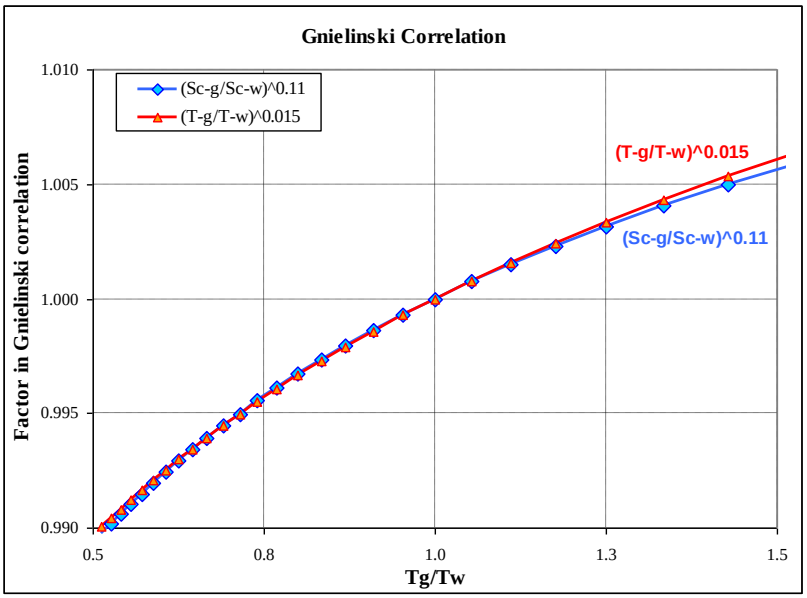


Figure 12-86 Comparison of correction factors $(Sc_g/Sc_w)^{0.11}$ and $(T_g/T_w)^{0.015}$

Once the Sherwood number is calculated, the mass transfer coefficient is obtained from the Sherwood number definition, as:

$$h_\delta = Sh \cdot \frac{D_C}{D_{FC}}$$

In the natural convection mode the D_{FC} is of course replaced by D_{NC}. The diffusion coefficient, D_C, may be calculated from one of the two methods.

Table 12-13 Molar weights [32] (page 1-7) and diffusion volumes, [31] (table 11-1).

Element	M_W	Σ	Compound	M_W	Σ
C	12.01	15.90	H_2	2.02	6.12
H	1.01	2.31	D_2	4.03	6.84
O	16.00	6.11	N_2	28.00	18.50
N	14.00	5.43	O_2	32.00	16.30
F	19.00	14.70	Air	28.01	19.70
Cl	35.45	21.00	CO	28.80	18.0
Br	79.90	21.90	CO_2	44.01	26.90
I	126.90	29.80	N_2O	44.00	35.90
S	32.06	22.90	NH_3	16.02	20.70
He	4.00	2.67	H_2O	18.02	13.10
Ne	20.18	5.98	SF_4	108.06	71.30
Ar	39.95	16.20	Cl_2	70.91	38.40
Kr	83.80	24.50	Br_4	159.80	69.00
Xe	131.30	32.70	SO_2	64.06	41.80

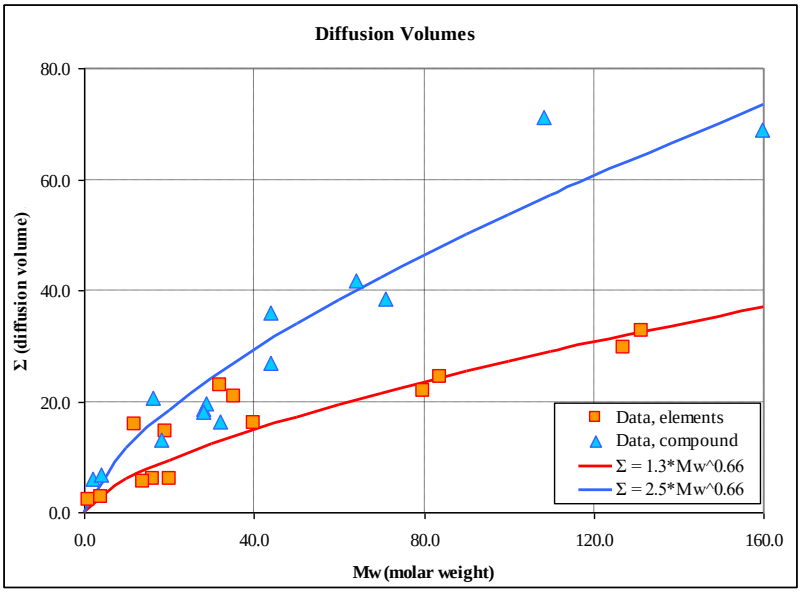


Figure 12-87 Diffusion volumes - data and correlations: $\Sigma = 1.3 \cdot M_w^{0.66}$, $\Sigma = 2.5 \cdot M_w^{0.66}$.

- A general method, applicable for any gas mixture, using the Blanc's law and the binary diffusion coefficients, as described in section 3.5.2.3. In such case the molar weight, M_W, of the adsorbed vapor, as well as the molecular diffusion volume, Σ, must be specified by the user. The molecular diffusion volumes may be found in literature, some values are listed in the Table 12-13 and Figure 12-87. For elements a simple correlation has been developed that may be used in case of lack of more precise data:

$$\Sigma \sim 1.3 \cdot M_w^{0.66}$$

In the above correlation A is the atomic number and Σ is the molecular diffusion volume. For the compounds, on the other hand, the correlation is:

$$\Sigma \sim 4.2 \cdot M_w^{0.66}$$

Results obtained with the above correlations are indicated in Figure 12-87.

The method described above is preferred for its generality, i.e. applicability to the eventually changing gas composition. The second method is:

- Diffusion coefficients are calculated from a temperature-dependent Chapman-Enskog correlation. The following form of the correlation is applied:

$$D_C = A_D \cdot \frac{T_g^{B_D}}{p}$$

A_D	user-defined coefficient
B_D	user-defined coefficient
T_g	gas temperature, (K)
p	gas pressure, (Pa)
D_C	diffusion coefficient, (m^2/s)

For example, in case of helium gas carrier and the fission product vapors of I, Cs, and Ag, reference [119] gives the following relation for the diffusion coefficient:

$$D_C = 10^{-8} \cdot T_g^{1.5} \cdot \left(\frac{p_0}{p} \right)$$

where T_g is in K p in Pa and $p_0 = 10^5$ Pa. This correlation is defined in SPECTRA by choosing the following input parameters:

$A_D = 1.0 \times 10^{-8} \times 1.0 \times 10^5 = 1.0 \times 10^{-3}$
$B_D = 1.5$

Note that the results of the Chapman-Enskog type correlations have been compared with the results of the Fuller/Blanc's law for the steam-air and steam-hydrogen mixtures in section 3.5.2.3. Below a similar comparison is presented for Iodine diffusion coefficient in helium. Calculations were performed for 1.0 bar and 10.0 bar pressure, using two methods:

- The method of Fuller was applied M_w = 126.9 and Σ = 29.8, as the values appropriate for I - Table 12-13. (Note that for the CV gases the molar weights and

diffusion volumes are tabulated inside the SPECTRA code, therefore there is no need to specify these numbers for helium.) Additionally calculations were performed assuming that iodine forms I_2 compound. For such compound the molar weight is equal to twice the value for the element, which means: M_w = 2×126.9 = 253.8. The diffusion volume is roughly twice the value for the element, thus Σ =2×29.8 = 59.6. The proposed correlations give for I, atomic number of A = 53: $\Sigma=2.4\times53^{0.66}=33$ and for I_2: $\Sigma=4.2\times53^{0.66}=58$, so quite similar values.

- The Chapman-Enskog correlation, with $A = 1.0\times10^{-3}$ and B = 1.5, as the values appropriate for I, Cs, Ag in He, based on [119].

Results are shown in Figure 12-88 and Figure 12-89. Results of the Fuller method with M_w and Σ appropriate for I are compared to the Chapman-Enskog correlation in Figure 12-88. Results of the Fuller method with M_w and Σ appropriate for I_2 are compared to the same Chapman-Enskog correlation in Figure 12-89. The Fuller method gives results within about ±30% of the values obtained from the Chapman-Enskog correlation. The values obtained for I_2 are closer to the correlation in the practically interesting, high temperature region. In the region of 600 - 1000 K the discrepancy is lower than 10%, which is considered as quite satisfactory.

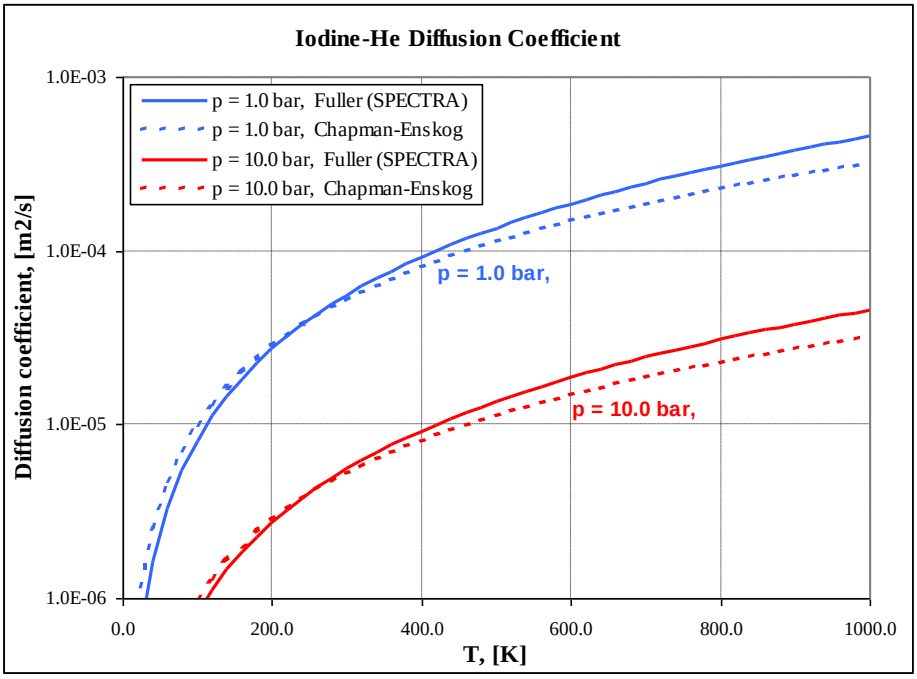


Figure 12-88 Comparison of diffusion coefficients for I in He.

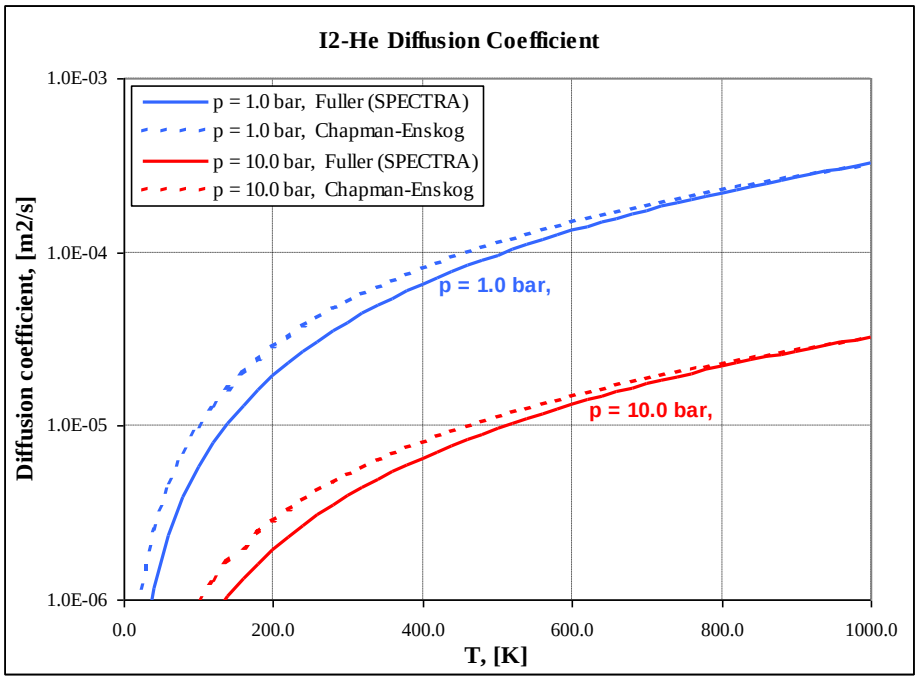


Figure 12-89 Comparison of diffusion coefficients for I_2 in He.

- **Adsorption and Penetration (Absorption)**

  The molecules that are adsorbed from the gas partly remain on the surface and partly penetrate into the material. The term penetration is used here, but an alternative term - absorption is often used. The total adsorption and penetration (absorption) rate is given by:

$$\alpha \cdot u \cdot C_\delta \cdot (1-\theta)$$

  α sticking factor, i.e. the probability that an atom hitting the free surface stays there
  u mean thermal velocity perpendicular to the surface, equal to:

$$u = \sqrt{\frac{k_B T_\delta}{2\pi \cdot m_m}}$$

  k_B Boltzmann constant (= 1.38×10^{-23}), (J/K)
  T_δ gas temperature in the δ-sublayer, (K) (assumed equal to the wall surface temperature, T_w, because of extremely small thickness of the δ-sublayer, ~10^{-8} m - see section 12.3.5.1)
  m_m mass of a single molecule of the fission product, (kg), equal to:

$$m_m = \frac{M_w}{N_A}$$

N_A Avogadro number (=6.022×10[26] - [32]), Avogadro number, (kmole^{-1})
M_w molar weight of the fission product vapor, (kg/kmole)
C_δ vapor concentration in the δ-sublayer, (kg/m^3)
θ reversible surface coverage, fraction of the adsorption sites that are already filled up, equal to:

$$(1-\theta)=\left(1-\frac{C_{rev}}{C_{\max}}\right)$$

$C_{\max}$ is the maximum number of adsorption sites per unit surface area, (m^{-2}), while C_{rev} is the number of adsorption sites occupied by reversibly bound molecules, again per unit surface area, (m^{-2}). This parameter is converted internally to the mass density using the conversion factor equal to (M_w/N_A): C_{max} (kg/m^2) = C_{max} (m^{-2}) × (M_w/N_A). The adsorption term is equal to:

$$\alpha\cdot\left(1-\frac{C_{rev}}{C_{\max}}\right)\cdot u\cdot C_\delta$$

where:

$$u=\sqrt{\frac{k_B T_\delta N_A}{2\pi\cdot M_w}}=36.36\cdot\sqrt{\frac{T_\delta}{M_w}}$$

The concentration of molecules, expressed in the number of molecules per cubic meter, may be converted to the mass concentration (in kilograms per cubic meter) using the following relation:

$$N=\frac{m}{V}\cdot\frac{N_A}{M_w}=\frac{N_A}{M_w}\cdot C$$

N number density a given fission product vapor, (m^{-3})
m mass of the fission product vapor, (kg)
C concentration of the fission product vapor, (kg/m^3)
V volume of the gas space, (m^3)
N_A Avogadro number (=6.022×10[26] - [32]), Avogadro number, (kmole^{-1})
M_w molar weight of the fission product vapor, (kg/kmole)

The number concentration is proportional to the mass concentration in kg/m^3, with the proportionality coefficient of (N_A/M_w). Therefore the mass concentrations may be used instead of the number concentrations provided that they are used consistently in all terms of the equation. The adsorption term is written as:

$$\alpha\cdot(1-\theta)\cdot u\cdot C_\delta \qquad with: \quad u=36.36\cdot\sqrt{\frac{T_\delta}{M_w}}$$

Fraction of adsorbed molecules that remains on the surface is denoted by β, (-). The difference (1–β) is called the penetration factor, since it gives the fraction of adsorbed material that penetrates into the wall (is absorbed by the wall). This parameter is calculated from a temperature-dependent correlation:

$$1-\beta=(1-\beta_0)\cdot\exp\left(\frac{A_\beta}{T_w}\right)$$

$(1–\beta_0)$, A_β, are user-defined input coefficients. A constant value of the penetration factor is obtained in the model by setting $A_\beta = 0.0$. For example, reference [120] gives for Cs on Incoloy-800:

$$1-\beta = 1\times10^{-5}$$

This is obtained by setting $(1–\beta_0) = 1.0\times10^{-5}$. Therefore:

$$1-\beta_0 = 1\times10^{-5} \qquad A_\beta = 0.0$$

Reference [117] fit empirical data from [118] to obtain for Cs a temperature-dependent formula, with the coefficients equal to: $A_\beta = 15121$, $(1–\beta_0) = 4.1\times10^{-11}$. Therefore:

$$1-\beta = 4.1\times10^{-11}\cdot\exp\left(\frac{15121}{T_w}\right)$$

Both relations are shown in Figure 12-90. They yield the same value for $T = 1200$ K, but very different values are obtained for different temperatures. The temperature-dependent formula gives better agreement with experimental data.

In SPECTRA limits may be imposed on the value obtained from the correlation:

$$(1–\beta)_{min} \le 1–\beta \le (1–\beta)_{max}.$$

$(1–\beta)_{min}$ and $(1–\beta)_{min}$ are user-defined values. For example, the correlation for Cs on Incoloy-800, shown above, has been applied with the limits of $10^{-6} \le (1–\beta) \le 0.1$. The resulting line is shown in Figure 12-90.

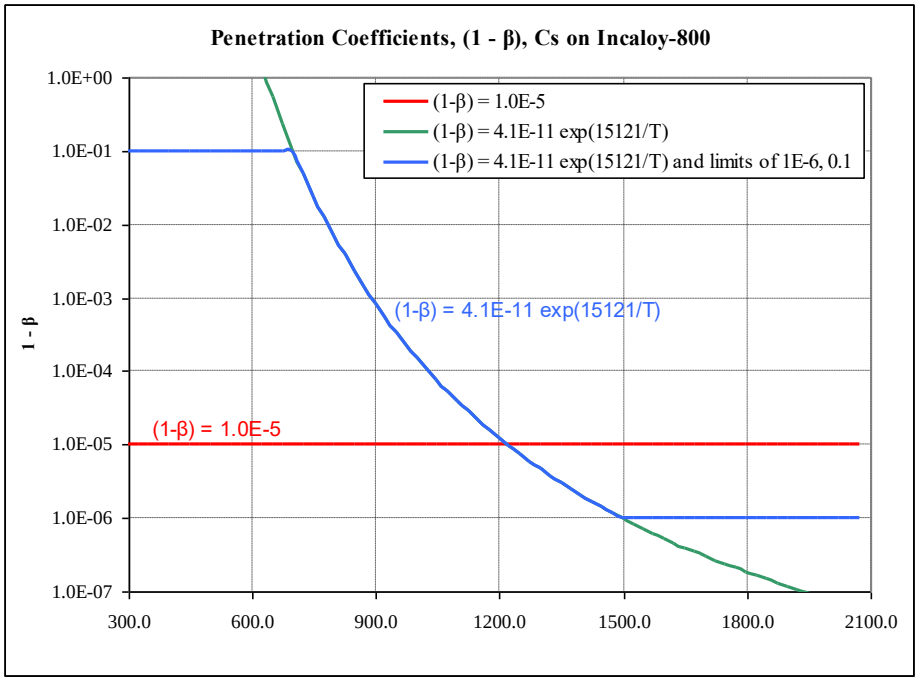


Figure 12-90 Penetration coefficient, $(1–\beta)$, Cs on Incoloy-800.

- **Desorption**

The desorption term is given by:

$$\vartheta \cdot C_{rev}$$

ϑ desorption coefficient, (s^{-1}) (note that for water covered surface different desorption coefficients are used - see section 12.3.7.2)

C_{rev} surface concentration of particles reversibly bound at the surface, (kg/m^2)

The desorption coefficient is calculated from a temperature-dependent correlation:

$$\vartheta = \vartheta_0 \cdot \exp\left(-\frac{E_\vartheta}{RT_w}\right) = \vartheta_0 \cdot \exp\left(-\frac{A_\vartheta}{T_w}\right)$$

E is the activation energy, (J), T_w is the wall surface temperature (K), R is the universal gas constant. Limits may be imposed on the value obtained from the correlation: $\vartheta_{\min} \le \vartheta \le \vartheta_{\max}$. The values of $\vartheta_0, A_\vartheta, \vartheta_{\min}, \vartheta_{\max}$, are user-defined input coefficients. For example, reference [120] gives for Cs on Incoloy-800:

$$\vartheta_0 = 1\times10^{11} \qquad A_\vartheta = (E_\vartheta / R) = (234000 / 8.314) = 2.815\times10^4$$

Therefore:

$$\vartheta = 1.0\times10^{11} \cdot \exp\left(-\frac{2.815\times10^4}{T_w}\right)$$

- **Evaporation**

The evaporation term is given by:

$$\eta \cdot C_{d,i}$$

η evaporation coefficient, (s^{-1})

$C_{d,i}$ concentration of particles in the cell i (boundary cell of the structure), (kg/m^3) (in order to simplify the notation the subscript i is skipped and the symbol C_d is used in the following discussion)

The evaporation coefficient is calculated from a temperature-dependent correlation:

$$\eta = \eta_0 \cdot \exp\left(-\frac{E_\eta}{RT_w}\right) = \eta_0 \cdot \exp\left(-\frac{A_\eta}{T_w}\right)$$

E_η is the activation energy, (J), T_w is the wall surface temperature (K), R is the universal gas constant. Limits may be imposed on the value obtained from the correlation: $\eta_{\min} \le \eta \le \eta_{\max}$. The values of η_0, A_η $\eta_{\min}$, $\eta_{\max}$, are user-defined input coefficients. Quite often calculations are performed neglecting evaporation, which means:

$$\eta_0 = 0.0 \qquad A_\eta = 0.0$$

- **Calculation Procedure Applied for the Sorption Model 2**

The total sorption mass transfer rate is obtained from:

$$\left(\frac{dC_S}{dt}\right)_{total} = \alpha \cdot (1-\theta) \cdot u \cdot C_\delta - \vartheta \cdot C_{rev} - \eta \cdot C_d$$

In order to calculate the mass transfer rate the individual terms, such as C_δ, C_{rev}, C_d, must be known. Calculation of the individual terms is described below. The concentration in the δ-sublayer is obtained using a mass balance for the layer (per unit surface area):

$$\delta \frac{dC_\delta}{dt} = h_\delta (C_{CV} - C_\delta) - \alpha \cdot (1-\theta) \cdot u \cdot C_\delta + \vartheta \cdot C_{rev} + \eta \cdot C_d$$

The derivative is in the numerical representation given by:

$$\frac{dC_\delta}{dt} = \frac{C_\delta - \overline{C_\delta}}{\Delta t}$$

The value of $\overline{C_\delta}$ is the value at the beginning of the time step. An implicit scheme is applied, which means that the end-of-time-step values are used for the concentrations on the right hand side of the equation. Therefore:

$$\delta \frac{C_\delta - \overline{C_\delta}}{\Delta t} = h_\delta (C_{CV} - C_\delta) - \alpha \cdot (1-\theta) \cdot u \cdot C_\delta + \vartheta \cdot C_{rev} + \eta \cdot C_d$$

After rearrangement:

$$C_\delta = \frac{\frac{\delta}{\Delta t} \cdot \overline{C_\delta} + \left[h_\delta C_{CV} + \vartheta \cdot C_{rev} + \eta \cdot C_d\right]}{\frac{\delta}{\Delta t} + \left[h_\delta + \alpha \cdot (1-\theta) \cdot u\right]}$$

SPECTRA uses a stationary (asymptotic) version of this equation, obtained for $\delta/\Delta t \to 0$ and an explicit (beginning of time step) value of θ:

$$C_\delta = \frac{h_\delta C_{CV} + \vartheta \cdot C_{rev} + \eta \cdot C_d}{h_\delta + \alpha \cdot (1-\overline{\theta}) \cdot u}$$

where:

$$(1-\overline{\theta}) = \left(1 - \frac{\overline{C_{rev}}}{C_{\max}}\right)$$

Use of the stationary form is justified by an extremely small thickness of the δ-sublayer (of order of 10^{-8} m, see section 12.3.5.1). Therefore the inertia of the δ-sublayer is very small: $\delta \cdot (C_\delta - \overline{C_\delta}) / \Delta t \approx 0.0$.

Recalling that:

$$u = 36.36 \cdot \sqrt{\frac{T_g}{M_w}}$$

The above formula allows computing the new δ-sublayer concentration, if the values of C_{rev}, C_{CV}, and C_d, are known. The next parameter to be calculated is C_{rev}. The mass balance for the first of these parameters, C_{rev}, is (see Figure 12-84):

$$\frac{dC_{rev}}{dt} = \alpha \cdot (1-\theta) \cdot \beta \cdot u \cdot C_\delta - \vartheta \cdot C_{rev}$$

This equation is written using fully implicit numerical scheme, as:

$$\frac{C_{rev} - \overline{C_{rev}}}{\Delta t} = \alpha \cdot \left(1 - \frac{C_{rev}}{C_{\max}}\right) \cdot \beta \cdot u \cdot C_\delta - \vartheta \cdot C_{rev}$$

After transformations:

$$C_{rev} = \frac{\overline{C_{rev}} + \Delta t \cdot \alpha \cdot \beta \cdot u \cdot C_\delta}{1 + \Delta t \cdot (\alpha \cdot \beta \cdot u \cdot C_\delta / C_{\max} + \vartheta)}$$

With the above formula C_{rev} will never become larger than $C_{\max}$. However, C_{rev} may become slightly larger than $C_{\max}$ due to round-off errors (~10^{-16}). Therefore the following limit is imposed on the value obtained from the above formula:

$$C_{rev} \le C_{\max}$$

The Control Volume concentrations, C_{CV}, are obtained from the mass balance in Control Volume and the implicit inter-volume flow solution scheme, described in section 12.3.8. The concentrations inside the material, C_d, are obtained from the diffusion equation, described in section 12.3.5.6. With these values known all parameters that are needed to calculate the mass transfer rate on the wall surface are known.

The equations shown above are valid only for sorption of a single vapor. In case of multiple vapors one must take into account that some adsorption sites may be occupied by the molecules of the other vapors. Therefore the following method is adopted for general solution scheme. The value of $(1-\theta)$ for a given vapor, i, is given by:

$$(1-\theta) = \left(1 - \frac{C_{rev,i} + \sum_{j \ne i} C_{rev,j}}{C_{\max}}\right)$$

The term: $C_{\max} - \sum_{j \ne i} C_{rev,j}$ gives the number of sites available for the vapor class i: $C_{avl,i}$.

An option is available (see parameter ICSRRT, Volume 2) to use this equation or to use $C_{\max}$ for each vapor independently.

For simplicity the subscript *i* is skipped in the following discussion, i.e.: $C_{avl} = C_{avl,i} = C_{\max} - \sum_{j \neq i} C_{rev,j}$ The formula for C_δ:

$$C_\delta = \frac{h_\delta C_{CV} + \vartheta \cdot C_{rev} + \eta \cdot C_d}{h_\delta + \alpha \cdot \left(1 - \overline{\theta}\right) \cdot u}$$

is written using an explicit term: $\left(1 - \overline{\theta}\right) = (\overline{C_{avl}} - \overline{C_{rev}}) / C_{\max}$. The formula for C_{rev} is written using an implicit term for the current vapor and explicit for all other vapors.

$$\frac{C_{rev} - \overline{C_{rev}}}{\Delta t} = \alpha \cdot \left(\frac{\overline{C_{avl}}}{C_{\max}} - \frac{C_{rev,i}}{C_{\max}} \right) \cdot \beta \cdot u \cdot C_\delta - \vartheta \cdot C_{rev}$$

After transformations:

$$C_{rev} = \frac{\overline{C_{rev}} + \Delta t \cdot \alpha \cdot \beta \cdot u \cdot C_\delta \cdot \overline{C_{avl}} / C_{\max}}{1 + \Delta t \cdot (\alpha \cdot \beta \cdot u \cdot C_\delta / C_{\max} + \vartheta)}$$

If there is only a single vapor that is being sorbed, then $C_{avl} = C_{\max}$ and the above formula becomes identical to the formula for a single vapor sorption, shown above.

Because the number of available sites is calculated for each vapor using explicit value for other vapors, there is some time-step sensitivity of the obtained results. Test calculations have shown that if sorption of one of the vapors is strong compared to the others, then use of large time steps may result in underestimation of sorption of the weakly adsorbed vapors, as the strongly sorbed vapor tends to fill-in all the adsorption sites during a single time step. In order to prevent this from happening a limit is imposed on each vapor preventing it from filling more than a certain fraction (default value of $X_{\lim} = 0.99$ - see Volume 2) of the available sites during a single time step. With this limit the formula is:

$$C_{rev} = \frac{\overline{C_{rev}} + \Delta t \cdot \alpha \cdot \beta \cdot u \cdot C_\delta \cdot \overline{C_{avl}} \cdot X_{\lim} \cdot / C_{\max}}{1 + \Delta t \cdot (\alpha \cdot \beta \cdot u \cdot C_\delta / C_{\max} + \vartheta)}$$

Test calculations showed that with the applied $X_{\lim}$ there is very little time step sensitivity of the sorption results. The error introduced on a given vapor by using $X_{\lim}$ is only 1%, which is still quite small compared to the general accuracy of the sorption models.

Finally, the total mass transfer rate (kg/s-m^2), used to calculate the masses of molecules in the CV atmosphere and in the SC, is obtained from:

$$\left(\frac{dC_S}{dt} \right)_{total} = \alpha \cdot \left(1 - \theta\right) \cdot u \cdot C_\delta - \vartheta \cdot C_{rev} - \eta \cdot C_d$$

The mass transfer rate that is used as a source for the diffusion equation is given by:

$$\left(\frac{dC_S}{dt} \right)_{diff} = \alpha \cdot \left(1 - \theta\right) \cdot u \cdot C_\delta \cdot \left(1 - \beta\right) - \eta \cdot C_d$$

In the above formulae $\left(1 - \theta\right) = (\overline{C_{avl}} - C_{rev}) / C_{\max}$.

A user-defined multiplier, X1SRRT (for 1-D structures) and X2SRRT (for 2-D structures) is applied for the values obtained from the above formulae (see Volume 2).

12.3.5.6 Diffusion Inside Solid Materials

Diffusion results in transport of the penetrated molecules from the surface node to the nodes deeper in the material. Diffusion is calculated using the diffusion equation and the user supplied diffusion coefficient in solid structure materials, D_{CS}. If the diffusion coefficients are not specified, or are specified as zeroes, the diffusion is not calculated. In such case all the penetrated molecules are assumed to stay in the surface node. Diffusion calculation may consume significant fraction of computing time, therefore it is recommended to activate diffusion only when it is necessary.

The diffusion coefficients are specified in the material property data. The values must be specified in a consistent manner. This means if a positive diffusion coefficient is specified for one of the materials used by a given 1-D or 2-D Solid Heat Conductor, it must be specified as positive for all materials that are used to build this Solid Heat Conductor.

For a given material the diffusion coefficients are defined separately for each vapor class. The consistency mentioned above is valid within a single class only. Different vapor classes do not need to have the consistent diffusion coefficient. Therefore the user may for example specify a positive diffusion coefficient for one vapor class and zero for all other classes.

The diffusion coefficient is calculated from a temperature-dependent correlation:

$$D_{CS} = D_0 \cdot \exp\left(-\frac{E_D}{RT_i}\right) = D_0 \cdot \exp\left(-\frac{A_D}{T_i}\right)$$

E is the activation energy, (J), T_i is the temperature of node i (K), R is the universal gas constant. Limits may be imposed on the value obtained from the correlation: $D_{\min} \le D_{CS} \le D_{\max}$. The values of D_0, A_D, $D_{\min}$, $D_{\max}$, are user-defined input coefficients. For example, reference [121] gives for Cs in metals:

$$D_0 = 3 \times 10^{-9}\, cm^2/s = 3 \times 10^{-13}\, m^2/s$$
$$A_D = (E_D/R) = (15/1.987 \times 10^{-3}) = 7.55 \times 10^3\, K$$

Therefore:

$$D_{CS} = 3.0 \times 10^{-13} \cdot \exp\left(-\frac{7.55 \times 10^3}{T_i}\right)$$

In order to calculate the concentrations the following diffusion equation is solved:

$$\frac{\partial C_d}{\partial t} = \nabla(D_{CS} \cdot \nabla C_d) + S_V$$

S_V is an "external" source of molecules per unit volume, (kg/s-m^3), which include:

- Penetration: $+\alpha u(1-\beta)(1-\theta)N_s$, applicable for the surface node only
- Evaporation: $-\eta C_d$, applicable for the surface node only
- Source and removal by decay, applicable for all nodes, discussed later

The source term, S_d, is written explicitly, that is beginning of time step values are used. This is in practice sufficient because diffusion process is very slow. The decay is calculated using the general solution scheme applied for the decay chains, described in section 12.3.1. The calculated net source (or removal) of the diffusing isotope is then partitioned among all SC nodes proportional to the mass in each node.

Solution of the above diffusion equation is discussed below for the case of 1-D Solid Heat Conductors and 2-D Solid Heat Conductors.

- **Diffusion in 1-D Solid Heat Conductors**

In a 1-D structure, with the material properties depending on temperature, and the external source being a function of time, the diffusion equation is:

$$\frac{\partial C_d}{\partial t} = \frac{\partial}{\partial x}\left(D_{CS} \cdot \frac{\partial C_d}{\partial x} \right) + S_V$$

C_d is the concentration of the diffusing isotope (kg/m^3), D_{CS} is the diffusion coefficient in solid material (m^2/s), S_V is the external source per unit volume (kg/s-m^3). The external sources includes removal of the isotope due to its decay, source from decay of other isotopes present in the solid material, and sorption processes on the surface.

To solve the diffusion equation, a finite difference version of the equation is constructed. In order to do that, the derivatives are approximated by the finite differences. For the node i the time derivative is approximated as:

$$\frac{\partial C_d}{\partial t} = \frac{C_{d,i} - \overline{C_{d,i}}}{\Delta t}$$

Δt is the time step, and $\overline{C_{d,i}}$ is the mass concentration in the cell i at the beginning of the time step. Similarly the space derivative between the nodes i–1 and i may be approximated by:

$$\left(\frac{\partial C_d}{\partial x} \right)_{i-1,i} = \frac{C_{d,i} - C_{d,i-1}}{x_i - x_{i-1}}$$

Note that in the above approximation the end of time step temperatures are used: $C_{d,i}$, $C_{d,i-1}$. That means an implicit solution scheme is applied, which gives stable solution, independently of the time step size [21]. If the diffusion coefficient was constant between the nodes i–1 and i, the diffusion flux would be given simply by multiplying the derivative by D_{CS}: $J = -D_{CS} \cdot (C_{d,i} - C_{d,i-1})/(x_i - x_{i-1})$. However, material properties (and therefore the diffusion coefficient) may be different in different cells. The node-to-node diffusion flux sees two cell materials, which in general may have different properties. The diffusion flux, written from the node i–1 to the node i, is obtained by using a summed diffusion resistance:

$$J = -D_{CS} \cdot \left(\frac{\partial C_d}{\partial x} \right)_{i-1,i} = \frac{C_{d,i-1} - C_{d,i}}{\frac{d_{d,i-1}}{D_{CS,i-1}} + \frac{d_{d,i}}{D_{CS,i}}}$$

d_i half-thickness of the cell i for interior cells, full thickness for boundary cells, (m)

Similarly the flux is written for the diffusion between nodes i and i+1. After simple transformations the finite difference approximation of the diffusion equation takes the following form:

- Interior node, $1 < i < N$:

$$V_i \cdot \frac{C_{d,i} - \overline{C_{d,i}}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}} \cdot \left(C_{d,i-1} - C_{d,i} \right) + \frac{A_{R,i}}{R_{R,i}} \cdot \left(C_{d,i+1} - C_{d,i} \right) + S_i$$

$A_{L,i}$ heat transfer area at the left boundary of cell i, (m^2)
$A_{R,i}$ heat transfer area at the right boundary of cell i, (m^2)
V_i volume of cell i, (m^3)
S_i external source for cell i, (kg/s)
$R_{L,i}$ diffusion resistance between the node i–1 and i, (m)
$R_{R,i}$ diffusion resistance between the node i and i+1, (m)

The diffusion resistances, $R_{L,i}$ and $R_{R,i}$, are defined as:

$$\textit{left resistance} \quad R_{L,i} = \frac{d_{i-1}}{D_{CS,i-1}} + \frac{d_i}{D_{CS,i}}$$

$$\textit{right resistance} \quad R_{R,i} = \frac{d_{i+1}}{D_{CS,i+1}} + \frac{d_i}{D_{CS,i}}$$

- Left boundary node, $i = 1$:

$$V_i \cdot \frac{C_{d,i} - \overline{C_{d,i}}}{\Delta t} = \frac{A_{R,i}}{R_{R,i}} \cdot \left(C_{d,i+1} - C_{d,i} \right) + S_i$$

- Right boundary node, $i = N$:

$$V_i \cdot \frac{C_{d,i} - \overline{C_{d,i}}}{\Delta t} = \frac{A_{L,i}}{R_{L,i}} \cdot \left(C_{d,i-1} - C_{d,i} \right) + S_i$$

The above formulae present a set of N equations, with unknown concentrations, $C_{d,i}$. This equation set may be written shortly in a matrix form:

$$AC = B$$

The matrix ***A*** is a tridiagonal matrix, because the equations for internal nodes contain three unknown variables: $C_{d,i-1}$, $C_{d,i}$, and $C_{d,i+1}$. Therefore the matrix equation may be written as:

$$a_{i-1,i} \cdot C_{d,i-1} + a_{i,i} \cdot C_{d,i} + a_{i+1,i} \cdot C_{d,i+1} = b_i$$

The matrix coefficients, $a_{i,j}$, and the right-hand side terms, b_i, are equal to:

- Interior node, $1 < i < N$:

$$a_{i-1,i} = -\frac{A_{L,i}}{R_{L,i}}$$

$$a_{i+1,i} = -\frac{A_{R,i}}{R_{R,i}}$$

$$a_{i,i} = \frac{V_i}{\Delta t} + \frac{A_{L,i}}{R_{L,i}} + \frac{A_{R,i}}{R_{R,i}}$$

$$b_i = \frac{V_i \overline{C_{d,i}}}{\Delta t} + S_i$$

- Left boundary node, $i = 1$:

$$a_{i+1,i} = -\frac{A_{R,i}}{R_{R,i}}$$

$$a_{i,i} = \frac{V_i}{\Delta t} + \frac{A_{R,i}}{R_{R,i}}$$

$$b_i = \frac{V_i \overline{C_{d,i}}}{\Delta t} + S_i$$

- Right boundary node, $i = N$:

$$a_{i-1,i} = -\frac{A_{L,i}}{R_{L,i}}$$

$$a_{i,i} = \frac{V_i}{\Delta t} + \frac{A_{L,i}}{R_{L,i}}$$

$$b_i = \frac{V_i \overline{C_{d,i}}}{\Delta t} + S_i$$

The tridiagonal matrix is solved using the procedure specifically suitable for this type of matrices (section 17.4).

The diffusion model described above is quite general and accurate in practical applications. Below an example problem is shown, where the model results are compared to analytical solutions of the diffusion equation.

- *Example Problem: Transient Diffusion in a Semi-Infinite Slab*

  Transient diffusion in a semi-infinite slab with a step change concentration of the diffusive molecules at the surface is considered. The initial concentrations are C_0 = 0.0 kg/m^3. At time equal to zero the surface concentration is set to 0.001 kg/m^3. The theoretical solution of the diffusion equation is analogous to the heat conduction equation, with the thermal diffusivity replaced by the diffusion coefficient. This means:

  - $a = k/(\rho c_p)$, thermal diffusivity, (m^2/s) $\rightarrow$ D, diffusion coefficient, (m^2/s)

  The result is based on the solution presented in [20], chapter 3, equation 55 (see section 5.2):

$$C(x,t) = C_w \cdot erfc\left(\frac{x}{2\sqrt{Dt}}\right)$$

  $C(x,t)$ concentration at location x from the left boundary, at time t, (kg/m^3)
  C_w concentration at the boundary surface, (=0.001 kg/m^3)
  D diffusion coefficient, (=1×10^{-6} m^2/s)

  Calculations were performed using a solid heat conductor with 0.01 m thickness. As shown below, this thickness is sufficient to represent a semi-infinite slab for the present, short term calculations. The heat conductor was nodalized using 21 nodes - internal nodes of 5.0×10^{-4} m (half millimeter) and boundary nodes of 2.5×10^{-4} m - Figure 5-4. At the left surface an adsorption flux is specified using a user-defined Control Function. The Control Function is defined in such a way as to keep a constant concentration of 0.001 kg/m^3 at the surface node.

  Results are shown in Figure 12-91 through Figure 12-95. Figure 12-91 through Figure 12-94 show visualization pictures of SPECTRA results at times 1.0, 5.0, 10.0, and 20.0 s.

  Figure 12-95 shows comparisons of SPECTRA results to analytical solution obtained for a semi-infinite slab. The calculated values are in good agreement with the theoretical values, except near the right boundary, at t = 20 s (Figure 12-95). At that time the diffusive isotope has penetrated into the right side of the conductor and the analytical solution, obtained for a semi-infinite slab, becomes a bad approximation of the real geometry in the region close to the right boundary.

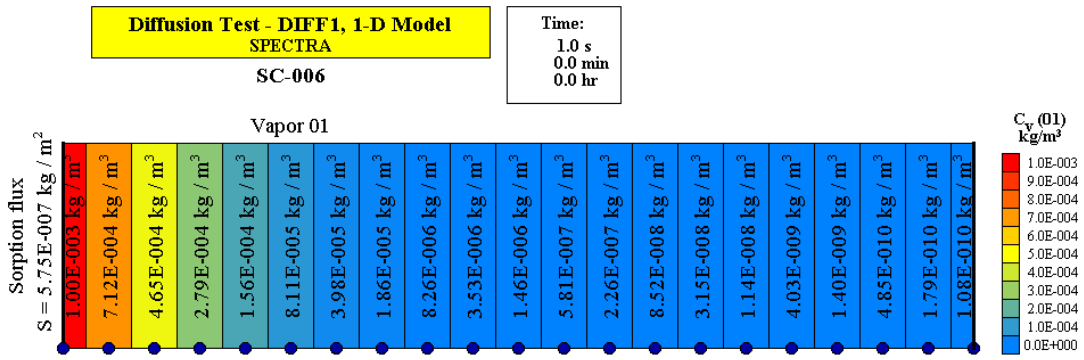


Figure 12-91 Diffusion test, constant boundary concentration, SPECTRA, t=1.0 s.

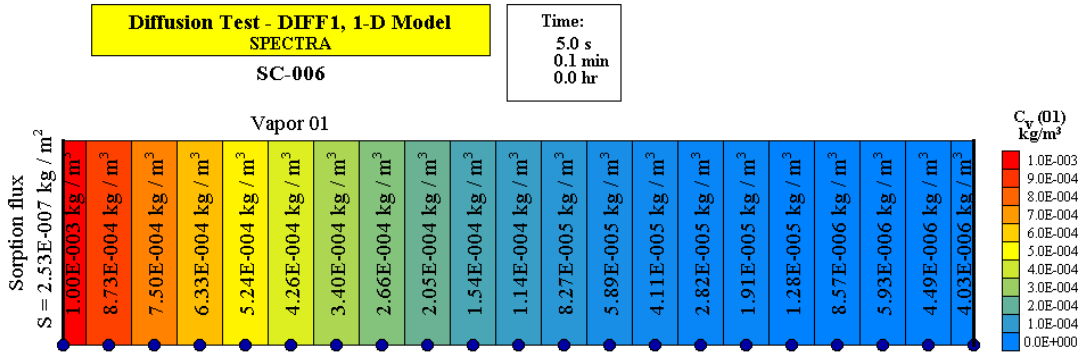


Figure 12-92 Diffusion test, constant boundary concentration, SPECTRA, t=5.0 s.

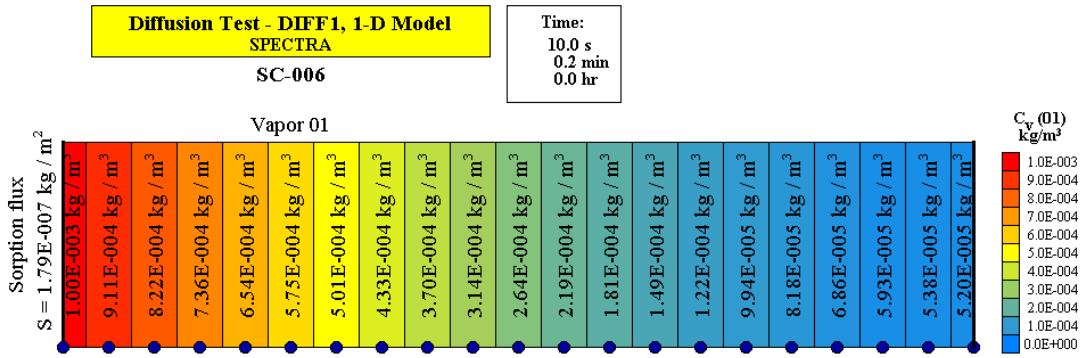


Figure 12-93 Diffusion test, constant boundary concentration, SPECTRA, t=10.0 s.

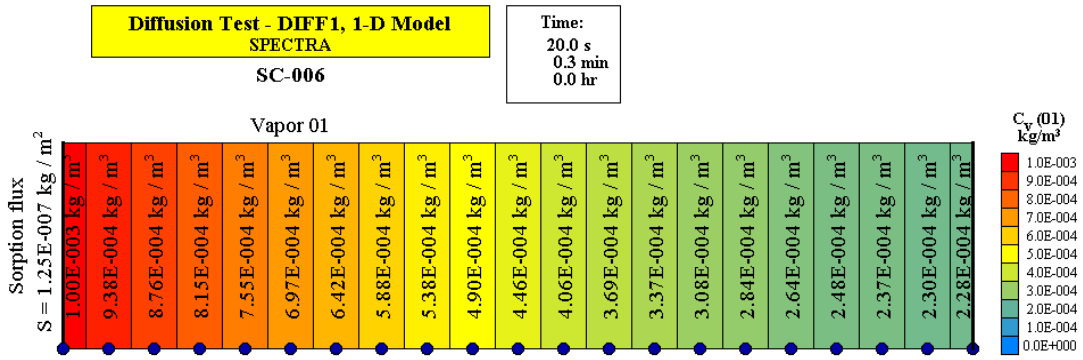


Figure 12-94 Diffusion test, constant boundary concentration, SPECTRA, t=20.0 s.

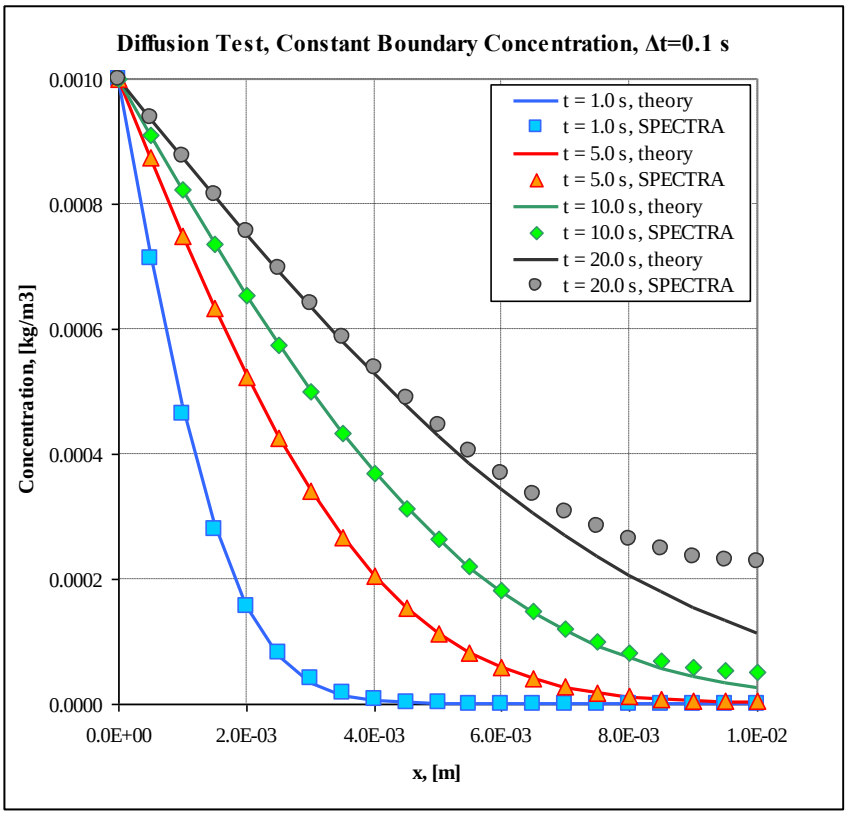


Figure 12-95 Diffusion test, constant boundary concentration, theory and SPECTRA.

- **Diffusion in 2-D Solid Heat Conductors**

In case of a 2-D structure, the diffusion equation is:

$$\frac{\partial C_d}{\partial t} = \frac{\partial}{\partial x}\left(D_{CS} \cdot \frac{\partial C_d}{\partial x} \right) + \frac{\partial}{\partial y}\left(D_{CS} \cdot \frac{\partial C_d}{\partial y} \right) + S_V$$

The finite difference approximation of the diffusion equation in 2-D is analogous to the 1-D equation. The only difference is the presence of additional diffusion fluxes from the "up" and the "down" cells. The equation takes the following form:

$$V_i \cdot \frac{C_{d,i} - \overline{C_{d,i}}}{\Delta t} =$$
$$= \frac{A_{L,i}}{R_{L,i}} \cdot \left(C_{d,L} - C_{d,i} \right) + \frac{A_{R,i}}{R_{R,i}} \cdot \left(C_{d,R} - C_{d,i} \right) +$$
$$+ \frac{A_{U,i}}{R_{U,i}} \cdot \left(C_{d,U} - C_{d,i} \right) + \frac{A_{D,i}}{R_{D,i}} \cdot \left(C_{d,D} - C_{d,i} \right) + S_i$$

$C_{d,L,\,i}$ concentration in the cell left to the cell *i*, (kg/m^3)
$C_{d,R,\,i}$ concentration in the cell right to the cell *i*, (kg/m^3)
$C_{d,U,\,i}$ concentration in the cell up of the cell *i*, (kg/m^3)
$C_{d,D,\,i}$ concentration in the cell down of the cell *i*, (kg/m^3)
S_i external source for the cell *i*, (kg/s), (due to decay or sorption processes)

Analogically to the 1-D case, the resistances are written:

$$\textit{left resistance} \quad R_{L,i} = \frac{d_L}{D_{CS,L}} + \frac{d_i}{D_{CS,i}}$$
$$\textit{right resistance} \quad R_{R,i} = \frac{d_R}{D_{CS,R}} + \frac{d_i}{D_{CS,i}}$$
$$\textit{up resistance} \quad R_{U,i} = \frac{d_U}{D_{CS,U}} + \frac{d_i}{D_{CS,i}}$$
$$\textit{down resistance} \quad R_{D,i} = \frac{d_D}{D_{CS,D}} + \frac{d_i}{D_{CS,i}}$$

The subscripts *L, R, U, D,* stand for the cell left of the cell *i*, right of the cell *i*, up of the cell *i*, and down of the cell *i*. The numbers of these cells depend on the internal cell numbering scheme. SPECTRA numbers the nodes either horizontally or vertically, in order to minimize the number of operations required to solve the matrix, i.e. to obtain small band width.

The above formulae present a set of N equations, with unknown concentrations, $C_{d,i}$. This equation set may be written shortly in a matrix form:

$$AC = B$$

In contrast to the 1-D case the matrix $\boldsymbol{A}$ is not a tridiagonal but a band-diagonal matrix. The matrix coefficients, $a_{i,j}$, and the right-hand side terms, b_i, are equal to:

$$a_{L,i} = -\frac{A_{L,i}}{R_{L,i}} \qquad a_{R,i} = -\frac{A_{R,i}}{R_{R,i}}$$

$$a_{U,i} = -\frac{A_{U,i}}{R_{U,i}} \qquad a_{D,i} = -\frac{A_{D,i}}{R_{D,i}}$$

$$a_{i,i} = \frac{V_i}{\Delta t} + \frac{A_{L,i}}{R_{L,i}} + \frac{A_{R,i}}{R_{R,i}} + \frac{A_{U,i}}{R_{U,i}} + \frac{A_{D,i}}{R_{D,i}}$$

$$b_i = \frac{V_i \overline{C_{d,i}}}{\Delta t} + S_i$$

The above terms are written separately for each quarter-cell and then summed up over all quarter-cells existing in a given cell. Types of cells are shown in Figure 12-96 (left). Examples of 2-D structures are shown in Figure 12-96 (right).

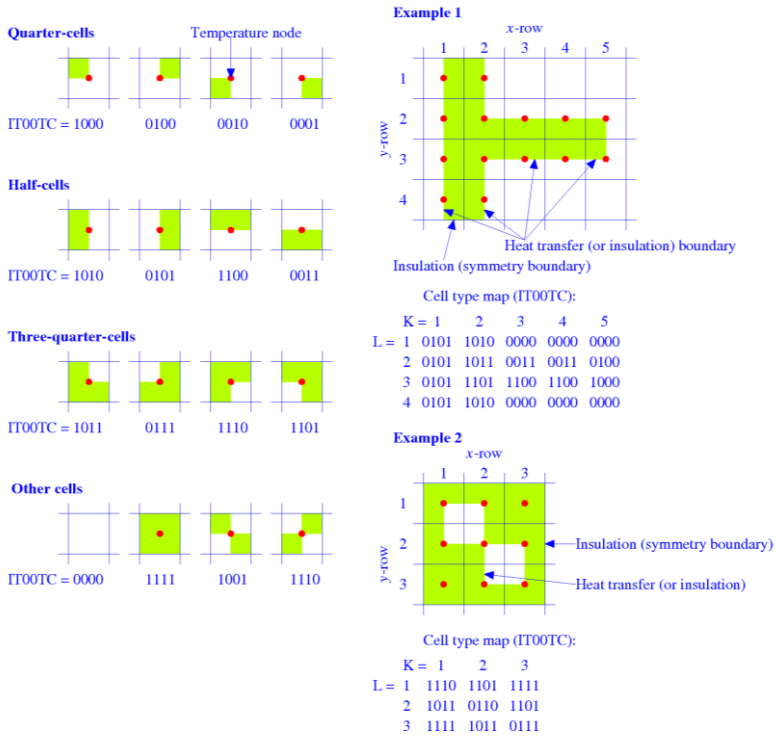


Figure 12-96 Types of 2-D cells (left) and examples of 2-D structures (right).

