NUREG/CR--5545

TI91 002433

VICTORIA: A Mechanistic Model of Radionuclide Behavior in the Reactor Coolant System Under Severe Accident Conditions

Manuscript Completed: September 1990
Date Published: October 1990

Prepared by
T. J. Heames[1], D. A. Williams[2], N. E. Bixler, A. J. Grimley,
C. J. Wheatley[3], N. A. Johns[2], N. M. Chown[2]

Sandia National Laboratories
Albuquerque, NM 87185

Prepared for
Division of Systems Research
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555
NRC FIN A1837

[1]SAIC, Albuquerque, NM
[2]UKAEA, Winfrith, England
[3]UKAEA, Culcheth, England

ABSTRACT

This document provides a description of a model of the radionuclide behavior in the reactor coolant system (RCS) of a light water reactor during a severe accident. This document serves as the user's manual for the computer code called VICTORIA, based upon the model.

The VICTORIA code predicts fission product release from the fuel, chemical reactions between fission products and structural materials, vapor and aerosol behavior, and fission product decay heating.

This document provides a detailed description of each part of the implementation of the model into VICTORIA, the numerical algorithms used, and the correlations and thermochemical data necessary for determining a solution. A description of the code structure, input and output, and a sample problem are provided.

The VICTORIA code was developed upon a CRAY-XMP at Sandia National Laboratories in the U.S.A. and a CRAY-2 and various SUN workstations at the Winfrith Technology Centre in England.

CONTENTS

Page

ABSTRACT iii

LIST OF FIGURES ix

LIST OF TABLES xi

EXECUTIVE SUMMARY xii

NOMENCLATURE xiii

1. INTRODUCTION 1-1

1.1 Light Water Reactor Severe Accident Chemistry ... 1-1

1.1.1 Two Stages of Severe Accidents 1-1
1.1.2 In-Vessel Product Release 1-2

1.2 Observed Chemistry Effects in Accidents 1-3
1.3 Organization of the Report 1-4
1.4 Chemical Modeling in VICTORIA 1-5

2. METHODOLOGY AND FRAMEWORK OF THE VICTORIA CODE 2-1

2.1 Solution Domain 2-1
2.2 Modeling Overview 2-2
2.3 Calculational Flow 2-3
2.4 Time Step Control 2-4

3. SPECIES BEHAVIOR AND TRANSPORT WITHIN THE FUEL GEOMETRY 3-1

3.1 Governing Equations 3-2
3.2 Fuel Intra-granular Processes 3-3

3.2.1 Equilibrium Chemistry in the Fuel Grains 3-3
3.2.2 Solid-State Diffusion 3-6

3.3 Fuel Open Porosity Processes 3-8

3.3.1 Chemical Equilibrium 3-8
3.3.2 Diffusive Transport 3-9
3.3.3 Numerical Scheme 3-10

3.4 Fuel Cladding Gap Process 3-13

3.4.1 Equilibrium Chemistry 3-14
3.4.2 Transport 3-14

CONTENTS (continued)

Page

3.5 Information for the Code User 3-15

4. BULK GAS SPECIES TRANSPORT 4-1

4.1 Species Conservation Equation 4-1
4.2 Numerical Scheme 4-3
4.3 Information for the Code User 4-7

5. CHEMISTRY 5-1

5.1 Background 5-1
5.2 Equilibrium Chemistry 5-3

5.2.1 Derivation of the Equations 5-3

5.2.1.1 Conditions for chemical equilibrium 5-3

5.2.2 Equations for Two Phases at Given T and V 5-6
5.2.3 Two Approximations 5-10

5.3 Applications 5-11

5.3.1 Application to the Bulk Gas Region 5-11
5.3.2 Application to the Structure Surface Region 5-13
5.3.3 Application to the Fuel Cladding/Gap Region 5-14
5.3.4 Application to the Fuel Open Porosity 5-15

5.4 Solution Method 5-15

5.4.1 Properties of the Equation 5-15
5.4.2 Ill-conditioning of the Jacobian and Implications 5-18
5.4.3 Change of Variable 5-20
5.4.4 Solution Algorithm 5-22
5.4.5 Convergence Criteria 5-24

5.5 Zirconium Oxidation Model 5-25

5.5.1 Urbanic and Heidrick Experiments 5-25
5.5.2 Implementation of the Model 5-27

CONTENTS (continued)

Page

6. RADIOACTIVE DECAY HEATING 6-1

6.1 Volumetric and Surface Decay Powers 6-1
6.2 Information for the Code User 6-2

7. AEROSOL SPECIES BEHAVIOR AND TRANSPORT 7-1

7.1 Description of the Model 7-1

7.1.1 Gas Properties 7-2
7.1.2 Aerosol Properties 7-4
7.1.3 Flow Properties 7-5
7.1.4 Deposition Rates 7-8
7.1.5 Agglomeration Rates 7-12
7.1.6 Source Distributions 7-15

7.2 Numerical Methods 7-16

7.2.1 Discretization of the Governing Equations 7-16
7.2.2 Solution Method 7-20
7.2.3 Moments of the Discretized Distribution .. 7-20
7.2.4 Aerosol Transport 7-22

8. REFERENCES 8-1

APPENDIX A - ELEMENTS AND SPECIES A-1

A.1 Element and Species Selection A-1
A.2 Gibbs Free Energy Data A-2
A.3 References A-25

APPENDIX B - INPUT/OUTPUT FILE DESCRIPTIONS B-1

B.1 Input File Description B-1
B.2 Output File Description B-19

B.2.1 Input Edit B-19
B.2.2 Transient Mass Balance B-19
B.2.3 Transient Printed Output B-19

B.3 Graphics File Description B-25

CONTENTS (continued)

Page

APPENDIX C - SUBROUTINE GLOSSARY AND VARIABLE DEFINITIONS.................................. C-1

C.1 Subroutine Glossary............................ C-1

C.1.1 Main Program and Subroutines............. C-1
C.1.2 Data Input Subroutines.................. C-2

C.1.3 Output Subroutines...................... C-4
C.1.4 Chemistry Subroutines................... C-4
C.1.5 Aerosol Behavior and Transport Subroutines.............................. C-6
C.1.6 Mathematical Software Routines.......... C-13

C.2 VICTORIA Data Management....................... C-27
C.3 VICTORIA Variable Definitions.................. C-29
C.4 References..................................... C-38

LIST OF FIGURES

Figure Page

2.1 Simple mesh cell diagram........................ 2-6

2.2 Schematic of relevant geometry and phenomena for species behavior............................. 2-7

2.3 The program structure of the VICTORIA module..... 2-8

2.4 Flowchart of the logic flow in subroutine VIKI... 2-9

3.1 Sample of VICTORIA diffusion calculation mesh.... 3-16

3.2 Species transport in fuel cell m,n............... 3-17

4.1 Surface film parameters......................... 4-8

4.2 Species molar flow at bulk gas cell m,n.......... 4-9

6.1 Flowchart of the logic flow in the DECAY subroutine...................................... 6-4

A.1 Solid-state diffusion coefficients............... A-6

A.2 Cs species Gibbs free energy.................... A-17

A.3 Ba species Gibbs free energy.................... A-18

A.4 Sr species Gibbs free energy.................... A-19

A.5 Zr species Gibbs free energy.................... A-20

A.6 I species Gibbs free energy..................... A-21

A.7 Sn species Gibbs free energy.................... A-22

A.8 Te species Gibbs free energy.................... A-23

A.9 Mo species Gibbs free energy.................... A-24

B.1 Sample problem input edit....................... B-21

B.2 Sample problem output edit--Cs species densities and elemental masses.......................... B-22

B.3 Sample problem output edit--Cs aerosol densities. B-23

B.4 Sample problem output edit--elemental masses for all cells........................... B-24

LIST OF FIGURES (continued)

Figure		Page
B.5	Sample problem graphics--elemental release fraction, exit..................................	B-27
B.6	Sample problem graphics--elemental vapor fraction exit..................................	B-28
C.1	Main calling sequence flowchart..................	C-8
C.2	Input calling sequence flowchart.................	C-9
C.3	Output calling sequence flowchart................	C-10
C.4	Chemistry calling sequence flowchart.............	C-11
C.5	Aerosol calling sequence flowchart...............	C-12

LIST OF TABLES

Table		Page
7.1	Roughness of Common Pipe Materials...............	7-6
A.1	VICTORIA Element Set............................	A-4
A.2	Solid-State Diffusion Coefficients..............	A-5
A.3	VICTORIA Species Set............................	A-7
A.4	Gibbs Free Energy Data for VICTORIA Species......	A-11
B.1	Graphics Output Variables.......................	B-26

EXECUTIVE SUMMARY

The consequences of a severe reactor accident depend on the quantity, characteristics, and timing of the release of radionuclides from the containment; what reactor engineers call "the source term." The behavior of the radionuclides within the vessel and reactor coolant system will have a large effect on determining the size of the source term. VICTORIA is a computer code used to provide a predictions of the radionuclide release, transport, and deposition within the reactor vessel and reactor coolant system during a severe accident.

VICTORIA follows the evolution of 25 different elements, including the major vessel materials, as they interact with each other under the influence of the changing pressures, temperatures, and material motions during a severe reactor accident. The code requires an initial element distribution, which can be provided by any isotopic depletion code (such as ORIGEN), and the variation of pressures, temperatures, velocities and geometry provided by a core degradation analysis code (such as MELPROG). With this information VICTORIA determines the transport of the fission products from fuel grains, through the open porosity in the fuel, and into the coolant channels when there exists a break in the fuel cladding. During the course of the calculation, the fission products are allowed to interact chemically and form gaseous or condensed species that will either enhance or degrade the transport processes. When the species reach the coolant channel, aerosols can be formed which are allowed to deposit on structural surfaces where they can reheat, revaporize, and chemically interact with each other and the surface itself.

VICTORIA gives the user the fission product fuel release fractions, the quantities that are vapor, and the particular radionuclides that have interacted chemically and possibly condensed onto a surface. With this information the user can calculate the radionuclide release rates to the containment occurring through valves and breaks.

This document provides a detailed description of the phenomenological models, numerical methods, engineering correlations and thermochemical data in the VICTORIA code. In addition sufficient user information is provided to correctly prepare input data, understand results, and make code modifications.

NOMENCLATURE

Symbols

a	:	Number of molecules of an element in a species
A	:	Inverse matrix of a
	:	Surface area for species transport (m^2)
B	:	Particle mobility (s/kg) (7.9)
c	:	Species molar density, or concentration (kg-moles/m^3)
C	:	Elemental concentration, or molar density (kg-moles/m^3)
	:	Aerosol number concentration (particles/m^3)
Cn	:	Cunningham slip correction factor ((7.11)
d	:	Set of species indicies used to eliminate the Lagrangian for the chemical potential solution
d_h	:	Hydraulic diameter in the bulk flow (m)
D	:	Species diffusion coefficient (m^2/s)
f	:	Donor cell parameter
f_i	:	Fraction of species i that has diffused from the grain (3.3)
g_c	:	Gravitational constant
g_j	:	Partial of the variation in elemental concentration with respect to the species concentration (5.31)
G	:	Gibbs Free Energy (kcal/mole)
J	:	Jacobian matrix in the equilibrium chemistry calculation (5.46)
k	:	Boltzmann's constant (J/K-molecule)
K_n	:	Knudsen Number (7.8)
K_p	:	Rate constant in the Zr - ZrO_2 reaction (kg^2/m^4-s)
	:	Aerosol agglomeration kernel (particles/m^3)
l_p	:	Aerosol mean free path length (m) (7.6)
M	:	Species molecular weight (gm/gm-mole)
n	:	Number of moles of a species (moles)
n_e	:	Total number of elements
n_r	:	Number of unique species, $n_r = n_s - n_e$
n_s	:	Total number of species
N	:	Number of moles of an element (moles)
P	:	Pressure (Pa)
	:	Decay power (w/m^3)
r	:	Radial coordinate (m)
R_i	:	Equilibrium constant of the species for dissociation
	:	Aerosol removal rate (particles/m^3)
s	:	Species concentration source rate (kg-mole/m^3-s)
S	:	Aerosol source rate (particles/s-m^3)
t	:	Time (s)
T	:	Temperature (K)

NOMENCLATURE (continued)

Symbols

u	:	Radial velocity (m/s)
	:	Aerosol deposition velocity (m/s)
U	:	Internal energy (J)
v	:	Axial velocity (m/s)
V	:	Cell volume (m^3)
w	:	Zr consumption due to oxidation (kg/m^2)
x	:	Condensed phase mole fraction
z	:	Axial coordinate (m)

Symbols Greek

χ	:	Aerosol shape factors (input)
δ	:	Dirac and Kronecker delta functions
Δ	:	Thickness, mesh size (m)
	:	Step, interval (s)
ϵ	:	Particle efficiencies
ϕ	:	Agglomeration rates (s^{-1})
γ	:	Constants in the Gibbs free energy polynomial
λ	:	Lagrangian multiplier (5.4)
	:	Decay constant (6.1)
	:	Deposition rate (s^{-1}) (7.37)
μ	:	Chemical potential (kcal/mole)
	:	Dynamic viscosity (Pa-s)
ρ	:	Density (kg/m^3)
σ	:	Lennard-Jones interaction distance (3.8)
	:	Cube root of the standard deviation (7.58)

Subscripts

b	:	Bends
B	:	Brownian
c	:	Condensed phase
cl	:	Ceiling structure surfaces
D	:	Diffusiophoretic
f	:	Fuel/cladding film
fl	:	Floor structure surfaces
g	:	Bulk gas, gaseous phase, geometric mean
gas	:	Fraction that is vapor (gaseous)
G	:	Gravity
h	:	Elimination set (d) index
i	:	Species index
j	:	Element index
k	:	Gibbs free energy power series constant index
	:	Radioisotope index
l	:	Index to elements contained in species set
L	:	Laminar

NOMENCLATURE (continued)

Subscripts

m	:	Radial space coordinate index (+,- indicates the interface is used)
	:	Aerosol mass bin
max	:	Largest value
n	:	Axial space coordinate index (+,- indicates the interface is used)
o	:	Standard state pressure for pure species
p	:	Aerosol particle
r	:	Radial direction
s	:	Structure film on ceilings(cl), walls(w), or floors(fl)
T	:	Thermophoretic
w	:	Wall structure surfaces
x	:	Generic subscript for f,g,s
z	:	Axial direction
50	:	mass median
*1	:	Supermicron particles in turbulent flow
*2	:	Submicron particles in turbulent flow
*I	:	Turbulent inertial
*S	.	Turbulent shear
ω	:	Aerosol mass bin

Superscripts

chm	:	Chemistry
g	:	Fuel grains
k	:	Gibbs free energy power series constant index
o	:	Standard state for pure species
α	:	Current time step

1. INTRODUCTION

The release of radionuclides into the environment has been identified as the main concern in the event of an accident affecting a nuclear power reactor [1.1] The physics and chemistry which lead to this release are among the most complex that take place during the course of an accident. In order to be able to predict the outcome of these complex and interconnected events, it is necessary to model as many of these events as possible, and as accurately as possible. VICTORIA is a mechanistic computer code designed to model the release, transport, and deposition of fission products during a severe reactor accident.

1.1 Light-Water Reactor Severe Accident Chemistry

Perhaps the most difficult, yet most important problems associated with light-water reactor (LWR) safety research are those encountered in assessing the accident source term; that is, the quantity, timing, and characteristics of the release of radioactive materials from a nuclear reactor plant during the course of a severe accident. The importance of the source term is obvious: Nearly all the health risks to the public arising from the operation of nuclear power plants are associated with such releases. The difficulty of the source term problem is evident when one realizes that modeling fission product release phenomenology involves tracking the evolution of hundreds (if not thousands) of radioactive chemical species during an accident. While the accident is progressing, temperatures can range up to ceramic melting points; pressures can vary from roughly 0.1 to 20 MPa; the core geometry can change from largely intact fuel rods in an intact reactor system, to a highly degraded core, to molten pools involving tens of tons of metallic and ceramic melts; and the reactor atmospheric conditions can vary from highly oxidizing to reducing. These extremes of conditions compound the difficulty of the source term problem.

Although reactor accidents can last for a few days, the most critical period typically involves the first few tens of minutes to the few hours after loss of core cooling. This is because the period from initial rod degradation to vessel failure largely determines the subsequent course of the accident.

1.1.1 Two Stages of Severe Accidents

The accident progression, and the fission product release problems associated with it, can be divided into two stages. The first stage involves the in-vessel accident progression which includes the course of the accident within the vessel and reactor coolant system (RCS) up to the time of vessel failure and the ejection of core and structure materials to the containment building. The second stage (ex-vessel) involves the behavior of those reactor materials in the containment building and reactor vessel after vessel failure.

During the in-vessel stage of the accident, the containment building can be affected by several in-vessel processes. For most accidents of interest, the main processes affecting the containment behavior during the in-vessel stage are the releases of steam, hydrogen, noble-gas fission products (Xe and Kr), and highly volatile fission product species (notably Cs, I, Te and certain alkaline and rare-earth oxides) from the RCS to the containment through relief valves or breaks. Steam contributes to containment pressurization and heating, while hydrogen can lead to hydrogen combustion and/or detonation. Other phenomena of interest are confined to the vessel and RCS.

During the ex-vessel accident progression, important phenomena take place both in the vessel and RCS, and in the containment. This is because, while typically over half the core materials are calculated to be promptly ejected from the vessel upon vessel failure, the remaining core materials continue to heat up, interact with steam (and possibly air), and release fission products.

The VICTORIA code concentrates exclusively on the fission product behavior during the in-vessel stage of the accident.

1.1.2 In-Vessel Product Release

The fission products released during the in-vessel phase of risk dominant accidents are largely retained in the vessel and RCS due to their condensing on relatively colder structures between the fuel and the relief valves or breaks in the RCS, and/or condensing on aerosols which tend to settle and deposit within the vessel and RCS. The exceptions to the general retention of such fission products are threefold: the incondensable species Xe and Kr, the fraction of highly volatile species that is released through the relief valves or RCS break as vapors, and the fraction of all fission products that is carried out of the vessel and RCS as aerosols. An important concern involves the potential revaporization of deposited volatile species late in the accident due to the heating of local RCS surfaces above the threshold for significant vaporization. A somewhat less-studied concern involves the possible re-entrainment of deposited aerosols due to a dramatic increase in RCS flow velocities that could occur at the time of gross vessel failure.

A possibility not considered at all in the past studies is that once the RCS and the reactor vessel have been ruptured, air will be drawn through the vessel, changing the chemistry affecting release of radionuclides from residual fuel and vaporization of materials, including radionuclides deposited in the RCS.

The study of the release, transport, and deposition of fission products during the in-vessel stage of the accident involves:

1. Assessing the prompt source of gaseous and aerosolized fission products to the containment during the in-vessel accident progression.

2. Assessing the fractions of the various fission products released <u>and</u> retained within the vessel during the progression.

3. De :rmining the quantity and physiochemical state of fission products delivered in the core debris to the containment at the time of gross vessel failure.

4. Calculating the behavior of fission products retained within core materials that do not promptly exit the vessel at the time of gross vessel failure.

5. Assessing the potential revaporization and re-entrainment releases of volatile fission products from the RCS late in the accident.

1.2 Observed Chemistry Effects in Accidents

How important is it to study the fission product chemistry occurring in the course of a severe nuclear reactor accident and, does this chemistry affect the observed release of radioactive species? In the two most recent and most publicized severe nuclear reactor accidents (Three Mile Island and Chernobyl) how important were the effects of chemistry?

One of the more surprising results of the accident at Three Mile Island (TMI) was the small release of iodine [1.2]. The observed iodine release to the containment was orders of magnitude below the values predicted by the best models available at the time. The main reason for this difference is believed to be due to the chemical form assumed for iodine and the release mechanism in the models used. These predictions were based on the iodine existing as molecular elemental iodine in the reactor coolant system and upon the instantaneous release of the iodine from the fuel. Using chemical equilibrium considerations and the concentration of cesium available for release, it has since been postulated [1.3] that the iodine in the RCS did not exist as highly volatile I_2 but rather as the water soluble salt cesium iodide. The water solubility of both CsI and I_2 accounts for part of the low observed release of this element, while the significant retention of iodine, as well as tellurium and cesium, in the fuel accounts for the remainder. In this example, the effect of fuel release rates and chemistry within the RCS brought about a significant recuction in the release to the containment of a biologically important element.

The severe accident at Chernobyl in the USSR produced some equally surprising results. The release of many nonvolatile species (for example, Ru, Ba, Sr, Ce, etc.) was higher than one would anticipate based on the observed release of the volatile species, such as Cs, I, Te. A variety of possible chemical effects can be postulated to account for these observations. For example, ruthenium has a significantly higher vapor pressure in an oxidizing environment than in a reducing environment. The opposite is true for barium, strontium, and cerium. Possibly, because of the variety of mitigating efforts used and the reaction of graphite (used to moderate the fission neutrons) with air, the oxygen potential of the core varied with time and temperature. Another possibility that has been suggested was that the graphite moderated the temperature of the fuel by the carburization of UO_2 [1.4]. This latter mechanism would also account for the gross temporal features of the observed release. Again we see chemistry having a pronounced affect on the release of fission products in a severe accident, and conclude that the study of the chemistry of product release is an important part of reactor safety research.

1.3 Organization of Report

This report is intended to explain to users of the VICTORIA code the basis for the models and correlations of the program, how the code functions, and to allow a user to modify the code parameters for his or her specific project. The report is organized into seven chapters and three appendices in order to allow easy access to particular aspects of VICTORIA.

Chapter 1 includes a brief introduction to the rationale for VICTORIA's modeling of severe reactor accidents. The solution domain and calculational framework of the code are discussed in Chapter 2. Chapters 3 and 4 describe fission product species behavior within the fuel and the bulk gas, respectively. Chapter 5 describes the numerical approach and physical basis for the equilibrium chemistry analysis and Zircaloy oxidation reaction. Chapter 6 is concerned with the calculation of radioactive decay heating power.

Recently, the CHARM code [1.5] was developed to analyze aerosol behavior and has been incorporated into the current version of VICTORIA. CHARM is discussed in Chapter 7.

Appendix A discusses the elements and species used in VICTORIA, as well as presenting the relevant free energies and diffusion coefficients. Appendix B describes the input to, and output of the program. Finally, a subroutine glossary and the definitions of variable names are included in Appendix C, along with a brief description of the data management scheme currently used within VICTORIA.

1.4 Chemical Modeling in VICTORIA

The primary basis of the modeling in VICTORIA is the calculation of thermodynamic chemical equilibrium. We assume that this is reasonable because the ultimate driving force for the release of fission products is the inherent tendency of physical and chemical systems to move toward a state of thermochemical equilibrium. Thus, whenever the partial pressures of fission product chemical species in the coolant vapor overlying the fuel are lower than the species equilibrium partial pressures, there exists a driving force for release. Conversely, whenever the partial pressures existing above any surface are higher than the species equilibrium partial pressures above that surface, there is a driving force for condensation on the adjacent surface.

The tendency for radionuclide release is hindered by a number of transport resistances acting in series and parallel. These resistances are:

1. Transport from within the fuel to its surface--VICTORIA models the diffusive transport of each species within the fuel grains and through the open porosity.

2. Transport within the fuel-cladding gap—VICTORIA models transport along the gap as well as species accumulation in plenums and in condensed forms on the cladding inner surface.

3. Transport through a clad breach into the coolant channel--VICTORIA models the release through a specified breach.

4. Transport within the coolant channel and interaction with adjacent structures—VICTORIA calculates transport within the channel and models vapor condensation and aerosol deposition on structures. Additionally, a reduction in the partial pressures of the condensed species will cause revaporization.

Fission product release from all physical states is treated on a similar thermochemical and transport basis. However, the actual physical transport limitations vary greatly from one state to another. For intact rods, transport within the fuel can be limiting because of relatively low temperatures. The equilibrium vapor pressures themselves may also be low enough to limit release; A largely intact cladding is another major barrier to fission product release. However, mass transport in the vapor phase over intact rods is almost never limiting.

2. METHODOLOGY AND FRAMEWORK OF THE VICTORIA CODE

This section gives a general overview of the calculational method of the VICTORIA code. The code attempts to model the behavior of chemical species (with modeling emphasis on radioactive fission-product species) in the primary system of a Light Water Reactor (LWR) during a severe accident transient. This accident transient is assumed to subject the core to high temperatures, high pressures, metal oxidation, and substantial internal-structure geometry change. In order to model the behavior of chemical species in a severe accident environment, VICTORIA solves systems of conservation equations that describe the transport, phase, and chemical changes experienced by VICTORIA's species set.

2.1 Solution Domain

The spatial domain of VICTORIA covers the interior parts of a reactor vessel. In particular, the fuel rods, structural surfaces, and the volume normally occupied by the reactor coolant are VICTORIA's spatial domain. Since the reactor's structural geometry varies in a severe accident transient the spatial domain also varies: fuel rods degrade into distorted shapes (possibly into liquids or debris), structures melt and relocate, and the coolant volume can increase in some regions and decrease in others. In this report, fuel pellets that have intact cladding, degraded cladding, or no cladding at all will be referred to as the "fuel geometry." Note that debris beds and molten pools will not discussed.

In order to numerically solve the species transport equations for the coolant flow region, the vessel interior is discretized into a two-dimensional cylindrical (r,z) Eulerian mesh as illustrated in Figure 2.1. This mesh scheme normally has its origin at the bottom center of the vessel (although it can include the reactor cavity as well). The z-axis represents the vessel center line; the mesh can contain an arbitrary number of radial rings and axial levels (cells of lengths Δr and Δz, respectively). The sample mesh in Figure 2.1 (which is much coarser than that normally used in actual calculations) has four radial rings and six axial levels. The cross-hatched area represents the core region. An upper plenum above the core, a lower plenum below the core, and a vessel downcomer at the right periphery are also shown. (Note that this illustration is not to scale) Boundary conditions must be prescribed at all the outer surfaces of the domain depicted in Figure 2.1.

Within each of the two-dimensional transport cells there are subdomains that further discretize the fuel geometry. In this was the fuel rod can be separated into several radial fuel zones, a fuel-cladding gap, and the cladding.

Within each transport cell there are so-called "chemistry regions" that represent volumes of space where the fission species are allowed to experience chemical change. At the present time there are a maximum of five chemistry regions per transport cell: the fuel grains, the fuel open porosity, the fuel/cladding gap (which includes the inner cladding surface), the bulk gas (which includes the aerosol particles), and the structure surface. Logically, if one of these regions does not exist within the cell, then the associated chemistry is not performed. Within each transport cell the structure surfaces are treated as either a horizontal or a vertical surface area. Subsequent sections will illustrate these points more clearly.

The VICTORIA code was initially designed to be coupled to MELPROG [2.1] and TRAC [2.2]. This coupling was necessary to provide VICTORIA the geometry, mass, momentum, and energy of the fuel, cladding, structure, and coolant as the accident progressed. In the stand-alone version discussed here, all of this information must either be input or in some way made available to VICTORIA. An example of using the input to provide this information is given in the sample problem.

The time domain in VICTORIA is discretized into a series of equal time-steps (Δt) whose size is controlled by the spatially-coupled transport terms (convection and diffusion) in the conservation equations. Although some parts of the conservation equations are solved implicitly, the time step must be constrained by a material Courant condition [2.3] because of the presence of explicit source terms in the equations.

Another independent variable that can be associated with the solution domain is the discrete species index. The total number of species that can be treated by VICTORIA is arbitrarily large; that is, the maximum number of species is determined only by available computing resources and a knowledge of the necessary species thermophysical properties (such as Gibbs free energies and diffusion coefficients). At the present time VICTORIA treats only the 167 species listed in Appendix A.

2.2 Modeling Overview

VICTORIA discretizes the solution time domain into a series of equal time steps. During each of these steps, systems of equations are solved over the spatial meshes that describe the following coupled physical phenomena:

1. Species release from the fuel geometry that is controlled by chemistry and transport mechanisms

2. Species behavior in the reactor coolant volume that is controlled by gaseous transport, aerosol evolution and transport, and chemistry

3. Species behavior on structural surfaces that is controlled by aerosol deposition, vapor deposition and revaporization, and chemistry

Figure 2.2 illustrates these modeled phenomena. This figure portrays schematically the most complicated situation for a computational cell: a fuel domain from which species are being released into a bulk-gas environment that contains aerosols and structure surfaces. As an aside, if one were to disregard the species release from the fuel geometry (an application specific to reactor safety), VICTORIA could be used in any application that contains a species continuity problem with equilibrium chemistry and aerosol behavior. (Typical applications could be found in the non-nuclear power generation industry and in waste incinerator design.)

All of the phenomena illustrated in Figure 2.2 are spatially dependent, but only the convective transport terms are spatially coupled.

Practically speaking, many of the computational cells in the vessel domain will contain only a subset of the phenomena shown in Figure 2.2 at any given time. For example, the fuel rod upper plenum cell does not contain fuel material, and thus will not have to contend with the associated fuel geometry transport and chemistry (item 1 above).

For clarification purposes a distinction is made here between intra-cell (r,z dependent) and inter-cell (gradient dependent) phenomena. At the current time, only the indicated diffusive and convective transport terms in VICTORIA's modeling are spatially coupled.

2.3 Calculational Flow

The calculational framework contained within the VICTORIA module consists of these five main parts or submodules:

1. Equilibrium chemistry calculations for all species located within the fuel geometry, the bulk gas and aerosol particles, and on all structure surfaces

2. A behavior and transport calculation of the species within the UO_2/Zr fuel rod geometry

3. A transport calculation of the species in the bulk gas medium

4. A species continuity calculation on all structure surfaces including the fuel geometry surface

5. An aerosol formation, size evolution and transport calculation in the bulk gas medium and depos tion onto structure surfaces.

VICTORIA's calculational framework is illustrated in Figure 2.3. Each of the submodules (each of the boxes in Figure 2.3) solves its own internal system of equations which is explicitly coupled to the other submodules. This means that communication between submodules is not instantaneous but is typically one time step out-of-phase. An example of this explicit coupling is the solution of the fuel geometry diffusion equation which interfaces with the previous time interval bulk gas concentration. The physical couplings between the submodels in Figure 2.3 are explained in more detail in subsequent sections. Also indicated in Figure 2.3 are subroutine names that represent the drivers for the calculations associated with each submodule.

The Initial Species Inventory box of Figure 2.3 represents the acquisition of the initial species concentrations. This information must be provided by the user in the input deck. At Sandia, this information is currently obtained by post-processing the output from an ORIGEN calculation [2.4], although the results produced by any isotopic depletion code would supply the required initial condition.

Figure 2.4 illustrates the sequence of subroutine calls in VICTORIA. The main driver routine, VIKI, follows the "A" loop in Figure 2.4 for each time step. The names of the main physics submodules in Figure 2.3 called from inside the "A" loop are also included in parentheses.

2.4 Time Step Control

VICTORIA's time-step must be user input constrained to a material Courant condition that is defined here to include both the bulk gas convective and all diffusive transport terms. The time step should then be

$$\Delta t_{user} = \mathrm{Min}\;(\Delta t_{g,}\;\Delta t_{f}\;)\;, \tag{2.1}$$

where

$$\Delta t_{g} = \mathrm{Min}\left(\frac{\Delta r}{u}\;,\;\frac{\Delta r^{2}}{D_{max}}\;,\;\frac{\Delta z}{v}\;,\;\frac{\Delta z^{2}}{D_{max}}\right)\;, \tag{2.2}$$

$$\Delta t_{f} = \mathrm{Min}\left(\frac{\Delta rf^{2}}{D_{max}}\;,\;\frac{\Delta f}{D_{max}}\right)\;, \tag{2.3}$$

Δt_g is a bulk-gas Courant and cell diffusion time step limit, Δt_f is the fuel rod sub-domain and surface film diffusion time step limit, Δt_{user} is the resulting user-input maximum time step size. Δr and Δz are the radial and axial cell dimensions, u and v are the radial and axial gas velocity, and D_{max} is the largest of the species' diffusion coefficients in the geometry domain of interest. Δrf and Δf are the fuel rod minimum sub-domain radial dimension and the thickness of the structure surface film respectively. $D_{max}/\Delta rf$ can be thought of as a diffusion velocity, in this case the velocity with which the vapor species are transported across the fuel-cladding gap.. The Min function appearing in Equations (2.2) and (2.3) implies that the arguments are evaluated for all cells in the spatial domain and that the smallest value is then selected.

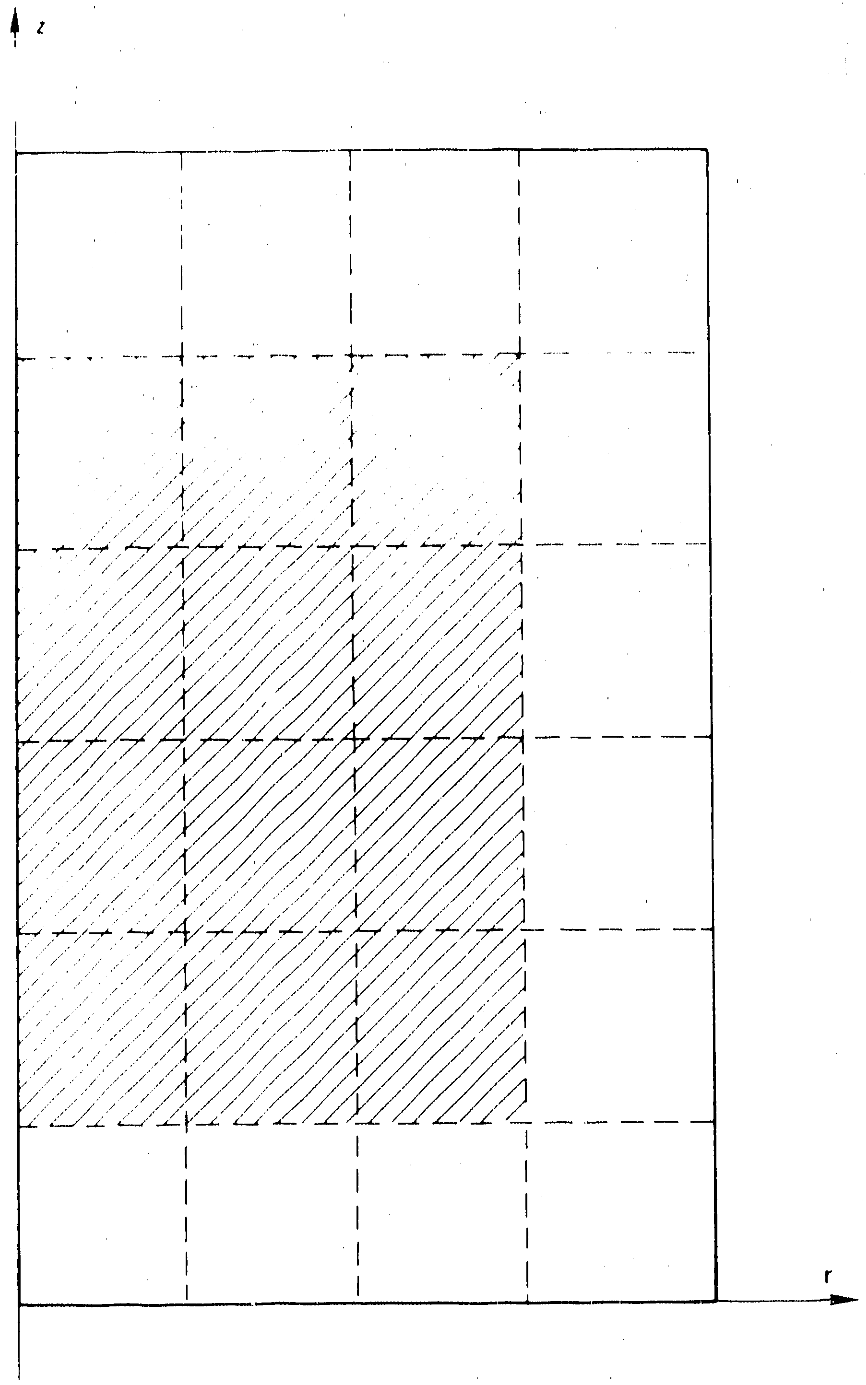


Figure 2.1. Simple mesh cell diagram dividing the reactor vessel into six axial levels and four radial rings.

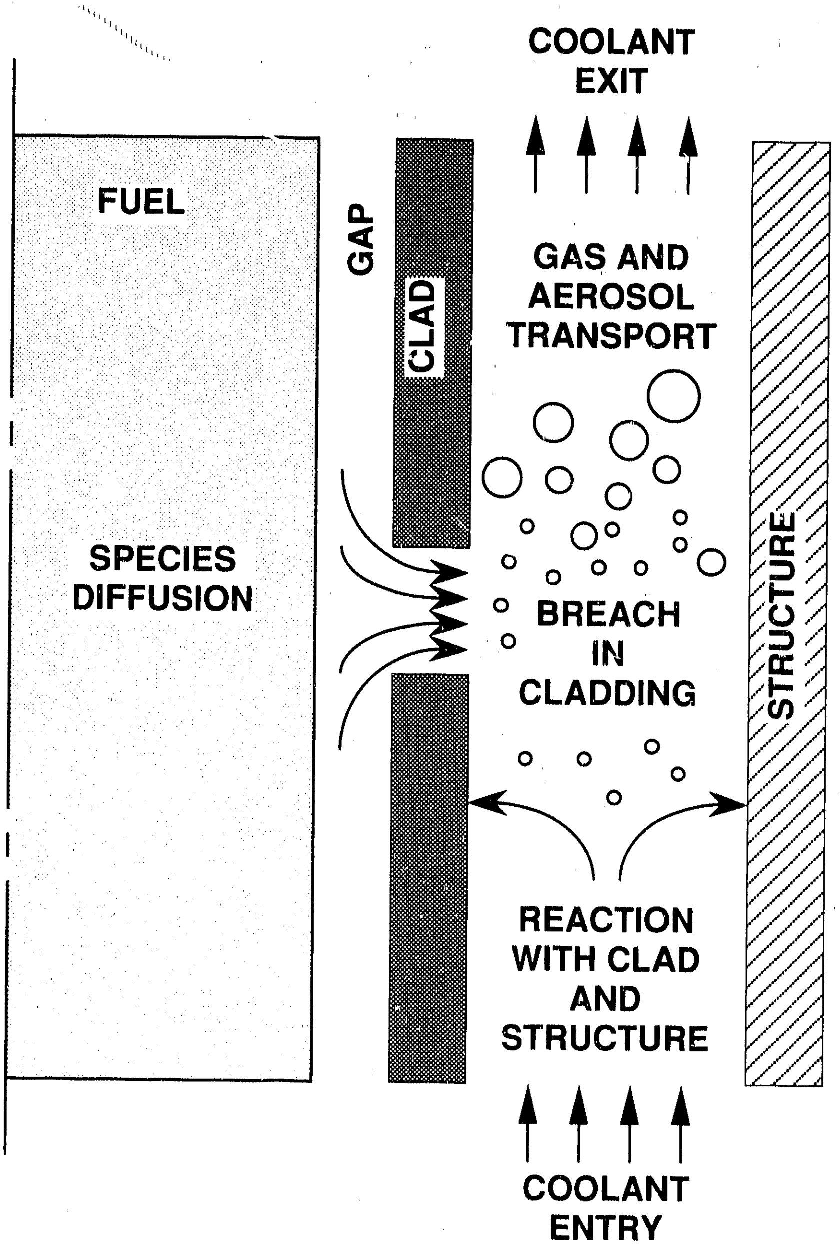


Figure 2.2. Schematic of the relevant geometry and phenomena related to species behavior and transport which are modeled within each gross two-dimensional mesh cell.

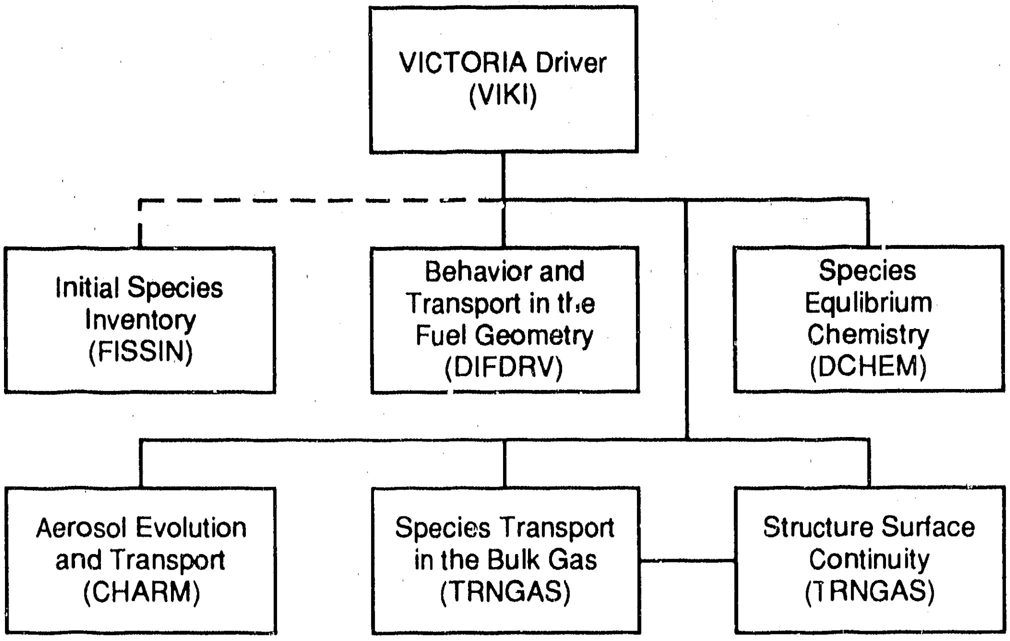


Figure 2.3. The program structure of the VICTORIA module. Each box depicts an area of modeling that solves its own internal system of equations coupled explicitly to the other modeling areas (other boxes). The dotted line indicates a call to initialize the problem.

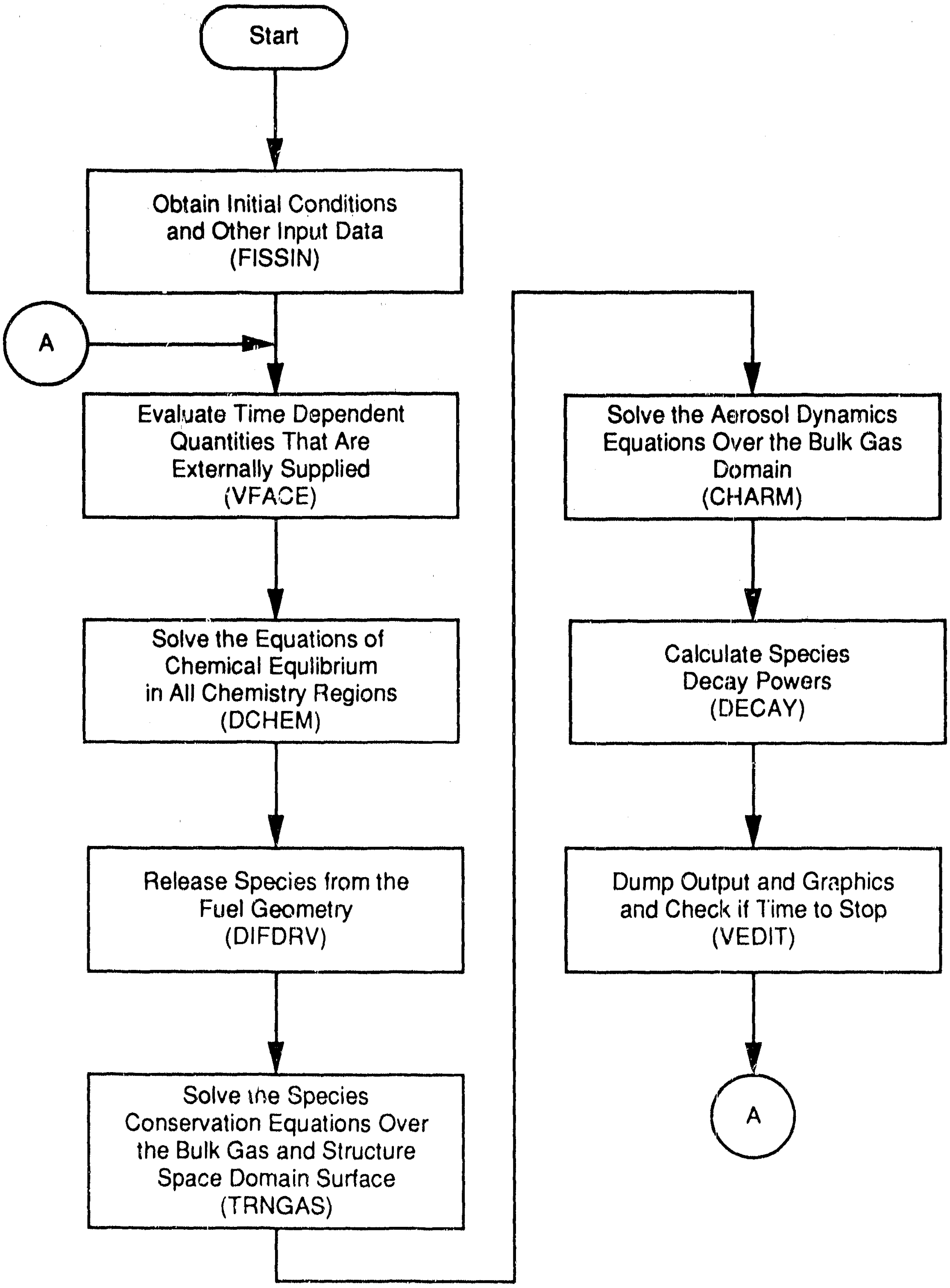

Start
Obtain Initial Conditions and Other Input Data (FISSIN)
A
Evaluate Time Dependent Quantities That Are Externally Supplied (VFACE)
Solve the Equations of Chemical Equlibrium in All Chemistry Regions (DCHEM)
Release Species from the Fuel Geometry (DIFDRV)
Solve the Species Conservation Equations Over the Bulk Gas and Structure Space Domain Surface (TRNGAS)
Solve the Aerosol Dynamics Equations Over the Bulk Gas Domain (CHARM)
Calculate Species Decay Powers (DECAY)
Dump Output and Graphics and Check if Time to Stop (VEDIT)
A

Figure 2.4. Flow chart of the logic flow in subroutine VIKI. Main subroutine names are indicated in parentheses.

3. SPECIES BEHAVIOR AND TRANSPORT WITHIN THE FUEL GEOMETRY

The fuel geometry for the VICTORIA code includes the fuel pellets and cladding. The only fuel geometry treated in the current version of the code includes intact (but possibly truncated) pellet stacks. These pellet stacks may be covered by metallic cladding, oxidized cladding, or partially oxidized cladding. Additionally, the cladding may be ruptured, fractured, partially removed or completely removed.

VICTORIA treats the fuel as consisting of two separate physical regions: The first region is the interior of the fuel grains; the second region can best be described as extra-granular and encompasses the open porosity, gap, and cladding along with their mutual and individual boundaries. Equilibrium chemistry and intra-granular solid-state diffusion are of concern in the interior of the fuel grains, while equilibrium chemistry and molecular transport are of concern in the open porosity, gap, and cladding.

Within the grains (which are treated as uniform spheres) diffusion by a Fick's Law equation, first proposed by Booth [3.1], is used to determine a source for the main transport equation that is solved within the intra-granular region. Other sources and sinks for the governing equation are generated by the equilibrium chemistry solver and by cladding oxidation and relocation models in the extra-granular region. The only transport mechanism considered in the current version of VICTORIA in the extra-granular region is diffusion. Any axial level that contains fuel pellets is considered to be homogeneous. Both gas phase and surface diffusion are dealt with depending on the physical state of the diffusing species.

The extra-granular region is divided into a number of sub-regions. These sub-regions are the open porosity, the gap (including the cladding inner surface), the cladding, and the fuel surface film. A typical extra-granular noding diagram is shown in Figure 3.1. The axial dimension for all cells and the axial noding (that is, the number of cells in the axial direction) is specified in the input. The open porosity, gap, and fuel and cladding have radial dimensions that are also input by the user. In addition to the physical dimensions, the user is required to set the number of mesh cells in the radial direction which are to be occupied by the open porosity and the cladding. The values assigned to these variables are based on a trade-off between computing time, time step limitations, and accuracy in the transport routines. A larger number of radial nodes gives greater accuracy but at the expense of computing time. The value set for the open porosity also determines the setting for the fuel grains. In the example shown in Figure 3.1, the open porosity occupies two radial mesh cells and the cladding occupies one. The gap always uses a mesh cell. The cladding-coolant interface cell has a radial film thickness that is fixed in the code. This value can be changed by the user by altering the block data subprogram DATGEN.

In a two-dimensional solution to the governing equation over this noding pattern, each mesh cell communicates with the cells immediately above and below it, along with those adjacent to it on the left and right. This pattern of communication is depicted by the arrows in Figure 3.1. The exception to this pattern occurs when there is no fuel element at an axial level. In such a case, there is no vertical communication within the fuel between this axial level and its neighbors. The example in Figure 3.1 has no fuel in axial level 4, and therefore there is no species transport from levels 3 or 5 to level 4.

3.1 Governing Equations

The transport processes within both physical regions of the fuel are treated by the solution of one general transport equation. This equation is solved over each region for each chemical species. The solution of the transport equation is obtained independently for each species at each time step.

The general transport equation VICTORIA solves is commonly known as Fick's Second Law. The driving force for diffusion is taken to be the concentration gradient exclusively; a source/sink term is included for intra-granular fission product transport leading to release from the grains to the extra-granular region. Chemical action leads to changes in the species concentrations and this provides a second source/sink term for the diffusive transport. The equation that is solved for each species is

$$\frac{\partial c_i}{\partial t} = -\nabla \bullet (-D_i \nabla c_i) + s_i^g + s_i^{chm} \tag{3.1}$$

where c_i is the molar concentration of the i^{th} species per cell volume, D_i is the species-dependent diffusion coefficient, s^g_i is the molar concentration source per unit grain volume of species i escaping from the fuel grains, and s_i^{chm} is the molar concentration source/sink due to the changes in concentration brought about by the equilibrium chemistry calculation. The values of the volumetric source terms and the form of the diffusion coefficient are dependent on the spatial region within the fuel, as well as on the species phase and identity. The dependencies of the source terms and the diffusion coefficient will be discussed in the sections that follow.

