



KfK 4950
Dezember 1991

IVA3: Computer Code for Modelling of Transient Three Dimensional Three Phase Flow in Complicated Geometry

**Program Documentation:
Input Description**

**N. I. Kolev
Institut für Neutronenphysik und Reaktortechnik**

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE
Institut für Neutronenphysik und Reaktortechnik

KfK 4950

IVA3: COMPUTER CODE FOR MODELLING OF TRANSIENT THREE
DIMENSIONAL THREE PHASE FLOW IN COMPLICATED GEOMETRY

Program documentation: Input description

N. I. Kolev

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript gedruckt
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Abstract

This report describes the input and output of IVA3 computer code and the procedure how to compile, link, and run the code. The common blocs recorded for restarts files and post processing are described in detail as well as the IVA3 interface for thermodynamic and thermo physical properties. Some recommendations for the input preparation together with some detailed comments on some architectural and functional features of the code are given in order to give some insight of the caused actions by changing some control parameters.

IVA3: Rechenprogramm zur Modellierung der transienten Dreidimensionalen Dreiphasen
Strömung in komplizierter Geometrie

Programm Dokumentation: Eingabebeschreibung

Kurzfassung

Dieser Bericht beschreibt die Eingabe und die Ausgabe für das Computerprogramm IVA3 sowie die Prozedur wie man den Code übersetzt, linkt and ausführt. Die COMMON Blöcke, die für den Restart und für das Postprocessing notwendig sind, sind detailliert beschrieben. Das Interface für die thermophysikalischen und die Transportgrößen ist detailliert dokumentiert. Es wurden einige Empfehlungen zusammen mit einigen Details der Codearchitektur und funktionale Charakteristiken gegeben, um zu verdeutlichen, welchen Einfluß die Änderung einiger Eingabeparameter auf die Kontrolle der numerischen Simulation hat.

PREFACE

What IVA3 can do, and how it is done?

Our natural environment consists of air, water, steam and solid materials and the natural processes in this environment are flows consisting of air, water, steam and solid particles. Since the beginning of the human civilization mankind started to gather knowledge about these processes in order to use them. Using this knowledge machines are invented having as working materials flows consisting of some combinations of air, water, steam, and solid particles, or mixtures of all of them.

The computer simulation of physical problems is now widely accepted as the third investigative technique in science — alongside theory and experiment. The rapid sophistication of the computational technology makes possible the utilization of the existing knowledge in universal computational tools modeling flows in complicated geometries. IVA3 is one of the possible steps on the way to create a universal flow analyzer — computer code modeling of transient, multidimensional, multi-phase flows.

The framework used for the mathematical description of this flow mixture is *three fluid modeling*. Three fluid modeling means that *the flow is represented by means of mutually interacting abstract fluids* having their own temperatures, velocities and internal consistence. *Each of the fluids consists of two abstract components*, the first allowed to exert evaporation or condensation and the second — not exerting any changes of its aggregate state. The correspondence between the components of the abstract velocity fields and the physical phases like air, steam, water, microscopic and macroscopic solid particles is the link between the abstract mathematical formalism and the real flow systems as shown below:

Abstract mathematics	Real world
Velocity field Nr.1	Disperse or continuous gas
Inert component	Air
Non inert component	Steam
Velocity field Nr.2	Disperse or continuous liquid – solid mixture
Disperse inert component	Microscopic solid particles
Non inert component	Continuous or disperse water
Velocity field Nr.3	Disperse liquid – solid mixture
Disperse or continuous inert component	
– single phase	Micro– or macroscopic liquid metal or solid particles
– two phases	Macroscopic freezing metal
Non inert component	Water droplets

The abstract mathematical formalism used in IVA3 is universal for all flows consisting of three fields where each of the fields consists of two components.

The real physical world system in IVA3 is represented by equations of state and transport properties for

- (a) air,
- (b) water,
- (c) steam, and
- (d) one of the following materials

- (i) stainless steel,
- (ii) uranium dioxide,
- (iii) corium, mixture of 60% UO_2 , 16% ZrO_2 , 24% Stainless steel (67% Fe, 21% Cr, 12% Ni).