The band-diagonal matrix is solved using the procedure specifically suitable for this type of matrices (section 17.4).

- **Source and Removal due to Radioactive Decay**

Ideally diffusion of each isotope inside the solid structures should be considered separately. With this treatment one would in general need to reserve memory for all isotopes diffusing through all nodes of the 1-D and 2-D Solid Heat Conductors. This would result in enormous memory requirement. Therefore in SPECTRA diffusion is calculated for the vapor classes rather than isotopes. This allows considerable reduction of memory requirement, since the code has to keep track of up to 20 vapor classes rather than up to 220 isotopes, in each node of a 1-D or a 2-D structure.

Since the vapor classes group similar isotopes, this treatment is physically correct, but it introduces certain difficulty in calculating the sources and removal rates due to radioactive decay. This is explained below.

- The vapor classes are tracked for each node, therefore the available parameter is: Concentration of the diffusing molecules of the vapor class *VK*, in the cell number *i* of a 1-D or a 2-D Solid Heat Conductor, $C_{d,i,VK}$, (kg/m^3).

- The isotopes are tracked per structure, therefore the available parameter is: Total number, N_K (-), or total mass, m_K (kg), of the isotope number *K*, in all cells of a 1-D or a 2-D Solid Heat Conductor.

As a consequence, when isotope *K* decays, the code "does not know" in which node the decay process occurred. This is a straightforward consequence of the decision of saving computer memory and writing the diffusion equation for vapor classes rather than isotopes. In order to solve the diffusion equation the code needs to "guess" where the decay process occurred. Probability that the decay occurred in the cell *i* is of course equal to the mass fraction of the isotope *K* in the cell *i* (mass of the isotope *K* in the cell *i* divided by the total mass of the isotope *K* in all cells). Since the vapor classes group similar isotopes (for example I-131, I-133, I-135, etc.), they do behave (diffuse) in a similar way. Therefore it is reasonable to assume that the fraction of the isotope *K* in the cell *i* is proportional to the fraction of the vapor class *VK* in the same node. With this assumption it may be written:

$$\frac{N_{i,K}}{N_K} \cong \frac{m_{i,VK}}{\sum_j m_{j,VK}} = \frac{m_{i,VK}}{m_{VK}} = \frac{C_{d,i,VK} \cdot V_{cell,i}}{m_{VK}}$$

$N_{i,K}$ number of molecules of the isotope number *K* in the cell number *i*, (-)
N_K number of molecules of the isotope number *K* in all cells, (-)
$m_{i,VK}$ mass of the fission product vapor class *VK* in the cell number *i*, (kg)
m_V total mass of the fission product vapor class *VK* in all cells, (kg)
$C_{d,i.VK}$ concentration of the fission product vapor class *VK* in the cell number *i*, (kg/m^3)
$V_{cell,i}$ volume of the cell number *i*, (m^3)

The summation is performed over all nodes of the 1-D or 2-D Solid Heat Conductor. This provides a recipe for the guessed value of the amount of isotope *K* concentration in the node *i*, $N_{i,K}$:

$$N_{i,K} \cong N_K \cdot \frac{C_{d,i,VK} \cdot V_{cell,i}}{m_{VK}} \qquad m_{VK} = \sum_j C_{d,j,VK} \cdot V_{cell,j}$$

Mass source of the vapor class *VK* due to radioactive decay is written by summing up the isotopes that are being removed from the class *VK* due to the radioactive decay on one hand (negative source), and summing up the isotopes that are coming into the class *VK* due to the radioactive decay of isotopes that are not members of this class (positive source). The source needs to be expressed in kg/m^3, therefore the molecule numbers, *N*, are converted to the mass concentrations using the multiplier of (M_w/N_A). Therefore the source of the vapor class *VK* for node *i* due to radioactive decay is given by:

$$S_{i,VK} = -\sum_{K \in VK} \left(\lambda_K \cdot N_{i,K} \cdot \frac{M_{w,K}}{N_A} \cdot \gamma_{K \to L} \right) + \sum_{L \in VL \neq VK} \left(\lambda_L \cdot N_{i,L} \cdot \frac{M_{w,L}}{N_A} \cdot \gamma_{L \to K} \right)$$

$S_{i,VK}$ source of the fission product vapor class number *VK*, in the cell number *i* of a 1-D or a 2-D Solid Heat Conductor, (kg/s)
$K \in VK$ isotope number *K*, a member of the fission product vapor class number *VK*
$L \in VL$ isotope number *L*, a member of the fission product vapor class number *VL*, other than the class *VK*, $VL \neq VK$
λ_K decay constant of the isotope number *K*, (s^{-1})
$N_{i,K}$ number of molecules of the isotope number *K* in the cell number *i*, (-)
$\gamma_{K \to L}$ yield fraction, probability of isotope *L* being created from decay of the isotope *K*, (-)
$M_{w,K}$ molar weight of the isotope number *K*, (kg/kmol)
N_A Avogadro number

The guessed values of $N_{i,K}$ and $N_{i,L}$ are substituted to the above formula, the final result is:

$$S_{i,VK} = -\sum_{K \in VK} \left(\lambda_K \cdot N_K \cdot \frac{M_{w,K}}{N_A} \cdot \gamma_{K \to L} \cdot \frac{C_{d,i,VK} \cdot V_{cell,i}}{m_K} \right) + \\ + \sum_{L \in VL \neq VK} \left(\lambda_L \cdot N_L \cdot \frac{M_{w,L}}{N_A} \cdot \gamma_{L \to K} \cdot \frac{C_{d,i,VL} \cdot V_{cell,i}}{m_L} \right)$$

The first term represents removal of the vapor class *VK* from the node *i* due to decay of all radioactive isotopes *K* that are members of the class *VK*, into isotopes *L* that are members of any class *VL* other than *VK*. Similarly, the second term represents the source of the vapor class *VK* in the node *i* due to decay of all radioactive isotopes *L* that are members of any class *VL* other than *VK*, into isotopes *K* that are members of *VK*.

An example problem shown below, illustrates the calculation procedure. The test case consists of the same 1-D Solid Heat Conductor as that used for the previous example (see Figure 12-91 through Figure 12-95 and the corresponding text). In the present case there are two isotopes:

- Isotope A, a non-diffusive, long life isotope decaying into B
- Isotope B, a diffusive, non-radioactive isotope

The isotope A is present in the left surface node. Here it provides a continuous source of the isotope B, due to radioactive decay. Since the isotope A is a long life (half-life of 1 month was assumed, $\lambda_A = \ln(2)/(1\times31\times24\times3600) = 2.59\times10^{-7}$ s^{-1}) the source of isotope B is practically constant, and equal to $N_A\lambda_A$, where N_A is the concentration of the isotope A.

The initial mass of isotope A was set to $m_A = 0.01$ kg. Both isotopes A and B are assumed to have the same molar weight. Therefore the source for the diffusion equation of the isotope B may be written as $m_A\lambda_A$ (in general the mass source for the isotope B is equal to: $m_A\times M_{w,B}/M_{w,A}$, where $M_{w,B}$, $M_{w,A}$ are the molar weights of B and A respectively). The diffusion coefficients were assumed in the calculations as: $D_A = 10^{-20}$ (m^2/s) for the non-diffusive isotope A (zero could be used as well to eliminate completely diffusion of the isotope A) and $D_B = 10^{-6}$ (m^2/s) for the diffusive isotope B.

From the point of view of the diffusive isotope B, this case represents a constant source in the surface node. Solution of analogous heat transfer case, with constant heat flux at the surface is presented in [21] (section 3.4.2, equation 3.60).

Using an analogy between the heat conduction and diffusion, the solution is written here by replacing the thermal diffusivity by the the diffusion coefficient and the surface heat flux by the source due to decay of the isotope A:

- $a = k/(\rho c_p)$, thermal diffusivity, (m^2/s) $\rightarrow D_B$, diffusion coefficient, (m^2/s)
- q, boundary heat flux, (W/m^2) $\rightarrow m_A\lambda_A$ mass source, (kg/s)

Furthermore, in order to obtain the diffusion equation of exactly the same form as the conduction equation, the volumetric heat capacity (ρc_p) in the conduction equation must be equal to 1.0. Consequently the thermal conductivity is equal to D_B:

- $(\rho c_p) = 1.0$
- $k = D_B$

The analytical solution is (see [21], section 3.4.2, equation 3.60):

$$C_B(x,t) = \frac{m_A\lambda}{D_B}\cdot\left[\left(\frac{4D_Bt}{\pi}\right)^{1/2}\cdot\exp\left(-\frac{x^2}{4D_Bt}\right) - x\cdot erfc\left(\frac{x}{\sqrt{4D_Bt}}\right)\right]$$

Results are shown in Figure 12-97 through Figure 12-102. Figure 12-97 through Figure 12-101 show visualization pictures of SPECTRA results at times 0.0 s (starting time), 1.0, 5.0, 10.0, and 20.0 s. Figure 12-102 shows comparisons of SPECTRA results to analytical solution obtained for a semi-infinite slab.

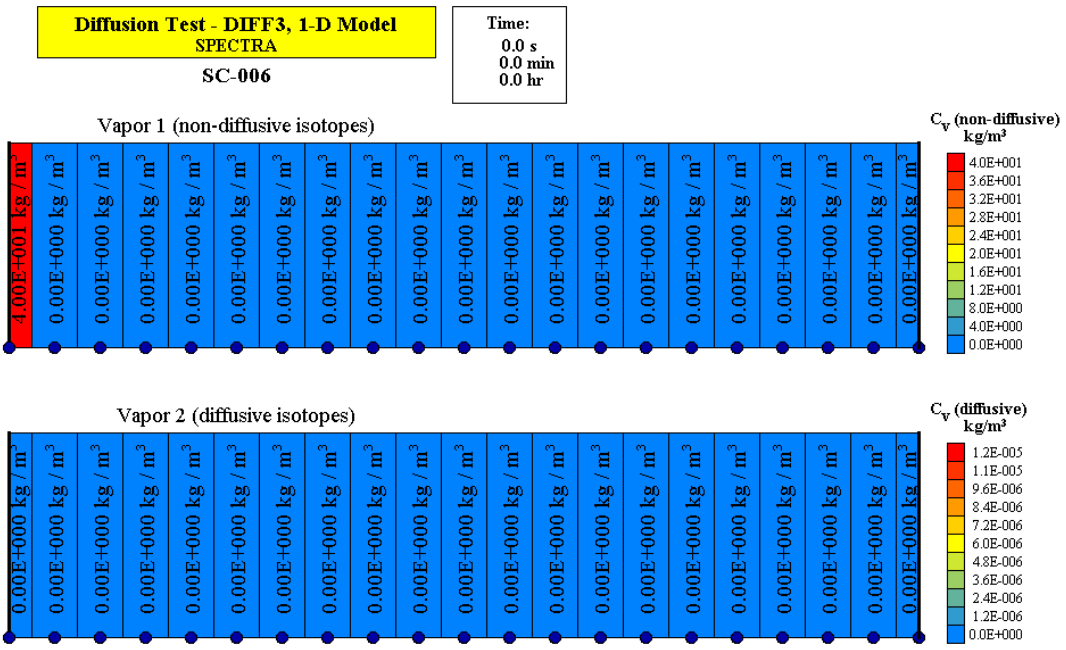


Figure 12-97 Decay of a non-diffusive into a diffusive isotope, SPECTRA, t=0.0 s.

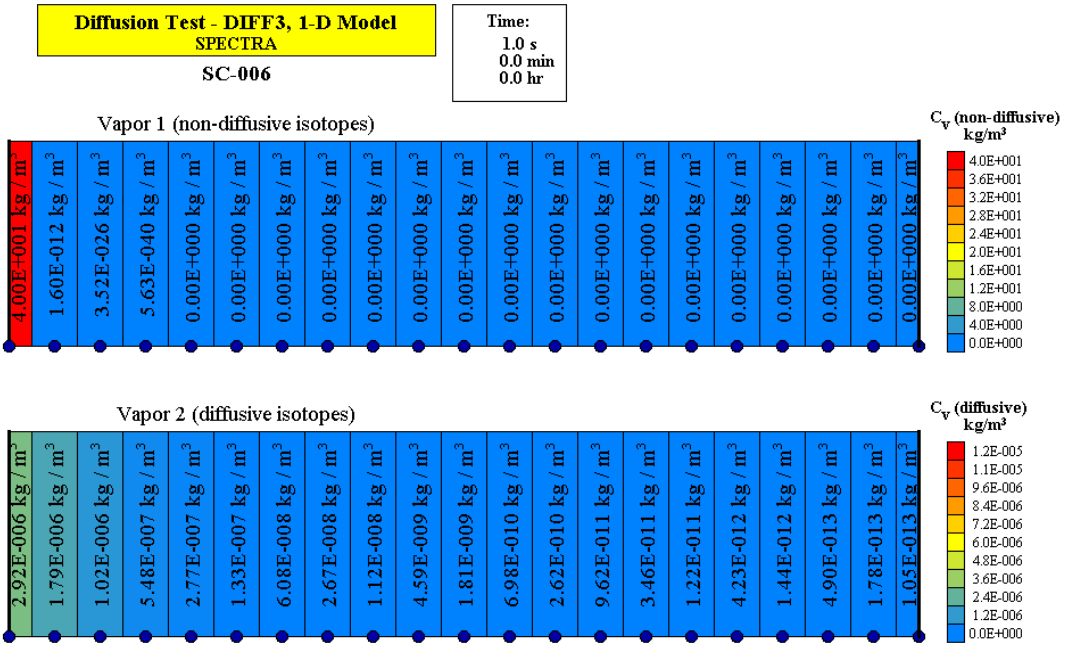


Figure 12-98 Decay of a non-diffusive into a diffusive isotope, SPECTRA, t=1.0 s.

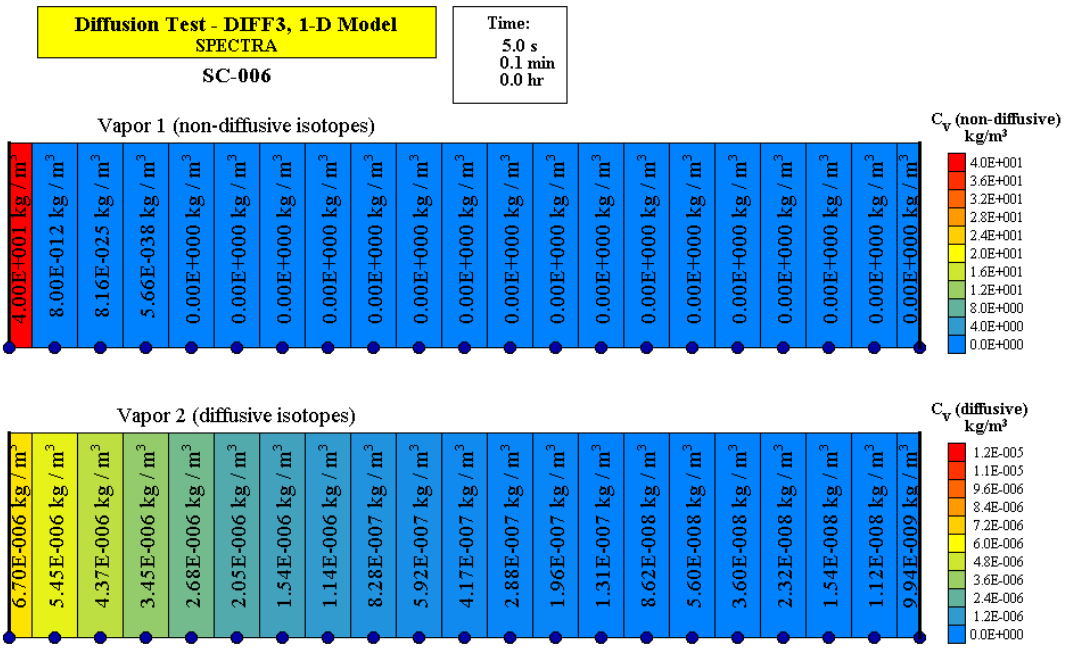


Figure 12-99 Decay of a non-diffusive into a diffusive isotope, SPECTRA, t=5.0 s.

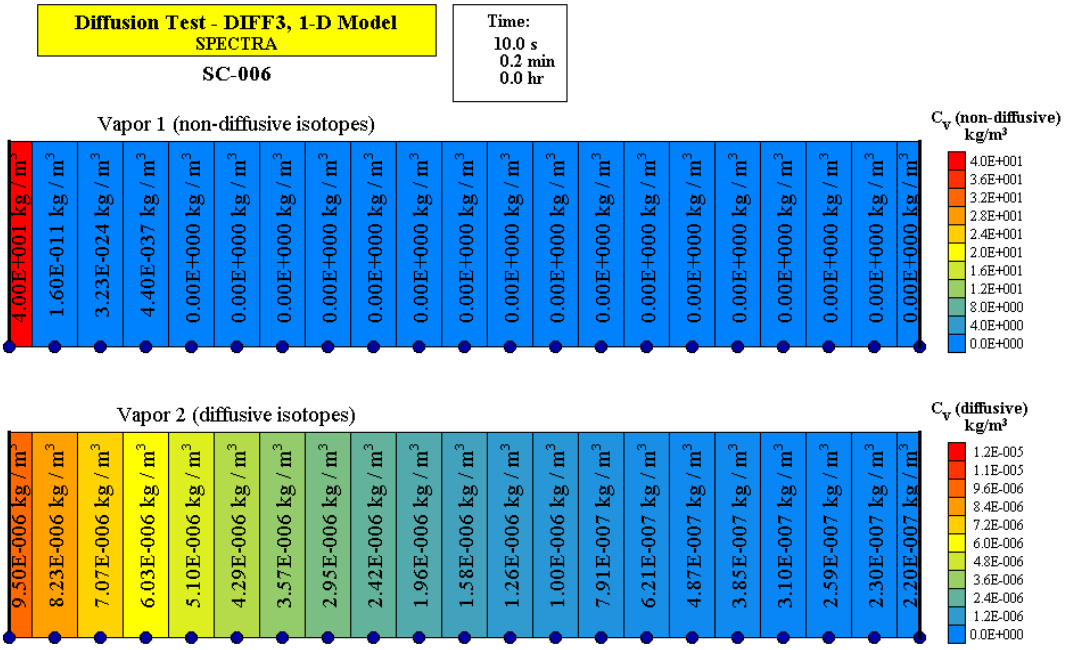


Figure 12-100 Decay of a non-diffusive into a diffusive isotope, SPECTRA, t=10.0 s.

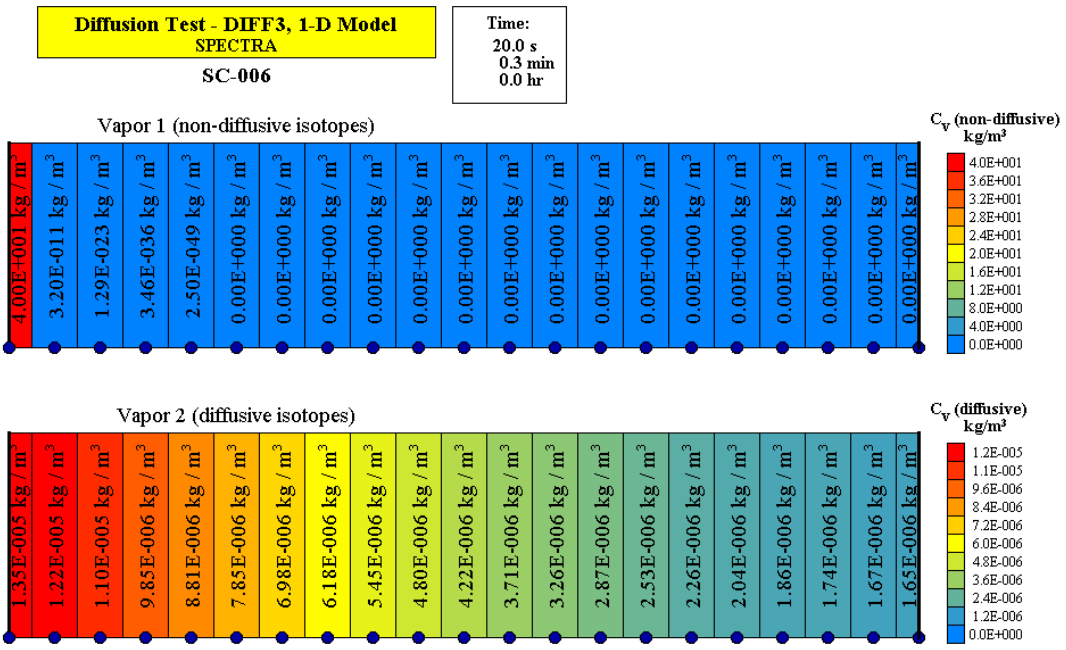


Figure 12-101 Decay of a non-diffusive into a diffusive isotope, SPECTRA, t=20.0 s.

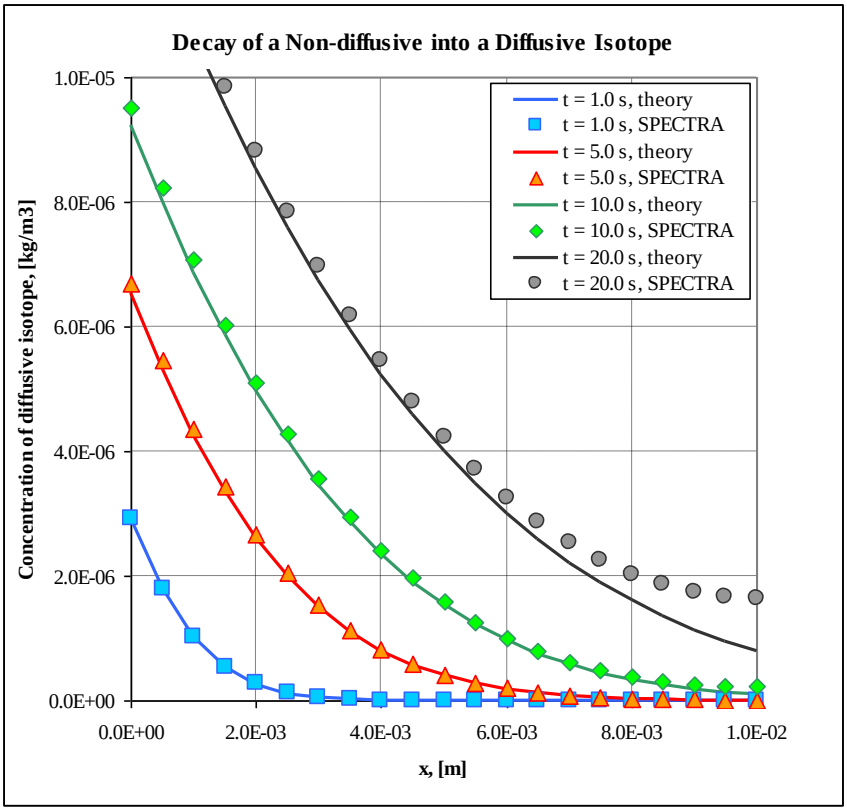


Figure 12-102 Decay of a non-diffusive into a diffusive isotope, theory and SPECTRA.

- **Source and Removal due to Sorption**

Sorption provides a source for boundary cells (left and right boundary cells in case of 1-D Solid Heat Conductors, and all boundary cells of 2-D Solid Heat Conductors). The source for the boundary cells is equal to the sorption flux multiplied by the heat transfer area:

$$S_i = \left(\frac{dC_S}{dt} \right)_{diff} \cdot A \cdot (1 - X_{pool})$$

S_i	source due to sorption for the cell *i* (boundary cell), (kg/s)
$(dC_S/dt)_{diff}$	sorption flux for diffusion calculation, (kg/s-m^2)
A	total heat transfer area of the Solid Heat Conductor boundary cell, (m^2)
X_{pool}	fraction of the boundary cell surface covered by water, (-)

If the user-defined Control Function (section 12.3.5.3) or the simple sorption model (section 12.3.5.4) is used, then the sorption flux for diffusion calculation is equal to the total sorption flux. In case of full adsorption model these fluxes are in general different, as described in section 12.3.5.5:

$$\left(\frac{dC_S}{dt} \right)_{diff} = \left(\frac{dC_S}{dt} \right)_{total}$$ user-defined CF or sorption model 1

$$\left(\frac{dC_S}{dt} \right)_{diff} \neq \left(\frac{dC_S}{dt} \right)_{total}$$ sorption model 2

12.3.5.7 Summary of Sorption Models

Three models are available. The sorption rate, expressed as the mass of adsorbed material per second, per unit surface area, is calculated as follows.

- User-defined model:

$$\left(\frac{dC_S}{dt}\right)_{total} = \left(\frac{dC_S}{dt}\right)_{diff} = CF$$

  The model may be applied for either gas-covered or liquid-covered surfaces, depending on the user-defined selector (ISRLRT, see Volume 2).

- Sorption model 1:

$$\left(\frac{dC_S}{dt}\right)_{total} = \begin{cases} A_S(T)\cdot C_V^{x_A} - B_S(T)\cdot C_d & for\ C_V \le C_{sat}(T) \\ A_S(T)\cdot C_V^{x_{red}}\cdot\left(\frac{C_{sat}^{x_A}}{C_{sat}^{x_{red}}}\right) - B_S(T)\cdot C_d & for\ C_V > C_{sat}(T) \end{cases}$$

  The model may be applied for either gas-covered or liquid-covered surfaces, depending on the user-defined selector (ISRLRT, see Volume 2).

- Sorption model 2:

$$\left(\frac{dC_S}{dt}\right)_{total} = \alpha\cdot(1-\theta)\cdot u\cdot C_\delta - \vartheta\cdot C_{rev} - \eta\cdot C_d$$

$$\left(\frac{dC_S}{dt}\right)_{diff} = \alpha\cdot(1-\theta)\cdot u\cdot C_\delta\cdot(1-\beta) - \eta\cdot C_d$$

  The model is applicable for gas phase only. In case of liquid-covered surfaces, only desorption of the reversibly-bound particles is possible (APSRRT > 0.0, see Volume 2).

A user-defined multiplier, X1SRRT (for 1-D structures) and X2SRRT (for 2-D structures), is applied for the values obtained from any of the three models (see Volume 2).

12.3.6 Leaching Mechanism and Modeling

12.3.6.1 Introduction

Leaching of chromium from alloys at high temperatures and in the presence of salt was observed in the Molten Salt Reactor Experiment (MSRE). Chromium was selectively removed from the Hastelloy N alloy in high-temperature regions and deposited in low-temperature regions [210]. Chromium depletion was expected to a depth of less than 0.13 mm/year in metal at 704°C. The mechanism and mathematical model of chromium leaching from Hastelloy N is described in [211].

Hastelloy N (UNS N10003), a Ni-based alloy that was originally developed during the MSRE program at the ORNL specifically for a combination of corrosion resistance in molten fluoride salts and air-side oxidation resistance. Hastelloy N is among the structural materials being considered for the Fluoride salt-cooled High-temperature Reactor (FHR). One experiment in the MSRE program was a flow loop made of UNS N10003, with a fuel-bearing molten fluoride salt (FLiBe) that operated successfully for 9.2 years in the temperature range of 560°C (cold section) to 700°C (hot section). Examination of the inner surface of the flow loop after this long-term test showed a Cr-depleted attack depth of 100 μm in the hot section and the deposition of Cr on the colder section of the loop.

Reference [211] presents experimental results and a mathematical model based on the assumption of diffusion-controlled corrosion for UNS N10003 in a nonfuel bearing FLiBe salt. The removal of Cr is through an outward diffusion in the alloy. The effective diffusion coefficient was calculated to be 8.72×10^{-19} m^2/s at 700°C, which is lower than for UNS N10003 in fueled molten fluoride salt (containing 1%UF_4), ~2.9×10^{-18} m^2/s [211]. This indicated that the corrosion in nonfuel bearing FLiBe is significantly slower than in UF_4 dissolved FLiBe. Cr concentration profiles obtained with this model for 1,000 hours and 365 days are shown in Figure 12-103.

12.3.6.2 Leaching Model

In order to use the leaching model in SPECTRA, the user must do the following:

- Define an isotope that is being leached (using the fission product chains)
- Activate the leaching option on the surface of a SC and define the initial concentration of the isotope.
- Define the modeling parameters, discussed below.

Leaching can be modeled in two ways.

- A user-defined Control Function model,
- Diffusion-controlled model, based on the model presented in [211].

Control Function (CF) model. The model may be applied to define the mass flux at the surface, using the general framework of the sorption models, described in section 12.3.5 (the CF is discussed in section 12.3.5.3). When the leaching model is applied (defined in records 385XXX - see Volume 2), then the value obtained from the CF is restricted to negative values (only removal is possible) and the absolute value is restricted to a maximum of 10^{-6} kg/m^2-s.

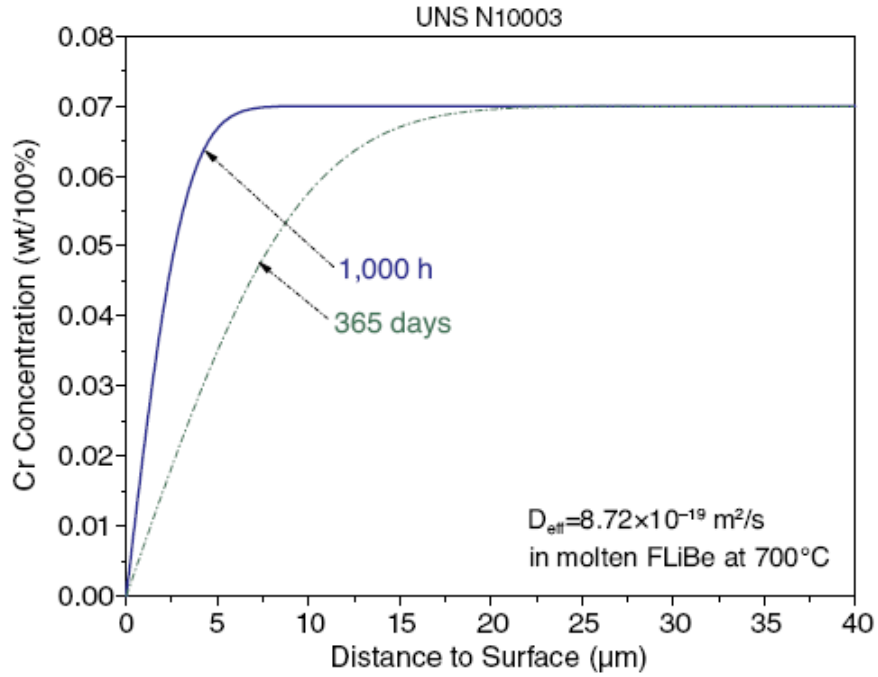


Figure 12-103 Cr concentration profiles, Hastelloy N (UNS N10003) in FLiBe at 700°C [211]

Diffusion-controlled model. The model is basically the same as the one used in [211]. The user must supply the diffusion coefficient for the vapor class to which the leached isotope belongs. The diffusion coefficient may be a function of temperature. Two stages may be distinguished.

- **Stage I**. During this stage the concentration on the surface (in the boundary surface node) decreases from the initial value to ~0.0. The following formula is used:

$$S_I = \frac{C_0 \cdot \Delta x_0}{T_0}$$

Here S_I is the leaching mass flux (kg/m^{2}s), C_0 is the concentration of the leached isotope in the boundary surface node (kg/m^3), Δx_0 is the thickness of the boundary surface node (m), and T_0 is the characteristic time (s) (input parameter T0LCRT, see Volume 2). The default value of T_0 is 1000 s. The effect of T_0 on the results is discussed below and in Volume 3. If the diffusion from the neighboring node is negligible, then the solution of the above equation gives:

$$C_0(t) \approx C_0(0) \cdot \exp\left(-\frac{t}{T_0}\right)$$

In such case the concentration in the boundary node would be ~0.0 at time equal to ~5T_0. With simultaneous diffusion from the neighboring node it takes somewhat longer, in the considered example roughly 10T_0 as seen in Figure 12-107.

- **Stage II**. During this stage, the concentration on the surface is approximately equal to zero. The mass flux is controlled by the diffusion from the internal nodes. The mass flux is approximately equal to:

$$S_{II} = -D_C \cdot \frac{C_1}{\Delta x_{01}}$$

Here C_1 is the concentration of the leached isotope in the node next to the surface (kg/m^3), Δx_{01} is the distance between the surface and the next node (m), and D_C is the diffusion coefficient (m^2/s).

The diffusion-controlled leaching model was tested using the values calculated in [211]. The stable element Cr-52 was defined (isotope 191). The wall is represented by SC-006; the fluid volume by CV-006. The wall material is Hastelloy N. The diffusion coefficient was defined as: $D_C = 8.72\times10^{-19}$ m^2/s. The wall temperature and the fluid temperature are set to 973 K (700°C). The base node size was 2 μm for the depth of about 50 μm, with a boundary surface node of 1.0 μm.

Calculations were performed for 365 days (3.1536×10^7 s). The applied time step was Δt = 10.0 s. Lack of time step sensitivity was checked by applying time steps between 5 s and 100 s. Lack of nodalization sensitivity was checked by varying the size of the base node size between 1 μm and 10 μm. Sensitivity calculations are discussed in more detail in Volume 3.

12.3.6.3 Results

Results are shown in Figure 12-104 through Figure 12-108. Figure 12-104 shows the visualization of the Cr-52 concentrations at the end of the calculation period (365 days). Figure 12-105 shows the calculated concentration profiles at the times of 1000 hours and 365 days. The values calculated by SPECTRA were superimposed on the graph obtained from [211]. A very good agreement is observed between the SPECTRA values and the source data.

Effect of T_0

Figure 12-106 shows the concentration profiles in the two stages. The results were obtained using the default value of the Stage I characteristic time T_0 = 1000 s and node size of 2 μm. During Stage I the concentrations on the surface decrease to zero. During Stage II the concentration at the surface remains as ≈0.0 and the mass transfer is limited by the diffusion from the inner nodes.

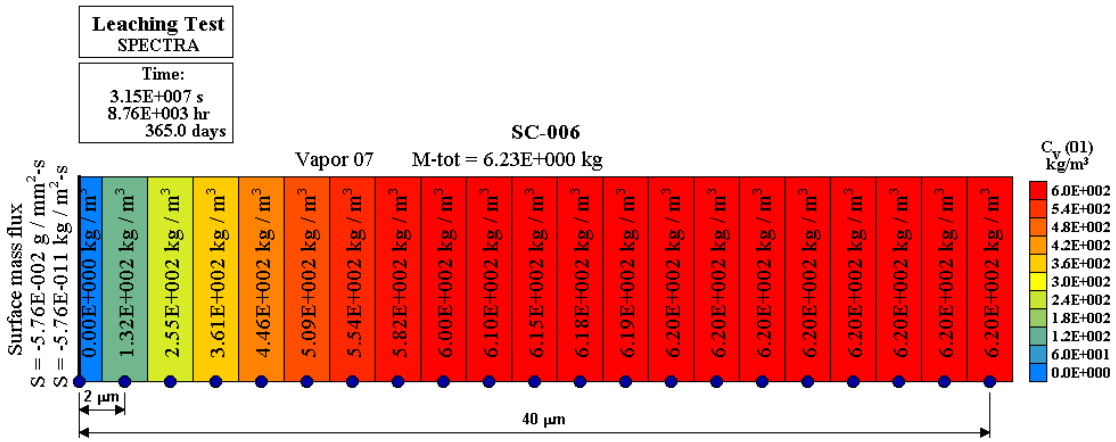


Figure 12-104 Cr concentrations, Hastelloy N in FLiBe at 700°C, SPECTRA model, t = 365 days

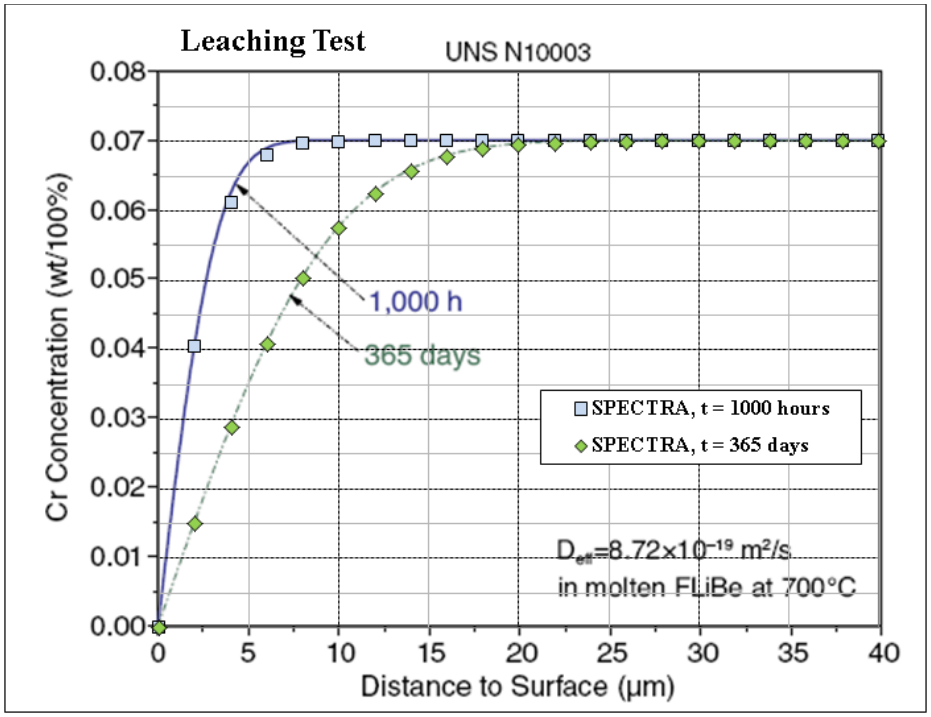


Figure 12-105 Cr concentration profiles, Hastelloy N in FLiBe at 700°C
SPECTRA versus model of [211]

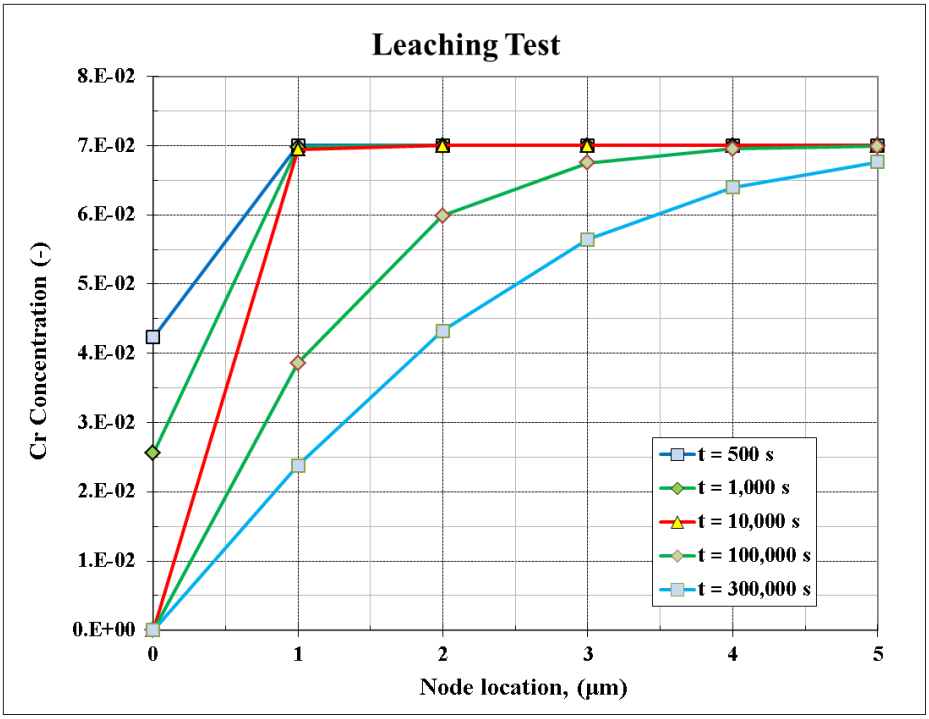


Figure 12-106 Cr concentration profiles, T_0=1000 s,
Stage I: t <~10,000 s, Stage II: t > ~10,000 s

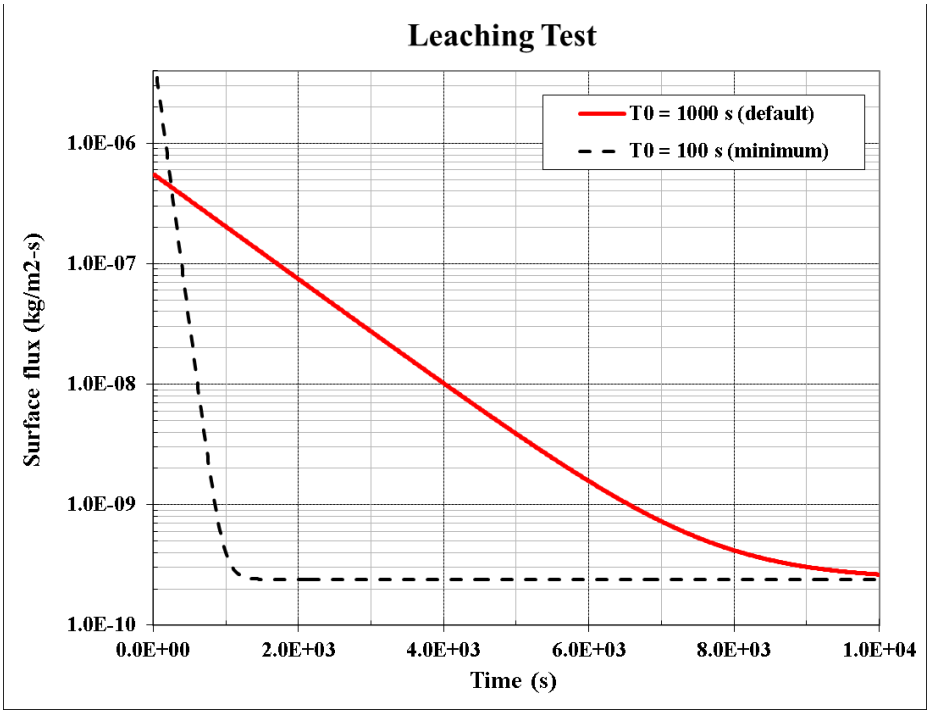


Figure 12-107 Surface mass fluxes during Stage I (<~10,000 s)

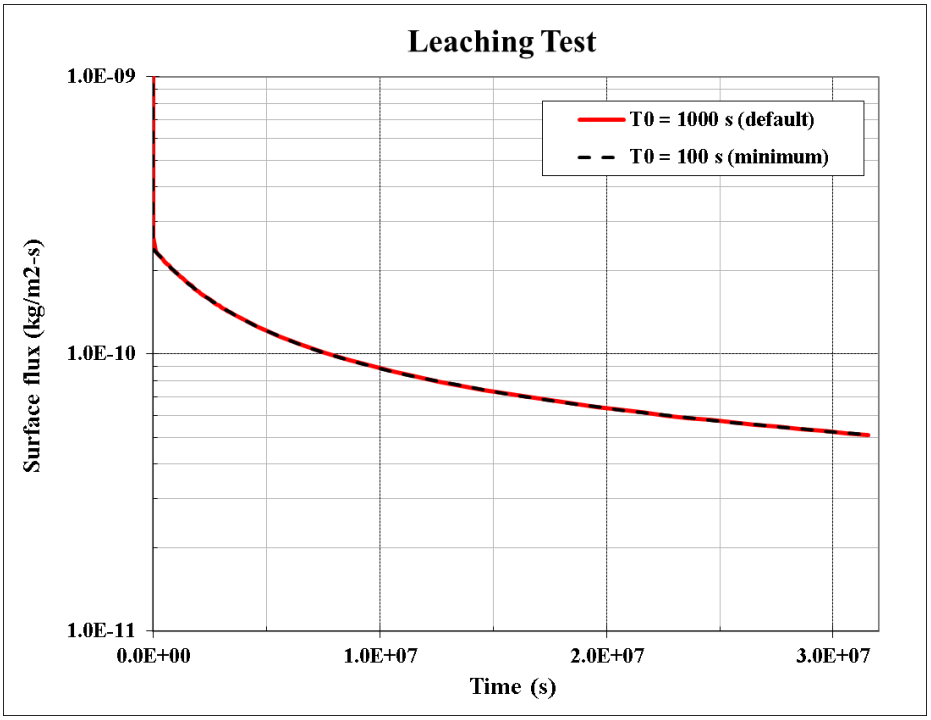


Figure 12-108 Surface mass fluxes during Stage II (> ~10,000 s)

The end of Stage I is at approximately 10,000 s. This is also seen in Figure 12-107, showing the comparison of two cases:

- T_0 = 1000 s (default),
- T_0 = 100 s (minimum).

It is seen that with smaller value of T_0 the initial mass transfer rate is larger and the diffusion-limited value of the surface mass flux (~2×10^{-10} kg/m^2-s) is reached sooner. Figure 12-108 shows the long-term (Stage II) values of the surface mass flux. The values are of order of 10^{-10} kg/m^2-s and slowly decreasing. (The values shown in Figure 12-107 and Figure 12-108 are plotted with reversed sign, as the convention in SPECTRA is that the mass flux from the surface is negative.)

A smaller value of T_0 provides theoretically a better agreement with the model from [211], where the initial surface concentration is taken as zero (the corresponding T_0 is infinitely short). Sensitivity calculations, discussed in Volume 3, showed that the concentration profiles at 1000 h and 365 days, shown in Figure 12-105, are practically identical with both T_0 = 1000 s and T_0 = 100 s.

Temperature effect

The reaction occurs at high temperatures. The effect of temperature can be to a certain extent taken into account in the diffusion coefficient which may be a function of temperature. Additionally, a lower temperature limit is introduced. The reaction does not proceed below the specified minimum temperature, T_{min} (TMLCRT, see Volume 2). An interpolation range, ΔT_{int}, is defined where a linear interpolation is performed. The temperature-dependent multiplier is:

- =0.0 for $T < T_{min}$
- $=(T - T_{min})/\Delta T_{int}$ for $T_{min} < T < T_{min} + \Delta T_{int}$
- =1.0 for $T > T_{min} + \Delta T_{int}$

This multiplier is defined for all leaching processes and for both the diffusion-limited and the CF model.

Nodalization requirements

Diffusion model. In case of the diffusion-controlled model (applied if a positive diffusion coefficient is defined for the wall material), the user must define small nodes close to the boundary surface. The requirement is that the nodes must not be larger than 10 μm for a depth of at least 50 μm. An error message is printed in the diagnostics file and the execution is stopped if these conditions are not met. The recommended nodalization is 1 - 2 μm.

CF model. In case of the CF model, the reverse is true; the boundary node size cannot be too small. This model is applied when the diffusion coefficient is zero. Consequently, there is no diffusion from the neighboring nodes into the boundary node. If the boundary node is very small, then the amount of material available for leaching may be exhausted and the process will stop due to lack if material. The requirement is that the boundary node cannot be smaller than 0.1 mm. The recommended node thickness is ≥ 1 mm. A warning message is printed in the diagnostics file if the node is smaller than the recommended value. An error message is printed and the execution is stopped if the node size is smaller than the required value.

12.3.7 Fission Product Vapor Behavior Within a Single Control Volume

12.3.7.1 Atmosphere of a Control Volume

For each Control Volume a mass balance equation is written separately for all isotopes that are members of a given vapor class.

Isotope mass balance

Within a single Control Volume atmosphere a mass balance equation is written for each isotope:

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + S_E + S_R - S_S - C + F_{gas}$$

N_i	number of molecules of the isotope *i*, (-)
λ_i	decay constant of the isotope *i*, (s^{-1})
$\gamma_{d,j \to i}$	yield fraction of isotope *i* from the decay of isotope *j*, (-)
S_E	external source (user-defined tabular or control functions), (1/s)
S_R	source due to fission product release, (1/s)
S_S	removal due to all sorption mechanisms, (1/s)
C	removal due to condensation of FP vapors, (1/s)
F_{gas}	net source due to inter-volume flows (transport of FP vapors with the gas flow), ($1/m^3$-s)

The first and the second term on the right hand side of the above equation represent the source and the removal of isotopes due to radioactive decay (see decay chains - section 12.3.1). The release term, S_R, is discussed in section 12.3.2. The sorption models are discussed in section 12.3.5, the inter-volume flow term, F, is discussed in section 12.3.8. The condensation term is shortly discussed below.

Condensation of FP vapors occurs whenever the vapor pressure exceeds the saturation pressure, which is a function of temperature:

$$C > 0 \quad if \quad p_{KV} > p_{sat}(T_{CV})$$

p_{KV}	partial pressure of FP vapor number KV, (Pa)
p_{sat}	saturation pressure of FP vapor number KV, (Pa)
T_{CV}	gas temperature in a Control Volume, (K)

The vapor pressure is related to the vapor mass by the following relation:

$$p_{KV} = \frac{m_{KV}}{V_{CV}} \cdot \frac{R}{M_{KV}} \cdot T_{CV}$$

m_{KV} mass of FP vapor class number KV, (kg)
V_{CV} gas volume, (m^3)
R universal gas constant (J/kg-K)
M_{KV} molar weight of FP vapor class number KV, (kg/kmole)
T_{CV} gas temperature, (K)

Therefore the saturation mass corresponding to the saturation pressure is obtained from

$$m_{sat,KV} = p_{sat,KV} \cdot \frac{M_{KV} \cdot V_{CV}}{R \cdot T_{CV}}$$

In order to calculate the condensation term for an isotope number K, a member of the vapor class KV, the saturation mass of isotope K is obtained from the following relation:

$$m_{sat,K} = m_{sat,KV} \cdot \frac{m_K}{\sum_{K \in KV} m_K} = p_{sat,KV} \cdot \frac{M_{KV} \cdot V_{CV}}{R \cdot T_{CV}} \cdot \frac{m_K}{\sum_{K \in KV} m_K}$$

$m_{sat,K}$ saturation mass of the isotope number K, (kg)
m_K mass of the isotope number K, (kg)

The summation is performed over all isotopes K that are members of the vapor class KV. Finally, the condensation term is obtained from:

$$C = \begin{cases} m_K - m_{sat,K} & if \quad m_K < m_{sat,K} \\ 0.0 & if \quad m_K \geq m_{sat,K} \end{cases}$$

Alternatively the condensation term may be written as:

$$C = (m_i - m_{sat,i}) \cdot H(m_i - m_{sat,i})$$

H is the Heavyside step function:

$$H(m_i - m_{sat,i}) = \begin{cases} 0.0 & if \quad m_i < m_{sat,i} \\ 1.0 & if \quad m_i > m_{sat,i} \end{cases}$$

The FP vapor balance equation is written in a finite difference form, by replacing the derivative by the difference:

$$\frac{dN_k}{dt} = \frac{N_k - N_k^0}{\Delta t}$$

N_k FP isotope molecules, new value (-)
N^0_k FP isotope molecules, old time step value ($1/m^3$)
Δt time step size, (s)

In order to solve the equation in an implicit way the right hand side terms need to be written using N_k, rather than the old (known) value, N_k^0. As a result one obtains a linear set of equations, which is then easily solved by one of the standard matrix solvers (see section 17.4). A large amount of test runs performed showed that this method provides a fast and stable solution even with very large time steps.

$$\frac{N_k}{\Delta t} - \frac{N_k^0}{\Delta t} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} + S_E + S_R - S_S - C + F_{gas}$$

The above set of equations can be written shortly in a matrix form:

$$A \cdot N = B$$

where ***N*** is a vector of unknown particle concentrations, n_i, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. The elements of matrix ***A*** are equal to:

$$a_{i,j} = \begin{cases} 1/\Delta t + \lambda_i + H_i & if \quad i = j \\ -\lambda_j \gamma_{d,j \to i} & if \quad i \neq j \end{cases}$$

The elements of vector ***B*** are equal to:

$$b_k = \frac{n_k^0}{\Delta t} + S_E + S_R + S_S + F_{gas} + m_{sat,k}^0 \cdot H_k$$

FP vapor mass

The FP vapor class consists of a number of isotopes. The total mass within each class is obtained as the sum of individual masses of isotopes belonging to the class:

$$m_{KV} = \sum_{K \in KV} m_K$$

m_{KV} mass of FP vapor class number KV, (kg)
m_K mass of the isotope number K, member of the vapor class KV, (kg)

12.3.7.2 Pool of a Control Volume

Isotopes may reside in a pool of a Control Volume either as adsorbed vapors or as attached to aerosol particles that are deposited in the pool. The adsorbed vapor masses are calculated from the equation similar to the one used for the vapors present in the atmosphere (section 12.3.7.1):

$$\frac{dN_i}{dt} = -\lambda_i N_i + \sum_{j \neq i} \lambda_j N_j \gamma_{d,j \to i} - S_S + F_{liq}$$

Note that for water covered surfaces adsorption, penetration, and evaporation are neglected (uC_δ=η=0), while desorption of the reversibly bound molecules is calculated using a constant value $\vartheta = \vartheta_{pool}$ (different from the desorption coefficient in the gas space - see Volume 2):

$$\left(\frac{dC_{rev}}{dt}\right) = -\vartheta_{pool} \cdot C_{rev}$$

Assuming large values for this coefficient a quick desorption of the reversibly bound molecules may be achieved, which is what typically occurs in reality.