3.2 Fuel Intra-granular Processes

The initiation of the release of fission products from intact or rubblized fuel in a nuclear reactor accident takes place within the fuel grains. A variety of mechanisms have been postulated that can lead to the movement of fission products from the inside of the grains to the surface of the grain and the subsequent transport through the extra-granular portion of the fuel. In the current version of VICTORIA we consider primarily two intra-granular phenomena: equilibrium chemistry, which determines the speciation of the fission products within the solid matrix (and hence their effective mobility in this environment); and the the actual movement of fission products within the grains. In the current version of the code, only solid state diffusion of atomic species is considered. Both of these topics will be discussed in greater detail in the sections following.

3.2.1 Equilibrium Chemistry in the Fuel Grains

Within the fuel grains the concentration of fission products is sparse but not negligible. Experiments have shown that the fission products do not all exist in their elemental form within the lattice [3.2]. Since the motion of the fission products within the ceramic lattice can be affected by their chemical form, VICTORIA attempts to account for the form of the fission products.

The simplest approach to chemical speciation is obviously one of neglect; that is, the fission products exist only in atomic form within the lattice. Since this is not the actual case [3.2], a better approach is needed. A slightly more complex solution to the problem is the assumption of complete thermo-chemical equilibrium within a defined set of chemical species. VICTORIA uses a modified version of this more complex approach.

At present, the chemical equilibrium package in VICTORIA consists of a set of 167 chemical species made from 25 elements. The identities of the species and elements are given in Appendix A. In this model, chemical equilibrium is calculated for an ideal solution consisting of the elements present in the fuel at the time of interest. For a mixture as complex as irradiated ceramic fuel, the ideal solution assumption is probably not the best possible choice. Studies of irradiated fuel [3.2] have shown that there are several solid phases present, each containing a number of different species. It is not clear that each or any of these phases satisfies the conditions of equilibrium for an ideal solution model.

Therefore, another choice for the chemical equilibrium package would be some sort of non-ideal solution model based on data from studies of irradiated fuel. The solution would be a function of oxygen to metal ratio in the fuel and the concentrations of the various elements due to power history and burnup. Such a model would require corrections to the free energy data as a function of the temperature and the concentrations of all the species present. While all of the data for such a calculation is not available, it is possible to make guesses about the unavailable values. However, there still would be no guarantee of improved accuracy using this method and the increase in computing cost makes the implementation of such a model difficult to justify. Therefore VICTORIA uses the ideal solution model.

The small concentration of fission products within the lattice and the relatively low mobility of the atoms in the lattice raises questions about the validity of the full equilibrium approximation. A calculation based upon random walk statistics allows one to determine collision frequency using the solid-state diffusion coefficients of Oi [3.3], Oi and Takagi [3.4] and typical concentrations and temperatures. It can be shown that collisions between random fission product atoms take place on a 50 msec timescale.

This calculation does not take account of the requirements for the formation of a molecule involving more than two fission product atoms. For such a molecule to form, it is necessary for an atom to bind an existing chemical moiety which will not, in general, be thermodynamically favored. This precursor entity will therefore have a small concentration, which reduces the probability for the necessary collision to occur to a level below that of a three body collision. This small probability for the requisite collision reduces the likelihood of the formation of such molecules to the point were equilibrium is, at best, a tenuous assumption. This likelihood is even further reduced by the concept of orientation, that is, the atom not only has to find the existing molecule but the approach must be in a suitable direction for the new molecule to form. As a result of these considerations, VICTORIA does not allow non-oxide species which consist of more that two fission product atoms to form. The formation of oxides (for example, simple binary oxides, along with uranates, zirconates, and molybdates) containing fission products is allowed.

Species which consist of two similar or dissimilar fission product atoms are not subject to the same steric or collisional probability factors as the species consisting of more than two fission product atoms, which were discussed above. There is, however, a different steric constraint which must be taken into account. This constraint is a question of molecular size and the availability of a suitable lattice site for molecule formation. Non-oxides, in particular iodides or tellurides, consisting of two fission product atoms are, in general, much larger molecules than oxides. As a result, the ceramic lattice spacing is too small to allow formation of the molecules within the lattice. This size mismatch can be overcome by the existence of a suitable defect in the lattice or by the formation of a separate phase. The formation of a separate phase is, in our formulation of the problem, equivalent to the migration of the fission product atoms from the grains into the open porosity region of the fuel. The deviation from equilibrium of fission product diatoms has been observed experimentally. [3.5,3.6] VICTORIA has a modification to the free energy of the diatomic iodides and tellurides in the fuel grains to account for this observed deviation.

Oxides are observed in irradiated fuel [3.2] and there is no evidence that their concentration is not at thermodynamic equilibrium. However, note that these oxide species can exist in separate phases consisting of one or more species. This phase separation is clearly an indication that a single ideal solution is not present. The existence of the oxides does, however, demonstrate the need for a model to account for chemistry within the fuel grains.

The general treatment of the chemistry within the fuel grains follows a slightly different set of assumptions than that in other regions. The volume used is that of the fuel grains and the volume of the gas phase in this region is taken to be the same as that for the condensed phase. The reasoning for this treatment has to do with the concept of phases within a solid solution. An isolated atom or molecule within the lattice is termed gaseous or condensed based solely on its free energy and not on its actual physical phase. Under lattice conditions, all atoms or molecules are essentially condensed since there is not a separate region in which a bulk gas exists in equilibrium with the condensed material and occupies a different region of space. Therefore, the release models used for motion within the lattice do not differentiate between condensed or gaseous atoms. Additionally, molecules are considered immobile regardless of their physical or thermodynamic phase.

This last statement concerning the immobility of molecules within the fuel grains delineates the primary effect of chemical interaction on the release process. What is occurring chemically is a reduction of the concentration of fission product atoms. This reduction of the concentration provides a lessening in the number of atoms of a given element that are available for diffusion through the lattice, and hence for release, at any given time. Also, since the concentration gradient between the grain interior and grain surface is the driving force for the solid state diffusion process, the lessening of fission product atom concentration within the grai..s reduces the diffusion rate as well as the atomic availability and, therefore, meters the transport to the open porosity. A similar release effect could be produced if one used a smaller value of the diffusion coefficient and neglected chemical interaction within the fuel.

Neither the modified ideal solution chemistry that is presently used in the code nor a complete neglect of fuel chemistry (with concomitant modification of the solid state diffusion constants to account for the change in atomic concentration) is an accurate model of the actual physics that is occurring in the fuel leading to fission product release. A modification to the diffusion coefficients does not accurately reflect the effect of the concentration of the fission products on the release process. Modifying the coefficients assumes a smooth change in the atomic concentration, whereas the effect of chemistry is to make this change not smooth. It is likely, however, that either of these models, combined with an accurate time and temperature dependent model of fuel morphology, will reproduce observed releases to an acceptable accuracy. Furthermore, this solution to the problem does reflect the level of experimental knowledge of the actual process of fission product release from degraded fuel.

3.2.2 Solid-State Diffusion

Within the fuel grains, transport of fission products is considered to take place by solid state diffusion through the lattice. The driving force for this motion is the concentration gradient between the interior and the surface of the fuel grains. The resulting concentration at the surface of the fuel grain is used as a source term for the extra-granular transport Equation (3.1). The source term, per unit fuel grain volume, of species "i" at the grain surface is determined using Booth's analysis [3.1]. Booth derived an expression for the fraction of a species that had diffused out of a fuel grain by a specified time, which he referred to as f. Using Booth's notation, the amount of species i that diffuses out from the grain between time planes α and $\alpha+1$ is given by $f_i^{\alpha+1} - f_i^{\alpha}$. If this difference were multiplied by the difference in the initial concentrations of the grain and the grain surface, the result would be a volumetric source term for release from the fuel grain. However, VICTORIA retains the current concentrations rather than those at the initial time. This problem is easily remedied by noting that $1 - f_i^{\alpha}$ is the fraction of the initial concentration difference that is retained at time plane α. Therefore, the expression for the source term over the time step, Δt, can be written using Booth's notation as follows:

$$s_i^g \,\Delta t = \left[c_{ig}^{\alpha} - c_{is}^{\alpha}\right] \left[\frac{f_i^{\alpha+1} - f_i^{\alpha}}{1 - f_i^{\alpha}}\right] \qquad (3.2)$$

where c_{ig} is the concentration within the grain, c_{is} is the concentration at the grain surface and α represents the current time plane. The diffused fraction is defined

when $\tau < 1.0$ as

$$f_i^\alpha = 6 \left[\frac{\tau_i^\alpha}{\pi^3} \right]^{1/2} - 3 \frac{\tau_i^\alpha}{\pi^2} \quad , \qquad (3.3)$$

and when $\tau > 1.0$ as

$$f_i^\alpha = 1.0 - \frac{6 \, e^{-\tau_i^\alpha}}{\pi^2} \quad , \qquad (3.4)$$

where τ is a dimensionless time and is given by

$$\tau_i^\alpha = \frac{\pi^2 D_i t^\alpha}{r^2} \quad . \qquad (3.5)$$

where r is the radius of the grain and t is the current time of the calculation. The diffusion constant D_i is taken to be a simple function of the temperature for each diffusing species. The functional form used is

$$D_i = D_{oi} e^{-\theta_i/T} \qquad (m^2/s) \qquad (3.6)$$

where D_{oi} is the pre-exponential factor, and θ_i is the activation energy for diffusion. The default values of D_{oi} and θ_i are given in Appendix A, and the user has the option of modifying them in the initial fuel grain input data. It is our opinion that single crystal diffusion coefficients are the best starting place for the models in VICTORIA. Diffusion out of individual grains is effectively the same as that out of a small single crystal. As irradiation progresses, however, the individual grains become highly defected and one would expect that the value of the diffusion coefficients would change. This change could be either an increase or a decrease depending on whether the defects serve as traps or tunnels for the diffusing atoms. We intend that as the code matures a uniform set of these parameters can be determined.

The species for which solid-state diffusion is allowed are listed in the fuel grain input. We use only UO_2 and the condensed phase atomic species whose concentrations are determined from the isotopic depletion calculations. Generally, molecular species are observed to diffuse through solids at rates much slower than the rates of diffusion for atoms and would not be expected to contribute to release. The one possible exception to this would be the basic lattice constituent, UO_2, which can be assigned a diffusion coefficient to account for sublimation of the lattice. In this case, the activation temperature should correspond to the temperature of sublimation of UO_2. For the high temperature regime typically operative in a severe accident, a considerable amount of fuel could be sublimed. This amount is dependent on the removal of the cladding and on conditions where transport limitations will allow for the removal of the products of the sublimation. If the user wants to consider sublimation, a model also needs to be added to VICTORIA to account for the decrease in grain size due to the sublimation of the lattice and the effect of this change in grain size on the diffusion process.

3.3 Fuel Open Porosity Processes

Once a fission product species is at the surface of the grains in either a condensed or gaseous form, the extra-granular transport processes of Equation (3.1) take over. In the current version of VICTORIA, transport of gaseous species within the open porosity is by gas phase diffusion. Condensed species in the open porosity are transported by a surface diffusion mechanism. In other geometric configurations either solid-state or Knudsen diffusion mechanisms could be operative [3.7]. The physical state of a species is determined by chemical equilibrium considerations. The nature of the transport process depends on both the physical state of the particular species and the geometry of the particular region being considered.

3.3.1 Chemical Equilibrium

Within the fuel open-porosity region, significant concentrations of fission product atoms that have migrated out of the fuel grains can develop. Since the migration is a random process, these atoms are well mixed within the open porosity. Given this well-mixed volume with gaseous elements in intimate contact with those residing on the surface, the achievement of chemical equilibrium in this case is considered a reasonable approximation.

The calculation of equilibrium in the open porosity region is carried out using the fuel temperature and all the species in the data set. As in the fuel grains, this calculation can have a significant effect on transport. The diffusion of gaseous and condensed species proceeds at different rates, and therefore chemical speciation can have a significant impact on the transport of fission products out of the open porosity.

3.3.2 Diffusive Transport

Transport of chemical species within the open porosity is treated directly by Equation (3.1). The only means of transport currently allowed in this region is diffusion due to concentration gradients. Convective transport due to pressure gradients is not considered. The determination of the diffusion coefficient to be used in the solution of the governing equation is dependent on the chemical identity and physical phase of the transported species in question.

For condensed phase species, diffusion is postulated to take place along the surfaces of the grains, and the coefficient for the governing equation is given by

$$D_i = 40e^{-54125/T} \quad (m^2/sec) \qquad (3.7)$$

where T is the temperature (in Kelvin) of the region in question. This value of the diffusion coefficient was developed [3.8] for surface self-diffusion of UO_2, but is taken to be valid for all condensed phase species in the present version of VICTORIA.

For gas phase atoms and molecules, VICTORIA uses a diffusion coefficient based on the Chapman-Enskog model [3.9]. The value of the coefficient is given by

$$D_i = 1.8583x10^{-7}T^{3/2} \frac{[1/M_i+1/M_m]^{1/2}}{P\sigma^2_{im}\Omega_{im}} \quad (m^2/sec) \qquad (3.8)$$

where T is the temperature in Kelvins, P is the pressure in atmospheres, M_i is the average molecular weight of a diffusing species, M_m is the molecular weight of the gas in the cell, σ_{im} is the Lennard-Jones interaction distance in meters, and Ω_{im} is a dimensionless collision integral. The values of Ω_{im}, σ_{im}, M_i, and M_m are defined in data statements. In VICTORIA the values for σ_{im} and Ω_{im}, 4.04 and 0.775 respectively, were determined from an average for small molecules at high temperatures using values from Bird [3.9]. The value of M_m is set for a mixture of gases with a mean molecular weight of 3. The value of M_i is set in a data statement to the molecular weight for the gaseous fission product species being diffused. A modification for determining the value of M_m during execution would consume a large amount of run time, for a small increase in accuracy and was not implemented. The values of T and P are determined during execution.

While it is true that all of these parameters affect the calculated diffusion coefficient, the effects are small compared to the uncertainty in the pressure and temperature dependence of this formulation. A small error in the power of either of these parameters will cause a large change in the diffusion coefficient for the temperatures and pressures operative in a severe reactor accident. The correct functional form of D_i in the high temperature and pressure regime is not known. Further comparison of the code against experimental results may help to determine the correct form of D_i.

3.3.3 Numerical Scheme

The solution of the governing equation for species transport within the fuel geometry (Equation [3.1]) is performed on a grid as shown in Figure 3.1. The chemistry source term is treated as an instantaneous change to the species concentrations in a given cell brought about by the equilibrium calculation. The chemistry source term over the time step, Δt, is of the form

$$s_i^{chm} \Delta t = \delta(t - t^\alpha) \; (c_i - c_{io}) \qquad (3.9)$$

where c_i is the concentration of species i at time t^α and c_{io} is the concentration of species i at time $t^{\alpha-1}$. The Dirac delta function, δ, is defined as

$$\int_{t^\alpha-\epsilon}^{t^\alpha+\epsilon} \delta(\tau - t^\alpha) \; d\tau = 1 \quad . \qquad (3.10)$$

This function ensures that the total concentration change of species i will occur from time $t^{\alpha-1}$ to time t^α.

The source from the fuel grains s_i^g (see Equation [3.2]) is simply added into Equation (3.1) as an explicit source term. The remainder of the governing equation is discretized using a second-order accurate finite differencing scheme. The equation written in two-dimensional cylindrical coordinates is

$$\frac{\partial c_i}{\partial t} = \frac{1}{r}\frac{\partial}{\partial r}\left(rD_i \frac{\partial c_i}{\partial r}\right) + \frac{\partial}{\partial z}\left(D_i \frac{\partial c_i}{\partial z}\right) \quad . \qquad (3.11)$$

In order to discretize this equation, we need to stipulate indices for the cells over which the equation is being solved. The indexing for a sample cell is shown in Figure 3.2. As Figure 3.2 shows, the r,z spatial coordinate indices are m,n respectively, and the notation m+, n+ or m−, n− is shorthand for m+1/2, n+1/2 or m−1/2, n−1/2, respectively. Using this indexing and the fact that the diffusion coefficient is a function of axial level only, we can discretize Equation (3.11) in both time and space. For any given cell m,n the radial component of this equation is

$$\frac{1}{r}\frac{\partial}{\partial r}\left[rD_i\frac{\partial c_i}{\partial r}\right] = \frac{1}{r_m}\left[\frac{r_{m+}D_{m+,n}\left(\frac{\partial ci}{\partial r}\right)_{m+,n} - r_{m-}D_{m-,n}\left(\frac{\partial ci}{\partial r}\right)_{m-,n}}{r_{m+} - r_{m-}}\right] \qquad (3.12)$$

and the axial component is

$$\frac{\partial}{\partial z}\left[D_i\frac{\partial c_i}{\partial z}\right] = \frac{D_{m,n+}\left(\frac{\partial ci}{\partial z}\right)_{m,n+} - D_{m,n-}\left(\frac{\partial ci}{\partial z}\right)_{m,n-}}{z_{n+} - z_{n-}} \quad . \qquad (3.13)$$

The inner differential terms are given by the gradient at the boundary in question. For each species i, these terms are

$$\left.\frac{\partial c_i}{\partial r}\right|_{m+,n} = \frac{c_{m+1} - c_m}{r_{m+1} - r_m} \quad , \qquad (3.14)$$

$$\left.\frac{\partial c_i}{\partial r}\right|_{m-,n} = \frac{c_m - c_{m-1}}{r_m - r_{m-1}} \quad , \qquad (3.15)$$

$$\left.\frac{\partial c_i}{\partial z}\right|_{m,n+} = \frac{c_{n+1} - c_n}{z_{n+1} - z_n} \quad , \qquad (3.16)$$

$$\left.\frac{\partial c_i}{\partial z}\right|_{m,n-} = \frac{c_n - c_{n-1}}{z_n - z_{n-1}} \quad , \qquad (3.17)$$

The values of these gradients need to be fixed at the boundaries of the grid, and are given for each species i by

$$\left.\frac{\partial c_i}{\partial r}\right|_{m-,n} = 0 \quad \text{at } r = 0 \quad , \qquad (3.18)$$

$$\left.\frac{\partial c_i}{\partial r}\right|_{m+,n} = 0 \quad \text{at } r = r_{max} \; , \qquad (3.19)$$

$$\left.\frac{\partial c_i}{\partial z}\right|_{m,n-} = 0 \quad \text{at } z = 0 \quad , \tag{3.20}$$

$$\left.\frac{\partial c_i}{\partial z}\right|_{m,n+} = 0 \quad \text{at } z = z_{max} \ . \tag{3.21}$$

Along with the boundary conditions on the gradients, it is also necessary to consider the volume occupied by the open porosity and the degree of interlinkage that is present in the open porosity. Since it is the total number of molecules of a given species rather than the concentration that must be conserved in the calculation, we need to multiply by the volume of the region of interest. The volume used here is that of the entire cell (including that occupied by fuel or cladding) weighted by a factor to account for the excluded volume. Within the fuel pellets, this weighting factor is taken as being the fraction of the volume not occupied by fuel grains (given by the fractional theoretical density for the fuel). For the gap and surface-film regions, the volume is considered to be entirely open. The cladding volume fraction is fixed by the user depending on such factors as cladding rupture, melting, and so on. In addition to the open volume consideration, it is necessary for the user to account for the fact that the fuel porosity may not be fully interconnected, or that the gap may be pinched, or that the cladding may form an incomplete barrier to fission product migration.

The parameter holding the volume correction term is called α. α is contained in the "LALF" array and is calculated for each cell in subroutine VFACE.

The parameters that contain the interconnectedness between two adjacent cells are called β_r and β_z. They are also calculated in VFACE (the user may wish to modify this routine to affect the species transport) and sorted in the arrays "LBETAR" and "LBETAZ". A volume for β is calculated corresponding to each arrow in Figure 3.1. The values used for the β's are set at 0.0001, except for fuel frothing and cladding failure. The fuel frothing expansion is calculated, as shown, using a model developed from ACRR-ST1 data that is a function of temperature only. The cladding failure model is based on either the failure temperature or the remaining concentration of Zr.

For the fuel

$\beta = 0.0001$ when $T < T_{froth}$

$\beta = 0.35$ when $T > T_{froth} + 200$ K

β linearly interpolated over the 200 degree range

$T_{froth} = 2200$ K;

For the failure of the cladding

$\beta = 0.0$ until $T_{clad} > T_{fail}$

or when $c_{Zr} < 0.5$

VICTORIA solves the set of equations developed above in two steps. First, all the radial transport terms are determined. The result is a tridiagonal matrix which is solved using a standard mathematical software routine (described in Appendix C). The solution of this tridiagonal matrix is then used as the new right hand-side vector for the second step in which the axial terms are calculated. A new tridiagonal matrix results that is solved using the same routine. The complete solution is then used to update all the stored concentrations for the fuel region.

The fuel region is treated as a completely closed system in these calculations. The boundary conditions are all zero gradient. There is, however, a hidden explicit source term. The fuel surface film region is treated as part of both the fuel and the gas flow regions. This gives the fuel region an explicit source term with information that is at the previous time step.

The use of a standard system tridiagonal solver insures that the tridiagonal equations are solved to machine accuracy rather than some user-defined accuracy as would be the case in an iterative scheme. Several trials have been run reversing the order of solution for the radial and axial transport equations. No difference in the solution due to the order has been found.

3.4 Fuel Cladding Gap Process

Once a fission product species reaches the surface of the fuel pellet stack (or an exposed debris bed surface or other surface external to the fuel), transport through a different medium must be considered. With intact fuel there is a fuel-cladding gap which provides a path for axial gas phase diffusion but may not provide a suitable venue for condensed-phase diffusion. If the fuel-cladding gap is closed, then there would be a suitable surface diffusion site for the condensed phase. For other potential geometries there are obstructions such as crusts or structural materials, which will present changes in transport properties and must be treated. In any new geometry, one must also consider the effect upon interfaces between fuel, gap, cladding, and the bulk gas region.

3.4.1 Equilibrium Chemistry

For any of the fuel configurations treated in VICTORIA, the ex-pellet volume contains at least one region that has a more open geometry than that interior to the pellets. This open geometry allows for a well-mixed gas phase above a condensed phase lying on one or more interfacial surfaces. Given the well-mixed nature of the gas phase in this region, the assumption of chemical equilibrium for this phase is a reasonable one. Furthermore, the ready and open access of the gas phase to the condensed phase in this region makes the assumption of chemical equilibrium between the gas and condensed phase acceptable as well.

There still remains the question of whether or not the condensed phase is a single equilibrium phase or if it consists of several independent phases, each composed of one or more species at equilibrium. The answer to this question depends on the composition and area of the surface and the rates of reactions and rates of transport of the various species in the condensed phase. At the present time, the gaseous and condensed phase species are treated as one equilibrium system. The equilibrium calculation in this region includes both the gaseous species residing in the gap and the condensed species that are found in the innermost cladding cell. Once the position of equilibrium is determined, the species are partitioned such that gaseous species are only found in the gap and condensed species are placed in the innermost cladding cell. This treatment of the chemistry provides a transport mechanism for some elements, which can be explained by considering a gaseous species in the gap reacting with some other species to form a third species which condenses onto the inner cladding surface.

3.4.2 Transport

Fission product transport across the fuel-cladding gap is similar to that used in the open porosity region. Gas phase species are treated the same as they are in the open porosity with the exception that β, the interfacial, interconnected porosity fraction for the gap, is always taken to be unity. β can be altered by the user in the VFACE subroutine to account for a closed or partially-closed gap. The diffusion coefficients used for the gas phase species are calculated in the same way as those used in the open porosity and are subject to the same restrictions. The diffusion of condensed species across the gap is not allowed; for condensed species the diffusion coefficient is set to zero in the gap region. This is to account for the fact that in the absence of a closed gap there is no physical mechanism to allow a condensed species to migrate across this space. There is no continuous surface for migration like that available in the fuel pellet open porosity. We anticipate that a surface diffusion mechanism will be added for a gap cell in which there is contact between the cladding and the fuel pellets.

3.5 Information for the Code User

We caution the user regarding the choice of a time step, as was indicated in section 2.4, gas diffusion across the gap (where the cell width is smallest) is the most likely place where a time step limit will be found in the fuel. With regard to solid state diffusion out of the solid grains, only when the problem is at high temperatures (> 2800 K) with small grain sizes (< 10 μm) will there be potential time step problems. It is probably worthwhile to be certain that the solid-state diffusion velocity from the highest input temperature in the calculation does not allow diffusion distances greater than the grain radius (10 μm) in a single time step. Violating this condition has been found to cause non-conservation of mass and can even lead to negative masses under extreme conditions.

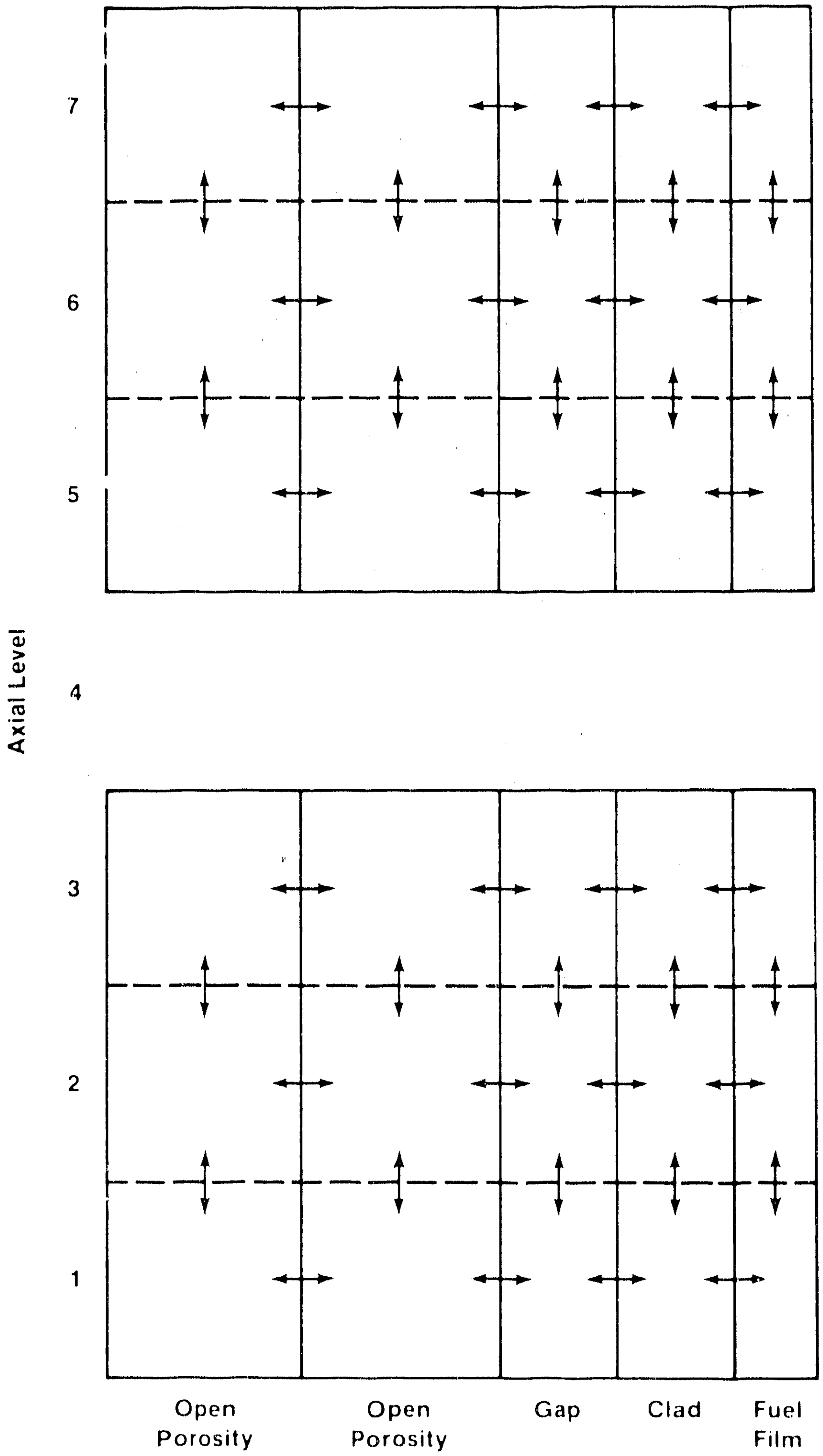


Figure 3.1. Sample of the type of mesh used in the VICTORIA diffusion calculation.

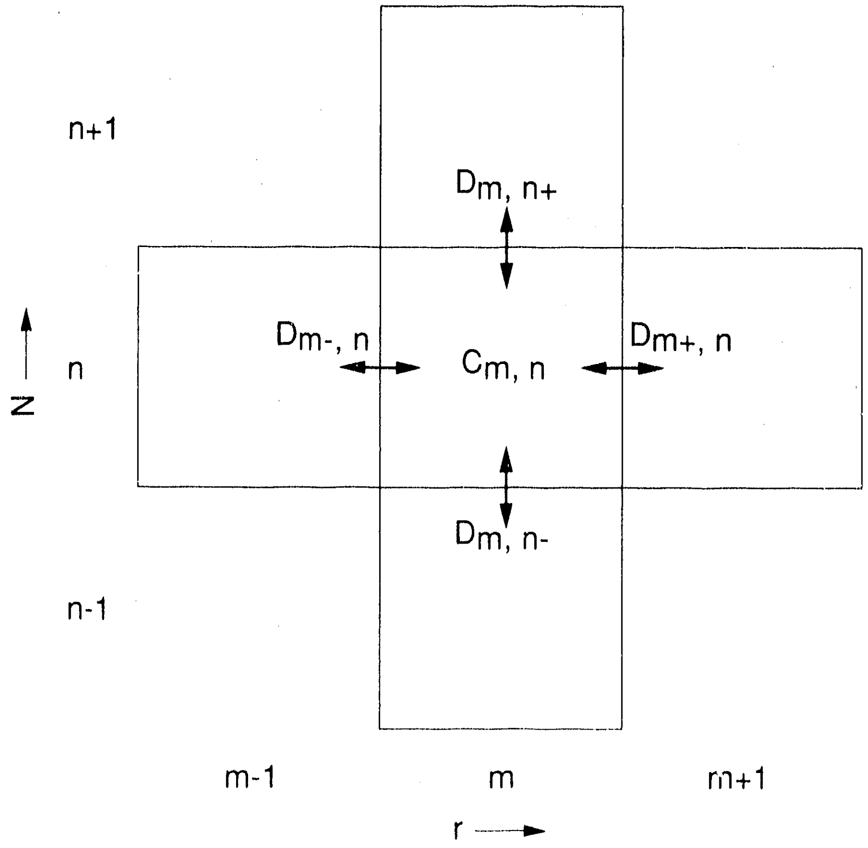


Figure 3.2. Species molar transport within the fuel rod cell m,n.

4. BULK GAS SPECIES TRANSPORT

The conservation of gaseous species in the bulk gas flow within the reactor vessel is explained in this section. The relevant two-dimensional conservation equation is first presented in exact form, followed by the approximation of the partial derivatives with finite differences. This yields a large system of algebraic equations (one equation for each species) that must be solved at every time-step. Although not explicitly indicated, the nominal functional dependence of all the dependent variables is with respect to time and two-dimensional space.

The geometry involved in discretizing the continuity equation is a combination (and idealization) of that shown in Figures 2.1 and 2.2. The overall geometry is that of concentric cylindrical rings. Transport is achieved by convection and/or diffusion between adjacent rings within an axial level, and between adjacent axial levels within a radial ring. The addition of fuel and/or structures with a given cell adds the requirement for transporting mass to and from the fuel or structures.

The fuel surface film is considered to be located on a vertical structure with a surface area that is calculated from the fuel dimensions. The annular volume of film surrounding the fuel has a constant film thickness that has been set, in a data statement, to 0.0005m. Transport between the fuel surface film and the bulk flow channel is by diffusion. This film volume is also considered part of the fuel geometry. As a result, it can serve as an explicit source term for the bulk gas calculation. This is the transfer point for fission products between the fuel and the bulk gas flow regions.

The structure's surface film region can be either a vertical (wall) or horizontal (floor or ceiling) surface within the cell. The surface areas of all three types of film regions, for each cell, must be entered in the input deck. The thickness of the surface film is given by the same parameter, delta (Δ_f), used for the fuel surface film (see Figure 4.1). The transport equations model diffusive transport of gaseous species between the gas flow and each of the three types of surface areas.

4.1 Species Conservation Equation

The transport of species in the coolant channel is governed by the flux of species between the fuel film (f), the bulk gas (g), and the structure films (s), as shown in Figure 4.2. The transport equations are written in the following two-dimensional cylindrical coordinates for each species, i:

$$\frac{\partial c_{fi}}{\partial t} dV_f = - \frac{\partial}{\partial r} (-D_i \nabla c_{fi} A_{fr}) dr - \frac{\partial}{\partial z} (-D_i \nabla c_{fi} A_{fz}) dz$$

$$+ s_{if}^{chm} \quad ; \tag{4.1}$$

$$\frac{\partial c_{gi}}{\partial t} dV_g = - \frac{\partial}{\partial r} (u c_{gi} A_{gr} - D_i \nabla c_{gi} A_{gr}) dr$$

$$- \frac{\partial}{\partial z} (v c_{gi} A_{gz} - D_i \nabla c_{gi} A_{gz}) dz + s_{ig}^{chm}$$

$$+ \frac{\partial}{\partial r} (-D_i \nabla c_{fi} A_{fr}) dr + \frac{\partial}{\partial z} (-D_i \nabla c_{fi} A_{fz}) dz$$

$$+ \frac{\partial}{\partial r} (-D_i \nabla c_{si} A_{sr}) dr + \frac{\partial}{\partial z} (-D_i \nabla c_{si} A_{sz}) dz \quad ; \tag{4.2}$$

$$\frac{\partial c_{si}}{\partial t} dV_s = - \frac{\partial}{\partial r} (-D_i \nabla c_{si} A_{sr}) dr - \frac{\partial}{\partial z} (-D_i \nabla c_{si} A_{sz}) dz$$

$$+ s_{is}^{chm} \quad , \tag{4.3}$$

volume x designated as the films or bulk gas. D_i is the molecular diffusion coefficient for the species in each volume, A is the interfacial area for transport, and u and v are the velocities in the radial and axial directions, respectively.

The chemistry source per unit time for each volume x, s_{ix}^{chm} appearing in Equations (4.1-4.3), are cast in terms of a delta-function adjustment made to the species initial condition at the beginning of each time-step

$$s_{ix}^{chm} \Delta t = \delta\{t - t^{\alpha}\} (c_{xi} - c_{xio}) \quad , \tag{4.4}$$

where the Dirac delta function integrates to one at each time t^{α} in the following manner:

$$\int_{t^{\alpha}-\epsilon}^{t^{\alpha}+\epsilon} \delta\{\tau-t^{\alpha}\}d\tau = 1 \quad . \qquad (4.5)$$

The intent behind the form of Equation (4.5) is that the effect of an equilibrium chemistry calculation at the beginning of each VICTORIA time step is modeled by discontinuously adjusting the concentration before chemistry (represented by c_{xi_o}) to its new value after chemistry (represented by c_{xi}). After this adjustment of the species initial values, Equations (4.1-4.3) are then integrated over the time step in the usual finite difference fashion as will be described below. The changes to the gas species concentrations described by Equation (4.4) not only account for chemical changes among gaseous species, but also includes molecular exchange with aerosol particles due to phase change and chemical reaction.

4.2 Numerical Scheme

The spatial partial derivatives appearing in Equation (4.1-4.3) are approximated using first-order-accurate, donor-cell differences for the convective terms, and second-order-accurate central differences for the diffusive terms.

In the difference equations given below, the time (t) index is α, the r,z space coordinate indices are m,n respectively, and the notation m+, n+ or m-, n- is shorthand for m+1/2, n+1/2 or m-1/2, n-1/2, respectively. The absence of the superscript α and the subscripts m and n was done intentionally to emphasize those terms that are not a α,m, and n, respectively. The reader is referred to Figure 4.2 for an illustration of the space discretization scheme.

The bulk gas convective derivatives from Equation (4.2) are approximated first as:

$$\frac{\partial}{\partial r}(uc_{gi}A_{gr})\,dr \sim A_{m+}u_{m+}[c_{gi}f_{m+} + c_{gim+1}g_{m+}]$$
$$- A_{m-}u_{m-}[c_{gim-1}f_{m-} + c_{gi}g_{m-}] \quad , \qquad (4.6)$$

$$\frac{\partial}{\partial z}(vc_{gi}A_{gz})\,dz \sim A_{n+}v_{n+}[c_{gi}f_{n+} + c_{gin+1}g_{n+}]$$
$$- A_{n-}v_{n-}[c_{gin-1}f_{n-} + c_{gi}g_{n-}] \quad . \qquad (4.7)$$

where the donor-cell parameters are defined as

$$f_{m\pm} = \left\{\begin{matrix} 1 \text{ if } u_{\pm} \geq 0 \\ 0 \text{ if } u_{\pm} \leq 0 \end{matrix}\right\} \quad , \tag{4.8}$$

and

$$g_{m\pm} = 1 - f_{m\pm} \quad . \tag{4.9}$$

The convective derivatives for the surface films, those terms that would allow film flow from cell to cell, are small compared to the bulk flow terms. Therefore, setting the film convection to 0.0 introduces only a small error and reduces the computational time.

The diffusive terms are next approximated with central differences:

$$\frac{\partial}{\partial r}\left(-D_i A_{gr} \frac{\partial c_{gi}}{\partial r}\right) dr \sim - A_{m+} D_{im+} \left[\frac{c_{gm+1} - c_{gm}}{r_{m+1} - r_m}\right]$$

$$\Bigg) \qquad + A_{m-} D_{im-} \left[\frac{c_{gm} - c_{gm-1}}{r_m - r_{m-1}}\right] \quad , \tag{4.10}$$

$$\frac{\partial}{\partial z}\left(-D_i A_{gz} \frac{\partial c_{gi}}{\partial z}\right) dz \sim - A_{n+} D_{in+} \left[\frac{c_{gn+1} - c_{gn}}{z_{n+1} - z_n}\right]$$

$$+ A_{n-} D_{in-} \left[\frac{c_{gn} - c_{gn-1}}{z_n - z_{n-1}}\right] \quad . \tag{4.11}$$

Diffusion from the fuel film is

$$\frac{\partial}{\partial r}\left(-D_i A_{fr} \frac{\partial c_{fi}}{\partial r}\right) dr \sim - A_{fr} D_{im} \left[\frac{c_{fm} - c_{gm}}{\Delta_f}\right] \quad , \tag{4.12}$$

$$\frac{\partial}{\partial z}\left(-D_i A_{fz}\frac{\partial c_{fi}}{\partial z}\right)dz \sim - A_{fn}D_{in+}\left[\frac{c_{fn+1} - c_{fn}}{z_{n+1} - z_n}\right]$$

$$+ A_{fn-}D_{in-}\left[\frac{c_{fn} - c_{fn-1}}{z_n - z_{n-1}}\right] \quad . \qquad (4.13)$$

Diffusion from the structure film is:

for the walls

$$\frac{\partial}{\partial r}\left(-D_i A_{sr}\frac{\partial c_{si}}{\partial r}\right)dr \sim - A_w\, D_{is}\left[\frac{c_{sm} - c_{gm}}{\Delta_f}\right] \quad , \qquad (4.14)$$

and for the floors and ceilings

$$\frac{\partial}{\partial z}\left(-D_i A_{sz}\frac{\partial c_{si}}{\partial z}\right)dz \sim - (A_{cl} + A_{fl})\ D_{is}\left[\frac{c_{sn} - c_{gn}}{\Delta_f}\right] \; . \qquad (4.15)$$

Lastly, the time derivative of Equations (4.1-4.3) is differenced as

$$\frac{\partial c_{xi}}{\partial t} \sim \frac{c_{xi}^{\alpha+1} - c_{xi}^{\alpha}}{\Delta t} \qquad (4.16)$$

where $\alpha+1$ refers to the new time and Δt is the integration time step. The distances used in the denominators of the film to bulk gas equations (Equations [4.12,4.14 and 4.15]) are taken to be the distance from the surface in question to the middle of the bulk gas. Note that the diffusion coefficient in the above difference equations can be defined at the boundary between cells. Since this quantity is typically calculated from cell-centered information using Equations (3.4-3.5), a space-averaged value is needed. The use of the same boundary value for D_i by juxtapositioning cells is necessary to avoid a non-conservative scheme.

Substituting the sources (Equation [4.4]), and the difference equations (Equations [4.6-4.16]) into Equations (4.1-4.3) and approximating the differential volume dV with a finite cell volume ΔV, produces an algebraic equation set containing the unknown species concentrations c_g, c_f, and c_s for each spatial cell. Once appropriate boundary conditions are specified, and since the concentrations are being implicitly solved for, the resulting system of equations can then be solved to give the concentration distribution over the entire spatial domain.

Note that the solution of the above system assumes a knowledge of geometric, chemical, and transport coefficient information. For example, the diffusion coefficient appearing in Equations (4.1-4.3) uses the Chapman-Enskog formula [3.9], which requires temperature and pressure information for evaluation. The calculation of the diffusion coefficients in this section of the code again uses average values for the Ω and σ parameters, as explained in section 3.3.2. The masses, however, are handled differently. The mass of the bulk gas is calculated from the hydrogen and steam concentrations in the cell, and a molar average mass is used. The mass of each diffusing species is used so that a different diffusion coefficient is calculated for each diffusing species. In other words, temperature, pressure, and velocity field information provided by some means external to VICTORIA must be available before the bulk gas continuity problem can be solved.

Boundary conditions must be provided by the user at all the physical edges of the mesh (at the top and bottom of the system as well as at the inner edge of the innermost radial ring and at the outer edge of the outermost radial ring). The boundary conditions can take on either of two forms: a zero concentration gradient for all species across a given boundary, or there can be a source for all species at a given boundary. A zero source implies a zero concentration gradient. A fixed source means that the concentration in a dummy cell outside the boundary is fixed and can be drawn on in the calculation. This source is effectively replenished at each time step.

At each time step, then, a seven-stripe matrix of equations must be solved for the species concentrations. The seven stripes are due to the maximum number of possible interaction terms, six (upper, lower, left and right bulk gas and the fuel and structure films), and the bulk gas term at the cell being calculated. VICTORIA originally solved the full matrix using a Gaussian elimination system subroutine; but this used a significant amount of computer time. VICTORIA now solves this matrix by dividing the matrix into three tridiagonal matrices that are solved sequentially using a mathematical software subroutine, SGTSL, described in Appendix C.

The sequential decomposition of the seven-strip matrix is in the order: axial transport, ring-to-ring transport, and finally, fuel surface film and structure surface film interactions. Because VICTORIA treats boundary conditions as source terms in the equation set, the effect of these source terms is dependent upon the order of solution. If the film regions, where there is a difference in concentration between the film and the source, were solved first, the source term would cause significant changes in the film species concentration which, in turn, causes numerical instabilities to arise. By VICTORIA calculating the axial transport first, where there is only a small difference between the bulk gas and the boundary condition, numerical instabilities do not occur.

At the top of the uppermost axial level, there is a formalism in the code to integrate the total mass of each gaseous species that passes this boundary. This integration considers transport across this boundary by both convection and diffusion.

4.3 Information for the Code User

In the bulk gas continuity calculation, the time step limitations tend not to be the obvious Courant convection limit unless velocities are high (approximately 1 m/sec). This is because the transport terms and part of the convective source are treated implicitly in the numerical scheme above. Instead, it is the explicit coupling (no instantaneous exchange of information) of the chemistry source term, s^{chm}_i, that will result in an oscillation of the species concentrations, c_i, if the time step becomes too large.

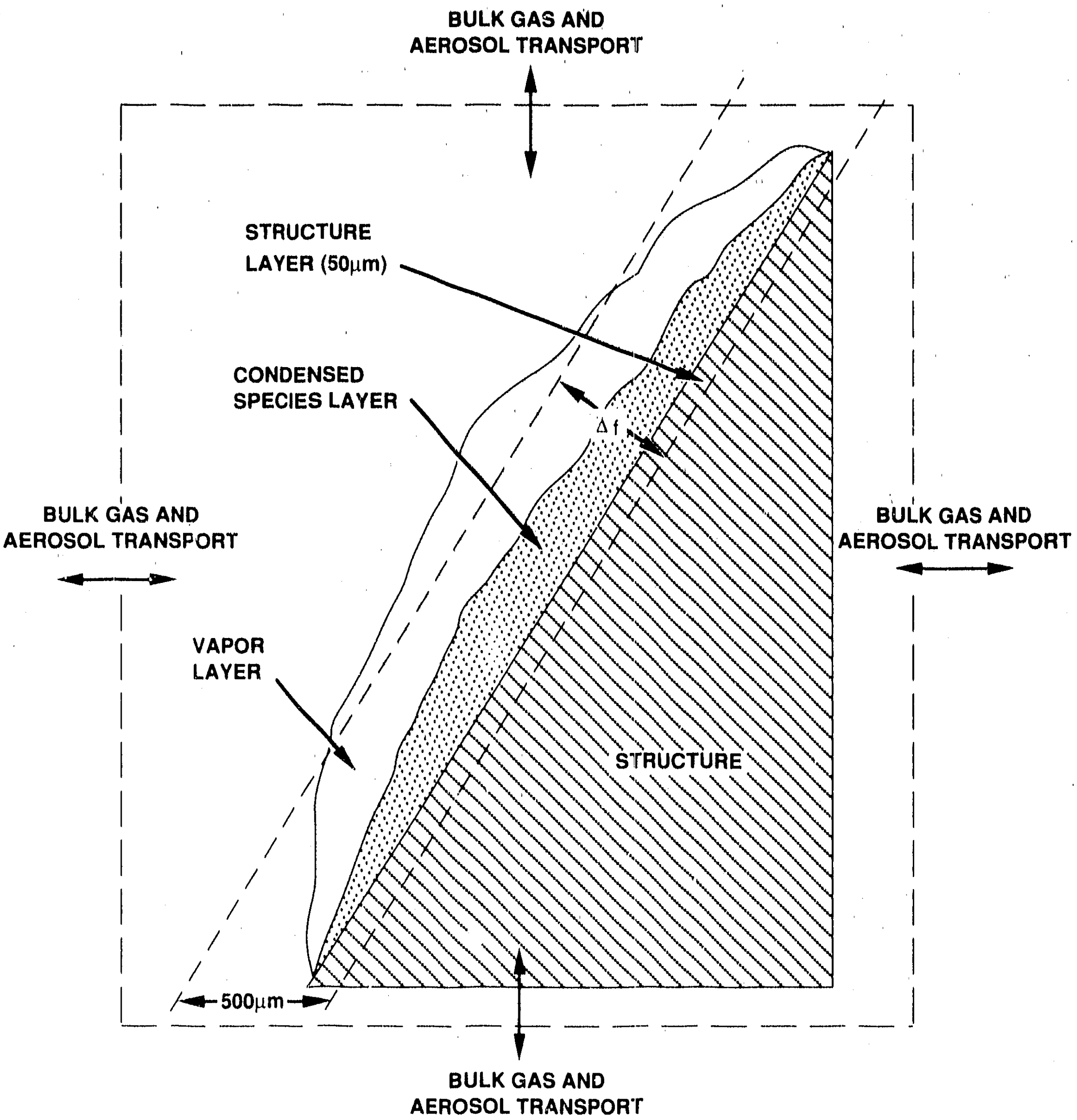


Figure 4.1 Surface film parameters

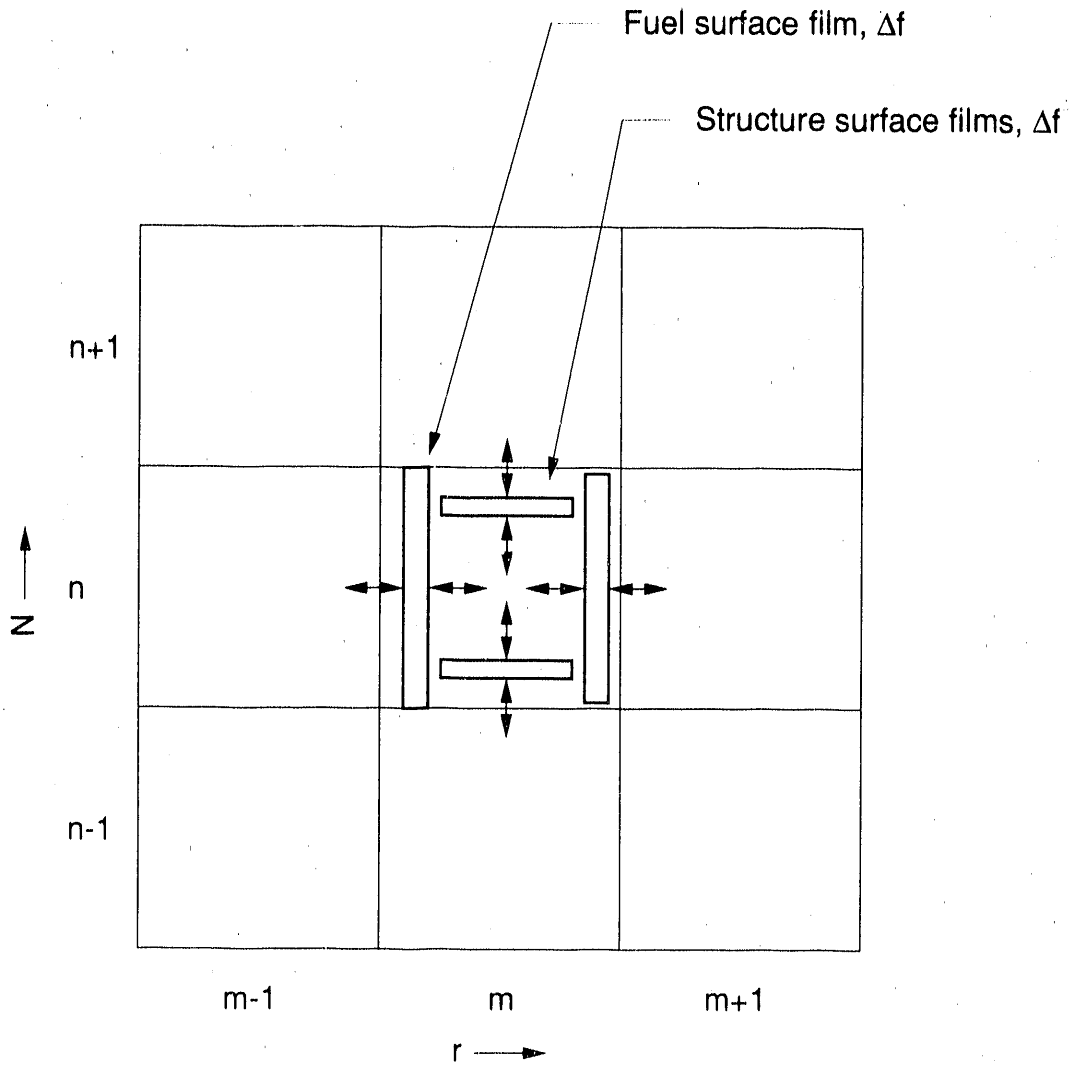


Figure 4.2. Species molar flow at bulk gas cell m,n.