The last group of materials is in

- (a) liquid state, or in
- (b) liquid – solid equilibrium state, or in
- (d) solid disperse state.

The geometry in which the flow is modeled is three dimensional

- (a) cylindrical or
- (b) Cartesian.

VII

Arbitrary obstacles inside the flow region can be dynamically modelled by means of *dynamical surface permeabilities* which are defined either

- (a) as constants or
- (b) as prescribed functions of time.

Arbitrary structures not occupied by the fluid are described by means of *volumetric porosities of the space*.

The formalism automatically recognizes whether the flow in the local control volume is characterized as

- (a) channel flow – with strong wall – flow interaction, or as
- (b) pool flow – with no wall – flow interaction.

The consequences of such decisions are the automatic inclusion of appropriate flow pattern oriented constitutive models. The physical models used in IVA3 are flow pattern dependent. The code models in each particular cell one of the following 20 generalized flow patterns:

Table 1 Postulated flow regimes in IVA3

One velocity field only:

Velocity field 1 (gas) only.

Velocity field 2 (water) only.

Velocity field 3 (liquid metal) only.

Two velocity fields only:

Two phase bubble flow – continuous velocity field 2 and discrete velocity field 1 (gas).

Two phase slug flow – continuous velocity field 2 and discrete velocity field 1 (gas).

Two phase churn turbulent flow – continuous velocity field 2 and discrete velocity field 1 (gas).

Two phase dispersed flow – continuous velocity field 1 (gas), discrete velocity field 2 (droplets) for pool flow.

Annular flow – continuous velocity field 1 (gas) and continuous velocity field 2 (water film), identified only for flow in a confined geometry.

Dispersed velocity field 3 (liquid metal droplets or solid particles) in continuous velocity field 2 (water).

Dispersed velocity field 2 (water droplets) in the continuous velocity field 3 (liquid metal).

Continuous velocity field 1 (gas) and dispersed velocity field 3 (water droplets).

Continuous velocity field 1 (gas) and dispersed velocity field 3 (solid particles).

Dispersed velocity field 1 (gas bubbles) in continuous velocity field 3 (liquid metal).

Continuous velocity field 1 (gas) and dispersed velocity field 3 (solid particles or liquid metal droplets).

Three velocity fields:

Three phase bubble flow – dispersed velocity field 1 (gas bubbles), continuous velocity field 2 (water) and dispersed velocity field 3 (liquid metal or solid particles).

Continuous velocity field 1 (gas), dispersed velocity field 2 (water droplets) and dispersed velocity field 3 (solid particles or liquid metal droplets or porous velocity field 3) – for pool flow only.

Continuous velocity field 1 (gas), disperse velocity field 2 (water droplets) surrounded by continuous velocity field 3 (porous liquid metal with large particle diameters).

Continuous velocity field 1 (gas), continuous velocity field 2 (water) and discrete velocity field (3) (water droplets) – only for flow in confined geometry.

Continuous velocity field 1 (gas), continuous velocity field 2 (water) and discrete velocity field (3) (solid particles) – only for flow in confined geometry.

Disperse velocity field 1 (gas bubbles), continuous velocity field 2 (water) and continuous velocity field (3).

The *characteristic scale of the flow structure in each flow pattern is dynamically modelled*. It means that the modeling of the local gas – liquid jet, particles and bubbles fragmentation and/or coalescence is described with models taking into account

- (a) the finite time needed for each particular process and
- (b) the memory for the transported structure perturbation leading to changes in bubble or droplet sizes.

The dynamical model of the structure size ensures *continuous transition from disperse into continuous structure and vice versa*.

The *closure models* in IVA3 describe the following mechanical mass transfer between the fields in case of channel flow

- (a) droplet entrainment from the film attached at the wall and
- (b) droplet deposition into the film.