Fission products present in the pool may be transported to gas (Volume 2, option IPATRT). The mass transfer is calculated in a very similar way as the mass transfer on aerosol particles present in the pool - section 12.3.5.4.

The mass transfer correlation has the following general form:

$$Sh = A_{Sh} + B_{Sh} \frac{Re^{C_{Sh}}\ Sc^{D_{Sh}}}{1 + E_{Sh}\ Re^{F_{Sh}}\ Sc^{G_{Sh}}}$$

This model is used if a positive diffusion coefficient (DXPART) is defined. Alternatively the model based on Brownian diffusion and inertial impaction may be used, by defining additionally the diffusion layer boundary thickness δ_{BL} (TBL1RT) and the constant in Langmuir-Blodgett correlation for inertial impaction C_{LB}.(CLB1RT). The model is available only if fission products are transported as particles (DPFPRT>0.0, RHFPRT>0.0) and transport on liquid-covered surfaces is active (ISRLRT=2).

- *Brownian diffusion*
  The Brownian diffusion is obtained from:

$$v_{Brown} = \frac{D_C}{\delta_{BL}}$$

  v_{Brown} Brownian deposition velocity, (m/s)
  D_C diffusion coefficient, (m^2/s)
  δ_{BL} diffusion boundary layer thickness, (m)

- *Inertial impaction*
  The inertial impaction is obtained from:

$$v_{inertial} = v_\infty \cdot \eta \cdot A_{h,bubb} / A_{bubb} = v_\infty \cdot \eta / 4$$

  $v_{inertial}$ inertial impaction deposition velocity, (m/s)
  $A_{h,bubb}$ horizontal cross section area of a bubble, (m^2), $= \pi \times D_{bubb}^2 / 4$
  A_{bubb} total surface area of a bubble, (m^2), $= \pi \times D_{bubb}^2$
  v_∞ fission product particle-to-bubble relative velocity
  η collection efficiency, calculated from one of three correlations, described below.

  Langmuir and Blodgett (L-B) correlation

The L-B correlation was developed for a single sphere. The correlation is:

$$\eta = \eta_{\max} \cdot \frac{Stk^2}{(Stk + C_{LB})^2}$$

C_{LB} constant (=0.25, see [206], eq. 3)
η_{max} maximum value of collection efficiency (=1.0)
Stk Stokes number, (-), defined as:

$$Stk = \frac{\rho_p \cdot d_p^2 \cdot v_\infty}{9 \cdot \mu_f \cdot D_{bubb}}$$

ρ_p fission product particle density, (kg/m^3) = RHFPRT
d_p fission product particle diameter, (m) = DPFPRT
v_∞ fission product particle-to-bubble relative velocity, (m/s)
μ_f liquid viscosity, (kg/m/s)
D_{bubb} bubble diameter, (m)

Collection efficiency obtained from the correlation is shown in Figure 12-81. The effect of the constant C_{LB} on the collection efficiency is shown. The value of η_{max} determines the maximum value of collection efficiency (η_{max} = 1.0 in Figure 12-81). The values are defined by the user in the input deck (C_{LB} = CLB2RT, η_{max} = ELB2RT, Volume 2).

Modified Langmuir and Blodgett correlation

The modified correlation is defined as follows:

$$\eta = \eta_{\max} \cdot \begin{cases} \dfrac{0.25}{C_{LB}} \cdot Stk & for \quad Stk < C_{LB} \\ \dfrac{Stk^2}{(Stk + C_{LB})^2} & for \quad Stk > C_{LB} \end{cases}$$

As seen in Figure 12-109, for Stk = 0.25 the line $\eta = Stk$ is tangential to the L-B correlation. In general, the line $\eta = (0.25/C_{LB}) \times Stk$ is tangential for any value of C_{LB}. In the modified correlation this line is used for $Stk < C_{LB}$, which gives somewhat higher collection efficiency in this region. Comparison of both correlations is shown in Figure 12-110. The modified L-B line is very similar to the L-B line; the difference is smaller than the scatter of the source data ([206], figure 1). The modified correlation gives much better agreement with the experimental data for the Static Column experiments, as shown in Volume 3.

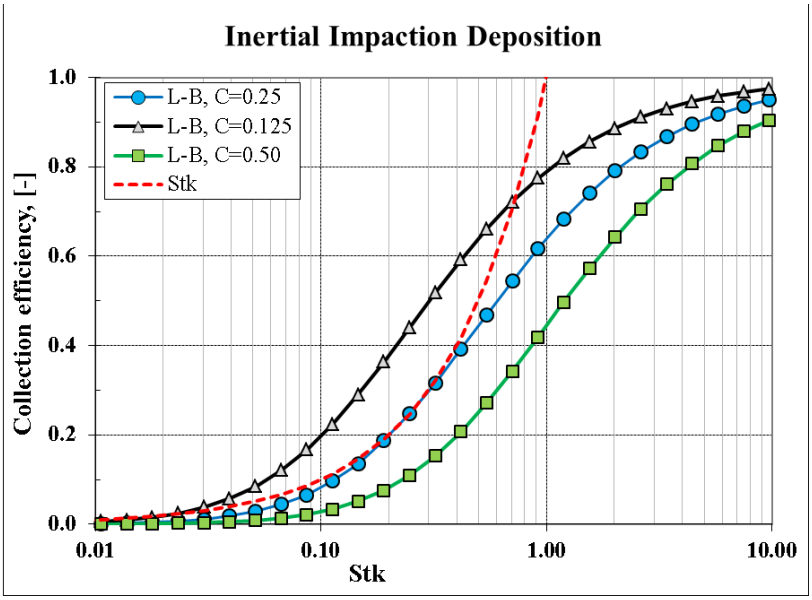


Figure 12-109 Langmuir and Blodgett correlation

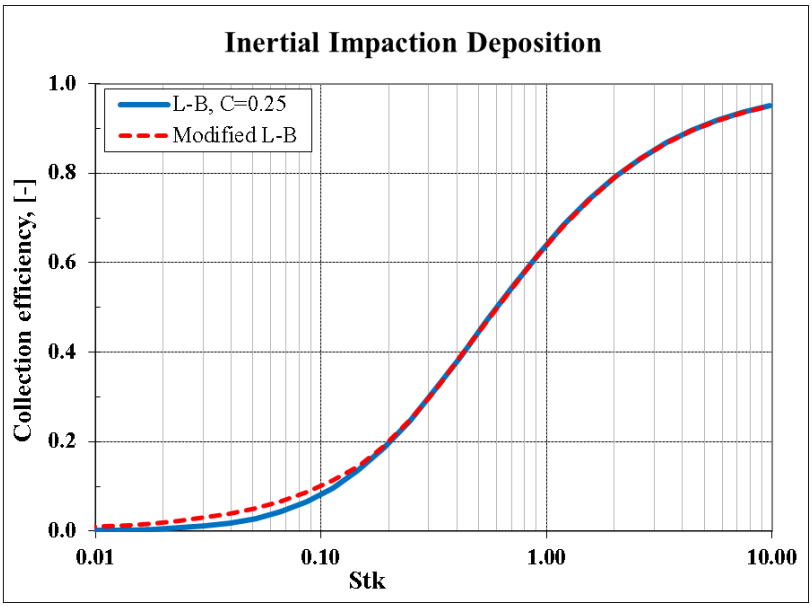


Figure 12-110 Modified versus original Langmuir and Blodgett correlation

Correlation developed based on data of Yoon & Lutrell and Afruns & Kitchener
The following correlation has been developed specifically for use in the SPECTRA code, based on data of Yoon & Lutrell [207] for coal particles and Afruns & Kitchener [208] for quartz particles. The data, (copied from [209], figure 9.5) is shown in Figure 12-111. The correlation is:

$$\eta = Min\left(\eta_{\max}, A \cdot \frac{d_p^{\,2}}{D_b}\right)$$

Performed calculations showed that best agreement is obtained with A = 13,000. The lines obtained with A = 13,000 are shown in Figure 12-111 (red lines, yellow markers). It was found that due to scatter of data the that values between 9,000 ≤ A ≤ 17,000 can be justified.

NOTE: The correlations described above are applied for migration of fission product isotopes to bubbles. Exactly the same correlations are applied for migration to aerosol particles, as described in section 12.3.5.4.

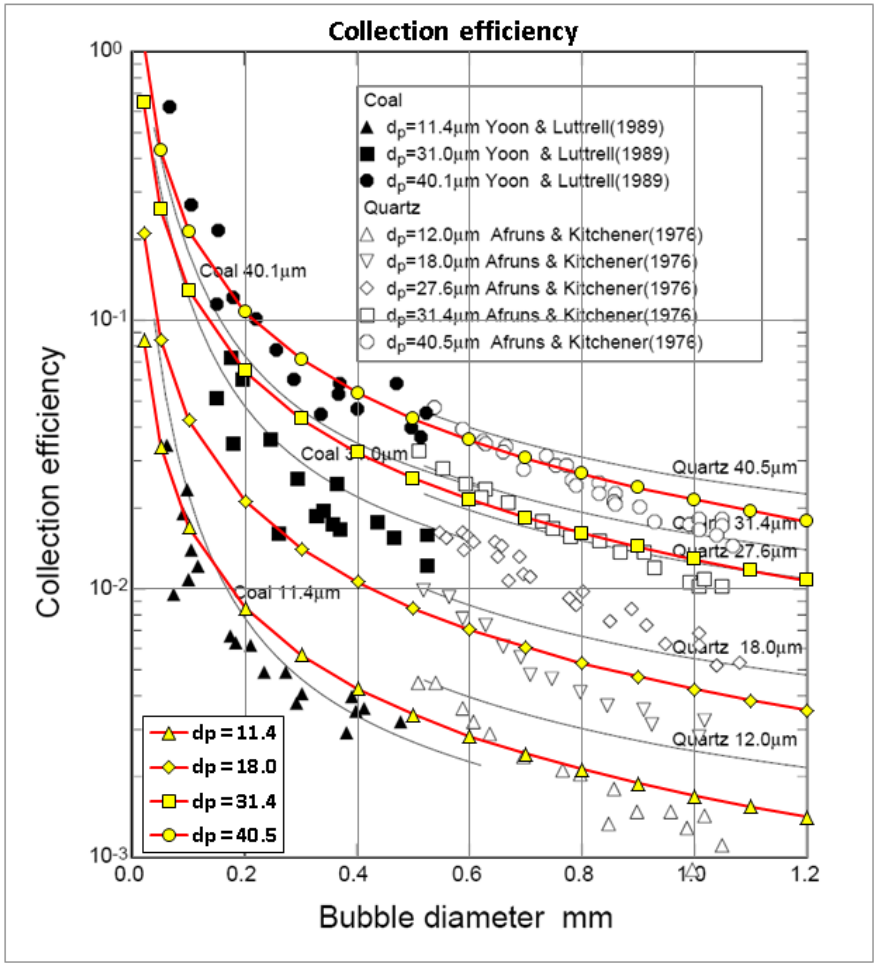


Figure 12-111 Data of Yoon & Lutrell and Afruns & Kitchener and correlation

12.3.8 Fission Product Vapor Transport (Inter-Volume Flows)

In order to calculate the fission product (FP) vapor transport through Control Volumes an implicit formulation is needed, similar to the one adopted for the aerosol transport (see section 12.2.4). The same requirement was set for the FP vapor flow solution – the FP vapor flow solution must be able to find an accurate solution, independent and insensitive to the applied time step, for the time steps that exceed the Courant limit by several orders of magnitude.

The derivation of the FP balance equation, shown below, concerns a single isotope. The same equation is solved for every isotope. In order to limit the amount of subscripts in the discussion that follows, the subscript indicating the isotope type has not been used. It should be clear to the reader that all parameters introduced below are specific to a single isotope.

FP vapor velocities

FP vapors are transported between Control Volumes with fluid flows, including both atmosphere gas flow and pool flows (FP vapors may reside in the pool as dissolved vapors - see section 12.3.7).

Typically fission products are assumed to be transported in molecular form, at least until they are attached to aerosol particles (typical size of aerosol particles is $>10^{-6}$ m). In such case the FP vapors are assumed to have velocities that are equal to the fluid velocities:

$$v_{FP,l} = v_g$$

$v_{FP,l}$ velocity of FP vapor flowing with atmosphere gas through the junction l, (m/s)
v_g gas velocity, (m/s)

$$v_{FP,l} = v_l$$

$v_{FP,l}$ velocity of FP vapor flowing with liquid through the junction l, (m/s)
v_l liquid (pool) velocity, (m/s)

The user may select an option to transport fission product vapors as small particles, typical size $<10^{-6}$ m. In order to do this the user must specify the size of particles and the particle density, which is done separately for every fission product vapor class (records 8932XX - Volume 2). This modeling is affecting only fission products in the liquid phase (pool of CV). The FP isotopes will have the following velocities:

$$v_{FP,l} = v_g$$

$$v_{FP,l} = v_l + J_{JN} \cdot v_\infty$$

J_{JN} junction direction indicator, equal to 0 for horizontal junctions, +1 for vertical up, and –1 for vertical-down junctions (see Volume 2, record 200XXX, IVERJN):
v_∞ vertical velocity (m/s) of a single particle in stagnant atmosphere or pool.

Including the v_∞ velocity allows the fission product particles to flow through vertical junctions when there is no liquid flow.

In case of liquid, positive value of v_∞ means that the flow will be upwards (particle density smaller than the density of the liquid). Negative value of v_∞ means that the flow will be downwards (particle

density larger than the density of the liquid). The user may define a constant value of v_∞ or use the following formula:

$$v_\infty = \left(\frac{4/3}{C_D} \cdot \frac{D_p g(\rho_f - \rho_p)}{\rho_f} \right)^{0.5}$$

g gravity constant, = 9.81 (m/s^2)
ρ_p particle density, (kg/m^3)
ρ_f fluid density, (kg/m^3)
D particle diameter, (m)
C_D drag coefficient, (-)

On top of the drag coefficient correlation, a correlation specific for very small particles may be used, as follows.

$$v_\infty = \frac{1}{18} \cdot \frac{D_p{}^2 g(\rho_f - \rho_p)}{\eta_f} \cdot C_{small}$$

Here C_{small} is the user-defined constant (CSMLCV). This correlation is applicable for very small particles, $D_p < \sim 10^{-4}$ m. The best estimate value of C_{small} is 1.0.

FP vapor flow through a junction

The mass of FP vapor transported through a junction per second is equal to:

$$v_{FP,l} \cdot A_l \cdot \frac{m_{FP,i}}{V_i}$$

$v_{p,f}$ velocity of FP vapor in the junction l (in atmosphere or pool), (m/s)
A_l fluid flow area (gas or liquid) for the junction l, (m^2)
V_i fluid volume (gas or liquid) of the source volume i, (m^3)
m_i mass of particles in the source control volume i (in atmosphere or pool), (kg).

Loss of FP vapors due to flow out of a CV

The mass removal rate for a Control Volume is written for those junctions for which the flow is out of the control volume i. For this purpose an outgoing flow indicator is used:

O_{li} outgoing flow indicator
= 1: flow through the junction l is out of the control volume i.
= 0: otherwise.

Using this indicator, the mass removal rate for the volume i due to flow through junction l is written as:

$$- v_{FP,l} \cdot A_l \cdot \frac{m_i}{V_i} \cdot O_{li}$$

The rate of change of particles due to flow out of the volume *i* is obtained by summing the junction flows for all junctions connected to the volume *i*:

$$-\sum_{l\in i}\left(\frac{A_l v_{FP,l}}{V_i} O_{li}\right)\cdot m_i$$

Gain of FP vapors due to flow into a CV

The mass source rate for a Control Volume is written for those junctions for which the flow is into the Control Volume *i*. For this purpose an incoming flow indicator is used:

I_{li} incoming flow indicator
= 1: flow through the junction *l* is into the volume *i* (from the volume *j*).
= 0: otherwise.

The source rate is given by:

$$+v_{FP,l}\cdot A_l\cdot\frac{m_i}{V_j}\cdot I_{li}\cdot(1-\varepsilon_l)$$

Where the term $(1 - \varepsilon_l)$ takes into account eventual aerosol removal in the junction *l*. This removal may occur in two cases:

- A vapor filter exists in the junction *l*; in this case ε_l is the filter efficiency (= ε_F, section 12.3.9)
- The stream of gas enters the pool in the receiving volume; in this case ε_l is the pool adsorption efficiency for FP vapors (= ε_{PA}, section 12.3.10). This mass is removed from the atmosphere and added into the pool.

The total source of particles for all flows into the control volume *i* is obtained by summing junction flows for all junctions connected to the volume *i*:

$$\sum_{l\in j}\left(\frac{A_l v_{FP,l}}{V_j} I_{li}\right)\cdot(1-\varepsilon_l)\cdot m_j$$

Using the above terms, the total mass balance for a given CV takes into account aerosol sources and sinks:

$$\frac{dm_i}{dt}=\sum_{l\in i}\left(\frac{A_l v_{FP,l}}{V_j} I_{li}\right)\cdot(1-\varepsilon_l)\cdot m_j-\sum_{l\in i}\left(\frac{A_l v_{FP,l}}{V_i} O_{li}\right)\cdot m_i+S_E+S_R-S_{SP}+S_{SN}+R-C$$

m_i mass of particles in volume *i*, (kg)
S_E external source (user-defined tabular or control functions), (kg/s)
S_R source due to fission product release, (kg/s)
S_{SP} removal due to positive sorption mechanisms, (kg/s)
S_{SN} source due to negative sorption mechanisms, (kg/s)
R source or removal due to radioactive decay, (kg/s), equal to: $-\lambda_i N_i + \Sigma(\lambda_j N_j \gamma_{d,j\text{-}i})$ see section 12.3.7)
C removal due to condensation, (kg/s)

The sorption mechanisms are divided here into positive (when the vapors are transported from the gas to the wall or the aerosol particles) and negative (when the vapors are transported from the wall or the aerosol particles to the gas) because those two terms are treated differently in the solution scheme, as will described later in this section. The positive and negative terms are given below for the three sorption models available in the code (see section 12.3.5):

- User-defined model:

$$S_{SP} = CF \quad if\ CF > 0.0, \quad zero\ otherwise$$
$$S_{SN} = -CF \quad if\ CF < 0.0, \quad zero\ otherwise$$

- Sorption model 1:

$$S_{SP} = A_S(T_w) \cdot C_V^{x_A}$$
$$S_{SN} = B_S(T_w) \cdot C_d^{x_B}$$

- Sorption model 2 (see Figure 12-84)

$$S_{SP} = \alpha \cdot (1-\theta) \cdot u \cdot C_\delta$$
$$S_{SN} = \eta \cdot C_d + \eta\vartheta \cdot C_{rev}$$

The condensation term, *C*, is given treated separately in an internal iteration to solve the flow matrix, as will be shown below. For each Control Volume occurrence of condensation is identified by a separate function:

$$H(m_i - m_{sat,i}) = \begin{cases} 0.0 & if \quad m_i < m_{sat,i} \\ 1.0 & if \quad m_i > m_{sat,i} \end{cases}$$

H is the Heavyside step function.

Note that the above equation is similar to the equation for the FP vapor dynamics within a single Control Volume, shown in section 12.3.7. Here the sum given by the first two terms on the right hand side of the equation determines the term *F*, for the equation in section 12.3.7.

The mass balance is written here using the molecule numbers, rather than the particle densities, as in section 12.2.2. SPECTRA is using and printing both the concentrations (1/m^3) and the total masses (kg) within a CV (in MELCOR for example only masses are available, which makes it sometimes difficult to analyze the results – see the Vent test case in Volume 3). The conversion from particle concentrations to particle masses is:

$$n = \frac{m}{V_{CV}\rho_p V_p}$$

V_{CV} fluid volume within a CV (atmosphere or pool), (m^3)
ρ_p particle density, (kg/m^3)
V_p volume of a single particle (see Table 12-1), (m^3)

In order to solve the equation implicitly it is assumed that the sorption removal term is proportional to the FP vapor concentrations, and consequently the FP mass. For a single deposition surface *k*, in the control volume *i*, the sorption rate (in kg/s) is given by:

$$A_{S,k} \cdot v_{SP,k} \cdot \frac{m_i}{V_i} = \left(\frac{A_{S,k} \cdot v_{SP,k}}{V_i} \right) \cdot m_i$$

$A_{S,k}$	sorption area, (m^2)
$v_{SP,k}$	sorption velocity, equal to S_{SP}/ρ_i, (m/s)
m_i	mass of FP vapor in the Control Volume *i*, (kg)
V_i	volume (gas or liquid) in the Control Volume *i*, (m^3)
ρ_i	density of FP vapor in the Control Volume *i*, (kg/m^3), equal to m_i/ρ_i,

The total removal rate from volume *i* due to sorption is obtained by summing all the sorptions in the volume *i*, including sorptions on 1-D and 2-D Solid Heat Conductors, as well as sorptions on the pool surface (if present) and aerosol particles (if present).

$$S_{SP} = \sum_{k \in i} \left(\frac{A_{S,k} \cdot v_{SP,k}}{V_i} \right) \cdot m_i$$

Since the sorption must be calculated using the end-of-time-step value of FP vapor mass, m_i, the positive sorption term must be included implicitly in the balance equation. The mass balance equation for a volume without condensation (C=0.0) becomes:

$$\frac{dm_i}{dt} = \sum_{l \in i} \left(\frac{A_l v_{FP,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j - \sum_{l \in i} \left(\frac{A_l v_{FP,l}}{V_i} O_{li} \right) \cdot m_i - \sum_{k \in i} \left(\frac{A_{S,k} v_{SP,k}}{V_i} \right) \cdot m_i + \\ + S_E + S_R + S_N + R$$

The above equation is written in a finite difference form, by replacing the derivative by the difference:

$$\frac{dm_i}{dt} = \frac{m_i - m_i^0}{\Delta t}$$

m_i	FP vapor mass in volume *i*, new value (kg)
m_i^0	FP vapor mass in volume *i*, old time step value (kg)
Δt	time step size, (s)

The particle balance equation becomes:

$$\frac{m_i - m_i^0}{\Delta t} = \sum_{l \in i} \left(\frac{A_l v_{FP,l}}{V_j} I_{li} \right) (1 - \varepsilon_l) m_j - \sum_{l \in i} \left(\frac{A_l v_{FP,l}}{V_i} O_{li} \right) m_i - \sum_{k \in i} \left(\frac{A_{S,k} v_{SP,k}}{V_i} \right) m_i + \\ + S_E + S_R + S_{SN} + R$$

The above set of equations can be written shortly in a matrix form:

$$A \cdot M = B$$

where $\boldsymbol{M}$ is a vector of unknown FP vapor masses, m_i, $\boldsymbol{A}$ is a square matrix, and $\boldsymbol{B}$ is a vector of right hand side quantities. Elements of the matrix $\boldsymbol{A}$ are equal to:

- diagonal elements ($i=j$):

$$a_{ii} = 1 + \Delta t \cdot \sum_{k \in i} \left(\frac{A_{S,k} \cdot v_{SP,k}}{V_i} \right) + \Delta t \cdot \sum_{l \in i} \left(\frac{A_l \cdot v_{FP,l}}{V_i} \cdot O_{li} \right)$$

- other elements ($i \neq j$):

$$a_{ij} = -\Delta t \cdot \sum_{l \in j} \left(\frac{A_l \cdot v_{FP,l}}{V_i} \cdot I_{lj} \right) \cdot \left(1 - \varepsilon_l\right)$$

$A_{S,k}$	sorption area, sorption surface k, within Control Volume i, (m^2),
$v_{SP,k}$	sorption velocity calculated for the deposition surface k, (m/s),
V_i	fluid volume of the Control Volume i, (m^3),
A_l	flow area, junction l, connected to Control Volume i and j, (m^2),
$v_{p,l}$	velocity of fluid, junction l, (m/s),
ε_l	efficiency of filter (=ε_F, section 12.3.9)

The first sum in the diagonal element is over all deposition surfaces k in the Control Volume i, the second sum is over all junctions l connected to or from the Control Volume i.

The elements of vector ***B*** are equal to:

$$b_i = m_i^0 + \Delta t \cdot \left(S_E + S_R + S_{SN} + R\right)$$

If condensation is encountered in the volume i, $H(m_i - m_{sat,i}) = 1$, then the matrix elements are re-written for the row i, to give:

$$m_i = m_{sat,i}$$

This means the elements of the row i become:

- diagonal element ($i=j$):

$$a_{ii} = 1.0$$

- other elements ($i \neq j$):

$$a_{ij} = 0.0$$

- right-hand side (element of vector ***B***):

$$b_i = m_{sat,i}$$

In this way it is ensured that the vapor mass in volume i, m_i, will be exactly equal to the saturation mass, $m_{sat,i}$.

The matrix equation is solved using one of the standard matrix solvers (see section 17.4). As a result the particle masses, m_i, in all Control Volumes are calculated.

The value of the Heaviside function, H, is not known before the matrix solution. Therefore the following procedure is applied:

- As a first step the values of H are guessed based on previous (beginning of time step) values, m_i^0. This means if the condensation is in progress at the beginning of the time step, then H=1.0, otherwise H=0.0.
- The inter-volume transport matrix is solved, resulting in the end-of time step values of all FP vapor masses. These masses are called the "projected masses", since they represent the expected (projected) end-of time step masses, computed based on the estimated value of the condensation term.
- The new values are checked for the condensation. If the condensation status changes in one or more Control Volume, then the appropriate value or values of H is changed in the flow matrix and the flow matrix is re-solved. This is an internal iteration in the flow solution.
- The FP vapor mass balance within a single CV (section 12.3.7) is solved using the term F_{gas} based on the projected masses. Those masses are called the "true masses" of FP vapors, since they represent the exact (conservative) masses of vapors within each Control Volume, for the inter-volume transport of particles as calculated by the projected masses.

Usually no iteration is needed because condensation status (the values of H) does not change frequently in time. Typically condensation of FP vapors occurs if a fluid from a relatively warm CV enters a relatively cold CV and such conditions, if arise, usually persist over many time steps. If the condensation status does change usually one or two iterations are needed to converge the solution. A maximum number of iteration is set (default of 10 - see Volume 2). If the solution does not converge in a maximum number of iterations, then the RT Package requests an "immediate time step cut". In such case a warning message is written to the diagnostics file *.DIA.

Because of the applied solution strategy the aerosol masses are conserved (within the accuracy of the double precision arithmetic – relative error ~10^{-15}). The projected masses, m_i, coming from the inter-volume transport equation are not conservative; their error (discrepancy between the conservative values, obtained from the single volume dynamics equation) is typically within 1%. Note that because of including the sorption terms in the implicit formulation, the sorption rates are calculated based on the projected masses and not the true masses. This is considered unimportant because the accuracy of the correlations used to compute the sorption rates (see section 12.3.5) is at best an order of magnitude worse (~10%) than the accuracy of the projected masses (~1%).

The stability of the FP vapor flow solution, including the vapor transport with the atmosphere gas, and the transport of the vapors adsorbed in the pool, was verified by performing time step sensitivity studies for the test cases shown in Volume 3. In all tests shown in Volume 3 the results (FP vapor concentrations) were practically insensitive to the applied time step. The PBMR cases were run with Δt = 1000 s, 100 s, and 10 s, and no significant differences were observed.

12.3.9 Fission Product Vapor Filters

A simple vapor filter model is available in SPECTRA. The filter efficiency is a constant, user-defined value. The filter efficiency, ε_F, is defined as a fraction of the incoming vapor that is removed by the filter:

$$\varepsilon_F = \frac{W_{rem}}{W_{in}}$$

Of course the filter efficiency must be within the range:

$$0.0 \le \varepsilon_F \le 1.0$$

Two options are available:

- Vapor filter - filter efficiency is defined for each vapor class. All isotopes within a given vapor class are removed with the same efficiency.
- Isotope filter - filter efficiency is defined for each isotope.

Note that other codes (see for example [46]) often use the decontamination factor, *DF*, rather than the filter efficiency, as an input parameter. The decontamination factor is defined as a ratio of aerosols coming into the filter to the aerosols leaving the filter:

$$DF_F = \frac{W_{in}}{W_{out}}$$

Taking into account that $W_{rem} = W_{in} - W_{out}$, it can be easily shown that the relation between the filter efficiency and the decontamination factor is:

$$DF_F = \frac{1}{1-\varepsilon_F} \qquad \varepsilon_F = \frac{DF_F - 1}{DF_F}$$

The range of the decontamination factor is:

$$1.0 \le DF_F \le \infty$$

12.3.10 Pool Adsorption

When a stream of gas enters a pool region of the receiving Control Volume, then it forms bubbles in the pool. The gas is assumed to quickly reach equilibrium at the entrance to the pool (the equilibrium conditions are calculated by the Bubble Collapse Model – see Chapter 2). At the end of the bubble collapse, the mass flow of gas entering the pool, W_B, is divided into the mass flow of gas, W_G, and condensed liquid, W_L. The flow W_G forms bubbles in the pool, while the condensed steam, W_L, remains in the pool. The vapors entering the pool with the stream of gas are divided as follows:

- The part W_L / W_B remains in the pool.
- The part W_G / W_B is transported partly to the atmosphere and partly remains in the pool, depending on the pool scrubbing efficiency, E_{PV}.

The division of aerosol particles is:

- Fraction remaining in the pool: $\frac{W_G}{W_B} E_{PV} + \frac{W_L}{W_B}$
- Fraction reaching the atmosphere: $\frac{W_G}{W_B}(1 - E_{PV})$

The overall pool adsorption efficiency, ε_{PV}, is defined as the fraction of aerosols remaining in the pool:

$$\varepsilon_{PV} = \frac{W_G}{W_B} E_{PV} + \frac{W_L}{W_B}$$

When the pool adsorption efficiency is equal to 1.0, then all vapors remain in the pool and the overall pool adsorption efficiency is also 1.0:

$$\varepsilon_{PV} = \frac{W_G}{W_B} \cdot 1.0 + \frac{W_L}{W_B} = \frac{W_B}{W_B} = 1.0 = \varepsilon_{PV}(\max)$$

When the pool adsorption efficiency is equal to 0.0, then the overall pool adsorption efficiency is equal to:

$$\varepsilon_{PV} = \frac{W_G}{W_B} \cdot 0.0 + \frac{W_L}{W_B} = \frac{W_L}{W_B} = \varepsilon_{PV}(\min)$$

The pool adsorption efficiency, E_{PV}, is defined for each FP vapor class in the input data using either a Tabular or a Control Function.

12.3.11 Inter-Volume Flows of Fission Products Attached to Aerosols

12.3.11.1 Introduction

This section describes the method applied to calculate inter-volume transport of fission products attached to aerosol particles. The matrix equation used here is similar to the one used to calculate the inter-volume aerosol transport, described in section 12.2.4, and the inter-volume transport of fission product vapors, described in section 12.3.8. These three matrix equations constitute the main solution procedures of the RT Package. As described in section 12.2.4, the formulation of the matrix equations allows the use of large time steps, violating the courant limits by many orders of magnitude.

The present equation calculates the mass fraction of fission products that are attached to aerosols. Therefore the equation is similar to the aerosol equation, described in section 12.2.4. Before the full equation is presented, it is instructive to show the equation for an idealized, simple case. This simple case is shown in section 12.3.11.2. The full equation is presented in section 12.3.11.3.

12.3.11.2 Simple Case Equation

Let's consider a simple case aerosol balance equation for a Control Volume number *i*, which has two sources only:

- An external source of aerosols; the source is equal to *S*, kg/s.
- Removal of aerosols; the removal rate is given by: $R \cdot m_i$ in kg/s, where *R* is the removal rate in 1/s and m_i is the aerosol mass in Control Volume *i*, kg.

The aerosol mass balance is given by:

$$\frac{dm_i}{dt} = S - R \cdot m_i$$

This is a simple version of the aerosol transport equation, described in section 12.2.4. Next, let's write the isotope balance for the fission product isotope that is carried by the aerosol particles. The aerosols coming from the source S carry the fission product isotope, with the mass fraction of Y_S. The average mass fraction of the isotope on the aerosols in the Control Volume i, Y_i, is calculated from the balance equation for the isotope:

$$\frac{d(m_i \cdot Y_i)}{dt} = S \cdot Y_S - R \cdot m_i \cdot Y_i$$

An implicit solution of the aerosol and the isotope balances for the simple case are given below.

- Aerosol balance:

$$\frac{dm_i}{dt} = S - R \cdot m_i$$

In the numerical approximation the derivative dm_i/dt is written as $(m_i - m_i^0)/\Delta t$. Here m_i is the current (end-of-time step) value, m_i^0 is the previous (beginning of time step) value, and Δt is the time step size.

$$\frac{m_i - m_i^0}{\Delta t} = S - R \cdot m_i$$

Therefore:

$$m_i = m_i^0 + \Delta t \cdot S - \Delta t \cdot R \cdot m_i$$

and:

$$m_i = \frac{m_i^0 + \Delta t \cdot S}{1 + \Delta t \cdot R}$$

- Fission product balance:

$$\frac{d(m_i \cdot Y_i)}{dt} = S \cdot Y_S - R \cdot m_i \cdot Y_i$$

In the numerical approximation the derivative $d(m_iY_i)/dt$ is written as $(m_iY_i - m_i^0Y_i^0)/\Delta t$. Here m_iY_i is the current (end-of-time step) value, $m_i^0Y_i^0$ is the previous (beginning of time step) value, and Δt is the time step size.

$$\frac{m_i \cdot Y_i - m_i^{\,0} \cdot Y_i^0}{\Delta t} = S \cdot Y_S - R \cdot m_i \cdot Y_i$$

Therefore:

$$m_i \cdot Y_i = m_i^0 \cdot Y_i^{\,0} + \Delta t \cdot (S \cdot Y_S - R \cdot m_i \cdot Y_i)$$

which leads to:

$$m_i \cdot Y_i = \frac{m_i^0 \cdot Y_i^{\,0} + \Delta t \cdot S \cdot Y_S}{1 + \Delta t \cdot R}$$

The term $m_i + \Delta t\, R\, m_i$ may be replaced using the mass balance by $m_i^0 + \Delta t\, S$. Therefore:

$$Y_i = \frac{m_i^0 \cdot Y_i^0 + \Delta t \cdot S \cdot Y_S}{m_i^0 \quad + \quad \Delta t \cdot S}$$

The above equations for m and Y_i are solved using excel with the following data:

- Removal rate: R = 0.80 (1/s)
- Source: S = 0.30 (kg_{aer}/s)
- Isotope fraction in the source: Y_S = 0.50

The initial conditions were assumed as:

- Initial mass of aerosols: $m_i(t=0)$ = 1.0
- Initial isotope fraction: $Y_i(t=0)$ = 0.0

Time steps of Δt = 1.0 s and 2.0 s were used. Results are shown in Figure 12-112 and Figure 12-113. The obtained results are stable and independent of the applied time step.

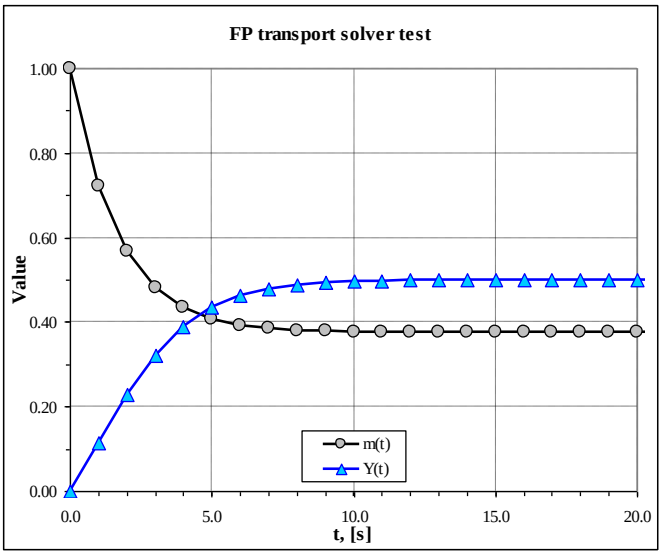


Figure 12-112 Simple test case, Δt = 1.0 s

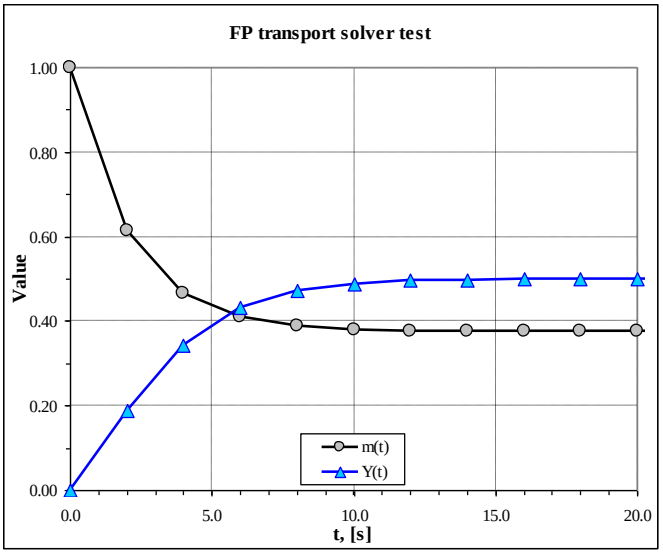


Figure 12-113 Simple test case, Δt = 2.0 s

In a summary, the isotope balance equation is compared to the aerosol balance equation.

- For the simple case the aerosol balance and the isotope balance are given by:

$$m_i = \frac{m_i^0 + \Delta t \cdot S}{1 + \Delta t \cdot R}$$

$$m_i \cdot Y_i = \frac{m_i^0 \cdot Y_i^0 + \Delta t \cdot S \cdot Y_S}{1 + \Delta t \cdot R}$$

- There is a correspondence between these equations.
  - The source term is multiplied by isotope fraction in the source:

$$S \rightarrow S \cdot Y_S$$

  - The aerosol masses are multiplied by the isotope fractions:

$$m_i \rightarrow m_i \cdot Y_i$$
$$m_i^0 \rightarrow m_i^0 \cdot Y_i^0$$

The correspondence between the aerosol balance and the isotope balance is quite clear for this case. Understanding this correspondence is helpful to understand the equation described in the next section. There is similar correspondence between the full equation of aerosol balance, described in section 12.2.4 and the full equation for the isotopes attached to aerosol particles, described in the next section.

12.3.11.3 Full Equation

The individual terms of the aerosol balance equation are discussed in detail in section 12.2.4. Here the same terms are mentioned and transformed into the form suitable for the isotope balance.

Loss of particles due to flow out of a CV

The rate of change of particles due to flow out of the volume *i* is obtained by summing the junction flows for all junctions connected to the volume *i* (see section 12.2.4):

$$\text{Flow of particles out of Control Volume } i = -\sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_i} O_{li}\right) \cdot m_i$$

$v_{p,l}$ velocity of particles in the junction *l* (in atmosphere or pool), (m/s)
A_l fluid flow area (gas or liquid) for the junction *l*, (m^2)
V_i fluid volume (gas or liquid) of the source volume *i*, (m^3)
m_i mass of particles in the source control volume *i* (in atmosphere or pool), (kg).
O_{li} outgoing flow indicator
= 1: particle flow through the junction *l* is out of the control volume *i*.
= 0: otherwise.

If the particles in the control volume *i* carry the mass fraction Y_i of the isotope being considered, the isotope balance term is given by:

$$\text{Flow of isotope out of Control Volume } i = -\sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_i} O_{li}\right) \cdot m_i \cdot Y_i$$

Gain of particles due to flow into a CV

The total source of particles for all flows into the control volume *i* is obtained by summing junction flows for all junctions connected to the volume *i*:

$$\text{Flow of particles into Control Volume } i = \sum_{l \in j}\left(\frac{A_l v_{p,l}}{V_j} I_{li}\right) \cdot (1-\varepsilon_l) \cdot m_j$$

I_{li} incoming flow indicator
= 1: particle flow through the junction *l* is into the volume *i* (from the volume *j*).
= 0: otherwise.

Here the term $(1 - \varepsilon_l)$ takes into account eventual aerosol removal in the junction *l*. This removal may occur in tree cases:

- A filter exists in the junction *l*; in this case ε_l is the filter efficiency (= ε_F, section 12.2.10)
- The stream of gas enters the pool in the receiving volume; in this case ε_l is the overall pool scrubbing efficiency (= ε_{PS}, section 12.2.9). This mass is removed from the atmosphere and added into the pool.
- An inertial impaction model is associated with this junction. In this case ε_l is the collection efficiency (= η, section 12.2.5.6). This mass is removed from the atmosphere and added into the pool.

If the particles in the control volume *j* carry the mass fraction Y_j of the isotope being considered, the isotope mass balance term is given by:

$$\text{Flow of isotope into Control Volume } i = \sum_{l \in j} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j \cdot Y_j$$

Using the above terms, the total mass balance for a given CV takes into account aerosol sources and sinks:

$$\frac{d(m_i \cdot Y_i)}{dt} = \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_j} I_{li} \right) \cdot (1 - \varepsilon_l) \cdot m_j \cdot Y_j - \sum_{l \in i} \left(\frac{A_l v_{p,l}}{V_i} O_{li} \right) \cdot m_i \cdot Y_i + \\ + S_E \cdot Y_E + S_R \cdot Y_R - D \cdot Y_i + R \cdot Y_{SC} + K \cdot Y_K + \\ + S_{SP} \cdot x_{CV} - S_{SN} \cdot x_{aer} + R_D + C_V \cdot x_{CV}$$

m_i mass of particles in volume *i*, (kg)
S_E external source (user-defined tabular or control functions), (kg/s)
S_R source due to fission product release, (kg/s)
D removal due to all deposition mechanisms, (kg/s)
R source due to resuspension of deposited aerosols, (kg/s)
K net source due to coagulation of all size sections, (kg/s)
S_{SP} source due positive sorption flux of FP vapor on the aerosol surface, (kg/s)
S_{SN} source due negative sorption flux of FP vapor on the aerosol surface, (kg/s)
R_D source or removal due to radioactive decay, (kg/s), equal to: $-\lambda_i N_i + \Sigma(\lambda_j N_j \gamma_{d,j-i})$ see section 12.3.7)
C_V condensation of FP vapor *V* containing the considered isotope, (kg/s)
Y_i mass fraction of the isotope on particles in volume *i*, (-)
Y_E mass fraction of the isotope on particles coming from the external source, (-)
Y_R mass fraction of the isotope on particles coming from the release from fuel, (-)
Y_{SC} mass fraction of the isotope on particles deposited on Solid Conductors, (-)
Y_K mass fraction of the isotope in coagulating particles, (-)
x_{CV} mass fraction of the isotope in the fission product vapor in volume *CV*, (-)
x_{aer} mass fraction of the isotope in the fission product vapor on aerosol particles, (-)

The total removal rate from volume *i* due to deposition is obtained by summing all the depositions in the volume *i*, including depositions on 1-D and 2-D Solid Heat Conductors, as well as depositions on the pool surface (if present).

$$\text{Removal of particles due to deposition in Control Volume } i = \sum_{k \in i} \left(\frac{A_{D,k} \cdot v_{D,k}}{V_i} \right) \cdot m_i$$

$A_{D,k}$ deposition area, (m^2)
$v_{D,k}$ deposition velocity, calculated as shown in section 12.2.5, (m/s)

For the isotope:

$$\text{Removal of the isotope due to deposition in Control Volume } i = \sum_{k \in i} \left(\frac{A_{D,k} \cdot v_{D,k}}{V_i} \right) \cdot m_i \cdot Y_i$$

Since the deposition must be calculated using the end-of-time-step value of particle mass, m_i, the deposition term must be included implicitly in the balance equation. The mass balance equation becomes:

$$\frac{d(m_i \cdot Y_i)}{dt} = \sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_j} I_{li}\right) \cdot (1-\varepsilon_l) \cdot m_j \cdot Y_j - \sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_i} O_{li}\right) \cdot m_i \cdot Y_i -$$
$$- \sum_{k \in i}\left(\frac{A_{D,k} v_{D,k}}{V_i}\right) \cdot m_i \cdot Y_i + S_E \cdot Y_E + S_R \cdot Y_R + R \cdot Y_{SC} + K \cdot Y_K +$$
$$+ S_{SP} \cdot x_{CV} - S_{SN} \cdot x_{aer} + R_D + C_V \cdot x_{CV}$$

The above equation is written in a finite difference form, by replacing the derivative by the difference:

$$\frac{d(m_i \cdot Y_i)}{dt} = \frac{m_i \cdot Y_i - m_i^0 \cdot Y_i^0}{\Delta t}$$

m_i particle mass in volume *i*, new value (kg)
m_i^0 particle mass in volume *i*, old time step value (kg)
Δt time step size, (s)

The particle balance equation becomes:

$$\frac{m_i \cdot Y_i - m_i^0 \cdot Y_i^0}{\Delta t} = \sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_j} I_{li}\right) \cdot (1-\varepsilon_l) \cdot m_j \cdot Y_j - \sum_{l \in i}\left(\frac{A_l v_{p,l}}{V_i} O_{li}\right) \cdot m_i \cdot Y_i -$$
$$- \sum_{k \in i}\left(\frac{A_{D,k} v_{D,k}}{V_i}\right) \cdot m_i \cdot Y_i + S_E \cdot Y_E + S_R \cdot Y_R + R \cdot Y_{SC} + K \cdot Y_K +$$
$$+ S_{SP} \cdot x_{CV} - S_{SN} \cdot x_{aer} + R_D + C_V \cdot x_{CV}$$

The above set of equations can be written shortly in a matrix form:

$$A \cdot Y = B$$

where ***Y*** is a vector of unknown fractions Y_n, ***A*** is a square matrix, and ***B*** is a vector of right hand side quantities. Elements of the matrix ***A*** are similar to the elements of matrix presented in section 12.2.4, and are equal to:

- diagonal element ($i=j$):

$$a_{ii} = m_i \cdot \left\{1 + \Delta t \cdot \sum_{k \in i}\left(\frac{A_{D,k} \cdot v_{D,k}}{V_i}\right) + \Delta t \cdot \sum_{l \in i}\left(\frac{A_l \cdot v_{p,l}}{V_i} \cdot O_{li}\right)\right\}$$

- other elements ($i \neq j$):

$$a_{ij} = -m_j \cdot \left\{\Delta t \cdot \sum_{l \in j}\left(\frac{A_l \cdot v_{p,l}}{V_i} \cdot I_{lj}\right) \cdot (1-\varepsilon_l)\right\}$$

$A_{D,k}$ deposition area, deposition surface *k*, within Control Volume *i*, (m^2),
$v_{D,k}$ deposition velocity calculated for the deposition surface *k*, (m/s),
V_i fluid volume of the Control Volume *i*, (m^3),
A_l flow area, junction *l*, connected to Control Volume *i* and *j*, (m^2),

$v_{p,l}$ velocity of particle flowing with fluid, junction l, (m/s),

ε_l efficiency of filter (= ε_F, section 12.2.10), or overall pool scrubbing efficiency (= ε_{PS}, section 12.2.9) if present in the junction l, or inertial impaction collection efficiency (= η, section 12.2.5.6) if present in the junction l.

The first sum in the diagonal element is over all deposition surfaces k in the Control Volume i, the second sum is over all junctions l connected to or from the Control Volume i.

The elements of vector $\boldsymbol{B}$ are equal to:

$$b_i = m_i^0 \cdot Y_i^0 + \Delta t \cdot \left(S_E \cdot Y_E + S_R \cdot Y_R + R \cdot Y_{SC} + K \cdot Y_K + R_D + C_V \cdot x_{CV} + S_{SP} \cdot x_{CV} + S_{SN} \cdot x_{aer} \right)$$

The matrix equation is solved using one of the standard matrix solvers (see section 17.4). As a result the isotope fractions, Y_i, on all aerosol particles are calculated for all Control Volumes.

12.3.12 Decay Heat Distribution

This section describes how the decay heat of the isotopes that are released from the core is distributed among the gas and liquid spaces of a Control Volume and solid structures. The default modelling is based on the following assumptions:

- β-radiation is absorbed by the gas in the atmosphere of a Control Volume.
- γ-radiation passes through the gas and is absorbed by solid structures and water pool.

Division of the decay heat is discussed below in the following parts.

- Isotopes present in the atmosphere of a Control Volume are discussed in section 12.3.12.1.
- Isotopes present in the pool of a Control Volume are discussed in section 12.3.12.2.
- Isotopes present in the solid structures (1-D and 2-D Solid Heat Conductors) are discussed in section 12.3.12.3 and 12.3.12.4.

Calculation of the decay heat is generation in Control Volumes and Heat Conductors is described in section 12.3.12.5.

12.3.12.1 Isotopes Present in the Atmosphere of a Control Volume

Radioactive isotopes may be present in the atmosphere of a Control Volume as vapors or being attached to the aerosol particles. If there are no structures in contact with the CV and no water pool in the CV, then the total heat is deposited in the CV atmosphere. Otherwise the decay heat produced by these isotopes is distributed as follows.

- Fraction of the decay heat absorbed in the atmosphere of the CV:

$$\beta \cdot x_{atms,\beta} + \gamma \cdot x_{atms,\gamma}$$

β	fraction of energy carried by β-radiation
γ	fraction of energy carried by γ-radiation
$x_{atms,\ \beta}$	fraction of β-radiation absorbed in the gas space of the CV (default value of 1.0)
$x_{atms,\ \gamma}$	fraction of γ-radiation absorbed in the gas space of the CV (default value of 0.0)

The range of β-radiation of energy of 2 - 3 MeV in air is about 10 m ([92], section 3.9, figure 3.25). Within this distance all energy of β-particles is absorbed. Therefore, if the radiation beam length in the Control Volume is smaller than that, then a value smaller than one may be more appropriate. For example, for the beam length of about 1 m the value should be $x_{atms,\ \beta}$~0.1. The γ-radiation practically passes through the gas unabsorbed, therefore the default value, $x_{atms,\ \gamma} = 0.0$, is appropriate for all calculations.

- Fraction of the decay heat absorbed by the structures in contact with the CV:

$$\beta \cdot (1 - x_{atms,\beta}) + \gamma \cdot (1 - x_{atms,\gamma})$$

This fraction is distributed among all structures in contact with the same CV, proportional to the surface area of the structure. The proportions may be changed by the user-defined weighting factors, w_{SC}. The fraction of heat absorbed by the structure SC, $F_{abs,SC}$ is equal to:

$$F_{abs,SC} = \frac{A_{SC} \cdot |w_{SC}|}{\sum_L A_L \cdot |w_L|}$$

A_{SC} surface area of the solid heat conductor in contact with the CV, (m^2)
w_{SC} weighting factor of the surface SC (default value of –1.0, the meaning of a negative value is explained in section 12.3.12.3)

The summation is performed over all structures (1-D as well as 2-D) that are in contact with the CV, including the one with the radioactive source. Therefore the index *L* stands for any SC as well as TC in contact with the Control Volume CV. The values of $F_{abs,SC}$ are pre-computed during the input processing and are printed next to the weighting factors for CV related data of the RT Package, as a "CV enclosure data for decay heat calculations".

If a structure is partly covered by the water pool, then only part of this energy is absorbed by the SC:

$$F_{abs,SC} \cdot (1 - X_{pool,SC} \cdot w_{pool,CV})$$

The remaining part is absorbed in the pool:

$$F_{abs,SC} \cdot X_{pool,SC} \cdot w_{pool,CV}$$

$X_{pool,SC}$ fraction of the surface area covered by the pool, (-)
$w_{pool,CV}$ weighting factor, (-), equal to:

$$w_{pool,CV} = \begin{cases} 0.0 & if \quad Z_{pool} < Z_1 = 1 \times 10^{-3} \\ \dfrac{Z_{pool} - Z_1}{Z_2 - Z_1} & if \quad Z_1 < Z_{pool} < Z_2 \\ 1.0 & if \quad Z_{pool} > Z_2 = 2 \times 10^{-3} \end{cases}$$

Z_{pool} water level in the CV, (m)

The interpolation described above is applied in order to eliminate the pool heating when there is a very small amount of water in the CV.

12.3.12.2 Isotopes Present in the Pool of a Control Volume

Radioactive isotopes may be present in the pool of a Control Volume due to deposition of radioactive particles, adsorption on the pool surface, or desorption from solid surfaces that are covered by the water pool. The decay heat produced by these isotopes is distributed as follows.

- Fraction of the decay heat absorbed in the pool of the CV:

$$w_{pool,CV}$$

- Fraction of the decay heat absorbed by the atmosphere of the CV:

$$1 - w_{pool,CV}$$

$w_{pool,CV}$ is defined in section 12.3.12.1.

12.3.12.3 Isotopes Present on the Surface of Solid Structures

Radioactive isotopes may be present on the surface of solid structures (1-D or 2-D Solid Heat Conductors) due to deposition of radioactive dust particles or as reversibly bound particles that have been adsorbed from the atmosphere of a Control Volume. The decay heat produced by these isotopes is distributed as follows.

- Fraction of the decay heat absorbed in the structure:

$$\frac{1}{2} \cdot (\beta + \gamma)$$

β fraction of energy carried by β-radiation
γ fraction of energy carried by γ-radiation

Half of the decay heat is absorbed by the structure itself at the point of emission. The rest may go to the atmosphere, pool, other structures, or the structure itself (absorbed in another place than emitted).