5. CHEMISTRY

5.1 Background

The chemical interactions of fission products can have an important effect on the timing, chemical form, and amount of products which may be released from the vessel or RCS should either fail. Two areas of particular concern are: the chemical form of iodine (which affects its volatility), and the importance of resuspension by reentrainment of deposited aerosols and by revaporization, particularly at the time of vessel or RCS failure [5.1]. An assessment of these concerns requires detailed consideration of chemical interactions within the fuel, and in gas regions, and with structures.

Microscopic studies [3.2] of LWR fuel have shown that fission product elements diffuse within the fuel grains to form associated species, either as part of the UO_2 matrix or as separate phases at grain interstices. This affects fission product element concentrations within the grains and, hence, the driving force for diffusion to the grain edges and into the open porosity and their vapor pressures there.

Some fission products may condense on the cladding to react with its constituents; for example, Te reacts favorably with Zr and Sn [5.2]. Sn can vaporize to react with fission products in the vapor phase [5.3]. Both processes affect fission product diffusion between the fuel and the cladding.

Cladding oxidation by steam generates large amounts of heat. It affects the chemical behavior of fission products, both through the altered temperature in the vicinity of the cladding, and by reducing the oxygen chemical potential in the gas.

Fission products mix and react with steam, molecular hydrogen, and vaporized structural constituents in the gas region. This affects the partition between condensed (aerosol) and gaseous species, and the physical characteristics of the aerosol which, in turn, affects transport mechanisms between the gas and structure surfaces.

Structural components are generally made of Zircaloy, Inconel (an alloy consisting of 76% Ni, 16% Cr, and 8% Fe [5.4]), or stainless steel. Special components are also present such as control rods in PWR's containing a Ag/In/Cd alloy and control blades in BWR's containing B_4C powder. Fission products and other gas-borne materials can condense or deposit onto structures and react amongst themselves and with structural constituents; trace constituents such as Si and Cr in steel may be important [5.5]. Such interactions affect removal rates of the fission products from the gas and their resuspension.

Treating these processes (and others) comprehensively and realistically is a challenge. Some simplifying assumptions, however, must be made to avoid prohibitive computational labor. The assumptions made in VICTORIA in order to treat chemical processes effectively are described in this chapter.

Water ($H_2O[c]$) is assumed to not be present in the vessel after fission products are released from the fuel because VICTORIA is intended only to model unrecovered severe accidents such as those for which no emergency cooling water injection occurs during or after fuel failure. This scenario allows fission product interactions with water, which could otherwise be present (for example, as a pool in the lower head) to be ignored. Steam, on the other hand, may be present and therefore chemical interactions involving steam are treated. Temperatures within the vessel are assumed to be too high for steam condensation to occur.

All chemical interactions are assumed to be transport-limited. Species diffusion is therefore treated as required, and chemical equilibrium pertains in all regions. This has significant advantages over the alternative of treating chemical rate limiting processes. Because computation costs are less, many species and elements can be treated. Although as was discussed in Chapter 3, neither all possible species are used nor are all elements included.

All chemical phases are assumed to be ideal. This assumption means that within a cell the mass of any species is primarily determined from the availability of its constituent elements and by the mass fraction of the species. Hence the total mass of the other species is important but not which particular species constitute that mass. Data are not readily available to treat non-ideality comprehensively and chemical equilibrium would be more expensive to calculate for non-ideal systems. The breakdown of this assumption would be more likely to occur in the condensate and in low temperature gases [5.6].

At most, one condensed phase of each chemical substance is assumed to exist throughout the prevailing conditions of any modeled accident.

Transmutation and isotope effects are assumed to be unimportant. Consequently, element masses are conserved, relative isotopic abundances are everywhere uniform (but not necessarily time independent), and radioactive decay can be independently calculated.

Some of these assumptions are evaluated in more detail later in this chapter; their validity is discussed at the time of their application. Using these assumptions, a unified approach to calculating chemical equilibria will be derived and its application to the gas and fuel regions and structure surfaces discussed. Careful attention is paid to how the approach might be extended to achieve greater reality in the treatment.

Next, the solution method of the equations which determine chemical equilibrium is presented. With the simplifications outlined above, these calculations can still dominate the computational costs of VICTORIA, even when coupled with thermal hydraulics codes like MELPROG and TRAC. Therefore, considerable effort has been made to achieve maximum calculational efficiency. Standard methods suffer numerical difficulties or have been found to be inefficient when applied to this chemical equilibrium problem. For example, the Villars-Cruise-Smith (VCS) algorithm [5.6] is commonly used to solve chemical equilibria problems and was tried early on. The number of nonlinear equations solved equals the number of independent reactions, n_r. The method is fast when n_r is small, as is the case in systems with small numbers of species. However, in this application, n_r is generally large (typically in excess of 100) and the VCS method was found to be prohibitively expensive. For ideal systems, however, the number of equations can be reduced to the number of elements--which is considerably smaller than n_r in the VICTORIA application (25 or so). The equations in this form can be solved more economically. When n_r is large, the VCS method is better suited to non-ideal systems for which the reduction in the number of unknowns from n_r to n_e is not possible.

New methods of solving the equations have had to be developed. These methods have proved to be efficient and are discussed here in detail. The chapter concludes with a description of the zirconium oxidation model.

5.2 Equilibrium Chemistry

5.2.1 Derivation of the Equations

5.2.1.1 Conditions for Chemical Equilibrium

The condition for chemical equilibrium in a system at temperature, T, and pressure, P, is

$$\sum_i \mu_i dn_i = 0 \text{ and } n_i \geq 0 \ , \qquad (5.1)$$

where μ_i is the chemical potential of the i^{th} species and n_i is the corresponding number of moles. The sum is over all species, counting separately species with the same chemical formula in different phases. In this and subsequent formulae, the limits of summation are not shown when the sum is over all values of the index for which the variables in the sum are defined.

Often, Equation (5.1) is derived for a system held at constant T and P. This is done by minimizing the Gibbs free energy of the system, G, in which case the Equation (5.1) is equivalent to dG = 0. In fact, it is generally valid whatever the constraints on the system. This is important because systems with a ariety of externally applied constraints will be considered here.

The n_i are not independent when phase changes or chemical reactions occur; they are constrained by conservation of mass of the elements. Let the number of moles of each element be denoted by N_j for the j^{th} element, counting contributions from all species. Let a_{ij} denote the number of moles of the j^{th} element in the i^{th} species then for each element

$$N_j = \sum_i a_{ij} n_i \quad . \tag{5.2}$$

In differential form, this is

$$dN_j = \sum_i a_{ij} dn_i \quad . \tag{5.3}$$

The method of Lagrangian multipliers can be used to find the solution to Equation (5.1) subject to these constraints. The result is

$$\mu_i = \sum_j a_{ij} \lambda_j \quad , \tag{5.4}$$

where λ_j is the Lagrangian multiplier corresponding to the j^{th} element constraint. Equation (5.2) is assumed to yield the only constraint. Additional constraints arise when true chemical equilibrium has not been established, as might be the case at low temperatures or concentrations. This possibility is excluded here.

There are n_s of these relations and n_e parameters, where n_s and n_e are the number of species and elements respectively. The λ_j can be eliminated to find $n_r = n_s - n_e$ relationships between the μ_i. There are a number of ways this can be done, but the simplest is when the elements are included in the sum over species (that is, the unassociated elements are present in the system). Let the first n_e species be chosen to be these elements and denote this set by e. Then $\lambda_j = \mu_j$ $j \in e$ and

$$\mu_i = \sum_j a_{ij} \mu_j \quad . \tag{5.5}$$

Therefore, only the last n_r of these equations contain the useful chemical reaction information.

Throughout the remainder of this chapter the following definitions will be used:

$j \epsilon e$ indicates j is a member of the elemental species set e

$h \epsilon d$ indicates h is an element of the elimination set d

$i \epsilon g$ indicates i is a member of the gaseous species set g

$i \epsilon c$ indicates i is a member of the condensed species set c

More generally, chemical potentials for n_e species can be chosen to eliminate the λ_j. Let this set of indices be denoted by d. If the determinant of the n_e rows of a_{hj} for $h \epsilon d$ is non-zero, then A_{jh} for $j \epsilon e$ and $h \epsilon d$, the inverse matrix, exists and can be used to invert Equation (5.4), giving

$$\lambda_j = \sum_h A_{jh}\mu_h \quad . \tag{5.6}$$

Now, eliminating the λ_j from Equation (5.4), gives

$$\mu_i = \sum_{j,h} a_{ij}A_{jh}\mu_h \quad . \tag{5.7}$$

Again, only the last n_r of these equations are useful.

The requirement that A_{jh} exists means that one cannot choose any set of n_e chemical potentials to eliminate the λ_j. Indeed, one must choose a set whose corresponding species are linearly independent in the sense that all other species can be made from linear combinations of them. This can always be done when rank $(a_{ij}) = n_e$, Ferrar [5.7].

Should $\tilde{n}_e \equiv$ rank $(a_{ij}) < n_e$, then the constraints in Equation (5.3) are not linearly independent. Equation (5.4) remains valid, but only $\tilde{n}_e$ independent species can be found. n_r is now $n_s - \tilde{n}_e$. (A simple example of such a system is: HF(g), $(HF)_2$(g) for which $n_s = 2$, $\tilde{n}_e = 1$, and $n_r = 1$, corresponding to the reaction $2HF \rightleftharpoons (HF)_2$.) The outcome is that the n_e λ_j can be expressed as linear combinations of the $\tilde{n}_e \mu_j$. So, Equation (5.6) also remains valid, except A_{jk} becomes a square matrix of dimension $\tilde{n}_e$. This simplifies the calculation because there are now fewer unknowns in the final equations. Details are not given because $\tilde{n}_e = n_e$ in all our applications.

The usual form in which the conditions for chemical equilibrium are expressed [5.6] is

$$\sum_i \nu_{ki}\mu_i = 0 \quad . \tag{5.8}$$

The ν_{ki} k = 1,...n_r are the stoichiometric coefficients in the k^{th} chemical reaction, expressed as

$$\sum_i \nu_{ki}S_i = 0 \quad , \tag{5.9}$$

where S_i denotes the chemical formula of the i^{th} species. The equivalent to Equation (5.8) in this application is

$$\sum_{m,i} T_{km}\, \delta(m,i-n_e)\, \{\mu_i - \sum_{j,h} a_{ij}A_{jh}\mu_h\} = 0 \quad , \tag{5.10}$$

where T_{km} k,m = 1...n_r is of rank n_r and $\delta(m,i-n_e)$ is the Kronecker delta, which is zero unless m = i - n_e when it is unity. We see that Equation (5.7) corresponds to a special choice of the n_r chemical reactions, that is, T_{km} is the unit matrix. This special choice has the advantage of leading to a reduction in the number of unknowns. This would be more difficult with a formulation not including the elements in the species set.

5.2.2 Equations for two phases at given T and V.

There are n_e unknowns, μ_j for j ϵ e, and n_e constraints, in the conservation equations for N_j. So, in principle, the equations are closed once the functional dependence of the μ_i on the n_i are known. Further progress cannot be made without making the dependence explicit.

Henceforward, assume that the separate phases in the system are ideal. There are two practical reasons for this: First, data do not exist to treat in general the non-ideality in the systems considered here (though models exist which may be suitable). Second, the equations for non-ideal systems are much more difficult and (more to the point) expensive to solve.

Consider first a system with just two phases, one condensed (either solid or liquid) and the other gaseous. Suppose further that the temperature and total volume, V, of the system are specified prior to each equilibrium calculation. This is the present assumption in VICTORIA in all regions for which equilibrium chemistry is assumed. This will be evaluated later.

The chemical potentials of the gaseous species are

$$\mu_i = \mu_i^o(P_o, T) + RT \ln(P_i/P_o), \quad i \in g , \qquad (5.11)$$

where μ_i^o is the standard chemical potential of the pure i^{th} gaseous species. In writing this equation we follow the Raoult convention [5.6] for an ideal solution, that is, one in which the standard state is the pure species state at P_o, and evaluate μ_i^o as the Gibbs free energy of formation, ΔG_i^o, at P_o and T. P_o is typically atmospheric pressure, therefore μ_i^o is a function of T only. R is the universal gas constant, P_i is the species partial pressure, and g denotes the set of indices which correspond to gaseous species.

It is convenient to rewrite this equation. Define the (pseudo) concentration of each gaseous species by $c_i = n_i/V$ for all $i \in g$. Set the $P_i = c_iRTV/V_{gas}$, where V_{gas} is the volume of the gaseous phase, then for gaseous species Equation (5.11) becomes

$$\mu_i = \mu_i^o(P_o, T) + RT \ln(c_iRTV/V_{gas}P_o), \quad i \in g . \qquad (5.12)$$

Similarly the chemical potentials of the condensed species are

$$\mu_i = \mu_i^o(P_o, T) + RT \ln(x_i), \quad i \in c , \qquad (5.13)$$

where μ_i^o is the standard Gibbs free energy of the pure i^{th} condensed species (specify liquid or solid), x_i is the molar fraction of the species in the condensed phase and c denotes the set of indices which correspond to condensed species. If c_i is defined by $c_i = x_in_c/V$ for all $i \in c$, where n_c is $\Sigma_{i\epsilon c}n_i$, then

$$\mu_i = \mu_i^o(P_o, T) + RT \ln(c_iV/n_c), \quad i \in c \quad . \qquad (5.14)$$

Substitute Equations (5.12) and (5.14) into Equation (5.5), where the independent chemical potentials in Equation (5.5) are chosen to be those for the gaseous elements, to find

$$c_i = R_i \prod_j c_j^{a_{ij}} \{V/V_{gas}\}^{-1} \prod_k \{V/V_{gas}\}^{a_{ik}} \quad , \quad i \in g \qquad (5.15)$$

and

$$c_i = R_i \prod_j c_j^{a_{ij}} \, n_c/n_c^o \prod_k \{V/V_{gas}\}^{a_{ik}} \quad , \quad i \in c \quad , \qquad (5.16)$$

R_i is the equilibrium constant for dissociation to form the gaseous elements, defined by

$$R_i = \exp(-\Delta G_i/RT) \; \{RT/P_o\}^{-1} \prod_j \{RT/P_o\}^{a_{ij}}, \quad i \in g \quad , \qquad (5.17)$$

and

$$R_i = \exp(-\Delta G_i/RT) \; \{V/n_c^o\}^{-1} \prod_j \{RT/P_o\}^{a_{ij}}, \quad i \in c \quad , \qquad (5.18)$$

n_c^o is a reference value of n_c, whose purpose will become apparent shortly. ΔG_i is the relative standard species Gibbs free energy (relative to the gaseous elements), defined by

$$\Delta G_i = \mu_i^o - \sum_j a_{ij}\mu_j^o \quad . \qquad (5.19)$$

Notice that the equilibrium constants have been defined so that they are independent of n_i; they need only be calculated once each time a chemical equilibrium calculation is done since T and V are fixed at the start of calculation of a time interval.

It is convenient here to discuss how the ΔG_i are calculated. Apparently, the sum in Equation (5.19) has to be calculated at the start of every chemical equilibrium calculation since μ_j^o for $j \in e$ depends on T. This is computationally expensive when a large number of species and elements are treated. However, the standard Gibbs free energies are calculated as a power series in T as follows

$$\mu_i^o = \sum_k \gamma_{ik} T^k \quad , \qquad (5.20)$$

where γ_{ik} for $k = 0,1 \ldots f_i$ are constants. Substitute this into Equation (5.19) and reverse the order of the summations to find

$$\Delta G_i = \sum_k \Delta\gamma_{ik} T^k \quad , \qquad (5.21)$$

where

$$\Delta\gamma_{ik} = \gamma_{ik} - \Sigma\, a_{ij}\gamma_{jk} \quad . \tag{5.22}$$

$\Delta\gamma_{ik}$ need be calculated once since only these are independent of T. Therefore, the calculation of ΔG_i requires only the calculation of a power series at each new value of T. Typically, f_i is small because the power series expansions for μ_i^o converge rapidly (f_i is 4 in VICTORIA), and therefore ΔG_i can be calculated economically from Equation (5.21).

Now a closed set of equations can be formed.

Start with the mass balance equation (Equation [5.2]) and divide this through by V to obtain

$$C_l = \sum_i a_{il} c_i \quad , \tag{5.23}$$

where $C_l = N_l/V$ for all $l \in e$. Substituting Equations (5.15) and (5.16) into this to yield

$$\begin{aligned} C_l = & \sum_{i \epsilon g} a_{il} R_i \prod_j c_j^{a_{ij}} \{V/V_{gas}\}^{-1} \prod_k \{V/V_{gas}\}^{a_{ik}} \\ & + \sum_{i \epsilon c} a_{il} R_i \prod_j c_j^{a_{ij}} n_c/n_c^o \prod_k \{V/V_{gas}\}^{a_{ik}} \quad . \end{aligned} \tag{5.24}$$

V_{gas} is V - V_c where V_c is the volume of the condensed phase which is a function of n_i for all $i \in c$. So, the unknowns in these equations are c_j for $j \in e$, V_{gas}, and n_c. n_c is determined by the requirement that the sum of the mole fractions in the condensed phase is unity

$$n_c = V \sum_{i \epsilon c} c_i \quad . \tag{5.25}$$

The solution must satisfy $c_i \geq 0$ for all i, and either one or two solutions exist. One of the solutions always consists of a gaseous phase and no condensed phase. If the second solution also exists, then it will consist of two phases; the gas phase in the first solution must, then, have been supersaturated.

5.2.3 Two Approximations

So far, no approximations have been made in the formulation of these equations. In order to reduce the computational cost of obtaining the solution of these equations, two approximations will be made.

First, n_c^o is chosen to be n_c at the end of the previous time step, and n_c in Equation (5.24) is approximated as n_c^o. Equation (5.25) is used to update n_c after the other equations have been solved. We require $|n_c - n_c^o|/n_c << 1$, which can be achieved if the time step is made sufficiently small. Notice that this approximation implies that the condensed phase can never vanish if it existed initially, though n_c may become small after many time steps. The approximation breaks down when there is no initially condensed phase, $n_c^o = 0$.

Second, V_{gas} is approximated as V, and $|V - V_{gas}|/V << 1$ also.

It is important to note that neither of these approximations leads to a mass balance error (they amount to making approximations in Equations (5.15) and (5.16) only). They affect the equilibria between the species and, hence, alter the relative proportions of species.

With these two approximations, Equation (5.24) becomes

$$C_l = \sum_i a_{il} R_i \prod_j c_j^{a_{ij}} \quad . \qquad (5.26)$$

It is closed and the method of solution is discussed in Subsection 5.4.

Since V, T, and N_l are specified, P is determined by the solution and is given by the summation of the gaseous species partial pressures

$$\begin{aligned} P &= \sum P_i \quad , \qquad i \in g \\ P_i &= c_i RTV/V_{gas} = c_i RT \quad . \end{aligned} \qquad (5.27)$$

The implications of P for the application of these equations in VICTORIA are discussed in the following section.

The cell volume, V, may change from one time step to the next as the core geometry changes. This affects the calculation of C_l. Let V^o and C_l^o denote their values at the end of the preceding time step. C_l^o generally will have been modified from its value found from the equilibrium calculations at that time step by transport to C_l according to

$$C_1^o V^o = C_1 V \quad . \tag{5.28}$$

5.3 Applications

In this section the arguments are made concerning the applicability of the Equation set (5.26) to the various regions in the reactor vessel where VICTORIA performs a chemistry calculation.

5.3.1 Application to the Bulk Gas Region

The bulk gas region is the space between structures and fuel containing gas and aerosol; species condensed onto or immediately adjacent to fuel or structure surfaces are excluded. The bulk gas region is divided up into a number of fluid cells whose interfaces are determined by the axial levels and radial rings input, as discussed in Chapter 4. Each is assumed to be well mixed; T, P and gaseous species concentrations are, therefore, taken to be spatially constant inside. Further, a number of assumptions are made with regard to the chemistry calculations done in these cells. These will be discussed in turn.

The cells are assumed to be materially isolated during the chemistry step. This appears unrealistic at first sight because the cells are, in fact, interconnected. This assumption is justified by the overall numerical approach in VICTORIA. In this approach, the governing equations are split into convenient parts and each part is solved in turn, uncoupled from all others during the execution of its solution. Coupling between the parts occurs after having solved one equation before starting the solution of the next. The method is called "splitting" or "the fractional step method" [5.8]. Here the parts are defined so that transport of material into or out of the cells is not done during the chemistry step.

It is easy to show that the fractional step method is accurate to order Δt, where Δt is the time step, and errors can be made small by choosing Δt small enough to provide the desired accuracy. In this derivation, assume that the user has chosen an accurate Δt. Its value will depend on the problem at hand, and calculations should be made in which Δt is varied to test the error requirement. Experience with the code to date indicates that the material Courant condition discussed in Sections 2.4 and 3.5 are more stringent than this accuracy condition.

One other external constraint is required: At first sight, a reasonable choice would be that the cells are thermally isolated, that is, $\Delta Q = 0$, where ΔQ is the amount of heat which enters the cell. Since V is constant, this is equivalent to $\Delta U = 0$, where U is the internal energy of the cell contents. U is easily found from the species chemical potentials, with the result

$$U = \sum_i n_i \left\{ \mu_i^o - T \frac{d\mu_i^o}{dT} \right\} - PV \quad . \tag{5.29}$$

This applies to a mixture with uniform temperature which need not be in chemical equilibrium. With this constraint, the cell temperature would not be constant during the chemistry step, and the equilibrium constants (Equations 5.16 and 5.17) would have to be calculated many times to obtain the solution for each cell. This is computationally costly and is best avoided.

Therefore, VICTORIA does the calculation at constant T. Heat, therefore, flows into or out of the cell; the amount is be calculated by using Equation (5.29). This second choice for the external constraint does not necessarily amount to an approximation: it affects the size of the time step, Δt, required for convergence and the coupling of the parts in the fractional step method; the internal energy change, ΔU should be accounted for in the thermal hydraulics and heat transfer calculations. In fact, ΔU is ignored in calculations done with stand-alone VICTORIA because the thermal hydraulic calculations, etc. are done in advance. ΔU is assumed to be negligible compared to other sources of heat, but this assumption is not verified here because its validity depends on the case being studied; however, reducing the time step would be an easy way for the user to verify it for their case.

Subsection 5.2.1 noted that P is determined by the solution to the equations and, therefore, generally has a value different from that immediately before the chemistry step. This change in a principle parameter should be fed back to other parts of the calculation. Among those parts affected are the thermal hydraulics calculations which, as just noted, are done in advance. The effect of the change in P is assumed to be negligible. Again, the validity of this depends on the case being studied and the user should examine the variation of P in time.

Lastly, consider the underlying assumption that the cell contents are in chemical equilibrium. This assumption is convenient because only one equilibrium calculation is required per cell. Two aspects, however, require further consideration.

First, since the gas is in equilibrium with the condensate, this prevents treating aerosol evaporation and condensation dynamically. Therefore, it is difficult to evaluate whether dynamic effects are important, and a degree of arbitrariness enters over how the aerosol mass distribution should be updated due to the formation or removal of condensed species during the chemistry step.

These problems can be overcome by relaxing the chemical equilibrium assumption. For example, we might assume that the gaseous species are in mutual chemical equilibrium (Equation (5.24) with $n_c = 0$ is valid here and no approximation is needed) and the condensed species are in mutual equilibrium, with particle compositions and equilibria depending on their size. This requires that an equilibrium calculation be done for each condensate particle size group.

Second, mutual chemical equilibrium of the condensate (whether or not it is in equilibrium with the gas) is a diffusion-limited process which, even for submicron particles, has a time constant large compared to typical time steps in VICTORIA. It may be, therefore, more realistic to assume that the condensed species do not interact with one another; that is, they are effectively immiscible. This is not amenable to a simple treatment if equilibrium between the gas and the condensate is maintained since the Gibbs phase rule limits the number of condensed species to no more than n_e at given T. (The rule for a system with chemical reactions is $f \leq \tilde{n}_e + 2 - \alpha$, where f is the number of intensive degrees of freedom and α is the number of phases in equilibrium. $\tilde{n}_e$ is defined in Subsection 5.2.1. Here f is unity (since T is specified) and $\tilde{n}_e$ equals n_e, therefore $\alpha \leq n_e$.) However, it is amenable to a dynamic treatment of condensation and evaporation for which only an equilibrium calculation for the gas is required.

5.3.2 Application to the Structure Surface Region

Species can be transported to or from structure surfaces by turbulent and molecular diffusion in the gaseous phase and by aerosol deposition in the condensed phase.

Diffusion rates depend on vapor pressure differences between the bulk flow and the structure surfaces, and condensed species at the surfaces can evaporate and be reentrained into the bulk flow. A treatment of surface chemical effects is therefore important to the calculation of transport rates to and from surfaces [see Equation (4.1)].

A thin constant-thickness region, Δ_f, which extends over each structure surface is defined as shown on Figure 4.1. Each region contains all gaseous and condensed material transported to the surface and is the source of vapor reentrained into the bulk flow. It can also contain some of the structure material (a 50 μm layer is set in the code) to enable chemical interactions with the surface to be treated. Chemical equilibrium is assumed in this region.

A number of assumptions are made which require evaluation for this region. These are discussed in turn.

The temperature of the surface region is assumed to equal the structure surface temperature. Heat flows into or out of the region during the chemistry step (see the related discussion in the previous subsection). This heat flow is assumed to be negligible compared to decay heating at the surface and heat transferred between the surface and other regions such as the bulk gas and the structure interior.

The volume of the structure film region, V_s, during the chemistry step is constant. It must be sufficiently small compared to the volume of the bulk gas region so that it is justifiable to assume that the gas within the surface region in equilibrium with the condensed solid there. It is also assumed that transport between the gas and solid is the rate limiting process, therefore it is required that gas concentration differences across the thickness of the gaseous part of the surface region, Δ_f, be small compared to gas concentration differences between the surface and the bulk gas regions. The former can be found from standard wall profiles for smooth surfaces, and the latter can be found from mass transfer correlations for the bulk flow. Generally, for fully turbulent bulk flow, Δ_f must be less than, or of the order of, the thickness of the laminar sub-layer.

V_s must also be large enough that $V_s \simeq V_{gas}$ is valid; that is, we require $V_c << V_{gas}$ implies that the thickness of the condensed layer must be much smaller than the thickness of the laminar sub-layer.

Evaluation of these requirements will depend on the application in hand, but notice that they will become conflicting with increasing amounts of condensed surface deposits.

5.3.3 Application to the Fuel Cladding/Gap Region

The cladding/gap interface is part of the fuel region, which is discussed in detail in Chapter 3. In particular Subsection 3.4.1 details the reasons for the assumed 2-phase distribution, that is condensed species on the cladding inner surface and gaseous species in the fuel clad gap itself. The discussion here centers on the application of the equations derived in Subsection 5.2.1.

The equilibrium is treated in a similar manner to that for the structure surfaces. In this case the entire gap is the assumed constant-thickness region. Therefore the discussion of Subsection 5.3.2 applies with the exception that the gap gas is quiescent

5.3.4 Application to the Fuel Open Porosity

The fuel consists of the fuel pellets only. They may be sub-divided into a number of radial rings. The gaseous and condensed species within each ring are each in chemical equilibrium. In this case the condensate is on the grain surfaces and the gases in the open porosity.

V_{gas}, the open porosity volume, is assumed to equal the cell volume, V, for the chemistry calculation and Equation (5.26) therefore applies. Wheatley [5.9] has examined the alternatives to this treatment, each of which require more computational effort with only small improvements.

5.4 The Solution Method

In this section a description is given of the properties of the equations when there is either a dominant or a zero concentration of the species. Thus the resulting Jacobian can be so ill-defined that Newton-like methods do not converge well. But by taking advantage of the dominant species in the fuel, UO_2, and bulk flow, steam, a change of variable technique can be employed that results in an exceptionally fast and accurate solution.

5.4.1 Properties of the Equation

Equation (5.26) is set of n_e non-linear algebraic equations for the n_e unknowns c_j for $j \in e$. Although approximations have been made in their derivation for the two-phase system, they are exact for single phase gaseous systems at constant V. If it was necessary the solution could be reformulated for gaseous systems at constant P, a set of $n_e + 1$ equations would then be obtained whose structure is the same as Equation (5.26). The succeeding discussion is thus applicable with minor modification to this alternative formulation.

We begin by noting some properties of the equations. First, consider the variational problem

$$\delta\Gamma = 0 \text{ and } c_j \geq 0 \text{ for } j \in e \quad , \qquad (5.30)$$

where Γ is a function of the c_j for $j \in e$ given by

$$\Gamma = \sum_i c_i - \sum_j C_j \ln(c_j) \qquad , \qquad (5.31)$$

here the c_i are given by

$$c_i = R_i \prod_j c_j^{a_{ij}} \quad . \qquad (5.32)$$

and $C_j \geq 0$ and $R_i > 0$ and are as defined by Equations (5.23) and (5.18) respectively.

If we define $\tilde{g}_j$ as $\partial\Gamma/\partial c_j$ for $j \in e$, then the solution of Equation (5.30) is

$$\left.\begin{aligned} \tilde{g}_j &= 0 \text{ when } C_j > 0 \\ c_j &= 0 \text{ when } C_j = 0 \quad . \end{aligned}\right\} \qquad (5.33)$$

Without the loss of generality, we can assume that all the C_j are greater than zero. If this should not be so, then the problem can be redefined in terms of only those elements and species present in non-zero amounts. Hence, Equation (5.33) reduces to $\tilde{g}_j = 0$. We assume a solution exists with $c_j > 0$ for all $j \in e$.

Equation (5.26) is equivalent to

$$g_j = 0 \quad , \qquad (5.34)$$

where g_j is given by

$$g_j = \sum_i a_{ij} c_i - C_j \quad . \qquad (5.35)$$

It is easily shown that Equations (5.33) and (5.34) are equivalent with $g_j = c_j \tilde{g}_j$.

Notice that the g_j are the differences between the element abundances calculated from the c_j for $j \in e$ and the prescribed element abundances.

Equation (5.34) can be used to obtain upper bounds on the c_i,

$$c_i \le \lim_{\epsilon \to 0} \left[\min \left\{ \frac{C_1}{a_{i1} + \epsilon} , \frac{C_2}{a_{i2} + \epsilon} , \ldots , \frac{C_{n_e}}{a_{in_e} + \epsilon} \right\} \right] . \qquad (5.36)$$

We define two Jacobians

$$\tilde{J}_{jk} = \frac{\partial \tilde{g}_k}{\partial c_j} = \frac{\partial^2 \Gamma}{\partial c_j \partial c_k} \quad , \qquad (5.37)$$

and

$$J_{jk} = c_j \frac{\partial g_k}{\partial c_j} \qquad (5.38)$$

for $j \in e$. ($\tilde{J}_{jk}$ could also be referred to as a Hessian in view of the existence of Γ.)

J_{jk} is given by

$$J_{jk} = \sum_i a_{ij} a_{ik} c_i \quad , \qquad (5.39)$$

and $\tilde{J}_{jk}$ is related to J_{jk} by

$$c_j c_k \tilde{J}_{jk} = J_{jk} - \delta(j, k)\, g_k \ . \qquad (5.40)$$

J_{jk} is positive definite if

$$\sum_{j,k} e_j J_{jk} e_k > 0 \qquad (5.41)$$

for $|e_j| \neq 0$ and e_j otherwise arbitrary. J_{jk} is positive definite if

$$\sum_i \left(\sum_j a_{ij} e_j\right)^2 c_i > 0 \ . \qquad (5.42)$$

All the c_i are greater than zerc because we restrict the c_j for $j \in e$ to be greater than zero (see the discussion concerning Equation [5.33]). The condition in Equation (5.42) will, therefore, be satisfied providing the sum in brackets is non-zero for at least one value of j. A standard result of linear algebra is that this condition will be satisfied if rank $(a_{ij}) = n_e$, which, as noted in Subsection 5.2.1, is the case.

J_{jk} is therefore positive definite for all positive values of c_j $j \in e$. J_{jk}, on the other hand, is only necessarily positive definite when $g_j \leq 0$ or when $g_j << J_{jj}$ for all $j \in e$. This will be satisfied in the vicinity of the solution and when the c_j for $j \in e$ are made sufficiently small.

5.4.2 Ill-conditioning of the Jacobian and Implications

Assume for the moment that some of the c_i are zero. This amounts to eliminating the rows in a_{ij} corresponding to the absent species. The condition for positive definiteness of J_{jk} now requires rank $(a_{ij}) = n_e$, where $\tilde{a}_{ij}$ is a_{ij} without these rows; that is, there must be at least n_e linearly independent species (in the sense defined in Subsection 5.2.1) amongst those remaining. It follows that if there are less than n_e such species then J_{jk} is singular.

Of course, all the c_i are non-zero, but chemical systems are often dealt with in which one or a few species are present in large amounts. Unless at least n_e of these species are linearly independent and present in roughly equal amounts, then our observation above admits the possibility than J_{jk} can be ill-conditioned.[1] This can be true of J_{jk} also; e.g., in the vicinity of the solution. In our case, $H_2O(g)$ can dominate in the gas region and $UO_2(c)$ dominates in the fuel, and this possibility is relevant to our application. Further, if the c_j for $j \in e$ are far from the solution, then J_{jk} can be ill-conditioned there, even though it may be well-conditioned near to the solution. In view of the importance of this observation, the following illustration is provided.

Consider the system composed of H(g), O(g), $H_2(g)$ and $H_2O(g)$; label these as species 1 to 4, respectively. (Note that it is not essential to include the gaseous elements in this list, but we include them to maintain consistency with our

[1] An ill-conditioned matrix is one whose rows or columns are nearly linearly dependent. A convenient measure of this dependency for our purpose is $\phi = \det(j_{jk})/\{J_{jk}\}$, where $\{J_{jk}\}$ denotes $\Sigma_p \, |\Pi_j \, J_{jp(j)}|$, p(j) is a permutation of the indices J, and the sum is over all permutations. J_{jk} is ill-conditioned when $\phi << 1$.

previous discussion.) The elements are H and O, labeled 1 and 2, respectively.

J_{jk} is

$$\begin{array}{cccc} H & O & H_2 & H_2O \\ \end{array} \qquad\qquad \begin{array}{cc} H & O \end{array}$$

$$\begin{bmatrix} 1 & 0 & 2 & 2 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & c_4 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 2 & 0 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} c_1 + 4c_3 + 4c_4 & 2c_4 \\ 2c_4 & c_2 + c_4 \end{bmatrix}$$

If c_1 and c_2 are both small compared to c_3 and c_4, then $\det(J_{jk})/\{J_{jk}\}^2$ is $c_3c_4/(c_3c_4 + 2c_4^2)$. This is much smaller than unity when c_4 is much greater than c_3 and is otherwise of order unity. In fact, in this simple case, it is clear from inspection that J_{jk} has nearly linearly dependent rows and columns when c_4 is large. An important point to note is that the ill-conditioning of J_{jk} is not caused by species which comprise one element only ($H_2(g)$ in this case).

This has important ramifications for the selection of algorithms to solve Equations (5.33) or (5.34). With Equation (5.34), Newton-like methods can be used far from the solution with the guarantee that the resulting step is always in a direction of decreasing Γ since J_{jk} is everywhere positive definite. However, all of the Newton-like methods can suffer numerical difficulties when J_{jk} or J_{jk} is ill-conditioned. The difficulty is essentially due to the finite number of significant figures to which real numbers are stored on computers. This truncation causes a computer-generated solution to a set of linear equations to become ill-defined as the determinant of their coefficient matrix becomes small.

Gordon and McBride [5.10] have found the equivalent problem in a slightly different formulation of the equilibrium equations. They suggest adding small numbers to the C_j to alleviate the ill-conditioning. However, this works at the expense of significantly affecting the relative proportions of the dominant independent species (defined later), and is not a valid remedy in the present case.

This rules out all the usually more effective Newton-like methods. Of those that remain, the method of steepest descents generally has a poor rate of convergence, even near the solution, and our calculations have shown that cyclic methods, though generally not having rapid convergence rates, have exceedingly poor rates of convergence when J_{jk} is ill-conditioned.

5.4.3 Change of Variables

The key to resolving the above difficulties lies in exploiting the existence of dominant species [5.11]. We will reformulate the equations in terms of new independent variables which are taken to be the set of c_i's corresponding to any set of n_e independent species (denoted by ω). It will be shown that Equation (5.34) becomes almost linear for special choices of the new variables.

We define $\tilde{g}_h(\omega)$ as $\partial\Gamma/\partial c_h$ for $h \in \omega$ and Γ expressed as a function of the new variables. The solution to Equation (5.30) is

$$\tilde{g}_h(\omega) = 0 \quad . \tag{5.43}$$

Defining $g_h(\omega)$ to be $c_h\tilde{g}_h(\omega)$ then

$$g_h(\omega) = 0 \quad . \tag{5.44}$$

$g_h(\omega)$ is given by

$$g_h(\omega) = \sum_j A_{jh}g_j \quad , \tag{5.45}$$

where A_{jh} $h \in \omega$ is the matrix defined in Subsection 5.2.1. We require A_{jh} to exist (which is consistent with the definition of d).

We also define new Jacobians, $\tilde{J}_{hm}(\omega)$ and $J_{hm}(\omega)$ $h, m \in \omega$ in the new variables analogously to Equations (5.37) and (5.38). These are given by

$$J_{hm}(\omega) = \sum_{j,k} J_{jk}A_{jh}A_{km} \quad , \tag{5.46}$$

and

$$c_hc_m\tilde{J}_{hm}(\omega) = J_{hm}(\omega) - \delta(h, m)\ g_m(\omega) \quad . \tag{5.47}$$

$J_{hm}(\omega)$ is clearly positive definite whenever J_{jk} is positive definite, and is, therefore, positive definite everywhere. $J_{hm}(\omega)$ is positive definite near the solution.

It is valuable now to express $g_h(\omega)$ and $J_{hm}(\omega)$ in terms of the c_i's:

$$g_h(\omega) = c_h - \sum_j C_j A_{jh} + \sum_{i \epsilon \bar{\omega}} (\sum_j a_{ij} A_{jh}) \, c_i \quad , \tag{5.48}$$

and

$$J_{hm}(\omega) = \delta(h, m) \, c_m + \sum_{i \epsilon \bar{\omega}} (\sum_j a_{ij} A_{jh}) (\sum_k a_{ik} A_{km}) \, c_i \quad , \tag{5.49}$$

where $\bar{\omega}$ denotes the set of species indices not contained in ω.

It can be seen that if $c_m >> c_i$ for $m \epsilon \omega$ and $i \epsilon \bar{\omega}$ then $J_{hm}(\omega)$ is nearly diagonal and is numerically well-conditioned.

A suitable choice for ω is the set of n_e dominant independent species defined as follows. Start with ω consisting of the species with largest c_i for $i \epsilon s$, where s is the set of all species. Add to ω the species with largest c_i for $i \epsilon s$ which is independent of those currently in ω and repeat until ω contains n_e species.[2,3] Henceforward, h stands for this set of species unless stated otherwise. It is easily shown that this definition results in $c_m > c_i$ for $m \epsilon \omega$ and all $i \epsilon \bar{\omega}$ for which $\Sigma_k aw_{ik} A_{km}$ is non-zero. We illustrate this by returning to our example.

Consider transforming to a new set of independent variables; since c_1 and c_2 are both assumed small, a suitable choice for ω is $H_2(g)$ and $H_2O(g)$.

[2] Although adequate as a definition, in practice this procedure can be computationally inefficient because some elements may be present in small amounts, and many iterations may be required before the dominant independent species which contain those elements are found. A faster but less straightforward procedure can be devised by exploiting the requirement that at least one species in d must contain the j^{th} element for all $j \epsilon e$.

[3] A convenient method of testing whether a species is independent of those in h is to perform a Schmidt orthogonalization on the composition vectors of the species in ω and then to attempt to orthogonalize the composition vector of the species being tested with respect to the orthogonalized vectors of those in ω. The orthogonalization breaks down if the species is not independent.

$$[a_{kj}] = \begin{bmatrix} 2 & 0 \\ 2 & 1 \end{bmatrix}$$

$$[a_{hj}]^{-1} = [A_{jh}] = \begin{bmatrix} \frac{1}{2} & 0 \\ -1 & 1 \end{bmatrix}$$

Therefore, from Equation 5.49 $J_{hm}(\omega)$ is

$$\begin{array}{cccc} H & O & H_2 & H_2O \\ \end{array} \qquad\qquad\qquad \begin{array}{cc} H_2 & H_2O \end{array}$$

$$\begin{bmatrix} \frac{1}{2} & -1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & c_4 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & 0 \\ -1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{4} c_1 + c_2 + c_3 & -c_2 \\ -c_2 & c_2 + c_4 \end{bmatrix} .$$

Clearly, when c_1 and c_2 are both assumed small, $J_{hm}(\omega)$ is numerically well-conditioned, whether or not c_4 is large compared to c_3.

5.4.4 Solution Algorithm

Equations (5.48) are nearly linear and $J_{hm}(\omega)$ is nearly diagonal. This suggests that a cyclically ordered Newton method,[4] taking one equation at a time, will be effective in determining an initial estimate.

Applying the Newton method to Equation (5.44),

$$s_h = - g_h(\omega)/J_{hh}(\omega) \quad . \tag{5.50}$$

[4] The general method [5.12] is to define subsets of the nonlinear equations. Each subset has associated with it a subset of equal size of the independent variables. One step of an iterative solution method is applied to each subset of equations in turn. The cycle is repeated until convergence is achieved. Commonly, each subset consists of one equation only and the Newton method is used. Convergence is guaranteed for less stringent constraints than those when the Newton method is applied to all equations simultaneously.

This equation could also have been derived by applying the Newton method to Equation (5.43), but doing the Taylor expansion about the term c_h $(1 + s_h)$.

There are some potential disadvantages to this approach however the Newton method is applied. First, ω may not be known in advance. Generally, the algorithms will converge however ω is chosen, but their convergence rates will not be optimum unless the right choice is made for ω. We have chosen to compromise in our application of this method by choosing ω to comprise the obviously dominant species and the remaining gaseous elements. We choose ω to be O(g), UO_2(c) and the remaining gaseous elements in the fuel region and H_2(g), H_2O(g) and the remaining gaseous elements elsewhere. A_{jh} is diagonal except terms involving O and U in the former case and terms involving H and O in the latter case. The O, U submatrix of A_{jh} in the former case is

$$\begin{bmatrix} 1 & 0 \\ -2 & 1 \end{bmatrix}$$

The H, O submatrix in the latter case was given previously.

With these matrices, it is easy to obtain the following results

$$g_{O(g)}{}^{(\omega)} = g_{O(g)} - 2g_{U(g)} \ ,$$

$$g_{UO_2(c)}{}^{(\omega)} = g_{U(g)} \ ,$$

$$J_{O(g)\ O(g)}{}^{(\omega)} = J_{O(g)\ O(g)} - 4J_{O(g)\ U(g)} + 4J_{U(g)\ U(g)} \ ,$$

and

$$J_{UO_2(c)\ UO_2(c)}{}^{(\omega)} = J_{U(g)\ U(g)}$$

for the fuel region, and

$$g_{H_2(g)}{}^{(\omega)} = \frac{1}{2}\, g_{H(g)} - g_{O(g)} \ ,$$

$$g_{O(g)}{}^{(\omega)} = g_{O(g)} \ ,$$

$$J_{H_2(g)}\ H_2(g)^{(\omega)} = \frac{1}{4}\ J_{H(g)}\ H(g) - J_{H(g)}\ O(g) + J_{O(g)}\ O(g) \quad ,$$

and

$$J_{H_2O(g)}\ H_2O(g)^{(\omega)} = J_{O(g)}\ O(g)$$

elsewhere.

The cyclically ordered Newton method will generally give convergence in a reasonable number of iterations. On occasions however, convergence is very slow. In these situations, after an input specified number of cyclic iterations have been tried, the full Newton iterations are performed. This will nearly always give convergence, but if it is still not converged then 3 additional combinations of cyclic and full iterations will be attempted. Should the iteration still not converge, VICTORIA will edit a series of warning messages, and if there is not a significant mass balance error, the calculation will continue.

A few situations have been found where there was a significant mass imbalance. Further investigation determined that, at this time, all of these were related to a failure of the first of the approximations made in subsection 5.2.3 (no initial condensed phase for a species). We have found that by repeating the chemistry calculation using the current value for n_c, this problem is resolved. VICTORIA therefore, attempts 3 additional iteration series using the current concentrations. If the chemistry calculation is still unable to converge, a final edit is produced and the total calculation terminated. This final edit provides all the information necessary to attempt a solution by hand.

These algorithms have been found to be effective, even when the initial guesses for c_h $h \in \omega$ are out by many orders of magnitude. For example, in the latter case, the oxygen potential is found always to be determined by $H_2(g)$ and $H_2O(g)$ and, hence, is calculated rapidly. Subsequently, the molar concentrations of the remaining species also converge rapidly. Typically, on the order of ten iterations are required to achieve a relative accuracy in the c_i's of 1 part in 10^8.

5.4.5 Convergence Criteria

The residual element abundances are zero at the exact solution therefore the absolute magnitudes of the residuals normalized by the element abundances, E_j $j \in e$, are a relative measure of how well the governing equations are satisfied. That is, we require

$$E_j = |g_j|/C_j << 1 \quad . \tag{5.51}$$

A relationship between these errors and the relative errors in the gaseous element molar concentrations can be found by doing a Taylor expansion about the c_j, giving

$$E_j \simeq \left| \sum_k \left\{ \frac{\sum_i a_{ij} a_{ik} c_i}{\sum_i a_{ij} c_i} \right\} \frac{\delta c_k}{c_k} \right| , \tag{5.52}$$

when δc_k $k \epsilon e$ is the error in c_k ($\delta c_k = c_k(\text{exact}) - c_k$).

The quantity in brackets is of order unity or zero, and therefore $E_j < \max_{k \epsilon e}(|\delta c_k / c_k|)$. E_j can be substantially smaller than this limit when J_{jk} is ill-conditioned. From Equation (5.32), it is found that $\max_{k \epsilon e}(|\delta c_k / c_k|) \simeq \max_{h \epsilon \omega}(|\delta c_h / c_h|)$ and therefore $E_j < \max_{h \epsilon \omega}(|\delta c_h / c_h|)$. Estimate $\delta c_h / c_h$ as s_h and require $\max_{h \epsilon \omega}(|s_h|) \leq \sigma$ where $\sigma << 1$. Typically, σ is chosen to be 10^{-8}. If s_h is a valid estimate of $\delta c_h / c_h$ then $\max_j(E_j) < \sigma$. However, caution is required: s_h can be significantly smaller than $\delta c_h / c_h$ (for example, when many small steps are taken in the direction of the solution). Therefore $\max_j(E_j)$ should be calculated to confirm that convergence has been achieved.

5.5 Zirconium Oxidation Model

5.5.1 Urbanic and Heidrick Experiments

The oxidation model for the zirconium in the fuel rod cladding tube is based on the work of Urbanic and Heidrick [5.13]. Their experiments and findings are summarized here.

Steam reacts with Zr to form a superficial layer of ZrO_2. It was known beforehand that an intermediate layer of oxygen stabilized α-Zr will also be formed when the temperature is above 1150K, the $(\alpha + \beta)/\beta$ phase transformation temperature, and that the mechanism which governs the reaction is diffusion of oxygen anions through the ZrO_2 lattice.

Urbanic and Heidrick performed experiments with Zircaloy-2 and Zircaloy-4 alloys to study whether the reaction rate could be affected by reducing the steam concentration, to measure the growth rates of the ZrO_2 and α-Zr layers individually, and to renove uncertainties in previously published data by obtaining data over a wide temperature range (1323K to 2123K, just below the Zr melting point at 2140K).

The experiment involved exposing heated Zircaloy bar stock (2.54cm by 1.25cm diameter) to pure steam at atmospheric pressure and constant temperature, and measuring the amount of hydrogen evolved as a function of time. The final amount of hydrogen collected was converted to the equivalent amount of Zr reacted according to the stoichiometry

$$Zr + 2H_2O \rightarrow ZrO_2 + 2H_2 \qquad (5.53)$$

and checked by measuring the weight gain of the sample; good agreement was always obtained.

The specimen surface temperature was measured with an optical pyrometer calibrated against the melting points of metals. The ZrO_2 and α-Zr layer thicknesses were measured, and the metallography of both layers was studied in all specimens.