IVA3 provides closure models for heat, mass and momentum transfer mechanism divided into the following groups:

- (a) droplets,
- (b) bubbles,
- (c) films,
- (d) solid particles,
- (e) heated surfaces.

If the field already exists IVA3 addresses for the first three group (a, b, c) one of

the following special heat and mass transfer models:

- (a) convective heat transfer,
- (b) flushing,
- (c) evaporation into two component atmosphere,
- (d) condensation of pure steam,
- (e) condensation from steam air mixtures.

The smooth origination and disappearance of velocity fields in each cell is automatically ensured by careful mathematical treatment of the discretized equations.

The origination of the velocity fields can be

- (a) due to kinetic reasons like

- (i) evaporation or
- (ii) condensation, or

- (b) due to macroscopic transport like

- (i) convection from the neighboring cells,
- (ii) convection from regions outside of the computational region (external sources),
- (iii) entrainment and
- (iv) deposition in case of channel flow.

Thermal interaction between flow and external sources is modeled too in IVA3.

IVA3 has an nuclear reactor core model in distributed parameter. The energy release is user defined as function of space and time. The 2D heat conduction in nuclear reactor rods is modeled in interaction with the flow model. Flow regimes oriented heat transfer based on local parameters is described by means of the complete boiling curve having the following regimes

- (a) convective heat transfer liquid – wall or gas – wall,
- (b) sub-cooled nucleate boiling,
- (c) saturated nucleate boiling in pools or convective nucleate boiling in channels,
- (d) critical heat flux predictions by different methods,
- (e) transition boiling,
- (f) film boiling in pools or convective film boiling in channels predicted by different methods.

The main characteristics of the mathematical methods are given below. IVA3 presents an integral of a system of 21 partial differential equation in 3D space with a variety of initial and boundary conditions. The equations are

- (a) 3 equations for conservation of mass for each velocity field,
- (b) 3 equations for conservation of the inert component in each velocity field,
- (c) 9 momentum equations containing the viscous tensor, drag and virtual forces, and forces caused by the mass transfer,
- (d) 3 energy conservation equation for each velocity field transformed into the entropy form,
- (e) 3 conservation equation for the particle number density transport in each velocity fields (number of droplets or bubbles per unit volume of the flow).

The space discretization is either

- (a) equidistant or
- (b) non equidistant in each direction.

A staggered mesh is used for the discretization of the momentum equations.

The method is fully implicit and unconditionally stable.

The convective terms of all conservation equations are discretized by using the *first order donor - cell concept*.

The viscous terms in the momentum equations are discretized by using *second order central differencing*.

The main characteristic of the IVA3 solution concept is the *analytical inclusion of the influence of the change of the dependent variables on the pressure field in one single step*. It makes the computation extremely stable and very fast.

The algebraic system representing the discretized Poisson like equation is solved by means of four optional user defined methods

- (a) plane by plane - rectangles,
- (b) plane by plane - cylinders (in cylinder coordinates) or rectangles (in Cartesian coordinates),
- (c) plane by plane - circles and
- (d) line by line.

Automatic time step control dictates time step changes in order to perform the integration within the user prescribed accuracy. Automatic time step optimization control the time step to have the maximum possible value for optimum integration speed.

Some programming characteristics of IVA3 are given below:

Programming language	FORTRAN 77
Program units:	158
Program steps:	≈ 8 700
Program lines:	≈ 18 000
Computer:	IBM 3090
Memory requirement:	≈ 1 200 K + 6.2 x (number of the computational cells)

The division of the model in 158 functional program units makes possible model improvements exchanging constitutive correlations with new ones or adding new models for the description of a particular problem.

IVA3 is a result of long years code development and contains the present day experience and know-how of multi fluid modeling. The preceding codes belonging to this series are the one dimensional codes

SONJA 1 (1977),
 SONJA 2 (1979),
 RALIZA 1 to 3 (1980-1983),
 IVA1 (1984),

and the three dimensional code

IVA2 (1985).