- Fraction of the decay heat absorbed in the CV atmosphere:

$$\frac{1}{2} \cdot \left(\beta \cdot x_{atms,\beta} + \gamma \cdot x_{atms,\gamma}\right) \cdot \left(1 - X_{pool,SC} \cdot w_{pool,CV}\right)$$

$x_{atms,\beta}$ fraction of β-radiation absorbed in the gas space of the CV (default value of 1.0)
$x_{atms,\gamma}$ fraction of γ-radiation absorbed in the gas space of the CV (default value of 0.0)
$X_{pool,SC}$ fraction of the surface area covered by the pool, (-)
$w_{pool,CV}$ weighting factor, (-), described in section 12.3.12.1

- Fraction of the decay heat absorbed in the CV pool:

$$\frac{1}{2} \cdot (\beta + \gamma) \cdot \left(X_{pool,SC} \cdot w_{pool,CV}\right)$$

- Fraction of the decay heat absorbed by other structures, SC', in contact with the same CV as the generating structure SC:

$$\frac{1}{2}\cdot\left[\beta\cdot(1-x_{atms,\beta})+\gamma\cdot(1-x_{atms,\gamma})\right]\cdot(1-X_{pool,SC}\cdot w_{pool,CV})\cdot F_{abs,SC'}$$

  This fraction is distributed among all structures in contact with the same CV, proportional to the surface area of the structure. The proportions may be changed by the user-defined weighting factors, w_{SC}, described in section 12.3.12.1.

If the weighting factor is specified as negative, w_{SC}, < 0.0 (default), the generating surface is not included in the distribution of the radiation emitted from its surface. Physically it means that the structure cannot radiate to itself (Figure 12-114, left). In such case the structure absorbs only half of the radiation generated at its surface. The other half is distributed among other structures of the enclosure, gas or pool. This means the fraction of the generated decay heat that the source structure receives is equal to:

$$\frac{1}{2}(\beta+\gamma)$$

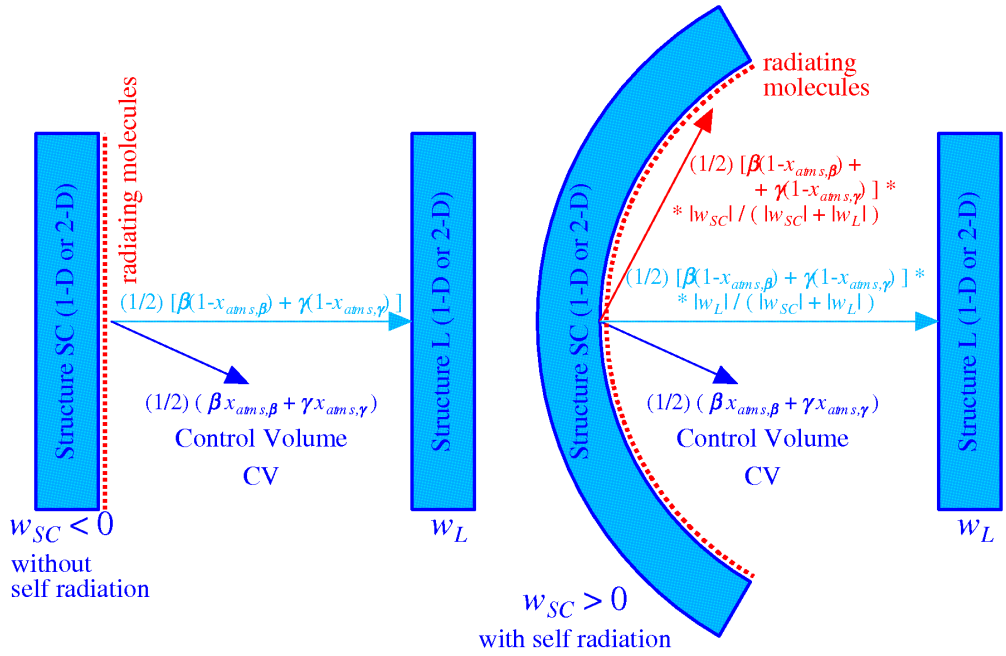


Figure 12-114 Left: w_{SC}<0, no self-radiation, right: w_{SC}>0, with self-radiation.

If the weighting factor is specified as positive, $w_{SC} > 0.0$, the generating surface is included in the distribution of the radiation emitted from its surface. Physically it means that the structure can radiate to itself (Figure 12-114, right). The total fraction of the generated decay heat that the source structure receives is equal to:

$$\frac{1}{2}(\beta+\gamma)+\frac{1}{2}\left[\beta(1-x_{atms,\beta})+\gamma(1-x_{atms,\gamma})\right]\cdot(1-X_{pool,SC}\cdot w_{pool,CV})\cdot F_{abs,SC}$$

For example, if there is no absorption in the atmosphere and in the pool ($x_{atms,\beta} = x_{atms,\gamma} = X_{pool,SC} = 0.0$) the fraction of radiated energy that is absorbed by the generating structure is equal to:

$$\begin{aligned} &\frac{1}{2}(\beta+\gamma) && \text{if} \quad w_{SC} < 0.0 \\ &\frac{1}{2}(\beta+\gamma)+\frac{1}{2}(\beta+\gamma)\cdot F_{abs,SC} && \text{if} \quad w_{SC} > 0.0 \end{aligned}$$

Another example is shown in Figure 12-115. In the example presented in this figure there is absorption in the atmosphere, with the default values of $x_{atms,\beta} = 1.0$ (full absorption of β-radiation in the gas) and $x_{atms,\gamma} = 0.0$ (no absorption of γ-radiation in the gas). The fractions of energy carried by β- and γ-radiation are assumed to be equal, $\beta = 0.5$, $\gamma = 0.5$.

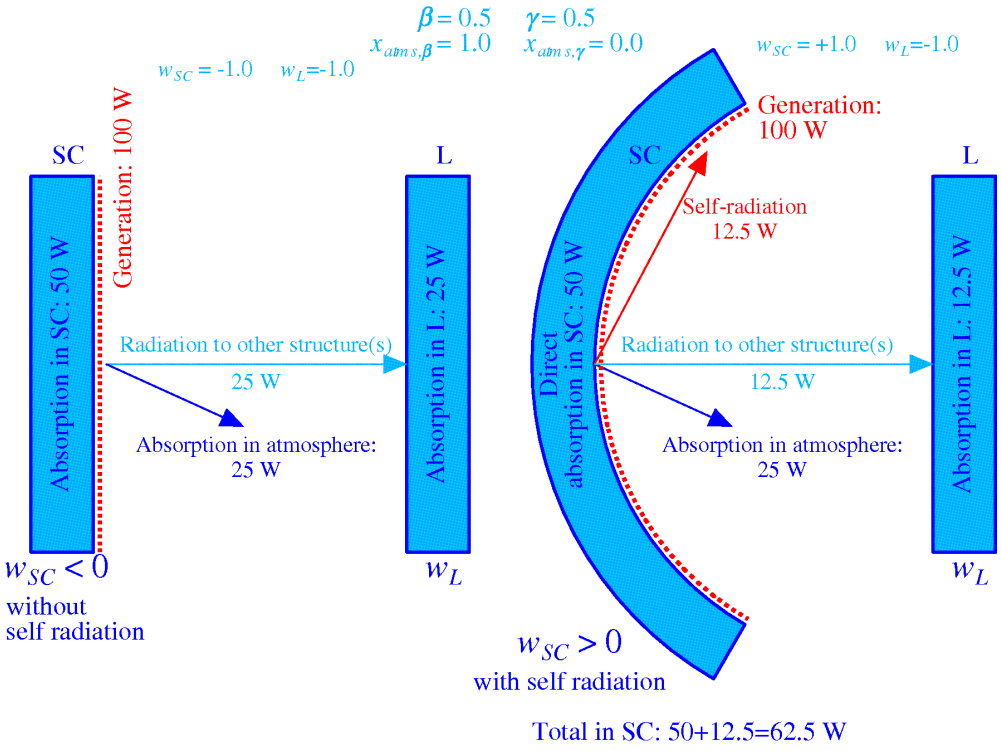


Figure 12-115 Example case with no pool, left: w_{SC}<0, right: w_{SC}>0.

The surface areas of the structures are the same, and equal to 1.0 m^2. Radioactive particles are presented on the surface of the structure SC, in the form of adsorbed, reversibly bound molecules. The energy generated by these particles is 100 W. No other radioactive sources are present (this is a test case described in more detail in Volume 3). Two cases are considered:

- Case 1: weighting factor of the SC surface equal to $w_{SC} = -1.0$ (no self radiation)
- Case 2: weighting factor of the SC surface equal to $w_{SC} = +1.0$ (with self radiation)

- **Case 1**
  The case 1 is shown in the left part of Figure 12-115. Half of the energy, namely 50 W, is absorbed by the SC, (radiation is uniform in all directions). The remaining half is entering the atmosphere, where the β-radiation is fully absorbed. Since $\beta = 0.5$, the amount of energy absorbed in the atmospheric gas (and thus generated in the CV atmosphere) is equal to 25 W. The γ-radiation passes through the atmosphere to be absorbed by other structures. The only other structure present is the structure L. Since the structure SC cannot radiate to itself in this case ($w_{SC} = -1.0$), all this γ-radiation is absorbed by the structure L. Therefore the energy generated in L is 25 W. If there were no other structures present in the CV except for the SC, then this energy would be generated in the atmosphere of CV. Therefore the energy division in this case is:

  - Structure SC: 50 W
  - Structure L: 25 W
  - Atmosphere of CV: 25 W

- **Case 2**
  The case 2 is shown in the right part of Figure 12-115. Again, half of the energy, namely 50 W, is absorbed by the SC, (uniform radiation in all directions). The remaining half is entering the atmosphere, where the β-radiation is fully absorbed. Since $\beta = 0.5$, the amount of energy absorbed in the atmospheric gas (and thus generated in the CV atmosphere) is equal to 25 W. The γ-radiation passes through the atmosphere to be absorbed by other structures. This energy is now distributed between the two structures (SC and L) present because the structure SC can radiate to itself in this case ($w_{SC} = +1.0$). Since the surface areas are equal and the absolute values of the weighting factors are equal, the energy is equally distributed in both structures. This means 12.5 W in the structure L and 12.5 W in the structure SC. Therefore the energy division in this case is:

  - Structure SC: 50 + 12.5 = 62.5 W
  - Structure L: 12.5 W
  - Atmosphere of CV: 25 W

Another example is shown in Figure 12-116. This example is identical to the previous one, but now a water pool is present in the CV. This water pool is assumed to cover half of the structure SC and half of the structure L. Again two cases are considered

- Case 1: weighting factor of the SC surface equal to $w_{SC} = -1.0$ (no self radiation)
- Case 2: weighting factor of the SC surface equal to $w_{SC} = +1.0$ (with self radiation)

- **Case 1**
  The case 1 is shown in the left part of Figure 12-116. Half of the energy, namely 50 W, is absorbed by the SC, (uniform radiation in all directions). The remaining half is entering the atmosphere and pool, with equal fractions, since the pool covers half of the surface. This means 25 W enters the pool and 25 W enters the atmosphere. In the atmosphere β-radiation is fully absorbed. Since $\beta = 0.5$, the amount of energy absorbed in the atmospheric gas (and thus generated in the CV atmosphere) is equal to 12.5 W. The γ-radiation passes through the atmosphere to be absorbed by other structures. The only other structure present is the structure L. Since the structure SC cannot radiate to itself in this case ($w_{SC} = -1.0$), all this γ-radiation is directed to the structure L. However, half of the surface of the structure L is covered with water. Therefore, half of this energy, namely 6.25 W, is absorbed in the water, while the other half is absorbed by the structure L, where it generates 6.25 W. The energy division in this case is:

  - Structure SC: 50 W
  - Structure L: 6.25 W
  - Atmosphere of CV: 12.5 W
  - Pool of CV: 25 + 6.25 = 31.25 W

- **Case 2**
  The case 2 is shown in the right part of Figure 12-116. Again, half of the energy, namely 50 W, is absorbed by the SC, (uniform radiation in all directions). The remaining half is entering the atmosphere and pool, with equal fractions, since the pool covers half of the surface. This means 25 W enters the pool and 25 W enters the atmosphere. In the atmosphere β-radiation is fully absorbed. Since $\beta = 0.5$, the amount of energy absorbed in the atmospheric gas (and thus generated in the CV atmosphere) is equal to 12.5 W. The γ-radiation passes through the atmosphere to be absorbed by other structures. This energy is now distributed between the two structures (SC and L) present because the structure SC can radiate to itself in this case ($w_{SC} = +1.0$). Since the surface areas are equal and the absolute values of the weighting factors are equal, the energy is equally distributed in both structures. This means 12.5 W in the structure L and 12.5 W in the structure SC. However, half of the surface of the structure L is covered with water. Therefore, both of these structures are partly (half) covered by the water pool. Therefore half of the energy that is directed to SC (6.25 W) and to L (6.25 W) is absorbed by the water pool, while the other half is absorbed by the structures SC and L themselves. The energy division in this case is:

  - Structure SC: 50 + 3.125 = 53.125 W
  - Structure L: 3.125 W
  - Atmosphere of CV: 12.5 W
  - Pool of CV: 25 + 3.125 + 3.125 = 31.25 W

Note that in both cases 1 and 2 the energy generated in the water pool is the same (31.25 W). The radiation absorbed by the pool does not depend on the surface weighting factors.

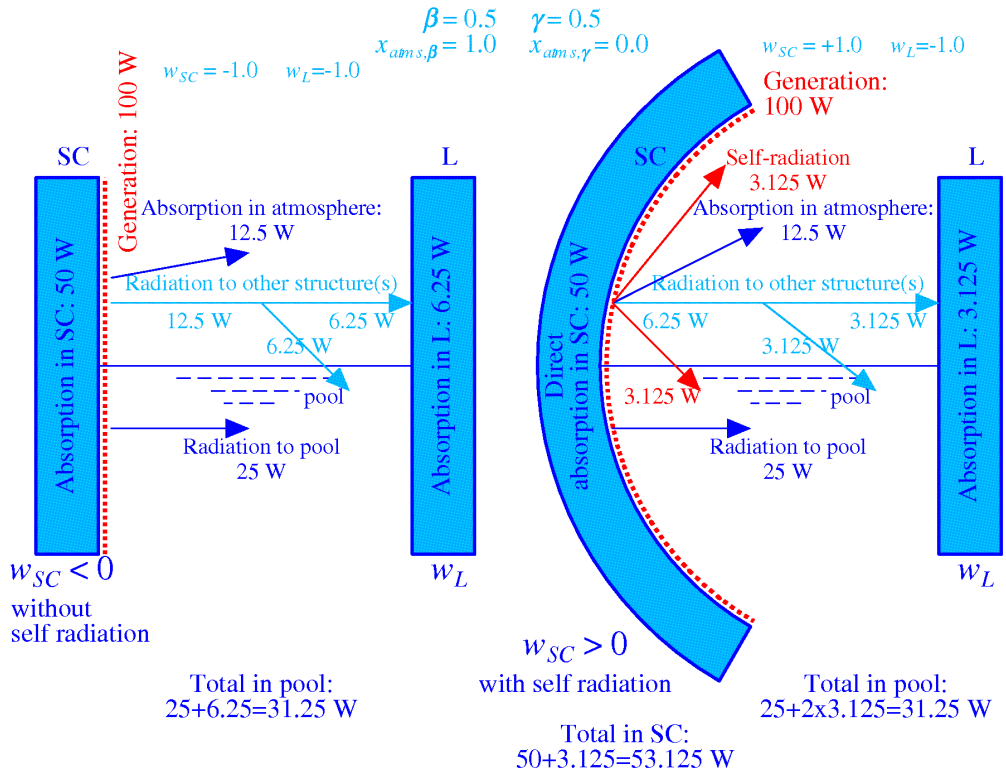


Figure 12-116 Example case with pool, left: $w_{SC}<0$, right: $w_{SC}>0$.

12.3.12.4 Isotopes Present inside Solid Structures

Radioactive isotopes may be present inside solid structures due to penetration of the adsorbed vapors into the material of the 1-D or 2-D Solid Heat Conductors (diffusion part of the adsorption flux in section 12.3.5.5). The decay heat produced by these isotopes is distributed as follows.

- Fraction of the decay heat absorbed in the structure:

$$\beta \cdot x_{solid,\beta} + \gamma \cdot x_{solid,\gamma} = \beta + \gamma$$

β	fraction of energy carried by β-radiation
γ	fraction of energy carried by γ-radiation
$x_{solid,\beta}$	fraction of β-radiation absorbed in the solid material of SC (default value of 1.0)
$x_{solid,\gamma}$	fraction of γ-radiation absorbed in the solid material of SC (default value of 1.0)

Currently only the default values may be used. Implementation of other values may be done in the future if necessary. With the default values of $x_{solid,\beta}$, $x_{solid,\gamma}$ the above formula gives the value of one (full absorption within the solid material). Therefore there is no heat generation in other materials than the generating SC/TC.

- Fraction of the decay heat absorbed in the CV atmosphere: 0.0. With the current modelling there is no heat generation due to decay of the isotopes that have penetrated inside the SC/TC material.

- Fraction of the decay heat absorbed in the CV pool: 0.0. With the current modelling there is no heat generation due to decay of the isotopes that have penetrated inside the SC/TC material.

12.3.12.5 Calculation of the Decay Heat Generation in CV, SC, and TC

Formulae used to calculate the decay heat generated within the Control Volumes (CV), 1-D Solid Heat Conductors (SC), and 2-D Solid Heat Conductors (TC), are described below. The formulae are based on the divisions of decay heat, discussed in sections 12.3.12.1 through 12.3.12.4.

- **Atmosphere of a Control Volume**

  The heat generated within the atmosphere of a Control Volume is equal to (compare sections 12.3.12.1 through 12.3.12.4):

$$
\begin{aligned}
Q_{CV} = & \sum_{i \in atms} \left(\lambda_i N_{i,CV} q_i\right) \cdot \left(\beta_i \cdot x_{atms,\beta} + \gamma_i \cdot x_{atms,\gamma}\right) \quad + \\
& + \sum_{i \in pool} \left(\lambda_i N_{i,pool} q_i\right) \cdot \left(1 - w_{pool,CV}\right) \quad + \\
& + \sum_{SC} \left\{ \quad \frac{1}{2} \sum_{i \in SC} \left(\lambda_i N_{i,SC-IBC} q_i\right) \cdot \left(\beta_i \cdot x_{atms,\beta} + \gamma_i \cdot x_{atms,\gamma}\right) \cdot \left(1 - X_{pool,SC} \cdot w_{pool,CV}\right) \quad \right\} \quad + \\
& + \sum_{TC} \left\{ \quad \frac{1}{2} \sum_{i \in TC} \left(\lambda_i N_{i,TC-IBC} q_i\right) \cdot \left(\beta_i \cdot x_{atms,\beta} + \gamma_i \cdot x_{atms,\gamma}\right) \cdot \left(1 - X_{pool,TC} \cdot w_{pool,CV}\right) \quad \right\}
\end{aligned}
$$

  Q_{CV} total decay heat generated within the CV atmosphere, (W)
  λ_i decay constant of the isotope i, (s^{-1})
  q_i heat generated per decay of the isotope i, (W)
  β_i fraction of energy carried by β-radiation in case of decay of the isotope i, (-)
  γ_i fraction of energy carried by γ-radiation in case of decay of the isotope i, (-)
  $x_{atms,\beta}$ fraction of β-radiation absorbed in the gas space of the CV (default value of 1.0)
  $x_{atms,\gamma}$ fraction of γ-radiation absorbed in the gas space of the CV (default value of 0.0)
  $N_{i,CV}$ number of molecules of the isotope i, in the CV atmosphere
  $N_{i,pool}$ number of molecules of the isotope i, in the CV pool
  $N_{i,SC-IBC}$ number of molecules of the isotope i, on the boundary surface IBC (left or right) of the structure SC (reversibly bound molecules or deposited aerosols)
  $N_{i,TC}$ number of molecules of the isotope i on the boundary surface IBC of the structure TC (reversibly bound molecules or deposited aerosols)

  The first summation is performed over all isotopes from all isotope chains that are present in the CV atmosphere. The second summation is performed over all isotopes from all isotope chains that are present in the CV pool.

The next summation is performed over all SC surfaces in contact with the CV. Within this sum a summation over all isotopes from all isotope chains that are present on the surface of the SC, is performed.

Finally, the last summation is performed over all TC surfaces in contact with the CV. Within this sum again a summation over all isotopes from all isotope chains that are present on the surface of the TC.

- **Pool of a Control Volume**

The heat generated within the pool of a Control Volume is equal to (compare sections 12.3.12.1 and 12.3.12.2):

$$
\begin{aligned}
Q_{pool} = & \sum_{i \in pool} \left(\lambda_i N_{i,pool} q_i \right) \cdot w_{pool,CV} \quad + \\
& + \quad \sum_{i \in atms} \left(\lambda_i N_{i,CV} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \quad \times \\
& \times \quad \left(\sum_{SC} X_{pool,SC} \cdot F_{abs,SC} + \sum_{TC} X_{pool,TC} \cdot F_{abs,TC} \right) \cdot w_{pool,CV} \; + \\
& + \sum_{SC'} \left\{ \frac{1}{2} \sum_{i \in SC'} \left(\lambda_i N_{i,SC'} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \cdot \left(X_{pool,SC'} \cdot w_{pool,CV} \right) \right\} + \\
& + \sum_{TC'} \left\{ \frac{1}{2} \sum_{i \in TC'} \left(\lambda_i N_{i,TC'} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \cdot \left(X_{pool,TC'} \cdot w_{pool,CV} \right) \right\}
\end{aligned}
$$

All symbols are explained above in the atmospheric heat calculation and in the sections 12.3.12.1 through 12.3.12.4. The first summation is performed over all isotopes from all isotope chains that are present in the CV pool. The second summation is performed over all isotopes from all isotope chains that are present in the CV atmosphere.

- **1-D Solid Heat Conductors**

The heat generated within the 1-D Solid Heat Conductors is equal to (compare sections 12.3.12.3 and 12.3.12.4):

$$
\begin{aligned}
Q_{SC} = & \quad \sum_{i \in SC} \left(\lambda_i N_{i,SC} q_i \right) \cdot \left(1 - \frac{1}{2} \cdot \frac{N_{i,SC-left} + N_{i,SC-right}}{N_{i,SC}} \right) \quad + \\
& + \quad \sum_{i \in atms} \left(\lambda_i N_{i,CV} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \cdot \left(1 - X_{pool,SC} \cdot w_{pool,CV} \right) \cdot F_{abs,SC} \quad + \\
& + \sum_{SC'} \left\{ \frac{1}{2} \sum_{i \in SC'} \left(\lambda_i N_{i,SC'-IBC} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \cdot \left(1 - X_{pool,SC'} \cdot w_{pool,CV} \right) \cdot F_{abs,SC'} \right\} + \\
& + \sum_{TC'} \left\{ \frac{1}{2} \sum_{i \in TC'} \left(\lambda_i N_{i,TC'-IBC} q_i \right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma}) \right) \cdot \left(1 - X_{pool,TC'} \cdot w_{pool,CV} \right) \cdot F_{abs,TC'} \right\}
\end{aligned}
$$

$N_{i,SC\text{-}left}$ number of molecules of the isotope *i* on the left surface of the SC (reversibly bound molecules or deposited as aerosols)
$N_{i,SC\text{-}right}$ number of molecules of the isotope *i* on the right surface of the SC
$N_{i,SC}$ total number of molecules of the isotope *i* inside and on both surfaces of the SC
$N_{i,SC'\text{-}IBC}$ number of molecules of the isotope *i* on the boundary surface (left or right) IBC of a SC (reversibly bound molecules or deposited as aerosols), (-)
$N_{i,TC\text{-}IBC}$ number of molecules of the isotope *i* on the surface of the boundary cell IBC of a TC (reversibly bound molecules or deposited as aerosols), (-)

All symbols are explained above in the atmospheric heat calculation and in the sections 12.3.12.1 through 12.3.12.4.The first two summations are performed over all isotopes from all isotope chains that are present on the surface or inside the SC. The next summation is performed over all isotopes from all isotope chains that are present in the CV atmosphere.

The next summation is performed over all 1-D surfaces, SC' in contact with the same CV as the generating surface SC, SC' ≠ SC. Within this sum a summation over all isotopes from all isotope chains that are present on the surface of the SC', is performed.

Finally, the last summation is performed over all TC' surfaces in contact with the same CV as the generating surface SC. Within this sum again a summation over all isotopes from all isotope chains that are present on the surface of the TC' is performed.

- **2-D Solid Heat Conductors**

The heat generated within the 2-D Solid Heat Conductors is equal to (compare sections 12.3.12.3 and 12.3.12.4):

$$
\begin{aligned}
Q_{TC} = & \sum_{i \in TC} \left(\lambda_i N_{i,TC} q_i\right) \cdot \left(1 - \frac{1}{2} \cdot \sum_{IBC} \frac{N_{i,TC-IBC}}{N_{i,TC}}\right) + \\
& + \sum_{i \in atms} \left(\lambda_i N_{i,CV} q_i\right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma})\right) \cdot (1 - X_{pool,SC} \cdot w_{pool,CV}) \cdot F_{abs,SC} + \\
& + \sum_{SC'} \left\{ \frac{1}{2} \sum_{i \in SC'} \left(\lambda_i N_{i,SC'-IBC} q_i\right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma})\right) \cdot (1 - X_{pool,SC'} \cdot w_{pool,CV}) \cdot F_{abs,SC'} \right\} + \\
& + \sum_{TC'} \left\{ \frac{1}{2} \sum_{i \in TC'} \left(\lambda_i N_{i,TC'-IBC} q_i\right) \cdot \left(\beta_i (1 - x_{atms,\beta}) + \gamma_i (1 - x_{atms,\gamma})\right) \cdot (1 - X_{pool,TC'} \cdot w_{pool,CV}) \cdot F_{abs,TC'} \right\}
\end{aligned}
$$

$N_{i,TC\text{-}IBC}$ number of molecules of the isotope *i* on the surface of the boundary cell IBC of a TC (reversibly bound molecules or deposited as aerosols), (-)
$N_{i,TC}$ total number of molecules of the isotope *i* inside the TC and on all surfaces of the TC, (-)

All symbols are explained above in the atmospheric heat calculation and in the sections 12.3.12.1 through 12.3.12.4. The first two summations are performed over all isotopes from all isotope chains that are present on the surface or inside the TC. The next summation is performed over all isotopes from all isotope chains that are present in the CV atmosphere.

The next summation is performed over all 1-D surfaces SC' in contact with the same CV as the generating surface TC. Within this sum a summation over all isotopes from all isotope chains that are present on the surface of the SC', is performed.

Finally, the last summation is performed over all 2-D surfaces TC' in contact with the same CV as the generating surface TC, TC' ≠ TC. Within this sum again a summation over all isotopes from all isotope chains that are present on the surface of the TC' is performed.

12.3.13 Radiation Exposure and Doses

12.3.13.1 Radiation Exposure and Dose from γ-radiation

In SPECTRA an external exposure and dose from γ-rays is estimated if the Radioactive Particle Transport Package is used. The method of calculating those values is described below.

- Radiation Exposure

  External exposure from γ-rays of energy E_γ is calculated for each Control Volume using the following formula ([92], section 9.9, equations 9.20, 9.29):

$$X_\gamma(E_\gamma) = 1.83\times10^{-11}\cdot\varphi_\gamma\cdot E_\gamma\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{air}$$

  X exposure rate, (R/s)
  φ_γ γ-radiation flux, ($m^{-2}\ s^{-1}$)
  E_γ energy of photons, (MeV) (the energy of photons is equal to the total decay energy multiplied by the fraction of energy carried by photons. The data for each built-in isotope is shown in Figure 12-53 through Figure 12-62)
  $(\mu_{a,\gamma}/\rho)_{air}$ mass absorption coefficient in the air, (m^2/kg)

  In SPECTRA the SI units are used, therefore the radiation flux is expressed in ($m^{-2}\cdot s^{-1}$), rather than in ($cm^{-2}\cdot s^{-1}$), and the mass absorption coefficient in (m^2/kg), rather than (cm^2/g), as in [92]. Consequently the constant multiplier in the above equation is equal to 1.83×10^{-11}, rather than 1.83×10^{-8}. The mass absorption coefficient $(\mu_a/\rho)_{air}$, is tabulated versus energy E_γ, in [92], Table II.5. The values applied in SPECTRA (SI units) are shown in Table 12-14. The radiation flux, φ_γ, represents the total flux from multi-directions, and is related to the mono-directional radiation intensity, I, by:

$$\varphi_\gamma = \int_\Omega I(\Omega)\, d\Omega$$

  The integral is taken over all directions, Ω.

- Dose

  The dose from γ-rays of energy E_γ is calculated for each Control Volume using the following formula ([92], section 9.9, equations 2.25, 2.30):

$$D_\gamma(E\gamma) = 1.60\times10^{-11}\cdot\varphi_\gamma\cdot E_\gamma\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{tissue}$$

  D_γ dose rate, (rad/s)
  $(\mu_{a,\gamma}/\rho)_{tissue}$ mass absorption coefficient in the tissue, (m^2/kg)

Table 12-14 Mass absorption coefficients for air and tissue [92].

Energy, (MeV)	$(\mu_a/\rho)_{air}$, (m^2/kg)	$(\mu_a/\rho)_{tissue}$, (m^2/kg)
0.10	0.00233	0.00271
0.15	0.00251	0.00282
0.20	0.00268	0.00293
0.30	0.00288	0.00312
0.40	0.00296	0.00317
0.50	0.00297	0.00320
0.60	0.00296	0.00319
0.80	0.00289	0.00311
1.00	0.00280	0.00300
1.25	0.00268	0.00288
1.50	0.00256	0.00276
2.00	0.00238	0.00256
3.00	0.00211	0.00220
4.00	0.00194	0.00206
5.00	0.00181	0.00192
6.00	0.00172	0.00182
8.00	0.00160	0.00168
10.00	0.00153	0.00160

The mass absorption coefficients are shown in Table 12-14. The total exposure and dose rates are calculated by summing the dose from all radioactive isotopes:

$$X_\gamma = \sum_i X_{\gamma,i} = \sum_i 1.83\times10^{-11}\cdot E_{\gamma,i}\cdot\varphi_{\gamma,i}\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{air} \quad \text{(R/s)}$$

$$D_\gamma = \sum_i D_{\gamma,i} = \sum_i 1.60\times10^{-11}\cdot E_{\gamma,i}\cdot\varphi_{\gamma,i}\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{tissue} \quad \text{(rad/s)}$$

The above formulae give exposure and dose rates in the still commonly used units of R/s and rad/s respectively. In the SI units exposure is measured in C/kg rather than Roentgen, where 1 R = 2.58×10^{-4} C/kg. The SI unit of dose is J/kg = Gy, equal to 1 Gy = 100.0 rad. Therefore in SI units the exposure rate and the dose rate are given by:

$$X_\gamma = \sum_i X_{\gamma,i} = \sum_i 4.72\times10^{-15}\cdot E_{\gamma,i}\cdot\varphi_{\gamma,i}\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{air} \quad \text{(C/kg-s)}$$

$$D_\gamma = \sum_i D_{\gamma,i} = \sum_i 1.60\times10^{-13}\cdot E_{\gamma,i}\cdot\varphi_{\gamma,i}\cdot\left(\frac{\mu_{a,\gamma}}{\rho}\right)_{tissue} \quad \text{(Gy/s)}$$

The summation is performed over all isotopes *i* from all isotope chains. In order to compute the above formulae, one needs to determine the radiation fluxes for all energies. The method of calculating the flux is described below.

The radiation can be coming from radioactive isotopes suspended in the atmosphere of a Control Volume (fission product vapors, aerosols), and from the isotopes deposited on various surfaces within the Control Volume. Before the fluxes from these sources are calculated it is useful to write a formula for a radiation flux at a distance *r* from an isotropic point source, emitting *S* (photons/s):

$$\varphi(r) = \frac{S}{4\pi r^2} \cdot B(\mu r) \cdot e^{-\mu r}$$

S	γ-radiation source, (s^{-1})
r	distance, (m)
$\varphi(r)$	γ-radiation flux at the distance *r* from the point source, ($m^{-2}\ s^{-1}$)
μ	absorption coefficient in the air, (1/m)
$B(\mu r)$	build-up factor, (-)

In the typical applications of SPECTRA the absorption of γ-rays in the air may be neglected, μ=0. In such case absorption and buildup terms disappear, $B(\mu r)\cdot\exp(-\mu r) = 1.0$, and:

$$\varphi(r) = \frac{S}{4\pi r^2}$$

Calculation of the γ-radiation flux is described below. Two cases are considered: first the airborne sources and next the deposited sources.

- Radiation flux from airborne sources.

  Consider an airborne source with the density of S_V, (Bq/m^3). In general the source may be non-uniform, so $S_V = S_V(r)$. The total flux at a given location is obtained by the following integral:

$$\varphi_{air}(\bar{r}) = \iiint_V \frac{S_V(\bar{r}_0)}{4\pi(\bar{r} - \bar{r}_0)^2}\, d^3\bar{r}_0$$

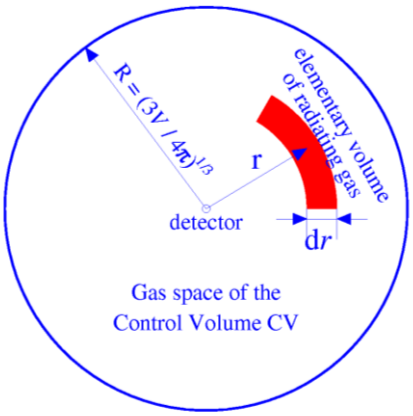
  The integration is performed over all volume of a CV. Calculation of this integral may be in general quite complicated, even when the source is uniform, S_V = const (this is by definition the case in all SPECTRA Control Volumes). Moreover the information on Control Volume geometry, available in the Control Volume Package, is insufficient to calculate such integral. Therefore the flux is conservatively estimated using a simplified method.

  Control Volumes are assumed to be of spherical shapes, with the flux detector point at the sphere center ("equivalent CV" - Figure 12-117). This gives a conservative flux ([92], section 11.5) because in case sphere the average distance between the center and any radiating point of the atmosphere is the smallest compared to all other shapes.

  The elementary source at the distance *r* from the detection point is equal to:

$$dS(r) = S_V\, 4\pi r^2 dr$$

The elementary flux from this source is, for the geometry shown in Figure 12-117, given by:



$$d\varphi_{air} = \frac{S_V \, 4\pi r^2 dr}{4\pi r^2} = S_V \, dr$$

Note that reference [92], section 11.5, equation 11.41 gives $d\varphi_{air} = (S_V/2)dr$. This is because in [92] a hemi-sphere is considered, a geometry that is appropriate for large radioactive clouds in the atmosphere. For the same reason (large cloud) absorption in the air is not negligible and the term $B(\mu r)\cdot\exp(-\mu r)$ is kept in the equation in [92].

Figure 12-117 "Equivalent CV", airborne sources.

The total flux is obtained by integrating over all volume:

$$\varphi_{air} = \int_0^R S_V \, dr = S_V R = S_V \cdot \sqrt[3]{\frac{3V_{CV}}{4\pi}} = 0.620 \cdot S_V \cdot V_{CV}^{1/3}$$

where V_{CV} is the volume of gas in the Control Volume CV. The volumetric source is available from the RT package. To be consistent with the rest of the RT package the total source, S, in (Bq), is used instead of the volumetric source S_V, in (Bq/m^3). The volumetric source is, of course, equal to: $S_V = S/V$. Therefore.

$$\varphi_{air} = 0.620 \cdot \frac{S}{V_{CV}^{2/3}}$$

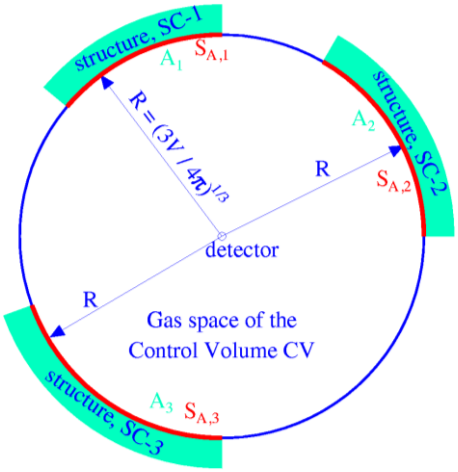
- Radiation flux from deposited sources.

  Consider a deposited source with the density of S_A, (Bq/m^2). In general the source may be non-uniform, so $S_A = S_A(r)$. The total flux at a given location is obtained by the following integral:

$$\varphi_{dep}(\bar{r}) = \iint_A \frac{S_A(\bar{r_0})}{4\pi(\bar{r} - \bar{r_0})^2} \, d^2\bar{r_0}$$

  where the integration is performed over all radioactive surfaces. Again the calculations are performed assuming that the Control Volume is spherical and that all surfaces are located at the distance equal to the sphere radius, R, from the detection point (sphere center) – see Figure 12-118.

The elementary flux from this source is, for the geometry shown in Figure 12-118, given by:



$$d\varphi_{dep} = \frac{S_A dA}{4\pi R^2}$$

The elementary flux from this source is therefore given by:

$$d\varphi_{dep} = \frac{S_A dA}{4\pi R^2}$$

Figure 12-118 "Equivalent CV", deposited sources.

The radiation flux from a surface with a uniform source density is equal to:

$$\varphi_{dep} = \frac{S_A A}{4\pi R^2} = \frac{S_A A}{4\pi \left(3V_{CV} / 4\pi\right)^{2/3}} = 0.207 \frac{S}{V_{CV}^{2/3}}$$

where A is the total area of a given structure and S is the total source from the surface, in (Bq), equal to: $S_A \cdot A$. In SPECTRA a number of 1-D and 2-D structure surfaces can be facing the same Control Volume.

If a water pool is present in the CV, it is treated as another radiating surface. If filters are present, radiation from the radioactive isotopes deposited on the filters is added in the same way. The radiation flux from deposited sources is obtained by calculating a sum over the pool, all filters, all 1-D structures (SC), and all 2-D structures (TC) present in a given Control Volume:

$$\varphi_{dep} = \frac{0.207}{V_{CV}^{2/3}} \cdot \left(S_{CV-pool} + \sum_{JN \in CV} S_{i,JN-filter} + \sum_{SC \in CV} S_{SC} + \sum_{TC \in CV} S_{TC} \right)$$

- Total radiation flux

The total radiation flux for a given isotope i, is obtained by adding the flux from the airborne sources and the deposited sources:

$$\varphi_{\gamma,i} = \frac{0.620}{V_{CV}^{2/3}} \cdot S_{i,CV} + \frac{0.207}{V_{CV}^{2/3}} \cdot \left(S_{i,CV-pool} + \sum_{JN \in CV} S_{i,JN-filter} + \sum_{SC \in CV} S_{i,SC} + \sum_{TC \in CV} S_{i,TC} \right)$$

V_{CV} volume of gas in the Control Volume CV, (m^3)
$S_{i,CV}$ γ-radiation source from the isotope *i* in the atmosphere of the CV, (Bq)
$S_{i,CV\text{-}pool}$ γ-radiation source from the isotope *i* in the pool of the CV, (Bq)
$S_{i,SC}$ γ-radiation source from the isotope *i* present on the SC (in contact with the atmosphere of the CV), (Bq)
$S_{i,TC}$ γ-radiation source from the isotope *i* present in the TC (in contact with the atmosphere of the CV), (Bq)

Finally, in order to taken into account non-spherical shapes of the Control Volumes, two user-defined multipliers are introduced. These are the multiplier on the airborne source, $C_{\gamma,1}$, and the multiplier on the deposited source, $C_{\gamma,2}$. Default value of those multipliers is 1.0 (see Volume 2). The final formula for the γ-radiation flux calculation is:

$$\varphi_{\gamma,i} = C_{\gamma,1} \cdot \frac{0.620}{V_{CV}^{2/3}} \cdot S_{i,CV} + \\ + \quad C_{\gamma,2} \cdot \frac{0.207}{V_{CV}^{2/3}} \cdot \left(S_{i,CV-pool} + \sum_{JN \in CV} S_{i,JN-filter} + \sum_{SC \in CV} S_{i,SC} + \sum_{TC \in CV} S_{i,TC} \right)$$

- Source strength calculation

The strengths of the radioactive sources are calculated from:

$$S_{i,CV} = \lambda_i \cdot N_{i,CV}$$
$$S_{i,CV-pool} = \lambda_i \cdot N_{i,CV-pool}$$
$$S_{i,JN-filter} = \lambda_i \cdot N_{i,JN-filter}$$
$$S_{i,SC} = \lambda_i \cdot N_{i,SC}$$
$$S_{i,TC} = \lambda_i \cdot N_{i,TC}$$

λ_i decay constant of the isotope *i*, (s^{-1})
$N_{i,CV}$ number of molecules of the isotope *i* in the gas space of Control Volume CV, (-)
$N_{i,CV\text{-}pool}$ number of molecules of the isotope *i* present in the pool of CV, (-)
$N_{i,JN\text{-}filter}$ number of molecules of the isotope *i* deposited on a filter in the junction JN, (-)
$N_{i,SC}$ number of molecules of the isotope *i* deposited on the 1-D structure SC, (-)
$N_{i,TC}$ number of molecules of the isotope *i* deposited on the 2-D structure TC, (-)

In the above formulae $N_{i,SC}$ and $N_{i,TC}$ are the total numbers of radioactive molecules on the 1-D structure SC or the 2-D structure TC. This includes deposition on all surfaces as well as penetration into the material. This approach is taken because the γ-radiation is very weakly absorbed in structures. Most of the γ-radiation is passing through the structures unless those are heavy metal (for example lead), thick walls. Therefore the present approach is appropriate for most practical cases. Furthermore it gives the most conservative estimation of the radiation exposures and dose rates.

12.3.13.2 Radiation Dose from β-radiation

Radiation dose from β-radiation in rad/s is calculated from the same formula as the γ-ray exposure (in R/s) with the multiplication factor of 0.875 ([92], section 11.5):

$$D_\beta = 0.875 \cdot X_\gamma(\varphi_\beta, E_\beta)$$

Therefore:

$$D_\beta = \sum_i D_{\beta,i} = 0.875 \cdot \sum_i 1.83 \times 10^{-11} \cdot E_{\beta,i} \cdot \varphi_{\beta,i} \cdot \left(\frac{\mu_{a,\gamma}}{\rho}\right)_{air} \quad \text{(rad/s)}$$

In SI units:

$$D_\beta = \sum_i D_{\beta,i} = 0.875 \cdot \sum_i 1.60 \times 10^{-13} \cdot E_{\beta,i} \cdot \varphi_{\beta,i} \cdot \left(\frac{\mu_{a,\gamma}}{\rho}\right)_{air} \quad \text{(Gy/s)}$$

Calculation of the β-radiation flux is described below. Two cases are considered: first the airborne sources and next the deposited sources.

- Radiation flux from airborne sources.

  The geometry considered is shown in Figure 12-117. The elementary flux is given by:

$$d\varphi_{air} = \frac{S_V 4\pi r^2 \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho r)}}{4\pi r^2} dr = S_V \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho r)} dr$$

  Compared to the γ radiation (section 12.3.13.1) the absorption term is present since the absorption of β in the air is not negligible. The total flux is obtained by integrating over all volume:

$$\varphi_{air} = \int_0^R S_V \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho r)} dr = \frac{S_V \cdot \rho}{(\mu_{a,\beta}/\rho)} \cdot \left(1 - e^{-(\mu_{a,\beta}/\rho)\cdot(\rho R)}\right)$$

  The equivalent radius R, is equal to $R = (3V_{CV}/4\pi)^{1/3}$. In this formula the air density is used, equal to 1.3 kg/m^3 ([92], section 11.5). Therefore $\rho R = 0.8065 \times V_{CV}^{(1/3)}$.

  The mass absorption coefficient is obtained from the following correlation ([92], section 3.9, equation 3.83):

$$\left(\frac{\mu_{a,\beta}}{\rho}\right) = \frac{1.7}{E_\beta^{1.14}}$$

  In SPECTRA SI units are used and the mass absorption coefficient is expressed in m^2/kg rather than in cm^2/g as in [92]. Consequently the constant is 1.7 rather than 17.

- Radiation flux from deposited sources.

  The geometry considered is shown in Figure 12-118. The elementary flux is given by:

$$d\varphi_{dep} = \frac{S_A dA \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho r)}}{4\pi R^2}$$

  The total radiation flux from a surface with a uniform source density is equal to (see section 12.3.13.1):

$$\varphi_{dep} == 0.207 \frac{S}{V_{CV}^{2/3}} \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho R)}$$

  where A is the total area of a given structure and S is the total source from the surface, in (Bq), equal to: $S_A \cdot A$. In SPECTRA a number of 1-D and 2-D structure surfaces can be facing the same Control Volume.

- Total radiation flux

  As in case of γ radiation, two user-defined multipliers are introduced. These are the multiplier on the airborne source, $C_{\beta,1}$, and the multiplier on the deposited source, $C_{\beta,2}$. Default value of those multipliers is 1.0 (see Volume 2). The final formula for the γ-radiation flux calculation is:

$$\varphi_{\beta,i} = C_{\beta,1} \cdot \frac{\rho}{(\mu_{a,\beta}/\rho)} \cdot S_{i,CV} \cdot \left(1 - e^{-(\mu_{a,\beta}/\rho)\cdot(\rho R)}\right) +$$

$$+ \quad C_{\beta 2} \cdot \frac{0.207}{V_{CV}^{2/3}} \left(S_{i,CV-pool} + \sum_{JN \in CV} S_{i,JN-filter} + \sum_{SC \in CV} S_{i,SC} + \sum_{TC \in CV} S_{i,TC} \right) \cdot e^{-(\mu_{a,\beta}/\rho)\cdot(\rho R)}$$

12.3.13.3 Dose Equivalent Rate

The dose equivalent rates are calculated from:

$$H = D_\gamma \cdot Q_\gamma + D_\beta \cdot Q_\beta$$

H dose equivalent rate, (Sv/s) (or, if the dose rate is expressed in (rad/s), in (rem/s))
D_γ dose rate from γ radiation, (Gy/s) (or in (rad/s))
Q_γ quality factor of the γ radiation, (-)
D_β dose rate from β radiation, (Gy/s) (or in (rad/s))
Q_β quality factor of the β radiation, (-)

The quality factors, Q_γ and Q_β, are user-defined parameters, with default values of 1.0, based on [92], section 9.2, table 9.2).

13 MCCI

13.1 Introduction

Molten Core - Concrete Interactions (MCCI) is very complicated and involves a large number of different phenomena. The model implemented in the current version is preliminary and is based on a number of simplifications. The model is intended to obtain realistic estimation of the depth of concrete ablation and the amount of released gases. Currently the model is appropriate for LWR fuel. In the future it will be extended to be applicable to molten salt reactors.

Numerically, MCCI is a part of the TC Package; the MCCI equations are solved on the 2-D network of the 2D Solid Heat Conductor. However, since MCCI is a separate subject, and quite broad in itself, it is described in a separate chapter.

13.2 MCCI Phenomena

A very good description of MCCI phenomena may be found, among others, in the master's thesis of T. Sevón [226]. The short discussion of MCCI provided below is based on references [226] and [230].

In case of a hypothetical severe accident large amounts of molten corium may enter the reactor cavity after the reactor pressure vessel has failed. As a consequence, MCCI situations will be established during which some major hazards for the environment may be encountered:

- Because of the continuous release of decay heat in the corium there is a potential for a melt-through of the concrete foundation of the containment by ablation of the concrete, thus opening a downward pathway for radioactive fission products into the soil and groundwater located underneath.
- Concrete ablation generates gas release – especially the gases H_2, H_2O, CO, and CO_2 – into the containment atmosphere. The gas production contributes to the pressure increase in the containment and may lead to the formation of explosive gas mixtures. Both effects have impact on the boundary conditions for long-term leakage processes and may even lead directly or indirectly to an over-pressurization failure of the containment.

The core of a nuclear reactor consists of uranium dioxide in the fuel, zirconium in the fuel rod cladding and carbon steel and stainless steel in other structures. At high temperatures zirconium is oxidized by water vapor, so the main constituents of core melt, or corium, are UO_2, ZrO_2, Zr, Fe, Cr and Ni. The melting point of the pure oxides is around 2700 °C, while the metals melt at 1350 - 1900°C. Mixtures of different species do not have single melting points. Instead, they change from solid to liquid over a range of temperatures, between so-called solidus and liquidus temperatures. The density of corium is around 6000-7000 kg/m^3.

Molten core - concrete interaction (MCCI) is illustrated in Figure 13-1. The solid concrete and the molten corium pool may be separated by a thin layer of corium crust and molten concrete. The concrete melt rises upwards as "streamers" because it is less dense than the overlying core melt. Also the gas bubbles, rising from the decomposing concrete, cause mixing of the liquids. A layer of solid corium crust may also exist at the core-concrete interface. The crust is probably porous and permeable to gases from the concrete.

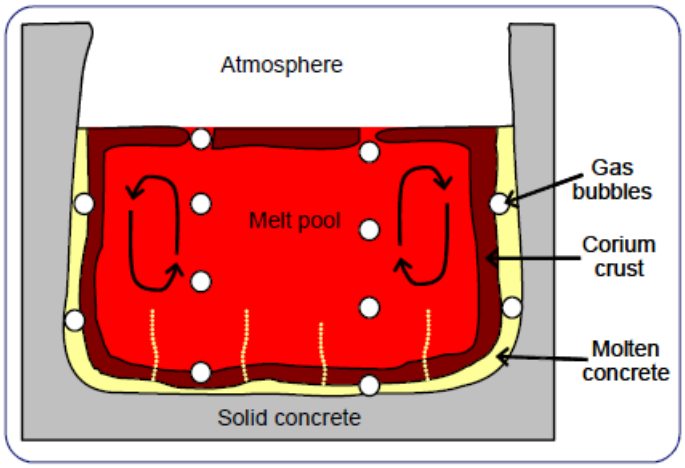


Figure 13-1 Illustration of MCCI phenomena [226]

The oxides in corium and concrete are miscible with each other, but the metallic species are immiscible with the oxides. Because the metals are lighter than the corium oxides, a metallic layer may be formed on the surface of the oxidic pool. When concrete oxides are added to the melt, its density decreases eventually below the density of the metals. After this, the metallic layer may relocate to the bottom of the pool. On the other hand, intense stirring of the pool by the rising gas bubbles may cause the metals and the oxides to be mixed with each other.

The rate of concrete ablation is controlled by the heat transfer from the melt to concrete. The core melt pool is stirred by the rising gas bubbles, which enhances the heat transfer. On the other hand, the possible corium crust at the interface inhibits heat transfer. Because the heat conductivity of concrete is very poor, almost all of the heat goes to the heat-up and melting of the surface of the concrete wall.

The viscosity of the melt pool also affects the heat transfer: Higher viscosity means lower heat transfer rate. Between the solidus and liquidus temperatures the viscosity of the mixture increases sharply with decreasing temperature. An additional complication comes from the fact that the melt composition and its physical properties change over time as more concrete is added to the melt. It is unknown whether the oxides with high melting points (UO_2 and ZrO_2) are segregated to the crust at the pool boundaries, or if they are dispersed as solid particles among the melt, which would increase the viscosity of the melt.

Some of the heat is transferred away from the surface of the pool by thermal radiation and convection. Usually there is air above the pool surface, but the surface can also be cooled by water. The surface temperature of the pool is also affected by the presence of a crust cover and the passing of gas bubbles through the surface.

The core melt is continuously heated by the radioactive decay of the fission products in the melt. Another heat source is the chemical reaction heat (section 13.6).

The model is described in section 13.3. The material properties are discussed in section 13.4. Concrete decomposition reactions are discussed in section 13.5. Chemical reactions are discussed in section 13.6.

13.3 Model

The MCCI model is built on the network of TC cells. If several TCs are defined in the input, MCCI is defined using TC with the lowest number (first on the list), which must be a vertical cylinder. Furthermore, the mesh must be uniform (i.e. all cells in given direction must have the same size), which is required by the numerical scheme of the Navier-Stokes equation.

MCCI is very complicated and involves a large number of different phenomena. The model implemented in the current version is preliminary and is based on a number of simplifications. The model is intended to obtain a realistic estimation of the dept of concrete ablation and the amount of released gases. Currently the model is appropriate for LWR fuel. In the future it will be extended to be applicable to molten salt reactors.

13.3.1 Model Versions

- Model 1: liquid flow calculated from Navier-Stokes equation

The governing equations are the continuity equation, the Navier-Stokes equation and energy conservation equation [232]:

$$\nabla \bar{u} = 0$$

$$\frac{d\rho\bar{u}}{dt} = -\nabla p + \mu\nabla^2\bar{u} + \rho\bar{g}$$

$$\rho c_p \frac{\partial T}{\partial t} = k\nabla^2 T + Q$$

Here ρ, t, u, p, μ, F, c_p, k, T, and Q are density, time, velocity vector, pressure, kinematic viscosity, external force (gravity), specific heat, thermal conductivity, temperature, and heat source (decay power), respectively. Currently the model based on Navier-Stokes equation is not yet available.

- Model 2: simplified model

Here no flow equation is solved, so the problem is becoming the same as the 2D conduction problem and a very similar equation is solved. In order to account for mixing of liquid material, an effective conductivity is defined; it is assumed that the effective conductivity in the two-phase region is inversely proportional to the effective viscosity in the two-phase region (section 13.4.4, Figure 13-4).

In both models, the specific heat in the phase change region is modified to account for the heat of phase change and concrete ablation, as:

$$\int c_p \, dT = \Delta H$$

Here ΔH is the phase change enthalpy (Table 13-1) or the ablation enthalpy (Table 13-2). The integral is between the solidus, T_{sol}, and the liquidus, T_{liq}, temperatures.