The square of w, the weight of Zr consumed per unit surface area exposed, was plotted against exposure time, t. At given temperature, the points correlated well with a straight line after allowing for transient heating effects in the early stages of the experiments done at higher temperatures. A rate constant K_p, was determined from the slope of each line. The results are therefore consistent with

$$w^2 = K_p t \ , \qquad (5.54)$$

where K_p depends on temperature.

The tempe.:ature dependence of the rate constants was studied on an Arrhenius plot ($\ln(K_p)$ versus 1/T, where T is absolute temperature).

A discontinuity was seen between the points at 1738K and 1853K. Above and below the discontinuity, the following equations were found to correlate the points well

$$\left.\begin{aligned} K_p &= 29/6\ e^{-16820/T} \quad 1323K \le T \le 1738K \\ K_p &= 87.9\ e^{-16610/T} \quad 1853K \le T \le 2123K \end{aligned}\right\} \qquad (5.55)$$

where the units of T and K_p are Kelvins and $kg^2m^{-4}s^{-1}$ respectively.

Growth rate constants $\delta_{\alpha\text{-Zr}}$ and δ_{ZrO2} were calculated from

$$\delta_{\alpha\text{-Zr}} = \xi_{\alpha\text{-Zr}}/t^{1/2} \qquad (5.56)$$

and

$$\delta_{ZrO_2} = \xi_{ZrO_2}/t^{1/2} \ , \qquad (5.57)$$

where $\xi_{\alpha\text{-Zr}}$ and ξ_{ZrO2} are the measured thicknesses of the α-Zr and ZrO_2 layers. On an Arrhenius plot, the growth rates were found to be well correlated by the following equations.

$$\delta_{\alpha-Zr} = 3.90 \times 10^{-3}\ e^{-9915/T} \quad 1323K \le T \le 2123K \qquad (5.58)$$

and

$$\left.\begin{aligned} \delta_{ZrO_2} &= 3.60 \times 10^{-4}\ e^{-6793/T} \quad 1323K \le T \le 2123K \\ \delta_{ZrO_2} &= 1.44 \times 10^{-3}\ e^{-8007/T} \quad 1853K \le T \le 2123K \end{aligned}\right\} , \qquad (5.59)$$

where T is in Kelvins and the units of $\delta_{\alpha-Zr}$ and δ_{ZrO2} are $ms^{-1/2}$.

The discontinuity in δ_{ZrO2} coincided with Urbanic and Heidrick's metallographic observations of smaller and less well defined stringers and globules of α-Zr in the inner part of the ZrO_2 layer of the samples oxidized at and above 1853K, and with the discontinuity seen in K_p. They concluded that this was consistent with the Zr-O phase diagram which suggests that the ZrO_2 layer comprises ZrO_2 (tetragonal) below 1850K [5] and is subdivided into ZrO_2 (tetragonal) and ZrO_2 (cubic) above 1850K.

The measured growth rates of the ZrO_2 and α-Zr layers were shown to be consistent with published data for oxygen diffusion in ZrO_2 and α-Zr. Urbanic and Heidrick concluded that the oxidation was indeed controlled by oxygen diffusion in the solid phase.

5.5.2 Implementation of the Model

In VICTORIA, it is sufficient to calculate the takeup of steam by cladding oxidation. The heat generated and the oxide layer thickness are required in MELPROG [2.1], but are not calculated here. VICTORIA assumes that there is no essential difference between the experiment and reactor situations.

The model is based on the correlations for the rate of Zr removal, w, Equations (5.54) and (5.55). The ranges of temperatures in the equation for K_p are modified to

$$\left.\begin{aligned} K_p &= 29.6\ e^{-16820/T} \quad T \le 1763K \\ K_p &= 87.9\ e^{-16610/T} \quad T > 1763K \end{aligned}\right\} \qquad (5.60)$$

[5] More recent publications (for example, reference [5.14]) shown the phase transition to occur at 1763K. This is not inconsistent with the Urbanic and Heidrick data.

(note that T can be as large as the melting point of zirconia at ~2800K) and the equation for w is generalized to take account of changes in temperature. Assume

$$\frac{dw^2}{dt} = K_p \ . \qquad (5.61)$$

Discretizing this gives

$$\Delta w = (w^2 + K_p \Delta t)^{1/2} - w \ , \qquad (5.62)$$

where $w(t + \Delta t) = w(t) + \Delta w$, and Δt is the time step.

This is assumed to apply irrespective of the steam concentration in the gas region. However, Δw is also limited by the availability of steam and zirconium and Equation 5.62 must therefore take into account the stoichiometric relationship, Equation 5.53, and be modified as

$$\Delta w = \min\{(w^2 + K_p \Delta t)^{1/2} - w \ , \ n_{Zr} m_{Zr}/A \ , \ n_{H_2O} m_{Zr}/2A\} \ , \qquad (5.63)$$

where n_{Zr} is the number of moles of unreacted zirconium remaining in the cladding region, n_{H2O} is the number of moles of steam in the gas region, m_{Zr} is the molecular weight of zirconium, and A is the area of exposed cladding.

The changes in n_{Zr} and n_{H2O}, and the number of moles of hydrogen in the gas region, n_{H2}, are then given by

$$\Delta n_{Zr} = \Delta w A/m_{Zr} \ , \ \Delta n_{H_2O} = 2\Delta n_{Zr} \ , \ \Delta n_{H_2} = -2\Delta n_{Zr} \ , \qquad (5.64)$$

where the amount of oxygen taken up as oxygen-stabilized α-Zr and in the inner β-Zr region is assumed to be negligible.

6. RADIOACTIVE DECAY HEATING

The treatment of the heating of the reactor vessel environment by the decay of radioactive species is described in this section. Expressions are derived for the fuel material, aerosol particles, and bulk gases that are contained within the vessel. Decay powers are calculated only for those species that are identified as "VICTORIA species;" other radioactive species that may be associated with the fuel, but are not tracked by VICTORIA, must be treated by some external means.

Volumetric and surface decay powers produce a temperature feedback effect between the fission product species and the vessel environment. Release of radioactive species from the fuel rod decreases the decay heating levels in the fuel, changes its temperature profile, and thus affects the species release via the temperature-dependent chemistry and transport coefficients. Once released from the fuel, the same chain of events is again repeated as the radioactive species heat the bulk gas, aerosols, and structure surfaces. It should be noted that, although VICTORIA does calculate the new geometric distribution of decay powers, the code does not solve the appropriate energy conservation equations that would use these powers to calculate temperature profiles. This solution would be performed by other reactor safety codes that VICTORIA will ultimately couple with (MELPROG or TRAC). Indeed, the proper modeling of this temperature feedback would require a "hard" coupling (information would be transferred between codes every time step) of VICTORIA to one of these thermal-hydraulics codes.

6.1 Volumetric and Surface Decay Powers

The activity of radioisotope k of element j is determined by a decay constant, λ_{kj}, and an isotopic number density, N_{kj}, as

$$A_{kj} = \lambda_{kj} N_{kj} \quad , \tag{6.1}$$

where A_{kj} has the dimensions of decays per time per unit volume. This activity can then be converted into a power density by multiplying by the energy of decay Q_{kj}

$$P_{kj} = \gamma_{kj} N_{kj} Q_{kj} \quad . \tag{6.2}$$

If we now define γ_{kj} as the fraction of isotope i present in the number density of element j, and a_{ji} as the number of atoms of element j in species i, then the isotopic number density can be written

$$N_{kj} = \gamma_{kj} \sum_{i=1}^{n_s} a_{ji} N_i \quad , \tag{6.3}$$

where N_i is the number density of species i, and n_s is the total number of species under consideration. Substituting Equation (6.3) into Equation (6.2), and defining the summation of P_{kj} over elements and isotopes to be the total decay power, P, gives

$$P = \sum_{j=1}^{n_e} \sum_{k=1}^{K_j} \lambda_{kj} Q_{kj} \gamma_{kj} \sum_{i=1}^{n_s} a_{ji} N_i \quad , \tag{6.4}$$

where n_e is the total number of elements under consideration, and K_j is the number of isotopes associated with element j. The dimension of the total decay power is energy per time per unit volume. Equation (6.4) is used to calculate decay powers for the fuel open-porosity region, the bulk gas, and the aerosol particles. For the structure surface decay powers, Equation (6.4) is multiplied by the structure film thickness, Δ_f.

6.2 Information for the Code User

To obtain the radioactive decay data needed for Equation (6.4) (λ_{kj}, γ_{kj}, and Q_{kj}), the ORIGEN code [2.4] is used to perform the appropriate nuclear fuel depletion calculation, and thus provide not only the decay data, but also the initial radioisotope inventory for the fuel geometry release calculation described in Chapter 3. The ORIGEN code is not unique in its ability to perform depletion calculations, and other codes exist that can provide this decay and initial condition information to VICTORIA (for example, the CINDER code [6.1]).

The most important assumption behind the use of Equation (6.4) is the neglect of decay chains. Isotopes are not lost from the number densities when they decay, and the creation of radioactive daughters upon decay of their parents is ignored. This has significant implications for long time (> 3 hr) accident sequences because, for example, the fission fragment tellurium will decay into iodine and xenon on that time scale. Only by running several depletion calculations will the user be able to determine if the change in number densities for a given sequence is important.

A second assumption is also implied from the neglect of decay chains. It is that there is no strict accounting of whether the decay emitted β's or γ's. VICTORIA uses only one decay energy value, Q_{kj}, for each element and therefore the fact that β's are locally deposited and that γ's will penetrate (~40% absorption in the thin control rod housing and ~85% absorption in the first inch of other structures) must be combined by the user in Q_{kj}.

The decay powers are calculated in subroutine DECAY for the fuel geometry (which sums both the open porosity and grain contributions), the bulk gas flow (which for this purpose includes the aerosol particles), and for the structure surfaces. The simple calculational flow of subroutine DECAY is illustrated in Figure 6.1.

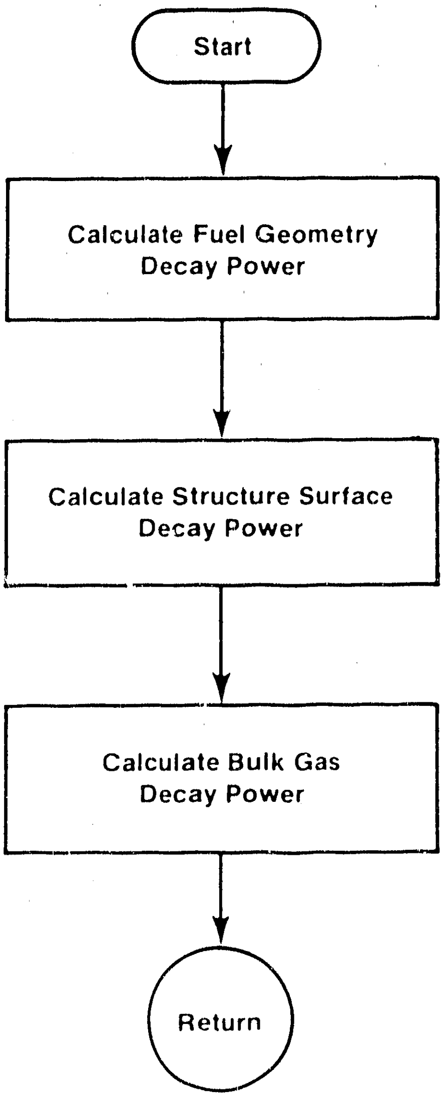

Start
Calculate Fuel Geometry Decay Power
Calculate Structure Surface Decay Power
Calculate Bulk Gas Decay Power
Return

Figure 6.1 Flowchart of the logic flow in the DECAY subroutine.

7. AEROSOL SPECIES BEHAVIOR AND TRANSPORT

The aerosol treatment in VICTORIA uses the CHARM model developed by Wheatley [1.5]. CHARM models aerosol behavior in a single computational cell; time-varying external conditions are assumed to have been calculated in advance and are supplied as data to the model. The aerosol particles have a single, constant composition and can agglomerate, deposit on surfaces, and leak from the cell. The transport of aerosol particles between adjacent cells provides a time-varying source and sink of particles for each cell. A combination of the agglomeration and deposition models from MAEROS [7.1] and TRAP-MELT2 [7.2] have been used in VICTORIA. In addition, models have been added to treat additional deposition by turbulence and to estimate boundary layer thicknesses and turbulence properties of the flow field.

7.1 Description of the Model

The governing equation of the aerosol phenomena treated by CHARM is the following.

$$\begin{aligned}\frac{\partial C(m,t)}{\partial t} &= \frac{1}{2}\int_0^m K(\omega,m-\omega,t)C(\omega,t)C(m-\omega,t)\,d\omega \\ &- C(m,t)\int_0^\infty K(\omega,m,t)C(\omega,t)\,d\omega \qquad (7.1)\\ &- R(m,t)C(m,t) + S(m,t)\end{aligned}$$

where C(m,t) is the aerosol number concentration distribution. Hence, C(m,t)dm is the number of particles per unit volume with mass in the range m to m+dm at time t. Drake [7.3] gives an excellent survey of the properties of Equation (7.1).

K is the agglomeration kernel; it is symmetric and takes account of particle-particle collisions due to Brownian motion, differential gravitational settling, and turbulence. The integral of the first term with K corresponds to the production of particles of mass m due to collision and coalescence of particles of masses m - ω and ω. The integral of the succeeding term corresponds to destruction of particles of mass m due to collision and coalescence with particles of mass ω. K can depend explicitly on time through changes in the external conditions; gas temperature, pressure and velocity are particularly important.

R is the removal rate for particles of mass m. Leakage, and deposition onto surfaces induced by thermophoresis, diffusiophoresis, gravitational settling, Brownian diffusion, and turbulence all contribute to R which, like K, may be time-dependent.

S is the number concentration source rate for particles of mass m.

Implicit in Equation (7.1) are four assumptions: First, the aerosol is well-mixed throughout the cell. This may require that the flow in the cell be turbulent in order to promote mixing, or it may require the cell to be a small part of a larger region within which the aerosol exists and the aerosol has nearly uniform properties within this cell by virtue of its small size. Second, particles are characterized by their mass only, that is, particles can have a shape which is a function of m, but particles of given mass all have the same shape. Clearly, this assumption is invalid because actual particles having the same mass will have different shapes, which affects agglomeration and deposition rates. Such dependence, however, would be extremely difficult to treat computationally. Third, boundary layers and their effect on deposition are not treated in detail. Fourth, particles do not break-up into smaller particles.

Detailed expressions for K, R, and S will be considered in the succeeding sub-sections where the primary aim will be to give formulae in detail and to reference their origin. Ranges of validity and possible uncertainties will be indicated where appropriate. S.I. units are used throughout except for molecular weights, which have units of kg/kmole.

7.1.1 Gas Properties

Agglomeration and deposition rates depend on the properties of the gas in the bulk of the cell, and adjacent to surfaces where they are relevant to the calculation of diffusiophoresis. It is assumed that a mixture of hydrogen and steam is the dominant component of the bulk gas. The thermophysical properties in the bulk gas are determined from the hydrogen and steam composition. The formulations are generally taken from Bird [3.9] unless otherwise noted.

The dynamic viscosity of the bulk gas, μ_g, is estimated from a correlation of data for hydrogen, steam or a mixture of the two. The viscosity correlations are as follows:

$$\mu_{H_2} = \frac{5 \left[\frac{MkT}{\pi N_o}\right]^{1/2}}{(16\ \Omega\ \sigma^2)} \qquad (7.2)$$

where M is the molecular mass of H_2, k is Boltzmann's constant, T the temperature, N_o is Advogadro's number, Ω is a Lennard-Jones collision integral, and σ is the Leonard-Jones collision diameter for H_2. The correlation for steam is taken from Meyer [7.4] as

$$\mu_{H_2O} = \mu_r \left(2 \frac{T}{T_r} - 1\right) \tag{7.3}$$

where μ_r and T_r are constants with the values 3.082 x 10^{-6} Pa-s and 151.4 K, respectively.

For a mixture of hydrogen and steam the viscosity is given by

$$\mu_g = x_{H_2} \mu_{H_2} / \left(x_{H_2} + x_{H_2O} \phi_{H_2-H_2O}\right) + x_{H_2O} \mu_{H_2O} / \left(x_{H_2} \phi_{H_2O-H_2} + x_{H_2O}\right) \quad ; \tag{7.4}$$

where

$$\phi_{kl} = \left[1 + \left(\frac{\mu_k}{\mu_l} \left(\frac{M_l}{M_k}\right)^{1/2}\right)^{1/2}\right]^2 \left(8 \left(1 + \frac{M_k}{M_l}\right)\right)^{-1/2} \tag{7.5}$$

and x_{H2}, x_{H20} are the mole fractions of the indicated species in the mixture.

The mean free path, l_p, is calculated from kinetic theory as

$$l_p = \mu_g \left[\frac{\pi}{2P\rho_g}\right]^{1/2} , \tag{7.6}$$

where P is the pressure, and ρ_g is the gas density.

The vapor concentration, c_s, in the vapor/gas mixture adjacent to a surface is estimated from the ideal gas law as follows:

$$c_s = \frac{(1-x_s)PM_v}{10^3 RT_s} , \tag{7.7}$$

where x_s is the molar fraction of the aerosol in the mixture, M_v is the molecular weight of the vapor, and T_s is the temperature of the mixture adjacent to the surface. The subscript s, stands for cl, w, or fl according to whether the surface is the ceiling, wall or floor (see Section 7.1.4 for the definition of these surfaces).

7.1.2 Aerosol Properties

The evaluation of the Schmidt and Knudsen Numbers as well as the particle mobility with the slip correction due to Cunningham are necessary to evaluate the aerosol behavior.

The particle density, ρ_p, is calculated from a weighted average of the condensed species in a cell. The densities of the species are set in data statements and can be modified by the user for all cells or modified for particular cells in the input as discussed in Appendix B.

The particle radius, r_p, is determined from the ratio of the average mass at the collocation point to the particle density.

The particle Knudsen number, K_n, is the ratio of the mean free path l_p to the particle radius r_p.

$$Kn = \frac{l_p}{r_p} \tag{7.8}$$

The particle mobility, B, is determined from Stoke's mobility law, B_{St}, with Cunningham's slip correction [7.5], C_n as

$$B = B_{St}\ C_n \quad . \tag{7.9}$$

With B_{St} given by

$$B_{St} = \frac{1}{6\pi \chi_d \mu_g r_p} \quad , \tag{7.10}$$

where the dynamic shape factor, χ_d, is an empirical dimensionless correction factor introduced to account for the aerodynamic effects associated with nonspherical particles. A value for the shape factor can be supplied by the user, the default value of 1.0 represents spherical particles.

C_n is given by

$$C_n = 1 + k_a Kn + k_q Kn\ e^{-k_b/Kn} \quad , \tag{7.11}$$

where Kn is the particle Knudsen number and k_a, k_q, and k_b are empirical dimensionless constants. Values for these constants can be supplied by users in the input deck. Default values used for k_a, k_q, and k_b (1.25, 0.41, and 0.88, respectively) are from Davies [7.6].

The particle Schmidt number, Sc, is given by

$$Sc = \frac{\mu_g}{\rho_g D_p} \quad , \tag{7.12}$$

where D_p is the particle Brownian diffusivity and is given by

$$D_p = kTB \quad , \tag{7.13}$$

where k is the Boltzmann constant and B is the mobility.

7.1.3 Flow Properties

Turbulence parameters of the bulk flow are needed for estimating agglomeration and deposition rates. Viscous and diffusion boundary layer thicknesses are needed for estimating thermophoretic deposition.

The friction velocity, v_*, is calculated from formulae based on those given in Schlichting [7.7] for flow through a cylindrical pipe of any aerodynamic roughness and other formulae for flow through aerodynamically smooth pipes of arbitrary cross-sectional shape. By definition, v_* in terms of the Fanning friction factor, f_*, (see below) is given by

$$v_* = v_g \left[\frac{f_*}{2}\right]^{1/2} \quad , \tag{7.14}$$

where v_g is the mean flow speed of the bulk gas in the cell relative to surfaces in the cell, usually the cell walls.

The Fanning friction factor, f_*, is determined from a correlation established by Colebrook [7.8] for turbulent flow through cylindrical pipes of any aerodynamic roughness. The correlation is

$$\frac{1}{2\sqrt{f_*}} = 1.74 - 2 \log_{10}\left[\frac{2z_s}{d_h} + \frac{18.7}{Re2\sqrt{f_*}}\right] \quad , \tag{7.15}$$

where z_s is the equivalent sand roughness of the pipe surface adjacent to the flow, d_h is the diameter of the pipe, and Re, the pipe Reynolds number is given by

$$Re = \rho_g d_h v_g / \mu_g \quad . \tag{7.16}$$

It is assumed that $f_*=0$ when Re<2300 (laminar flow).

Equation (7.15) reduces to the quadratic resistance formula for completely rough pipes ($\rho_g z_s v_* / \mu_g > 70$), derived by von Karman, and Prandtl's universal law of friction for smooth pipes ($\rho_g z_s v_* / \mu_g < 5$).

The equivalent sand roughness, z_s, equals the height of protrusions on sand roughened pipes of equivalent roughness. z_s must, in general, be determined empirically but values have been established for common materials, which are given in Table 7.1 which has been reproduced from Schlichting [7.7] for pipes of 1 m in diameter.

Table 7.1

Pipe Roughness Heights for Some Common Materials

material	z_s (m)
reinforced concrete	.0003 - .003
cast iron	.00026
galvanized steel	.00015
structural and forged steel	.000045
drawn pipes	.0000015

Note that certain types of protrusion, such as regularly spaced ribs perpendicular to the flow, can give rise to values of z_s significantly larger than the height of the protrusion.

The hydraulic diameter, d_h, is typically the diameter of the pipe. However, experiments by a number of authors have shown that the Fanning friction factor of aerodynamically smooth pipes of non-circular cross-section (such as square, rectangular, triangular and annuli) equals that of circular pipes over a wide range of Reynolds number when d_h is generalized as given in Equation (7.17), despite the complications of secondary flows induced in non-circular pipes. d_h is given by

$$d_h = \frac{4A}{p} , \qquad (7.17)$$

where A is the cross-sectional area and p is the "wetted" perimeter of the pipe perpendicular to the flow. This generalization is also used for aerodynamically rough pipes, even though it has only been verified for smooth pipes.

The Nusselt number, Nu, variation with flow uses the models explained in the TRAC Correlations and Models document [7.9] which are:

for forced convection

in turbulent flow (Re>2300.)

$$Nu = 0.023\ Re^{0.8} \quad , \qquad (7.18)$$

in laminar flow (Re<2300.)

$$Nu = 4 \quad , \qquad (7.19)$$

for natural convection

in laminar flow ($Ra<10^9$)

$$Nu(Ra) = 0.59\ Ra^{\ 0.25} \quad , \qquad (7.20)$$

in turbulent flow ($Ra>10^9$)

$$Nu(Ra) = 0.1\ Ra^{0.333} \quad , \qquad (7.21)$$

using the maximum value

$$Nu = \max\ (\ Nu,\ Nu(Ra)\ ) \quad . \qquad (7.22)$$

where Re is the Reynolds number and Ra is the Rayleigh number defined as

$$Ra = \frac{Pr\ g_c \left| T_s - T_g \right|\ \rho_g^2\ d_h^3}{\mu^2\ T_s} \qquad (7.23)$$

where g_c is the accel ration due to gravity and the Prandtl number, Pr, is assumed to be 1.0.

A set of boundary layers are also calculated. First a thermal boundary layer, δ_*, as

$$\delta_* = \frac{d_h}{Nu} \quad . \qquad (7.24)$$

and also a diffusion boundary layer, δ_D, estimated from Keller [7.10], assuming that the Prandtl number is 1.0, as

$$\delta_D = \frac{\delta_*}{Sc^{0.333}} \qquad (7.25)$$

where Sc is the particle Schmidt number. Both of these thicknesses may be set in the input to constant values, as discussed in Appendix B.

7.1.4 Deposition Rates

Six deposition mechanisms are described (gravitational, turbulent, laminar, thermophoretic, diffusiophoretic, and bend). The surfaces(s) exposed to aerosols within the cell and at the cell boundaries are denoted as ceiling(cl), wall(w), and floor(fl) which are horizontal and downward facing, vertical, and horizontal and upward facing, respectively.

The deposition velocity due to gravitational settling, u_G, is given by

$$u_G = g_c mB \quad , \tag{7.26}$$

where g_c is the acceleration due to gravity. Strictly, a term $(\rho_p - \rho_g)/\rho_p$ should appear as a multiplicative factor in this equation, but this small error is ignored.

The deposition of aerosol particles from turbulent flow can be divided into supermicron and submicron deposition.

For supermicron particle deposition, a correlation derived by Sehmel [7.11] from data for particles with diameters greater than one micron is used. This correlation is given by the formula

$$u_{*1} = 1.47 \times 10^{-16}(.001\ \rho_p)^{1.01}\ \bar{R}^{2.1}\ Re^{3.02}\ v_* \tag{7.27}$$

where ρ_p is the particle density and R is the ratio of the particle diameter in μm to the hydraulic diameter in cm. This is implemented as

$$\bar{R} = \left(2r_p \times 10^4\right)/d_h \ . \tag{7.28}$$

A theoretical model by Davies [7.6] is used to calculate deposition of submicron particles from turbulent flow. This model was derived for particle deposition onto smooth surfaces and assumes that the particle transport can be described as the sum of the particle Brownian diffusivity and the turbulent diffusivity of the fluid. The formula is given by

$$u_{*2} = \frac{Sc^{-2/3} \; v_*}{14.5 \left[\frac{1}{6} \ln \left(\frac{(1 + \phi)^2}{1 - \phi + \phi^2} \right) + \frac{1}{\sqrt{3}} \tan^{-1} \left(\frac{2\phi - 1}{\sqrt{3}} \right) + \frac{\pi}{\sqrt{3}} \right]} \qquad (7.29)$$

where $\phi = Sc^{1/3}/2.9$ and Sc is the Schmidt number from Equation (7.12).

To calculate the turbulent deposition velocity for both particle sizes, these two values are added, giving

$$u_* = u_{*1} + u_{*2} \qquad (7.30)$$

Particle deposition from laminar flow is a well analyzed phenomena. The deposition velocity is calculated as

$$u_L = \left(\frac{C_o}{C} - 1 \right) \frac{d_h}{L_n} \; v_g \qquad (7.31)$$

where L_n is the length of the cell and the fraction of particles deposited from the bulk flow, C_o/C, is determined from the data of Gormley and Kennedy [7.12]

when $h > 0.0156$ as

$$\frac{C}{C_o} = 0.8191e^{-7.314h} + 0.0975e^{-44.6h} + 0.0325e^{-114h}, \qquad (7.32)$$

when $h < 0.0156$ as

$$\frac{C}{C_o} = 1 - 4.07 \; h^{2/3} + 2.4 \; h + 0.446 \; h^{4/3} \qquad , \qquad (7.33)$$

where h is a measure of the diffusive to convective transport and is given as

$$h = \frac{2 \; L_n \; D_p}{d_h^2 \; v_g} \qquad . \qquad (7.34)$$

where D_p was defined in Equation (7.13).

Thermophoretic deposition, or the motion of aerosol particles due to a thermal gradient, was first observed by Tyndal [7.13] in 1870, who observed a dust-free zone surrounding a hot body. Under the influence of a thermal gradient and friction in a fluid, in the steady state, aerosol particles are observed to move with constant velocity toward the lower temperature. The thermophoretic force arises from collisions between the aerosol particles and the fluid molecules. Since the mean momentum transfer due to these collisions is proportional to the square of the gas temperature, there is a net force exerted on the aerosol in the direction of the lower temperature region.

The hydrodynamic analysis of this phenomenon was performed by Brock [7.14] in 1962. The derived formula is still regarded as being the most reliable and is used by most aerosol physics codes. The Brock formula for the thermophoretic deposition velocity, u_{Ts}, for a particle of radius r_p upon a surface s, is given by

$$u_{Ts}(r_p) = \frac{-2\, C_s \left[\frac{k_g}{k_p} + C_t\, K_n(r_p)\right] \frac{C_n\, \mu_g}{\rho_g}\, \frac{T - T_s}{T\, \delta_*}}{(1 + 3\, C_m\, K_n(r_p)) \left[1 + 2\, \frac{k_g}{k_p} + 2\, C_t\, K_n(r_p)\right]} \qquad (7.35)$$

where K_n is the particle Knudsen number, δ_* is the thermal boundary layer thickness, and k_g/k_p is the conductivity ratio set in VICTORIA to 0.01. The thermal slip coefficient, C_s, is set to 0.75 as first obtained by Maxwell [7.15]. The thermal accommodation coefficient, C_t, derived from kinetic theory, is given the value 2.49 and the momentum accommodation coefficient, C_m, is set to 1.0. More refined kinetic theory analysis [7.16, 7.17] and comparison with experimental data [7.18] have shown these coefficients to be in error. In VICTORIA the default input values for these coefficients use this information, they are:

$$C_s = 1.17,\; C_t = 2.18,\; \text{and}\; C_m = 1.14.$$

According to Talbot [7.19], the agreement between the theoretical thermophoretic deposition velocity calculated with these new coefficients and the experimental data is good over the entire range of Knudsen numbers $0 \leq K_n(r_i) \leq \infty$, though at most, these predictions will only be 50 percent greater than those calculated using the earlier coefficients.

The diffusiophoretic deposition velocity, u_{Ds}, of a species on a surface s, where s can stand for cl, w, or fl, is estimated from

$$u_{Ds} = D_p \frac{dc_s}{dx} \frac{1}{c_s} \frac{f_s}{f_s + (1 - f_s)\, (M_g/M_p)^{1/2}} \; , \qquad (7.36)$$

where c_s is the vapor species concentration Equation (7.7), dc_s/dx is the outward facing vapor concentration gradient near the ceiling, wall, or floor, and f_s is the fraction of the total

The calculated deposition rate is added to the total wall deposition rate for that control volume. Users can identify those control volumes incorporating bends with a flag i: the input data (see Appendix B).

The net deposition velocity to any surface, u_s, is estimated by combining the contributions in the following way

$$u_{cl} = u_L + u_* + u_{Tcl} + u_{Dcl} - u_G \quad , \tag{7.40}$$

$$u_w = u_L + u_* + u_{Tw} + u_{Dw} + u_b \quad , \tag{7.41}$$

and

$$u_{fl} = u_L + u_* + u_{Tfl} + u_{Dfl} + u_G \quad . \tag{7.42}$$

Note that since either or both of u_{Ts} and u_{Ds} could be negative (that is, when T_s is greater than T, or dc_s/dx is negative) and u_G is subtracted in Equation (7.40) some of the u_s may be negative. The equations above are only applied when the result for u_s is positive; otherwise, u_s is set to zero.

The deposition rate to a surface per unit cell volume, λ_s, is given by

$$\lambda_s = u_s A_s / V \quad , \tag{7.43}$$

where A_s is the area of surface, s, exposed to aerosol. R(m,t) is then given by

$$R(m,t) = \lambda_c + \lambda_w + \lambda_f \quad . \tag{7.44}$$

All possible deposition mechanisms have not been included in this treatment. Some of the mechanisms that were left out are deposition mechanisms associated with electrostatic charges.

7.1.5 Agglomeration Rates

Mechanisms that result in the collision and subsequent coalescence of particles may be classified as agglomeration processes. Three agglomeration processes are considered by VICTORIA 1) Brownian motion, 2) gravity, and 3) turbulence. Each of these processes cause particles of different masses to have a non-zero relative velocities and therefore collide. In VICTORIA, a particle-particle sticking efficiency, ϵ_s, is included in the equations to provide the coalescence probability. This efficiency may be modified in the input, the code uses a default value of 1.0 for ϵ_s.

surface area in a cell that is associated with the wall, ceiling, or floor. Comparison with Derjaguin and Yalamov [7.20] shows this estimate to be applicable in the limit Kn $\geq$ 1 and f_s << 1, and when the flow of vapor to or from the surface is purely diffusive.

The ability to model the behavior of aerosols as they are transported through 90° changes in the flow streamlines has been incorporated into VICTORIA. The model used in VICTORIA is based on experimental and theoretical work by Pui, Romay-Novas and Liu [7.21]. The experiments included a range of pipe diameters and surface materials, and also covered flow Reynolds numbers in the range 100-10,000. Monodisperse aerosols were generated using a vibrating orifice device, and the results were found to be in good agreement with theoretical predictions of Cheng and Wang [7.22] for turbulent conditions. An exponential curve of deposition efficiency versus Stokes number was fitted to the experimental data, and a theoretical justification of the form of the curve is given in the paper.

The Pui model gives the deposition efficiency, η_b, of aerosols in turbulent flow within a bend as

$$\eta_b(r_p) = 1 - 10^{-0.963\ St(r_p)} \qquad (7.37)$$

where $St(r_p)$ is the particle Stokes number, defined as

$$St(r_p) = \frac{4\ Cn\ \rho_p\ r_p^2\ v_g}{9\ \mu\ d_h} \quad . \qquad (7.38)$$

where Cn is the Cunningham slip correction factor.

For laminar flow conditions, the bend deposition efficiency is based on the data of Cheng and Wang [7.22] for a Reynolds' number of 1000. These data indicate that the the efficiency, η_b, equals 1.0 for Stokes numbers greater than 1.0 and gradually decreases to 0. as the number falls to 0.15. VICTORIA uses a data table to implement this efficiency information.

The deposition efficiency is then converted into a deposition velocity given as

$$u_b = \eta_b(r_p)\ /\ \Delta t \quad . \qquad (7.39)$$

where Δt is the time taken for the fluid to flow through the bend.

For notational convenience the agglomeration rate formulae are given corresponding to the j^{th} and k^{th} collocation points (see Section 7.2.1 for their definition).

The Brownian agglomeration rate, ϕ_B, is estimated as

$$\phi_B(m_j,m_k) = 4\pi kT\ (B_j+B_k)\ \chi_c(r_j+ r_k)\ Fu(m_j,m_k) \quad , \qquad (7.45)$$

where B_j and B_k are the particle mobilities, from Equation (7.9), at the j^{th} and k^{th} collocation points, respectively; χ_c is the input collision shape factor which corrects the collision cross-section of particles when they depart from spherical geometry (VICTORIA uses the default value for spheres of 1.0 for χ_c); and Fu is a factor introduced by Fuchs [7.23] to extend the particle range. The Fuchs interpolation factor provides a transition from small molecular sized particles determined from kinetic theory to particles larger than the mean free path, l_p, to be determined from diffusion theory. Fu is given by

$$1/Fu(m_j,m_k) = 1/Fu_1(m_j,m_k) + 1/Fu_2(m_j,m_k) \quad , \qquad (7.46)$$

where Fu_1, from kinetic theory, is

$$Fu_1(m_j,m_k) = \epsilon_s \frac{r_j+r_k}{kT(B_j+B_k)} \left[\frac{8kT}{\pi}\left(\frac{1}{m_j}+\frac{1}{m_k}\right)\right]^{1/2} \quad . \qquad (7.47)$$

and Fu_2, from diffusion theory, has been modified according to Sitarski and Seinfeld [7.24]

$$Fu_2(m_j,m_k) = 1 + \frac{2\sqrt{\left(\tilde{a}_j{}^2+\tilde{a}_k{}^2\right)}}{r_j+r_k} \quad , \qquad (7.48)$$

where $\tilde{a}$ is

$$\tilde{a}_j = \frac{(r_j+a_j)^3 - (r_j^2+a_j^2)^{3/2}}{3r_j a_j} - r_j \qquad (7.49)$$

and

$$a_j = B_j \left[\frac{2kTm_j}{\pi}\right]^{1/2} \quad . \qquad (7.50)$$

ϵ_s in Equation (7.47) is the input particle-particle sticking efficiency which is the probability that particles stick to one another when they collide.

The gravitational agglomeration rate, ϕ_G, is estimated as

$$\phi_G(m_j,m_k) = \pi\epsilon_s\epsilon_{Pk}\chi_c^2\left(r_j+r_k\right)^2|v_G(m_j)-v_G(m_k)| \quad . \qquad (7.51)$$

It arises from collisions of particles travelling under the influence of gravity at different velocities. v_G, is the terminal velocity due to the gravitational acceleration of a particle of mass, m, and is estimated as

$$v_G(m_j) = g_c m_j B_j \quad . \qquad (7.52)$$

Notice this is just u_G from Equation (7.26) but a different symbol is used here to avoid confusion. As with u_G, a factor $(\rho_p-\rho_g)/\rho_p$ could be included but is ignored to keep the equation identical to that used in MAEROS.

The collision efficiency, ϵ_{Pk}, is a correction factor which is applied to account for the deflection of the particle stream lines from straight-lines when they approach one another. For the collision efficiency, ϵ_{Pk}, VICTORIA follows Dunbar's recommendation [7.25], which is given by

$$\epsilon_{Pk}(m_j,m_k) = \frac{1}{2}\frac{\min\left(r_j,r_k\right)^2}{\left(r_j+r_k\right)^2} \quad , \qquad (7.53)$$

where *min* stands for the minimum value of r_j ana r_k. The terms r_j and r_k are the radii of the equivalent spherical particles evaluated at the j^{th} and k^{th} collocation points, respectively.

The turbulent shear agglomeration rate, ϕ_{*S}, is estimated from Saffman and Turner [7.26] as

$$\phi_{*S}(m_j,m_k) = \epsilon_s\chi_c^3(r_j+r_k)^3\left[\frac{8\rho_g\pi\epsilon_*}{15\mu_g}\right]^{1/2} \quad , \qquad (7.54)$$

where ϵ_s is the input particle sticking efficiency and χ_c is the input collision shape factor. The turbulent energy dissipation rate per unit mass, ϵ_*, is estimated from Deliachasios and Probstein [7.27] as

$$\epsilon_* = \frac{0.03146\ v_g^3}{d_h\ Re^{0.375}} \quad . \qquad (7.55)$$

This agglomeration mechanism accounts for the action of turbulent shear which causes particles that follow the instantaneous stream lines to collide with one another.

The turbulent inertia agglomeration rate, ϕ_{*I}, is also estimated from Saffman and Turner [7.26] as

$$\phi_{*I}(m_j,m_k) = \epsilon_s \chi_c^2 \; (r_j+r_k)^2 \left[\frac{512\rho_g \pi^3 \epsilon_*^3}{15\mu_g}\right]^{1/4} |v_G(m_j)-v_G(m_k)|/g_c \quad . \tag{7.56}$$

It accounts for particles colliding with one another when, due to their inertia, they are unable to follow the instantaneous stream lines.

The total agglomeration rate combines the above contributions as follows

$$K(m_j,m_k,t) = \phi_B(m_j,m_k) + \phi_G(m_j,m_k) + \{\phi_{*S}(m_j,m_k)^2 + \phi_{*I}(m_j,m_k)^2\}^{1/2} \quad . \tag{7.57}$$

According to Saffman and Turner [7.26], the turbulence contributions are added in a quadratic form. However, Dunbar, et al. [7.25] point out that the reasoning used by Saffman and Turner also applies to the gravitational contribution which should therefore be added to $K(m_j,m_k,t)$ in the same way. To maintain consistency with MAEROS, the above formulation is used.

Note that all possible agglomeration mechanisms have not been included. Mechanisms not accounted for are, for example, agglomeration in laminar shear flows and body force effects (such as, van der Waals and electrostatic forces).

7.1.6 Source Distributions

For convenience, analytic formulae are used for the source distributions although more general formulations could be accommodated. The source number concentration distributions are chosen to be log-normal. Consequently, the source density distribution, ms(m,t), is given by

$$ms(m,t) = \left[\frac{dC/dt}{(2\pi)^{1/2}\, \ln(\sigma^3)}\right] e^{-} \{\ln^2(m/m_g)/2\, \ln^2(\sigma^3)\} \tag{7.58}$$

where C is the aerosol number concentration, m_g is the geometric mean mass, and σ is the cube root of the geometric standard deviation with respect to mass.

C and m_g are defined in terms of ρ_p and r_{50} as

$$m_{50} = \frac{4}{3} \pi \rho_p \, r_{50}^{\;3} \, , \tag{7.59}$$

$$\frac{dC}{dt} = \frac{d\rho/dt}{m_{50}} \, e^{\ln^2(\sigma^3)/2} \, , \tag{7.60}$$

and

$$m_g = m_{50} \, e^{-\ln^2(\sigma^3)} \, , \tag{7.61}$$

where r_{50} is the radius of the spherically equivalent particle of mass m_{50} and density ρ_p. dC/dt and dρ/dt are, respectively, the number concentration generation rate (the number generation rate divided by the cell volume) and the density generation rate (the mass generation rate divided by the cell volume) of the source. The default values for r_{50}, 0.5μm, and for σ, 2.0, are set in data statements. As discussed in Appendix B, the user provides values for these parameters when inputting a particular aerosol distribution.

7.2 Numerical Method

7.2.1 Discretization of the Governing Equations

Equation (7.1) is discretized with respect to mass using the collocation finite-element method. In this method, the governing equation is required to hold only at a set of collocation points (n in total). A finite-element expansion based on values of variables at the collocation points is used. Here, this will be needed for the estimation of the agglomeration integrals.

The particular method used is an extension of that studied by Wheatley [1.5], who showed that accurate solutions to the equations could be obtained with a small number of collocation points and the agglomeration kernel evaluated on the n^2 pairs of collocation points only. The extension used is similar to that performed by and enables calculations to be performed with a multi-component aerosol. This allows the compositions of the aerosol to vary across the co-location points rather than the fixed composition previously imposed. A single component aerosol can still be chosen by the user and has the advantage of being much faster than the multi-component option due to the smaller number of equations solved.

Using a logarithmic scale, m is discretized as follows:

$$\ln(m_i) = \ln(m_1) + (i-1)h, \quad i=1\ldots n \ . \tag{7.62}$$

h is a constant which can be found from m_1, and m_n, the smallest and largest values of discretized mass respectively, and n. h is chosen to be constant.

With this choice for m_i it is convenient to choose the mass distribution, given by $Y(m,t) = mC(m,t)$, as the dependent variable in Equation (7.1). Then, representing the mass distribution of the q^{th} component of the aerosol as Y^q and Y as the sum of Y^q over all components, Equation (7.1) can be written as

$$\frac{\partial Y^q(m,t)}{\partial t} = \int_0^m K(\omega,m-\omega,t)Y(\omega,t)Y^q(m-\omega,t)d\ln(\omega)$$

$$- Y^q(m,t)\int_0^\infty K(\omega,m,t)Y(\omega,t)d\ln(\omega)$$

$$- R(m,t)Y^q(m,t) + mS^q(m,t) \quad , \tag{7.63}$$

where $S^q(m,t)$ represents the source rate of the q^{th} component of the aerosol, $K(\omega,m-\omega,t) = K(m-\omega,\omega,t)$ and, for economy of display here and later, the integration limits are shown for ω rather than $\ln(\omega)$.

When this equation is evaluated at the points m_i, i = 1, ...n, it is obvious that although the extended trapezoidal rule could be used to estimate the second integral on the right hand side, the first integral will be troublesome. A finite-element expansion is needed for the first integral. The integrands are expanded as

$$K(\omega,m-\omega,t)\ Y(\omega,t)\ Y^q(m-\omega,t) =$$

$$\sum_{j,k} K_{jk}\ Y_j\ Y_k^q\ g_j\left(\ln(\omega)\right)\ g_k\ \left(\ln(m-\omega)\right) \quad , \tag{7.64}$$

where Y_i and K_{ij} are shorthand for $Y(m_i,t)$ and $K(m_i,m_j,t)$, respectively, and the i^{th} element g_i is defined in terms of a basic element g, by

$$g_i\left(\ln(m)\right) = g\left(\ln(m) - \ln(m_i)\right) \diagup h \quad . \tag{7.65}$$

The precise form of g will be discussed later, but note that it is always chosen so that $g(0) = 1$ and $g(\pm1), g(\pm2), g(\pm3), \ldots = 0$. This guarantees that the expansion in Equation (7.64) is consistent in the sense that the equation is satisfied identically when ω and $m-\omega$ are located at any of the collocation points.

The particular expansion in Equation (7.64) has been chosen for three reasons: First, the agglomeration kernel needs to be evaluated only at the collocation points. Second, since K and Y have not been expanded separately, the summations which result in the discretized equation are over, at most, two indices. This reduces computational labor. And third, the same result for the second integral in Equation (7.63) is obtained as it would be by applying the trapezoidal rule.

Substituting Equation (7.64) into Equation (7.63) and evaluating at the ith collocation point gives the following discretized form of the aerosol equation:

$$\frac{\partial Y_i^q}{\partial t} = \sum_{j,k} P_{jk}^i K_{jk} Y_j Y_k^q - Y_i^q \sum_j D_j^i K_{ij} Y_j - R_i Y_i^q + m_i S_i^q \tag{7.66}$$

where R_i and S_i^q are shorthand for $R(m_i,t)$ and $S^q(m_i,t)$ respectively, and $g_j(x_k) = \delta_{jk}$, where δ_{jk} is the Kronecker delta function. The terms with P_{jk}^i and D_j^i correspond to the particle production and destruction terms, respectively, on the right hand side of Equation (7.63). The indices in P_{jk}^i and D_j^i run from 1 to n.

$$D_j^i = \int_0^\infty g_j(\ln(\omega))d\ln(\omega) = h\int_{-\infty}^\infty g(x)dx = h \quad , \tag{7.67}$$

as desired, where g(x) must be chosen such that $\int_{-\infty}^\infty g(x)dx = 1$.

$$P_{jk}^i = n_{jk}\int_0^{m_i} g_j(\ln(\omega))g_k(\ln(m_i - \omega))d\ln(\omega) \quad . \tag{7.68}$$

This is simplified somewhat by making the transformation $y = \ln(\omega/m_j)/h$ to obtain

$$P_{jk}^i = h\, n_{jk}\int_{-\infty}^{\bar{j}} g(y)g(\tfrac{1}{h}\ln(1 - e^{(y-\bar{j})/h}) + \bar{k})dy \quad , \tag{7.69}$$

where $\bar{j} = i - j$ and $\bar{k} = i - k$. The integral must be calculated numerically and the integration range must be divided-up into sub-ranges since g is generally not smooth. The indices in the coefficient P^i_{jk}, in principle, take all values from 1, ...n; however, only a small fraction of the coefficients are non-zero. It is straightforward to find the conditions on i, j and k for this to be so. It can be shown that they depend only on j and k. The non-zero values of the coefficients are conveniently stored consecutively using an indexing based on the derived conditions.

The multiplicative factor, n_{jk}, is introduced as a correction to conserve mass. By integrating Equation (7.63) with respect to m from 0 to ∞ the following mass balance equation is obtained:

$$\frac{\partial \rho^q}{\partial t} = - \int_0^\infty R(m,t)\ Y^q(m,t)\ dm + \int_0^\infty mS^q(m,t)\ dm \quad . \qquad (7.70)$$

This equation, multiplied by the cell volume, states that the rate of increase of airborne mass of aerosol component q equals the rate supplied by the sources of component q less the rate removed by deposition and leakage. Clearly, the agglomeration terms have cancelled as one would expect. However, this is not the case for the discretized equivalent to Equation (7.70) without the factor n_{jk} in P^i_{jk}. In many applications, the removal and source terms in Equation (7.63) can be small compared to the agglomeration terms. Therefore it is important that the agglomeration terms cancel exactly to ensure that the removal and source terms are not swamped by cancellation errors.

The trapezoidal rule is used to esti ate ρ, Equation (7.81), from which the discretized form of the mass balance equation is

$$\frac{\partial \rho^q}{\partial t} = h \left\{ \sum_{i,j,k} m_i\ P^i_{jk}\ K_{jk}\ Y_j\ Y^q_k - \sum_{i,j} m\ D^i_j\ K_{ij}\ Y^q_i\ Y_j \right.$$

$$\left. - \sum_i m_i\ R_i\ Y^q_i + \sum_i m_i^2\ S^q_i \right\} \quad . \qquad (7.71)$$

The first two terms on the right hand side must cancel exactly, whatever the values of Y_i for i = 1, ...n and for each Y^q. A sufficient condition is, therefore, to require that the sum of all coefficients of terms involving $Y_r Y^q_s$ for all r, s and q must be equal to zero. This leads to a relationship for P^i_{jk} and D^i_j;

$$\sum_i m_i \, P^i_{rs} = m_s \, D^s_r = hm_s \quad . \tag{7.72}$$

Three options for g are provided in the input to VICTORIA. Wheatley has shown [1.5] that g(x) gives the best results when n is small and x is evaluated in a piecewise linear form as

$$\begin{aligned} g(x) &= 1 - |x| \quad , \quad |x| < 1 \\ g(x) &= 0 \qquad\quad\; , \quad |x| \geq 1 \quad . \end{aligned} \tag{7.73}$$

This can be chosen as option 2 in the input, the default value.

This choice gives rise to continuous piece-wise linear finite-element expansions. Equation (7.73) has the advantage here that g is non-negative everywhere and therefore all the P^i_{jk} are non-negative. This is sufficient to ensure that no component of the solution of the discretized equation changes sign, which must be the case due to physical considerations.

7.2.2 Solution Method

The ODE's are solved with the mathematical software package, DEPAC, described in Appendix C. For each aerosol component n + 5 coupled ODE's are solved in all, the additional five equations are integrators for the source mass and the masses deposited on the wall, floor and ceiling. The total number of equations solved is therefore Q(n+5), where Q is the total number of aerosol components. As the time spent in the solver routines is approximately proportional to the number of equations squared, it can be seen that a significant computational price is paid for using the multi-component facility.

The local absolute error in Y_i is constrained during the integration as follows

$$\delta Y_i < \epsilon/h \text{ minimum}(N, \rho/m_i) \quad , \tag{7.74}$$

where ϵ is a relative tolerance parameter supplied by the user. This test is designed to result in estimates for N and ρ with a relative accuracy equal to or less than ϵ.