All presently existing concepts like homogeneous equilibrium, homogeneous non equilibrium, non homogeneous non equilibrium, one-, two-, and three component flows, drift flux modeling, two and three fluid modeling, different kinds of thermodynamic description, different mathematical methods etc. are tested and the IVA3 method is reached as the simplest but the most effective multi component flow description among them.

Next we give a short description of successful tests performed with IVA3:

Single – phase flows:

- 1) Shock tube problem – 1D pressure wave propagation in gas, comparison with an analytical solution;
 - 2) Gas jet expansion – 2D pressure wave propagation in gas, comparison with experimental data;
-

Two – phase flows:

- 3) Gas jet expansions in a liquid with and without internals – 2D pressure and volumetric concentration wave propagation of a two phase mixture in a complicated and a simple geometry, comparison with experimental data of KfK;
 - 4) Dam break release of liquid – water over flooding of a 2D region, comparison with experimental data of KfK;
 - 5) Dam break release of liquid – water over flooding of 2D region, comparison with independently obtained results with other computer codes for 2D modeling – AFDM;
 - 6) Comparison of the prediction of drag forces model for air – water – bubble flow, water – solid particles flow in vertical channel with Sakaguchi's experimental data for steady state flow;
-

Three phase flows:

- 7) Comparison of the prediction of drag forces models for air – water – solid particles bubble flow in vertical channel with Sakaguchi's experimental data for steady state flow;
- 8) Abolfadl and Theofanous benchmark – three fluid flow in 2D geometry without internals, hot solid particles, water and steam flow, comparison with independently obtained results with other computer code for 2D modeling PM-ALPHA;
- 9) Four three – fluid tests in 2D geometry with complex internals typical for a pressurized water cooled nuclear reactor PWR – 1300, interaction of hot solid particles with water;
- 10) Low pressure molted corium – water interaction in a deep pool, liquid metal, water and steam flow in a complicated 3D geometry, fragmentation, coalescence, pressure peaks, comparison with experiment;

- 11) High pressure molted corium – water interaction in a deep pool, liquid metal, water and steam flow in a 2D geometry, fragmentation, coalescence;
 - 12) Molted corium water interaction in a nuclear reactor, liquid metal, water and steam flow in a complicated 2D PWR 1300 geometry, fragmentation, coalescence, pressure peaks;
 - 13) 3D pressure wave propagation in case of high pressure pipe break inside a heat exchanger;
 - 14) Disintegration of the bottom of the nuclear reactor vessel – estimation of the forces acting on the construction, very complicated 2D geometry, integral analysis of pressure wave propagation in reactor vessel and containment;
 - 15) Steady state and transient processes in a 3D nuclear reactor.
-

Further application of the code in different fields in science and technology will surely need improvement of some constitutive models or replacement of some of the existing by new ones. Nevertheless, the results obtained with IVA3 are very encouraging and show that the code possesses a meaningful code architecture and organization, and allows solution of versatile and very complicated tasks of practical interest with reasonable computation costs and stability not only in the nuclear safety analysis.

IVA3 may be interesting for theoreticians and experimentalists investigating new interfacial transfer models or gaining new experimental data for the verification of such models. The impact of improving interfacial models on the system behavior of complicated flows can be naturally investigated using IVA3.

N. I. Kolev.

CONTENTS		Page
1.	Input description	1
1.1	Problem dimensions, parameters	1
1.2	Program input – restart	4
1.3	Program input – start	8
1.3.1	Geometry	10
1.3.2	Initial conditions	18
1.3.3	Boundary conditions	24
1.3.3.1	Dependent variables	24
1.3.3.2	Variable geometry	28
1.3.3.3	External mass sources	30
2.	Check list for preparing IVA3 load module	32
3.	Output description	34
4.	Common blocks	37
4.1	Alphabetic list of the common blocks	37
4.2	Common blocks stored in restart files	41
5.	Test example	42
Appendix 1:	JCL's and input data for start and restart of IVA3 for the example discussed in Chapter 5.	43
Appendix 2:	Summary of the figures discussed in Chapter 5	46
Appendix 3:	IVA3 interface for thermodynamic and thermo –physical properties.	51