13.3.2 Mixing Model

A simple model is adopted for the mixing of corium with concrete. Decomposition of concrete results in a loss of certain amount of gases (section 13.5). The amount of mass that is lost in that way depends on the concrete composition (section 13.4.3). We denote $X = V_{deg}/V_{abl}$, where V_{deg} = degassed volume and V_{abl}, is the ablated volume. When a cell is fully ablated, a fraction of X is disappearing and is assumed to be replaced by corium. The user may affect this process by the following parameters:

- Radial ablation parameter (XRAPTC), default = V_{deg}/V_{abl},
- Axial ablation parameter (XZAPTC), default = V_{deg}/V_{abl},
- Maximum concrete dissolution in corium XDISTC, default = $1 - V_{deg}/V_{abl}$,

The fraction of concrete that is removed from a cell and replaced by corium is at least equal to V_{deg}/V_{abl}. It may be more if radial or axial parameters are defined as larger than this number. In such case, the additional concrete is assumed to be dissolved in the corium cells. If XRAPTC and XZAPTC are both equal to 0.0, there will be no dissolution of concrete in corium. The corium level will decrease as the corium sinks into the concrete cells. As the level decreases, the ablated concrete cells at the same level that are in liquid state are assumed to be removed and dissolved in corium. The user may speed up the ablation process by increasing the concrete decomposition in radial or axial direction. In such case there will be more concrete dissolved in the corium cells. Calculations will be stopped if the dissolved fraction exceeds XDISTC.

The gas that is created from concrete decomposition is assumed to bubble trough the corium and oxidize metals that may be present there - the reactions are described in section 13.6. The reactions proceed in a sequence: Zr is oxidized first, and Si, Cr, and Fe follow [226].

As a result of the oxidation process, the volume of corium increases somewhat so the decrease of level is a result of concrete volume decrease due to degassing and the corium volume increase due to oxidation. The user may also select an option where the corium level decrease is not calculated (IDEPTC =1) but this option will not conserve the corium mass.

In summary, the following main phenomena are occurring:

- Degassing of concrete that results in removal of certain volume of concrete and generation of gases such as H_2O (steam) and CO_2.
- The gas bubbles pass through the corium, oxidize metals that may be present there, to generate gases such as H_2 and CO.
- Due to volume change, the level of corium as well as the molten concrete decreases.

13.4 Material Properties

13.4.1 Corium Properties

The properties of corium and concrete are defined as follows.

- The material properties of solid state are defined in the usual way within the Material Property Package (Chapter 14). The user must define the material number for corium (ICMMTC) and concrete (ICCMTC) within the MCCI data. Only those materials may change phase. Any other material that may be present in the model will remain in solid state (and will not relocate).
- Properties of the liquid phase (as well as those related to the phase change) are defined within the MCCI input data. Default values of the constant (temperature-independent) properties are shown in Table 13-1. The temperature-dependent properties are discussed below.

Table 13-1 Property data for concrete and corium materials in liquid state

	Concrete	**UO_2**	**ZrO_2**	**Reference**
ρ (kg/m^3)	2200 2306	8860 10960	5150 5600	[226], sec. 3.5, p.33 [227], MP-RM
C_p (J/kg-K)	837.3	503.0	544.3	[227], MP-RM
k (W/m-K)	1.10	3.05	2.49	[226], sec. 3.5, p.35
μ (kg/m-s)	0.3 (LCS) 1.5 (L)	-	-	[226], sec. 3.5, p.33 [226], sec. 3.5, p.33
ΔH (J/kg)	-	2.74E5 2.18E5	7.07E5 2.60E5	[227], MP-RM [228]
T_m (K)	-	3113 3200	2990 3000	[227], MP-RM [228]

The temperature-dependent properties of corium and concrete are based on [228]. The functions are defined below.

Density. The density is defined as follows:

$$UO_2: \quad \rho = 11800 - 0.93 \times T$$
$$ZrO_2: \quad \rho = 8620 - 0.89 \times T$$

Here T is temperature (K). The resulting density is shown in Figure 13-2 (a). The user may put a multiplier on each of the above formulae.

Specific heat. The specific heat is defined as follows:

$$UO_2: \quad Cp = 390 - 0.90\times10^{-4} \times T$$
$$ZrO_2: \quad Cp = 940 - 3.36\times10^{-2} \times T$$

The resulting specific heat is shown in Figure 13-2 (b). The user may put a multiplier on each of the above formulae.

Thermal conductivity. The thermal conductivity is defined as follows:

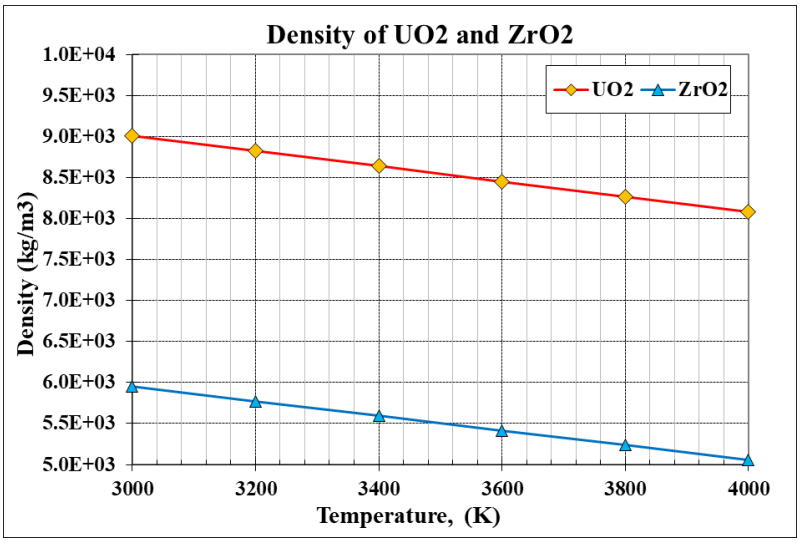
$$UO_2: \quad k = 1.64 + 4.74\times10^{-2} \times T \times \exp(-1.45\times10^{4}/T)$$
$$ZrO_2: \quad k = 2.11 + 4.46 \times T \times \exp(-2.75\times10^{4}/T)$$

The resulting thermal conductivity is shown in Figure 13-2 (c). The user may put a multiplier on each of the above formulae.

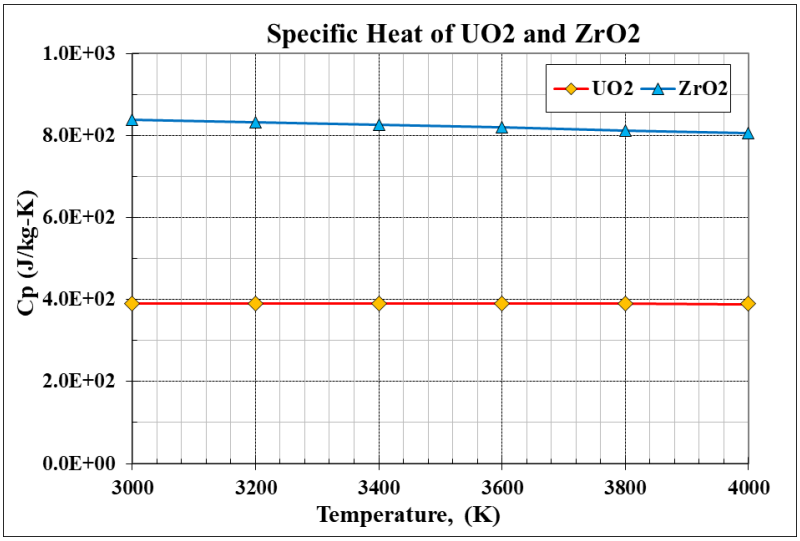
Dynamic viscosity. The dynamic viscosity is defined as follows:

$$UO_2: \quad \mu = 0.52 \times \exp(8.26\times10^{3}/T)$$
$$ZrO_2: \quad \mu = 0.32 \times \exp(8.79\times10^{3}/T)$$

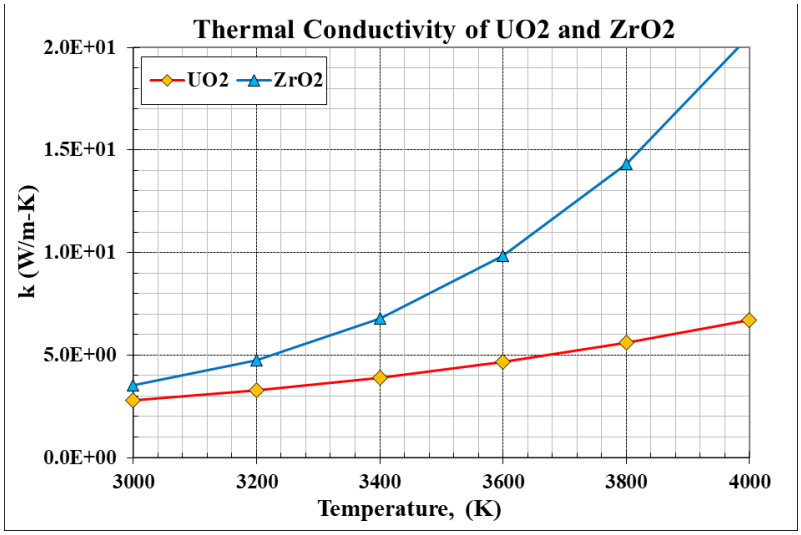
The resulting dynamic viscosity is shown in Figure 13-2 (d). The user may put a multiplier on each of the above formulae.



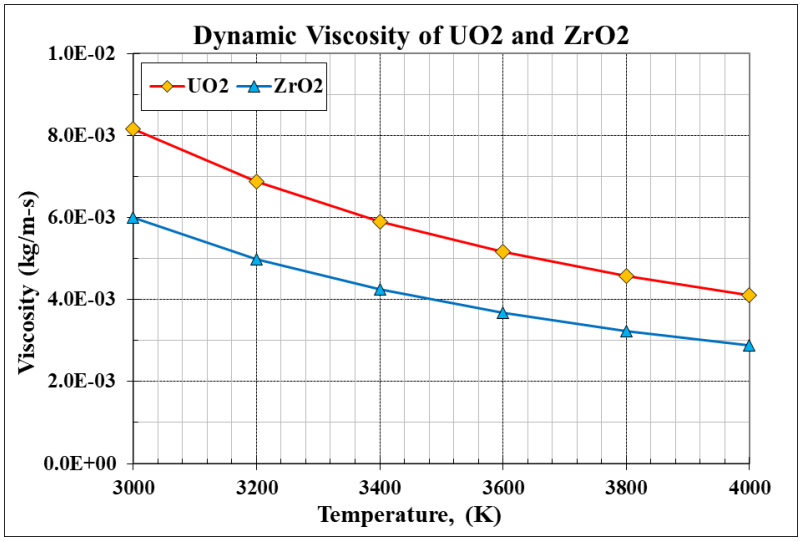
(a)



(b)



(b)



(d)

Figure 13-2 Properties of UO_2/ZrO_2 mixture

13.4.2 Properties of UO_2/ZrO_2 Mixture

The corium is assumed to be composed of UO_2, ZrO_2 and metals (Zr, Cr, Fe, Ni). The initial melt composition is defined by the user. The properties of corium are calculated based on [228]. The functions are defined below.

Melting temperature of UO_2/ZrO_2 mixture. The melting temperature of the mixture is approximated from the phase diagram [231] (Figure 13-3):

$$ZrO_2 < 0.6: \quad T_m = T_m(UO_2) - 438.33 \times X_{ZrO2} = 3113 - 438.33 \times X_{ZrO2}$$
$$ZrO_2 > 0.6: \quad T_m = T_m(ZrO_2) + 350 \times (X_{ZrO2} - 1) = 2990 + 350 \times (X_{ZrO2} - 1)$$

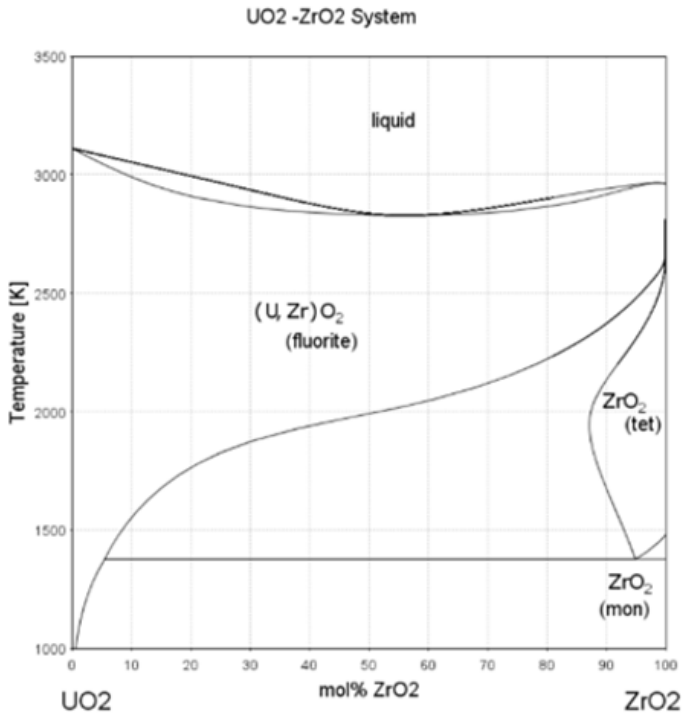


Figure 13-3 Phase diagram, UO_2/ZrO_2 system [231]

Heat of fusion of UO_2/ZrO_2 mixture. The heat of fusion, H (J/kg), is obtained from [228]:

$$H = (X_{UO2} \times M_{UO2} \times H_{ZrO2} + X_{ZrO2} \times M_{ZrO2} \times H_{ZrO2}) / (M_{UO2} + M_{ZrO2})$$

Density of UO_2/ZrO_2 mixture. The density, ρ (kg/m^3), is obtained from [228]:

$$\rho = (X_{UO2} \times M_{UO2} + X_{ZrO2} \times M_{ZrO2}) / (X_{UO2} \times M_{UO2} / \rho_{UO2} + X_{ZrO2} \times M_{ZrO2} / \rho_{ZrO2})$$

Specific heat of UO_2/ZrO_2 mixture. The specific heat, Cp (J/kg-K). is obtained from [228]:

$$Cp = (X_{UO2} \times M_{UO2} \times Cp_{UO2} + X_{ZrO2} \times M_{ZrO2} \times Cp_{ZrO2}) / (M_{UO2} + M_{ZrO2})$$

Thermal conductivity of UO_2/ZrO_2 mixture. The thermal conductivity, k (W/m-K), is obtained from [228]:

$$k = X_{UO2} \times k_{UO2} + X_{ZrO2} \times k_{ZrO2} - 0.72 \times X_{UO2} \times X_{ZrO2} \times | k_{UO2} - k_{ZrO2} |$$

Viscosity of UO_2/ZrO_2 mixture. The viscosity of mixture is obtained from [228]:

$$ln(\mu) = X_{UO2} \times \ln(\mu_{UO2}) + X_{ZrO2} \times \ln(\mu_{ZrO2})$$

13.4.3 Concrete Properties

The concrete composition is defined by the user (see Volume 2). The compositions of several common concrete types are built-in for user convenience. These are shown in Table 13-2

Table 13-2 Built-in concrete composition and properties.

N0.	Substance	Concrete composition, mass fractions (%)				
		Basaltic aggregate	Limestone c. sand 1	Limestone c. sand 2	Siliceous 1	Siliceous 2
1	CO_2	1.5	30.46	21.15	10.0	0
2	H_2O	5.68	4.46	4.7	3.78	4.0
3	K_2O	5.39	0.56	1.22	0.83	0
4	Na_2O	1.8	0.32	0.08	0.68	0
5	TiO_2	1.05	0.14	0.18	0.16	0
6	SiO_2	54.84	22.0	35.806	61.34	65.0
7	CaO	8.82	26.4	31.3	17.2	0
8	MgO	6.16	11.7	0.48	0.87	0
9	Al_2O_3	6.32	2.54	3.6	3.61	20.0
10	Fe_2O_3	6.26	1.42	1.44	1.53	0
11	Cr_2O_3	0	0	0.014	0	0
12	$CaCO_3$	0	0	0	0	3.0
13	$Ca(OH)_2$	0	0	0	0	8.0
14	MnO	0	0	0.03	0	0
15	SO_3	0	0	0	0	0
T(solidus) (K)		1350	1392	1420	1403	1350
T(liquidus) (K)		1650	1568	1670	1523	1650
T(ablation) (K)		1450	1500	1500	1450	1450
density ρ (kg/m^3)		2340	2340	2340	2340	2400
Abl. enth. ΔH (J/kg)		2.8×10^6	2.4×10^6	2.4×10^6	1.95×10^6	1.95×10^6
X(ablation)		0.333	0.614	0.320	0.392	0.333
Source		[227] (CAV-UG)	[225]	[227] (CAV-UG)	[225]	[227] (CAV-UG)

13.4.4 Properties of Corium/Concrete Mixture

In cases when corium and concrete are present in the same cell, the properties of a mixture need to be known. The properties of corium/concrete mixture are calculated in a similar way as shown in section 13.4.2. The reference is [228]. The functions are defined below.

Density of corium/concrete mixture. The density of mixture is obtained from the following formula. The subscripts "cor" and "con" are used for the corium and the concrete material, respectively.

$$\rho = (X_{cor} \times M_{cor} + X_{con} \times M_{con}) / (X_{cor} \times M_{cor} / \rho_{cor} + X_{con} \times M_{con} / \rho_{con})$$

Specific heat of corium/concrete mixture. The specific heat of the mixture is obtained from:

$$Cp = (X_{cor} \times M_{cor} \times Cp_{cor} + X_{con} \times M_{con} \times Cp_{con}) / (M_{cor} + M_{con})$$

Thermal conductivity of corium/concrete mixture. The thermal conductivity of the mixture is obtained from:

$$k = X_{cor} \times k_{cor} + X_{con} \times k_{con}$$

Viscosity of corium/concrete mixture. The viscosity of the mixture is obtained from:

$$ln(\mu) = X_{cor} \times \ln(\mu_{cor}) + X_{con} \times \ln(\mu_{con})$$

The viscosity obtained as shown above is valid for pure liquid phase. In the case of a solid/liquid mixture, the viscosity of such mixture is significantly larger. Two models are available to calculate the two-phase viscosity: Ramacciotti [229] and Kunitz (applied in CORCON and MELCOR [227]).

- Ramacciotti correlation for two-phase viscosity

$$\mu_2 = \mu_1 \exp(2.5 \times CRAMTC \times \varphi)$$

- Kunitz correlation for two-phase viscosity

$$\mu_2 = \mu_1 (1+0.5 \times \varphi) / (1+\varphi)^4$$

Here μ_1 is the liquid viscosity, μ_2 is the two-phase viscosity and φ is the solid fraction in the mixture. CRAMTC is a user-defined coefficient with a value between 4 and 8 (see Volume 2).

Both correlations are compared in Figure 13-4. Ramacciotti correlation is shown for CRAMTC=4. Since the Kunitz correlation gives an infinite value for $\varphi = 1.0$, a limit of is imposed on the solid fraction to prevent numerical overflow.

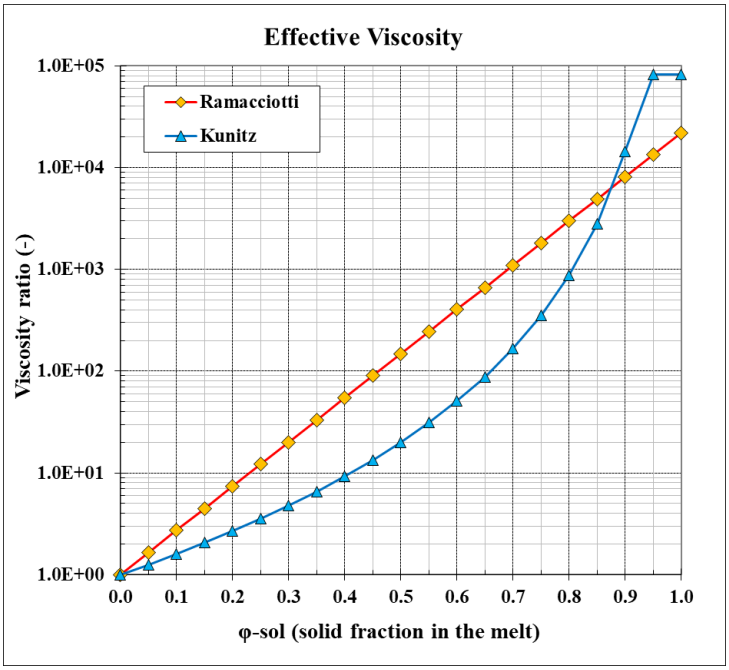


Figure 13-4 Comparison of Ramacciotti and Kunitz correlations.

Heat of fusion of corium/concrete mixture. The heat of fusion of the mixture is obtained from:

$$H = (X_{cor} \times M_{cor} \times H_{cor} + X_{con} \times M_{con} \times H_{con}) / (M_{cor} + M_{con})$$

Melting temperature of corium/concrete mixture. The melting temperature of mixture is defined using tabulated data of liquidus and solidus lines. Data for several concrete types, obtained from [226], is shown in Figure 13-5. The figure was digitized and the appropriate data files are available (see Volume 2).

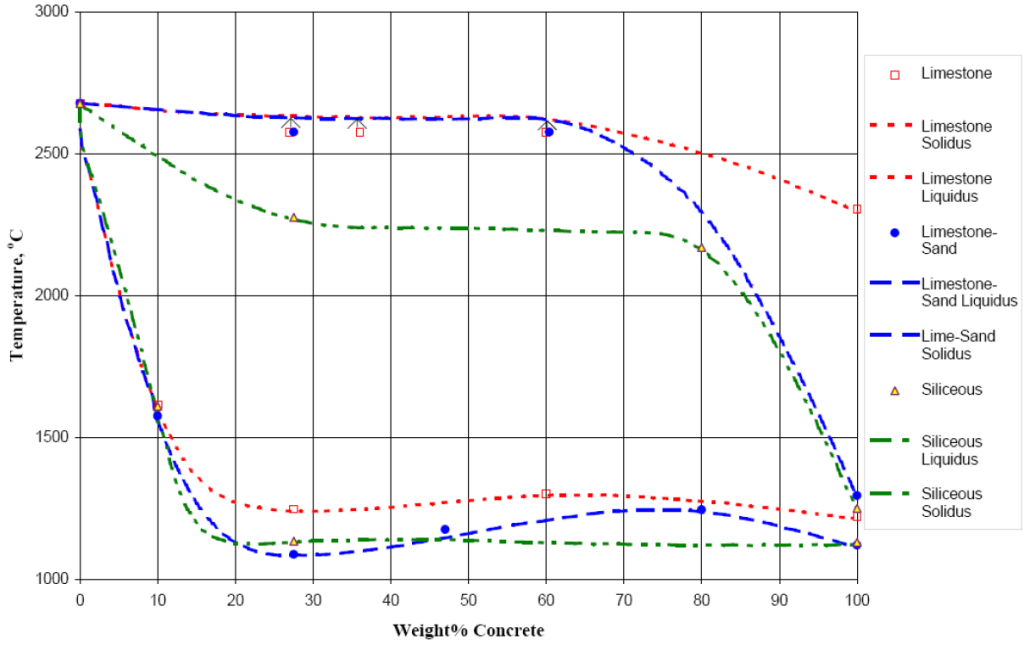


Figure 13-5 Solidus and liquidus temperature for corium/concrete mixtures [226]

13.5 Decomposition of Concrete

The model of concrete decomposition is based on [226] (section 3.2.2). The following reactions are assumed to take place during the concrete ablation.

Reactions leading to generation of steam:

- Loss of evaporable water:
  $H_2O(l) + 2.258\times10^6$ J/kg(H_2O) → $H_2O(g)$

- Dehydration of hydrates:
  $3CaO\cdot2SiO_2\cdot3H_2O + 0.375\times10^6$ J/kg(hydrate) → $2CaO\cdot SiO_2 + CaO\cdot SiO_2 + 3\ H_2O(g)$

- Dehydration of calcium hydroxide
  $Ca(OH)_2 + 1.340\times10^6$ J/kg($Ca(OH)_2$) → $CaO + H_2O(g)$

Reactions leading to generation of CO_2:

- Loss of CO_2 present in the concrete mixture
  $CO_2 \rightarrow CO_2(g)$

- Decomposition of calcium carbonate:
  $CaCO_3 + 1.637\times10^6$ J/kg($CaCO_3$) $\rightarrow CaO + CO_2(g)$

Siliceous concrete releases mainly water vapor, while calcareous concrete releases also substantial amounts of carbon dioxide.

It should be noted that the following assumptions/simplifications were made in the current model version:

- The energy effect from dehydration of hydrates is given in [226] as 250 - 500 kJ/kg. Here, the average value is taken.
- Other reactions listed in [226] (melting of quartz, decomposition of hematite into magnetite, melting of magnetite) do not result in producing gases and are currently not modeled.
- It is assumed that the gases are released during ablation; the release of material present in a given cell is completed when the cell is fully ablated.

These assumptions should be checked and, if needed, improved in the future.

13.6 Chemical Reactions

The chemical reactions are based on [226] (section 3.1). The following reactions are assumed to take place when the gas bubbles are passing through the corium.

Reactions leading to generation of H_2:

- $Zr + 2\ H_2O \rightarrow ZrO_2 + 2\ H_2 + 6.3\times10^6$ J/kg_{Zr}
- $2\ Cr + 3\ H_2O \rightarrow Cr_2O_3 + 3\ H_2 + 3.6\times10^6$ J/kg_{Cr}
- $Fe + H_2O + 3.0\times10^3$ J/$kg_{Fe} \rightarrow FeO + H_2$

Reactions leading to generation of CO:

- $Zr + 2\ CO_2 \rightarrow ZrO_2 + 2\ CO + 5.7\times10^6$ J/kg_{Zr}
- $2\ Cr + 3\ CO_2 \rightarrow Cr_2O_3 + 3\ CO + 2.8\times10^6$ J/kg_{Cr}
- $Fe + CO_2 + 4.8\times10^5$ J/$kg_{Fe} \rightarrow FeO + CO$

Equilibrium constants for the reactions indicate that Zr is oxidized first, and Si, Cr, and Fe follow in this order [226].

In the current model, the equilibrium constants are not used. There is a simple model instead, with a user-defined coefficient IRSQTC (Volume 2). The ratio of speed of different reactions is equal to $10^{(1-IRSQTC)}$. If IRSQTC =1 all reactions proceed simultaneously. The default value is IRSQTC = 3, thus the Zr reactions proceed 100 faster than Cr reactions, which in turn proceed 100 times faster than the Fe reactions.

Furthermore, in the current model the reduction of SiO_2 and Fe_2O_3 by Zr (see [226] section 3.1) is not taken into account. These reactions are not important if the content of Zr in the melt is low. This should be improved in the future.

14 Material Property Package

14.1 Properties of Solid Materials

The Material Property Package allows the user to define properties of solid materials. The properties are used by the 1-D and the 2-D Solid Heat Conduction Packages (chapters 5 and 6) for heat conduction calculations (sections 5.2 6.2) and by the Radioactive Particle Transport Package (chapter 12) to calculate diffusion of fission products inside solid materials (section 12.3.5.6).

The following material properties are specified as functions of temperature for each material:

- Thermal conductivity, k, W/m^2-K
- Specific heat, c_p, J/kg-K
- Density, ρ, kg/m^3
- Diffusion coefficient, D_{CS}, m^2/s

The first three are used for heat conduction calculations and must be specified for each material being used. The last on is used for diffusion calculation and need not to be specified. If the diffusion coefficient is not specified, the fission products (if present) will not diffuse within this material.

In SPECTRA density as other material properties may be a function of temperature, which is usually done by computer programs, for example RELAP, MELCOR, etc. If during calculations the material density changes, the total mass of material will change. This is because the volume of structures (SC or TC) remains constant during calculations. In reality a change of material density is accompanied by a change of volume, such that the total mass remains the same. Therefore it is recommended to ignore the the changes of density with temperature and use constant density.

The diffusion coefficients are used only if sorption of fission products is calculated. In such case they are used to calculate diffusion of the sorbed fission products into the SC or TC materials. The diffusion coefficient is calculated from:

$$D_{CS} = D_0 \cdot \exp\left(-\frac{A_D}{T}\right)$$

Lower and upper limits, defined by the user, are applied to the value obtained from the above formula:

$$D_{\min} \le D_{CS} \le= D_{\max}$$

14.2 Materials Composed of Several Materials

As described in [201], the overall thermal conductivity of a medium composed of several materials depends in a complex fashion on the geometry of the medium. If the heat conduction occurs in parallel (Figure 14-1, left), then the overall conductivity is the weighted arithmetic mean of the individual conductivities:

$$\bar{k} = \sum_i x_i \cdot k_i$$

On the other hand, if the structure and orientation of the medium is such that the heat conduction takes place in series, with all of the heat flux passing through all layers (Figure 14-1, middle), then the overall thermal resistance is the weighted average of individual resistances.

$$\overline{1/k} = \sum_i x_i / k_i$$

In general, the above two equations will provide upper and lower bounds, respectively, on the actual overall conductivity [201].

In SPECTRA the thermal conductivity of such material may be defined in one of the two ways described above. The heat capacity is obtained as a sum of volumetric heat capacities of all individual materials, which gives:

$$\overline{V \cdot \rho \cdot c_p} = \sum_i V_i \cdot \rho_i \cdot c_{p,i} = V \cdot \sum_i x_i \cdot \rho_i \cdot c_{p,i}$$

which leads to:

$$\overline{\rho \cdot c_p} = \sum_i x_i \cdot \rho_i \cdot c_{p,i}$$

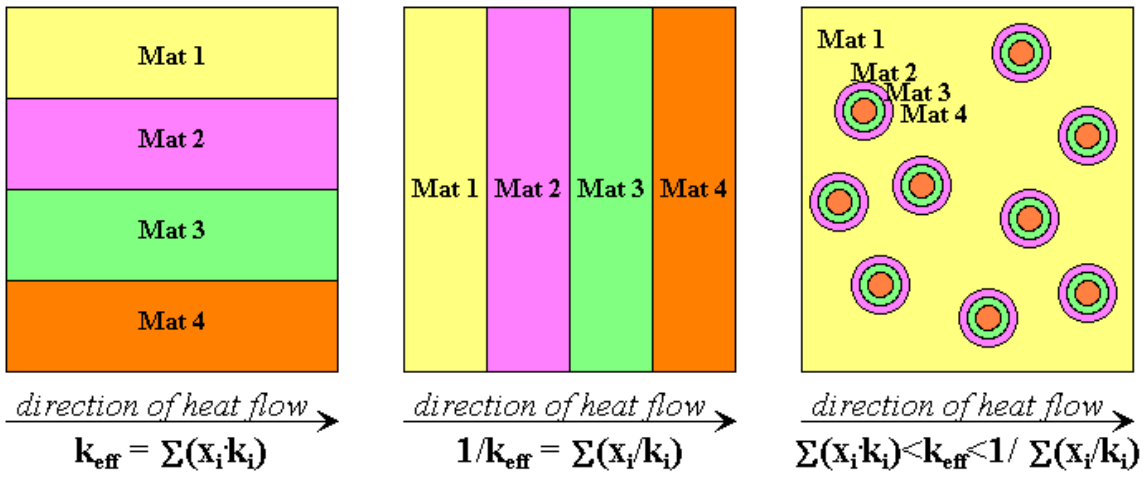


Figure 14-1 Examples of materials being a composition of different materials

Summarizing, the user may specify a material that is composed from several (up to 10) different materials. The average thermophysical properties of such material are obtained from:

$$\overline{\rho \cdot c_p} = \sum_i x_i \cdot \rho_i \cdot c_{p,i} \qquad \bar{k} = \sum_i x_i \cdot k_i \qquad or \qquad \overline{1/k} = \sum_i x_i / k_i$$

x_i = volumetric fraction of the material *i* in the mixture, (-)
ρ_i = density of the material *i*, (kg/m^3)
$c_{p,i}$ = specific heat of the material *i*, (J/kg-K)
k_i = conductivity of the material *i*, (W/m^2-K)

The user has to decide if the first or the second formulation should be used to compute average conductivity. The first formulation always gives higher values of *k*. The second formulation always gives lower values of *k*, therefore for conservatism it is selected as a default. Additionally, the user may an option that interpolates between the first and the second formula, which leads an intermediate value of thermal conductivity (see Volume 2).

An example of the situation where the model is useful, is the fuel of high temperature reactor, consisting of graphite and TRISO particles, which in turn consist of fuel kernel and several coating layers.

15 Tabular Function Package

15.1 Introduction

The Tabular and Control Function Packages allow the user to define functions of variables in all SPECTRA data blocks, and make them available to various packages. Those packages can therefore be considered as utility packages for the user. The modelling approach is quite general, and allows a variety of applications. For example, a chemical reaction can be modelled (see hydrogen recombiner example, section 16.5), a differential equation set can be set up and solved (Solver "stiff" equation set test, section 16.5), etc.

The principal difference between the Tabular and the Control Functions is that the former is a function of time, while the latter may be a function of any SPECTRA variable. Consequently the Tabular Functions need not to be included in the main iteration loop to achieve implicit solution. Their end-of-time step values are easily available by a straightforward interpolation. The Control Functions are meanwhile included in the iteration loop, since they depend on the variables from the program data blocks, whose values at the end of time step have to be determined iteratively.

The Tabular and Control Functions may be used to define certain quantities in the other packages, which may be function of time (through Tabular Functions), or functions of the current conditions (through Control Functions) in the system being modeled. In particular, the Tabular or the Control Functions may be applied to define the following quantities in various SPECTRA Packages:

- Control Volume Package
  - To define mass sources for Control Volumes (see Volume 2, record 13YXXX).
  - To define energy sources for Control Volumes (record 14YXXX).
  - To activate stratification models, and to define stratifications if the best estimate stratification model is not desired (record 100XXX).
  - To define conditions like temperature, pressure, etc. in those Control Volumes in which user-defined conditions are applied (record 125XXX).

- Junction Package
  - To define flows in those Junctions for which user defined flows are applied (record 230XXX).
  - To define pump/compressor/turbine speed, efficiency, power. To control synchronization of a turbo-generator (or an engine driving a pump/compressor) with the electrical grid (records 231XXX, 233XXX).
  - To define the fraction of a Junction area which is open in those Junctions for which the valve model is used (record 220XXX).
  - To define opening pressure difference for a check valve (record 220XXX).
  - To define flow composition parameters for those junctions in which uniform gas flow is not desired (record 240XXX).

- 1-D and 2-D Solid Heat Conductor Packages
  - To define an internal power source for 1-D and 2-D Solid Heat Conductors (records 300XXX, 400XXX).
  - To define a convective boundary condition for those 1-D and 2-D Solid Heat Conductors for which the standard heat transfer model is not desired (records 321XXX, 322XXX, 421XXX).
  - To define non-convective boundary heat fluxes for those 1-D and 2-D Solid Heat Conductors which do not use the thermal radiation model (records 321XXX, 322XXX, 421XXX).

- Thermal Radiation Package
  - To define wall surface emissivity for those surfaces of 1-D or 2-D Solid Heat Conductors which participate in radiative heat transfer (record 510XXX). If a tabular function is used to define the wall surface emissivity, then the argument for this function will be the wall surface temperature. The emissivity will therefore not be equal to the value of tabular function, as printed by the Tabular Function Package, since the Tabular Function Package calculates the values of Tabular Function as a function of time.

- Reactor Kinetics Package
  - To define control rod reactivity (record 760000).
  - To define external neutron sources (record 755000).
  - To determine fuel addition or removal from the core (record 747000).
  - To define the temperature of thermal neutrons (record 750000).

- Radioactive Particle Transport Package
  - To define external sources of aerosols (record 175XXX).
  - To define filter efficiency if a mechanistic filter model is not used (records 275XXX, 295XXX).
  - To define pool scrubbing efficiency if a mechanistic model is not used (record 276XXX, 296XXX).
  - To define external sources of isotopes for a Control Volume (record 192XXX).

- Hydrogen Burn Package
  - To define igniter temperature (record 180XXX).

- Control Function Package
  - As an argument for a Control Function. Any tabular or control function may be used as an argument for another Control Function (record 710XXX).

- Solver Package
  - To deactivate the thermal hydraulic packages, CV, JN, SC, TC (record 900XXX). Some or all of these packages may be deactivated to speed up calculations after a stationary state has been reached. This may be done for example to perform a long term fuel burn-up calculation, a long term dust deposition calculation, etc.; in short when the calculated period is of order of years and a constant (for example nominal) conditions are used. In practice deactivation of all thermal-hydraulic packages may lead to a gain on CPU time as much as a factor of 1000 or more.

15.2 Types for Tabular Functions

Tabular Functions are defined as pairs of variables: (time; value of TF). The time argument must be in ascending order. Each tabular function may consist of up to 10,000 data points. During calculations the actual value of a tabular function is obtained by interpolation of the tabulated data points. Two types of interpolation are available:

- Linear interpolation (see section 17.1)
- Third order (cubic) interpolation (see section 17.1)

With linear interpolation the obtained function is continuous, but it has discontinuous derivative. In case of cubic interpolation both function and its first derivative are continuous (Figure 15-1).

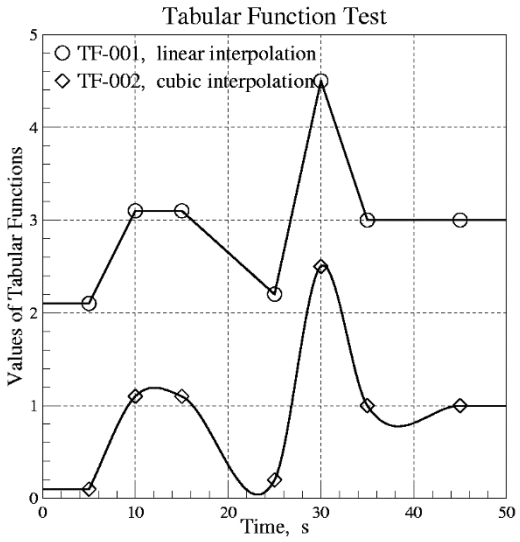


Figure 15-1 Types of Tabular Functions

Beyond the range defined by the data points, the end point values are kept. This approach is consider as safer than extrapolation for general application. If the user wants to extrapolate his data points, he can always do so by including an additional data point, sufficiently far from the normal data range.

The cubic interpolation ensures continuity of the first derivative in all range except for the boundary (first and last) data points. In those points the derivative "jumps" to zero, since flat lines are always used outside the tabulated data points. The discontinuity was programmed on purpose, because this formulation is considered safer for practical use. Furthermore the cubic interpolation subroutine is used in SPECTRA, among others, for the calculation of water properties. In this case the applied formulation allows to obtain correct behavior of the interpolated functions in the region close to the freezing point. The fact that the functions won't be smooth if the temperature drops below the freezing point has no practical meaning.

If the user wishes to avoid this discontinuity in his Tabular Function, he can do that by defining two points with the same value near the boundary. In the example shown in Figure 15-1 this was done near the right boundary. Two points with the same value (TF = 1.0) are specified at 35.0 and 45.0. This assures continuity of the TF derivative at the right boundary point 45.0.

The values of Tabular Functions are calculated at every time step by the Tabular Function Package. The argument used by Tabular Function Package to calculate the values of Tabular Functions is always the problem time, although Tabular Functions may also be calculated with a different argument. This however may only be done using a Control Function. The Control Function Package offers a more general type of a Tabular Function (see section 16).

15.3 Interactive Tabular Functions

Interactive Tabular Functions are useful in the simulation mode. At any time during a calculation the TF value may be modified in the TF Data file, *.TFD. The value that needs to be specified in the *.TFD file is simply the TF number and the value that it should take. Therefore, to set the value of TF-120 to 521.0 one needs to type (or the simulation tool needs to send) the following line in the *.TFD file:

120 521.0

The data present in the *.TFD file may have two formats:

- Arbitrary number of pairs, for example:
  120 1.0 200 1.5 250 –5.0
  The above record will set (immediately as it appears in the *.TFD file) the following values: TF-120=1.0, TF-200=1.5, TF-250=–5.0
- Three input parameters, single TF definition with TIMEON. For example:
  120 1.0 1000.0
  The above record will set TF-120=1.0 at the time = TIMEOF =1000.0 s (or immediately if the current time is larger than 1000.0 s.
  Another example. The following records:
  120 1.0 1000.0
  120 2.0 2000.0
  120 5.0 3000.0
  will set the value of TF-120 to 1.0 at t = 1000.0 s, to 2.0 at t = 2000.0 s, and finally to 5.0 at t = 5000.0 s.

The interactive function may perform two actions:

- Interactive action, when a prescribed value is present in the *.TFD file
- Interpolating action, when there is no prescribed value present in the *.TFD file.

The performance of such functions is shortly explained below.

If there is no data for the TF in the *.TFD file, then the value of the function will be determined by the table present in the input deck. If the value is present in the *.TFD file, then the value of TF will change, the rate of change being limited by the appropriate limits for this TF (see section 15.4), until it reaches the value specified in the *.TFD file, provided that it is not outside the limits given by the

appropriate input limits (section 15.4) for this TF. The TF will stay at that value for as long as it is present in the *.TFD file. If the value in the *.TFD value should change, the TF will start changing to reach the new value, again respecting the limits. If the value disappears from the *.TFD file, then the TF will start changing to reach the value prescribed for it by the tabulated data pairs, again respecting the limits appropriate for this TF.

Both interpolation types, linear and cubic (see section 15.2) may be used for an interactive function. Of course the interpolation type affects the results only when the TF is using the tabulated data (interpolating action), and not a value from the *.TFD file (interactive action). In other words, the interpolation type has an influence on the TF value only when the *.TFD file does not contain any value of this TF.

15.4 Limits, Scaling Factors, and Additive Constants

The user may impose limits on each Tabular Function, with respect to the value as well as the rate of change. The limits are:

- Value:

$$TF_{\min} \le TF \le TF_{\max}$$

- Rate of change

$$-\left.\frac{d(TF)}{dt}\right|_{\min} \le \frac{d(TF)}{dt} \le \left.\frac{d(TF)}{dt}\right|_{\max}$$

TF_{min}	Minimum value of TF (default value of -10^{99})
TF_{max}	Maximum value of TF (default value of $+10^{99}$)
$d(TF)/dt\,\big\vert_{min}$	Maximum rate of change for a decreasing value of TF (default value of 10^{99})
$d(TF)/dt\,\big\vert_{max}$	Maximum rate of change for an increasing value of TF (default value of 10^{99})

The default values result in practically no limits on the TF. In order to apply the limits, the user must enter appropriate values in the input deck.

Limits are particularly useful for interactive functions, i.e functions that receive their values from a simulation program, or through a TF Data file (*.TFD, see section 15.3) rather than from the tabulated data.

In case of tabulated values the limits are not so important because the user may already tabulate the values according to all desired limits. Even then, it is sometimes more convenient to use the limits to make sure that the tabulated values will always stay within a desired range.

For each Tabular Function a scaling factor and an additive constants are used. The value of a function is calculated from:

$$TF(t) = f(t) \cdot S + A$$

$f(t)$	value obtained from the tabulated data by means of a linear or a cubic interpolation (as described in section 15.2), or read from the *.TFD file (as described in section 15.3).
S	scaling factor, with the default value of 1.0
A	additive constant, with the default value of 0.0

The scaling factor and additive constant are applied before the limits are applied. The limits are applied to the final TF value and not the tabulated data points. For example, assume the following definition of TF-101:

The tabulated values are between –5.0 and –1.0; the scaling factor is –0.1; the limits are (0.0, 1.0).

- Tabulated values:

$$f(t) = \begin{cases} -1.0 & for \quad t = 0.0 \\ -5.0 & for \quad t = 1.0 \end{cases}$$

- Scaling factor, additive constant:

$$S = -0.1$$
$$A = 0.0$$

- Limits:

$$TF_{\min} = 0.0$$
$$TF_{\max} = 1.0$$

The SPECTRA input defining such function is shown below.

```
605101   Tabular Function TF-101
601001  -0.1    0.0     *  Scaling factor, Additive constant
602101   0.0    1.0     *  Limits
*
*       Time    Value
600101   0.0    -1.00   *
600101   1.0    -5.00   *
```

With the scaling factor applied to the tabulated values, the value of the function is between 0.1 and 0.5. This is within the acceptable limits and therefore the limits are not applied.

16 Control Function Package

16.1 Introduction

The Tabular and Control Function Packages allow the user to define functions of SPECTRA variables, and make them available to various packages. The packages can therefore be considered as a utility packages for the user. The modelling approach is quite general, and allows a variety of applications. For example, a chemical reaction can be modelled (see hydrogen recombiner example, section 16.5), a differential equation set can be set up and solved (Solver "stiff" equation set test, 16.5), etc.

The principal difference between the Tabular and the Control Functions is that the former is a function of time, while the latter may be a function of any SPECTRA variable. Consequently the Tabular Functions need not to be included in the main iteration loop to achieve implicit solution. Their end-of-time step values are easily available by a straightforward interpolation. The Control Functions are meanwhile included in the iteration loop, since they depend on the variables from the program data blocks, whose values at the end of time step have to be determined iteratively.

The Tabular and Control Functions may be used to define certain quantities in the other packages, which may be function of time (through Tabular Functions), or functions of the current conditions (through Control Functions) in the system being modeled.

A list of quantities that may be defined by a Tabular or a Control Function is presented in section 15.1. On top of the quantities listed in section 15.1, there are several quantities that may be defined using a Control Function but not a Tabular Function. Those are listed below.

- Junction Package
    - Control Functions may be used to define local loss factors, *K*, for forward and reverse flows, in those Junctions for which the *K* factors are not constant (Volume 2, input record 210XXX).

- Radioactive Particle Transport Package
    - Control Functions may be used to define release of fission products, if none of the built-in release models is desired (record 885XXY).
    - Control Functions may be used to define adsorption rate for those fission product vapors for which none of the built-in adsorption models is desired (record 895YXX).

16.2 Types for Control Functions

The list of all CF is given in Table 16-1. The control functions are divided into three groups. The first group contains some general functions like addition, multiplication etc. The second group contains FORTRAN intrinsic functions. The last group contains extensions to intrinsic functions. Control functions from all three groups are shortly discussed below.

16.2.1 CF Group 1

- **Add/Subtract Function (1, 01)**

The add/subtract function allows to calculate the sum of several arguments. The arguments may be taken with a plus sign (added) or minus sign (subtracted) to obtain the value of CF.

The following example converts the atmosphere temperature in CV-123 into degrees Celsius, using the add/subtract function, CF-101. The first argument is the temperature in CV-123. The second argument is a constant value of –273.15. A constant value is achieved by using a dummy argument (in this case a Solver time step) which is multiplied by zero (a very small number is entered for the multiplicative factor; if zero was entered, the default multiplicator of 1.0 would be assumed by SPECTRA - see Volume 2) and an additive constant of –273.15.

```
705101 Conversion of temperature into degree Celsius
*      Group   Number        Fact.   Const.
700101   1        1           1.0     0.0     *  Type    : Add/subtract
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.    Const.
710101       1  123  14  1   1.0        0.0   *  CV-123-Temp-atms
710101       9    0   1  0   1.0e-90 -273.15  *  -273.15
```

- **Multiply/Divide Function (1, 02)**

The multiply/divide function allows to calculate a product of several arguments. Each argument may be used as either a multiplier or a dividing argument to obtain the value of CF. For all dividing arguments the value of the argument must not be zero. If at any time during the calculations the value is zero, then an appropriate error message is written to the diagnostics file, and the calculations are stopped.

The following example calculates the ratio of the steam partial pressure to the total pressure in CV-123, using the multiply/divide function, CF-102. The first argument is the steam pressure in CV-123. The second argument is the total pressure in CV-123. The first pointer of the second argument is negative, which indicates that this is a “divide” argument - see Volume 2.

```
705102 Steam pressure / total pressure
*      Group   Number        Fact.   Const.
700102   1        2           1.0     0.0     *  Type    : Multiply/divide
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.    Const.
710102       1  123  36  3   1.0        0.0   *  CV-123-PPag-H2O_
710102      -1  123  13  1   1.0        0.0   *  CV-123-Pres-atms
```

Table 16-1 Types of Control Functions.

Group and No.		Function type	Number of arguments	FORTRAN definition
IGRPCF	INUMCF			
1	01	Add / subtract	≤ 100	f = a_1 +(-) a_2 +(-)...+(-) a_n
	02	Multiply / divide	≤ 100	f = a_1 ·(/) a_2 ·(/)...·(/) a_n
	03	Power function	2	f = a_1 ** a_2 if $a_1 > 0$ f = -|a_1| ** a_2 if $a_1 < 0$

	04	Selected argument	4	IF (a_1 .LT. a_2) THEN
				f = a_3
				ELSE
				f = a_4
				ENDIF

	05	General tabular function	≤ 100	(see example in section 16.2.1)
	06	Hysteresis	3	(see example in section 16.2.1)
	07	Derivative	1	f = (a_1 - a^0_1) / Δt
	08	Integral	1	f = f^0 + a_1·Δt
	09	Random number	0	f = rnd() · S(a_1) + A(a_1)
2	01	Absolute value	1	f = DABS(a_1)
	02	Minimum value	≤ 100	f = DMIN1(a_1, a_2,... a_n)
	03	Maximum value	≤ 100	f = DMAX1(a_1, a_2,... a_n)
	04	Square root	1	f = DSQRT(a_1)
	05	Exponent	1	f = DEXP(a_1)
	06	Natural logarithm	1	f = DLOG(a_1)
	07	Decimal logarithm	1	f = DLOG10(a_1)
	08	Sine	1	f = DSIN(a_1)
	09	Cosine	1	f = DCOS(a_1)
	10	Tangent	1	f = DTAN(a_1)
	11	Arc sine	1	f = DASIN(a_1)
	12	Arc cosine	1	f = DACOS(a_1)
	13	Arc tangent	1	f = DATAN(a_1)
	14	Hyperbolic sine	1	f = DSINH(a_1)
	15	Hyperbolic cosine	1	f = DCOSH(a_1)
	16	Hyperbolic tangent	1	f = DTANH(a_1)
3	01	Error function	1	f = DERF(a_1)
	02	Complementary error f.	1	f = DERFC(a_1)
	03	Gamma function	1	f = DGAMMA(a_1)
	04	Log gamma function	1	f = DLGAMA(a_1)
	05	Reactor Kinetics	1	see description below

- **Power Function (1, 03)**

The power function calculates the value of the first argument raised to the power equal to the second argument:

$$f(x_1, x_2) = x_1^{x_2}$$

If x_1 is equal to zero then: f=1.0 when x_2=0.0, and f=0.0 when x_2>0.0. The following restrictions are imposed on the arguments of this control function: The first argument must not be negative. If the first argument is equal to zero then the second argument must not be negative. Additionally the following limit is imposed on the second argument: $-99.0 < x_2 < 99.0$.

The following example calculates the square root of the gas velocity in JN-234. The first argument is the gas velocity in JN-234. The second argument is a constant value of 0.5. This is achieved using a dummy argument and an additive constant of 0.5, similarly as shown in case of the add/subtract function, above.

```
705103 Square root of gas velocity
*      Group   Number        Fact.   Const.
700103   1        3           1.0     0.0     *  Type    : Power
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.    Const.
710103       2  234  1   1   1.0       0.0    *  JN-234-Velo-atms
710103       9   0   1   0   1.0e-90    0.5    *  = 0.5
```

- **Selected Argument (1, 04)**

This is the FORTRAN IF-THEN-ELSE function. When the value of the first argument, x_1, is smaller than the value of the second argument, x_2, then CF value is set to the value of the third argument, x_3, otherwise it is set to the value of fourth argument, x_4.

$$f(x_1, x_2, x_3, x_4) = \begin{cases} x_3 & if \quad x_1 < x_2 \\ x_4 & if \quad x_1 \geq x_2 \end{cases}$$

The following example calculates a scram signal based on the reactor fission power. It compares the reactor fission power with the maximum power (assumed equal 1000 MW = 10^9 W). If the power is below the maximum power, the value of function is zero. If the reactor power is equal to or larger than the maximum power, the value of function is set to 1.0 (scram signal is activated). The first argument is the reactor fission power. The second argument a constant value of 1000 (maximum power). The third and fourth arguments are zero and one respectively.

```
705104 Scram signal based on the reactor fission power
*      Group   Number        Fact.   Const.
700104   1        4           1.0     0.0     *  Type    : IF-THEN-ELSE
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.    Const.
710104       7    0  18   0    1.0     0.0     *  IF ( RK-000-Qfis-000
710104       9    0   2   0    1.0E-90 1.0E+9  *                       < 1.0E9 )
710104       9    0   2   0    1.0E-90 0.0     *  THEN signal = 0.0
710104       9    0   2   0    1.0E-90 1.0     *  ELSE signal = 1.0
```

The next example illustrates one special case, when a Tabular Function is one of the arguments of the IF-THEN-ELSE function. In case when a TF is used within an IF-THEN-ELSE function and the TF is not an interactive type, then the argument for the TF is always the current time minus the time when the corresponding IF-THEN-ELSE was last becoming true.

The example below defines the control rod reactivity after scram. The scram signal is given by the function CF-104. When the scram signal is received, the control rods start moving down with a delay of 0.1 s. The control rod reactivity is tabulated as a function of time after scram as TF-900.

```
705900 Control rod reactivity
*      Group   Number     Fact.      Const.
700900   1        4        1.0        0.0     *  Type : IF-THEN-ELSE
709900   2                                     *  Arg. : Explicit
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.  Const.
710900      7   104  1   0    1.0     0.0     *  IF ( CF-104 (scram signal) <
710900      9   0    2   0    1.0E-90 1.0     *                             < 1.0 )
710900      9   0    2   0    1.0E-90 0.0     *  THEN rod reactivity is zero
710900      6   900  1   0    1.0     0.0     *  ELSE use TF-900 to define rod reactivity
*
*
605900  Control rod reactivity after scram
*       Fact.      Const.
601900   1.0        0.0        *  Scaling
602900   -30.0  0.0  0.0  0.0  *  Limits: Min, Max, dTF/dt
610900   1                     *  Type:   NON-INTERACTIVE !! otherwise the program will
*                              *          use the actual value of TF and not TF(t-t0)
*
*       Time after scram    Reactivity ($)
600900       0.0              0.0
600900       0.1              0.0
600900       0.2             -1.0
600900       0.3            -30.0
```

- **General Tabular Function (1, 05)**

A General Tabular Function is foreseen for more complicated applications than the simple Tabular Functions offered by the TF Package (Chapter 15). The main difference between a Tabular Function from the TF Package and a General Tabular Function from the CF Package is the argument type. In case of a TF the argument is always time. This allows excluding all TF from the main iteration loop to obtain implicit solution (the implicit values of all TF is known; they are tabulated versus time, so the value at any given time point is readily available). On the other hand the General Tabular Function type of a CF may use any SPECTRA variable as an argument. Since the implicit value of its argument may not be available at the start of the time step, this CF, like all other CF, is included in the main iteration loop to obtain implicit solution (the implicit treatment of CF may be altered by setting argument calculation on explicit or by the implicit solution indicator - see Volume 2).