7.2.3 Moments of the Discretized Distribution

The moments considered are ρ, the aerosol density; N, the aerosol number concentration; m_{50}, the mass median mass, m_g, the geometric mean mass; and σ, the cube root of the geometric standard deviation with respect to mass. These moments are defined by

$$\rho = \int_0^\infty mY(m,t)\ d\ln(m) \quad , \qquad (7.75)$$

$$N = \int_0^\infty Y(m,t)\ d\ln(m) \quad , \qquad (7.76)$$

$$\rho/2 = \int_0^{m_{50}} mY(m,t)\ d\ln(n) \quad , \qquad (7.77)$$

$$N \ln\ (m_g) = \int_0^\infty \ln(m)\ Y(m,t)\ d\ln(m) \quad , \qquad (7.78)$$

$$\rho \ln^2(\sigma^3) = \int_0^\infty \ln^2(m/m_m)\ m\ Y(m,t)\ d\ln(m) \quad , \qquad (7.79)$$

where the cube power of σ is the conventional parameter, and m_m, the mass mean mass, is calculated as

$$\rho \ln(m_m) = \int_0^\infty \ln(m) m\ Y(m,t)\ d\ln(m) \quad , \qquad (7.80)$$

One departure from the CHARM methodology is the calculation of geometric standard deviation on mass rather than number. This change was made to avoid confusion, as experimental values for geometric standard deviation are always calculated on mass.

The total airborne aerosol mass is ρ times the cell volume. Similarly, N times the cell volume is the total number of airborne aerosol particles. m_{50} is sometimes called the mass median particle size. Approximately half the airborne particles have mass less than m_g. σ is a measure of the spread of the distribution. For a log normal number distribution 68% of the particles have masses in the range m_g/σ^3 to $m_g\sigma^3$ and 68% of the airborne mass derives from particles with masses in the range m_{50}/σ^3 to $m_{50}\sigma^3$. These relationships hold reasonably well for distributions found in practice.

The numerical estimation of these moments is now considered. Their evaluation with the extended trapezoidal rule is straightforward for all except m_{50} and illustrated only for ρ. The estimate for ρ is

$$\rho = h \sum_i m_i\ Y_i \quad , \qquad (7.81)$$

where Y_i is shorthand for $Y(m_i,t)$ and the summation extends over all values for which the indicated index is defined. $Y(m,t)$ has been assumed to decrease to a negligible value between m_1 and m_1e^{-h} and between m_n and m_ne^h.

m_{50} generally lies between adjacent grid points, therefore the integral in Equation (7.77) is estimated by using a finite-element expansion for the integrand. When the linear finite-element is chosen, this is equivalent to using the extended trapezoidal rule to estimate the contribution to the integral up to the grid point immediately below m_{50} and then using linear interpolation between the grid points either side of m_{50} to estimate the remainder.

7.2.4 Aerosol Transport

The modeling of between cell aerosol transport is similar in approach to the modeling for bulk gas transport described in Chapter 4. The obvious difference is that there is no transport of aerosols to surfaces, as this is calculated in the deposition and agglomeration modeling detailed in the previous sections.

Following the equations detailed in Chapter 4, the aerosol transport would be described by the general Equation (4.2). VICTORIA then assumes that there is no aerosol diffusion between cells and, with the diffusive transport to the surfaces removed, only the chemistry source and the convective fluxes between cells remain (D_x = 0.0 for all x's). Therefore, only Equations (4.4), (4.6), (4.7), and (4.16) need to be substituted into Equation (4.1). With this development, the same solution technique was chosen, where now a five-stripe matrix has developed that is first solved for the axial convection, and then for the radial ring-to-ring convection, as was done in the bulk gas transport calculation, using the mathematical software subroutine, SGTSL, described in Appendix C.

In order to model settling between control volumes, VICTORIA provides an input settling flag to modify the bulk gas velocity which determines the aerosol transport within the bulk gas. In this option, the gas velocity is modified by the aerosol settling velocity, calculated as part of the in-cell aerosol calculations. The user specifies whether this settling velocity is added or subtracted from the bulk gas velocity as a function of the orientation of their problem. Generally, the settling velocity would be subtracted from the bulk gas velocity. With this facility, it is possible to have aerosol particles of certain sizes falling in one direction which other particles of certain sizes are moving in the opposite direction. This will be useful in the modeling of certain situations in which small particles move up with the flow, and larger ones settle down against the flow.

8. REFERENCES

1.1 Reactor Safety Study - An Assessment of Accident Risks in U.S. Commercial Nuclear Power Plants, WASH-1400, NUREG-75/104, U.S. Nuclear Regulatory Commission (October 1975).

1.2 M.Rogovin, Three Mile Island: A Report to the Commission and to the Public, NUREG/CR-1250 (1979).

1.3 R.Wilson, et al., Report to the American Physical Society of the Study Group on Radionuclide Release from Severe Accidents at Nuclear Power Plants, Rev. Mod Phys 57, S1 (1985).

1.4 D.A.Powers, et al., The Chernobyl Source Term, Nuclear Safety, 28, 10 (1987).

1.5 C.J.Wheatley, CHARM: A Model for Aerosol Behavior in Time-Varying Thermal-Hydraulic Conditions, NUREG/CR-5162, SAND88-0745, Sandia National Laboratories, Albuquerque, NM (August 1988).

2.1 S.S.Dosanjh, editor, MELPROG/PWR-MOD1: A Mechanistic Code for Analysis of Reactor Core Melt Progression and Vessel Attack Under Severe Accident Conditions, NUREG/CR-5193, SAND88-1824 (1989).

2.2 Safety Code Development Group, TRAC-PF1/MOD1: An Advanced Best-Estimate Computer Program for Pressurized Water Thermal Hydraulic Analysis, NUREG/CR-3858, LA-10157-MS (1986).

2.3 D.Anderson, et al., Computational Fluid Mechanics and Heat Transfer, Hemisphere Publishing Corp., New York, NY (1984).

2.4 M. J. Bell, ORIGEN--The ORNL Isotope Generation and Depletion Code, ORNL-4628, Oak Ridge National Laboratory, Oak Ridge, TN (May 1973).

3.1 A. H. Booth, A Method of Calculating Fission Gas Diffusion from UO_2 Fuel and its Application to the X-2-f Loop Test, Chalk River Report CRDC-721 (1957).

3.2 H. Kleykamp, Nuc. Tech., 80, 412-422 (1988).

3.3 N. Oi, Z. F. Naturforschung, 20a, 1569-1571 (1965).

REFERENCES (continued)

3.4 N. Oi and J. Takagi, Z. F. Naturforschung, 19a, 1331-1332 (1964).
N. Oi and J. Takagi, Z. F. Naturforschung, 20a, 673-675 (1965).

3.5 D. Cubicciotti and J. H. Davies, Nuc. Sci. and Engr., 60, 314-319 (1976).

3.6 K. Konashi, et al., J. Nuc. Matl., 116, 86-93 (1983).

3.7 D. R. Olander, Fundamental Aspects of Nuclear Reactor Fuel Elements, TID-26711-P1, (1976).

3.8 P. S. Maiya, J. Nuc. Matl., 40, 57-65 (1971).

3.9 R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, John Wiley and Sons, Inc. (1960).

5.1 M. Silberberg, J. A. Mitchell, R. O. Meyer, and C. P. Ryder, 1986 Reassessment of the Technical Bases for Estimating Source Terms, NUREG 0956, U. S. Nuclear Regulatory Commission, Section 6.

5.2 J. L. Collins, et al., Nuc. Tech., 77, 18-31 (1987).

5.3 A. J. Grimley, III and P. J. Maudlin, Aerosol Formation in Non-Radioactive Species: Implications for the Source Term, Proc. of the International ANS/ENS Topical Meeting on Thermal Reactor Safety, San Diego, CA (February 1986).

5.4 T. Lyman, ed., Metals Handbook, V. 1, 8th Edition, Am. Soc. for Metals, Metals Park, OH (1977).

5.5 R. M. Elrick, et al., Reaction Between Some Cesium-Iodine Compounds and the Reactor Materials 304 Stainless Steel, Inconel 600 and Silver, NUREG/CR-3197 1 of 3, SAND83-0395, Sandia National Laboratories, Albuquerque, NM (1984).

5.6 W. R. Smith and R. W. Missen, Chemical Reaction Equilibrium Analysis: Theory and Algorithms, John Wiley and Sons, (1982).

5.7 W. L. Ferrar, Algebra: A Textbook of Determinants, Matrices and Algebraic Forms, 2nd. Edition, Oxford University Press (1957).

5.8 N. N. Yaneko, The Method of Fractional Steps: The Solution of Problems of Mathematical Physics in Several Variables, (M. Holt, ed.) Springer Verlag, New York, NY (1971).

REFERENCES (continued)

5.9 C. J. Wheatley, Algorithms for Calculating Chemical Equilibria and Application to Severe Reactor Accidents, SAND88-1369J (1988).

5.10 S. Gorden and B. J McBride, Computer Program for the Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations, U.S. National Aeronautics and Space Administration, NASA SP-273 (1971).

5.11 S. R. Brinkley, Jr., J. Chem. Phys., 15, 107 (1947).

5.12 S. Schechter, Trans. Amer. Math. Soc, 104, 179-189 (1962).

5.13 V. F. Urbanic and T. R. Heidrick, High Temperature Oxidation of Zircaloy-2 and Zircaloy-4 in Steam, J. of Nuc. Matl. 75, p 251-261 (1978).

5.14 G. V. Samsohov, The Oxide Handbook, 2nd Edition, Plenum (1982).

6.1 T. R. England, CINDER - A One Point Depletion and Fission Product Program, USAEC Report WAPD-TM-334, Bettis Atomic Power Laboratory and Westinghouse Electric Corp., NTIS, August 1962 (Rev. June 1964).

7.1 F. Gelbard, MAEROS User Manual, NUREG/CR-1391, SAND80-0822, Sandia National Laboratories, Albuquerque, New Mexico (1982).

7.2 H. Jordan and M. R. Kohlman, Trap-Melt2 User's Manual, NUREG/CR-4205, BMI-2621, Battelle Columbus Laboratory (1985).

7.3 R. L. Drake, A General Mathematical Survey of the Coagulation Equation, G. M. Hidy and J. R. Brock (Eds.), "International Reviews in Aerosol Physics and Chemistry," Vol. 3, Pergamon Press (1972).

7.4 C. A. Meyer et al., Thermodynamic and Transport Properties of Steam, New York: ASME (1967).

7.5 E. Cunningham, Proc. Roy. Soc., A83, 357 (1910).

7.6 C. N. Davies, Aerosol Science, Academic Press, London (1966).

7.7 H. Schlichting, Boundary Layer Theory, McGraw Hill, (1979).

REFERENCES (continued)

7.8 C. F. Colebrook, J. Institution Civil Engineers, 12, 133 (1939).

7.9 D. R. Liles et al., TRAC-PF1/MOD1 Correlations and Models, NUREG/CR-5069, LA-11208-MS R4 (1988).

7.10 K. Keller, Aerosol Behavior in Closed Containers, Kernforschungszentrum Karlsruhe, KfK 1758 (1973).

7.11 G. A. Sehmel, J. Geophys. Res., 75, 1766-1781 (1970).

7.12 P. G. Gormley and M. Kennedy, Proc. Roy. Irish Academy, A52, 163-169 (1949).

7.13 J. Tyndal, Proc. Roy. Inst. Great Britian, 6, 3 (1870).

7.14 J. R. Brock, J. Collid. Sci. 17, 768-780 (1962).

7.15 E. H. Kennard, Kinetic Theory of Gases, McGraw-Hill (1938).

7.16 S. K. Loyalka and J. H. Ferzinger, Phys. Fluids 10, 1833 (1967).

7.17 S. K. Loyalka, J. of Chem. Phys, 48, 5432 (1968).

7.18 I. N. Ivchenko and Yu I. Yalamov, Russian J. Phys. Chem., 45, 317, (1971).

7.19 L. Talbot et al., J. Fluid Mech., 101, 737-758 (1980).

7.20 B. V. Derjaguin and Yu I. Yalamov, The Theory of Thermophoresis and Diffusiophoresis of Aerosol Particles and their Experimental Testing, G. M. Hidy and J. R. Brock, (Eds.), "International Reviews in Aerosol Physics and Chemistry," Vol. 3, Pergamon Press (1972).

7.21 Pui, Romay-Novas, and Liu, Aerosol Sci. and Tech., 7, 301-315, (1987).

7.22 Cheng and Wang, J. Aerosol Sci., 6, 139-145 (1975).
Cheng and Wang, Atoms. Environ., 15, 301-306 (1981).

7.23 N. A. Fuchs, The Mechanics of Aerosols, Pergamon Press (1964).

7.24 M. Sitarski and J. H. Seinfeld, J. Colloid and Interface Sci., 61, 261-271 (1977).

REFERENCES (continued)

7.25 I. H. Dunbar, et al., Comparison of Sodium Aerosol Codes, Commission of the European Communities, EUR 9172 en, (1984).

7.26 P. G. Saffman and J. S. Turner, J. Fluid Mech., 1, 16-30 (1956).

7.27 M. A. Deliachasios and R. F. Probstein, Coagulation in Turbulent Flow-Theory and Experiment, AD-A031 297, 22 (1974).

A. ELEMENTS AND SPECIES DATA

A.1 Element and Species Selection

In the current ˎ rsion of VICTORIA, 25 elements and 167 species were chosen as being the most important for use in a typical pressurized water reactor and in related experiments. The 25 elements are tabulated in Table A.1 along with their respective atomic weights and where they are located in the species table. For example, cesium has an atomic weight of 132 and occurs in its gaseous form as species number 5 and in its condensed form as species number 8.

The 25 elements can be broken into different groups, but the most basic consists of those associated with the fuel and fissioning process itself, and those not associated with the fuel. The former group consists of Cs, Ba, I, Sr, Zr, Sn, Te, Kr, Xe, Mo, Ru, Sb, and Eu for the fission products; He for the fill gas typically used, and U and O for the fuel itself. These elements were chosen because of their potential effect on the final results. In particular the Cs, Ba, Mo, and Zr are chosen because of their high concentration in the fuel, their effect on the fuel's oxidation potential, and their chemical link with other elements; the I, Sr, and Te for their volatility and radiological consequence; Sn because of its potential chemical link with the Te; the Noble gases Kr and Xe because of their quantity and effect upon the release rates of other elements; Ru because it releases in an oxidizing environment; and Sb and Eu because they are measurable, have been reported in a number of experiments, and thus provide another method for validating results.

VICTORIA assumes that all of the above 16 elements can diffuse from within the fuel grain to the open porosity. Because of the scarcity of diffusion data similar elements are grouped together. Table A.2 lists the default solid state diffusion coefficients used for each of these groups and Figure A.1 is a graph of the resulting diffusion terms versus temperature. The values chosen are from the Oi experiments [3.3, 3.4] and represent release from a single crystal of UO_2. The use of the single crystal data allows the mechanistic physics and chemistry models in VICTORIA to calculate the release. An alternative method would be to use the integral release rates of Kelly [A.1] and remove the chemistry and physics models within the fuel as is essentially done in CORSOR [A.2] and the source term code package.

The non-fuel group originates in the cladding, control rods, grid spacers, coolant and poisons. VICTORIA uses Zircaloy-4 as its cladding material, it contains 98.4% (wt) Zr and 1.6% (wt) Sn and has an average density of 6550 kg/m^3. The code uses 80% (wt) Ag, 15% (wt) In, and 5% (wt) Cd for the control rod material, with an average density of 10170 kg/m^3. VICTORIA assumes that Inconel-600 is the grid spacer material and uses a 76% (wt) Ni, 16% (wt) Cr, and 8% (wt) Fe for the mixture, with an average density of 8420 kg/m_3. The code assumes that all steel components are stainless steel 304, and uses a mix of 71.7% (wt) Fe, 18.4% (wt) Cr, 8.7% (wt) Ni, and 1.2% (wt) Mn with an average density of 7930 kg/m_3 for this material. The potential for hydrogen and boron comes from the bulk coolant flow and poison rods respectively. The code uses the concentration of an element to conserve mass, therefore the above mass fractions and densities need to be divided by the atomic weight. VICTORIA provides standard mass distributions for the structural materials as a user convenience. The elemental concentrations are contained in data statements in the input routine FISSIN and could be changed by the user.

A.2 Gibbs Free Energy Data

The basic chemistry calculation performed in all regions is a determination of chemical equilibrium based on a minimization of the free energy of all the species in the system. VICTORIA uses a set of 167 species to define chemical equilibrium, and these are listed in Tables A.3 and A.4. Table A.3 separates the species into an elemental distribution and indicates allowable phases. Table A.4 lists the species in the order in which they are stored in VICTORIA, along with the coefficients used and appropriate references for their Gibbs free energy formulations. The species are listed in the order referred to in the input discussion of species present in the open channel. The sample problem uses species 32 - H_2 (gas); more typically species number 33 - H_2O (gas), or steam is used. Figures A.2 through A.9 show the Gibbs free energy versus temperature for some of the major groups of species. These groups were chosen on an elemental basis, that is, all species with a given element are plotted together. These graphs may give the user a better feel for the dominance of one species over another. A more complete discussion is given in the text in subsection 5.2.2 and 5.2.3 where the effect of pressure, volume, and elemental concentration are detailed.

The Gibbs free energy data for each of the 167 species used in the code is in the form

$$G = A + BT + CT^2 + DT^3$$

where G is in kilocalories/mole and T is temperature in Kelvins.

The zero of energy used in this formulation is that the enthalpy of the elements in their standard form at 298.15 K is zero. This choice of reference state allows phase changes for both the elements and more complex chemical species to be treated in an analogous manner. The values used for the coefficients A-D are stored in data statements and may be adjusted by the user.

The method for obtaining the constants for each species in the Gibbs free energy equation was as follows: The enthalpy and entropy at (typically) 300 K, and heat capacity data as a function of temperature above 300 K for each species, were taken from the source listed in the reference column. These data were then used to construct a table of the free energy as a function of temperature at 20 degree intervals from 300 K to 3000 K. The table for the condensed-phase species uses data for the solid below the melting point, and for the liquid above the melting point. Extrapolated values are used for those temperature ranges where no data was available. The table for each species is then fitted with a third-order polynomial using a least-squares fitting routine. A random comparison of the tabulated free energies taken from the references with values calculated using the polynomial for each species shows agreement to within ±5%.

Not all possible combinations of elements have been included in the species set. In most cases combinations have been excluded due to the lack of free energy data at high temperatures (>2000 K), a determination that they were of less significance for reactor-type problems, or to an oversight on our part. At this time, it is relatively difficult to change the code to include additional elements and/or species. In the near future we hope to have an automated technique to remove this inconvenience.

TABLE A.1

VICTORIA ELEMENT SET

NUMBER	ELEMENT	ATOMIC WEIGHT	SPECIES NUMBER
1	CESIUM	132.9	5,8
2	IODINE	126.9	28
3	HYDROGEN	1.0	31
4	BARIUM	137.3	21,22
5	OXYGEN	16.0	81
6	STRONTIUM	87.6	42,43
7	ZIRCONIUM	91.2	50,51
8	TIN	118.7	52,53
9	TELLURIUM	127.6	63,64
10	URANIUM	238.0	73,74
11	KRYPTON	83.8	78
12	XENON	131.3	79
13	IRON	55.8	82,83
14	MOLYBDENUM	95.9	93,94
15	SILVER	107.9	100,101
16	BORON	10.8	104,105
17	CADMIUM	112.4	120,121
18	CHROMIUM	52.0	126,127
19	INDIUM	114.8	130,131
20	MANGANESE	54.9	137,138
21	NICKEL	58.7	143,144
22	RUTHENIUM	101.1	146,147
23	ANTIMONY	121.8	151,152
24	EUROPIUM	152.0	158,159
25	HELIUM	4.0	167

TABLE A.2

SOLID-STATE DIFFUSION COEFFICIENTS

Diffusion Coefficient = $D_{\emptyset} \exp -\theta / T$ (m^2/sec)

T=Temperature (Kelvin)

GROUP	ELEMENTS	$D_{\emptyset}$	θ	REF.
Noble Gases	Xe,Kr,He	3.00e-07	31700.0	3.3
Alkali Metals	Cs	8.50e-13	3070.0	3.4
Alkaline Earths	Ba,Sr	4.50e-11	12280.0	3.4
Halogens	I	1.50e-07	29900.0	3.4
Chalogens	Te,O	6.60e-07	35230.0	3.4
Platinoids	Ru,Ni	8.61e-12	9662.0	3.4
Early Transitionals	Mo,Cr,Fe	3.90e-08	27330.0	3.4
Tetravalents	Zr,C	1.67e-05	52340.0	3.4
Trivalents	Eu,La,Al	2.20e-10	17413.0	3.4
More Volatile Others	Sb,Cd	4.50e-11	12280.0	N.A.
Less Volatile Others	Sn,In,Ag	4.50e-11	12280.0	N.A.
Uranium	U	1.00e-20	12280.0	N.A.

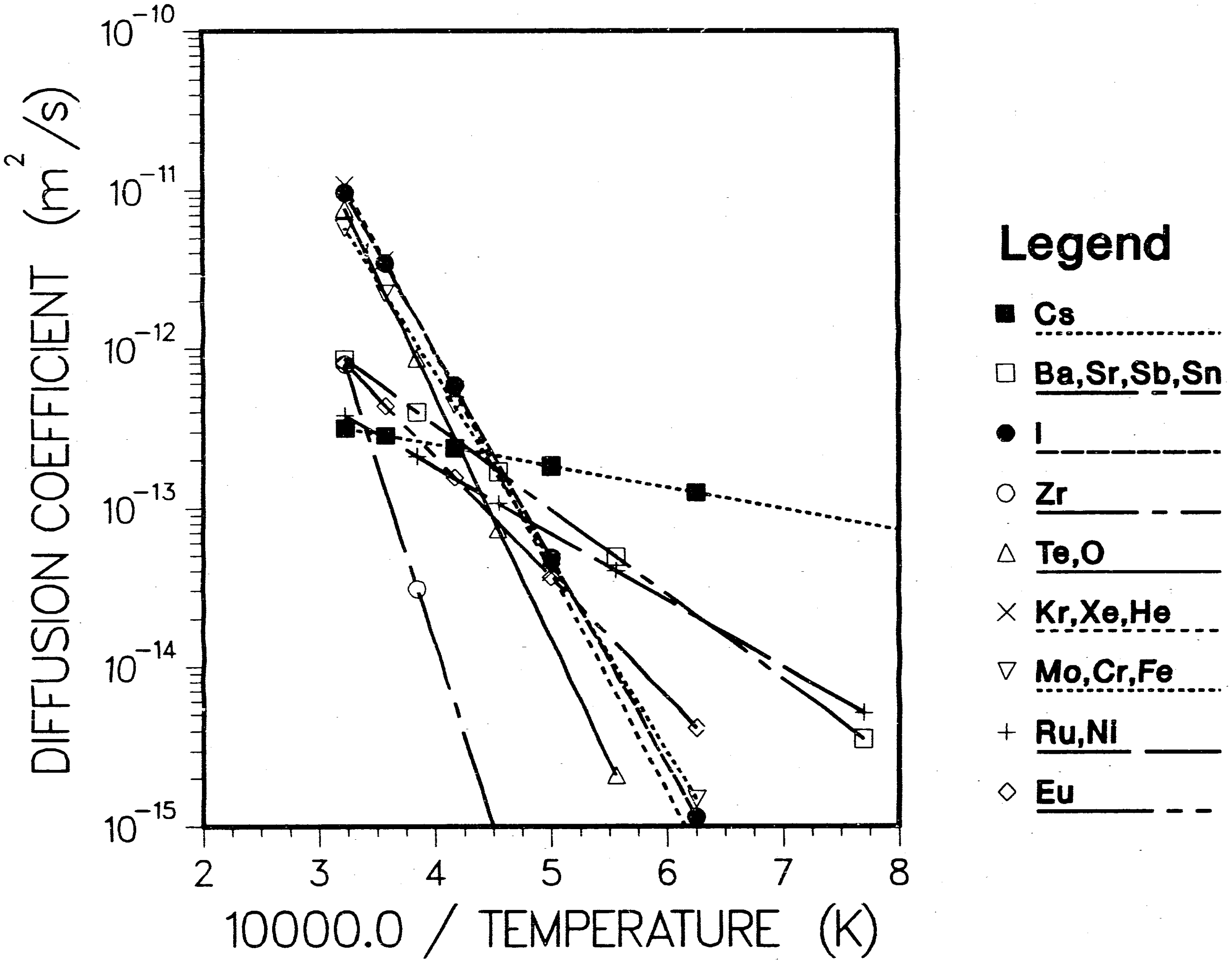


Figure A.1 Solid-state diffusion coefficients

TABLE A.3

VICTORIA (VIKI90 MOD 1) SPECIES SET

Molecule	Gaseous	Condensed
CsI	X	X
CsOH	X	X
$Cs_2(OH)_2$	X	
Cs_2O	X	X
Cs	X	X
Cs_2UO_4		X
$Cs_2U_2O_7$		X
Cs_2MoO_4		X
$CsBO_2$	X	
I	X	
I_2	X	
HI	X	
H	X	
H_2	X	
H_2O	X	
H_2Te	X	
H_2MoO_4	X	
HBO_2	X	X
H_3BO_3	X	X
BaOH	X	
$Ba(OH)_2$	X	X
BaO	X	X
Ba	X	X
BaTe	X	X
BaI	X	
BaI_2	X	X
$BaZrO_3$		X
$BaUO_4$		X
$BaMoO_4$		X
O_2	X	
O	X	

TABLE A.3 (continued)

Molecule	Gaseous	Condensed
Sr	X	X
SrO	X	X
SrOH	X	
$Sr(OH)_2$	X	X
SrI	X	
SrI_2	X	X
$SrZrO_3$		X
$SrUO_4$		X
$SrMoO_4$		X
Zr	X	X
ZrO	X	
ZrO_2		X
ZrB_2		X
ZrH	X	
Sn	X	X
SnO	X	X
SnO_2	X	X
SnI_2	X	X
SnTe	X	X
Sn_2Te_2	X	
SnH_4	X	
Te	X	X
Te_2	X	
TeO	X	
TeO_2	X	X
Te_2O_2	X	
U	X	X
UO_2	X	X
UO_3		X
U_3O_8		X
U_4O_9		X
Kr	X	

TABLE A.3 (continued)

Molecule	Gaseous	Condensed
Xe	X	
Fe	X	X
FeO	X	X
Fe_2O_3		X
Fe_3O_4		X
FeI_2	X	X
Fe_2I_4	X	
FeTe		X
$FeTe_2$		X
Mo	X	X
MoO	X	
MoO_2	X	X
MoO_3	X	X
AgTe	X	
Ag_2Te		X
B	X	X
BI	X	
BI_2	X	
BI_3	X	
BO	X	
BO_2	X	
B_2O_3	X	X
BTe	X	
Cd	X	X
CdO	X	X
CdTe	X	X
Cr	X	X
Cr_2O_3		X
CrTe		X

TABLE A.3 (continued)

Molecule	Gaseous	Condensed
In	X	X
In_2O_3		X
InTe	X	X
In_2Te	X	
InOH	X	
Mn	X	X
MnO		X
Mn_3O_4		X
MnTe		X
$MnTe_2$		X
Ni	X	X
NiO		X
Ru	X	X
RuO_2		X
RuO_3	X	
RuO_4	X	
Sb	X	X
Sb_2	X	
Sb_4	X	
Sb_2O_4		X
SbTe	X	
Sb_2Te_3		X
Eu	X	X
EuO	X	X
Eu_2O_3		X
EuTe	X	
He	X	

TABLE A.4

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

Gibbs Free Energy = $A + B*T + C*T^2 + D*T^3$ (kilocalories/mole)

T=Temperature (Kelvin)

(c) = condensed; (g) = gaseous

#	Species	A	B	C	D	Ref.
1	CsI (c)	-7.364E+01	-3.589E-02	-8.181E-06	-1.583E-10	A.3
2	CsI (g)	-2.524E+01	-7.401E-02	-4.486E-06	3.432E-10	A.3
3	CsOH (c)	-9.010E+01	-3.635E-02	-9.405E-06	7.195E-10	A.4
4	CsOH (g)	-5.792E+01	-6.517E-02	-6.490E-06	4.750E-10	A.4
5	Cs (g)	2.032E+01	-4.407E-02	-2.334E-06	1.619E-10	A.4
6	Cs_2O (g)	-1.720E+01	-8.144E-02	-6.693E-06	5.114e-10	A.4
7	$Cs_2(OH)_2$(g)	-1.579E+02	-9.141E-02	-1.342E-05	9.649E-10	A.4
8	Cs (c)	3.132E+00	-2.506E-02	-3.567E-06	2.729E-10	A.4
9	Cs_2O (c)	-7.053E+01	-3.476E-02	-1.177E-05	5.941E-10	A.4
10	Cs_2UO_4 (c)	-4.405E+02	-6.580E-02	-2.095E-05	1.444E-09	A.5
11	$Cs_2U_2O_7$ (c)	-7.134E+02	-1.411E-01	-3.760E-05	2.589E-09	A.6
12	Cs_2MoO_4 (c)	-3.506E+02	-6.815E-02	-2.469E-05	1.349E-09	A.7
13	BaOH (g)	-5.014E+01	-6.450E-02	-6.352E-06	4.498E-10	A.4
14	$BaOH_2$ (g)	-1.433E+02	-8.161E-02	-1.101E-05	7.071E-10	A.3
15	$BaOH_2$ (c)	-2.132E+02	-3.567E-02	-1.625E-05	1.244E-09	A.3
16	BaO (g)	-2.645E+01	-5.981E-02	-4.047E-06	2.781E-10	A.4
17	BaO (c)	-1.353E+02	-1.093E-02	-1.016E-05	6.290E-10	A.3
18	BaI (g)	-9.134E+00	-6.698E-02	-4.368E-06	-7.546E-10	A.4
19	BaI_2 (g)	-1.274E+02	-6.142E-02	-6.704E-06	5.129E-10	A.4
20	BaI_2 (c)	-1.317E+02	-5.118E-02	-1.302E-05	9.963E-10	A.3
21	Da (g)	4.581E+01	-4.400E-02	-1.430E-06	-6.106E-11	A.3
22	Ba (c)	5.148E+00	-2.015E-02	-4.679E-06	3.580E-10	A.3
23	BaTe (c)	-6.086E+01	-2.742E-02	-5.920E-06	3.764E-10	A.3
24	BaTe (g)	4.799E+01	-7.630E-02	1.930E-07	2.550E-11	A.3
25	$BaZrO_3$ (c)	-4.141E+02	-3.546E-02	-1.507E-05	1.068E-09	A.3
26	$BaUO_4$ (c)	-4.654E+02	-4.943E-02	-1.882E-05	1.072E-09	A.8
27	$BaMoO_4$ (c)	-3.506E+02	-4.279E-02	-1.862E-05	1.154E-09	A.3

TABLE A.1 (cont.)

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

#	Species	A	B	C	D	Ref.
28	I (g)	2.727E+01	-4.520E-02	-2.401E-06	1.800E-10	A.3
29	I_2 (g)	1.806E+01	-6.586E-02	-4.376E-06	3.295E-10	A.3
30	HI (g)	8.526E+00	-5.163E-02	-3.749E-06	2.327E-10	A.3
31	H (g)	5.387E+01	-2.944E-02	-2.396E-06	1.833E-10	A.3
32	H_2 (g)	2.313E+00	-3.373E-02	-3.538E-06	2.409E-10	A.3
33	H_2O (g)	-5.534E+01	-4.734E-02	-4.740E-06	2.648E-10	A.3
34	SrOH (g)	-4.521E+01	-6.284E-02	-6.408E-06	4.624E-10	A.4
35	$SrOH_2$ (c)	-2.190E+02	-3.231E-02	-1.818E-05	1.391E-09	A.4
36	$SrOH_2$ (g)	-1.360E+02	-7.910E-02	-1.100E-05	7.962E-10	A.4
37	SrO (c)	-1.317E+02	-2.794E-02	1.906E-07	-8.241E-10	A.3
38	SrO (g)	-2.891E-01	-5.846E-02	-3.792E-06	1.433E-10	A.3
39	SrI (g)	-4.105E+00	-6.864E-02	-4.403E-06	3.275E-10	A.4
40	SrI_2 (g)	-6.045E+01	-8.718E-02	-7.183E-06	5.493E-10	A.4
41	SrI_2 (c)	-1.231E+02	-4.930E-02	-1.302E-05	9.963E-10	A.3
42	Sr (g)	4.194E+01	-4.173E-02	-2.399E-06	1.835E-10	A.3
43	Sr (c)	4.951E+00	-1.757E-02	-3.613E-06	2.791E-10	A.3
44	$SrUO_4$ (c)	-4.692E+02	-4.602E-02	-1.864E-05	1.050E-09	A.9
45	$SrZrO_3$ (c)	-4.160E+02	-3.460E-02	-1.531E-05	1.054E-09	A.3
46	$SrMoO_4$ (c)	-3.605E+02	-3.930E-02	-1.883E-05	1.167E-09	A.3
47	ZrO (g)	1.432E+01	-5.297E-02	-6.391E-06	5.210E-10	A.3
48	ZrO_2 (g)	-6.430E+01	-6.939E-02	-6.690E-06	5.112e-10	A.4
49	ZrO_2 (c)	-2.467E+02	-3.311E-02	-7.736E-07	-6.911E-10	A.3
50	Zr (g)	1.505E+02	-4.598E-02	-2.989E-06	1.971E-10	A.3
51	Zr (c)	1.944E-01	-7.949E-03	-5.370E-06	3.999E-10	A.3
52	Sn (c)	4.120E+00	-1.840E-02	-3.280E-06	2.509E-10	A.3
53	Sn (g)	7.591E+01	-4.391E-02	-3.045E-06	2.330E-10	A.3
54	SnO (g)	7.775E+00	-5.843E-02	-4.200E-06	3.083E-10	A.3
55	SnO (c)	-6.506E+01	-1.640E-02	-6.356E-06	3.524E-10	A.3
56	SnO_2 (c)	-1.331E+02	-1.761E-02	-9.586E-06	6.367E-10	A.3
57	SnO_2 (g)	-7.587E+01	-5.457E-02	-6.356E-06	3.524E-10	A.3
58	SnI_2 (g)	3.676E+00	-8.548E-02	-7.099E-06	5.350E-10	A.3
59	SnI_2 (c)	-3.551E+00	-7.153E-02	-1.090E-05	8.339E-10	A.3
60	SnTe (g)	4.034E+01	-6.595E-02	-4.318E-06	3.305E-10	A.3
61	SnTe (c)	-1.177E+00	-3.531E-02	-8.172E-06	6.757E-10	A.3
62	SnH_4 (g)	4.229E+01	-5.578E-02	-1.040E-05	4.490E-10	A.3

TABLE A.4 (cont.)

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

#	Species	A	B	C	D	Ref.
63	Te (g)	5.227E+01	-4.551E-02	-2.463E-06	1.717E-10	A.3
64	Te (c)	6.646E+00	-1.965E-02	-4.341E-06	3.321E-10	A.3
65	Te_2 (g)	4.089E+01	-6.519E-02	-4.788E-06	3.054E-10	A.3
66	TeO (g)	2.060E+01	-6.049E-02	-4.210E-06	3.090E-10	A.3
67	TeO_2 (g)	7.195E+01	-9.438E-02	-4.210E-06	3.090E-10	A.3
68	TeO_2 (c)	-6.775E+01	-2.324E-02	-1.325E-05	9.942E-10	A.3
69	Te_2O_2 (g)	-1.972E+01	-8.455E-02	-1.006E-05	7.160E-10	A.3
70	H_2Te (g)	2.652E+01	-5.699E-02	-5.511E-06	3.108E-10	A.3
71	UO_2 (c)	-2.534E+02	-2.414E-02	-9.969E-06	6.970E-10	A.3
72	UO_2 (g)	-1.092E+02	-6.462E-02	-9.969E-06	6.970E-10	A.3
73	U (c)	2.852E+00	-1.251E-02	-6.735E-06	5.827E-10	A.3
74	U (g)	1.194E+02	-4.264E-02	-5.099E-06	3.901E-10	A.3
75	U_4O_9 (c)	-1.052E+03	-1.053E-01	-4.490E-05	3.096E-09	A.3
76	U_3O_8 (c)	-8.335E+02	-8.698E-02	-3.667E-05	2.456E-09	A.3
77	UO_3 (c)	-2.869E+02	-3.053E-02	-1.185E-05	8.065E-10	A.3
78	Kr (g)	0.000E+00	0.000E+00	0.000E+00	0.000E+00	N.A.
79	Xe (g)	0.000E+00	0.000E+00	0.000E+00	0.000E+00	N.A.
80	O_2 (g)	2.412E+00	-5.147E-02	-3.943E-06	2.630E-10	A.3
81	O (g)	6.138E+01	-4.064E-02	-2.404E-06	1.839E-10	A.3
82	Fe (c)	-6.072E-01	-2.826E-03	-7.397E-06	6.513E-10	A.3
83	Fe (g)	9.952E+01	-4.411E-02	-3.120E-06	2.387E-10	A.3
84	FeO (c)	-6.730E+01	-7.124E-03	-1.286E-05	1.235E-09	A.3
85	FeO (g)	6.279E+01	-6.072E-02	-4.223E-06	3.089E-10	A.4
86	Fe_2O_3 (c	-1.850E+02	-3.355E-02	-1.617E-05	1.170E-09	A.3
87	Fe_3O_4 (c	-2.836E+02	1.782E-03	-4.917E-05	4.748E-09	A.3
88	FeI_2 (c)	-7.378E+00	-5.784E-02	-1.302E-05	9.963E-10	A.3
89	FeI_2 (g)	2.606E+01	-8.919E-02	-7.293E-06	5.282E-10	A.3
90	Fe_2I_4 (g	1.304E+01	-1.425E-01	-1.545E-05	1.147E-09	A.3
91	FeTe (c)	-1.316E+00	-2.360E-02	-6.575E-06	4.634E-10	A.3
92	$FeTe_2$ (c	-1.160E+01	-2.913E-02	-1.077E-05	6.349E-10	A.3

TABLE A.4 (cont.)

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

#	Species	A	B	C	D	Ref.
93	Mo (c)	4.795E+00	-1.386E-02	-4.001E-07	-3.243E-10	A.3
94	Mo (g)	1.595E+02	-4.594E-02	-2.136E-06	1.312E-10	A.3
95	MoO (g)	9.528E+01	-5.982E-02	-4.209E-06	3.134E-10	A.3
96	MoO_2 (c)	-1.364E+02	-1.485E-02	-9.593E-06	4.909E-10	A.3
97	MoO_2 (g)	2.642E-01	-6.884E-02	-6.486E-06	3.099E-10	A.3
98	MoO_3 (c)	-1.627E+02	-2.964E-02	-1.525E-05	1.204E-09	A.3
99	MoO_3 (g)	-8.075E+01	-7.203E-02	-9.368E-06	7.082E-10	A.3
100	Ag (g)	6.968E+01	-4.337E-02	-2.398E-06	1.837E-10	A.10
101	Ag (c)	2.473E+00	-1.114E-02	-4.923E-06	4.390E-10	A.10
102	AgTe (g)	-2.078E-01	-6.349E-02	-9.313E-06	1.449E-09	A.3
103	Ag_2Te (c	-1.408E+01	1.956E-03	-7.775E-05	3.295E-08	A.3
104	B (c)	1.004E+00	-2.188E-03	-2.341E-06	-1.426E-11	A.3
105	B (g)	1.310E+02	-3.177E-02	-2.399E-06	1.838E-10	A.3
106	BI (g)	7.378E+01	-5.540E-02	-5.937E-06	5.618E-10	A.11
107	BI_2 (g)	5.911E+01	-7.346E-02	-9.052E-06	8.595E-10	A.11
108	BI_3 (g)	1.862E+01	-8.266E-02	-1.296E-05	1.232E-09	A.11
109	BO (g)	5.208E-01	-4.809E-02	-5.294E-06	4.781E-10	A.11
110	BO_2 (g)	-6.720E+01	-5.355E-02	-9.145E-06	8.452E-10	A.11
111	B_2O_3 (c)	-2.952E+02	-1.896E-02	-1.747E-05	1.558E-09	A.11
112	B_2O_3 (g)	-1.981E+02	-6.499E-02	-1.493E-05	1.344E-09	A.11
113	BTe (g)	9.866E+01	-5.300E-02	-8.882E-06	1.351E-09	A.10
114	$CsBO_2$ (s)	-2.274E+02	-2.030E-02	-2.385E-05	2.684E-09	A.12
115	HBO_2 (c)	-1.932E+02	-1.358E-03	-1.990E-05	2.924E-09	A.11
116	HBO_2 (g)	-1.335E+02	-5.468E-02	-1.022E-05	8.761E-10	A.11
117	H_3BO_3 (c)	-2.640E+02	-3.233E-03	-3.344E-05	4.584E-09	A.11
118	H_3BO_3 (g)	-2.611E+02	-5.819E-02	-1.937E-05	1.673E-09	A.13
119	ZrB_2 (c)	-4.429E+01	-3.264E-02	7.916E-06	-2.976E-09	A.11
120	Cd (g)	2.797E+01	-4.161E-02	-2.398E-06	1.837E-10	A.3
121	Cd (c)	3.861E+00	-1.743E-02	-3.428E-06	2.625E-10	A.3
122	CdO (c)	-5.824E+01	-1.683E-02	-6.299E-06	4.229E-10	A.3
123	CdO (g)	-5.322E-01	-5.186E-02	-4.250E-06	3.148E-10	A.3
124	CdTe (c)	-2.555E+01	-1.225E-02	-2.032E-05	4.410E-09	A.10
125	CdTe (g)	5.779E+01	-6.003E-02	-9.313E-06	1.449E-09	A.10

TABLE A.4 (cont.)

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

#	Species	A	B	C	D	Ref.
126	Cr (g)	9.679E+01	-4.385E-02	-2.191E-06	1.135E-10	A.3
127	Cr (c)	-2.210E+00	-6.753E-04	-6.681E-06	5.393E-10	A.3
128	Cr_2O_3 (c)	-2.529E+02	-4.470E-02	-4.475E-06	-1.061E-09	A.3
129	CrTe (c)	-1.752E+00	-5.781E-03	-3.563E-05	1.499E-08	A.10
130	In (g)	6.052E+01	-4.392E-02	-2.795E-06	2.140E-10	A.3
131	In (c)	3.234E+00	-1.843E-02	-3.355E-06	2.570E-10	A.3
132	In_2O_3 (c)	-2.228E+02	-1.954E-02	-2.075E-05	1.400E-09	A.3
133	InTe (c)	-1.607E+01	-2.987E-02	6.478E-06	-9.626E-09	A.10
134	InTe (g)	5.719E+01	-6.139E-01	-9.283E-06	1.443E-09	A.10
135	In_2Te (g)	5.585E+01	-8.288E-02	-9.575E-06	9.363E-10	A.13
136	InOH (g)	-2.403E+01	-6.402E-02	-8.798E-06	8.162E-10	A.13
137	Mn (g)	6.940E+01	-4.345E-02	-2.418E-06	1.852E-10	A.3
138	Mn (c)	3.212E-01	-4.218E-03	-7.824E-06	7.346E-10	A.3
139	MnO (c)	-9.883E+01	-1.674E-03	-1.339E-05	1.150E-09	A.3
140	Mn_3O_4 (c)	-3.483E+02	5.742E-03	-4.987E-05	4.954E-09	A.3
141	MnTe (c)	-2.766E+01	-1.352E-02	-1.881E-05	3.854E-09	A.10
142	$MnTe_2$ (c)	-2.852E+01	-4.003E-02	4.367E-06	-1.375E-08	A.10
143	Ni (g)	1.053E+02	-4.647E-02	-2.584E-06	1.979E-10	A.3
144	Ni (c)	-2.790E+00	-6.993E-04	-8.139E-06	7.709E-10	A.3
145	NiO (c)	-5.849E+01	-6.491E-03	-8.911E-06	5.167E-10	A.3
146	Ru (g)	1.572E+02	-4.684E-02	-2.869E-06	1.945E-10	A.3
147	Ru (c)	2.483E+00	-1.056E-02	-1.710E-06	-1.676E-10	A.3
148	RuO_2 (c)	-6.779E+01	-1.892E-02	-9.222E-06	6.071E-10	A.3
149	RuO_3 (g)	-1.302E+01	-7.151E-02	-9.267E-06	6.827E-10	A.3
150	RuO_4 (g)	-3.671E+01	-7.649E-02	-1.196E-05	8.828E-10	A.3
151	Sb (g)	6.499E+01	-4.514E-02	-2.344E-06	1.643E-10	A.3
152	Sb (c)	6.854E+00	-1.807E-02	-3.621E-06	2.773E-10	A.3
153	Sb_2 (g)	5.838E+01	-6.446E-02	-4.314E-06	3.304E-10	A.3
154	Sb_4 (g)	6.233E+01	-9.868E-02	-7.170E-06	5.491E-10	A.3
155	Sb_2O_3 (c)	-1.466E+02	-5.553E-02	-1.810E-05	1.387E-09	A.3
156	SbTe (g)	4.810E+01	-5.918E-02	-9.215E-06	1.423E-09	A.10
157	Sb_2Te_3 (c)	-8.608E+00	-8.386E-02	4.401E-05	-4.334E-08	A.10

TABLE A.4 (cont.)

GIBBS FREE ENERGY DATA FOR VICTORIA SPECIES

#	Species	A	B	C	D	Ref.
158	Eu (g)	4.386E+01	-4.746E-02	-2.225E-06	1.509E-10	A.3
159	Eu (c)	3.973E+00	-2.171E-02	-4.593E-06	3.635E-10	A.3
160	EuO (g)	-7.334E+00	-6.442E-02	-4.345E-06	3.328E-10	A.3
161	EuO (c)	-1.304E+02	-2.425E-02	-6.262E-06	4.306E-10	A.3
162	Eu_2O_3 (c)	-4.022E+02	-4.468E-02	-1.764E-05	1.283E-09	A.3
163	EuTe (g)	-6.521E+01	-6.589E-02	-9.274E-06	1.440E-09	A.10
164	ZrH (g)	1.240E+02	-5.105E-02	-5.608E-06	5.140E-10	A.11
165	Sn_2Te_2 (g)	4.274E+01	-9.441E-02	-1.365E-05	1.333E-09	A.13
166	H_2MoO_4 (g)	-2.015E+02	-8.167E-02	-2.211E-05	2.013E-09	A.11
167	He (g)	0.000E+00	0.000E+00	0.000E+00	0.000E+00	N.A.