1. INPUT DESCRIPTION

1.1 PROBLEM DIMENSIONS, PARAMETERS

The dimensions of the vectors and arrays used in IVA3 are defined by means of the following parameter statement

```
PARAMETER (LM=3,IM= ,JM= ,KM= )
```

in the main program

```
MAIN IVA3
```

and in the following, limited number of subroutines

```
I2READ,  
INTGRA,  
WRESTA,  
RRESTA,
```

where

IM is the number of the real "radial" cells,
JM is the number of the real "angular" cells,
KM is the number of the real axial cells.

In case of Cartesian coordinates radial is equivalent to the x-axis and angular to the y-axis. In both cases z is the vertical axis.

All other dimensions in the code are derived from this three parameters, e.g.

```
PARAMETER (IMP1=IM+1,JMP1=JM+1,KMP1=KM+1)  
PARAMETER (IMP2=IM+2,JMP2=JM+2,KMP2=KM+2)  
PARAMETER (IMM1=IM-1,JMM1=JM-1,KMM1=KM-1).
```

According to the problem to be solved the parameter statements should be changed in all five routines mentioned above and a new load module should be produced before starting an analysis.

The code automatically checks the correctness of the parameters comparing them with the input values of IMX, JMX and KMX and if they disagree stops the program execution with STOP 10 and the following message

```
"Change the first parameter statement as follows  
PARAMETER (LM=3,IM=...,JM=...,KM=...)  
compile, link IVA3 and start again".
```

Additionally the solver of the linear algebraic equations needs space

```
DIMENSION AM(IAM,JAM),BM(IAM)
```

defined also with a parameter statement in the

2 IVA3 Computer Code. Input description...

MAIN IVA3.

The form of this parameter statement depends on which of the four available options should be chosen for the actual analysis for the solution of the algebraic pressure – velocity problem. The choice of the method is controlled by the integer input parameter LVARIA. The following convention is used:

(a) Plane by plane in cylinder form (LVARIA=1)

PARAMETER (IAM=KM*JM,JAM=2*JM+1),

or

(b) plane by plane in circle form (LVARIA=2)

PARAMETER (IAM=IM*JM,JAM=2*JM+1),

or

(c) plane by plane in rectangle form (LVARIA=3)

PARAMETER (IAM=KM*IM,JAM=2*IM+1),

or

(d) line by line (LVARIA=4)

PARAMETER (IAM=KM,JAM=3).

The code automatically checks the correctness of the parameters comparing them with the appropriate expressions computed with the previous parameter values for IM, JM and KM, and if they disagree stops the program execution with STOP 11 and the following message

"Non consistent dimensions of the problem with the dimensions of the AM and BM fields. Change the parameter statements as follows

PARAMETER (IAM=...,JAM=...)".

One should perform the appropriate changes and repeat the same procedure.

The storage for the distributed temperature field in the reactor fuel rods is reserved by means of

PARAMETER (N=7,IMMM=x),

where

$x = \max(1, IM-1)$.

This parameter statement should be properly changed in the main program

MAIN IVA3

and in the subroutines

I2READ

INTGRA
WRESTA
RRESTA.

Next some examples of the memory requirement for IVA3 execution are given

4	points 1116 K,
20	points 1192 K,
100	points 1692 K,
1000	points 7224 K.

IVA3 is provided with three different groups of equations of state for (a) uranium dioxide, (b) corium, and (c) stainless steel. At least one of them must be included to create a operational load module.

1.2 PROGRAM INPUT – RESTART

The structure of the input description is the usual one. First the number of the record and the position of the variable is indicated followed by the name of the variable, its format in the code, dimension and the corresponding explanation.