There are two kinds of a General Tabular Function:

- A simple, one-dimensional General Tabular Function. This function has two arguments. The first argument defines an independent variable for this function. The second argument is a reference number of a Tabular Function from the TF Package. An example is shown below.

```
605105 Tabular Function
*        x      f(x)
600105 300.0    0.0
600105 400.0    1.0
*
705105 1-D General Tabular Function
*      Group   Number        Fact.   Const.
700105   1        5           1.0     0.0     *  Type    : General TF
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.    Const.
710105      1   123 14   1    1.0     0.0  *  Use CV-123-Temp-atms as argument
710105      6   105  1   0    1.0     0.0  *  Use table TF-105
```

In the above example a Tabular Function, TF-105, and a General Tabular Function, CF-105, are defined. The value of TF-105 will be equal to zero for times t<300.0 s, increase linearly to 1.0 during 300.0 < t <400.0 s, and remain equal to 1.0 afterwards. The value of CF-105 will be equal to zero when the atmosphere temperature in CV-123 is below T < 300.0 K, increase linearly to 1.0 with the temperature increasing between 300.0 < T < 400.0 K, and remain equal to 1.0 for higher temperatures.

- A two-dimensional Tabular Function, $F(x,y)$, may be created, where x is the first argument and y is the second argument. In this case the General Tabular Function should be defined by a number of TFs, each TF defining the value of function versus the argument x for one value of the argument y. An example of a 2-D function is shown below.

```
605101 Efficiency versus flow, pump speed of 20 rev/s
*        x      f(x)
600101  0.05    0.30
600101  0.2     0.60
600101  0.5     0.72
600101  0.8     0.80
600101  1.0     0.75
600101  1.1     0.70
*
605102 Efficiency versus flow, pump speed of 40 rev/s
*        x      f(x)
600102  0.05    0.40
600102  0.1     0.60
600102  0.2     0.70
600102  0.5     0.82
600102  1.0     0.85
600102  1.1     0.81
*
605103 Efficiency versus flow, pump speed of 50 rev/s
*        x      f(x)
600103  0.05    0.50
600103  0.1     0.70
600103  0.2     0.80
600103  0.35    0.90
600103  1.0     0.92
600103  1.1     0.88
*
705100 2-D General Tabular Function
*      Group   Number        Fact.   Const.
700100   1        5           1.0     0.0     *  Type    : General TF
*       y1    y2    y3
708100 20.0  40.0  50.0 * y-coordinate data points (pump speeds)
*      Arguments
* Pointers: (1) (2) (3) (4) Fact. Const.
710100       2  500 37   2  1.0   0.0  * x-argument: JN-500, pump flow (pool)
710100       6  101  1   0  1.0   0.0  * uses TF-101 for y = y1
710100       6  102  1   0  1.0   0.0  * uses TF-102 for y = y2
710100       6  103  1   0  1.0   0.0  * uses TF-103 for y = y3
710100       2  500 36   2  1.0   0.0  * y-argument: JN-500, pump speed (pool)
```

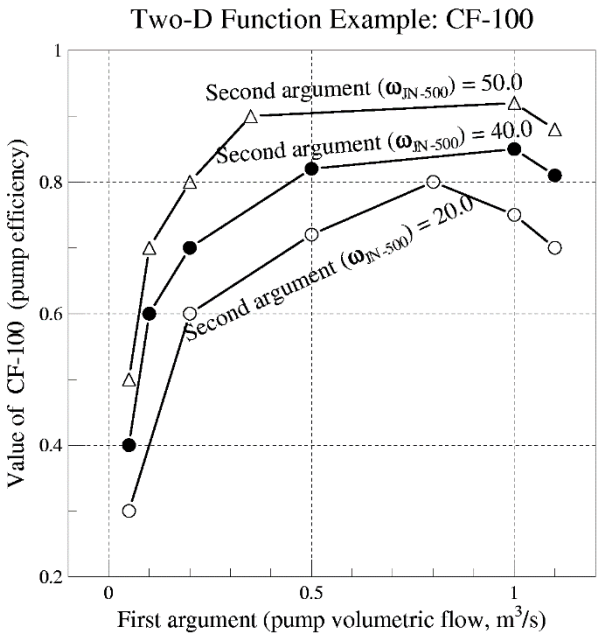


Figure 16-1 Example of a 2-D General Tabular Function

The above example defines the pump efficiency, as a function of pump volumetric flow and pump speed. During the calculations a 2-D interpolation is performed to find the value of a function. For example, suppose that during the execution the CF-100 (pump efficiency) should be calculated for the Volumetric flow of V = 0.15 m^3/s and speed of 45 rev/s. First the x-coordinate interpolations are performed; the calculated values are (see the data values above):

- ω = 40.0 rev/s TF-102 (x=0.15) = 0.65
- ω = 50.0 rev/s TF-103 (x=0.15) = 0.75

Now the linear interpolations between the two points shown above is performed, to give:

- ω = 45.0 rev/s CF-100 (x=0.15, y=45.0) = 0.70

- **Hysteresis (1, 06)**

The hysteresis function has three arguments. The first argument, x_1, is the independent variable, for which the function is calculated. The next two arguments point to tabular function numbers that define the hysteresis function. Two types of hysteresis are available:

Hysteresis Type 1

The type 1 hysteresis is shown Figure 16-2. There are two hysteresis points: $x_{H,1}$ and $x_{H,2}$, defined by the second and the third argument. When x_1 increases above $x_{H,2}$, then the function f is equal to 1.0. When it decreases below $x_{H,1}$, then f=0.0.

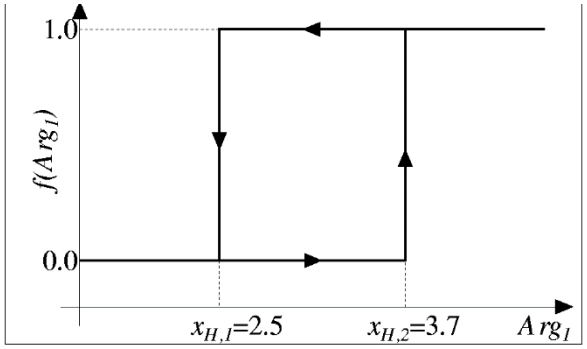


Figure 16-2 Hysteresis function type 1

Both tabular functions defining the hysteresis points should be constant (only one pair of data points). The value of the first hysteresis point, $x_{H,1}$, must be smaller than the value of the second hysteresis point $x_{H,2}$. Input defining a hysteresis function type 1 is shown below.

```
705200 Example of a Hysteresis Function Type 1
*      Group   Number        Fact.   Const.
700200   1        6           1.0     0.0     *  Type    : Hysteresis
*      Arguments
* Pointers: (1) (2) (3) (4) Fact.  Const.
710200       1  333 13   1  1.0E-6 0.0  * uses CV-333 pressure (in MPa) as an argument
710200       6  201  1   0  1.0    0.0  * uses TF-201 as a definition of the first point
710200       6  202  1   0  1.0    0.0  * uses TF-202 as a definition of the second point
*
605201 Hysteresis point 1
*        x      f(x)
600201   0.0    2.5   *  single data pair means hysteresis type 1
*
605202 Hysteresis point 2
*        x      f(x)
600202   0.0    3.7   *  single data pair means hysteresis type 1
```

Hysteresis Type 2

The type 1 hysteresis is shown Figure 16-3. There are two Tabular Functions, the "forward" and the "reverse" function, defined by the second and the third argument. When x_1 is increasing, the value of the hysteresis function is the maximum of its past value and that defined by the forward function. When x_1 is decreasing, the function takes the minimum of its past value and that given by the reverse function. That is:

$$f^n = \begin{cases} \max[\, f^{n-1}, TF_{for}(x_1^{\,n})] & if \quad x_1^{\,n} \geq x_1^{\,n-1} \\ \min[\, f^{n-1}, TF_{rev}(x_1^{\,n})] & if \quad x_1^{\,n} < x_1^{\,n-1} \end{cases}$$

Here f^n and f^{n-1} are the hysteresis function values at the current time step and the previous time step, $x_1^{\,n}$ and $x_1^{\,n-1}$ are the argument values at the current time step and the previous time step, TF_{for} and TF_{rev} are the Tabular Functions defining the function value at the increasing argument and the decreasing argument.

Input defining a hysteresis function type 1 is shown below.

```
705300 Example of a Hysteresis Function Type 2
*      Group   Number        Fact.   Const.
700300   1        6           1.0     0.0     *  Type    : Hysteresis
*      Arguments
* Pointers: (1) (2) (3) (4) Fact. Const.
710300       1  333 13   1  1.0E-6 0.0  * uses CV-333 pressure (in MPa) as an argument
710300       6  301  1   0  1.0    0.0  * uses TF-301 as the Forward function
710300       6  302  1   0  1.0    0.0  * uses TF-302 as the Reverse function
*
605301 Forward function
*        x      f(x)
600301   2.0    0.0   *  multiple data pair means hysteresis type 2
600301   3.0    0.0
600301   4.0    1.0
*
605302 Reverse function
*        x      f(x)
600302   2.0    0.0   *  multiple data pair means hysteresis type 2
600302   3.0    1.0
600302   4.0    1.0
```

Resulting value for the case when the first argument (pressure in CV-333) is increasing to 4.5 MPa and then decreasing to 0.1, is shown in Figure 16-3.

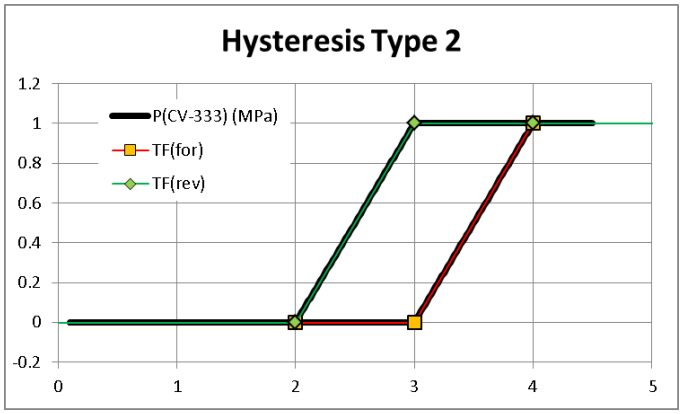


Figure 16-3 Hysteresis function type 2, P(CV-333) increases to 4.5 MPa and then decreases

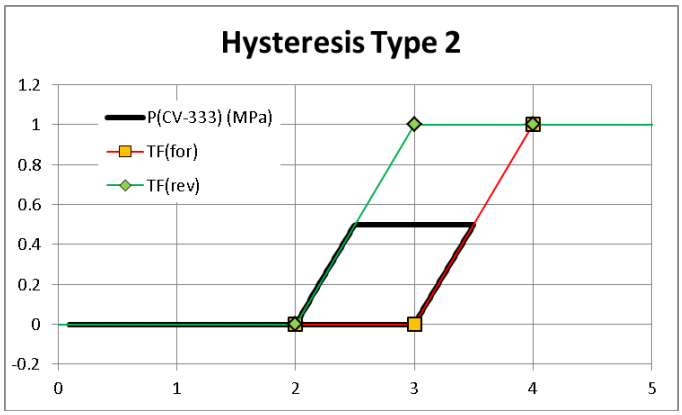


Figure 16-4 Hysteresis function type 2, P(CV-333) increases to 3.5 MPa and then decreases

In contrast to the type 1 function, which has only two values zero and one, the type 2 hysteresis can have a number of different values. For example, suppose that in the example case discussed above, the first argument (pressure in CV-333) is increasing to only to 3.5 MPa. The resulting hysteresis function is shown in Figure 16-4.

Selection of hysteresis type

- If both Tabular Functions have only one data point, then the program interprets it as the hysteresis function Type 1. The (constant) value of the first function must be smaller than the (constant) value of the second function.
- If both Tabular Functions have more than one data point, then the program interprets it as the hysteresis function Type 2. The first data points in both Tabular Functions must be the same. Similarly, the last data points in both Tabular Functions must be the same.
- If one of the Tabular Functions has only one data point and the other Tabular Function has more than one data point, then error message is issued and the program stops.

- **Derivative (1, 07)**

This Control Function evaluates the time derivative of an argument. The time derivative is calculated using the two time levels in the form:

$$f(x_1) = \frac{dx_1}{dt} = \frac{x_1 - \overline{x_1}}{\Delta t}$$

x_1 current value of the argument 1

$\overline{x_1}$ previous time step value of the argument 1

Δt time step, (s)

The following example calculates the rate of change of the water mass in CV-123 in tons per hour, using the derivative function, CF-107. The argument is the water mass in CV-123. The multiplier of 3.6 is used to convert the value in (kg/s) into tons per hour.

$$\text{CF-107} = 3.6 \cdot f(x_1) = 3.6 \cdot \left(\frac{dM_{pool}}{dt} \right)$$

```
705107 Rate of change of water mass in (t/hr)
*      Group   Number        Fact.   Const.
700107   1        7           3.6     0.0     *  Type    : Derivative
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710107       1  123 20   3    1.0     0.0     *  CV-123-Mass-pool
```

- **Integral (1, 08)**

This CF evaluates the time integral of an argument. The integral is calculated using an implicit formulation:

$$f(x_1) = \int_{t_0}^{t_0+\Delta t} x_1 dt = f(\overline{x_1}) + x_1 \cdot \Delta t$$

x_1 - current (end-of-time step, $t = t_0+\Delta t$) value of the argument 1

$\overline{x_1}$ - beginning of time step ($t = t_0$) value of the argument 1

An explicit formulation is also possible, with the old time step rather than the new time step value of x_1, if explicit argument calculation is requested by the user (see Volume 2). In such case:

$$f(x_1) = \int_{t_0}^{t_0+\Delta t} x_1 dt = f(\overline{x_1}) + \overline{x_1} \cdot \Delta t$$

A scaling factor, *S*, and additive constant, *A*, may be applied to any Control Function, as shown in section 16.3. In case of an integral, the use of these parameters is somewhat more difficult than for other types of CF, because for this function uses "itself" (i.e. it's own previous time step value) to calculate the new value. The user must remember that the scaling factor and the additive constant are applied after the integration is performed, in other words the integration process is not affected by *S* and *A*:

$$CF(t) = f(x_1, x_2, ..., t) \cdot S + A = \left(\int_{t_0}^{t_0+\Delta t} x_1 dt \right) \cdot S + A$$

The following example integrates the function:

$$x_1(t) = 1.0 - t$$

The integration gives:

$$f(x_1) = \int_0^t x_1 dt' = t - \frac{t^2}{2}$$

If a scaling factor of $S = -2.0$ and an additive constant of $A = 1.0$ are applied, then the final value of CF becomes:

$$CF(t) = \int_0^t x_1 dt' \cdot S + A = 1 - 2t + t^2$$

The input defining such function is shown below. The integral is defined by CF-108. The integrated function is defined by TF-108. The initial value of this kind of function is by default equal to zero. However, because one needs to obtain the function shown above, the initial value must be equal to 1.0. Therefore the user must define the initial value of CF-108, as shown in the example below.

```
705108 Example of an integral
*      Group   Number        Fact.   Const.
700108   1        8          -2.0     1.0     *  Type    : Integral
701108   2       1.0                          *  INITIAL VALUE
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710108       6  108  1   0    1.0     0.0     *  TF-108-Valu-0000
*
605108 Integrated function x = 1 - t
*        x      f(x)
600108   0.0    1.0
600108 100.0  -99.0
```

Results are shown in Figure 16-5. It may easily be checked that the values are in agreement with the equation above.

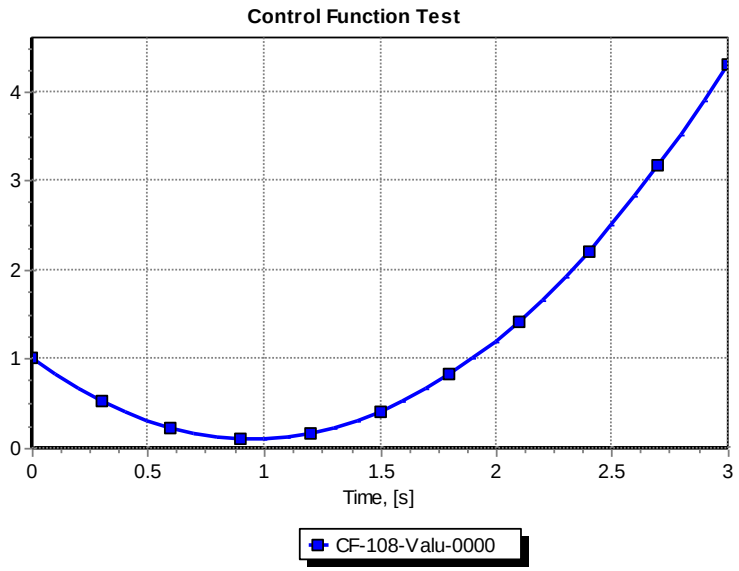


Figure 16-5 Results of the integrated function

- **Random number (1, 09)**

This Control Function returns a rundom number from the range determined by CF limits. The value is calculated as:

$$f = rnd() \cdot S(Arg_1) + A(Arg_1)$$

rnd() random function, returning a random number between 0.0 and 1.0
$S(Arg_1)$ scaling factor of argument 1
$A(Arg_1)$ additive constant of argument 1

Arg_1 is a dummy argument and may be anything, for example the value of Control Function itself.

16.2.2 CF Group 2

- **Absolute Value (2, 01)**

This function calculates absolute value of an argument:

$$f(x_1) = |x_1|$$

For example, in order to calculate an absolute value of gas velocity in JN-234, the following function is used:

```
705201 Absolute value of gas velocity in JN-234
*      Group   Number        Fact.   Const.
700201   2        1           1.0     0.0     *  Type    : Absolute value
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710201       2  234  1   1    1.0     0.0     *  JN-234-Velo-atms
```

- **Minimum Value (2, 02)**

This function calculates the minimum value of several arguments.

$$f(x_1, x_2, ...x_n) = Min(x_1, x_2, ...x_n)$$

For example, in order to calculate a minimum temperature of SC-345, the following function is used:

```
705202 Minimum temperature of SC-345
*      Group   Number        Fact.   Const.
700202   2        2           1.0     0.0     *  Type    : Minimum
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710202       3  345 16   1    1.0     0.0     *  SC-345-Tcel-0001
710202       3  345 16   2    1.0     0.0     *  SC-345-Tcel-0002
710202       3  345 16   3    1.0     0.0     *  SC-345-Tcel-0003
710202       3  345 16   4    1.0     0.0     *  SC-345-Tcel-0004
710202       3  345 16   5    1.0     0.0     *  SC-345-Tcel-0005
```

In this example it is assumed that there are 5 nodes in SC-345.

- **Maximum Value (2, 03)**

This function calculates the maximum value of several arguments.

$$f(x_1, x_2, ...x_n) = Max(x_1, x_2, ...x_n)$$

- **Square Root (2, 04)**

This function calculates the square root of an argument. The argument must be greater than, or equal to zero. If it is not, then an appropriate error message is written to the diagnostics file, and the calculations are stopped.

$$f(x_1) = \sqrt{x_1}$$
$$x_1 \geq 0.0$$

- **Exponent (2, 05)**

This function calculates the *e* to the power equal to the argument:

$$f(x_1) = \exp(x_1)$$

The argument must be within the limits:

$$-99.0 \leq x_1 \leq +99.0$$

- **Natural Logarithm (2, 06)**

This function calculates the natural logarithm of an argument.

$$f(x_1) = \ln(x_1)$$
$$x_1 > 0.0$$

- **Decimal Logarithm (2, 07)**

This function calculates the decimal logarithm of an argument.

$$f(x_1) = \log_{10}(x_1)$$
$$x_1 > 0.0$$

- **Sine Function (2, 08)**

This function calculates the sine of an argument expressed in radians:

$$f(x_1) = \sin(x_1)$$

- **Cosine Function (2, 09)**

This function calculates the cosine of an argument expressed in radians.

$$f(x_1) = \cos(x_1)$$

- **Tangent Function (2, 10)**

This function calculates the tangent of an argument expressed in radians.

$$f(x_1) = \tan(x_1)$$

- **Arc Sine Function (2, 11)**

This function calculates the arc sine of an argument expressed in radians.

$$f(x_1) = \arcsin(x_1)$$

The argument must be within the limits:

$$0.0 \le x_1 \le 1.0$$

- **Arc Cosine Function (2, 12)**

This function calculates the arc cosine of an argument expressed in radians.

$$f(x_1) = \arccos(x_1)$$

The argument must be within the limits:

$$0.0 \le x_1 \le 1.0$$

- **Arc Tangent Function (2, 13)**

This function calculates the arc tangent of an argument expressed in radians.

$$f(x_1) = \arctan(x_1)$$

- **Hyperbolic Sine Function (2, 14)**

This function calculates the hyperbolic sine of an argument.

$$f(x_1) = \sinh(x_1)$$

The argument must be within the limits:

$$-99.0 \le x_1 \le +99.0$$

- **Hyperbolic Cosine Function (2, 15)**

This function calculates the hyperbolic cosine of an argument.

$$f(x_1) = \cosh(x_1)$$

The argument must be within the limits:

$$-99.0 \le x_1 \le +99.0$$

- **Hyperbolic Tangent Function (2, 16)**

This function calculates the hyperbolic tangent of an argument.

$$f(x_1) = \cosh(x_1)$$

16.2.3 CF Group 3

- **Error Function (3, 01)**

The error function is defined as:

$$f(x_1) = erf(x_1) = \frac{2}{\pi}\int_0^{x_1} e^{-u}du$$

The argument must be within the limits:

$$0.0 \le x_1 \le 10^{99}$$

- **Complementary Error Function (3, 02)**

The complementary error function is defined as:

$$f(x_1) = erfc(x_1) = \frac{2}{\pi}\int_{x_1}^{\infty} e^{-u}du = 1.0 - erf(x_1)$$

The argument must be within the limits:

$$0.0 \le x_1 \le 10^{99}$$

- **Gamma Function (3, 03)**

The gamma function is defined as:

$$f(x_1) = \Gamma(x_1) = \int_0^\infty u^{x_1-1} e^{-u} du$$

The argument must be within the limits:

$$10^{-10} \le x_1 \le 99$$

- **Log Gamma Function (3, 04)**

The log gamma function is defined as:

$$f(x_1) = \ln \Gamma(x_1) = \ln \int_0^\infty u^{x_1-1} e^{-u} du$$

The argument must be within the limits:

$$10^{-10} \le x_1 \le 99$$

- **Reactor Kinetics Function (3, 05)**

This function gives total reactor power obtained from the reactor kinetics model. The argument must be time. The definition depends on the reactor type:

- In case of solid fuel (IFTORK=1): CF = fission power + decay power
- In case of circulating fuel (IFTORK>1): CF = fission power
  (the decay power is obtained in such case from the RT Package)

The user may apply scaling factor for convenience. For example, to obtain the relative power of a 1000 MW reactor, the scaling factor of SCLFCF=10^{-9} should be applied.

16.3 Limits, Scaling Factors and Additive Constants

The user may impose limits on each Control Function, with respect to the value as well as the rate of change. The limits are:

- Value:

$$CF_{min} \le CF \le CF_{max}$$

- Rate of change

$$-\left.\frac{d(CF)}{dt}\right|_{min} \le \frac{d(CF)}{dt} \le \left.\frac{d(CF)}{dt}\right|_{max}$$

CF_{min}	Minimum value of CF (default value of -10^{99})
CF_{max}	Maximum value of CF (default value of $+10^{99}$)
$d(CF)/dt\|_{min}$	Maximum rate of change for a decreasing value of CF (default value of 10^{99})
$d(CF)/dt\|_{max}$	Maximum rate of change for an increasing value of CF (default value of 10^{99})

The default values result in practically no limits on the CF. In order to apply the limits, the user must enter appropriate values in the input deck.

For each Control Function a scaling factor and an additive constant are used. The value of a function is calculated from:

$$CF(t) = f(x_1, x_2, ..., t) \cdot S + A$$

$f(x_1, x_2, ..., t)$	value of function, calculated for the current values of arguments, x_1, x_2, ..., t
S	scaling factor, with the default value of 1.0
A	additive constant, with the default value of 0.0

The scaling factor and additive constant are applied before the limits are applied (see the discussion in section 15.4).

Apart from the scaling factor and additive constant applied for the CF itself, the user may apply scaling factors and additive constants for each argument individually. In the example shown below the ratio (p/ρ) is calculated for CV-123, where p is pressure in bar and ρ is gas density in g/cm^3.

```
705999 CF-999 = p (bar) / rho (g/cm3)
*      Group   Number        Fact.   Const.
700999   1        2           1.0     0.0     *  Type    : Multiply/divide
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710999       1  123 13   1    1.0E-5  0.0     *  CV-123-Pres-atms (bar)
710999      -1  123 24   1    1.0E-1  0.0     *  CV-123-Dens-atms (g/cm3)
```

This is more clear and easier for verification than applying the overall scaling factor of 10^{-4}:

```
705999 CF-999 = p (bar) / rho (g/cm3)
*      Group   Number        Fact.   Const.
700999   1        2         1.0E-4    0.0     *  Type    : Multiply/divide
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710999       1  123 13   1    1.0     0.0     *  CV-123-Pres-atms (Pa)
710999      -1  123 24   1    1.0     0.0     *  CV-123-Dens-atms (kg/m3)
```

16.4 Limits for CF Arguments

SPRCTRA checks if the argument values are within acceptable limits for a given function. Those limits are given for each CF type in sections 16.2.1, 16.2.2, and 16.2.3. The limits imposed in SPECTRA on control function arguments are more restrictive than the limits imposed by the FORTRAN compiler. For example, in SPECTRA an argument of the hyperbolic sine function may be within the range $|x_1| < 99.0$. FORTRAN will accept values of $|x_1|$ up to a little more than 200 (for different compilers the number may be somewhat different), above which an arithmetic overflow will occur. In SPECTRA the range was narrowed to make sure that there are no FORTRAN overflow errors during the code execution and thus have a diagnostics message from the SPECTRA level, which is much more informative (SPECTRA will inform the user in which Control Function the problem occurred, what is the current argument value and the acceptable range), rather than from the FORTRAN level (resulting in a message like "arithmetic overflow", or "floating exception").

In order to avoid termination of calculation on CF arguments are out of range, the user should use limits in his CF definition. For example, if the user wishes to calculate hyperbolic sine of a pump speed, N_p, in rev/s:

$$f(x_1) = \sinh(N_p)$$

Assuming that the pump is located in JN-234, the function may be defined as follows:

```
705901 Sinh(N-pump)
*      Group   Number        Fact.   Const.
700901   2       14           1.0     0.0     *  Type    : Sinh
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710901       2  234 36   3    1.0     0.0     *  JN-234-PSpd-pool
```

With the above definition there is a risk that calculation will be terminated, which will occur if at any time of the transient the pump speed becomes larger than 99.0 revolutions per second. An easy way to avoid such termination is to define an intermediate Control Function, which will put the appropriate limits on the desired argument. In the example shown below, the limits of –98 and +98 are applied:

```
705902 (N-pump) with limits acceptable by the Sinh function
*      Group   Number        Fact.   Const.
700902   1        1           1.0     0.0     *  Type    : Add/subtract
702902    -98.0   98.0                        *  Limits
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710902       2  234 36   3    1.0     0.0     *  JN-234-PSpd-pool
*
*
705901 Sinh(N-pump)
*      Group   Number        Fact.   Const.
700901   2       14           1.0     0.0     *  Type    : Sinh
*      Arguments
* Pointers: (1) (2) (3) (4)  Fact.   Const.
710901       7  902  1   0    1.0     0.0     *  CF-902-Valu-0000 =
*                                             *  = JN-234-PSpd-pool with limits
```

16.5 Examples of Control Functions

Several simple examples of Control Functions are provided in previous section. Here some more complicated examples are shown to illustrate the possible applications of tabular and control functions.

16.5.1 Hydrogen Recombiner

16.5.1.1 Theoretical Basis of the Recombiner Performance

As the example problem, the model of a hydrogen recombiner, similar to the one presented in [123] is discussed. The aim was to model a hydrogen recombiner unit, in which reaction would proceed according to the experimentally determined reaction kinetics, expressed as:

$$m = (C_0 + C_1 p) \cdot Min(x_{H_2}, x_{H_2,\max})$$

m hydrogen consumption rate, (kg/s)
C_0, C_1 model constants
p pressure, (Pa)
x_{H2} hydrogen volumetric fraction in the vicinity of the recombiner, (-)
$x_{H2,\ max}$ saturation hydrogen concentration, above which the recombination rate is constant

The recombination proceeds as long as the hydrogen concentration in the vicinity of the recombiner is above the minimum value of $x_{H2,\ min}$, and if the oxygen fraction is above the minimum value of $x_{O2,\ min}$. The example shown here uses the set of constants for the recombiner type FR-90/1-360:

C_0 = (0.012×100) (g/s) = 1.20×10^{-3} kg/s
C_1 = (0.010×100) (g/s-bar) = 1.00×10^{-8} kg/(s-Pa)
$x_{H2,\ min}$ = 0.02
$x_{H2,\ max}$ = 0.08
$x_{O2,\ min}$ = 0.02

16.5.1.2 SPECTRA Model

The recombiner function is to combine some hydrogen and oxygen to produce steam. Therefore, from calculational standpoint it represents a certain negative mass source of hydrogen and oxygen, and a positive mass source of steam. The mass sources may be defined for a Control Volume using Tabular or Control Functions (see section 15.1). In the present example a set of Control and Tabular Functions was applied to determine the mass sinks and source. This set of functions is described below.

Mass transfer rate in a recombiner unit, described in the section 16.5.1.1, was modelled using three Tabular Functions and six Control Functions, as follows.

- Pressure multiplier, TF-121 = $C_0 + C_1 p$

The pressure dependent multiplier is defined using 2 data points covering the expected pressure range during the analysis. The values of TF-121 are:

No.	Argument = p	TF value
1	0.0	$C_0 = 1.20\times10^{-3}$
2	10^6	$C_0 + 10^6\times C_1 = 1.12\times10^{-2}$

The function TF-121 is shown in Figure 16-6. SPECTRA input for TF-001 is provided below. In order to calculate the pressure multiplier one needs to create a Control Function (type: General Tabular Function). The Control Function specifies the argument that needs to be used with the tabulated data and the table number. Of course for recombiners located in different parts of the system, different CF must be used, because the argument is always the pressure in the CV where the recombiner is located. TF-121 on the other hand is a general definition of the pressure multiplier, and is applicable for all recombiners. In the example below, CF-003 is used to define the pressure multiplier for the recombiner unit 1 ("Reco-1"), located in CV-003.

```
605121  P  multiplier, recombiner type FR-90/1-360
*        x       y       *  TF-121:  P  multiplier, type 360
600121  0.000  1.200e-3 *  (1) (0.012)            * (100/1000)
600121  1.0E6  1.120E-2 *  (2) (0.012 + 10*0.010) * (100/1000)
*
705003 Reco-1,  P  multiplier
*       (General TF)       Scale     Additive
*      Group   Number      Fact.      Const.
700003   1        5         1.0        0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710003       1 003  13   1  *  CV-003-Pres-atms   Argument
710003       6 121   1   0  *  TF-121-Valu-0000   Tabulated data points
```

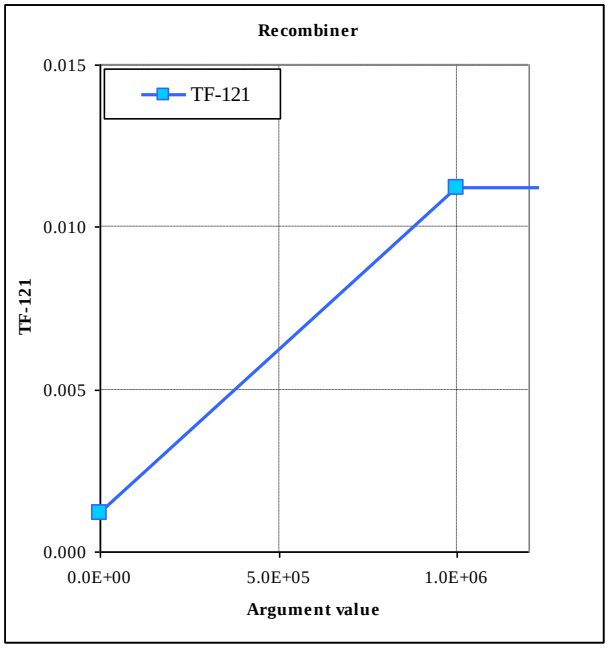


Figure 16-6 Pressure multiplier - function TF-121

- Hydrogen concentration multiplier, TF-019

The hydrogen concentration multiplier is expressing the fact that the reaction rate starts at minimum hydrogen fraction, is proportional to hydrogen concentration up to $x_{H2,\,max}$, and then is constant. TF-019 is defined by three data points, the values are:

No.	Argument = x_{H2}	TF Value
1	$x_{H2,\,min}$ = 0.02	0.0
2	$x_{H2,\,min}+\delta x$=0.021	$x_{H2,\,min}$ = 0.021
3	$x_{H2,\,max}$ = 0.080	$x_{H2,\,max}$ = 0.080

Here δx is a small number that needs to be added because TF argument values must be tabulated in ascending order (section 15.2). There is no need to include more points, since the end point values are kept beyond the range of data (section 15.2).

The function TF-019 is shown in Figure 16-7. The SPECTRA input for TF-019 is provided below. In order to calculate the hydrogen one needs to create a Control Function that specifies the argument that needs to be used with the tabulated data and the table number. In the example below, CF-001 is used to define the hydrogen multiplier for the recombiner Reco-1

It should be remembered that if a stratification model is used in the analysis, the local H_2 concentration at the structure representing the recombiner must be used. Therefore local concentration at SC-081 is used. If the stratification model is not used, the concentration in CV-003 could be used as well.

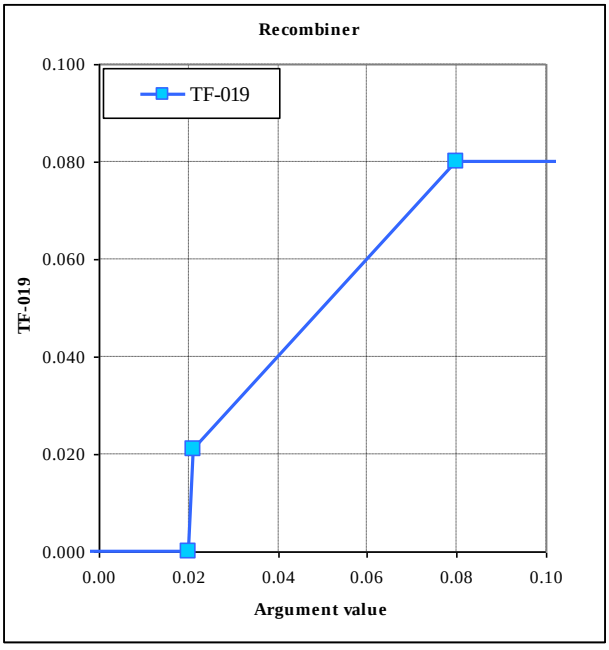


Figure 16-7 Hydrogen concentration multiplier - function TF-019

```
*        x       y       *  TF-019:  H2 multiplier
600019  0.020   0.000     *  (1)
600019  0.021   0.021     *  (2)
600019  0.080   0.080     *  (3)
*
705001 Reco-1,  H2  multiplier
*       (General TF)       Scale      Additive
*      Group   Number      Fact.       Const.
700001   1        5         1.0         0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710001       3 081  10   1  *  SC-081-Vfrc-H2_r   Argument
710001       6 019   1   0  *  TF-019-Valu-0000   Tabulated data points
```

- Oxygen concentration multiplier, TF-020

The reaction starts when the oxygen fraction exceeds the minimum value of $x_{O2,\,min}$. TF-003 is defined by:

No.	Argument = x_{O2}	TF Value
1	$x_{O2,\,min}$	0.0
2	$2x_{O2,\,min}$	1.0

A transition zone is defined, in which the reaction rate increases from zero to the maximum value. This zone is necessary since the independent argument must be in ascending order (section 15.2). Furthermore, step functions must be avoided for control function arguments since they may lead to numerical convergence problems.

The SPECTRA input for TF-020 is provided below. The function TF-020 is shown in Figure 16-8.

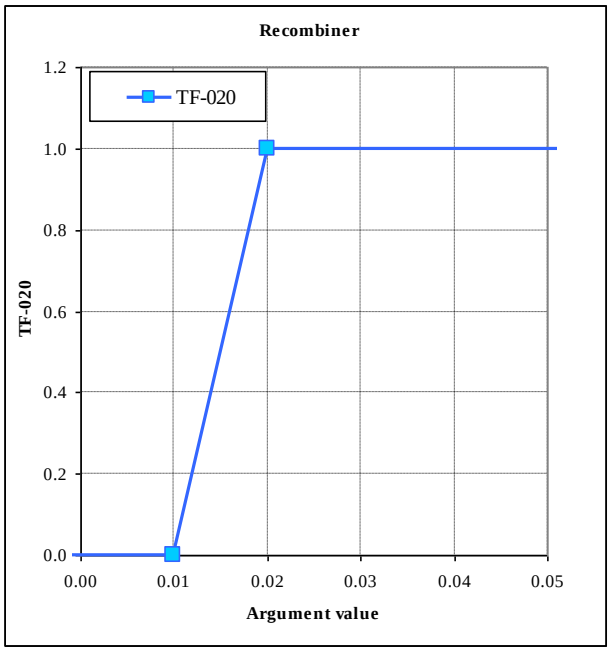


Figure 16-8 Oxygen concentration multiplier - function TF-020

In order to calculate the oxygen multiplier one needs to create a Control Function that specifies the argument that needs to be used data and the table number. In the example below, CF-002 is used to define the oxygen multiplier for the recombiner Reco-1, using local oxygen concentration at SC-089 (structure representing the recombiner). If the stratification model was not used the concentration in CV-003 could be used as well.

```
*        x      y       *  TF-020:  O2 multiplier
600020  0.010  0.000    *  (1)
600020  0.020  1.000    *  (2)
*
705002 Reco-1,  O2  multiplier
*       (General TF)      Scale     Additive
*      Group   Number     Fact.      Const.
700002   1        5        1.0        0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710002       3 081  10   5  *  SC-081-Vfrc-O2_r   Argument
710002       6 020   1   0  *  TF-020-Valu-0000   Tabulated data points
```

- $H_2 + O_2$ recombination mass transfer rate, CF-004

The total mass transfer rate due to hydrogen recombination is calculated as the product of the three multipliers, described above. The multiply/divide control function is used (see section 16.2.1), and it uses the Control Functions CF-001, CF-002, CF-003, as arguments. Additionally the scaling factor of –9.0 is used for the following reason. Consumption of one mole (2 kg) of hydrogen is accompanied by consumption of a half mole (16 kg) of oxygen. Thus the total consumption of H_2 and O_2 is 9 kg per 1 kg of hydrogen, hence the scaling factor of 9.0. Since this is a mass sink, the negative sign is used:

$$\text{CF-004} = -9.0 \times (\text{CF-001} \times \text{CF-002} \times \text{CF-003})$$

```
705004 Reco-1,  H2+02 consumption rate = -9 * ( 1 Recombiner type R1 )
*        (Multiply)       Scale     Additive
*      Group   Number     Fact.      Const.
700004   1        2       -9.0        0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710004       7 001   1   0  *  CF-001-Valu-0000
710004       7 002   1   0  *  CF-002-Valu-0000
710004       7 003   1   0  *  CF-003-Valu-0000
```

- H_2O source due to recombiner operation, CF-005

The mass source of steam is the same as the total consumption rate but with the plus sign, thus the multiplier –1.0 is used. The control function type is add/subtract type (see section 16.2.1), and it uses the control function CF-004 as an argument.

```
705005 Reco-1,  H20 generation rate = -1 * ( H2+02 consumption rate )
*      (Add/subtract)     Scale     Additive
*      Group   Number     Fact.      Const.
700005   1        1       -1.0        0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710005       7 004   1   0  *  CF-004-Valu-0000
```

- Heat of reaction, CF-006

According to [32] the heat of reaction is 241.8 kJ per mole of created steam. That means 2.418×10^8 J/kmol, which equals to 1.34×10^7 J per kilogram of created steam. The control function CF-006 calculates the heat generation due to reaction for the recombiner Reco-1. The function uses CF-005 as an argument, and the scaling factor of 1.34×10^7.

```
705006 Reco-1 - Mass & Energy Sources for CV-003
*      (Add/subtract)     Scale     Additive
*      Group   Number     Fact.      Const.
700006   1        1       1.34E7      0.0  *
*      Arguments
* Pointers: (1) (2) (3) (4) *
710006       7 005   1   0  *  CF-005-Valu-0000
```

- Mass and energy sources for CV-003

The last three control functions represent the mass and energy effects of the recombiners. The mass and energy sources for the Control Volume where recombiners are present, are defined by:

Mass source 1: Mass sink of hydrogen and oxygen
Mass source rate given by: CF-004
Gas composition: H_2 = 0.11111 (=2/18)
O_2 = 0.88889 (=16/18)

Mass source 2: Mass source of steam
Mass source rate given by: CF-005.
Gas composition: H_2O=1.0

Energy source: Reaction energy
Energy source rate given by: CF-006

The elevation of the source is 2.5 m above the floor of CV-003 (location of the recombiner in the real plant). The area and diameter of the mass sources are not very important for the present application and were arbitrarily chosen as 0.1 m^2 and 0.01 m respectively. The temperature and pressure of the source gas (needed only for positive sources) were defined using TF-022 and TF-009 respectively. The values were not very important for the analysis, therefore the pressure was simply set to 2 bar and temperature to 400 K (approximately saturation temperature at about 2 bar).A more exact way would be to define these parameters using Control Functions, equal to the pressure and the saturation temperature in CV-003. The SPECTRA input is shown below.

```
*       Reco-1 - Mass & Energy Sources for CV-003
*        Z      A     D    IM/ET ITT  IPT    Gas Mass   Gas Mass
*       (m)    (m2)   (m)    (-)  (-)  (-)    No. Frac.  No. Frac.
131003  2.5    0.1    0.01    -4   22    9     1 0.1111   5 0.8889  * m: H2, O2
132003  2.5    0.1    0.01    -5   22    9     3 1.0000             * m: H2O
141003  2.5                   -6                                    * E
```

16.6 2-D Control Functions

16.6.1 2-D Tabular Function

A two-dimensional Tabular Function, *F(x,y)*, may be created, where *x* is the first argument and *y* is the second argument. All other arguments are pointers to Tabular Functions. A double interpolation is performed, as follows:

- First, an interpolation in the *x*-direction is performed. The values of TF-s are calculated for the argument *x*. The values are stored as a temporary vector.
- Second, an interpolation in the *y*-direction is performed. Here the argument is *y*, the independent argument is the user-defined *y*-coordinates (ARG2CF) while the dependent argument is the temporary vector.

The general Tabular Function should be defined by a number of TFs, each TF defining the value of function versus the argument *x* for one value of the argument *y*. The *y*-arguments are defined in this record. An example of a 2-D function is shown below.

```
605101 Efficiency versus flow, pump speed of 20 rev/s
*        x      f(x)
600101  0.05    0.30
600101  0.2     0.60
600101  0.5     0.72
600101  0.8     0.80
600101  1.0     0.75
600101  1.1     0.70
*
605102 Efficiency versus flow, pump speed of 40 rev/s
*        x      f(x)
600102  0.05    0.40
600102  0.1     0.60
600102  0.2     0.70
600102  0.5     0.82
600102  1.0     0.85
600102  1.1     0.81
*
605103 Efficiency versus flow, pump speed of 50 rev/s
*        x      f(x)
600103  0.05    0.50
600103  0.1     0.70
600103  0.2     0.80
600103  0.35    0.90
600103  1.0     0.92
600103  1.1     0.88
*
*
705100 2-D General Tabular Function
*      Group   Number        Fact.   Const.
700100   1        5           1.0     0.0     *  Type    : General TF
*       y1    y2    y3
708100 20.0  40.0  50.0 * y-coordinate data points (pump speeds)
*      Arguments
* Pointers: (1) (2) (3) (4) Fact. Const.
710100       2  500 37   2  1.0   0.0  * x-argument: JN-500, pump flow (pool)
*
710100       6  101  1   0  1.0   0.0  * uses TF-101 for y = y1
710100       6  102  1   0  1.0   0.0  * uses TF-102 for y = y2
710100       6  103  1   0  1.0   0.0  * uses TF-103 for y = y3
*
710100       2  500 36   2  1.0   0.0  * y-argument: JN-500, pump speed (pool)
```

The values of CF-100, defined as shown above, are shown in Figure 16-9. Results of the pump test case, defined in (\Z-INPUTS\CF\TF-2D) are shown in Figure 16-10.

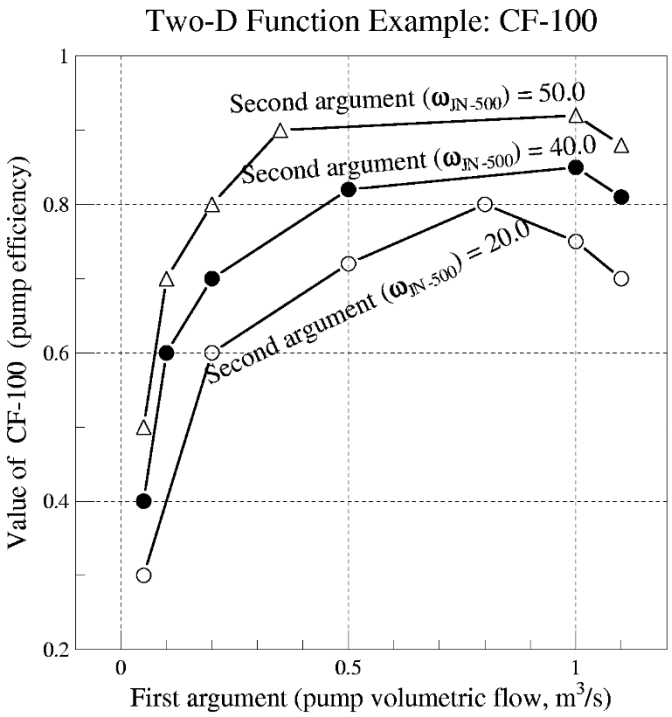


Figure 16-9 Pump efficiency, defined using a 2-D General Tabular Function

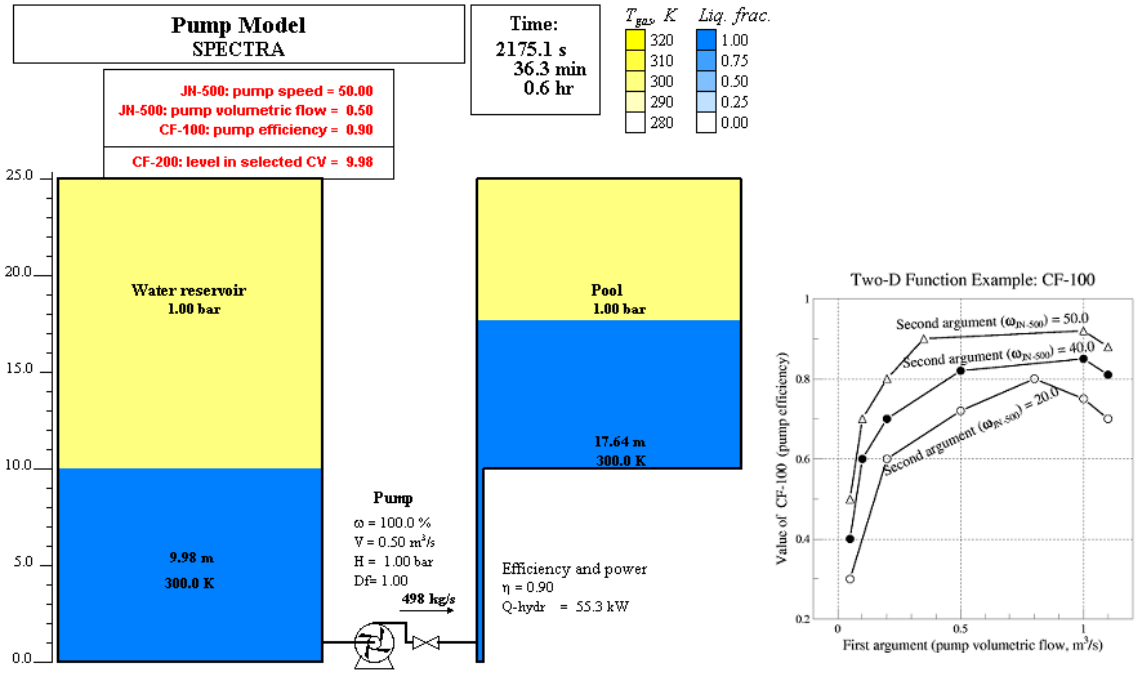


Figure 16-10 Pump flow test with efficiency defined using a 2D function

16.6.2 Time-Dependent 2-D Function

In the previous example any parameter could be used as the *x* argument, however, the next arguments had to be Tabular Functions. In the present example, the first argument is time. In such case, the following arguments may be any time-dependent parameter in the SPECTRA database.
An example of such function is shown below.

```
*
705200 2-D General Tabular Function - INTERPOLATION AMONG SELECTED ARGUMENTS
*      Group   Number        Fact.   Const.
700200   1        5           1.0     0.0     *  Type    : General TF
*       y1    y2    y3
708200  1.0   2.0   3.0 * y-coordinate data points (SELECTOR VALUES)
*      Arguments
* Pointers: (1) (2) (3) (4) Fact. Const.
710200       9   0   1   0  1.0   0.0  * x-argument: TIME (MUST BE TIME FOR THIS TYPE OF CF)
*
710200       1  001  8   0  1.0   0.0  * uses liquid level in CV-001 versus time
710200       1  002  8   0  1.0   0.0  * uses liquid level in CV-002 versus time
710200       1  001  8   0  1.0   0.0  * uses liquid level in CV-001 versus time
*
710200       6  201  1   0  1.0   0.0  * y-argument: SELECTOR
*
*
605201 SELECTOR
600201      0.0      1.
600201   1000.0      1.
600201   1100.0      2.
600201   2000.0      2.
600201   2100.0      3.
```

The input above, defines the following function:

SELECTOR	CF Value
1.0	Zpool(CV-001)
2.0	Zpool(CV-002)
3.0	Zpool(CV-001)

Linear interpolation is performed for the intermediate values of the selector function. The function was defined within the pump test case, defined in (\Z-INPUTS\CF\TF-2D). The values of CF-200, defined as shown above, are shown in Figure 16-11. Figure 16-10 shows the state of the case for *t* = 2175 s. At this time, the value of the selector function (TF-201) is 3.0. Therefore the value of CF-200 is equal to the liquid level in CV-001 at this time: 9.98 m.

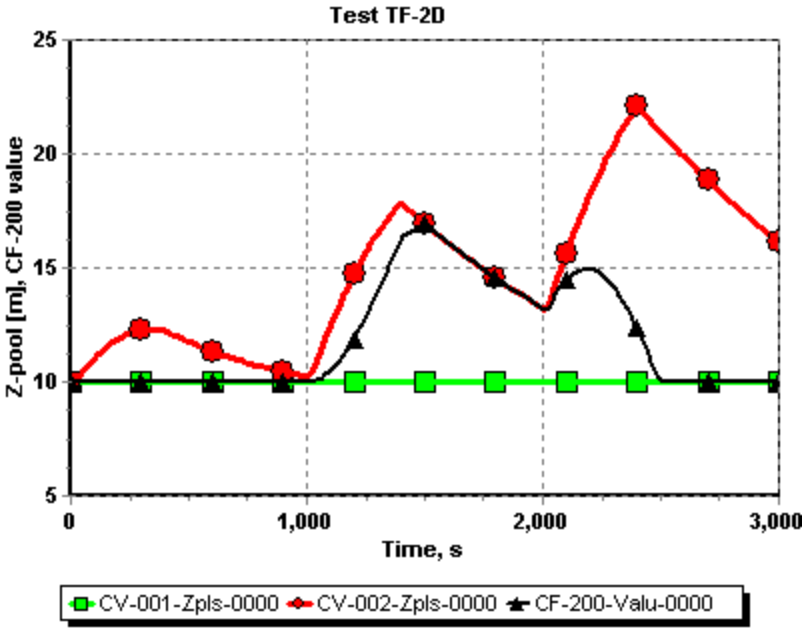


Figure 16-11 Values of a 2D function, CF-200 in the example case

16.6.3 Stiff Set of Ordinary Differential Equations (ODE)

16.6.3.1 Theoretical Basis

As an example of the applicability of Control Functions in SPECTRA, a set of differential equations is solved. A "stiff" set of differential equation was selected, taken from [1]. Stiff equation sets are particularly difficult to solve. The equation set is:

$$(1)\quad \frac{dy_1}{dt} = -0.013 \cdot y_1 - 1000 \cdot y_1 \cdot y_3$$

$$(2)\quad \frac{dy_2}{dt} = -2500 \cdot y_2 \cdot y_3$$

$$(3)\quad \frac{dy_3}{dt} = -0.013 \cdot y_1 - 1000 \cdot y_1 \cdot y_3 - 2500 \cdot y_2 \cdot y_3$$

The initial conditions are:

$$y_1(0) = 1$$
$$y_2(0) = 1$$
$$y_3(0) = 0$$

The exact solution was obtained using a solver specifically useful for stiff equation system, taken from [1]. These results are presented in the next section, as the "theoretical" solution, to distinguish them from SPECTRA results, although this is also a numerical solution but obtained using a different tool than SPECTRA.