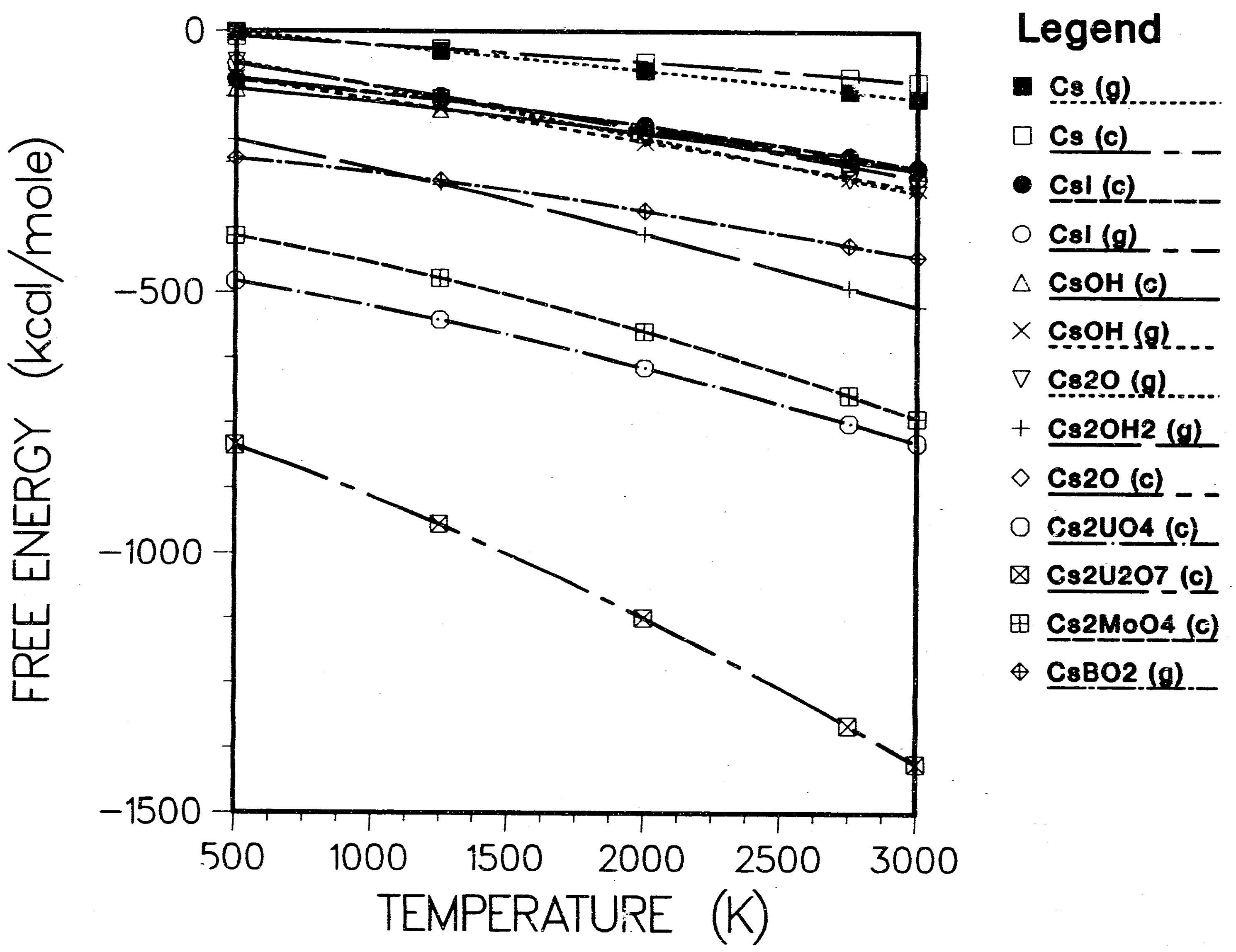


Figure A.2 Cs species Gibbs free energy

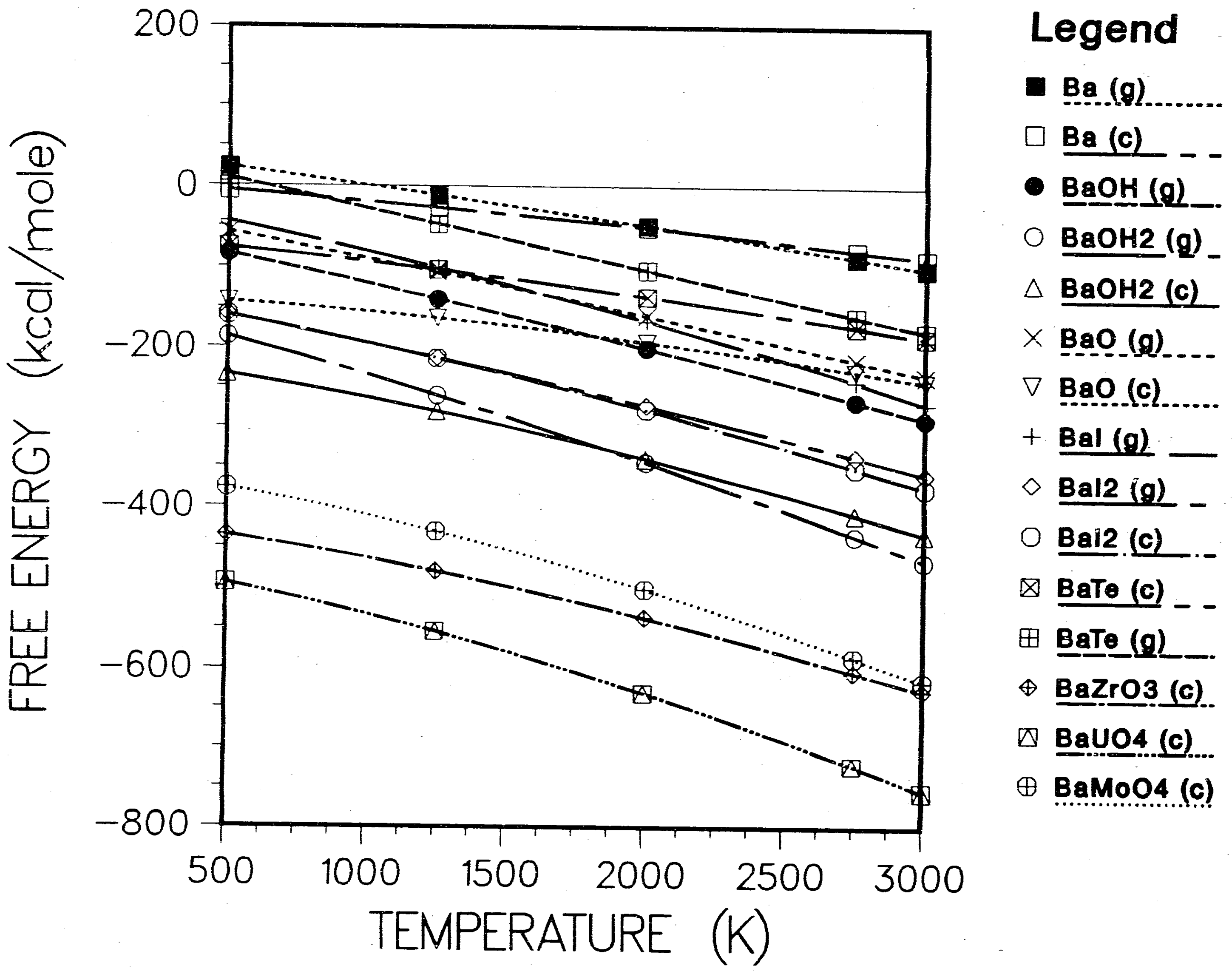


Figure A.3 Ba species Gibbs free energy

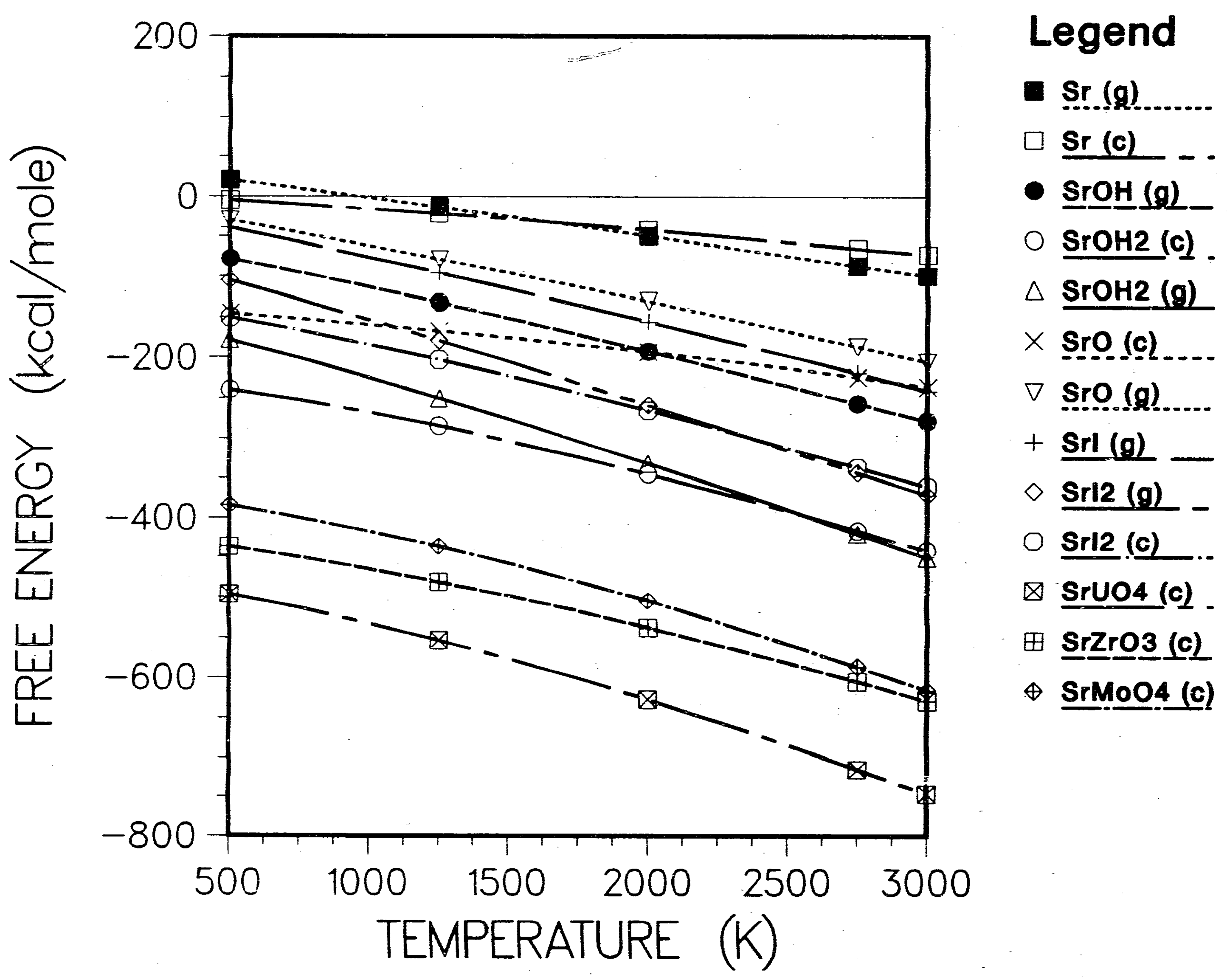


Figure A.4 Sr species Gibbs free energy

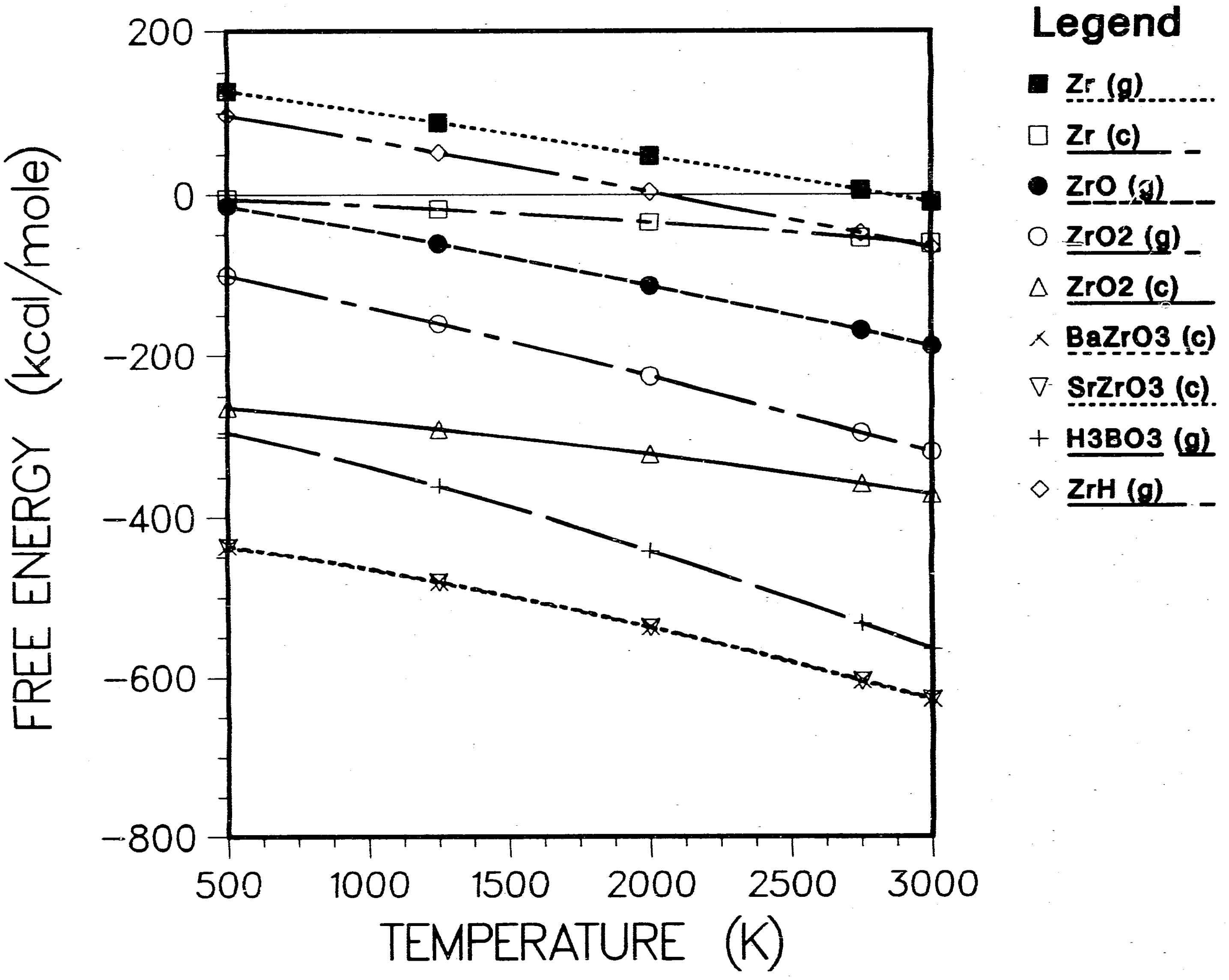


Figure A.5 Zr species Gibbs free energy

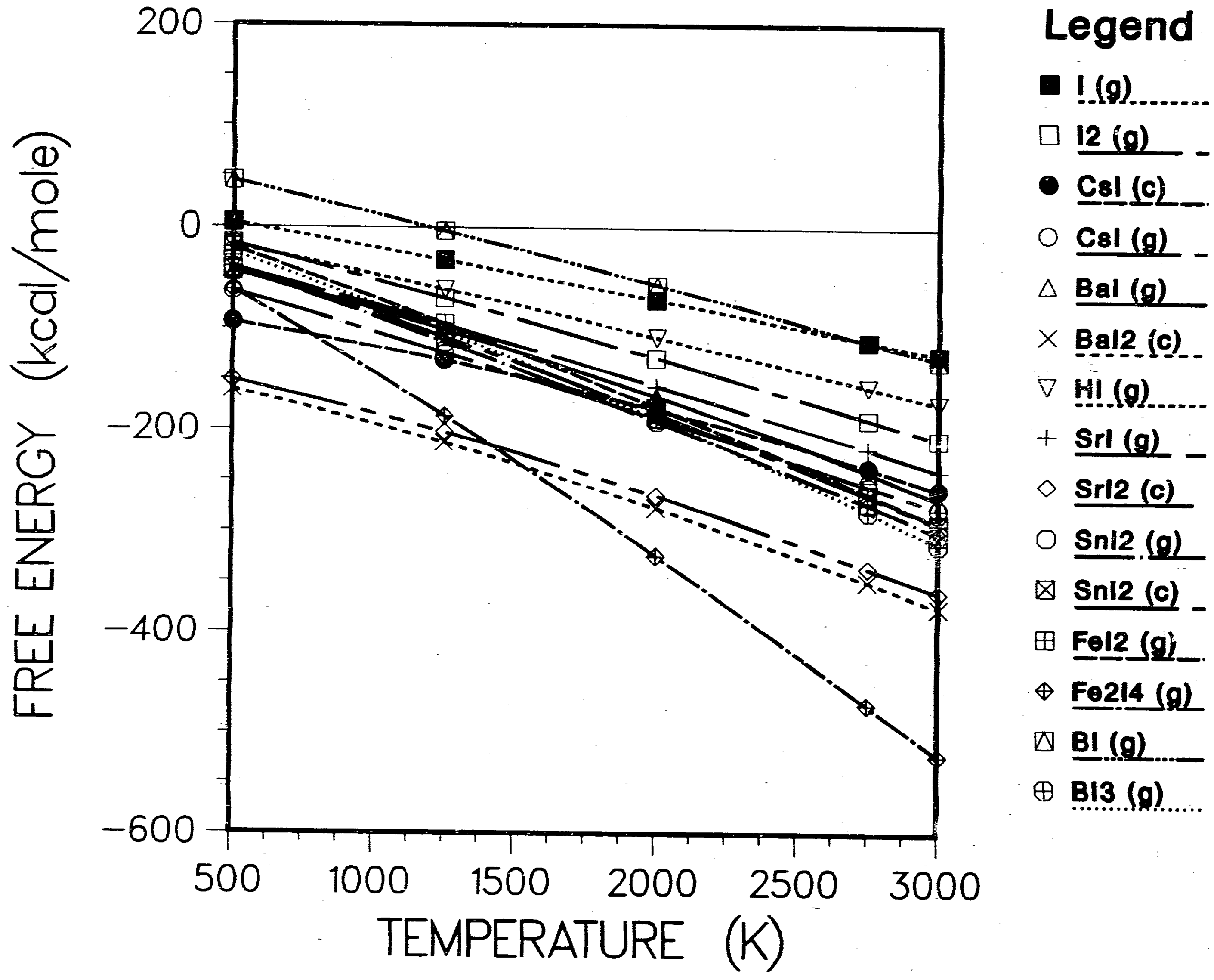


Figure A.6 I species Gibbs free energy

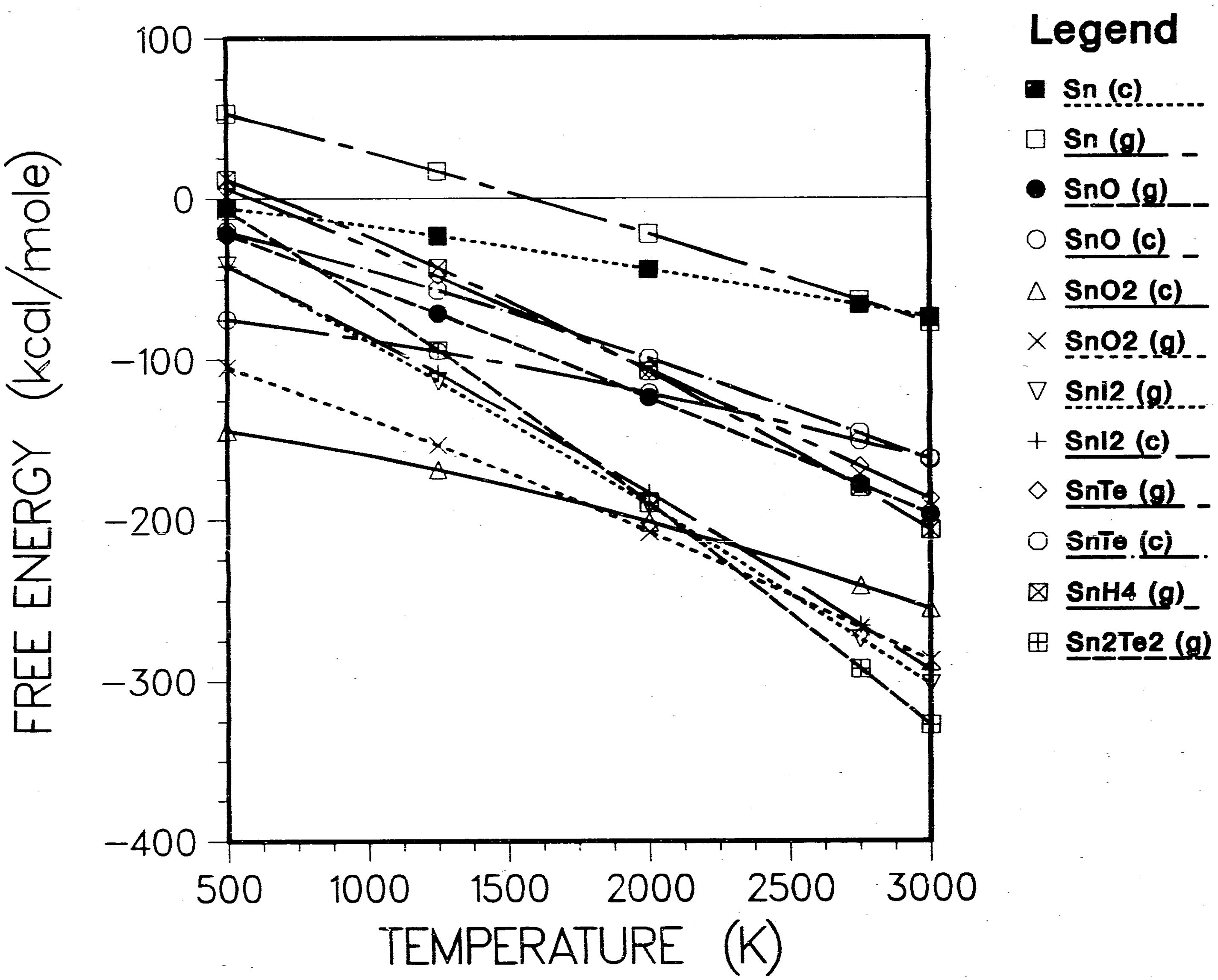


Figure A.7 Sn species Gibbs free energy

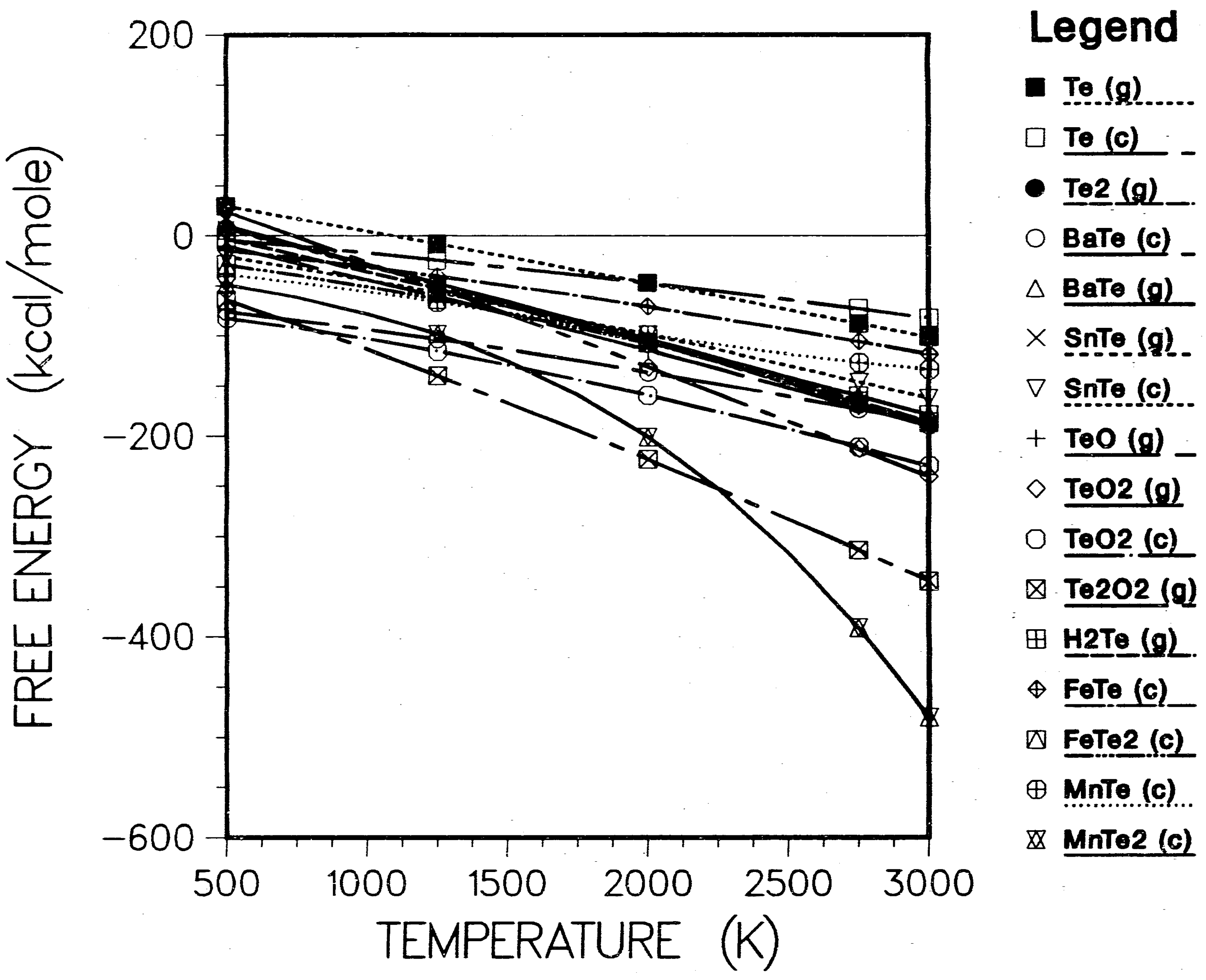


Figure A.8 Te species Gibbs free energy

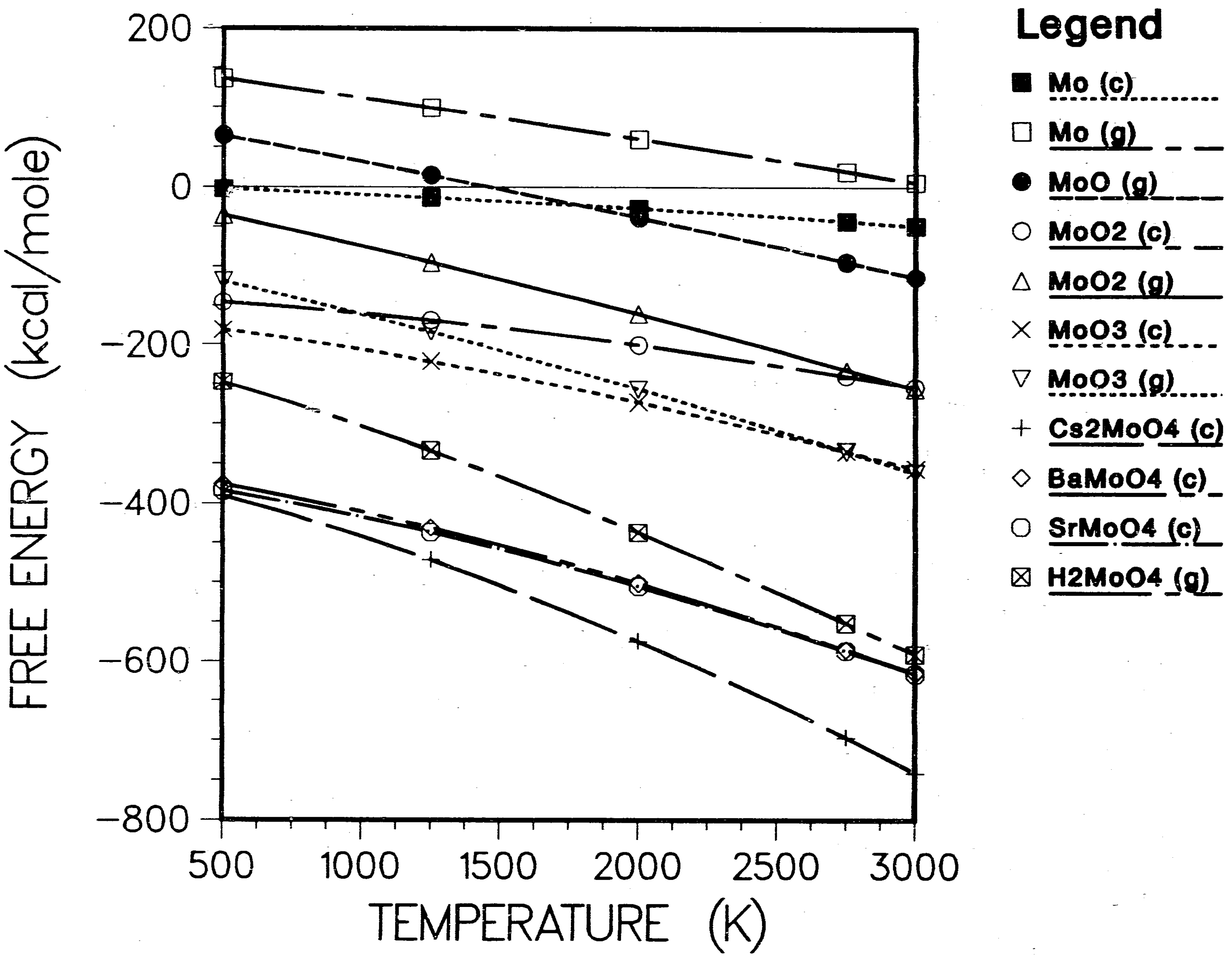


Figure A.9 Mo species Gibbs free energy

A.3 References

A.1 J. L. Kelly, et al., Nuc. Sci. and Engr. 88, 184-190 (1984).

A.2 M. R. Kuhlane, et al., CORSOR User's Manual, NUREG/CR-4173, BMI-2122 (1985).

A.3 I. Barin, et al., Thermochemical Properties of Inorganic Substances, Springer-Verlag, Berlin (1977).

A.4 D. R. Stull, et al, JANAF Thermochemical Tables, 2nd Ed., NSROS-NBS 37 (1971).

A.5 P. A. G. O'Hare and H. R. Hoekstra, J. Chem. Thermodynamics, 6, 251-258 (1974). D. W. Osborne, et al., J. Chem. Thermodynamics, 8, 361-365 (1976).

A.6 P. A. G. O'Hare, et al., J. Chem. Thermodynamics, 13, 1075-1080 (1981).

A.7 P. A. G. O'Hare and H. R. Hoekstra, J. Chem. Thermodynamics, 5, 851-856 (1973). R. Kohli and W. Lacom, Thermochimica Acta, 57, 155-160 (1982).

A.8 P. A. G. O'Hare, et al., J. Chem. Thermodynamics, 8, 845-855 (1976). P.A.G. O'Hare, et al., J. Chem. Thermodynamics, 12, 1003-1008 (1980).

A.9 H. Tagawa, et al., J. Inorg. Nuc. Chem., 41, 1729-1735 (1979).

A.10 K. C. Mills, Thermodynamic Data for Inorganic Sulphides, Selenides, Tellurides, Butterworth, London (1974).

A.11 E. H. P. Cordfunke, P. E. Potter, et al., Eur.-Contract no.ETSN-0005-NL (Feb 1988).

A.12 B. R. Bowsher, et al., AEEW-R 2341 (Mar 1988).

A.13 S. Dickinson, et al., Infra-red Spectroscopic and Mass Spectrometric Studies of High Temperature Molecules Relevant to Severe Nuclear Reactor Accidents, University of Southampton (January 1990).

B. INPUT/OUTPUT FILE DESCRIPTIONS

B.1 Input File Description

The input for the VICTORIA Version 90.1 code consists of one file, which the program expects to read from Logical Unit Number 75. The remainder of this appendix is a listing of the sample problem, with annotations to describe its content and format.

The input reader in VICTORIA uses a free-field format where either spaces or commas may be used to separate entries on a single line of input. Many input locations have default values associated with them, to activate these values the user should leave a space (not a zero) followed by a comma. VICTORIA also uses a subroutine called SKIP (explained below) that skips records beginning with a $ character. This enables the user to comment the input file, as illustrated below, so that modifications and interpretations of data are relatively easy.

Users who have used a previous version of VICTORIA should notice that the input file has been restructured in Version 90.1 to a more logical order. Thus, an input file from an earlier version must be modified before it can be used in the current version. However, the modification should be relatively straightforward because most of the parameters used in previous versions have a one-to-one correspondence with those in the current version. Furthermore, much of the input data has been eliminated and placed in data statements within the source code. Thus the input structure is greatly simplified and more readable than in earlier versions.

The following example is a simulation of part of an experiment, called ST-1, that was conducted at Sandia National Laboratories. The first several lines of the example contain comments. Any line that begins with a dollar sign ($) is treated as a comment statement rather than as data. Unless otherwise noted, the input to VICTORIA is in SI units.

Card numbers have been added to this annotated listing for ease of reference. The card numbers are not needed in an input deck and are not referenced within the code.

```
$       Input for VICTORIA Version 90.1
$
$       Job details
$       -----------
$
```

```
$  trial run for ST-1 with pure H2 at 1.5 bars; maximum
$  temperature is 2505 K; cladding failure temperature is 2125
      K;
$
```

CARD 1 + 2

The next two lines provide some information about the job that is being run that might be useful for future reference. These lines will appear near the beginning of the output.

```
$
$     computer type (uses first 8 characters):
'CRAY XMP'

$     Job description:
'Trial run for ST-1 experimental simulation'
$
$
```

CARD 3

```
$     cell input section
$     ------------------
```

The next four entries contain information about the spatial discretization that will be used. The user should refer to Figures 2.1 and 3.1 for an illustration. The first two entries define the number of vertical levels and the number of radial rings in the volume to be analyzed. The third entry determines the number of radial cells into which the fuel pellets are divided. The fourth entry specifies the number of cells into which the clad is divided for the transport calculation in the fuel region.

```
$     axial;  radial;  fuel radial;  clad radial
        7        1          2             1
$
```

CARD 4

```
$     time step control section
$     ------------------------------------
```

The next three parameters specify the initial time, the time-step size, and the final time for the calculation (in seconds). It is necessary that the time step be small enough (see subsections 2.4, 3.5, and 4.3) that the Courant condition on material transport is not violated in any portion of the code. The user can vary this time step in order to verify numerical accuracy.

```
$     initial time;  time step;  final time
          0.0            1.0        2250.
```

CARD 5

```
$       numerical methods control section
$       ---------------------------------
```

The following three entries give the maximum number of iterations allowed for any equilibrium chemistry calculation, the convergence criterion, and the maximum number of the faster cyclic Newton iterations. If convergence is not achieved after the maximum number of iterations has been reached, an error message is sent to the output file. The convergence criterion is on the relative change of each of the variables during an equilibrium chemistry calculation. The value of this criterion should be chosen so that mass will be conserved to some desired accuracy. A rule of thumb is to choose it so that exp(final time*conv. crit./time step)-1 is less than the fractional change in mass that is considered acceptable over the course of the calculation. The third entry gives an upper limit on the number of cyclic Newton iterations performed to solve the entire system of equations. When counting the cyclic iterations, VICTORIA does not count those that resulted in an order of magnitude change in a species concentration (such changes are considered excessive and are reset). Reasonable ranges of the three parameters are 100 to 300, 1.e-6 to 1.e-9, and 50 to 150, respectively.

```
$       max. chem. iter.; convergence criterion;  max. Newton
        iter.
              200                 1.e-8                 100
$
```

CARD 6

```
$       output control section
$       ---------------------
```

The frequency with which data is sent to print and graphics files is controlled by the following two entries. For a value of N, data is sent to the appropriate file every Nth time step.

```
$       print frequency;  graphics data frequency
            500                  50
```

CARD 7

```
$      internal model options selection
$      ---------------------------------
$
```

The following line of input consists of five entries which serve as flags during the running of the code. A value of one (1) indicates that a particular calculation will be performed while a value of zero (0) indicates that it will not. The identity of the calculation is given immediately above the flag. By varying these flags the user can determine the influence of various modules in VICTORIA on the results of a simulation. If the user sets the rod release flag to 0, they must also remove the appropriate fuel rod related data as will be indicated.

```
   chemistry    aerosols    convection    rod release    decay heat
       1            1            1              1             0
$
```

CARD 8 + 9

```
$      geometrical data
$      ----------------
$
```

The next two lines of input give the physical dimensions of the cylindrical grid for the problem. The grid has seven (7) vertical levels and one (1) radial ring (Card 3). The first line gives the radii (in meters) of each of the horizontal rings. The position of zero for this measurement is the vessel center line. The actual dimension given is the distance from the center of the concentric rings to the middle perimeter of each ring.

```
$      cell-centered radii (m)
          7*0.007193
```

The second line gives the height (in meters) of each vertical cell measured from the bottom of the vessel to the middle of the axial level.

```
$      location of cell axial midpoint (m)
   0.01524, 0.04572, 0.07620, 0.10668, 0.13716, 0.16764, 0.19812
```

CARD 10

```
$       bend flags
$       ----------
```

The following flags indicate whether bends occur in each of the cells. In general, these flags should be set to zero unless the corresponding cell represents a section of pipe. A value of unity indicates a single 90 degree bend in the pipe within a cell. A value of the flag between zero and unity would be used to describe a smaller angle bend. Likewise, a larger value than unity would be used to represent a larger angle or multiple bends. This value is used to determine the amount of aerosol deposition due to flow through pipe bends as described in deposition subsection 7.1.4.

```
$       bend flag by cell
         7*0.0
```

CARD 11

```
$       Settling flags
$       --------------
```

The following entry gives values for the variable that controls the application of settling of aerosols, as described in subsection 7.2.4, in each axial level and radial ring. A value of zero indicates that aerosol settling is ignored while a value of unity indicates that the settling velocity is fully applied. Any other value indicates a partial or, possibly, over application of the settling velocity. In most cases this flag should be within the range of zero to one.

```
$       settling velocity flag by cell
              7*0.0
```

CARD 12

```
$       Boundary condition flag
$       -----------------------
```

The next line specifies the type of concentration boundary condition, as detailed in subsection 4.2, for the continuity equations for each of the boundary cells. The unique ordering of the entries is up the inner boundary (r=rmin), across the top from left to right (z=zmax), across the bottom from left to right (z=zmin), and then up the outside of the outer ring (r=rmax). A value of zero indicates a zero concentration gradient (Neumann condition); a value of unity indicates a fixed value (Dirichlet condition). In this example the conditions are a zero gradient at both sides and the top and a constant value at the bottom inlet, i.e., concentration is available from the bottom.

```
$       boundary condition flag by edge
          7*0, 0, 1, 7*0
```

CARD 13 + 14

```
$       flow channel geometry
$       ---------------------
```

The following two lines give the cell-average axial flow area and hydraulic diameters for the bulk gas flow. Both are in units of meters. Note that in this case the flow area and hydraulic diameter vary with axial position.

```
$       axial flow areas (m**2)
        5*5.522e-4, 2*6.502e-4
$
$       hydraulic diameters (m)
        5*0.01818, 2*0.02877
```

CARD 15 - 17

Structure area data

The following three lines give the area, exclusive of fuel rods, of structure surfaces within each cell. These areas are used to calculated surface film diffusion to the bulk gas as well as providing a surface for aerosol deposition. In this case only an outer wall has been included in the calculation.

```
$       floor area in cell (m**2)
              7*0.0
$
$       wall area in cell (m**2)
           7*0.002756
$
$       ceiling area in cell (m**2)
               7*0.0
```

CARD 18

The following line contains identification flags for the structure in each cell. The following convention is employed: a value of zero (0) indicates a nonreactive structure; a value of one (1) indicates standard Zr-4 cladding with 98.4% Zr and 1.6% Sn; a value of two (2) indicates Inconel-600 with 76% Ni, 16% Cr and 8% Fe; and a value of three (3) indicates 304 stainless steel with 71.7% Fe, 18.4% Cr, 8.7% Ni, and 1.2% Mn. Only one of these materials may be present in a given cell. The appropriate flag will cause a concentration distribution, equivalent to the mass fractions above, to be included in the structure film. Subsection 5.3.2 discusses the use of this information. In the example, all seven cells contain nonreactive structure.

```
$       structure identification flags
                  7*0
```

Note: The next two cards are not input if the rod release model (card 7) has been turned off.

```
$       fuel data
$      ----------
```

CARD 19

The following line of data contains five parameters describing the fuel rods as detailed in Chapter 3. These are fuel pellet diameter, the radial gap thickness between the fuel and cladding, the radial clad thickness, the helium fill pressure in the fuel, and the fraction of the theoretical density (1.0-open porosity) of the fuel pellet.

```
$ pel. d. (m); gap th. (m); clad th. (m); He p. (Pa); dens.
     frac.
   8.04e-3        3.6e-4        5.7e-4       1.5e5       0.931
```

CARD 20

The next line gives the number of fuel rods that are found in each radial ring at each vertical level. This input explains the variation in flow areas with axial position noted earlier, i.e., the upper two cells have no fuel rods.

```
$      number of fuel rods per radial ring
              5*4.0, 2*0.0
```

CARDS 21 through 26

```
$
$      bulk gas initial and boundary condition input section
$     ----------------------------------------------------------
```

The initial condition input consists of a master input and a set of species inputs. The first entry is a master input that indicates how many species will initially exist in the bulk gas.

CARD 21

```
$      number of initial gas species
              1
```

CARD 22 + 23

The input for each species consists of two lines. The first contains the species number from Table A.4 for the species of interest. The second is the species initial density times its volume fraction in the bulk gas region (typically the coolant channel) for each cell in the problem. Note that this case uses hydrogen as a coolant and we chose the average bulk gas temperature to be 1790 K.

```
$      h2(gas)
             32

$      density (kg/m**3)
          7*0.020
```

CARD 24

The boundary condition input follows the pattern of the initial condition input. The first entry is the number of species at a boundary and is typically the same as the input for the initial conditions specified above.

```
$      number of species at a boundary
              1
```

CARD 25 + 26

The entry for each species again consists of two lines. The first is the species number from the Table A.5 and the second is the species fractional density at each boundary location. These locations correspond in type and order to the boundary condition flags set above (Card 12).

```
$       h2(gas)
              32

$       density (kg/m**3)
              7*0.0  0.0  0.020  7*0.0
$
```

CARDS 27 through 39

```
$       thermal/hydraulic input section
$       -------------------------------
```

The next section contains the thermal and hydraulic input that the user would either get directly from the experiment or else from a thermal hydraulic code (MELPROG or TRAC). The first variable in this section is the number of entries, corresponding to particular transient times, at which the thermal and hydraulic properties will be input. This value specifies the number of columns in the indicated array input of these properties. For each of the thermal and hydraulic properties, one line of data contains the transient time data. Followed by the array itself containing the thermal-hydraulic data for each axial cell (rows), at each time (columns), for each radial ring (starting with the innermost ring). Note that the last two rows in the fuel temperature array contain zeros because no fuel exists at the corresponding vertical level. Note that fuel temperatures are currently applied to all radial fuel volumes (two in this example) as well as the fuel-cladding gap and the cladding itself. Note also that the velocities given below correspond to the cell edge position.

CARD 27

```
$       number of time values for arrays
                   4
```

Note: The next three cards are not input if the rod release model (card 7) has been turned off.

CARD 28 + 29

```
$     time values (s) used for the fuel rod temperatures
       0.    367.    1459.   2250.
$     fuel rod temperature array (K)
$               ring 1                          ring 2
    2044.  2480.  2480.  1697.
    2065.  2505.  2505.  1714.
    2003.  2430.  2430.  1663.
    1900.  2305.  2305.  1577.
    1755.  2129.  2129.  1457.
              4*0.
              4*0.
```

CARD 30

The next item is a single value specifying the cladding failure temperature for all cells. This parameter controls the timing of the release of fission products into the coolant channel.

```
$     fuel cladding failure temperature (K)
                2125.
```

CARD 31 + 32

The following transient time entries must in agree in number but do not have to agree in value with the previous fuel rod time entries.

```
$     time values (s) for bulk gas temperatures
           0.    367.    1459.   2250.
$     bulk gas temperature array (K)
$                 ring 1                          ring 2
        1934.  2370.  2370.  1587.
        1955.  2395.  2395.  1604.
        1893.  2320.  2320.  1553.
        1790.  2195.  2195.  1467.
        1645.  2019.  2019.  1347.
        1645.  2019.  2019.  1347.
        1645.  2019.  2019.  1347.
```

CARD 33 + 34

```
$     time values (s) for structure surface temperatures
           0.    367.    1459.    2250.
```

```
$       structure surface temperature array (K)
$                  ring 1                                    ring 2
           1919.   2355.   2355.   1572.
           1940.   2380.   2380.   1689.
           1878.   2305.   2305.   1538.
           1775.   2180.   2180.   1452.
           1630.   2004.   2004.   1332.
           1630.   2004.   2004.   1332.
           1630.   2004.   2004.   1332.
```

CARD 35 + 36

```
$       time values (s) for cell pressures
             0.   367.   1459.   2250.
$       cell pressure array (Pa)
                   28*1.5e5
```

CARD 37, 38, + 39

Note: The following velocities should not cause a particle to cross a cell within a single time step and therefore violate the Courant condition.

```
$       time values (s) for fluid flow velocities
             0.   367.   1459.   2250.

$       radial velocity array (at cell right edge)
                  28*0.0

$       axial velocity array (at cell top edge)
$                 ring 1                              ring 2
                 4*0.025
                 4*0.025
                 4*0.025
                 4*0.025
                 4*0.025
                 4*0.02172
                 4*0.02172
```

Note: The next group of cards are not input if the rod release model (card 7) has been turned off.

CARD 40 through 55

```
$       initial fuel grain species data
$       -------------------------------
```

The following section contains the fuel grain initial conditions and diffusion coefficients. The data for the 16 species must be entered in the following order: U, O, Cs, Ba, I, Sr, Zr, Sn, Te, Kr, Xe, Mo, Ru, Sb, Eu, He. Three parameters are specified for each specie: initial concentration in kg moles/m**3 and two coefficients, in units of m**2/s and K, respectively, that determine diffusivity (as defined in Equation 3.3). Values of the diffusion coefficients that are left blank are replaced by default values given in Table A.2, those set to 0.0 will remain at zero. In the sample problem we do not allow the elemental uranium to diffuse, or sublimate, from the fuel grains as discussed in subsection 3.2.2. Initial species concentration can be obtained from an isotopic depletion code, such as ORIGEN [2.4].

```
$
$             U   (# 73)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
             37.01 ,         0.0         ,
$             O   (# 81)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
             74.02 ,                     ,
$             Cs  (#  8)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
            0.2791 ,                     ,
$             Ba  (# 22)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
            0.1640    ,                  ,
$             I   (# 28)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.0198        ,                 ,
$             Sr  (# 43)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.1707        ,                 ,
$             Zr  (# 51)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.4278        ,                 ,
$             Sn  (# 52)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.0000        ,                 ,
$             Te  (# 64)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.0424        ,                 ,
$             Kr  (# 78)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.5532        ,                 ,
$             Xe  (# 79)
$     conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
         0.5532        ,                 ,
$              Mo   (# 93)
$        conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
           0.3635        ,                         ,
$              Ru   (#147)
$        conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
           0.0000        ,                         ,
$              Sb   (#152)
$        conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
           0.0000        ,                         ,
$              Eu   (#159)
$        conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
           0.0000        ,                         ,
$              He   (#167)
$        conc. (kgm/m**3);  dzero (m**2/s);  theta (K)
           0.0000        ,                         ,
```

CARD 56 through 78

The remaining input pertains to the aerosol calculations performed by VICTORIA. The input must be included even if the user has chosen not to dp the aerosol calculation. The input is arranged to allow one to change the default values to user specifications for many key parameters in the module. A discussion is included in Chapter 7 of the text. The user should be aware that failure to include the aerosol calculation as part of the total VICTORIA calculation will result in the loss of all condensed species in the bulk gas zone.

The values given below represent the defaults; these same values would be assigned if they were left blank. Blank values can be specified by using a comma (,) and blank lines can be specified by using a slash (/).

CARD 56

```
$        Boundary layer data
$        -------------------
```

The boundary layer data allow the user to override the calculation of the boundary layer thickness in the model described in subsection 7.1.3. This is done by setting the boundary layer flag to be a non-zero integer and providing values for the thermal and diffusion boundary layer thicknesses. Deposition by thermophoresis is ignored when the thermal boundary layer is set to zero.

```
$
$        b.l. flag          thermal b.l.          diffusion b.l.
            0                   0.0                    0.0
```

CARD 57, 58 + 59

```
$       Definition of collocation points
$       --------------------------------*
```

The collocation points can be specified in two ways. The first is to specify the number of points and the masses of the lower and upper points. Alternatively, values for the interval and the lower and upper points can be specified. The second method is used when NCOLL is set to zero. In either case the spacing is logarithmic, i.e., the logarithms of the masses of particles are spaced uniformly.

The element number determines the halfwidth of the collocation points as well as the integration form of g(x) as discussed in subsection 7.2.1. The user may specify 3 values for this entry. Element = 2 is the default, it corresponds to a piecewise linear finite element, and should be used in normal circumstances. Element = 3, a piecewise cubic formulation, may be used when high accuracy (corresponding to a large number of collocation points) is required. Element = 1, a piecewise constant formulation, reduces the number of points and has resulted in a less accurate, but computationally faster, result [1.5].

```
$
$       ncoll     spacing
          0        10.0
$
$       mlower    mupper
        4.e-21    4.e-9
$
$       element number
           2
```

CARD 60 + 61

```
$       Tolerance specification
$       -----------------------
```

EPSA is a relative error parameter. It is used to set tolerances for the integration of the differential equations and set a relative tolerance for the location of zeros of functions and the values for integrals. A reasonable range for EPSA is 10^{-3} to 10^{-8}. The model may not work satisfactorily when EPSA is too large or too small. The default values for ETA and ZETA should always be used. Up to MAXCLS evaluations of a function are made in an attempt to locate its zero the desired tolerance. This value may need to be increased when EPSA is small. MAXTRY determines how many integration steps are attempted by DEBDF to integrate the equations to the next specified time. This value may need to be increased when EPSA is small. ETA defines an absolute tolerance on the value of a function whose zero is sought. ZETA determines when bisection is used in favor of inverse quadratic interpolation to locate zeros of a function.

```
$
$      epsa          maxcls          maxtry
      1.e-6           60               10
$
$      eta           zeta
      1.e-9          0.5
```

CARD 62, 63, and 64

```
$      Aerosol physics data
$      -------------------*
```

The aerosol physics data comprise physical properties of the aerosol material and constants that appear in the models for agglomeration and deposition.

```
$
$       χc (7.1.5)             χd (7.1.2)              ϵs (7.1.5)
$      collision shape        dynamic shape             sticking
$          factor                 factor               efficiency
           1.0                    1.0                     1.0
$
$      Cunningham slip correction factors (7.1.2)
$        ka                 kq                  kb
        1.25               0.41                0.88
$
$      Thermophoretic deposition coefficients (7.1.4)
$       cs                 cm                  ct
       1.17               1.14                1.28
```

CARD 65

```
$
$      Single/multi component aerosol
$      ------------------------------
```

The following entry allows the user to choose the level of composition detail across a collocation point. A discussion of the subject is included in subsection 7.2.1, the value chosen also effects the input to cards 69 through 71.

```
$     Input the word 'single' or 'multi'
'single'
$
```

CARD 66 through 73

```
$
$     Aerosol source input
$     --------------------
```

The following entries provide the user with a method of inputing a time dependent aerosol source at a particular cell location. A discussion of the implementation is in subsection 7.1.5. The values shown are typical, the sample problem does not choose this option.