1/1	LRREST	L5	—	=T A restart file should be used as input. =F No restart file should be used as input.
-----	--------	----	---	---

In case LRREST=T the subroutine RRESTA is called in order to read a previously prepared restart file. Thereafter the following input statements are required:

Logical control variables:

2/1	LRREST	L5	—	The same as 1/1
2/2	LWREST	L5	—	=T A restart file should be recorded after the run exceeds the prescribed CPU time minus DTAU0/100. =F No restart file required.
2/3	LPLOT	L5	—	=T The velocities for two vertical planes should be recorded. =F No velocities should be recorded.
2/4	LSTAZ	L5	—	=T The heat conduction problem in the nuclear reactor fuel elements is considered during all simulation as quasi steady state. Time saving if the end purpose of the simulation is reaching a steady state process. =F Transient heat conduction simulation is required.
2/5	IPRINT	I5	—	Printing frequency.
2/6	IPLOTI	I5	—	Plot frequency for the velocity vectors.

Parameters controlling the accuracy of the computation.

3/1	DCONVV	E12.5	m/s	1.E-5
3/2	DCONVP	E12.5	Pa	

During the solution of the pressure – velocity algebraic problem the outer iterations are repeated as much as necessary to reach an absolute value of the maximum pressure change in the integration region from iteration to iteration less than DCONVP. If this was not achieved after a prescribed number of iterations the time step is reduced and the integration starts again. Together with DCONVR these are the most important parameters controlling the accuracy which are prescribed by the user.

Note: Decreasing the prescribed value of the tolerated maximum pressure error

increases the computer costs. While some problems allow an approximate integration in order to obtain not very accurate but fast solutions, there are others, e.g. strong transients with severe parameter excursions, that require a high accuracy in order to reach a solution at all.

3/3	DCONXS	E12.5	-	1.E-5
3/4	DCONXN	E12.5	-	1.E-5
3/5	DCONVS	E12.5	J/(kgK)	1.E-5
3/6	DCONVR	E12.5	-	< 0.05

For each computational cell the mass conservation is controlled in the following way: The mixture density in the new time plane is computed by means of the result of the integration. The same is repeated by using the mixture mass conservation equation. The difference of the both results divided by the density in the old time plane is controlled. If the so computed relative mixture density error is greater than DCONVR the time step is reduced and the integration is started again from the old time plane. In spite of the fact that IVA3 the method can operate tolerating a large error it is recommended to limit this requirement at least to 0.05.

4/1	TAUEND	E12.5	s	Duration of the physical process which has to be simulated. Once the physical time TAU exceeds TAUEND a restart file is written and the computation is regularly finished with STOP 3.
4/2	DTAU0	E12.5	s/100	

Writing the restart files consumes computer time. That is why it is necessary to provide enough time until the end of the run for writing the restart file if required. A restart file should be recorded after the run exceeds the prescribed CPU time minus DTAU0/100 minus the CPU time needed for the previous time step. Normally the successive time steps have comparable performance and the restart file is written successfully. Depending on the transients it may happen that the last time step takes much more CPU time than the previous one and the restart file cannot be successfully written. In this case increase DTAU0 with a value larger than the last time step and repeat the run.

4/3	DTAU	E12.5	s	Proposed time step.
4/4	LVARIA	I6	-	

Controls the method of solution of the algebraic problem pressure – velocity.

LVARIA=1,2,3 means that the pressure velocity problem is reduced to a Poisson like equation which is solved by successive iteration using direct inversion of the coefficients matrix plane by plane.

For LVARIA=1 the planes have cylinder form (I=const).

For LVARIA=2 the planes have circle form (K=const).

For LVARIA=3 the planes have rectangle form (J=const).

6 IVA3 Computer Code. Input description...

LVARIA=4 means solution of the pressure equation by means of line by line iterations (for 1D flow the most effective method).