16.6.3.2 SPECTRA Model and Results

The equation set was modeled using Control Functions. Nine Control Functions were used to define the analyzed equation set. The definitions of the Control Function are shown in Table 16-2. A listing of the SPECTRA input for these functions is shown below. For the first three functions the initial values are specified. Note that these are the integral-type functions, and for such functions SPECTRA requires the user to enter the initial values (see Volume 2). For other Control Functions the initial values were left to be calculated by the code. The input deck is quite self-explanatory and no more comments are needed.

Calculations were performed using the time step of 1.0 (s). The run was completed in 10 advancements, without any time step reductions, with a small number of iterations per time step (10 - 20). Results are shown in Figure 16-12. It is seen that the results are very accurate. Further discussion of this test run is provided in Volume 3.

Table 16-2 Definition of Control Functions for the Stiff ODE test.

CF No. and type		Value	Definition of CF	Initial Value
CF	Type			
CF-001	Integral	y_1	$\int$ (CF-007) dt	1.0
CF-002	Integral	y_2	$\int$ (CF-008) dt	1.0
CF-003	Integral	y_3	$\int$ (CF-009) dt	0.0
CF-004	Multiply	$0.013 \times y_1$	0.013×CF-001	
CF-005	Multiply	$1000 \times y_1 \times y_3$	1000×CF-001×CF-003	
CF-006	Multiply	$2500 \times y_2 \times y_3$	2500×CF-002×CF-003	
CF-007	Add	dy_1/dt	– CF-004 – CF-005	
CF-008	Add	dy_2/dt	– CF-006	
CF-009	Add	dy_3/dt	– CF-004 – CF-005 – CF-006	

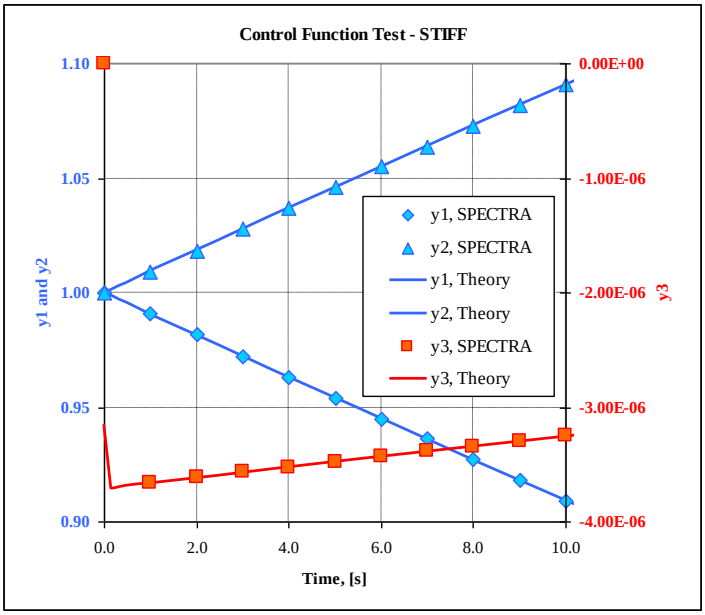


Figure 16-12 Stiff equation system - SPECTRA with Δt=1.0 s and “theoretical” solution.

```
*
*
705001   CF-001 = y1
701001   2   1.0              *  Initial value
700001   1    8     1.0  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710001       7  007  1   0    *  CF-007-Valu-0000
*
*
705002   CF-002 = y2
701002   2   1.0              *  Initial value
700002   1    8     1.0  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710002       7  008  1   0    *  CF-008-Valu-0000
*
*
705003   CF-003 = y3
701003   2   0.0              *  Initial value
700003   1    8     1.0  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710003       7  009  1   0    *  CF-009-Valu-0000
*
* ------------------------------------------------
*
*
705004   CF-004 = 0.013*y1
700004   1    2   0.013  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710004       7  001  1   0    *  CF-001-Valu-0000
*
*
705005   CF-005 = 1000*y1*y3
700005   1    2   1000.0 0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710005       7  001  1   0    *  CF-001-Valu-0000
710005       7  003  1   0    *  CF-003-Valu-0000
*
*
705006   CF-006 = 2500*y2*y3
700006   1    2   2500.0 0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710006       7  002  1   0    *  CF-002-Valu-0000
710006       7  003  1   0    *  CF-003-Valu-0000
*
* ------------------------------------------------
*
*
705007   CF-007 = y1'
700007   1    1     1.0  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710007      -7  004  1   0    *  CF-004-Valu-0000
710007      -7  005  1   0    *  CF-005-Valu-0000
*
*
705008   CF-008 = y2'
700008   1    1     1.0  0.0  *  Type, Scale, Add
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710008      -7  006  1   0    *  CF-006-Valu-0000
*
*
705009   CF-009 = y3'
700009   1    1     1.0  0.0  *  Type, Scale, Add
```

```
*      Arguments
* Pointers: (1) (2) (3) (4)   *
710009      -7  004  1   0    *  CF-004-Valu-0000
710009      -7  005  1   0    *  CF-005-Valu-0000
710009      -7  006  1   0    *  CF-006-Valu-0000
```

17 Mathematical Library

While the Tabular and Control Function Packages can be considered as a utility packages for the user, Mathematical Library (ML) is a utility package for the program. The ML Package provides a set of general procedures, used by the physics packages. Most of the procedures present in Math Library were adapted from the "Numerical Recipes in FORTRAN" [1]. The ML Package contains subroutines and functions for the following purposes:

- Interpolation, including linear and cubic (third order) interpolation of functions given by tabular data with either constant or variable spacing of the independent variable. The interpolation routines are widely used by nearly all packages of the program, most extensively by the Fluid Property Package (chapter 3). The interpolating subroutines are described in section 17.1.

- Differentiation and integration of functions given by tabular data, with either constant, or variable spacing of independent variable. Those procedures are being used by the Control Volume Package (chapter 2), to perform overall mass and energy balance check. To perform this check it is necessary to integrate masses and energies from all mass sources and energy sources present in the system, as well as heat fluxes entering CV from the surface of all Solid Heat Conductors. The procedures from this group are described in section 17.2.

- Calculation of real roots of quadratic and cubic equations. The procedures are used by the Control Volume Package (chapter 2), to calculate atmosphere-bubble pressure difference, and the Heat Transfer Package (chapter 7), to calculate shear enhancement in the KSP correlation. The procedures from this group are described in section 17.3.

- Solution of linear equations. The linear equation solvers are used by the Junction Package (chapter 4), the Solid Heat Conductor Packages (chapters 5 and 6), the Thermal Radiation Package (chapter 8), and the Radioactive Particle Transport Package (chapter 12). The procedures from this group are described in section 17.4.

- Bessel functions. Bessel functions are used by the Solid Heat Conductor Package, to model the extended surfaces (chapter 5). The procedures from this group are described in section 17.5.

17.1 Interpolation

Two types of interpolation are available:

- Linear interpolation
- Third order (cubic) interpolation.

With linear interpolation the obtained function is continuous, but it has discontinuous derivative. In case of cubic interpolation both function and its first derivative are continuous (Figure 15-1). Beyond the range defined by the data points, the end point values are kept. This approach is consider as safer than performing extrapolations.

Different interpolating functions are available, applicable for:

- Equal spacing of independent variable data points
- Unequal spacing of independent variable data points.

In case of equal spacing of the data points, such as for example: (1.0, 2.0, 3.0, 4.0, etc.), there is no need to search for the nearest data points, and the interpolation is relatively fast.

In case of unequal spacing, like (1.0, 2.0 2.5, 2.9, 3.0, 11.0, etc.), the nearest data points have to be found before the interpolation can be performed. The search is performed using the bisection method, adapted from [1]. To increase efficiency of calculations, an option is provided to tell the interpolating subroutine the numbers of the nearest data points. In such case the search part is bypassed and calculations are faster. This of course can be done only if the nearest points numbers are known before calling. In the program this occurs quite frequently. For example, when several different properties at the saturation line need to be found for the same temperature argument. All saturation properties are tabulated for the same temperature points. When several properties are calculated for the same temperature then the search is only needed during the first call to interpolating procedures. The calls to interpolating subroutine made to find other parameters are executed with the option to bypass the searching. Tests performed with different interpolating subroutines (Volume 3) show that the relative CPU requirements are:

- Unequal argument spacing, search for nearest points included: 100%
- Unequal argument spacing, search for nearest points bypassed: 20%
- Equal argument spacing, no need to search for nearest points: 40%

17.2 Differentiation and Integration

This section gives a description of the procedures performing differentiation and integration of tabulated functions.

- *Differentiation*

  Derivative of a tabular function is calculated for any argument *x*, as:

$$y' = \frac{y_i - y_{i-1}}{x_i - x_{i-1}}$$

Here x_i, x_{i-1}, are the values of the nearest arguments, such that $x_i < x < x_{i-1}$, and y_i, y_{i-1}, are the function values for those arguments. Outside the tabulated range the end-point values are kept, therefore the derivative is equal to zero: $y' = 0.0$.

- *Integration*

Integrating procedures calculate an integral of a tabular function:

$$I = \int_a^b y(x)dx$$

where $y(x)$ is a function defined through data pairs (x, y). The integration boundaries, a, b, may be any real numbers. Depending on the relative values of the integration boundaries, three cases are considered:

$$\begin{aligned} I &= 0.0 && \text{if} \quad a = b \\ I &= +\int_{a'}^{b'} y(x)dx \quad \text{with } a' = a, b' = b && \text{if} \quad a < b \\ I &= -\int_{a'}^{b'} y(x)dx \quad \text{with } a' = b, b' = a && \text{if} \quad a > b \end{aligned}$$

Before the integral is calculated the nearest data points must be found for both the lower and the upper integration bounds. For the lower integration boundary the nearest data points are denoted by $x_{a'}$, $x_{a'-1}$. In case of the upper boundary the nearest data points are $x_{b'}$, $x_{b'-1}$. The integral is then calculated as a sum of three parts:

$$I = I_1 + I_2 + I_3$$

The values of the integrals I_1, I_2, I_3 are given below:

$$I_1 = \begin{cases} y_1(x_1 - a') & \text{if} \quad a' < x_1 \\ \frac{1}{2}\frac{x_{a'} - a'}{x_{a'} - x_{a'-1}}[(y_{a'} + y_{a'-1})(x_{a'} - x_{a'-1}) + (y_{a'} - y_{a'-1})(a' - x_{a'-1})] & \text{if} \quad a' > x_1 \end{cases}$$

$$I_2 = \frac{1}{2}\sum_{j=a'+1}^{b'-1}(y_j + y_{j-1})(x_j - x_{j-1})$$

$$I_3 = \begin{cases} y_N(b' - x_N) & \text{if} \quad b' > x_N \\ \frac{1}{2}\frac{b' - x_{b'}}{x_{b'} - x_{b'-1}}[(y_{b'} + y_{b'-1})(x_{b'} - x_{b'-1}) + (y_{b'} - y_{b'-1})(x_{b'} - b')] & \text{if} \quad b' < x_N \end{cases}$$

In case when the integration boundaries are within a single data interval, i.e. when $x_{a'} = x_{b'}$, the integral is calculated as:

$$I = \frac{b'-a'}{2} \cdot \left(2y_{a'-1} + \frac{y_{a'} - y_{a'-1}}{x_{a'} - x_{a'-1}} \cdot (a'+b'-2x_{a'-1}) \right)$$

Similarly as in case of interpolating procedures, the procedures performing differentiation and integration of tabulated functions may be divided into two groups, applicable for:

- Equal spacing of independent variable data points
- Unequal spacing of independent variable data points.

The differentiating procedures for unequal independent argument spacing are again provided with an option to bypass a search for the nearest data points, which allows to speed up calculations when several calls are made for the same argument and the same independent variable array, but different dependent variable arrays.

Tests performed with different integrating and differentiating subroutines (see Volume 3) show that the relative CPU requirements are:

- Differentiation
  - Unequal argument spacing, search for nearest points included: 100%
  - Unequal argument spacing, search for nearest points bypassed: 15%
  - Equal argument spacing, no need to search for nearest points: 40%

- Integration
  - Unequal argument spacing: 100%
  - Equal argument spacing: 90%

The integrating subroutines perform a relatively large amount of floating point operations, therefore the gain from avoiding the nearest points searching is minor.

An example of differentiation and integration is presented in Figure 17-1. Figure shows a tabulated function, $y(x)$, its derivative, $y'(x)$, and two integrals:

$$I_1 = \int_{-1}^{x} y(x)dx$$

$$I_2 = \int_{2}^{x} y(x)dx$$

The difference between the values of the first and the second integral is equal to 3.5, which is equal to the value of the integral taken from –1.0 to 2.0.

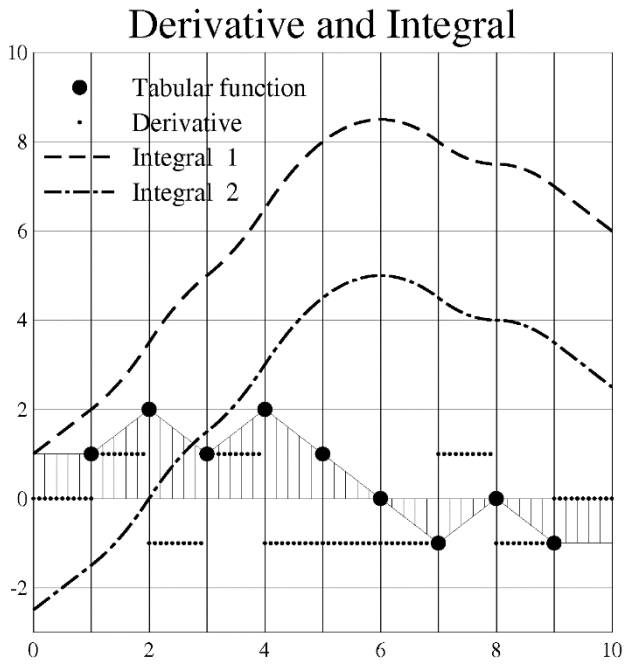


Figure 17-1 Derivative and integral of a tabulated function

17.3 Real Roots of Quadratic and Cubic Equations

The subroutines calculating real roots of quadratic and cubic equations are described below.

- *Quadratic equation*
  A quadratic equation has the following general form:

$$x^2 + ax + b = 0$$

  The discriminant, Δ, is calculated from the formula:

$$\Delta = a^2 + 4b$$

  Depending on the value of Δ two cases are possible:

  Δ < 0.0 - no real roots
  If the value of the discriminant is negative, then the equation has no real roots. The values of roots x1, x2 are in this case set to:

$$x_1 = -10^{99}$$
$$x_2 = +10^{99}$$

$\Delta \geq 0.0$ - two real roots. If the value of the discriminant is non-negative then the equation has two real roots. The values of the two roots become identical when $\Delta = 0.0$. If the absolute value of Δ is smaller than 10^{-15} it is interpreted as zero. The real roots are equal to:

$$x_1 = \frac{1}{2} \cdot (-a - \sqrt{\Delta})$$

$$x_2 = \frac{1}{2} \cdot (-a + \sqrt{\Delta})$$

With the above formulation the following condition is always fulfilled: $x_1 < x_2$. This fact is sometimes helpful in selecting this one out of the two roots which has a physical meaning.

- *Cubic equation*
  A cubic equation has the following general form:

$$x^3 + ax^2 + bx + c = 0$$

Calculation is performed using the Viète's formulae:

$$p = \frac{b}{3} - \frac{a^2}{9}$$

$$q = \frac{a^3}{27} - \frac{ab}{6} + \frac{c}{2}$$

$$Q = p^3 + q^3$$

Depending on the value of Q two cases are possible:

$Q < 0.0$ - three real roots
If the value of the p is negative, then:

$$\alpha = \frac{1}{3} \cdot \arccos\left(-\frac{q}{\sqrt{p^3}}\right)$$

and:

$$x_1 = -2\sqrt{-p} \cdot \cos\left(\alpha - \frac{\pi}{3}\right) - \frac{a}{3}$$

$$x_2 = -2\sqrt{-p} \cdot \cos\left(\alpha + \frac{\pi}{3}\right) - \frac{a}{3}$$

$$x_3 = \quad 2\sqrt{-p} \cdot \cos(\alpha) - \frac{a}{3}$$

Such case is shown in Figure 17-2. (a). In case when $p^{1/3} = -q$, the expression under arccos is equal to 1.0, which means that $\alpha = 0.0$ and $x_1 = x_2$ (which follows from parity of the cosine function). This case is shown in Figure 17-2 (d). Similarly when $\alpha = \pi/3$, then again a double root is obtained; this time the second and the third roots are equal: $x_2 = x_3$.

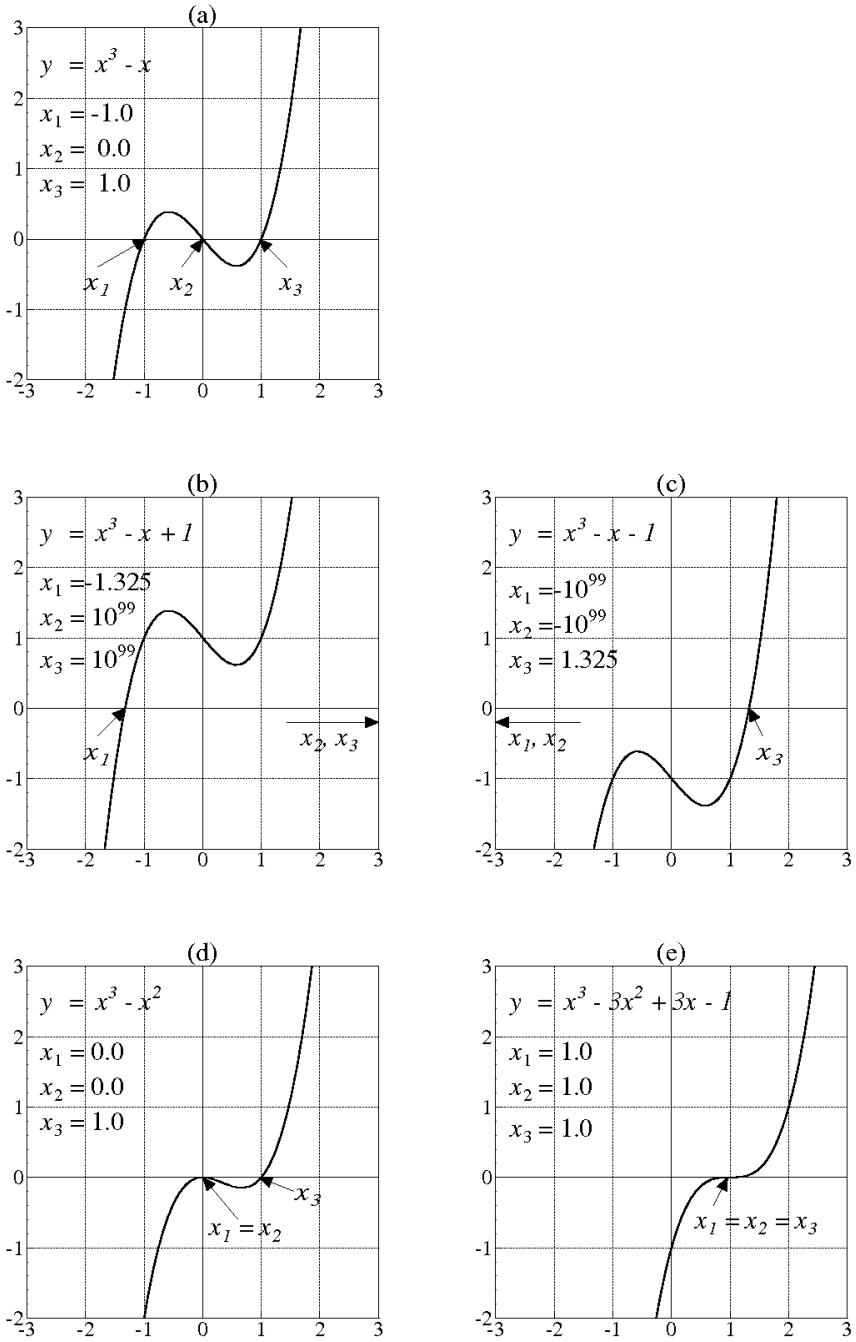


Figure 17-2 Examples of cubic function, real roots calculation.

If the value of Q is equal to zero (when the absolute value of Q smaller than 10^{-15} it is interpreted by the procedure as zero), then the equation has a triple real root:

$$x_1 = x_2 = x_3 = -\frac{a}{3}$$

This case is shown in Figure 17-2 (e). Note that with Q ≤ 0.0, p^3 can never be positive, thus the cases described above exhaust all possibilities.

Q > 0.0 - one real root

$$A_0 = sign(-q + \sqrt{Q}) \cdot \left| -q + \sqrt{Q} \right|^{1/3}$$

$$B_0 = sign(-q + \sqrt{Q}) \cdot \left| -q - \sqrt{Q} \right|^{1/3}$$

$$Y = A_0 + B_0$$

Two cases are distinguished. If the value of Y is positive then there is a single real root, placed on the right arm of the parabola, as shown in Figure 17-2 (c). In that case the following values are returned by the cubic root subroutine:

$$x_1 = -10^{99}$$

$$x_2 = -10^{99}$$

$$x_3 = Y - \frac{a}{3}$$

In case of negative Y the root is placed on the left arm of the parabola, as shown in the Figure 17-2 (b). The returned values are:

$$x_1 = Y - \frac{a}{3}$$

$$x_2 = +10^{99}$$

$$x_3 = +10^{99}$$

With the above formulation the following condition is always fulfilled: $x_1 < x_2 < x_3$. This fact is sometimes helpful in selecting this one out of the three roots which has a physical meaning.

17.4 Solution of Linear Equation Sets (Matrix Solvers)

The basic methods of solving linear algebraic equations include the Gauss-Jordan elimination method, and the LU (lower-upper) decomposition method. For extremely large equation sets iterative methods may be used [1]. According to reference [1], the most efficient method for solving linear equation sets is the LU decomposition. However, it was observed that this method failed to find proper solution in solving some decay chains(*).

The following matrix solvers are available:

- Gauss-Jordan method
  The method is generally applicable for all kind of matrices. The subroutine performing that task was adapted from [1]. The modifications made to the original subroutines include:
  - declaring all real numbers as double precision (to be consistent with the rest of the SPECTRA code, which has been written totally in double precision);
  - adding a failure indicator to allow failure handling by a method generally applied in the code (failure indicator allows to avoid immediate termination on failure, and therefore allows the program to provide the user with extensive diagnostics on where and why the error has occurred).

- The LU decomposition method
  The method is generally applicable for all kind of matrices. The subroutines performing that task were adapted from [1]. The modifications made to the original subroutines include:
  - declaring all real numbers as double precision (to be consistent with the rest of the SPECTRA code, which has been written totally in double precision);
  - adding a failure indicator to allow failure handling by a method generally applied in the code (failure indicator allows to avoid immediate termination on failure, and therefore allows the program to provide the user with extensive diagnostics on where and why the error has occurred);
  - extending the maximum array dimension (maximum number of equations being solved simultaneously) from 500 to 1000.

- Tridiagonal matrix solver
  Tridiagonal matrices are obtained from a 1-dimensional conduction equation (section 5.2). Such matrices are relatively easy to solve, and a special procedure exists which allows to solve them efficiently. The procedure was adapted from [1]. The modifications made to the original subroutines include declaring all real numbers as double precision.

(*) The LU decomposition subroutine was returning non-zero values for isotopes that were not present. Since the values were relatively small, this effect is believed to be caused by round-off errors. Nevertheless the non-zero values caused numerical problems, specifically when the values were negative. When Gauss-Jordan was applied the values were exactly zero.

- Band-diagonal matrix solvers
  Procedures specifically designed for band diagonal matrices, such as for example obtained from multi-dimensional conduction equation (section 6.2). The solution procedure was adapted from [1]. The modifications made to the original subroutines include:

  - declaring all real numbers as double precision;
  - adding a failure indicator.

  Since multidimensional heat conduction is currently not available, the procedures are not used by the code. They were included into the Math Library for eventual future use.

- Sparse matrix solver
  Bi-conjugate gradient method is an iterative solution procedure, specifically designed for large sparse matrices, such as for example obtained in a large flow networks (see section 4.2). The solution procedure was adapted from [1]. The modifications made to the original subroutines include adding a failure indicator.

In addition to the matrix solvers described above, this part of the Math Library contains procedures to calculate the norm of a vector, multiply sparse matrix, or its transpose by a vector, etc.

17.5 Bessel Functions

The procedures calculating Bessel functions: $J_n(x)$, $Y_n(x)$, and modified Bessel functions: $I_n(x)$, $K_n(x)$, where $n = 0, 1, ...$, is the order of a given function, were adapted from "Numerical Recipes in FORTRAN" [1]. The modifications made to the original subroutines include converting them into double precision and adding a failure indicator. Those modifications were needed for the reasons explained in the previous section. The values of Bessel functions of the low orders, as calculated by the Mathematical Library Package, are shown in Figure 17-3 and Figure 17-4.

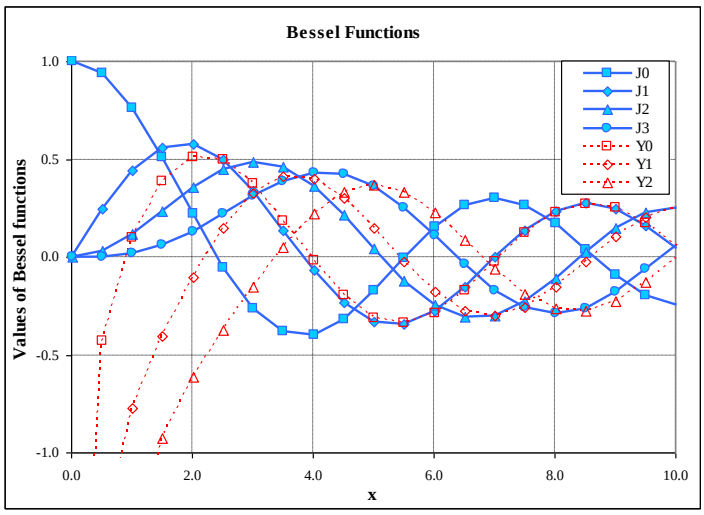


Figure 17-3 Bessel functions

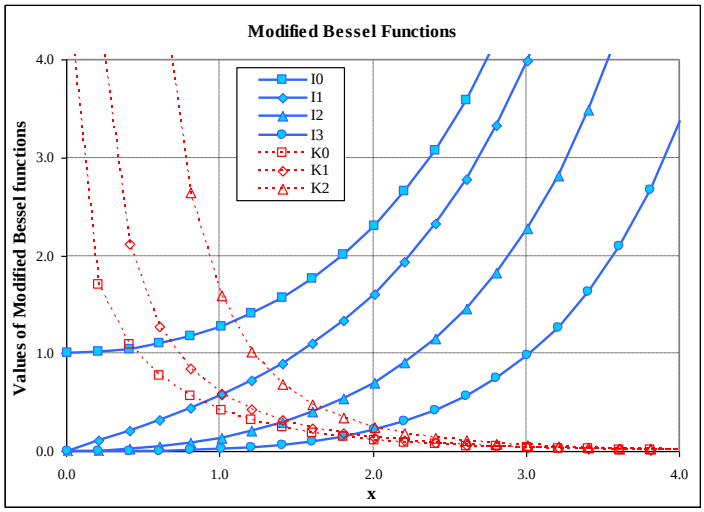


Figure 17-4 Modified Bessel functions

17.6 Utility Functions

The function DNLEXP calculates exp(x) using the FORTRAN double precision function DEXP(X). A separate function was built to calculate the exponent for the reason of computational efficiency, as explained below.

For very large negative arguments the value of function is very close to zero:

$$\exp(x) \to 0.0 \quad for: \quad x \to -\infty$$

The function DEXP(X) takes quite some time to calculate the value for $x \to -\infty$. In SPECTRA calculations exp($-\infty$) are quite frequent, for example in case of particle resuspension the characteristic exponent for weakly-bound particles are very large (resulting in immediate resuspension). In such case FORTRAN takes quite some time in order to obtain the value which is equal to zero with a very good accuracy. A similar problem arises for a very small absolute arguments:

$$\exp(x) \to 1.0 \quad for: \quad x \to 0.0$$

Again, the solution is known with a very good accuracy while FORTRAN takes quite some time to arrive at a value which is just slightly different that one.

The SPECTRA exponent function, DNLEXP(X), has been defined as follows:

$$DNLEXP(x) = \begin{cases} 1.0 & for: \quad |x| < 10^{-5} \\ 0.0 & for: \quad x < -60.0 \\ 1.426 \times 10^{26} & for: \quad x > +60.0 \\ DEXP(x) & for: \quad other\ x \end{cases}$$

The performed CPU tests (see Volume 3) showed that for $x \to -\infty$ ($x = -10^9$) the gain in speed is three orders of magnitude. For $|x| \to 0.0$ ($x = 10^{-9}$) the gain is much smaller but it is still about an order of magnitude. Note that for $x = -10^9$ DEXP takes about 60 times more time than for a "decent" value of $x = 1.0$.

```
                                          TOTAL CPU    CPU / TEST      %

  DEXP( 1.00E-09) = 1.00000E+00 :  CPU =  4.68750E-01  4.68750E-08   100.0
 DNLEXP( 1.00E-09) = 1.00000E+00 :  CPU =  6.25000E-02  6.25000E-09    13.3

  DEXP(-1.00E+09) = 0.00000E+00 :  CPU =  6.26992E+01  6.26992E-06   100.0
 DNLEXP(-1.00E+09) = 0.00000E+00 :  CPU =  6.25000E-02  6.25000E-09     0.1

  DEXP( 6.00E+01) = 1.14315E+26 :  CPU =  1.03125E+00  1.03125E-07   100.0
 DNLEXP( 6.00E+01) = 1.14200E+26 :  CPU =  7.81250E-02  7.81250E-09     7.6

  DEXP( 1.00E+00) = 2.71828E+00 :  CPU =  1.00000E+00  1.00000E-07   100.0
 DNLEXP( 1.00E+00) = 2.71828E+00 :  CPU =  1.15625E+00  1.15625E-07   115.6
```

18 External Data Files

The External Data File (EDF) Package serves as a utility in SPECTRA to communicate with external data files. SPECTRA has two external data files:

- "Write-EDF", and
- "Read-EDF"

The EDF written by SPECTRA may be used by another program as a source of data (e.g. as boundary conditions, etc.). The EDF read by SPECTRA may be used to supply data (e.g. boundary conditions, etc.) to SPECTRA. Such data could be generated by another program, spreadsheet, hand-calculations, etc.

Using the "Write-EDF" a set of variables from the SPECTRA data base may be sent to the EDF. Any variable from the SPECTRA data base may be used, for example Control Volume temperature or pressure, 1-D or 2-D structure temperature, etc. The full list of SPECTRA variables is provided in Volume 2.

Using the "Read-EDF" a set of Tabular Function values may be read by SPECTRA. The reason why only Tabular Functions are read is the following. Other variables, such as for example Control Volume temperature or pressure are being calculated by the SOLVER Package, and they cannot be redefined outside the SOLVER. If a user wishes to use a Control Volume with fixed, or time dependent pressure, then he has to use an inactive CV with time dependent parameter definitions (see Volume 2). In such case the CV temperature, pressure, etc. are defined using either a Tabular or a Control Function. Therefore reading TF values is sufficient to provide every kind of boundary condition that the SPECTRA user may wish to define in his calculations.

There main options of using EDF are:

- Normal run with EDF
- Synchronized run with EDF, explicit coupling
- Synchronized run with EDF, implicit coupling

These two options are shortly discussed below.

18.1 Normal Run with EDF

- Write-EDF is created during the calculations. It is very similar to the plot file; the main difference is the lack of variable identifiers (headers) in the EDF file.
- Read-EDF must be present when SPECTRA is started, and the values from the Read-EDF are read during the run and assigned to the appropriate tabular functions.

At the end of calculations, both EDF files contain full information from the run.

18.2 Synchronized Run with EDF, Explicit Coupling

Synchronized option is provided to allow interactively read/write EDF. With this option SPECTRA may run parallel with another program (or another model run by the SPECTRA code itself) and exchange information at every time step. In case of synchronized run each EDF contains data for a single (current) time step only. The following procedure is used.

Once SPECTRA comes to the point when EDF must be written, it first checks if the EDF that has been written at the previous step still exists. If it does, then it waits until it disappears. Thus, the non-existence of the file is treated as a sign from the other program that the information has been successfully received and the other program is ready to receive new information. Similarly, once SPECTRA comes to the point when EDF must be read, it first checks if it exists. If so, it reads the information and then deletes the file. Thus it gives the signal to the other program that the information has been successfully received and SPECTRA is ready to receive new information.

During both normal run and synchronized run the EDF information is written/read using the data exchange time step DTEXED, specified in the records 9800000 (see Volume 2). This time step must correspond to the EDF time step which, in whatever way, is defined in the other program, running in parallel in SPECTRA. The user also defines maximum acceptable time mismatch in synchronized runs, ERRSED. This is a fraction of the data exchange time step. If the time mismatch is smaller than the product: ERRSED × DTEXED, then the data is exchanged without any time step modification. If the mismatch is larger, the time step is reduced (Figure 17-4). A small increase of time step (10%) is also permitted, in order to avoid severe Δt reduction in the subsequent step.

Multiple data exchange processes (EDF-s) may be specified. In this way data may be read from or written to several data files in a normal run. In a synchronized run, a single job may communicate with several other jobs. Maximum number of EDF processes is 10, therefore one job may exchange data interactively with up to 10 other jobs.

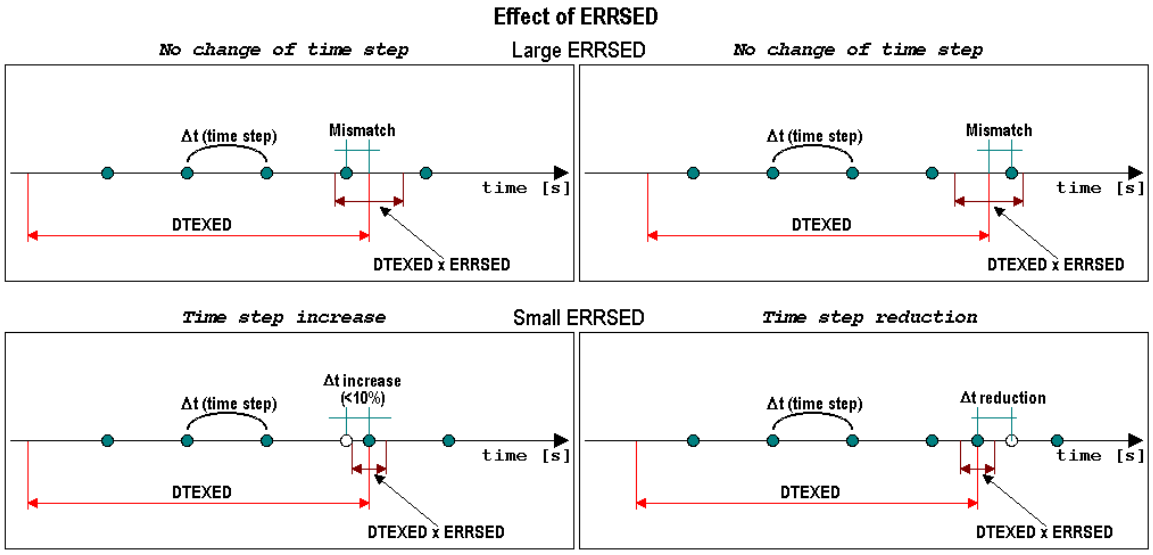


Figure 18-1 Illustration of data exchange process

18.3 Synchronized Run with EDF, Implicit Coupling

In the case of implicit coupling, data is exchanged multiple times every time step. Sub-cycling (time step reduction) is not possible. The minimum time step must be set as equal to the maximum time step. Below, a simple test case illustrates the difference between the implicit and the explicit coupling. The test SYNCH-IMPL-1 consists of two identical inputs, containing one Control Volume (CV-004), with parameters controlled by the tabular functions in such a way that the pressure increases from 1.0 bar to 1.1 bar and the temperature increases from 300 to 400 K in 10 s. The test is described in more detail in Volume 3. The results are shown in Figure 18-2 and Figure 18-3.

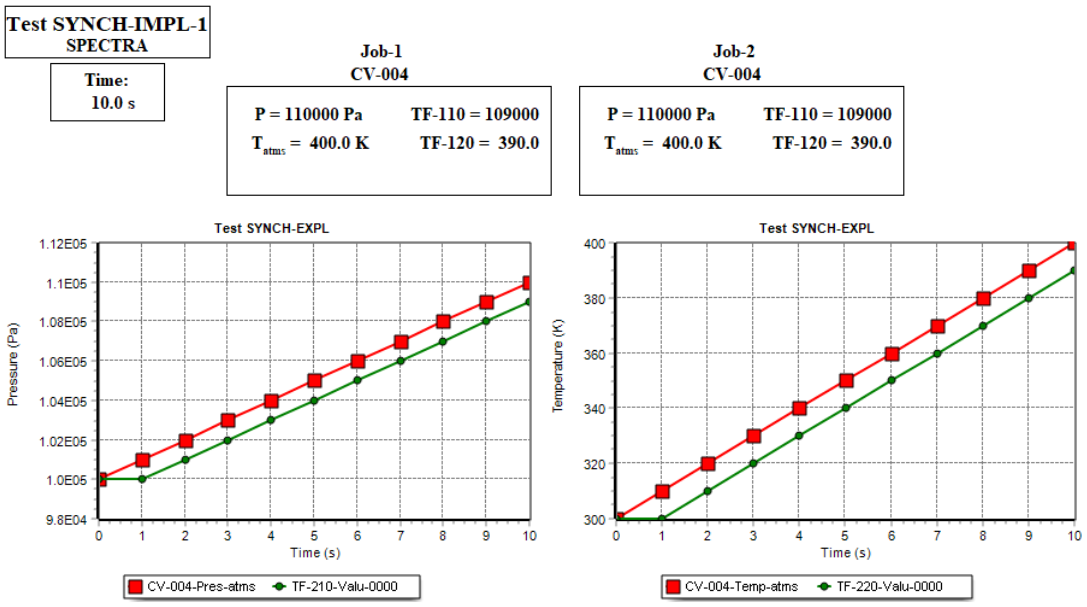


Figure 18-2 Test SYNCH-IMPL-1, explicit coupling; the read data is lagging by one time step

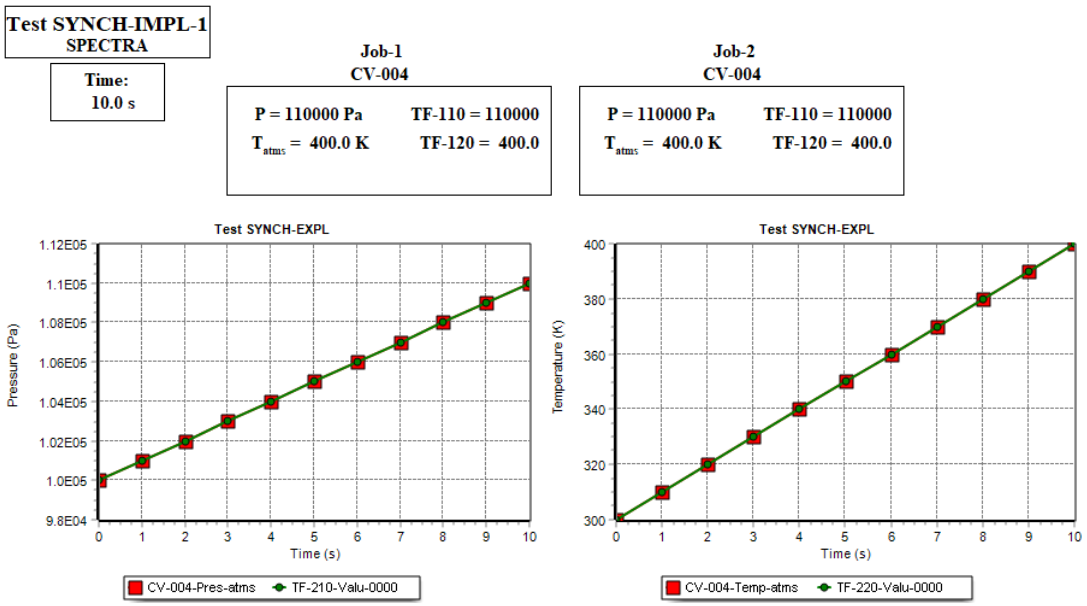


Figure 18-3 Test SYNCH-IMPL-1, implicit coupling

The temperature and pressure is passed from each job on to its partner, first using explicit scheme and then using implicit scheme. The data read is stored as TF-110 (pressure) and TF-120 (temperature).

Figure 18-2 shows that with explicit coupling, the values read are lagging by one time step (to be more precise by one data exchange step, but in this case the data exchange step is equal to the maximum as well as the minimum time step).

Figure 18-3 shows that with implicit coupling the read values are identical as the current values of pressure and temperature, as the data is exchanged in several iterations until sufficient convergence is reached. Printouts of messages from both explicit and implicit couplings are shown below.

- Explicit coupling, EDF messages from the last two time steps

```
 t [s] =  8.0000     :  EDF EXPLICIT DATA TRANSFER NO.:       9
                        DESIGNATED TRANSFER TIME:  8.00000E+00 ,  MISMATCH:  0.00
WRITTEN:    2  8.000E+00  1.080E+05  3.800E+02
READ   :    2  8.000E+00  1.080E+05  3.800E+02

 t [s] =  9.0000     :  EDF EXPLICIT DATA TRANSFER NO.:      10
                        DESIGNATED TRANSFER TIME:  9.00000E+00 ,  MISMATCH:  0.00
WRITTEN:    2  9.000E+00  1.090E+05  3.900E+02
READ   :    2  9.000E+00  1.090E+05  3.900E+02
```

- Implicit coupling, XIMPED = 1.0, EDF messages from the last two time steps

```
 t [s] =  8.0000     :  EDF EXPLICIT DATA TRANSFER NO.:       9
                        DESIGNATED TRANSFER TIME:  8.00000E+00 ,  MISMATCH:  0.00
WRITTEN:    2  8.000E+00  1.080E+05  3.800E+02
READ   :    2  8.000E+00  1.080E+05  3.800E+02

 t [s] =  9.0000     :  EDF IMPLICIT DATA TRANSFER NO.  1
WRITTEN:   -2  9.000E+00  1.090E+05  3.900E+02
READ   :   -2  9.000E+00  1.090E+05  3.900E+02

 t [s] =  9.0000     :  EDF IMPLICIT DATA TRANSFER NO.  2
WRITTEN:    2  9.000E+00  1.090E+05  3.900E+02
READ   :    2  9.000E+00  1.090E+05  3.900E+02

 t [s] =  9.0000     :  EDF EXPLICIT DATA TRANSFER NO.:      10
                        DESIGNATED TRANSFER TIME:  9.00000E+00 ,  MISMATCH:  0.00
WRITTEN:    2  9.000E+00  1.090E+05  3.900E+02
READ   :    2  9.000E+00  1.090E+05  3.900E+02

 t [s] =  10.000     :  EDF IMPLICIT DATA TRANSFER NO.  1
WRITTEN:   -2  1.000E+01  1.100E+05  4.000E+02
READ   :   -2  1.000E+01  1.100E+05  4.000E+02

 t [s] =  10.000     :  EDF IMPLICIT DATA TRANSFER NO.  2
WRITTEN:    2  1.000E+01  1.100E+05  4.000E+02
READ   :    2  1.000E+01  1.100E+05  4.000E+02
```

A semi-implicit scheme is possible, with the user-defined implicit factor (XIMPED - see Volume 2). The actual value that is written to the EDF is equal to:

$$V_{EDF} = V \times \text{XIMPED} + V_0 \times (1.0 - \text{XIMPED})$$

Here V is the current (end-of-time step) value and V_0 is the start of time value. The value of XIMPED = 0.0 will produce basically the same results as explicit coupling (small differences are possible, since with implicit coupling more iterations are needed to accept the solution). If a negative value is entered, the general Solver procedure (Chapter 19) is used to control the convergence of the parameter being sent to the Write-EDF. Further discussion and examples are shown in Volume 3, test cases SYNCH-IMPL.

19 Numerical Solver

The SPECTRA code is solving a large number of differential and algebraic equations. The Numerical Solver Package, referred to shortly as the Solver, is responsible for solving all equations of the model using a stable implicit numerical scheme. A general description of the implicit solution scheme is given below, followed by a short description of the features specific to the SPECTRA solver.

What does it mean in practice, that the Solver Package is finding an implicit solution of the analyzed system? Basically, the implicit formulation can be explained as follows. Suppose one seeks a solution of a system of ordinary differential equations, of the form:

$$\frac{dy_i}{dt} = f(y_1, y_2, ..., y_N, t)$$

For numerical solution the derivative on the left hand side of the equation is replaced by the finite difference approximation:

$$\frac{dy_i}{dt} \approx \frac{y_i - \bar{y}_i}{\Delta t}$$

where $\bar{y}_i$ is the value of the *i*-th function at the beginning of the time step, y_i is its value at the end of the time step, and Δt is the time step size. The finite difference version of the equation set may be constructed in several different ways. The easiest way to write it as:

$$y_i = \bar{y}_i + \Delta t \cdot f(\bar{y}_1, \bar{y}_2, ..., \bar{y}_N, t)$$

In the above formulation, the right hand side of the equations depends only on known, old time step values of all parameters, thus the equation set above provides explicit formulas for all y_i. Another way is to use new time step values, and thus obtain:

$$y_i = \bar{y}_i + \Delta t \cdot f(y_1, y_2, ..., y_N, t)$$

The right hand sides of the above set contain the unknown, end of time step values, y_i, so the equation set provides an implicit formulas for y_i. Other approaches are possible, for example Crank-Nicholson method, with:

$$y_i = \bar{y}_i + \Delta t \cdot f\left(\frac{y_1 + \bar{y}_1}{2}, \frac{y_2 + \bar{y}_2}{2}, ..., \frac{y_N + \bar{y}_N}{2}, t\right)$$

If the implicit or the Crank-Nicholson schemes are chosen, then the formulae have still to be somehow solved to obtain the unknown values of y_i. If the functions *f* are relatively simple, for example if the set consists of linear equations, then it can be solved rather easily, by one of the linear equation set solving procedures (matrix solvers).

For example, when a heat conduction equation is written using an implicit scheme, a matrix with unknown temperatures is obtained (sections 5.2 6.2). Such matrix is easily solved by an appropriate matrix solver (section 17.4). In fact the conduction equation described in section 5.2 or 6.2 is not exactly linear, since the material properties are temperature dependent. Therefore a short internal iteration is needed to solve the conduction equation, but it is typically quickly converging, since the material properties do not change drastically with temperature.

The fact that the conduction equation itself can easily be solved implicitly is only a minor encouragement, in view of the fact that SPECTRA contains a large amount of models, which all interact with each other. For example, the conduction model interacts with Control Volumes through boundary conditions. To solve the conduction equation implicitly, one needs to know the end of time step values of fluid temperatures in the control volumes. Those in turn have to be evaluated by solving mass and energy balance equations. Those equations too are defined implicitly, so among others they depend on the end of time step values of the surface temperature of the heat conductors. Apart from the packages dealing with physics (CV, JN, SC, etc.), there is the Control Function package, which also interacts with other packages, and which has to be solved implicitly. In other words the problem becomes complicated, when one tries to solve all system equations implicitly, and not only one particular package.

When the equations of the model become very complex, then an iterative solution must be used. Typically the iterative solutions are performed as follows. As a first step the values needed to calculate the right hand sides are guessed. Typically the old time step values are used, so that for the first step one assumes that:

$$y_i = \overline{y_i}$$

Next the equation system is solved. The obtained values are somehow being used to calculate the right hand side terms in the next iteration. The easiest way is to use the calculated values directly for the new iteration. This may lead to numerical catastrophes in case when equations are "stiff". For example, assume that atmosphere temperature is close to saturation and there is a negative energy source in the control volume. The gas temperature calculated in the first iteration decreases below saturation. This temperature is taken for the next iteration. Because subcooled steam is present, the non-equilibrium condensation occurs. This process is typically rapid and results in releasing large amounts of heat. Therefore the temperature calculated in the next iteration will be very large. If this temperature is then taken for the next iteration, then a large radiation flux will be emitted from the gas. Consequently the calculated gas temperature will be very low, possibly negative so outside physical limits.

To prevent such numerical catastrophes different techniques may be applied. An underrelaxation method may be used, in which the newly calculated value is taken for the next iteration with some weighting factor. As may be easily found out, this doesn't help at all in case of really stiff equations. To avoid the numerical catastrophes the underrelaxation factor would have to be kept very low. This would imply that in general a great number of iterations would be needed to obtain convergence.

The equations which are being solved by SPECTRA are particularly stiff. This is a result of mechanistic treatment of bubbles and droplets, and nonequilibrium assumptions. The bubbles heat and mass transfer introduce particularly stiff equations, since bubbles themselves have practically no heat capacity, and the heat and mass fluxes are very large. Therefore a specific solution procedure has been developed, and coded as the Solver Package. The SPECTRA Solver is a general procedure, specifically useful to solve stiff sets of equations.

In short, the approach to numerical solution, taken in SPECTRA, is that all packages are internally written, as far as possible, using an implicit scheme (for example, conductor internal temperatures are calculated using an implicit conduction equation). The parameters interfacing several packages, such as wall surface temperatures, fluid temperatures, pressures, velocities, values of control functions, etc., are controlled by the Solver in the main iteration to obtain fully implicit solution of all parameters. The iteration is continued until all controlled parameters converge. The convergence criterion is set by default at 10^{-4}, that means the iteration continues until the differences between the assumed value and the calculated value is smaller than 10^{-4} in relative terms, for all controlled parameters.

The solution procedure is shortly characterized as follows. A time advancement is attempted using the maximum (user defined) time step. If the desired convergence (10^{-4} relative error) cannot be achieved within certain number of iterations (default=50), then the time step is reduced by a certain factor (default=4), and the solution procedure is repeated. Once the solution is obtained, the program proceeds with the next time advancement. If the time step used to complete the previous advancement was shorter than the maximum time step, then the Solver may increase the time step by a certain factor (default=1.303), to reach back the maximum time step. This is done only if the previous advancement has been completed in less than a certain number of iterations (default=20). Several additional time step limits are imposed, like the Courant limit, or velocity change limit. Those are described in chapter 4.

The Solver package provides an easy to use and general tool for solution of differential and algebraic equations. A more detailed description of the procedure applied within the Solver to reach convergence is given in Volume 4. The source code of the Solver is considered proprietary and therefore the source as well as a full description may be provided only following a separate license agreement. Here two simple examples are provided, to illustrate the usefulness of the solution procedure.

The first example is a "stiff equation system", provided in [1]. This system is solved using three different methods: a general purpose ODE solver, a specialized stiff ODE solver, and the SPECTRA Solver (for this purpose the equations were modeled using Control Functions). This example is shown in section 19.1.

The next example is a simple problem involving evaporation from droplets. This example is shown in section 19.2. As mentioned above, bubbles and droplets introduce particularly stiff equations into the model. Here a droplet test case is analyzed, for which the analytical solution is available in literature.

19.1 Example Problem 1 - Stiff Set of Ordinary Differential Equations (ODE)

A "stiff" set of differential equation was selected, taken from [1]. Stiff equation sets are particularly difficult to solve. The equation set is:

$$(1) \quad \frac{dy_1}{dt} = -0.013 \cdot y_1 - 1000 \cdot y_1 \cdot y_3$$

$$(2) \quad \frac{dy_2}{dt} = -2500 \cdot y_2 \cdot y_3$$

$$(3) \quad \frac{dy_3}{dt} = -0.013 \cdot y_1 - 1000 \cdot y_1 \cdot y_3 - 2500 \cdot y_2 \cdot y_3$$

The initial conditions are:

$$y_1(0) = 1$$
$$y_2(0) = 1$$
$$y_3(0) = 0$$

A method to model this equation set in SPECTRA using Control Functions is shown in section 16.6.3. The solution of this test is described in more detail in Volume 3. Here only a short description of the results is given.

The results obtained by the SPECTRA Solver were obtained using the time step of 1.0 (s). As shown in Volume 3, the functions had to be evaluated about 120 times. For comparison, the 5-th order Runge-Kutta method required about 50,000 evaluations of the functions, while the Rosenbrock method specifically recommended for stiff equation sets (see [1] section 16.6) required about the same (120 evaluations) - see Volume 3.

Results are compared to the results of the fourth order Rosenbrock method in Figure 19-1 (see also Figure 16-12). Here the Rosenbrock method is referred to as the "theoretical" solution, although it is still a numerical solution. It is seen that SPECTRA results are very accurate.

The SPECTRA Solver was much more effective than the Runge-Kutta method. It turned out to be competitive even with the Rosenbrock method, developed specifically for stiff systems. The Rosenbrock method is not applicable for relatively complex problems, like that solved by SPECTRA, because of the need to calculate the Jacobian (see Volume 3).