These first three cards determine if there is a source and the times during the transient that it is active. An input of 0, the value used in the sample problem, for this first card forces the code to go to card 74.

```
$     Number of aerosol sources
$     This value is typically 1 for single component
$          and greater than 1 for multi component
      1
$
$     Number of time entries for the aerosol source table
      6
$
$     transient time entries (s)
      200., 201., 201.1, 850.0, 850.1, 1000.0
$
```

CARD 69, 70, + 71

```
      Aerosol source species and location
      -----------------------------------
```

The following cards determine the actual species used for the aerosol source as well as in which cell the source will be placed. Because of the previous option concerning single and multi component sources these three cards are a function of that input (card 65). Species numbers correspond to those used in Table A.4.

```
$      -----------------------------------
$      Single component chosen for Card 65
$            Aerosol source input
$      -----------------------------------
$
$      Number of species in this source aerosol  --  Card 69
       2
$
$      Entries to determine fraction of source that is a
$      particular species.  The sum of the fractions = 1.0.
$      Species Number           Fraction         --  Card 70
          1                         0.5
          3                         0.5

       Using half CsI and half CsOH for this example
$
$      Determine fluid cell location
$      Axial level       Radial Ring             --  Card 71
          1                   1
$
$      ----------------------------------
$      Multi-component chosen for Card 65
$            Aerosol source input
$      ----------------------------------
$
$      Determine species and fluid cell location
$      Species Number     Axial Level      Radial Ring - Card 69
$         1                   1                 1      -----------
       -$
$      No Card 70 or 71 are required for multi-component input
$
```

CARD 72 + 73

The following entries determine the aerosol source rate as well as the standard deviation (σ) and mass mean radius (r_{50}) of the source.

```
$      Source rates (kg/s) corresponding to times (card 68)
$              the following values have been reduced
$
       2.0e-14, 2.0e-14,    3.082e-17, 3.082e-17,    0.0, 0.0
$
$      Sigma         Radius
        2.0          0.5e-6
$
```

CARD 74 through 78

The following entries determine a vapor species input to a fluid cell volume vs time. With this system additional vapor species can be added to the bulk gas flow.

```
$      Number of vapor species to be added
$      An input of 0 forces the code to a normal end of data as
$      was done in the sample problem.
       1
$

                         CARD 75 + 76

               The following entries yields the time (s) table
               for the vapor source

$      Number of entries in the vapor source table
       6
$
$      Transient time (s)
       200., 201.,   201.1, 850.0,   850.1, 1000.0
$

                         CARD 77 + 78

               The following cards are input according to the
               entry on card 74 (1 in this case).  The first card
               determines the vapor species source and which bulk
               gas cell it will be injected into.  The second
               card determines the actual source rate for the
               species.

$      Vapor Species      Axial Level        Radial Ring
           2                   1                   1
$
$      Vapor Species Source Rate (6 entries 1 for each time)
$                       (kg/s)
$                 Values have been reduced
$
       2.0e-14, 2.0e-14,    3.082e-17, 3.082e-17,    0.0, 0.0
$
$$
       --------------------------*
$      End of the Input Data File
$      --------------------------*
```

B.2 OUTPUT FILE DESCRIPTION

Except for the echoing of the input data at the start of each run, all the printed and graphical output from VICTORIA is handled through a single routine--VEDIT. The user specifies in the input data how often the printed and the graphical output is to occur. These frequencies are given as the number of time steps between outputs.

B.2.1 Input Edit

The VICTORIA code provides an echo to the printer of each card as it is read. In addition, after reading the entire input deck, the code edits its interpretation of the data. Figure B.1 is an example of some of these edits occurring after the input has been read. With this combination of edits, the developers feel that most user input errors can easily be found.

B.2.2 Transient Mass Balance

In addition to stored information for each species, it is often useful to have the current mass balance information for each element. At the beginning of a full edit a routine, MASBAL, calculates the mass of each element in each cell in the fuel grains, fuel pores, fuel-cladding gap, bulk gas, aerosol, and structure film. In addition, MASBAL calculates for the entire problem the net amount of gas lost, the amount of aerosol lost, and the total amount of aerosol source for each element. MASBAL then calculates the overall mass balance for each element.

B.2.3 Transient Printed Output

The printed output is controlled by routine VPRINT.

Between full prints, subroutine VIKI edits the time step control information. This allows the user to monitor the stability of the calculation and to know which module may be having problems. The value edited for each module is the central processor unit (CPU) time necessary for the calculation. In particular, the columns are: Step Number, Total Time, Time Step, Fuel Release CPU Time, Species Chemistry CPU time, Bulk Gas Transport CPU Time, the In-cell Aerosol Behavior CPU Time, The Decay Power CPU Time, and the Between Cell Aerosol Transport CPU Time.

On a full edit, VPRINT writes the following information shown on figures B.2-B.4, for each cell:

* the mass density for each species in the fuel grains, fuel pores, fuel-cladding gap, cladding, cladding film, bulk gas, aerosol, and structure film. The fuel grains, fuel pore, and cladding densities are written for each fuel or cladding node. The total mass of each species in the cell is also written out.

- the mass of each element. Similar information is written out for each element, having been calculated as part of the mass balance routine. Only the total mass of each element in the fuel grains, fuel pores, and cladding is written for each cell.

- the aerosol mass density for each aerosol species. The mass at each collocation point is written together with other information relating to aerosol species.

- the aerosol mass median equivalent diameter, geometric standard deviation, geometric mean equivalent diameter, and mass deposition rate.

- the overall mass balance for each element calculated in routine MASBAL. Except for this overall mass balance no output is written for species or elements which are not present.

Routine VPRINT does not print output directly. Instead a separate routine, SCREEN, controls the print formatting. The VPRINT routine calls SCREEN, which indicates where on an imaginary screen a number or string of characters will be written. The screen is much larger than can possibly be printed, and is currently set at 500 characters wide by 256 lines long. When all the values have been written to the screen, the screen is printed in sections of suitable sizes, currently limited to 132 characters in width. This produces clear, readable output with a minimum of wasted paper. Routine SCREEN has several entry point locations to perform these tasks:

SCRFP - used to indicate that a real number will be printed at a particular location. Optionally a dot can be printed if the input value is zero.

SCRINT- similar to SCRFP and used to indicate that an integer will be printed.

SCRCHR- used to indicate that a character string will be printed at a particular location.

PRELUD- used to indicate those parts of the imaginary screen that are to be reprinted if the screen has to be split in two or more sections for printing.

SCREEN- used to indicate that the imaginary screen will be printed. After printing the screen is cleared.

```
Trial run for ST-1 experimental simulation                                    VICTORIA 90.1 July 4
                                                                              10:25:47    4 Jul 1990     page     7
     ******  MACHINE AND JOB DETAILS  ******

     VICTORIA 90.1 July 4

    Calculations Performed on CRAY XMP

    Trial run for ST-1 experimental simulation

     ******  Cell Data  ******

    Number of axial nodes  =    7
    Number of radial nodes =    1
    Number of fuel nodes   =    2
    Number of clad nodes   =    1

     ******  Time Step Control Data  ******

    Problem start time =      0.0 seconds
    Time step length   =      1.0 seconds
    Problem end time   =   2250.0 seconds

     ******  Numerical Methods Control Data  ******

    igsmax =      200
    epsc   =   0.1000e-07
    igscyc =      100

     ******  Output Control Data  ******

    Skip  500 time steps for printed output
    Skip   50 time steps for graphical output

     ******  Option Control Data  ******

    ichem =    1 iaero =    1 iconvc =    1 irodr =    1 idecay =    0

    ******  Cell Geometrical Data  ******

    ******  Cell Radii ******
    Radial node number =    1
    Radii =
     0.7193e-02  0.7193e-02  0.7193e-02  0.7193e-02  0.7193e-02  0.7193e-02  0.7193e-02

    ******  Ce'l Heights ******
    Heights =
     0.1524e-01  0.4572e-01  0.7620e-01  0.1067e+00  0.1372e+00  0.1676e+00  0.1981e+00
```

Figure B.1 Sample Problem Input

```
                                                                                    VICTORIA 90.1 July 4
      Trial run for ST-1 experimental simulation                                    08:01:36   5 Jul 1990    page    66


           species densities (kg/m**3) at i =   1 j =   1 time = 2.250e+03 s

   species     grains    1  pores    1  grains    2  pores    2  gap          clad    1   film         bulk gas     aerosol      structure

   CsIC        6.978e-39    9.313e-15   1.049e-38    6.561e-15            .   1.633e-17            .                4.652e-25    4.097e-22
   CsIG        3.438e-42    4.496e-16   5.170e-42    6.030e-16   2.774e-15    2.774e-15   4.772e-16    4.502e-16                 4.468e-16
   CsG         2.935e-05    1.984e-04   1.986e-05    1.334e-04   7.237e-05    7.236e-05   1.823e-05    1.740e-05                 1.740e-05
   Cs2OG       2.873e-21            .   1.364e-21            .           .            .           .            .                         .
   CsC         1.301e-03    1.121e-03   8.959e-04    7.710e-04           .    5.444e-08           .                 1.269e-15    1.051e-12
   Cs2OC       4.723e-19            .   2.243e-19            .           .            .           .                         .            .
   Cs2UO4C     4.727e-20            .   2.246e-20            .           .            .           .                         .            .
   Cs2U2O7C    2.056e-21            .   9.772e-22            .           .            .           .                         .            .
   Cs2MoO4C    2.965e-28            .   1.410e-28            .           .            .           .                         .            .


           species densities (kg/m**3) at i =   1 j =   1 time = 2.250e+03 s

   species  total  (kg)

   CsIC      3.129e-21
   CsIG      1.385e-20
   CsG       6.064e-10
   Cs2OG     1.004e-26
   CsC       6.112e-09
   Cs2OC     1.650e-24
   Cs2UO4C   1.652e-25
   Cs2U2O7C  7.185e-27
   Cs2MoO4C  1.037e-33


           element masses (kg) at i =   1 j =   1 time = 2.250e+03 s

   species     grains       pores        gap          clad         film         bulk gas     aerosol      structure    total

   Cs          5.874e-09    4.305e-10    8.383e-11    1.475e-10    3.630e-11    1.229e-10    8.960e-21    2.397e-11    6.719e-09
   I           2.173e-23    2.268e-21    2.777e-21    4.898e-21    1.180e-21    3.994e-21    1.605e-30    7.794e-22    1.592e-20
   H                   .    8.539e-09    2.316e-08    4.072e-08    3.983e-08    1.412e-07            .    2.756e-08    2.810e-07
   Ba          9.525e-05    4.722e-06    4.933e-11    1.319e-10    2.770e-11    9.344e-11    1.595e-17    1.824e-11    9.997e-05
   O           6.825e-03            .            .            .            .            .            .            .    6.825e-03
```

Figure B.2 Sample Problem Cs Species Densities and Masses

```
Trial run for ST-1 experimental simulation                                        VICTORIA 9Ø.1 July 4
                                                                                  Ø8:Ø1:36   5 Jul 199Ø    page    67


        aerosol densities (kg/m**3) at i =  1 j =  1 time = 2.25Øe+Ø3 s

species     collocation points(kg)
              4.Øe-21     4.Øe-2Ø     4.Øe-19     4.Øe-18     4.Øe-17     4.Øe-16     4.Øe-15     4.Øe-14     4.Øe-13     4.Øe-12

CsIC        4.652e-25   1.944e-29   6.573e-35        .           .           .           .           .           .           .
CsC         1.269e-15   5.3ØØe-2Ø   1.792e-25        .           .           .           .           .           .           .
BaI2C           .           .           .            .           .           .           .           .           .           .

total       3.94Øe-12   1.646e-16   5.567e-22        .           .           .           .           .           .           .


        aerosol densities (kg/m**3) at i =  1 j =  1 time = 2.25Øe+Ø3 s

species                                            lost mass (kg)                deposited mass (kg)
             4.Øe-11     4.Øe-1Ø     4.Øe-Ø9    upper bin   lower bin   ceiling   wall        floor       total

CsIC            .           .           .           .           .           .     6.799e-15      .      6.799e-15
CsC             .           .           .           .           .           .     1.225e-14      .      1.225e-14

total           .           .           .           .           .           .     1.981e-11      .      1.981e-11

mass median diameter                      1.969e-Ø6     (m)
geometric standard deviation              1.ØØ5e+ØØ
geometric mean diameter                   1.184e-Ø8     (m)
mass deposition rate                      1.Ø8Øe-16     (kg/s)
```

Figure B.3 Sample Problem Cs Aerosol Densities

```
                                                                                   VICTORIA 90.1 July 4
     Trial run for ST-1 experimental simulation                                     08:01:37   5 Jul 1990    page    68

          element masses (kg) in all cells at time = 2.250e+03 s

species   grains     pores      gap        clad       film       bulk gas   aerosol    structure  total      lost gas
Cs        4.150e-06  3.059e-07  2.292e-09  4.029e-09  1.426e-09  1.146e-08  1.784e-17  1.435e-09  4.476e-06  1.066e-03
I         3.657e-22  1.780e-16  8.331e-20  1.469e-19  5.174e-20  4.219e-19  3.748e-26  5.321e-20  1.788e-16  7.240e-05
H             .      4.266e-08  1.157e-07  2.034e-07  1.989e-07  1.426e-06      .      1.915e-07  2.179e-06 -3.621e-07
Ba        4.862e-04  7.084e-05  1.924e-10  8.274e-10  1.557e-10  1.019e-09  1.088e-12  8.669e-09  5.570e-04  9.110e-05
O         3.412e-02      .          .          .          .          .          .          .      3.412e-02      .
Sr        3.546e-04  6.475e-08  1.381e-10  2.463e-10  1.113e-10  7.323e-10  9.715e-17  1.019e-10  3.547e-04  7.617e-05
Zr        1.123e-03  1.627e-06  4.351e-20  1.610e-13  6.419e-03  7.218e-20  1.104e-19  8.518e-12  7.543e-03  5.073e-12
Sn        2.795e-09  2.073e-10  2.028e-16  3.453e-14  1.044e-04  1.023e-15  1.363e-18  9.884e-15  1.044e-04  6.644e-09
Te        3.987e-07  6.487e-05  1.546e-12  3.997e-10  1.181e-12  5.735e-12  1.009e-12  7.911e-09  6.528e-05  9.000e-05
U         2.538e-01      .          .          .          .          .          .          .      2.538e-01      .
Kr        3.017e-15  5.591e-17  7.231e-18  1.271e-17  4.539e-18  3.661e-17      .      4.579e-18  3.139e-15  1.336e-03
Xe        5.337e-15  9.935e-17  1.277e-17  2.244e-17  8.012e-18  6.463e-17      .      8.083e-18  5.553e-15  2.093e-03
Fe            .          .          .          .          .          .          .          .          .          .
Mo        9.350e-04  6.930e-05  3.055e-19  1.065e-07  1.222e-19  3.994e-19  7.097e-19  1.754e-10  1.004e-03  3.223e-11
Ag            .          .          .          .          .          .          .          .          .          .
B             .          .          .          .          .          .          .          .          .          .
Cd            .          .          .          .          .          .          .          .          .          .
Cr            .          .          .          .          .          .          .          .          .          .
In            .          .          .          .          .          .          .          .          .          .
Mn            .          .          .          .          .          .          .          .          .          .
Ni            .          .          .          .          .          .          .          .          .          .
Ru            .          .          .          .          .          .          .          .          .          .
Sb            .          .          .          .          .          .          .          .          .          .
Eu            .          .          .          .          .          .          .          .          .          .
He            .          .          .          .          .          .          .          .          .          .


        element masses (kg) in all cells at time = 2.250e+03 s

species  lost aero   aero sce    initial    balance     bal/init
Cs       4.153e-09      .       1.071e-03  -1.405e-10   0.000000
I        1.291e-09      .       7.240e-05   4.438e-12   0.000000
H            .          .       1.816e-06   4.744e-10   0.000261
Ba       6.554e-07      .       6.488e-04   2.712e-10   0.000000
O            .          .       3.412e-02   4.211e-09   0.000000
Sr       3.175e-10      .       4.309e-04   1.972e-09   0.000005
Zr       1.428e-11      .       7.319e-02  -6.564e-02   0.896938
Sn       4.904e-12      .       1.172e-03  -1.067e-03   0.910922
Te       6.091e-07      .       1.559e-04   2.639e-09   0.000017
U            .          .       2.538e-01   3.494e-12   0.000000
Kr           .          .       1.336e-03   8.327e-17   0.000000
Xe           .          .       2.093e-03  -1.246e-14   0.000000
Fe           .          .           .          .
Mo       2.323e-10      .       1.004e-03   5.894e-11   0.000000
```

Figure B.4 Sample Problem Total Elemental Masses

B.3 Graphics File Description

VICTORIA can also produce an unformatted dump for subsequent graphical output. This dump was designed for use with the TRAGIC graphics package, a local WINFRITH code package. The data written to the graphics file is similar to that which is printed out, but with some additional variables.

There is a nine item array written to the graphics file for each variable that tells the code the limits of the variables. The array pattern for each variable is:

1.	SPEC -	species number
2.	ELEM -	element number
3.	COMP -	vapor species number
4.	AERO -	aerosol species number
5.	SECT -	aerosol section
6.	NODE -	fuel rod node
7.	FNODE -	fuel pellet node
8.	J -	axial node
9.	I -	radial node

NODE and FNODE are both included because some variables have values for each node in the fuel and the clad, while others have values only for each fuel node.

Individual values of the SPEC dimension are given names (for example CsOHg, CsOHc) as are values of the ELEM dimension (for example Cs). COMP and AERO values are given names which consist of the corresponding species name with the final g or c replaced by V or A respectively, except that the Boron aerosol is called BAE to avoid conflict with the element Ba (Barium).

Individual values of the NODE dimension are also given values (for example FUEL1, FUEL2, GAP, CLAD). A list of all variables on the graphics file is given in Table B.1. Several examples, using the standard problem input and VICTORIA 90.1 July, are graphically displayed in figures B.5 and B.6.

TABLE B.1 Graphics Output Variables

Variable	Description
VOLUME	Volume (m^3)
STRCAREA	Structure surface area (m^2)
NODEVOL	Fuel node volume (m^3)
ALPHA	Fuel porosity
GRAIN	density in grains (kg/m^3)
PORE	Density in pores (kg/m^3)
GASDEN	Gas density in bulk gas (kg/m^3)
STRUCT	Structure density (kg/m^3)
GASLOST	Gas lost through boundary (kg)
XMASS	Mass at collocation points (kg)
from the AEROSOL calculation (cell dependent)	
AERODEN	Aerosol density (kg/m^3)
DEPOST	Total deposition (kg)
CEILING	Ceiling deposition (kg)
WALL	Wall deposition (kg)
FLOOR	Floor deposition (kg)
UPPER	Upper bin mass (kg)
LOWER	Lower bin mass (kg)
MMD	Mass median diameter (m)
G50	Geometric mean diameter (m)
DEPRAT	Mass deposition rate (kg/s)
GEOSD	Geometric standard deviation
AEROLOST	Aerosol lost through boundary (kg)
SOURCS	Aerosol source (kg)
from the Chemistry calculation (C.. is cell dependent)	
CLGRAIN	Mass in grains (kg)
CLPORE	Mass in pores (kg)
CLGAP	Mass in gap (kg)
CLCLAD	Mass in clad (kg)
CLFILM	Mass in film (kg)
CLGAS	Mass in bulk gas (kg)
CLAERO	Mass in aerosols (kg)
CLSTRUCT	Mass on structures
ELGRAIN	Total mass in grains (kg)
ELPORE	Total mass in pores (kg)
ELGAP	Total mass in gap (kg)
ELCLAD	Total mass in clad (kg)
ELFILM	Total mass in film (kg)
ELGAS	Totla mass in bulk gas (kg)
ELAERO	Total mass in aerosols (kg)
ELSTRUCT	Total mass on structures (kg)
ELOSTGAS	Total mass lost as gas (kg)
ELOSTAER	Total mass lost as aerosol (kg)
ELAERSRC	Total mass source of aerosol (kg)
EINITIAL	Total initial mass (kg)
ELBALANC	Mass balance (kg)
ELFRBLNC	Mass balance/total initial mass

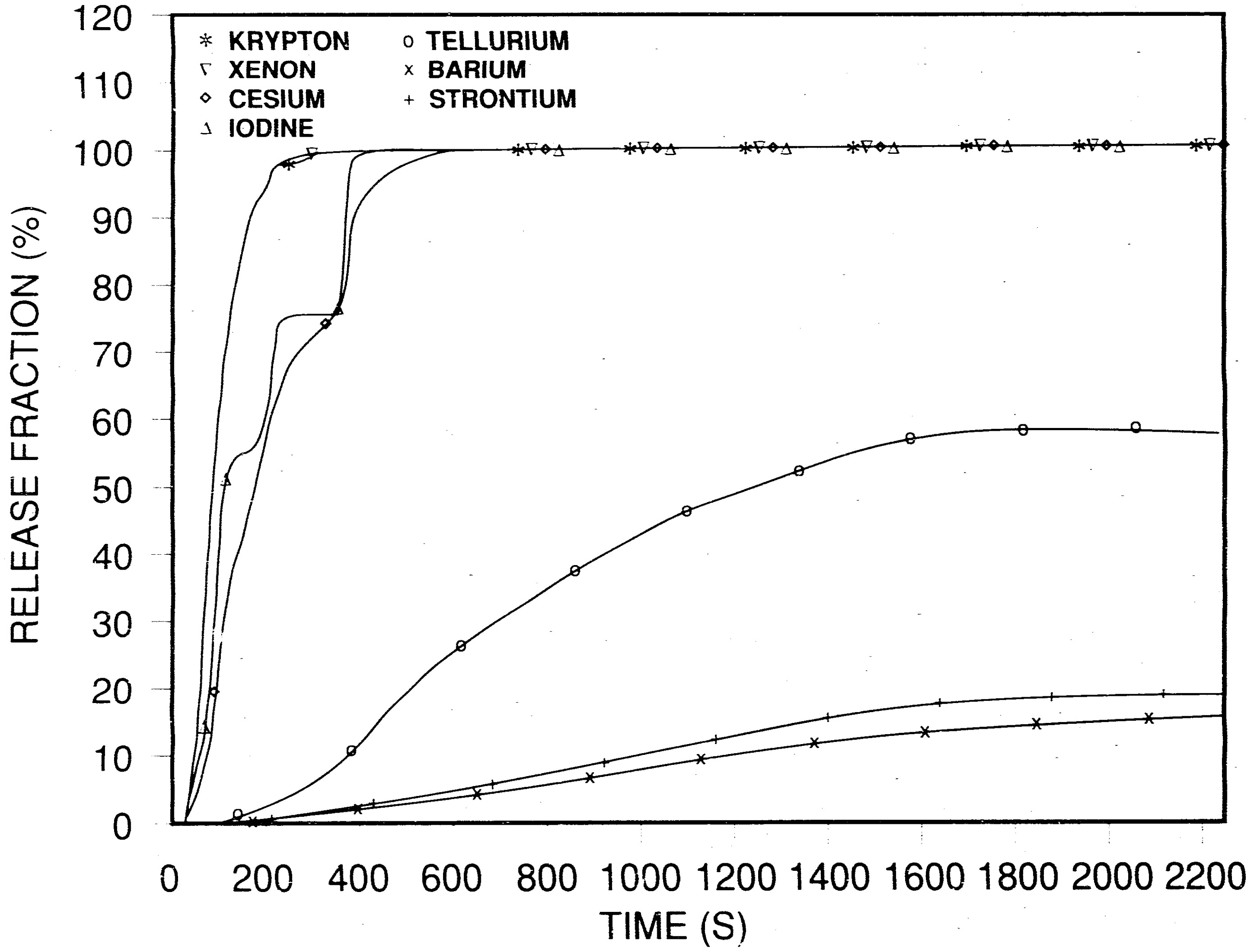


Figure B.5 Sample Problem Graphics Output
Elemental Release Fraction at the Exit Location.

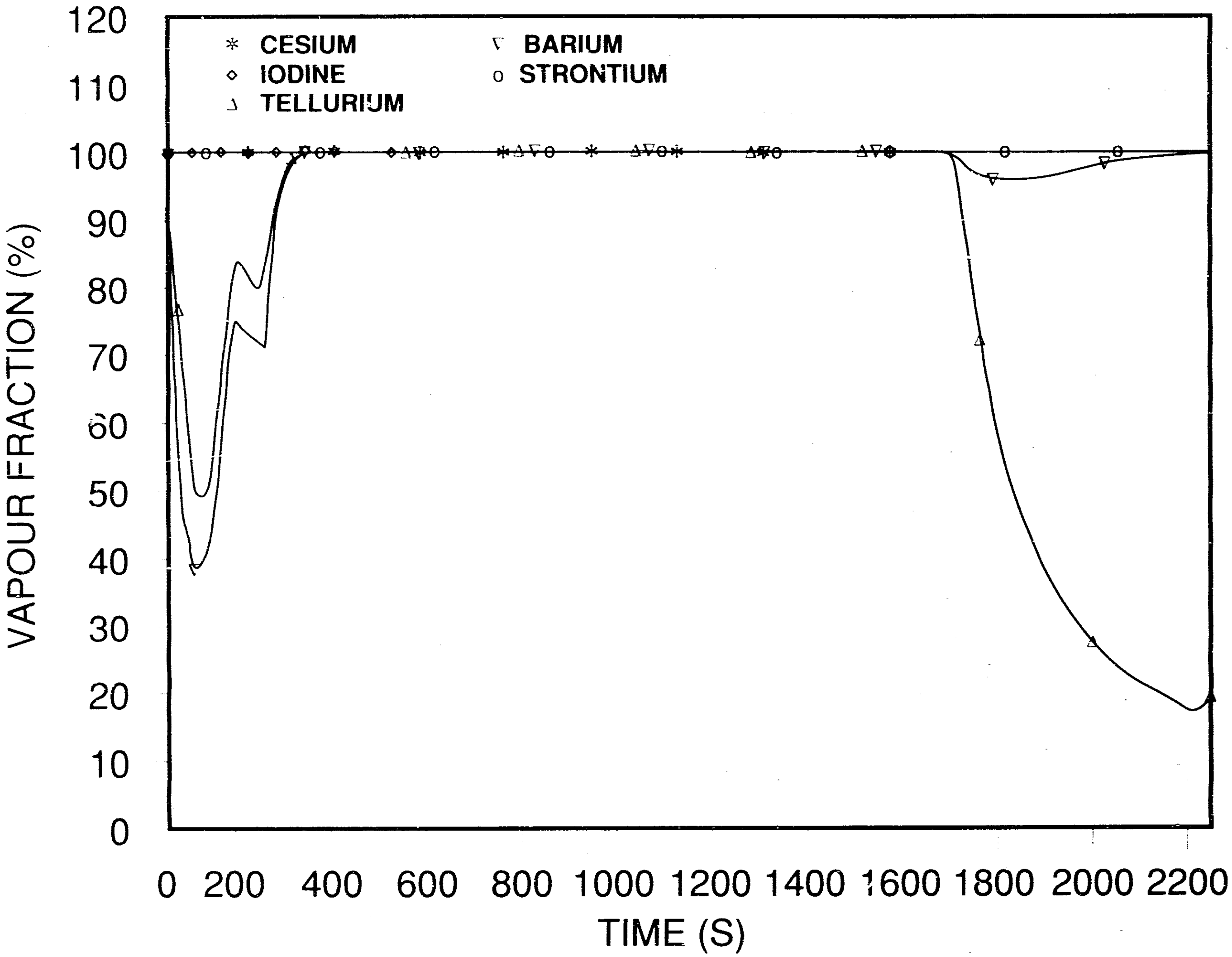


Figure B.6 Sample Problem Graphic Output
Elemental Vapor Fraction at the Exit Location.

C. SUBROUTINE GLOSSARY AND VARIABLE DEFINITIONS

C.1 SUBROUTINE GLOSSARY

Appendix C.1 summarizes the function of each subroutine in VICTORIA. Where a subroutine calls other subroutines, this is shown diagrammatically and not mentioned in the text. There are seventy-one different subroutines, seven of which are called from libraries, as explained below.

This appendix is divided-up according to the functions of the subroutines. Each section contains a figure showing the calling sequence for the subroutines summarized in that section. The subroutines are listed according to order of appearance in the figure, reading left to right, then top to bottom. Five subroutines appear in more than one figure; for convenience, the subroutine summary is repeated whenever this occurs.

Figures C.1-1 to C.1-5 are largely self-explanatory. The logic determining when a subroutine can be called from another is not shown; the reader is referred to the code for details. If a subroutine calls another subroutine in more than one place, this is shown with an asterisk; the called subroutine is shown in the sequence at the first time it could be called. A subroutine called in a sequence in more than one figure is shown with an ampersand (&).

C.1.1 The Main Program and Subroutines

The calling sequence of the main program is shown in Fig. C.1-1.

VIKI90 The main program.

VIKI The driver for the fission product behavior package. It is called once per time step.

FISSIN Controls data input and transfer to the main storage areas.

VFACE Transfers externally supplied information relating to the current time step to the working array in VICTORIA (the 'x' array). The information in the x array concerns flow field properties, such as temperature and speed, and geometric and other properties of the fuel and structures.

POLATE Performs a linear interpolation of data from a look-up table.

VEDIT Controls output of data for a formatted output file and for an unformatted graphics file.

DCHEM Controls the chemistry calculations. Obtains the chemistry 'boundary conditions' for the regions for which chemistry calculations are required and puts them in the chemistry subroutine storage areas. Transfers results of the chemistry calculations to the main storage areas. Calculates oxidation of the clad.

DIFDRV Controls the calculation of the material transport in the fuel region. Calculates diffusion and mass transfer coefficients. Calculates source terms. Sets up the finite difference equations. Transfers the new fuel region concentrations to the main storage areas and updates the grain and clad outer surface inventories.

RELDIF Calculates the transport coefficients for the fuel.

TRNGAS Calculates the transport coefficients for the gas.

SGTSL A SLATEC library subroutine that solves a system of real linear algebraic equations with an asymmetric tridiagonal coefficient matrix; subsection C.1.6 has additional information.

CHARM Calculates the physical properties of aerosols within a cell. Determines agglomeration, deposition, growth, and convection of aerosol particles. Sets up aerosol source and continuity arrays. Transfers the new aerosol distributions to the main storage area.

DECAY Calculates the decay heat power density in the fuel and bulk gas. Calculates the decay heat power per unit area on the structure surfaces.

C.1.2 The Data Input Subroutines

The calling sequence is shown in Fig. C.1-2.

FISSIN Controls data input and transfer to the main storage areas.

SKIP Skips comment lines in the input file. Comment lines must begin with a $ character.

PNTGEN Sets-up pointers in the CONPNTS common block. These pointers index variables associated with the continuity calculations in the fuel and bulk gas in the 'x' array.

PNTCHM Sets-up pointers in the CHMPNTS common block. These pointers index chemistry variables in the 'x' array.

REACTS Converts the species formula matrix to an integer array stored in the chemistry subroutine storage areas (the common block CSPACE here). Calculates the net change in the number of molecules in the gas phase for each species when it dissociates to its component elements in gaseous form. The Gibbs free energy parameters are converted so that the species Gibbs free energies will be calculated relative to energies for the component elements in gaseous form.

PNTSPC Sets-up pointers in the SRCPNTS common block. These pointers index variables in the 'x' array for the fuel region.

POLATE Performs a linear interpolation of data from a look-up table.

AERIN Performs the function of inputing the physical properties affecting the aerosol.

CHMCOL Calculates aerosol particle masses that will serve as collocation points in subsequent calculations.

PNTAER Sets up pointers for the AERPNT common block.

CHMIND Sets up indexing to right-hand side terms for aerosol calculations.

CHMFUN Calculates log(1.-exp(-x)) using series expansions to avoid rounding errors when necessary.

CHMCOE Calculates an integral used in the aerosol calculations.

GAUS8 A SLATEC subroutine that integrates real functions of one variable over finite intervals using an addaptive 8-point Legendre-Gauss algorithm; subsection C.1.6 has additional information.

CHMPJK Calculates the integrand used in the aerosol production term.

CHMFE Calculates values of the finite element basis functions at collocation points.

SSORT A SLATEC subroutine that sorts an array into either increasing or decreasing order ; subsection C.1.6 has additional information.

CHMNOR Normalizes the agglomeration terms to insure conservation of mass.

C.1.3 The Output Subroutines

The calling sequence is shown in Fig. C.1-3.

VEDIT Controls output of data for a formatted output file and an unformatted graphics file.

MASBAL Performs a mass balance of each element over each cell in the fuel grains, fuel pores, fuel-clad gap, clad, clad film, bulk gas, aerosol, and structure film. It also calculates for the entire problem the net amount of bulk gas and aerosol lost and the total amount of aerosol generated for each element. Finally, it calculates an overall mass balance for each element.

VPRINT Controls the flow of data to the output and graphics files.

PRELUD Determines which parts of the imaginary screen need to be reprinted if the screen has to be split into two or more parts.

SCRCHR Indicates that at character string will be printed at a particular location.

SCRINT Indicates that an integer will be printed at a particular location.

SCRFP Indicates that a real number will be printed at a particular location. Optionally, a dot can be printed if the value is zero.

SCREEN Indicates that the imaginary screen will be printed.

C.1.4 The Chemistry Subroutines

The calling sequence is shown in Fig. C.1-4.

DCHEM Controls the chemistry calculations. Obtains the chemistry ‘boundary conditions’ for the regions for which chemistry calculations are required and puts them in the chemistry subroutine storage areas. Transfers results of the chemistry calculations to the main storage areas. Calculates oxidation of the clad.

FUELSL Solves the system of non-linear algebraic equations which determines the equilibrium species concentrations in the fuel grains and at the grain surfaces. A cyclical Newton method is used on a specially chosen set of independent variables.

CHEMRF Calculates the position of an element in a symmetric matrix stored in compact form.

CHEMRS Calculates the equilibrium constants for dissociation of each species into its component elements in gaseous form.

CHEMGE Calculates element mass balances. It can also be used to calculate element abundance.

CHEMWS Calculates species molar concentrations from the gaseous element molar concentrations.

CHEMHE Calculates the Jacobian matrix for the system of non-linear algebraic equations whose solution determines the chemical equilibrium.

SPOFS A SLATEC subroutine that solves a real, positive-definite, symmetric system of linear algebraic equations using the LINPACK subroutines SPOCO and SPOSL; subsection C.1.6 has additional information.

CLADOX Calculates the extent of cladding oxidation.

GASSLV Solves the system of non-linear algebraic equations which determines the equilibrium species concentrations in the surface layer on the clad inner surfaces, in the bulk gas, and in the surface layer on the structures. A cyclical Newton method is used on a specially chosen set of independent variables, different from that chosen in FUELSL.

CHMAER Estimates the physical properties of the aerosol mixture needed for aerosol calculations.

CHMGAS Estimates the physical properties of the gas mixture needed for aerosol calculations.

OMEGLJ Calculates the collision integral based on Leonard-Jones potential theory.

INDXLJ Calculates the index that determines the range over which the collision integral for the Leonard-Jones potential is interpolated.

OMEGSM Calculates the collision integral based on Stockmeyer potential theory.

INDXSM Calculates the index that determines the range over which the collision integral for the Stockmeyer potential is interpolated.

CHMRAD Calculates particle radii at the collocation points.

C.1.5 The Aerosol Behavior and Transport Subroutines

The calling sequence is shown in Fig. C.1-5.

CHARM Calculates the physical properties of aerosols within a cell. Determines agglomeration, deposition, growth, and convection of aerosol particles. Sets up aerosol source and continuity arrays. Transfers the new aerosol distributions to the main storage area.

POLATE Performs a linear interpolation of data from a look-up table.

CHMSLN Sets up the number-density distribution for the aerosol source.

CHMGLN Calculates the log-normal number density distribution given statistical parameters describing the mass distribution of aerosol particles.

CHMRAD Calculates the particle radii at the collocation points.

CHMAER Estimates the physical properties of the aerosol mixture needed for the aerosol calculations.

CHMGAS Estimates properties of thc gas mixture that are needed for aerosol calculations.

CHMMOB Calculates the aerosol particle mobilities at the collocation points.

CHMFLO Calculates flow properties that are needed for calculating deposition velocities and agglomeration rates.

CO5WHE Locates zeros of a function in one variable. Employs inverse quadratic interpolation and bisection.

CHMFAN Calculates the Fanning friction factor for a cylindrical pipe with arbitrary roughness.

CHMAGG Calculates the agglomeration rate of particles at all combinations of the collocation points.

CHMDEP Calculates the particle deposition rates at the collocation points.

CHMDIF Sets up the input for calls to DEBDF, DEABM, or DERKF, which solve systems of ordinary differential equations.

DEBDF A SLATEC library subroutine that solves systems of ordinary differential equations by backward difference formula. This solver is intended for stiff systems; subsection C.1.6 has additional information.

CHMRHS Calculates the rate of change of aerosol particles at each of the collocation points arising from agglomeration, deposition, condensation, and leakage.

DEABM A SLATEC library subroutine that solves systems of ordinary differential equations by the Adams-Bashforth-Moulton formula. This solver is intended for mildly stiff systems; subsection C.1.6 has additional information.

DERKF A SLATEC library subroutine that solves systems of ordinary differential equations by Runge-Kutta formula. This solver is intended for non-stiff systems; subsection C.1.6 has additional information.

CHMZMO Calculates moments of the discretized number density distribution using the trapezium rule.

CHMMOM Called by CHMZMO to perform part of the number density distribution calculations.

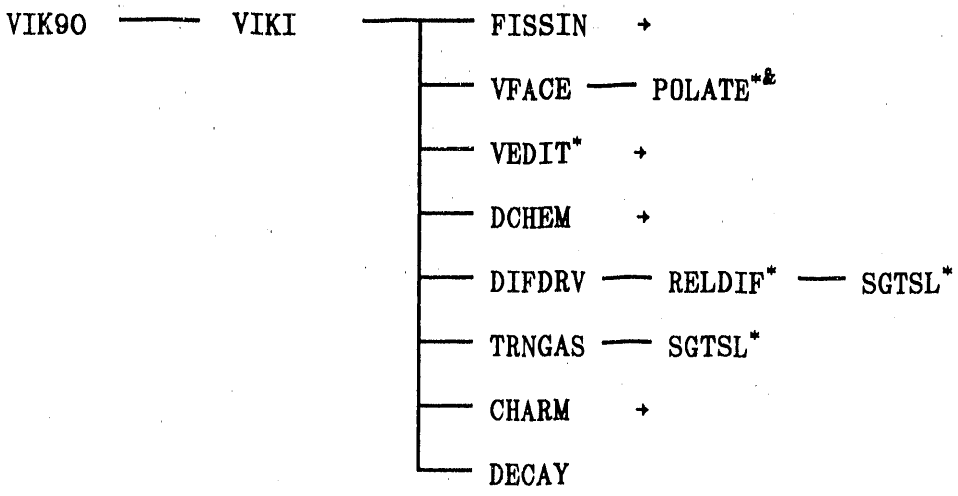
TRNAER Controls the calculation of aerosol convection in the bulk gas. Sets up the finite difference equations, revises the aerosol distribution array, and revises boundary conditions.

SGTSL A SLATEC library subroutine that solves a real system of linear algebraic equations with an asymmetric tridiagonal coefficient matrix; subsection C.1.6 has additional information.

CHMM50 Calculates the difference between the cumulative mass density integrated up to x and half the total mass density. Used to calculate x50.

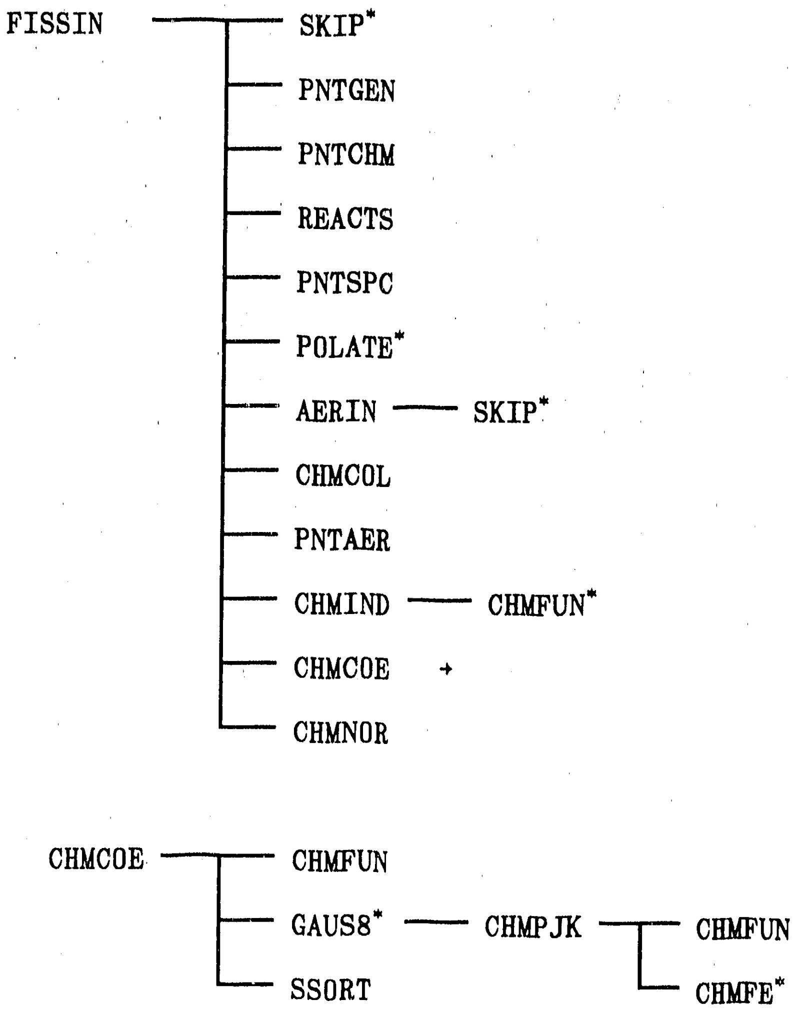
CHMFEO Calculates the integral of a finite element basis function up to a specified location.

CHMFE Calculates the value of one of several finite element basis functions at a point.



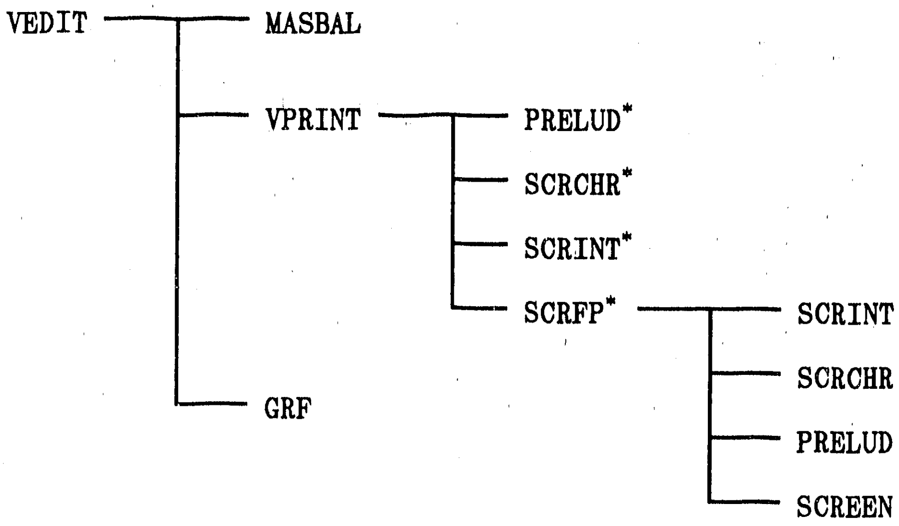
* The subroutine is called more than once.
& The subroutine is called in a sequence in another figure.
→ The subroutine calls other subroutines—see this and subsequent figures.

Fig. C.1-1. The calling sequence of the main program.



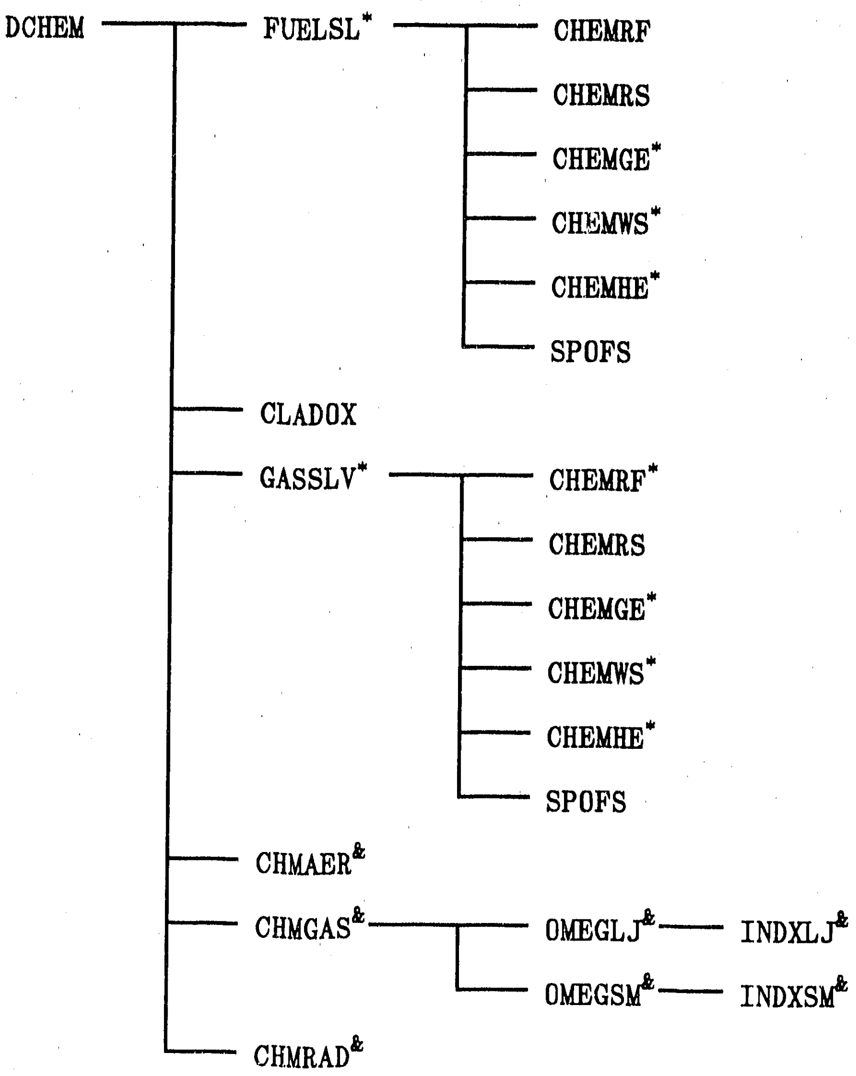
* The subroutine is called more than once.
→ The subroutine calls other subroutines—see this and subsequent figures.

Fig. C.1-2. Calling sequence for the data input subroutines.



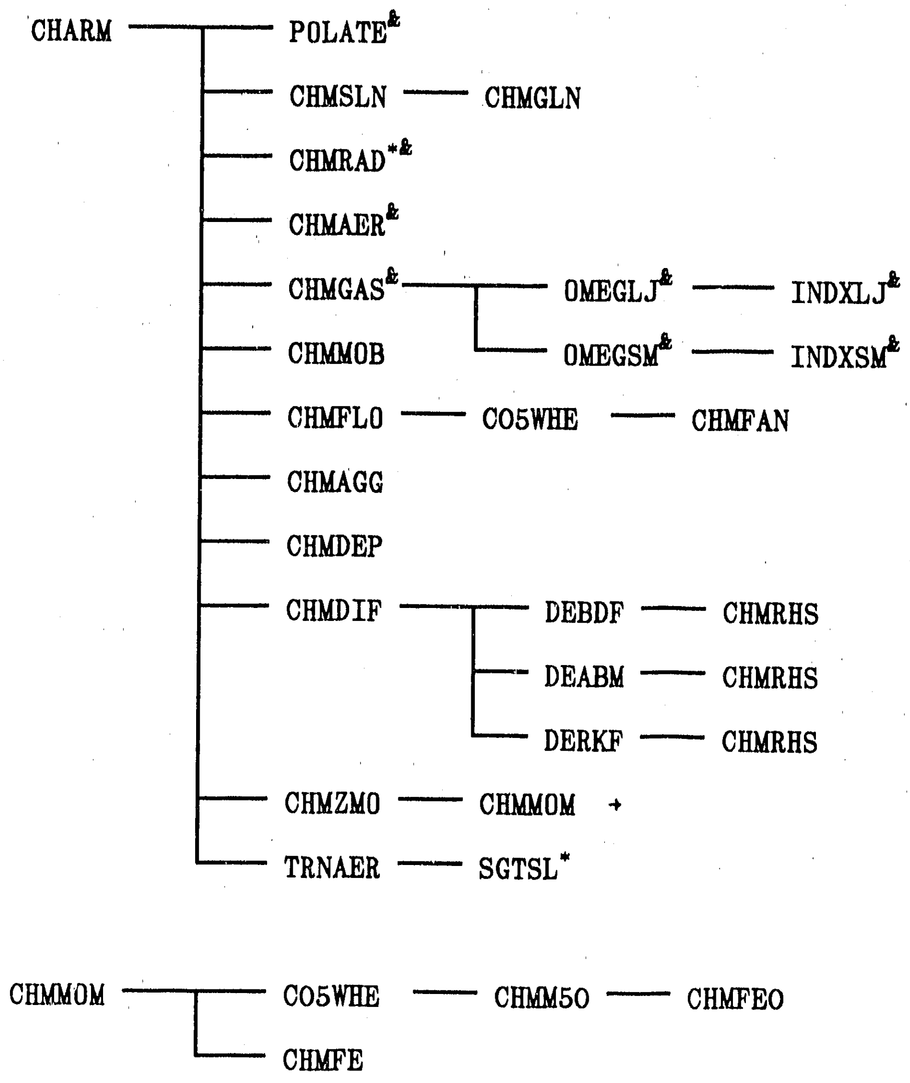
* The subroutine is called more than once.

Fig. C.1-3. Calling sequence for output subroutines.



* The subroutine is called more than once.
& The subroutine is called in a sequence in another figure.

Fig. C.1-4. Calling sequence for chemistry subroutines.



* The subroutine is called more than once.
& The subroutine is called in a sequence in another figure.
→ The subroutine calls other subroutines—see this and subsequent figures.

Fig. C.1-5. Calling sequence for the aerosol subroutines.

C.1.6 Mathematical Software Subroutines

Descriptions of the SLATEC [C.1] subroutines that are called by VICTORIA are included here for convenience. The routines are: SPOFS, SGTSL, GAUS8, SSORT, DEBDFS, DEABM, and DERKF. These descriptions are copied from the SLATEC documentation.

```
****************************************************************
                            SPOFS
****************************************************************

      SUBROUTINE SPOFS(A,LDA,N,V,ITASK,IND,WORK)
 ---DATE WRITTEN   800509   (YYMMDD)
 ---REVISION DATE  820801   (YYMMDD)
 ---CATEGORY NO.  D2B1B
 ---KEYWORDS  LINEAR EQUATIONS,POSITIVE DEFINITE,SYMMETRIC
 ---AUTHOR  VOORHEES, E., (LANL)
 ---PURPOSE  SPOFS solves a POSITIVE DEFINITE SYMMETRIC
           real NXN system of linear equations.
 ---DESCRIPTION
```

Subroutine SPOFS solves a real positive definite symmetric NxN system of single precision linear equations using LINPACK subroutines SPOCO and SPOSL. That is, if A is an NxN real positive definite symmetric matrix and if X and B are real N-vectors, then SPOFS solves the equation

A*X=B.