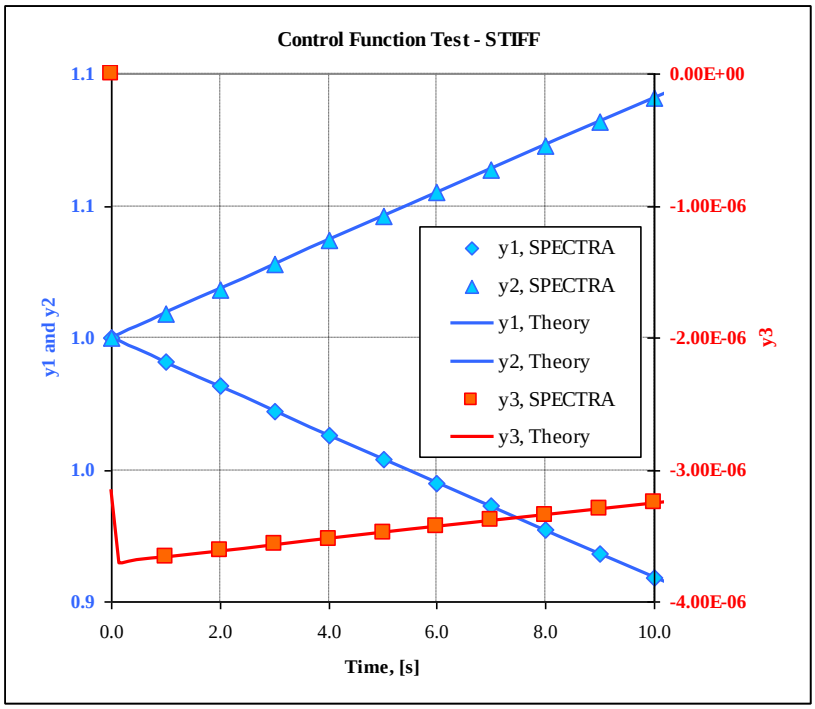


Figure 19-1 Stiff equation set - SPECTRA with Δt=1.0 s and Rosenbrock “theoretical” solution.

19.2 Example Problem 2 - Evaporation of a Droplet

An example problem of droplet evaporation is presented in [21] (example 9.10). A 50 µm diameter water droplet, initially at 315 K, is injected into an air stream at 315 K, 1.050×10^5 Pa, and 50.5% relative humidity. Reference [21] provides an estimation of several important parameters, like droplet lifetime, quasi-equilibrium droplet temperature, parameters of the initial temperature transient.

The analytical equations and the calculation procedure are presented in detail in [21]. The final results are:

- Droplet lifetime: 2.72 s
- Droplet temperature: 305 K
- Average parameters during the temperature transient (during the temperature transient a quasi stationary temperature is reached, which allows the convective heat flux into the droplet to balance the heat flux required for the evaporation):
  - Evaporation heat flux: +39,100 W/m^2
  - Average heat flux: –5,480 W/m^2
  - Duration: ~10^{-2} s,

SPECTRA calculation was performed using a model consisting of a single Control Volume, containing atmosphere and droplets at required initial conditions. The initial droplet diameter was set to 5×10^{-5} m.

The initial amount of droplets (droplet fraction) was set to 5×10^{-8}. This was done by trial and error method to achieve two goals. On one hand the droplet fraction must be small enough to avoid a significant change of the atmosphere temperature and humidity during the calculation period. On the other hand it must be large enough to avoid converting residual droplets into the CV pool, the action being performed by the numerical Solver at very small droplet fractions $\sim10^{-10}$ in order to improve the computational speed.

Two calculations were performed. One calculation was made using a short time step (10^{-3} s). This run allows to plot how the key parameters evolve in time. The comparison run was performed with a relatively large time step of 0.5 s. No internal reduction of time step was observed during this run.

Results are shown in Figure 19-2 and Figure 19-3. Figure 19-2 shows the temperatures of droplet and atmosphere. During about 0.02 s the droplet temperature decreases from the initial temperature to about 305 K. Thus the droplet-atmosphere temperature difference during the quasi-equilibrium period is about 10 K. This temperature difference results in a convective heat flux from atmosphere into the droplet equal to about 12,000 W/m^2, increasing to about 60,000 W/m^2 at the end of the droplet lifetime. This convective flux is necessary to support the evaporation process, which occurs because the atmosphere is relatively dry. It is seen in Figure 19-3 that quasi-equilibrium conditions, with convective heat balancing nearly exactly the evaporation heat, are reached at about 0.02 s. Note that the convective flux is shown in Figure 19-3 with a reverse sign, in order to have easier comparison of the two fluxes(*).

During the initial temperature transient the evaporation energy flux decrease from about 75,000 W/m^2 to about 12,000 W/m^2, which means that the average value during that period is, roughly speaking, 43,000 W/m^2. The convective flux meanwhile changes from zero to about –12,000 W/m^2, which gives the average value in that period of roughly –6,000 W/m^2.

The droplet evaporates totally after 2.73 s. At the end of that period an increase of the convective heat flux (in terms of absolute value) is observed (Figure 19-3). This is caused by decrease of the characteristic dimension for convection (droplet diameter). As a consequence the heat transfer coefficient and the convective heat flux increase. This increase in convective heat flux allows more intensive evaporation, and the evaporation flux follows the convective heat flux line.

It is seen that all the above parameters are in good agreement with the corresponding theoretical values from [21]. It is also seen that practically the same results are obtained with small and large time steps. Thus, the numerical solver is capable of solving the present problem, involving rather "stiff" equations, using the time step of 0.5 s.

(*) Negative convective heat flux means heat flow from the atmosphere to the droplet. Positive evaporation energy flux means evaporation from the droplet surface, while negative means condensation on the droplet surface.

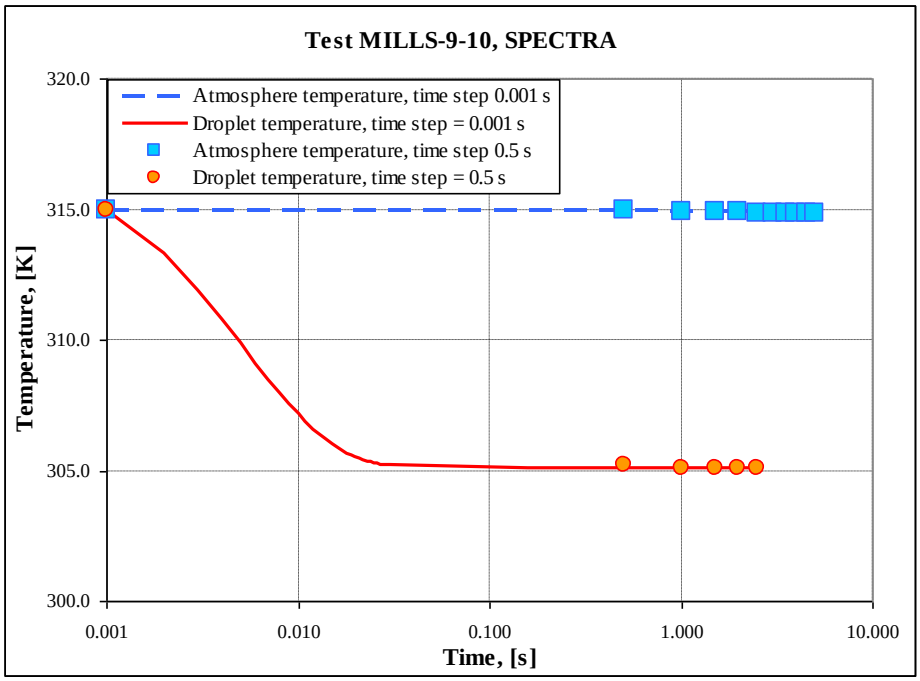


Figure 19-2 Atmosphere and droplet temperature, example problem from [21]

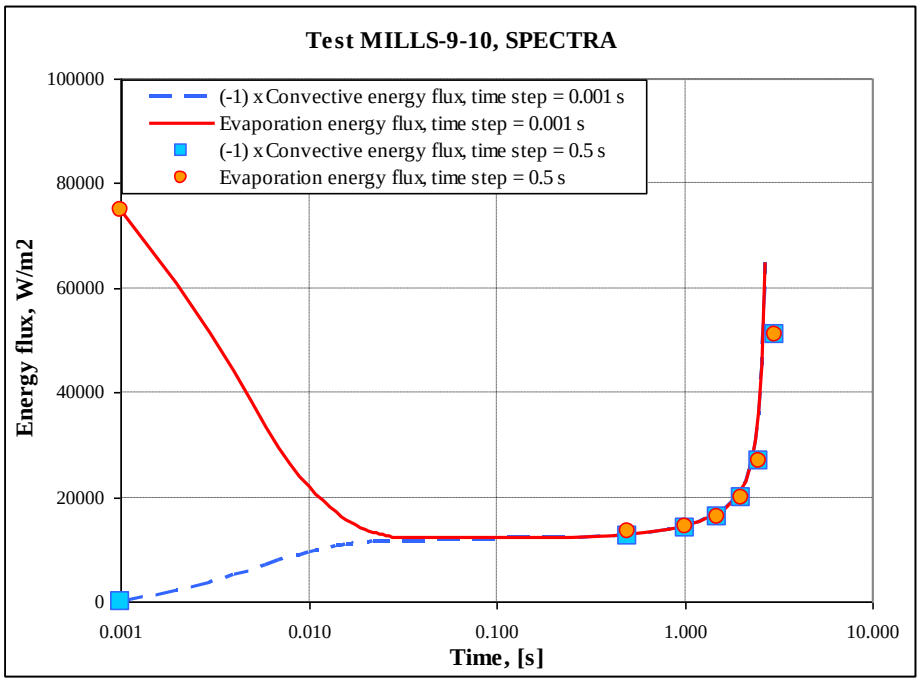


Figure 19-3 Convective heat flux and evaporation heat flux, example problem from [21]

Literature

[1] W.H. Press, S.A. Teukolsky, W.T. Wetterling, B.P. Flannery, "Numerical Recipes in FORTRAN. The Art of Scientific Programming", Second edition, ISBN 0-521-43064-X, Cambridge University Press, 1992.

[2] I.E. Idelchik, "Handbook of Hydraulic Resistance", Second Edition, ISBN 3-540-15962-2, Hemisphere Publishing Corporation, 1986.

[3] E.A. Preager, L.A. Samoilenko, "Investigation of hydraulic resistance of pipelines in the transient mode of flow of liquids and gases", Issled. Vodosnabzhen. Kanalizatsii (Trudy LISI), No. 50, 27-39, Leningrad, 1966.

[4] F. Colebrook, "Turbulent flow in pipes with particular reference to the transition region between the smooth and rough pipe laws", *J. Inst. Civ. Eng.*, No. **4**, pp. 14-25, 1939.

[5] I. Nikuradse, "Laws governing turbulent flows in smooth tubes", *Probl. Turbulentnosti*, **75**, ONTI Press, 1936.

[6] S.E. Dingman, et al., "HECTR Version 1.5 User's Manual", NUREG//CR-4507, SAND86-0101, Sandia National Laboratories, Albuquerque, NM, 1986.

[7] G. Ciccarelli, et al., "High Temperature Hydrogen-Air-Steam Detonation Experiments at BNL Small Scale Development Apparatus", NUREG/CR-6213, BNL-NUREG-52414, August 1994.

[8] D.W. Stamps, M. Berman, "High Temperature Hydrogen Combustion in Reactor Safety Applications", *Nuclear Science and Engineering*, Vol. **109**, pp. 39-48, 1991.

[9] J.E. Hustad, O.K. Sonju, "Experimental Studies of Lower Flammability Limits of Gases and Mixtures of Gases at Elevated Temperatures", *Combustion and Flame*, Vol. **71**, pp. 283-294, 1988.

[10] M.G. Plys, R.D. Astleford, M. Epstein, "A Flammability and Combustion Model for Integrated Accident Analysis", FAI 1988.

[11] V.L. Streeter, E.B. Wylie, "Fluid Mechanics", McGraw Hill, New York, 6-th edition, ISBN 0-07-62193-4, 1975.

[12] W. Breitung, "The Analysis of Hydrogen Behaviour in Severe Accidents", Lecture 11 on Eurocourse on: "Analysis of Severe Accidents in LWR's", 1998.

[13] W. Breitung, S.B. Dorofeev, "Summary of DDT Criteria for Nuclear Power Plants", *Jahrestgung Kerntechnik*, p. 185, 1999.

[14] W. Breitung, et al., "Flame Acceleration and Deflagration-to-Detonation Transition in Nuclear Safety", OECD report, NEA/CSNI/R(2000)7, August 2000.

[15] B. Lewis, G. von Elbe, "Combustion, Flames and Explosions of Gases", Academic Press Inc., ISBN 0-12-446751-2, 1987.

[16] D.D.S. Liu, R. MacFarlane, "Laminar Burning Velocities of Hydrogen-Air-Steam Mixtures", *Combustion and Flame*, Vol. **49**, pp.59-71, 1983.

[17] F.A. Williams, "Combustion Theory - The Fundamental Theory of Chemically Reacting Flow Systems", Second Edition, The Benjamin/Cummings Publishing Company, Inc., ISBN 0-8053-9801-5, 1985.

[18] A.M. Klimov, "Premixed Turbulent Flames - Interplay of Hydrodynamic and Chemical Phenomena", *Flames, Lasers, and Reactive Systems*, Vol. **88**, of *Progress in Astronautics and Aeronautics*, American Institute of Astronautics and Aeronautics, pp. 133 - 146, 1988.

[19] "Nuclear Engineering Handbook", ISBN-13: 978-0070197206, First Edition, Edited by H. Etherington, McGraw Hill, New York, 1958.

[20] W.M. Rohsenow, J.P. Harnett, "Handbook of Heat Transfer", McGraw Hill Book Company, ISBN 0-07-053576-0, New York, 1973.

[21] A.F. Mills, "Heat and Mass Transfer", Richard D. Irwin, Inc., ISBN 0-256-11443-9, Chicago, 1995.

[22] F.W. Dittus, L.M.K. Boelter, "Heat Transfer in Automobile Radiators of the Tubular Type", *Publications in Engineering, University of California, Berkeley*, Vol. **2**, 443, 1930.

[23] W.H. McAdams, "Heat Transmission", ISBN-13: 978-0898748765, Robert E. Krieger Publishing Company, Malabar, Florida, 1985.

[24] J.P. Holman, "Heat Transfer", Fourth edition, ISBN 0-07-029598-0, McGraw Hill, Inc., Tokyo 1976.

[25] C.O. Bennett, J.E. Myers, "Momentum, Heat and Mass Transfer", ISBN-13: 978-0070046719, McGraw Hill, third edition, New York, 1982.

[26] S. Whitaker, "Forced Convection Heat Transfer Correlations for Flow in Pipes, Pas Flat Plates, Single Cylinders, Single Spheres, and Flow in Packed Bids and Tube Bundles", *AIChE J*., Vol. **18**, p.361, 1972.

[27] "Hanbook of Lead-Bismuth Eutectic Alloy and Lead Properties, Materials Compatibility, Thermal-Hydraulics and Technologies", OECD/NEA Nuclear Science Committee, ISBN 978-92-64-99002-9, 2007.

[28] O.E. Dwyer, "Heat Transfer Through Liquid Metals Flowing Turbulently Between Parallel Plates", Nuc. Sci. Eng., Vol. 21, pp. 79-89, 1965.

[29] L. Haar, J.S. Gallagher, G.S. Kell, "NBS/NRC Wasserdampftafeln", ("NBS/NRC Steam Tables"), Springer-Verlag, Berlin, ISBN 3-540-18039-7, Springer-Verlag, New York, ISBN 0-387-18039-7, 1988.

[30] G.H. Thomson, K.R. Brobst, R.W. Hankinson, *AIChE Journal*, **28**, p. 671, 1982.

[31] R.C. Reid, J.M. Prausnitz, B.E. Poling, "The Properties of Gases and Liquids", Fourth Edition, ISBN 0-07-051799-1, McGraw Hill, New York, 1989.

[32] D.R. Lide, et al., "CRC Handbook of Chemistry and Physics, A Ready-Reference Book of Chemical and Physical Data", 72-nd edition, ISBN-0-8493-0472-5, CRC Press Inc., 1991-1992.

[33] C.R. Wilke, *J. Chem. Phys*., **18**, p. 517, 1950.

[34] A. Wassiljewa, *Physik. Z*., **5**, p. 737, 1904.

[35] L.I. Stiel, G.Thodos, *AIChE J*. **10**:26, 1964.

[36] E.A. Mason, S.C. Saxena, *Phys. Fluids*, **1**, p. 361, 1958.

[37] A. Blanck, *J. Physics*, **7**, p. 825, 1908.

[38] E.N. Fuller, P.D. Schettler, J.C. Giddings, *Ind. Eng. Chem.*, **58** (5), p. 18, 1966.

[39] J. Hilsenrath, et al., "Tables of Thermal Properties of Gases", National Bureau of Standards, Washington D.C., National Bureau of Standards Circular 564, November 1, 1955.

[40] T.H. Chung, L.L. Lee, K.E. Starling, "Applications of kinetic gas theories and multiparameter correlation for prediction of dilute gas viscosity and thermal conductivity", *Industrial and Engineering Chemistry Fundam.*, **23**: 8-13, 1984.

[41] K. Lucas, "Phase Equilibria and Fluid Properties in the Chemical Industry", *Proceedings, 2nd International Conference*, Berlin (West), 17-21 March, 1980.

[42] D. Richtenberg, "The Viscosities of Pure gases at High Pressures", *Natl. Eng/. Lab. Rept. Chem. 53*, East Kilbride, Glasgow, Scotland, August 1975.

[43] J.W. Miller, P.N. Shah, G.R. Schorr, P.M. Pattel, "Correlation constants for chemical compounds", *Chem. Eng.*, **83** (25), p. 153-162, 1976.

[44] V.H. Ransom, et al., "RELAP5/MOD2 Code Manual, Volume 1: Code Structure, System Models, and Solution Methods", NUREG/CR-4312, EGG-2396, Revision 1, March 1987.

[45] R.A. Dimenna, J.R. Larson, R.W. Johnson, T.K. Larson, C.S. Miller, J.E. Streit, R,G, Hanson, D.M. Kiser, "RELAP5/MOD2 Models and Correlations", NUREG/CR-5194, EGG-2531, August 1988.

[46] R.O. Gauntt, et.al., "MELCOR Computer Code Manuals, Reference Manuals, Version 1.8.6, September 2005", NUREG/CR-6119, Vol. 2, Rev. 3, SAND 2005-5713, published: September 2005.

[47] "MAAP - Modular Accident Analysis Program", https://www.fauske.com/nuclear/maap-modular-accident-analysis-program

[48] G.B. Wallis, "One-Dimensional Two-Phase Flow", ISBN-13: 978-0070679429, McGraw Hill, Inc., New York, 1969.

[49] G.B. Wallis, Papers No. 69-FE-45, 69-FE-46, *ASME Applied Mech.-Fluids Eng. Conf*., Northwestern University, June 1969.

[50] W.T. Hancox, W.B. Nicoll, "Prediction of Time Dependent Diabatic Two-phase Water Flows", *Progress in Heat and Mass Transfer*, **6**, pp. 119-135, 1972.

[51] J.A. Borkowski, et al., "TRAC-BF1/Mod1 Models and Correlations", NUREG/CR-4391, EGG-2680, R4, August 1992.

[52] S. Levy, "Steam Slip - Theoretical Prediction from Momentum Model", *Journal of Heat Transfer*, **82C** (1), pp. 113-124, 1960.

[53] S. Levy, "Prediction of Two-Phase Critical Flow Rate", ASME 64-HT-8, 1964.

[54] P. Griffith, W.M. Rohsenow, "Two-Phase Flow, Boiling and Condensation", notes for special summer program, Massachusetts Institute of Technology, 1973.

[55] Y.Y. Hsu, R.W. Graham, "Transport Process in Boiling and Two-Phase Systems", ISBN 0-07-030637-0, McGraw Hill, 1976.

[56] R.E. Henry, "A Study of One- and Two-Component, Two-Phase Critical Flows at Low Qualities", ANL-7430, Argonne National Laboratory, March 1968.

[57] K.H. Ardron, R.A. Furness, "A study of the critical flow models used in reactor blowdown analysis", *Nuclear Engineering and Design*, **39**, pp. 257-266, 1976.

[58] F.J. Moody, "Maximum flow rate of a single component two-phase mixture", *Trans. ASME, J. Heat Transfer*, **87C**, p. 134, 1965.

[59] "RELAP-4/MOD-5, A Computer Program for Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Related Systems, User's Manual, Volume 1, RELAP-4/MOD-5 Description", ANCR-NUREG-1335, Idaho Nuclear Engineering Laboratory, Idaho Falls, ID, September, 1976.

[60] R.T. Lahey, F.J. Moody, "The Thermal-Hydraulics of a Boling Water Reactor", ISBN 0-89448-010-3, American Nuclear Society, 1977.

[61] H.K. Fauske, "The discharge of saturated water through tubes", *Chem Eng. Progr. Symp. Ser.* **61**, p. 210, 1965.

[62] R.E. Featherstone, C. Nalluri, "Civil Engineering Hydraulics - Essential Theory with Worked Examples", ISBN-13: 978-0632038633, Granada Publishing Limited, London, 1982.

[63] E. Logan, "Handbook of Turbomachinery", Marcel Dekker Inc., New York, ISBN: 0-8247-9263-7, 1994, ISBN 0-8247-0995-0, 1995.

[64] K. Bemmert, P. Zehner, "Measurements of the Four-Quadrant Characteristics on a Multi-Stage Turbine", *Journal of Engineering for Power, Transactions of ASME*, Vol. **102**, pp. 316-321, April 1980.

[65] B. Leckner, "Spectral and Total Emissivity of Water Vapor and Carbon Dioxide", Combustion and Flame, Vol. 19, pp. 13-48, 1972.

[66] J.J. Duderstad, L.J. Hamilton, "Nuclear Reactor Analysis", ISBN 0-471-22363-8, John Wiley & Sons, Inc., New York, 1976.

[67] A.F. Henry, "Nuclear Reactor Analysis", ISBN 0-262-08081-8, Massachusetts Institute of Technology, Cambridge, 1975.

[68] "Reactor Physics Constants", ISBN-13: 978-0870794971, Argonne National Laboratory, ANL-5800, Second Edition, July 1963.

[69] "Chart of the Nuclides", https://www.nndc.bnl.gov/chart/

[70] NuChart Version 4.0.0, July 1998. Nuclear data from NUDAT data files, release October 1997, produced by National Nuclear Data Center.

[71] C.J. Dean, R.J. Perry, "The 1995 WIMS Nuclear Data Library", AEA Technology, Technical Service Division, AEA-TSD-0630, March 1995.

[72] "RELAP-4/MOD-5, A Computer Program for Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Related Systems, User's Manual, Volume 1, RELAP-4/MOD-5 Description", ANCR-NUREG-1335, Idaho Nuclear Engineering Laboratory, Idaho Falls, ID, September, 1976.

[73] American Nuclear Society Standards Committee Working Group ANS-5.1, "Decay Heat Power in Light Water Reactors", ANSI/ANS-5.1-1979, La Grange Park, IL, 1979.

[74] M.M.R. Williams, S.K. Loyalka, "Aerosol Science, Theory & Practice, With Special Applications to the Nuclear Industry", First Edition, ISBN 0-08-037209-0, Pergamon Press, Oxford 1991.

[75] W.C. Hinds, "Aerosol Technology", Second Edition, ISBN 0-471-19410-7, John Wiley and Sons, Inc., New York, 1999.

[76] P. Lengweiler, "Modelling Deposition and Resuspension of Particles on and from Surfaces", PhD Thesis, Swiss Federal Institute of Technology, Diss. ETH No. 13734, Zürich 2000.

[77] W.B. Kunkel, "Magnitide and character of errors produced by shape factors in Stokes' law estimates of particle radius", *J. Appl. Phys.*, August 1948.

[78] F. Boenio-Brocchieri, et al., "Nuclear Aerosol Codes", *Nuclear Technology*, Vol. **81**, p. 193, May 1988.

[79] J.A. Gieseke, "Analytic Studies of Aerosol Behaviour Predictions for Fast Reactor Safety", BMI-1932, Battelle Memorial Institute, 1975.

[80] B.H. McDonald, "Comparison of the Physical Models and Numerical Methods in the Codes Used GREST LWR Code Comparison Exercise", CSNI report number 116, OECD Nuclear Energy Agency, Paris, 1988.

[81] N.B. Wood, "The mass transfer of particles and acid vapour to cold surfaces.", *Journal of the institute of Energy* **76**, 76-93, 1981.

[82] J.R. Brock, “On the Theory of Thermal Forces Acting on Aerosol Particles”, *J. of Colloid Science*, No. **17**, p.768, 1962.

[83] C. He, G. Ahmadi, “Particle Deposition with Thermophoresis in Laminar and Turbulent Duct Flows”, *Aerosol Science and Technology*, **29**(6), pp. 525-546, 1998.

[84] J.Young and A. Leeming, "A theory of particle deposition in turbulent pipe flow", *Journal of Fluid Mechanics* **340**, 129-159, 1997.

[85] F. Fan and G. Ahmadi, "A sublayer model for turbulent deposition of particles in vertical ducts with smooth and rough surfaces", *Journal of Aerosol Science* **24**(1), 45-64, 1993.

[86] S.K. Kaer, “Numerical investigation of deposit formation in straw-fired boilers – Using CFD as the framework for slagging and fouling predictions”, PhD Thesis, Aalborg University, Institute of Energy Technology, October 2001.

[87] H.J. Allelein, et al., “State-of-the-Art Report on Nuclear Aerosols”, NEA/CSNI/R(2009)5, 17 December, 2009

[88] M.W. Reeks, D. Hall, “Kinetic models for particle resuspension in turbulent flows: theory and measurement”, *Aerosol Science*, **32**, 1-31, 2001.

[89] D.E. Grey, “American Institute of Physics Handbook”, ISBN-13: 978-0070014855, McGraw Hill Book Co., 1963.

[90] M.R. Kuhlman, D.J. Lemicke, R.O Meyer, “CORSOR User’s Manual”, NUREG/CR-4173, BMI-2122, March 1985.

[91] K.Y. Suh, R.J. Hammersley, “Modeling of Fission Product Release and Transport for Severe Fuel Damage Analyses”, *Nuclear Science and Engineering*, Vol. **109**, pp.26-38, 1991.

[92] J.R. Lamarsh, A.J. Baratta, "Introduction to Nuclear Engineering", Prentice Hall, Upper Saddle River, New Jersey 07458, ISBN 0-201-82498-1, 2001.

[93] A. Fromentin, “Particle Resuspension from a Multi-Layer Deposit by Turbulent Flow”, PhD Thesis, DISS ETH Nr. 8971, 1989.

[94] S. Paci, N. Forgione, F. Parozzi, M.T. Porfiri, “Bases for dust mobilization modeling in the light of STARDUST experiments”, *Nuclear Engineering and Design*, Vol. **235**, pp. 1129 – 1138, 2005.

[95] J.V. Cathcart, "Quarterly Progress Report on the Zirconium Metal-Water Oxidation Kinetics Program, Sponsored by the NRC Division of Reactor Safety Research for October-December 1976", ORNL/NUREG/TM-87 Oak Ridge National Laboratory, 1977.

[96] V.F. Urbanic, T.R. Heidrich, "High TemperatureOxidation of Zircaloy-2 and Zircaloy-4 in Steam", *Journal of Nuclear Materials*, **75**, pp. 251-261,1978.

[97] J.F. White, et al., "Fifth Annual Report - High Temperature Material Programs, Part A", GEMP-400A, February 1966.

[98] A.S. Benjamin, D.J. McCloskey, D.A. Powers, S.A. Dupree, "Spent Fuel Heatup Following Loss of Water During Storage", SAND77-1371, NUREG/CR-0649, Sandia National Laboratories, Albuquerque, NM, March 1979.

[99] J.Roes, "Experimentelle Untersuchungen zur Graphitkorrosion und Aerosolentstehung beim Lufteinbruch in das Core eines Kugelhaufen-Hochtemperaturreaktors", PhD Thesis, Jul-2956, KfA, 1994.

[100] L.J. Siefken, et al., "MATPRO - A Library of Material Properties for Light Water Reactor Accident Analysis", NUREG/CR-6150, Vol. 4, Rev. 2, INEL-96/0422, January 2001.

[101] "MATPRO - A Library of Material Properties for Light Water Reactor Accident Analysis", INEEL/EXT-02-00589, Vol. 4, Rev. 2.2, INEL-96/0422, October 2003.

[102] D.L. Hargman, et al., "MATPRO Version 11 (Revision 2), A Handbook of Material Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior" NUREG/CR-0479, TREE-1280, Rev. 2, August, 1981.

[103] J.T. Prater, E.L. Courtright, "Zircaloy 4 Oxidation at 1,300 to 2,400ºC", NUREG/CR-4889, PNL-6166, Pacific Northwest Laboratory, 1987.

[104] A.S. Benjamin, D.J. McCloskey, D.A. Powers, S.A. Dupree, "Spent Fuel Heatup Following Loss of Water During Storage", SAND77-1371, NUREG/CR-0649, Sandia National Laboratories, Albuquerque, NM, March 1979.

[105] https://chem.libretexts.org/Textbook_Maps/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Thermodynamics/Energies_and_Potentials/Enthalpy/Standard_Enthalpy_Of_Formation

[106] P. Nanni, V. Buscaglia, G. Battilana, "E. Reudl, "Air Oxidation of a Mn-Cr austenitic steel of potential use for fusion reactor structural applications between 1073 and 1473 K at 10^5 Pa", *Journal of Nuclear Materials*, Vol. **182**, pp. 118-127, 1991.

[107] E. Hontanon, A. de los Reyes, J.A. Capitao, "The CÆSAR Code for Aerosol Resuspension in Turbulent Pipe Flows. Assessment Against the STORM Experiments", *J. Aerosol Sci.* Vol. **31**, No. 9, pp. 1061-1076, 2000.

[108] P. Vainshtein, G. Ziskind, M. Fichman, C. Gutfinger, "Kinetic Model of Particle Resuspension by Drag Force", *Physical Review Letters*, Vol. **78**, No. 3, pp. 551 - 554, 1997.

[109] M.W. Reeks, J. Reed, D. Hall, "On the Resuspension of Small Particles by a Turbulent Flow", *Journal of Physics D: Applied Physics*, Vol. 21, pp. 574-589, 1988.

[110] K.L. Johnson, K. Kendall, and A.D. Roberts, "Surface energy and the contact of elastic solids", *Proceedings of the Royal Society A*, **324**, pp. 301-313, 1971

[111] G. Soltani, M. Ahmadi, "On Particle Adhesion and Removal Mechanisms in Turbulent Flows", *Journal of Adhesion Science and Technology*, Vol. **8**, No. 7, pp. 763-785, 1994.

[112] A.R. Costelo, J.A. Capitao, G.D. Santi, "International Standard Problem 40 – Aerosol Deposition and Resuspension, Final Comparison Report", OECD NEA/CSNI/R(99)4, February 1999.

[113] E. Kostowski, "Promieniowanie cieplne" ("Thermal Radiation"), ISBN 83-01-10847-9, PWN, Warszawa, 1993.

[114] E. Kostowski, "Model gazu szarego i model pasmowy promieniowania gazòw" ("Gray gas model and band energy model", *Energetyka*, **131**, pp. 109-125, 1999.

[115] H.C. Hottel, "Radiant Heat Transmission", Chapter 4 in W.H. McAdams, "Heat Transmission", ISBN-13: 978-0898748765, Robert E. Krieger Publishing Company, Third Edition, Malabar, Florida, 1985.

[116] J.A. Wiebelt, "Engineering Radiation Heat Transfer", Holt Rinehart and Winston, New York, 1966.

[117] G. Nothangel, "Theoretical Background of Plate-out", NECSA report RP-REP-0113, June 2005.

[118] K. Röllig, A. Christ, F.W. Haase, J. Wiedmann, "Caesium Deposition on HTR Primary Circuit Materials", IAEA Specialist's Meeting on Fission Product Release and Transport in Gas-cooled Reactors, Gloucester, 22 – 25 October 1985.

[119] K. Wegner, "Mass Transfer - Estimation of Diffusivities", ETH, Zurich, Lecture 8, 2017, https://www.ethz.ch/content/dam/ethz/special-interest/mavt/process-engineering/particle-technology-laboratory-dam/documents/lectures/mass-transfer/script17/Chapter_8_Diffusivities.pdf

[120] K. Röllig, A. Crist, F.W. Hasse, J. Wiedmann, "Cesium Deposition on HTR Primary Circuit Materials", IAEA Specialist's Meeting on Fission Product Release and Transport in Gas Cooled Reactors, Gloucester, 22-25 October 1985.

[121] L. Stassen, "Validation of the Plate-out Model in the RADAX code used for Plate-out and Dust Activity Calculations at PBMR", PHYSOR-2006, ANS Topical Meeting on Reactor Physics, Canada, September 10-14,2006

[122] V. Gnielinski, "Neue Gleichungen fur den Warmte- und den Stoffubergang in turbulent durchstromten Rohren und Kanalen", *Forschung im Ing. Wes.* **41**, No. 1, p. 8-16, 1975.

[123] M.M. Stempniewicz, "Generic Containment Run-2, Analyses with SPECTRA and MELCOR", SARNET-2 Meeting, Kaunas, Lithuania, February 19, 2013.

[124] M.Ishii, K. Mishima, "Study of two-fluid model and interfacial area", NUREG/CR-1873, ANL-80-111, December 1980.

[125] J.C. Chen, "A correlation for boiling heat transfer to saturated fluids in convective flow", ASME preprint 63-HT-34 presented at 6-th National Heat Transfer Conference, Boston, August 1963.

[126] R.J. Weatherhead, "Nucleate boiling characteristics and the critical heat flux occurrence in subcooled axial flow water systems", ANL 6675, 1963.

[127] D.D. Wittke, B.T. Chao, "Collapse of Vapor Bubbles with Translatory Motion", *Transactions of the ASME, Journal of Heat Transfer*, pp. 17-24, February 1967.

[128] E.K. Kalinin, I.I. Berlin, V.V. Kostyuk, E.M. Nosova, "Heat Transfer in Transition Boiling of Cryogenics", paper presented at *Cryogenics Engineering Conference*, Queens University Kingston, Ontario, Canada, July 1975.

[129] W. Friz, "Berechnung des maximal volume van Dampfblasen", *Phys. Z*, **36**, p.379, 1935.

[130] W. Nusselt, "Die Oberflachenkondensation des Wasserdampfes", *Zeitschr. Ver. Deutsh. Ing.* **60**, 541, 1916.

[131] J.C. Chato, "Laminar condensation inside horizontal and inclined tubes", *J. Am. Soc. Heating Refrig. Aircond. Engrs.*, p. 52, February 1962.

[132] J. Gerstmann, P. Griffith, "Effect of Surface Instability on Laminar Film Condensation", MIT Heat Transfer Lab. Report 5050-36, *Int. J Heat Mass Transfer*, **10**, p. 567, 1967.

[133] Kuhn S.Z., Schrock V. E., Peterson P.F., "An investigation of condensation from steam-gas mixtures flowing downward inside a vertical tube", *Nuclear Engineering and Design* **177** (1997) 53-69.

[134] D.G. Ogg, "Vertical Downflow Condensation Heat Transfer in Gas-Steam Mixtures", MSc Thesis, University of California, Berkeley, 1991.

[135] M.M. Stempniewicz, "Simulation of Containment Transient Response During Accidents in Advanced Reactor Types - The Computer Code SPECTRA", PhD Thesis, *Silesian Technical Unicersity*, Gliwice, May 2000, NRG report 21437/00.52167/P, Arnhem, May 2000.

[136] J.G. Collier, Convective Boiling and Condensation", ISBN 0-19-856296-9, McGraw Hill International Book Company, First Edition: 1972, Second Edition: 1981.

[137] J.G. Collier, D.J. Pulling, "Heat transfer to two-phase gas-liquid systems; Part II: Further data on steam-water mixtures in the liquid dispersed region in an annulus", *Trans Inst. Chem. Eng.*, Vol. **42**, p.127, 1962.

[138] O.P. Bergelin, P.K. Kegel, F.G. Carpenter, C. Gayley, ""Co-Current Gas-Liquid Flow in Vertical Tubes", *Heat Transfer and Fluid Mechanics Institute*, Berkeley, California, Published by ASME, pp. 19-28, 1949.

[139] W.M. Rohsenow, "A Method of Correlating Heat Transfer Data for Surface Boiling of Liquids", *Trans ASME*, **74**, p.969, 1952.

[140] H.K. Forster, N. Zuber, "Bubble Dynamics and Boiling Heat Transfer", *AIChE J.* Vol. **1**, pp. 532-535, 1955.

[141] N. Zuber, "Hydrodynamic Aspects of Boiling Heat Transfer", PhD Thesis, AECU-4439, 1959, doctoral dissertation, University of California at Los Angeles, 1959.

[142] L. Biasi et al., "Studies on burnout, Part 3", *Energia Nucleare*, **14**(9), pp. 530-536, 1967.

[143] Working Party of the Heat and Mass Transfer Section of the Scientific Council of the U.S.S.R. Academy of Sciences, "Tabular data for calculating burnout for boiling water in uniformly heated round tubes", *Teploenergetika*, **12**(3), pp. 47-51, 1977.

[144] Groeneveld, D.C., Cheng, S.C., Doan, T., "1986. AECL-UO critical heat flux lookup table.", *Heat Transfer Engineering*, Vol. **7**, No. 1-2, pp. 46-62, 1986.

[145] B. Matzner, "Basic experimental studies of boiling fluid flow and heat transfer at elevated pressures", TID 18978, 1963.

[146] L.A. Bromley, "Heat Transfer in Stable Film Boiling", *Chem Eng. Prog. Ser.*, **46**, pp.221-227, 1950.

[147] P.J. Berenson, "Film Boiling Heat Transfer from a Horizontal Surface", *ASME Journal of Heat Transfer*, **83**, No. 3, pp. 351-358, 1961.

[148] N. Zuber, M. Tribus, "Further Remarks on the Stability of Boiling Heat Transfer", UCLA report 58-5, 1958.

[149] F.F. Simon, S.S. Papell, R.J. Simoneau, "Minimum Film Boiling Heat Flux in Vertical Flow of Liquid Nitrogen", NASA TND-4307, 1968.

[150] R.S. Dougal, W.M. Rohsenow, "Film Boiling on the Inside of Vertical Tubes with Upward Flow of the Fluid at Low Qualities", MIT 9079-26, 1963.

[151] J.J. Ginoux, "Two-Phase Flows and Heat Transfer with Application to Nuclear Reactor Design Problems", ISBN 0-07-023305-5, Hemisphere Publishing Corporation, 1978.

[152] W.E. Ranz, .R. Marshall, "Evaporation from drops", *Chem. Eng. Prog.*, **48,** pp. 141-146, 1952.

[153] C.G. Downing, "The evaporation of drops of pure liquids at elevated temperatures: rates of evaporation and wet-bulb temperatures", *A.I.Ch.E.Jl.* **12**, pp. 760-766, 1966.

[154] J. Schwarz, J. Smolik, "Mass transfer from a drop - I. Experimental study and comparison with existing correlations", *Int. J. Heat Mass Transfer*, Vol **37**, No. 14, pp. 2139-2143, 1994.

[155] D. Moalem, S. Sideman, "The effect of Motion on Bubble Collapse", *Int. J. Heat Mass Transfer*, Vol. **16**, pp. 2321-2329, 1973.

[156] G.G. Brucker, E.M. Sparrow, "Direct Contact Condensation of Steam Bubbles in Water at High Pressure", *Int. J. Heat Mass Transfer*, Vol. **20**, pp. 371-381, 1977.

[157] B.F. LeClair, A.E. Hamielec, "Viscous Flow Through Particle Assemblages at Intermediate Reynolds Numbers - A Cell Model for Transport in Bubble Swarms", *Canadian J. Chem. Eng.*,. **49**, p.713, 1971.

[158] R. Clift, J.R. Grace, M.E. Weber, "Bubbles, Drops, and Particles", ISBN 0-12-176950-X, Academic Press, 1978.

[159] D.A. Powers, "A Simplified Model of Decontamination by BWR Steam Suppression Pools", NUREG/CR-6153, SAND93-2588, May 1997.

[160] A.B. Newman, "The drying of porous solids diffusion calculations", *Am. Inst. Chem. Engrs.*, **27**, p. 310-333, 1931.

[161] S.S. Kutateladze, M.A. Styrykovic, "Hydraulics of Gas-Liquid Systems", Wright Field Trans. F-TS-9814/V, Moscow, 1958.

[162] G.I. Taylor, *Proc. Roy. Soc.* (London), Vol. **A210**, 0.192, 1950.

[163] Rayleigh Lord: *Proc. London Math. Soc.*, Vol **10**, p. 1, 1878; *Proc. Roy. Soc.*, (London), Vol. **29**, p. 71, 1879; *Phil. Mag.*, Vol **34**, p. 177, 1892; for a discussion summarizing the work, see Sir Horace Lamb's "Hydrodynamics", 6-th Edition, pp.471-473, Dover Publications, Inc., New York.

[164] J. Hadamard, *Compt. Rend. Acad. Sci.*, Paris, Vol. **152**, pp. 1735-1738, London 1911.

[165] W. Rybczynski, *Bull. Acad. Sci.*, Cracovie, Vol A, pp. 40-46, 1911.

[166] L. Schiller, A. Naumann, "A Drag Coefficient Correlation", *Zeiting Ver. Deutsch. Ing.*, Vol. **77**, p.318-320, 1935.

[167] V.G. Levich, "Physicochemical Hydrodynamics", ISBN-13: 978-0136744405, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962.

[168] F.N. Peebles, H.J. Gerber, "*Chem Eng. Progr.*, Vol. **49**, pp. 88-97, 1953.

[169] R.M. Davies, G.I. Taylor, *Proc. Roy. Soc.*, London, Vol. **200**, Ser. A, pp. 375-390, 1950.

[170] R. Collins, *J. Fluid Mech.*, Vol **28**, Part I, pp. 97-112, 1967.

[171] N. Zuber, J.A. Findlay, "Average volumetric concentration in two-phase systems", *Journal of Heat Transfer*, Vol. **87**, pp. 453-468, 1965.

[172] K.H. Sun, J.H. Leinhard, "The Peak Pool Boiling Heat Flux on Horizontal Cylinders", *Int. J. Heat Mass Transfer*, **13**(9), pp. 1425-1439, 1970.

[173] C.A.J. Fletcher, "Computational Techniques for Fluid Dynamics", Springer-Verlag New York, Berlin, Heidelberg, First Edition: ISBN 0-387-18151-2, 1988, Second Edition: ISBN 0-387-53058-4, 1991.

[174] "State of the Art Report on Containment Thermalhydraulics and Hydrogen Distribution", Prepared by OECD/NEA Group of Experts, Nuclear Safety, NEA/CSNI/R(99)16, June 1999.

[175] J.K. Davidson, D. Harrison, "Fluidised Particles", ISBN-13: 978-0521047890, Cambridge University Press, London, 1963.

[176] J.D. Bugg, R.D. Rowe, "Bubbling Formation at the Wellhead of a Subsea Oilwell Blowout", *Transactions of the Institute of Chemical Engineers*, Vol. **38**, pp. 335-342, 1960.

[177] W. Fritz, "Berechnung des Maximal Volume von Dampfblasen", *Phys. Z.*, **36**, p. 379, 1935.

[178] F.R. Larson, J. Miller, "A Time-Temperature Relationship for Rupture and Creep Stress", *Transactions of the ASME,* July 1952, pp. 765-775.

[179] J.L. Rempe, et al., "Light Water Reactor Lower Head Failure Analysis", NUREG/CR-5642, EGG-2618, October 1993.

[180] L.E. Herranz, C.L. Del Pra, A. Dehbi, "Major Challanges to Modelling Aerosol Retention Near a Tube Breach During Steam Generator Tube Rupture Sequences", *Nuclear Technology*, Vol. **158**, pp. 83 - 93, April 2007.

[181] M.R. Sippola, W.W. Nazaroff, "Particle Deposition from Turbulent Flow: Review of Published Research and Its Applicability to Ventillation Ducts in Commenrcial Buildings", Lawrence Berkeley National Laboratory Report, LBNL - 51432, June, 2002.

[182] Y.S. Cheng, C.S. Wang, "Motion of Particles in Bends of Circular Pipes", *Atmospheric Environment*, Vol. **15**, pp. 301 - 306, 1981.

[183] R.P. Wichner, "Fission Product Plateout and Liftoff in the MGHTR Primary System: A Review", NUREG/CR-5647, ORNL/TM-11685, April 1991.

[184] A. Y. Inayatov, "Correlation of Data on Heat Transfer Flow Parallel to Tube Bundles at Relative Tube Pitches of $1.1 < s/d < 1.6$," Heat Transfer-Soviet Research, 7, 3, May-June 1975.

[185] A. Ullman, R. Acharia, D.R. Olander, "Theretical accommodation coefficients of inert gases on stainless steel and UO_2", *J. Nucl. Matl.,* **51**, p 277, 1974.

[186] A.M. Ross, R.L. Stoute, "Heat transfer coefficient between UO2 and Zircaloy-2", AECL-1552, 1962.

[187] T.N. Cetinkale, M. Fishenden, "Thermal Conductance of Metal Surfaces in Contact", Proceedings of the General Discussion on Heat Transfer, pp. 271-275, London, 1951.

[188] D.D. Lanning, C.R. Hann, "Review of Methods Applicable to the Calculation of Gap Conductance in Zircaloy-clad, UO_2-fuel rods", Batelle Pacific Northwest Laboratories, BNWL-1894, UC-78b, April 1975.

[189] J.B. Ainscough, "Gap Conductance in Zircaloy-Clad LWR Fuel Rods", OECD NEA, CSNI Report No. 72, April 1982.

[190] J.E. Rivera, "Performance of Light Water Reactors Fuel Rods During Plant Power Changes", MSc Thesis, Messachusetts Institute of Technology, 1981.

[191] A.E. Bergles, J.H. Leinhard V, G.E. Kendall, P. Griffith, "Boiling and Evaporation in Small Diameter Channels", *Heat Transfer Engineering*, **24**(1): 18-40, 2003.

[192] S.G. Kandlikar, "Critical heat flux in subcooled flow boiling - an assessment of current understanding and future directions for research", *Multiphase Science and Technology*, **13**, (3): 207-232, 2001.

[193] S.H. Kim, W.P. Baek, S.H. Chang, "Measurement of critical heat flux in narrow annuli submerged in saturated water", *Nuclear Engineering and Design*, **199**: 41-48, 2000.

[194] F.P. Incropera, D.P. DeWitt, "Fundamentals of Heat and Mass Transfer", ISBN-13: 978-0471304609, 4th Edition, 1996.

[195] K. Mikityuk, "Heat transfer to liquid metal: Review of data and correlations for tube bundles", *Nuclear Engineering and Design*, **239**, 680-687, 2009.

[196] W. Pfrang, D. Struwe, "Assessment of Correlations for Heat Transfer to the Coolant for Heavy Liquid Metal Cooled Core Designs", Institut für Reaktorsicherheit, report FZKA7352, October 2007.

[197] "RELAP5/MOD3.3 Code Manual, Volume 1: Code Structure, System Models, and Solution Methods", NUREG/CR-5535/Rev P4-Vol I, October 2010.

[198] "Status Report on Spent Fuel Pools Under Loss Of Cooling and Loss of Coolant Accidents, Final Report", NEA/CSNI/R(2015)2, May 2015

[199] J. Birchley, L. Fernandez-Moguel, "Simulation of air oxidation during a reactor accident sequence: Part 1 - Phenomenology and model development", *Annals of Nuclear Energy*, **40** (2012) 163-170

[200] M. Steinbrück, "Prototypical experiments relating to air oxidation of Zircaloy-4", *Journal of Nuclear Materials*, **392**, pp. 531-544, 2009.

[201] D.A. Nied, A. Bejan, "Convection in Porous Media", ISBN 978-1-4614-5540-0, Springer, 2013, http://www.springer.com/978-1-4614-5540-0

[202] Kerntechnischer Ausschuss, SAFETY STANDARDS of the Nuclear Safety Standards Commission (KTA), "KTA 3102.1, Reactor Core Design for High-temperature Gas-Cooled Reactor, Part 1: Calculations of the Material Properties of Helium", Carl Heymanns Verlag KG (1978).

[203] Wolfgang Wiesenack, "Fuel behavior modelling - basic principles and related data", Doc. No. IFE/HR/E-2010/008, ISBN 978-82-7017-820-9, IAEA Advanced Workshop, April 2010.

[204] S.A. Morsi, A. J. Alexander, "An investigation of particle trajectories in two-phase flow systems", *J. Fluid Mech.* **55**, pp. 193-208, 1972.

[205] https://en.wikipedia.org/wiki/Drag_coefficient

[206] D.E. Jacober, M.J. Matteson, "The Collection of Aerosols from Gas Streams by Impaction on Multiple Spherical Targets", *Aerosol Science and Technology* **4**, pp.:433-443, 1985.

[207] Yoon, R. H. and Luttrell, G. H. (1989), "The effect of bubble size on fine particle flotation", *Mineral Processing and Extractive Metallurgy Review*, **Vol 5**. pp. 101-122.

[208] Afruns, J. P. and Kitchener, J. A. (1976), "The absolute rate of capture of single particles by single bubbles", *Flotation. Gaudin Memorial Volume*, M. C. Fuerstenau Editor. SME pp. 625-637.

[209] http://www.mineraltech.com/MODSIM/ModsimTraining/Module5/TechnicalNotes9.pdf

[210] J.R. Engel, et al., "Conceptual Design Characteristics of a Denatured Molten-Salt Reactor with Once-Through Fueling", ORNL/TM-7207, July 1980.

[211] Guiqiu Zheng, Brian Kelleher, Lingfeng He, Guoping Cao, Mark Anderson, Todd Allen, Kumar Sridharan, "High-Temperature Corrosion of UNS N10003 in Molten Li2BeF4 (FLiBe) Salt", *Corrosion*, Vol. **71**, No. 10, October 2015.

[212] P.N. Haubenreich, J.R. Engel, B.E. Price, H.C. Claiborne, "MSRE Design and Operations Report, Part III: Nuclear Analysis", ORNL-TM-730, Feb. 1964.

[213] R.J. Kedl, A. Houtzeel, "Development of a Model for Computing Xe-135 Migration in MSRE", ORNL-4069, June 1967.

[214] N. Hottle, "Pebble Bed Reactor Plant Design Description", AREVA NP Inc. Technical Data Record, Document No.: 12-9149697-000, 20004-018 (10/18/2010), https://art.inl.gov/NGNP/NEAC%202010/Pebble%20Bed%20Reactor%20Plant%20Design%20Description%20-%20AREVA.pdf

[215] "Fuel performance and fission product behaviour in gas cooled reactors", IAEA-TECDOC-978, november 1977.

[216] B.P. Collin, "Diffusivities of Ag, Cs, Sr, and Kr in TRISO Fuel Particles and Graphite", INL/EXT-16-39548, Revision 0, September 2016.

[217] Makino, A. and C. K. Law, "Ignition and Extinction of CO Flame over a Carbon Rod," *Combustion Science and Technology*, **73**, 589-615, 1990.

[218] Fred Gelbard, "Graphite Oxidation Modeling For Application in MELCOR" SAND2008-7852, January 2009.

[219] R. van Bremen, "Water ingress scenario analyses of a thorium fuelled HTR", NERA-131-2013-006, 2 July 2013.

[220] Zhe Dong, Xiaojin Huang, Liangju Zhang, "A nodal dynamic model for control system design and simulation of an MHTGR core", *Nuclear Engineering and Design* **240** (2010) 1251–1261.

[221] Zhe Dong, Yifei Pan, Maoxuan Song, Xiaojing Huang, Yujie Dong, Zuoyi Zhang, "Dynamic Modeling and Control Characteristics of the Two-Modular HTR-PM Nuclear Plant", Science and Technology of Nuclear Installations, Volume 2017, Article ID 6298037, https://doi.org/10.1155/2017/6298037

[222] A. Beluco, E.B. Camano Schettini, "An Improved Expression for a Classical Type of Explicit Approximation of the Colebrook White Equation with Only One Internal Iteration", International Journal of Hydraulic Engineering 2016, 5(1): 19-23.

[223] J.M. Ramilison, J.H. Lienhard, "Transition Boiling Heat transfer and the Film Boiling Regime", *Transactions of ASME*, **736** / Vol. 109, 1987

[224] https://www.nuclear-power.com/nuclear-power/fission/delayed-neutrons/

[225] Ji-Hun Kim, Sejin Kwonb, Jinyoung Choi, Yong Jin Cho, "Concrete ablation analysis for molten corium-concrete interaction mitigation strategy", Annals of Nuclear Energy 132 (2019) 615–627.

[226] T. Sevón, "Molten Core - Concrete Interactions in Nuclear Accidents, Theory and Design of an Experimental Facility", VTT Research Notes 2311, ESPOO 2005 https://publications.vtt.fi/pdf/tiedotteet/2005/T2311.pdf

[227] L.L. Humphries, et.al., "MELCOR Computer Code Manuals, Vol. 1: Primer and User's Guide, Vol. 2: Reference Manuals, Version 2.1.6840", SAND2015-6691R, Published: August 2015.

[228] Woong Kee Kim, Ji Hoon Shim, Massoud Kaviany, "Thermophysical properties of liquid UO_2, ZrO_2 and corium by molecular dynamics and predictive models", Journal of Nuclear Materials 491 (2017) 126-137.

[229] M. Ramacciotti, et.al, "Viscosity Models for Corium Melts, Nuclear Engineering and Design 204, 377-389, 2001.

[230] C. Spengler, H.-J. Allelein, M. Cranga, F. Duval, J.-P. Van Dorsselaere, "Assessment and Development of Molten Corium Concrete Interaction Models for the Integral Code ASTEC", https://simplyinfo.org/wp-content/uploads/2011/09/pe_382_24_1_seminar2_01_2005.pdf

[231] Phase diagram, UO_2/ZrO_2 system, https://thermodb.jaea.go.jp/data/en/pd/ZrO2-UO2.html

[232] A. Tayebi, B.G. Dehkordi, "Development of a PISO-SPH method for computing incompressible flows", J Mechanical Engineering Science, May 14, 2013, https://www.researchgate.net/publication/278082436