The matrix A is first factored into upper and lower triangular matrices R and R-TRANSPOSE. These factors are used to find the solution vector X. An approximate condition number is calculated to provide a rough estimate of the number of digits of accuracy in the computed solution.

If the equation A*X=B will be solved for more than one vector B, the factoring of A does not need to be performed again and the option to solve only (ITASK .GT. 1) will be faster for the succeeding solutions. In this case, the contents of A, LDA, and N must not have been altered by the user following factorization (ITASK=1). IND will not be changed by SPOFS in this case.

Argument Description ---

A   REAL(LDA,N)
     on entry, the doubly subscripted array with dimension (LDA,N) which contains the coefficient matrix. Only the upper triangle, including the diagonal, of the coefficient matrix need be entered and will subsequently be referenced and changed by the routine.

on return, contains in its upper triangle an upper triangular matrix R such that A = (R-TRANSPOSE) * R .

LDA INTEGER
the leading dimension of the array N. LDA must be greater than or equal to N. (Terminal error message IND=-1)N INTEGER the order of the matrix A. N must be greater than or equal to 1. (Terminal error message IND=-2)

V REAL(N)
on entry, the singly subscripted array(vector) of dimension N which contains the right hand side B of a system of simultaneous linear equations A*X=B. on return, V contains the solution vector, X .

ITASK INTEGER
If ITASK = 1, the matrix A is factored and then the linear equation is solved. If ITASK .GT. 1, the equation is solved using the existing factored matrix A. If ITASK .LT. 1, then terminal error message IND=-3 is printed.

IND INTEGER
GT. 0 IND is a rough estimate of the number of digits of accuracy in the solution, X. LT. 0 see error message corresponding to IND below.

WORK REAL(N)
a singly subscripted array of dimension at least N.

Error Messages Printed ---

IND=-1	terminal	N is greater than LDA.
IND=-2	terminal	N is less than 1.
IND=-3	terminal	ITASK is less than 1.
IND=-4	Terminal	The matrix A is computationally singular or is not positive definite. A solution has not been computed.
IND=-10	warning	The solution has no apparent significance. The solution may be inaccurate or the matrix A may be poorly scaled.

Note- The above terminal(*fatal*) error messages are
LEVEL=1 (recoverable) and IFLAG=2.
LEVEL=0 for warning error messages from XERROR. Unless the user provides otherwise, an error message will be printed followed by an abort.

```
 ---ROUTINES CALLED  R1MACH,SPOCO,SPOSL,XERROR,XERRWV


***************************************************************
                             SGTSL
***************************************************************


      SUBROUTINE SGTSL(N,C,D,E,B,INFO)
 ---DATE WRITTEN   780814   (YYMMDD)
 ---REVISION DATE  820801   (YYMMDD)
 ---CATEGORY NO.  D2A2A
 ---KEYWORDS  LINEAR ALGEBRA,LINPACK,MATRIX,SOLVE,TRIDIAGONAL
 ---AUTHOR  DONGARRA, J., (ANL) [C.2]
 ---PURPOSE  Solves the system A*X=B where a is TRIDIAGONAL
 ---DESCRIPTION

SGTSL given a general tridiagonal matrix and a right-hand side
will find the solution.

On Entry

     N          INTEGER
                is the order of the tridiagonal matrix.

     C          REAL(N)
                is the subdiagonal of the tridiagonal matrix.
                C(2) through C(N) should contain the
                subdiagonal.  On output, C is destroyed.

     D          REAL(N)
                is the diagonal of the tridiagonal matrix.  On
                output, D is destroyed.

     E          REAL(N)
                is the superdiagonal of the tridiagonal matrix.
                E(1) through E(N-1) should contain the
                superdiagonal.  On output, E is destroyed.

     B          REAL(N)
                is the right hand side vector.

 On Return

     B          is the solution vector.

     INFO       INTEGER
                = 0 normal value.
                = K if the K-th element of the diagonal becomes
                exactly zero.  The subroutine returns when this
                is detected.
```

LINPACK. This version dated 08/14/78.
Jack Dongarra, Argonne National Laboratory.
No externals
Fortran ABS

---ROUTINES CALLED (NONE)

GAUS8

SUBROUTINE GAUS8(FUN,A,B,ERR,ANS,IERR)
---DATE WRITTEN 810223 (YYMMDD)
---REVISION DATE 830510 (YYMMDD)
---CATEGORY NO. H2A1A1
---KEYWORDS ADAPTIVE,AUTOMATIC,GAUSS QUADRATURE,INTEGRATION,
NUMERICAL INTEGRATION,QUADRATURE
---AUTHOR JONES, R. E., (SNLA)
---PURPOSE GAUS8 integrates real functions of one variable over finite intervals using an adaptive 8-point Legendre-Gauss algorithm. GAUS8 is intended primarily for high accurracy integration or integration of smooth functions.
---DESCRIPTION

Written by R.E. Jones

Abstract

GAUS8 integrates real functions of one variable over finite intervals using an adaptive 8-point Legendre-Gauss algorithm. GAUS8 is intended primarily for high accuracy integration or integration of smooth functions.

GAUS8 calls I1MACH, R1MACH, XERROR

Description of Arguments

Input--
FUN - name of external function to be integrated. This name must be in an EXTERNAL statement in the calling program. FUN must be a function of one real argument. The value of the argument to FUN is the variable of integration which ranges from A to B.

A - lower limit of integral
B - upper limit of integral (may be less than A)
ERR - is a requested pseudorelative error tolerance.

Normally pick a value of ABS(ERR) so that STOL .LT. ABS(ERR) .LE. 1.0E-3 where STOL is the single precision unit roundoff=R1MACH(4). ANS will normally have no more error than ABS(ERR) times the integral of the absolute value of FUN(X). Usually, smaller values for ERR yield more accuracy and require more function evaluations.

A negative value for ERR causes an estimate of the absolute error in ANS to be returned in ERR. Note the ERR must be a variable (not a constant) in this case. Note also that the user must reset the value of ERR before making any more calls that use the variable ERR.

Output--

ERR - will be an estimate of the absolute error in ANS if the input value of ERR was negative. (ERR is unchanged if the input value of ERR was non-negative.) The estimated error is solely for information to the user and should not be used as a correction to the computed integral.

ANS - computed value of integral
IERR- a status code

--Normal Codes

1 ANS most likely meets requested error tolernce, or A=B. -1 A and B are too nearly equal to allow normal integration. ANS is set to zero.

--Abnormal Code

2 ANS probably does not meet requested error tolerance.

---ROUTINES CALLED I1MACH,R1MACH,XERROR

```
******************************************************************
                              SSORT
******************************************************************


    SUBROUTINE SSORT(X,Y,N,KFLAG)
 ---DATE WRITTEN   761101   (YYMMDD)
 ---REVISION DATE  820801   (YYMMDD)
 ---CATEGORY NO.  N6A2B1
 ---KEYWORDS  QUICKSORT,SINGLETON QUICKSORT,SORT,SORTING
 ---AUTHOR  JONES, R. E., (SNLA)
            WISNIEWSKI, J. A., (SNLA) [C.3]
```

---PURPOSE SSORT sorts array X and optionally makes the same
interchanges in array Y. The array X may be sorted in increasing order or decreasing order. A slightly modified QUICKSORT algorithm is used.
---DESCRIPTION

Written by Rondall E. Jones

Modified by John A. Wisniewski to use the Singleton quicksort algorithm. Date 18 November 1976.

Abstract

SSORT sorts array X and optionally makes the same interchanges in array Y. The array X may be sorted in increasing order or decreasing order. A slightly modified quicksort algorithm is used.

Description of Parameters
X - array of values to be sorted (usually abscissas)
Y - array to be (optionally) carried along
N - number of values in array X to be sorted
KFLAG - control parameter = 2 means sort X in increasing order and carry Y along.
=1 means sort X in increasing order (ignoring Y)
=-1 means sort X in decreasing order (ignoring Y)
=-2 means sort X in decreasing order and carry Y along.

---ROUTINES CALLED XERROR

****
DEBDFS
****

SUBROUTINE DEBDF(F,NEQ,T,Y,TOUT,INFO,RTOL,ATOL,IDID,RWORK,LRW,
1IWORK,LIW,RPAR,IPAR,JAC)
---DATE WRITTEN 800901 (YYMMDD)
---REVISION DATE 820801 (YYMMDD)
---CATEGORY NO. I1A2
---KEYWORDS BACKWARD DIFFERENTIATION FORMULAS,DEPAC,
INITIAL VALUE PROBLEMS,ODE, ORDINARY DIFFERENTIAL
EQUATIONS, STIFF
---AUTHOR SHAMPINE, L. F., (SNLA)
WATTS, H. A., (SNLA) [C.4]
---PURPOSE Solves initial value problems in ordinary differential equations using backward differentiation formula. It is intended primarily for STIFF problems.

---DESCRIPTION

This is the backward differentiation code in the package of differential equation solvers DEPAC, consisting of the codes DERKF, DEABM, and DEBDF.

DEPAC - Design of a User Oriented Package of ODE Solvers. DEBDF is a driver for a modification of the code LSODE written by A. C. Hindmarsh, Lawrence Livermore Laboratory Livermore, California 94550

****

DEPAC PACKAGE OVERVIEW

****

You have a choice of three differential equation solvers from DEPAC. The following brief descriptions are meant to aid you in choosing the most appropriate code for your problem.

DERKF is a fifth order Runge-Kutta code. It is the simplest of the three choices, both algorithmically and in the use of the code. DERKF is primarily designed to solve non-stiff and mildly stiff differential equations when derivative evaluations are not expensive. It should generally not be used to get high accuracy results nor answers at a great many specific points. Because DERKF has low overhead costs, it will usually result in the least expensive integration when solving problems requiring a modest amount of accuracy and having equations that are not costly to evaluate. DERKF attempts to discover when it is not suitable for the task posed.

DEABM is a variable order (one through twelve) Adams code. Its complexity lies somewhere between that of DERKF and DEBDF. DEABM is primarily designed to solve non-stiff and mildly stiff differential equations when derivative evaluations are expensive, high accuracy results are needed or answers at many specific points are required. DEABM attempts to discover when it is not suitable for the task posed.

DEBDF is a variable order (one through five) backward differentiation formula code. It is the most complicated of the three choices. DEBDF is primarily designed to solve stiff differential equations at crude to moderate tolerances. If the problem is stiff at all, DERKF and DEABM will be inefficient compared to DERKF and DEABM on non-stiff problem because it uses much more storage, has a much larger overhead, and the low order formulas will not give high accuracies efficiently.

The concept of stiffness cannot be described in a few words. If you do not know the problem to be stiff, try either DERKF or DEABM. Both of these codes will inform you of stiffness when the cost of solving such problems becomes important.

****

ABSTRACT

****

Subroutine DEBDF uses the backward differentiation formulas of orders one through five to integrate a system of NEQ first order ordinary differential equations of the form

$$DU/DX = F(X,U)$$

when the vector Y(*) of initial values for U(*) at X=T is given. The subroutine integrates from T to TOUT. It is easy to continue the integration to get results at additional TOUT. This is the interval mode of operation. It is also easy for the routine to return with the solution at each intermediate step on the way to TOUT. This is the intermediate-output mode of operation.

DEBDF uses subprograms LSOD, INTYD, STOD, CFOD, PJAC, SLVS, VNWRMS, HSTART, VNORM, SVCO, RSCO, R1MACH, and the LINPACK routines SGEFA, SGESL, SGBFA, SGBSL (which also uses the BLAS routines SAXPY, SSCAL, ISAMAX and SDOT) and the error handling routine XERRWV. The only machine dependent parameters to be assigned appear in R1MACH.

****

DESCRIPTION OF THE ARGUMENTS TO DEBDF (AN OVERVIEW)

****

The parameters are:

F -- This is the name of a subroutine which you provide to define the differential equations.

NEQ -- This is the number of (first order) differential equations to be integrated.

T -- This is a value of the independent variable.

Y(*) -- This array contains the solution components at T.

TOUT -- This is a point at which a solution is desired.

INFO(*) -- The basic task of the code is to integrate the differential equations from T to TOUT and return an answer at TOUT. INFO(*) is an integer array which is used to communicate exactly how you want this task to be carried out.

RTOL, ATOL -- These quantities represent relative and absolute error tolerances which you provide to indicate how accurately you wish the solution to be computed. You may choose them to be both scalars or else both vectors.

IDID -- This scalar quantity is an indicator reporting what the code did. You must monitor this integer variable to decide what action to take next.

RWORK(*), LRW -- RWORK(*) is a real work array of length LRW which provides the code with needed storage space.

IWORK(*), LIW -- IWORK(*) is an integer work array of length LIW which provides the code with needed storage space.

RPAR, IPAR -- These are real and integer parameter arrays which you can use for communication between your calling program and the F subroutine (and the JAC subroutine).

JAC -- This is the name of a subroutine which you may choose to provide for defining the Jacobian matrix of partial derivatives DF/DU.

Quantities which are used as input items are
NEQ, T, Y(*), TOUT, INFO(*), RTOL, ATOL, RWORK(1), LRW,
IWORK(1), IWORK(2), and LIW.

Quantities which may be altered by the code are
T, Y(*), INFO(1), RTOL, ATOL, IDID, RWORK(*) and IWORK(*).

****

---ROUTINES CALLED LSOD,XERRWV

****

DEABM

****

```
SUBROUTINE DEABM(F,NEQ,T,Y,TOUT,INFO,RTOL,ATOL,IDID,RWORK,LRW,
1IWORK,LIW,RPAR,IPAR,JAC)

 ---DATE WRITTEN   800501   (YYMMDD)
 ---REVISION DATE  851111   (YYMMDD)
 ---CATEGORY NO.  I1A1B
 ---KEYWORDS  ADAMS METHOD,DEPAC,INITIAL VALUE PROBLEMS,ODE,
              ORDINARY DIFFERENTIAL EQUATIONS,PREDICTOR-CORRECTOR
 ---AUTHOR  SHAMPINE, L. F., (SNLA)
            WATTS, H. A., (SNLA) [C.4]
 ---PURPOSE  Solve initial value problems in ordinary
             differential equations using an Adams-Bashforth
             method.
 ---DESCRIPTION
```

This is the Adams code in the package of differential equation solvers DEPAC, consisting of the codes DERKF, DEABM, and DEBDF.

DEABM is a driver for a modification of the code ODE written by L. F. Shampine and M. K. Gordon

****

DEPAC PACKAGE OVERVIEW

****

You have a choice of three differential equation solvrrs from DEPAC. A brief description of each is given with the comments on routine DEBDF.

****

ABSTRACT

****

Subroutine DEABM uses the Adams-Bashforth-Moulton predictor-corrector formu1as of orders one through twelve to integrate a system of NEQ first order ordinary differential equations of the form

DU/DX = F(X,U)

when the vector Y(*) of initial values for U(*) at X=T is given.

The subroutine integrates from T to TOUT. It is easy to continue the integration to get results at additional TOUT. This is the interval mode of operation. It is also easy for the routine to return with the solution at each intermediate step on the way to TOUT. This is the intermediate-output mode of operation.

DEABM uses subprograms DES, STEPS, SINTRP, HSTART, VNORM, R1MACH and the error handling routine XERRWV. The only machine dependent parameters to be assigned appear in R1MACH.

****

DESCRIPTION OF THE ARGUMENTS TO DEABM (AN OVERVIEW)

****

The parameters are

F -- This is the name of a subroutine which you provide to define the differential equations.

NEQ -- This is the number of (first order) differential equations to be integrated.

T -- This is a value of the independent variable.

Y(*) -- This array contains the solution components at T.

TOUT -- This is a point at which a solution is desired.

INFO(*) -- The basic task of the code is to integrate the differential equations from T to TOUT and return an answer at TOUT. INFO(*) is an integer array which is used to communicate exactly how you want this task to be carried out.

RTOL, ATOL - These quantities represent relative and absolute error tolerances which you provide to indicate how accurately you wish the solution to be computed. You may choose them to be both scalars or else both vectors.

IDID -- This scalar quantity is an indicator reporting what the code did. You must monitor this integer variable to decide what action to take next.

RWORK(*),LRW-- RWORK(*) is a real work array of length LRW which provides the code with needed storage space.

IWORK(*), LIW -- IWORK(*) is an integer work array of length LIWwhich provides the code with needed storage space.

RPAR, IPAR -- These are real and integer parameter arrays which you can use for communication between your calling program and the F subroutine.

Quantities which are used as input items are
NEQ, T, Y(*), TOUT, INFO(*), RTOL, ATOL, RWORK(1), LRW and LIW.

Quantities which may be altered by the code are
T, Y(*), INFO(1), RTOL, ATOL, IDID, RWORK(*) and IWORK(*).

```
*****************************************************************
---ROUTINES CALLED  DES,XERRWV


****************************************************************
                              DERKF
****************************************************************


SUBROUTINE DERKF(F,NEQ,T,Y,TOUT,INFO,RTOL,ATOL,IDID,RWORK,LRW,
1IWORK,LIW,RPAR,IPAR,JAC)

---DATE WRITTEN   800501   (YYMMDD)
---REVISION DATE  820801   (YYMMDD)
---CATEGORY NO.  I1A1A
---KEYWORDS  DEPAC,INITIAL VALUE,ODE,ORDINARY DIFFERENTIAL
             EQUATIONS,RKF,RUNGE-KUTTA METHODS
---AUTHOR  WATTS, H. A., (SNLA)
           SHAMPINE, L. F., (SNLA) [C.3-4]
---PURPOSE  DERKF solves initial value problems in ordinary
            differential equations.
---DESCRIPTION
```

This is the Runge-Kutta code in the package of differential equation solvers DEPAC, consisting of thc codes DERKF, DEABM, and DEBDF.

DERKF is a driver for a modification of the code RKF45 written by

H. A. Watts and L. F. Shampine
Sandia Laboratories
Albuquerque, New Mexico 87185

****

DEPAC PACKAGE OVERVIEW

****

You have a choice of three differential equation solvers from DEPAC. A brief description of each is given with the comments on routine DEBDF.

****

ABSTRACT

****

Subroutine DERKF uses a Runge-Kutta-Fehlberg (4,5) method to integrate a system of NEQ first order ordinary differential equations of the form

$$DU/DX = F(X,U)$$

when the vector Y(*) of initial values for U(*) at X=T is given.

The subroutine integrates from T to TOUT. It is easy to continue the integration to get results at additional TOUT. This is the interval mode of operation. It is also easy for the routine to return with the solution at each intermediate step on the way to TOUT. This is the intermediate-output mode of operation.

DERKF uses subprograms DERKFS, DEFEHL, HSTART, VNORM, R1MACH, and the error handling routine XERRWV. The only machine dependent parameters to be assigned appear in R1MACH.

****

DESCRIPTION OF THE ARGUMENTS TO DERKF (AN OVERVIEW

****

The parameters are

F -- This is the name of a subroutine which you provide to define the differential equations.

NEQ -- This is the number of (first order) differential equations to be integrated.

T -- This is a value of the independent variable.

Y(*) -- This array contains the solution components at T.

TOUT -- This is a point at which a solution is desired.

INFO(*) -- The basic task of the code is to integrate the differential equations from T to TOUT and return an answer at TOUT. INFO(*) is an integer array which is used to communicate exactly how you want this task to be carried out.

RTOL, ATOL -- These quantities represent relative and absolute error tolerances which you provide to indicate how accurately you wish the solution to be computed. You may choose them to be both scalars or else both vectors.

IDID -- This scalar quantity is an indicator reporting what the code did. You must monitor this integer variable to decide what action to take next.

RWORK(*), LRW -- RWORK(*) is a real work array of length LRW provides the code with needed storage space.

IWORK(*), LIW -- IWORK(*) is an integer work array of length LIW which provides the code with needed storage space.

RPAR, IPAR -- These are real and integer parameter arrays which you can use for communication between your calling program and the F subroutine.

Quantities which are used as input items are
NEQ, T, Y(*), TOUT, INFO(*),
RTOL, ATOL, LRW and LIW.

Quantities which may be altered by the code are
T, Y(*), INFO(1), RTOL, ATOL,
IDID, RWORK(*) and IWORK(*).

---ROUTINES CALLED DERKFS,XERRWV

variable. If the user/developer wants to give the array an alias (for example P) then the array must be in the subroutine argument list. An example of this latter idea is shown in the subroutine VEDIT call to GRF and the resulting argument list within GRF.

At this time it is anticipated that either the X array will be removed in later versions or else be consistently used with an alias.

C.2 VICTORIA Data Management

When VICTORIA was originally developed there was a concern about the possibility that because of its large size and slow speed that the code would not be generally used. To increase the speed of the calculation several routines were vectorized and several numerical algorithms were replaced, and indeed the calculation time is now in the range of 1 cpu second per second of real time. In addition a series of input flags was added to turn off the calculation of a module, hence a user can run a chemistry problem without any aerosol calculation. To limit the size of the code, the developers chose to use a dynamic storage system. With this system the size of the code is determined from the input information. Therefore if a user wanted to only examine fission product release from the fuel they could choose to use only a single fluids cell and the code would automatically be smaller. Although the concept is worthwhile, its implementation has not always been even. Some developers chose to use aliases for the variable, hence in the output routines the developer sees the array VOLUM. Other developers chose to remain with the container array, hence in the input routines the developer sees the pointer X(LVOLUM+..). Both systems use the identical storage locations and are interchangeable. However some user/developers feel that the former system is more easily understandable.

The container array currently used by VICTORIA is the X array. Variables area addressed as X(L...+...) where the L... are listed in the variable dictionary in section C.3. In that dictionary we have listed a definition, an alias (that may not be currently implemented), an equivalent dimension (for example (JMAX,IMAX)), and a list of subroutines where the variable has been calculated. With this information the user/developer should be able to follow the code calculation and relate it to the text.

To modify the code and add arrays (for example making the pressure fuel grain dependent) the user/developer must become familiar with the pointer routines (PNTGEN, PNTCHM, PNTSPC, and PNTAER). These routines set up the geometry, chemistry, species, and aerosol information respectively. The user determines the size of the new array in terms of the maximum size (for example IMAX by JMAX) and after giving the array a pointer name (for example LP) adds it to the list of the pointer routine used for that modification. The code will automatically increase the X array correctly. If the pointer name is in common, then those routines with that common can address the new

C.3 VICTORIA VARIABLE DEFINITIONS

Variable	Common	Set In	Alias	Description
aggrt(dim,dim)	agglom	chmagg		agglomeration rates for pairs of particles at all combinations of collocation points
akn	aerslc	dataer,aerin		Knudsen-Weber coefficient a
amdens(lstval*lsrce)	addaer	dataer,aerin		aerosol source molar density
arad50(lsrce)	addaer	dataer,aerin		mass median particle radii of aerosol source
areac	cell	charm		ceiling area of current cell
areaf	cell	charm		floor area of current cell
areaw	cell	charm		wall area of current cell
asigma(lsrce)	addaer	dataer,aerin		standard deviation of aerosol source
atime(lstval)	addaer	dataer,aerin		times for aerosol source interpolation
atomwt(25)	chmprm	datchm		atomic weight of each species
avogad	vikprm	datgen		Avogadro's number
bkn	aerslc	dataer,aerin		Knudsen-Wever coefficient b
blflag	flow	dataer,aerin		boundary layer flag for aerosol calculations
boltsm	const	dataer,fissin		Boltsman constant
cladfl	inputs	fissin,vface		flag indicating clad failure
cladth	vikprm	fissin		thickness of cladding
cm	aerslc	dataer,aerin		Brock coefficient (k in CHARM manual)
comp	smtest	aerin		species either single- or multi-component aerosol system
cs	aerslc	dataer,aerin		Brock coefficient cs (cm in Charm manual)
cshpfc	aerslc	dataer,aerin		collision shape factor of aerosol
ct	aerslc	dataer,aerin		Brock coefficient ct
dblthk(dim)	flow	dataer,aerin, chmflo		diffusion boundary layer thicknesses
delta	vikprm	datgen		surface film thickness
densty(laero)	adders	dataer,chmsln, chmaer	cdense	mass density of aerosol particles
deprtc(dim)	agglom	chmdep		deposition rate of aerosol on ceiling
deprtf(dim)	agglom	chmdep		deposition rate of aerosol on floor
deprtw(dim)	agglom	chmdep		deposition rate of aerosol on walls
diffus	gaspro	charm		diffusiophoresis coefficient
dim		dataer		number of collocation points in aerosol calculation
dlogem	colpts	chmcol		log of collocation point spacing
dshpfc	aerslc	dataer,aerin		dynamic shape factor of aerosol
dynvis	gaspro	chmgas		dynamic viscosity
dsero(16)	chmprm	datchm,fissin		coefficient in binary diffusion function
edydis	flow	chmflo		eddy dissipation rate
elems(25)	chmnam	datchm		symbols for each element in VICTORIA
endtv	vikprm	fissin		end time for problem
epower(25)	chmprm	datchm		element decay powers
epsa	chmspc	c05whe,aerin		relative error tolerance in aerosol calc.
epsc	tolers	dataer,fissin		relative error tolerance in equilibrium chemistry calculations
eqvrgh	cell	charm		roughness of surfaces
eta	tolers	dataer,aerin, c05whe		absolute error tolerance in aerosol calc.

Variable	Common	Set In	Alias	Description
frmass(laero,lsrce)	addaer	dataer,aerin		mass fraction of each species in aerosol source
ftd	srcpnt	fissin		fuel fraction of theoretical density
ftemp(itemps)	inputs	fissin		fuel temperatures at each ftime
ftime(itime)	inputs	fissin		times when fuel temperatures are given
fuelod	vikprm	fissin		outside diameter of fuel pellet
gapth	vikprm	fissin		width of fuel-clad gap (radial)
gdense	gaspro	chmgas		total gas density
ge(25)	chmspc	chemge,gasslv, fuelsl		element mass balances
geomom	mominf	chmmom		geometric mean mass of aerosol
gfactr	const	dataer		gravitational acceleration
gibbs(4,167)	chmprm	datchm,reacts		coefficients for Gibbs free energies for each species
gomean(Ø:laero)	moment	chmsmo		geometric mean mass for aerosol at collocation points
gomsln(laero)	lognrm	chmgln	gomgln	log-normal parameter describing mass median mass
gscnst	const	dataer,fissin		ideal gas constant
gtemp(itemps)	inputs	fissin		gas temperatures at each gtime
gthrmc	gaspro	chmgas		mixture thermal conductivity
gtime(itime)	inputs	fissin		times when gas temperatures are given
he(325)	chmspc	chemhe,gasslv, fuelsl		Jacobian matrix stored in upper diaganol, see CHEMRF
hepr	vikprm	fissin		helium fuel fill pressure
hwidth	indcoe	chmind		half width of finite element
hydian	cell	charm		hydraulic diameter
i2j2	vikprm	fissin		imax2*jmax2
iaero	modu	fissin		flag for aerosol module (Ø=off,1=on)
ibend	cell	charm		flag for aerosol deposition in a bend
ibound	vikprm	fissin		2*(imax+jmax)
ichem	modu	fissin		flag for chemistry module (Ø=off,1=on)
iconvc	modu	fissin		flag for convection module (Ø=off,1=on)
idaero(167)	chmprm	datchm		identifies species treated in aerosol module
idecay	modu	fissin		flag for decay heat module (Ø=off,1=on)
identy(laero)	addens	dataer		identifies which aerosol species are metals
idfuel(167)	chmprm	datchm		identifies species treated in fission gas release calculations
idgas(167)	chmprm	datchm		identifies species treated in equilibrium chemistry module
idum	dumcnt	chmdif		number of active aerosol species
ifn	vikprm	fissin		nfuel+nclad+1
ifprd	vikprm	datgen,viki		prints data at initial time if set to Ø
ifuel	vikprm	fissin		nfuel+nclad+2
igscyc	chmspc	fissin		maximum no. of cyclic Newton iterations
igsmax	chmspc	fissin		maximum no. of chemistry iterations
ij	vikprm	fissin		imax*jmax
ij3	vikprm	pntgen		ij*3
ijfuel	vikprm	fissin		ifuel*jmax
imax	vikprm	fissin,gasslv, fuelsl		number of radial cells at an axial level

Variable	Common	Set In	Alias	Description
imax2	vikprm	fissin		imax+2
index(-2:dim)	indcoe	chmind,omeglj, omegsm		pointer index for right-hand side in the aerosol calculation
inev	vikprm	datgen,viki9Ø	ivpr	location of new values (1 or 2)
iolv	vikprm	datgen,viki9Ø	ivpr	location of old values (2 or 1)
irodr	modu	fissin		flag for rod release module (Ø=off,1=on)
ispnum(lsrce)	addaer	aerin		chemistry number of each aerosol source
istp	vikprm	viki9Ø		time step counter
itvals	inputs	fissin		number of times for transient variables
ivlnum(lsrce)	addaer	aerin		volume of aerosol source for each species
jbar	jkbar	chmcoe,chmmor, chmrhs		pointer used to generate the right-hand side of the aerosol equations
jbarmn	indcoe	chmind		minimum value of jbar
jmap(laero)	dumcnt	chmdif		maps aerosol source numbers
jmax	vikprm	fissin		number of axial cells at a radial ring
jmax2	vikprm	fissin		jmax+2
jobnam	jobdet	datgen,fissin		job name
jobtit	jobdet	datgen,fissin		job title
kbar	jkbar	chmmor		pointer used to calculate agglomeration terms
kbarmn(-2:dim)	indcoe	chmind		minimum values of kbar
kbarmx(-2:dim)	indcoe	chmind		maximum values of kbar
la	aerpnt	fissin,dchem, charm	q	pointer in x array to aerosol species concentration, q(nsaero,ncoll,jmax,imax)
labot	genpnt	reldif,trngas, trnaer	abot	pointer in x array to area of cell bottom, abot(ifuel,jmax,imax)
lacel1	genpnt	trngas	acel1	pointer in x array to fuel surface area, acel1(ifuel,jmax,imax)
lacel2	genpnt	trngas,trnaer	acel2	pointer in x array to fuel surface area, acel2(ifuel,jmax,imax)
lacel3	genpnt	trngas,trnaer	acle3	pointer in x array to fuel surface area, acel3(ifuel,jmax imax)
lacell	genpnt	reldif	acell	pointer in x array to area available for flow, acell
laclng	genpnt	fissin	aclng	pointer in x array to ceiling surface area, aclng(jmax,imax)
lacomp	aerpnt	dchem,charm	acomp	pointer in x array to variable, acomp(nsaero)
laflor	genpnt	fissin	aflor	pointer in x array to floor surface area, aflor(jmax,imax)
lafsf	genpnt	trngas	afsf	pointer in x array to variable, afsf(ifuel,jmax,imax)
lagmd	aerpnt	charm	agmd	pointer in x array to variable, agmd(jmax,imax)
laleft	genpnt	reldif,trngas, trnaer	aleft	pointer in x array to area of left edge of cell,aleft(ifuel,jmax,imax)
lalf	srcpnt	vface	alf	pointer in x array to pore volume fraction, alf(ifn,jmax,imax)
lalphv	genpnt	pntgen	alphv	pointer in x array to vapor volume fraction, alphv(jmax,imax)
lammd	aerpnt	charm	ammd	pointer in x array to aerodynamic mass median diameter, ammd(jmax,imax)
larear	genpnt	fissin	arear	pointer in x array to flow area on right surface, arear(jmax,imax)

Variable	Common	Set In	Alias	Description
lareaz	genpnt	fissin	areaz	pointer in x array to flow area at upper surface, areaz(jmax,imax)
larght	genpnt	reldif,trngas, trnaer	arght	pointer in x array to area of right edge of cell, arght(ifuel,jmax,imax)
lassf	genpnt	trngas	assf	pointer in x array to variable, assf(ifuel,jmax,imax)
lasum	aerpnt	fissin,trnaer	asum	pointer in x array to variable, asum(nsaero,ncoll)
lat	aerpnt	dchem	qt	pointer in x array to variable, qt(ncoll)
latop	genpnt	reldif,trngas trnaer	atop	pointer in x array to area of top of cell, atop(ifuel,jmax,imax)
lawall	genpnt	fissin	awall	pointer in x array to wall surface area, awall(jmax,imax)
lbdflg	genpnt	fissin	bdfld	pointer in x array to boundary flags, bdflg(jmax,imax)
lbetar	srcpnt	vface	betar	pointer in x array to variable, betar(ifuel,jmax,imax)
lbetaz	srcpnt	vface	betaz	pointer in x array to variable, betaz(ifuel,jmax,imax)
lcellm	chmpnt	masbal	cellm	pointer in x array to variable, cellm(jmax,imax,ns)
lclmas	chmpnt	masbal	clmas	pointer in x array to variable, clmas(jmax,imax,ne,8)
lconc	srcpnt	fissin,dchem, cladox,vface, trngas	concen	pointer in x array to fuel species concentration, concen(ifuel,jmax,2,ns,imax)
lcumd	aerpnt	fissin,charm	cumdep	pointer in x array to cumulative deposition of each aerosol, cumdep(nsaero,6,jmax,imax)
ldeltr	srcpnt	fissin	deltr	pointer in x array to delta radius of current cell, deltr(imax)
ldeltz	srcpnt	fissin	deltz	pointer in x array to delta height of current cell, deltz(jmax)
ldepra	aerpnt	charm	deprat	pointer in x array to deposition rate, deprat(jmax,imax)
leakrt	cell	charm		leak rate out of cell
lelmas	chmpnt	masbal	elmas	pointer in x array to variable, elmas(ne,14)
length	cell	charm		delta height of current cell
lfdpow	srcpnt	decay	fdpow	pointer in x array to decay heat in fuel, fdpow(ifuel,jmax,imax)
lffilm	srcpnt	reldif,trngas	ffilm	pointer in x array to outer clad film concentration, ffilm
lfldf	srcpnt	reldif	fldf	pointer in x array to variable, fldf(ifuel,jmax)
lfod	srcpnt	reldif	fod	pointer in x array to geometric diameter, fod(jmax,imax)
lform(25,167)	chmprm	datchm		number of occurances of each element in each species formula
lfpinv	srcpnt	fissin,dchem, reldif	origen	pointer in x array to variable, origen(ifuel,jmax,ns,imax)
lgasdf	crcpnt	trngas	gasdf	pointer in x array to variable, gasdf
lgcomp	aerpnt	dechem,charm	gascmp	pointer in x array to variable, gascmp(nsgas)
lgdpow	srcpnt	decay	gdpow	pointer in x array to decay heat in gas, gdpow(jmax,imax)
lgeosd	aerpnt	charm	geosd	pointer in x array to aerodynamic settling diameter(jmax,imax)
lgibbs	chmpnt	chemrs	gibbs	pointer in x array to species Gibbs free energy, gibbs(ns)
lhgts	genpnt	fissin	hgts	pointer in x array to height of cell centers, hgts(jmax2,imax2)

Variable	Common	Set In	Alias	Description
lhm	srcpnt	pntspc	hm	pointer in x array to mass transfer coefficients, hm
lhydr	genpnt	fissin	hydr	pointer in x array to variable, hydr(jmax,imax)
libcs	genpnt	fissin	bcs	pointer in x array to value of boundary condition, bcs(ibound)
lmat	genpnt	reldif,trngas, trnaer	mat	pointer in x array to matrix values for cell or fuel mode, mat
lmat1	genpnt	trngas	mat1	pointer in x array to variable, mat1
lmat2	genpnt	trngas,trnaer	mat2	pointer in x array to variabie, mat2
lmat3	genpnt	trngas,trnaer	mat3	pointer in x array to variable, mat3
lngas(167)	chmprm	reacts		number of moles of gas-phase reactants for each species less unity
lnrods	genpnt	fissin	rods	pointer in x array to number of rods per ring, rods(jmax,imax)
logemf	colpts	chmcol		value used to determine the index of current collocation point
lpar	genpnt	charm,trnaer	par	pointer in x array to aerosol density, par(jmax2,imax2,2,nsaero,ncoll)
lpgas2	genpnt	fissin,vface	pgas2	pointer in x array to total vapor pressure, pgas2(jmax,imax)
lrads	genpnt	fissin	rads	pointer in x array to cell center radius, rads(jmax2,imax)
lrhfpi	genpnt	fissin	rhfpi	pointer in x array to initial gas densities, rhfpi(jmax2,imax2,nsgas)
lrhofp	genpnt	fissin,dchem, cladox,trngas	rhofp	pointer in x array to alpha*rho for H2 and H20, rhofp(jmax2,imax2,2,nsgas)
lrhov	genpnt	dchem,charm	rhov	pointer in x array to vapor density, rhov(jmax,imax)
lrhs	genpnt	reldif,trngas, trnaer	rhs	pointer in x array to right hand side vector, rhs(ifuel,jmax,imax)
lrodr	srcpnt	reldif	rodr	pointer in x array to node radius in current cell, rodr(ifn)
lrodzs	srcpnt	fissin,vface	rodzs	pointer in x array to upper cell surface, rodzs(jmax)
lrsgas	chmpnt	chemrs	rsgas	pointer in x array to factor for equilibrium constants, rsgas(ns)
lrsliq	chmpnt	chemrs	rsliq	pointer in x array to factor for equilibrium constants, rsliq(ns)
lscum	aerpnt	fissin,charm	scum	pointer in x array to variable, scum(nsaero)
lsdpow	srcpnt	decay	sdpow	pointer in x array to decay heat on structure, sdpow(jmax,imax)
lsetaz	aerpnt	charm	setaz	pointer in x array to variable, setaz(jmax,imax,ncoll)
lsetfz	genpnt	fissin	setafz	pointer in x array to cell settling volume, setafz(jmax,imax)
lsetvz	aerpnt	charm	setvz	pointer in x array to variable, setvz(jmax2,imax2,ncoll)
lsfilm	genpnt	fissin,dchem, charm,trngas	sfilm	pointer in x array to species concentrations of structure surface film, sfilm(jmax,imax,ns)
lsoura	genpnt	pntgen	soura	pointer in x array to variable, soura(jmax,imax,ns)
lspsum	genpnt	trngas	spsum	pointer in x array to mass lost out of top in gas phase, spsum(nsgas)
lstrca	genpnt	fissin	strca	pointer in x array to structure area, strca(jmax,imax)
lstrfg	genpnt	fissin	strfg	pointer in x array to structure flag, strfg(jmax,imax)

Variable	Common	Set In	Alias	Description
lsuml1	chmpnt	dchem	suml1	pointer in x array to variable, suml1(jmax,imax)
lsuml2	chmpnt	dchem	suml2	pointer in x array to variable, suml2(jmax,imax)
lsuml3	chmpnt	dchem	suml3	pointer in x array to variable, suml3(jmax,imax)
lsuml4	chmpnt	dchem	suml4	pointer in x array to variable, suml4(jmax,imax,nfuel)
lsuml5	chmpnt	dchem	suml5	pointer in x array to variable, suml5(jmax,imax)
ltdd	genpnt	trngas,trnaer	tdd	pointer in x array to variable, tdd(3,jmax,imax)
ltdl	genpnt	trngas,trnaer	tdl	pointer in x array to variable, tdl(3,jmax,imax)
ltdr	genpnt	trngas,trnaer	tdr	pointer in x array to variable, tdr(3,jmax,imax)
ltds	genpnt	trngas,trnaer	tds	pointer in x array to variable, tds(3,jmax,imax)
ltfuel	srcpnt	fissin	tfuel	pointer in x array to fuel temperature, tfuel(jmax,imax)
ltgas2	genpnt	fissin,vface	tgas2	pointer in x array to vapor temperature, tgas2(jmax,imax)
ltstrc	genpnt	fissin,vface	tstrc	pointer in x array to stucture temperature, tstrc(jmax,imax)
lvolf	genpnt	fissin	volf	pointer in x array to variable, volf(jmax,imax)
lvolfl	genpnt	fissin	volfl	pointer in x array to variable, volfl(jmax,imax)
lvolfn	srcpnt	masbal	volfn	pointer in x array to volumes at nodes, volfn(ifuel,jmax,imax)
lvolum	genpnt	fissin	volum	pointer in x array to cell fluid volume, volum(jmax,imax)
lvrgas	genpnt	fissin,vface	vrgas	pointer in x array to radial vapor velocity, vrgas(jmax2,imax)
lvzgas	genpnt	fissin,vface	vzgas	pointer in x array to axial vapor velocity, vzgas(jmax2,imax)
lx1	chmpnt	dchem	x1	pointer in x array to values of x for gas region, x1(ne,imax,jmax)
lx2	chmpnt	dchem	x2	pointer in x array to values of x for clad/gap interface, x2(ne,imax,jmax)
lx3	chmpnt	dchem	x3	pointer in x array to values of x for structure surface region, x3(ne,imax,jmax)
lx4	chmpnt	dchem	x4	pointer in x array to values of x for fuel, x4(ne,imax,jmax,nfuel)
lx5	chmpnt	dchem	x5	pointer in x array to variable, x5(ne,imax,jmax,nfuel)
lxbcs	genpnt	fissin	xbcs	pointer in x array to boundary species concentration, xbcs(ibound,nsgas)
m5Ømom	mominf	chmmom		mass median mass
m5Øsln(laero)	lognrm	chmgln	m5Øgln	log-normal parameters for mass median mass
mass(dim)	colpts	chmcol		aerosol masses at each collocation point
mass5Ø(Ø:laero)	moment	chmsmo		mass median mass for each aerosol species
maxcls	tolers	dataer,cØ5whe		maximum no. of function calls to locate a sero
maxng	chmprm	reacts		maximum value of array elements pointed to by lngas
maxtry	tolers	dataer		maximum no. of integration points
mlower	colpts	dataer,aerin		mass of smallest aerosol particle
mdense(Ø:laero)	moment	dchme,charm chmgln,chmqmo		species densities used in aerosol calculations
mdensl(laero)	lognrm	charm	mdngln	species mass densities in aerosol source
mdnmom	mominf	chmmom		mass mean density of aerosol
mfrpth	gaspro	dataer,chmgas		mean free path
mobil(dim)	colpts	chmmob		mobility factor for aerosol particles at collocation points

Variable	Common	Set In	Alias	Description
molwt	gaspro	charm		average molecular weight of aerosol for diffusiophoresis
molwtv	gaspro	charm		average molecular weight of gas for diffusiophoresis
mupper	colpts	dataer,aerin, chmcol		mass of largest aerosol particle
ncaero	chmprm	aerin		number of components in aerosol module
nclad	vikprm	fissin		number of cladding rings
ncoll	colpts	dataer,aerin, chmcol	nsectn	number of collocation points
ndense(Ø:laero)	moment	dchem,charm chmsmo		component densities used in aerosol calculations
ndnmom	mominf	chmmom		number mean density of aerosol
ndnsln(laero)	lognrm	chmgln	ndngln	log-normal parameter for species number densities in aerosol source
ndump	vikprm	fissin		frequency at which results are written to graphics file
ne	chmprm	datchm,fissin		number of elements considered
nfelem	indcoe	dataer,aerin		type of finite element (2=linear)
nfuel	vikprm	fissin		number of fuel pellet rings
ngrf	vio	datgen		logical unit number used for graphics output
niv	vio	datgen		logical unit number used for printed output
njac	chmspc	reacts		ne*(ne+1)/2
njk(dim,dim)	coef	chmnor		normalisation factor to ensure conservation of mass in agglomeration term in aerosol calculation
nkbar(-2:dim)	indcoe	chmind		pointer used to calculate right-hand sides in aerosol calculations
nmprog	jobdet	datgen		program name
nouv	vio	datgen		logical unit number used for printed output
ns	chmprm	datchm		number of species
nsaero	chmprm	datchm,fissin		number of species in aerosol calculation
nse	addaer	aerin		nuber of aerosol sources
nsgas	chmprm	datchm		number of gas species, currently 89
nskip	vikprm	fissin		frequency at which results are written to printed output file
nstval	addaer	aerin		number of times for aerosol source interpolation
nsurf	vikprm	datgen		number of thermal surfaces per cell
ntape4	iotape	fissin		equal to niv
ntape5	iotape	fissin		equal to nttyv
ntape6	iotape	fissin		equal to nouv
nttyv	vio	datgen		logical unit number for output to terminal
origen		fissin	lfpinv	initial fission product concentration
pdense(dim)	aerslc	charm,chmaer		source particle mass density
pi	const	dataer,fissin		3.141592653
pi2	vikprm	datgen	pi	same as pi
pijk(3ØØ)	coef	chmcoe		values of intergral used in agglomeration of aerosol particles
press	gaspro	dchem,charm		gas pressure
ptemp(itemps)	inputs	fissin		gas pressures at each ptime
pthrmc(dim)	aerslc	chmaer		particle thermal conductivities at collocation points
ptime(itime)	inputs	fissin		times when gas pressures are given
qkn	aerslc	dataer,aerin		Knudsen-Weber coefficient q

Variable	Common	Set In	Alias	Description
r5Ømom	mominf	chmmom		mass median radius of aerosol
r5Øsln(laero)	lognrm	dataer,charm	r5Øgln	log-normal coefficien for mass median radius of aerosol species
rad5Ø(Ø:laero)	moment	chmsmo		mass median radius of each aerosol species
radius(dim)	colpts	chmrad		radius of aerosol particles at each collocation point
reynum	flow	chmflo		Reynolds number
rgasv	vikprm	datgen		ideal gas constant
rs(167)	chmspc	chemrs,fuelsl		species equilibrium constant
sigma(Ø:laero)	moment	chmsmo		cube root of geometric standard deviation with respect to mass for each component in aerosol
sigmom	mominf	chmmom		cube root of geometric standard deviation with respect to mass for aerosol
sigsln(laero)	lognrm	dataer,charm	siggln	cube root of geometric standard deviation with respect to mass for each component of aerosol source
spcies(167)	chmnam	datchm		symbols for each species in VICTORIA
spcing	colpts	dataer,aerin, chmcol		spacing between collocation points
specwt(167)	chmprm	fissin		species molecular weights
srcrat(dim,laero)	source	chmgln	zgln	aerosol source distribution at collocation points
stemp(itemps)	inputs	fissin		surface temperatures at each stime
stime(itime)	inputs	fissin		times when surface temperatures are given
stkeff	aerslc	dataer,aerin		sticking efficiency of aerosol
suml	chmspc	dchem,chemrs, gasslv,fuelsl		liquid molar concentration
tdelta	vikprm	fissin		time-step sise
te(25)	chmspc	gasslv,fuelsl		element abundances
temp	gaspro	dchem,charm		temperature of gas
tempc	cell	charm		ceiling temperature
tempw	cell	charm		wall temperature
tfail	inputs	fissin		temperature at which cladding fails
theta(16)	chmprm	datchm,fissin		exponential factor in binary diffusion function
thresh	vikprm	datgen		threshhold on number density of aerosol, 1Ø**12
time	times	charm		time step used in aerosol calculation, same as tdelta
ustar	flow	chmflo		scaled velocity used in friction calculation
vblthk	flow	dataer,aerin, chmflo,chmdep		momentum boundary layer thickness
vcgrcl	gaspro	charm		coefficient used in diffusiophoresis calculation
vcgrfl	gaspro	charm		coefficient used in diffusiophoresis calculation
vcgrwl	gaspro	charm		coefficient used in diffusiophoresis calculation
velos	gaspro	charm		mean flow speed
vftime(itime)	inputs	fissin		times when gas velocities are given
vgrav(dim)	vgrav	chmdep		gravitaional deposition velocity at collocation points
vlscm	vspace	datgen		memory allocation parameter that may be adjusted to accomodate probelm, currently 3ØØ,ØØØ
vmfrcl	gaspro	charm		coefficient used in diffusiophoresis calculation
vmfrfl	gaspro	charm		coefficient used in diffusiophoresis calculation
vmfrwl	gaspro	charm		coefficient used in diffusiophoresis calculation
volume	cell	charm		volume of cell

Variable	Common	Set In	Alias	Description
vrgas(jtemps)	inputs	fissin		radial velocity of gas at each vftime
vtime	vikprm	datgen,fissin, viki9Ø	tim	total elapsed time
vzgas(jtemps)	inputs	fissin		axial velocity of gas at each vftime
ws(167)	chmspc	dchem,chemws, gasslv,fuelsv		species molar concentration in current region
x(ivspac)	vspace			storage for main variables used throughout VICTORIA. aliases given under x pointers starting with letter l
xe(25)	chmspc	dchem,gasslv, fuelsv		gaseous element molar concentrations in current regions
zeta	tolers	dataer,aerin, cØ5che		determines when bisection is used to determine zero of function
zmom(dim)	mominf	chmzmo		moment of aerosol distribution at collocation points
zstore(dim+sizmas, Ø:laero)	distrb	dchem,chmdif, charm		aerosol distribution for each collocation point and component

C.4 References

C.1 B.L. Buzbee, The SLATEC Common Math Library, Sources and Development of Mathematical Software, W.Cowell, Ed., Prentice-Hall series in Computational Mathematics.

C.2 J.J. Dongarra, et al, LINPACK Users Guide, SIAM, (1979).

C.3 R.C. Singleton, Algorithm 347, An Efficient Algorithm for Sorting with Minimal Storage, CACM, 12(3),185-187, (1969).

C.4 L. F. Shampine and H. A. Watts, DEPAC - Design of a User Oriented Package of ODE Solvers, SAND79-2374, Sandia National Laboratories, Albuquerque, New Mexico (1979).

C.5 L.F. Shampine and H.A. Watts, Practical Solution of Ordinary Differential Equations by Ruge-Kutta Methods, SAND76-0585, Sandia National Laboratories, Albuquerque, New Mexico, (1976).

END

DATE FILMED

11 / 26 / 90