5/1	DTAUAU L6	-	=T	Automatic time step control is chosen. The main idea on which this options relies is the prediction of the time step so as to obtain some expected maximum pressure increase DPTAUS in the computational region. The actual maximum of the pressure increase after one integration step is compared with the value DPTAUI and if greater the time step is reduced and a new computation is started.
			=F	In the present version only DTAUUAU=F is allowed for post factum accuracy control and time step prediction based on the accumulated experience with the previous time steps.

5/2	DPTAUS E12.5	Pa
-----	--------------	----

5/3	DPTAUI E12.5	Pa
-----	--------------	----

The time step is also limited by the actual values of DTMAXI and DPMAXI assigned in the main program. DTMAXI and DPMAXI limit the maximum temperature and pressure change for a single time step. This limitation is dictated by the linearization of the strong nonlinear state equations for one single step. The following values are used successfully by the author for different applications

DTMAXI=10 through 100,
DPMAXI=1.E5 through 5.E5.

5/4	ICHF I6	-	Control the choice of the critical heat flux correlation:
		=1	Smolin.
		=2	Biasi.
		=3	Bowring-72.
5/5	IPCHT I6	-	Control the choice of the post critical heat transfer correlation.
		=1	Berenson.
		=4	Groeneveld - pipes and annuli.
		=5	Groeneveld - annuli.
		=6	Groeneveld - pipes.
		=7	Averaged value from Groeneveld annuli and Miropolski.

=8 Miropolski.

5/6 ISLIP0 I6 - =1 This parameter is reserved in case of modifying IVA3 for using the drift flux correlation for channel flow in order to compute the interfacial drag coefficients. This option is under testing.

5/7 LTAU00 L6 -

If LTAU00=T the actual process time is set to zero, TAU=0.0000000. This option is appropriate if one uses some restart file resulting from previous analysis, e.g. steady state for a complicated geometry, as initial condition for a transient process.

If LTAU00=F no change of the process time TAU as recorded in the restart file occurs. This is the standard options for successive restarts by analyzing one transient process.

5/8 LPTU1 L6 - =T The irreversible turbulent viscous dissipation term in the energy balance of the gas phase should be taken into account.

5/9 LPTU2 L6 - =T The irreversible turbulent viscous dissipation term in the energy balance of the liquid phase (velocity field 2) should be taken into account.

5/10 LPTU3 L6 - =T The irreversible turbulent viscous dissipation term in the energy balance of the liquid phase (velocity field 3) should be taken into account.

Not tacking into account of the irreversible turbulent viscous dissipation term in the energy balance in processes where the energy source terms due to other phenomena are considerable higher saves computer time and does not influence the adequacy of the modelling.

Since the restart procedure is required the initial geometry and the initial conditions are not required. They are already prepared and stored in the IVA3 common blocks in the previous runs. Only the boundary conditions, geometry changes and sources are required as given with statements 59 through 98 in Chapter 1.3.3.

1.3 PROGRAM INPUT – START

In case LRREST=F the subroutine RRESTA is not called in order to read a previously prepared restart file. This is the standard option to start for a first time an new analysis. The records 2 and 3 are unchanged. Only the records 4 and 5 are changed as follows:

2/1	LRREST	L5	—	
2/2	LWREST	L5	—	
2/3	LPLOT	L5	—	
2/4	LSTAZ	L5	—	
2/5	IPRINT	I5	—	
2/6	IPLOTI	I5	—	
3/1	DCONVV	E12.5	m/s	
3/2	DCONVP	E12.5	Pa	
3/3	DCONXS	E12.5	—	
3/4	DCONXN	E12.5	—	
3/5	DCONVS	E12.5	J/(kgK)	
3/6	DCONVR	E12.5	—	
4/1	TAUEND	E12.5		
4/2	DTAU0	E12.5		
4/4	TAU	E12.5	s	Process time.
4/5	DTAU	E12.5	s	< 0.000001. Time step. The automatic time step control starts with small time step at the very beginning of the computation and adjusts the time step necessary
				(a) to satisfy the prescribed accuracy requirements, and
				(b) to find the maximum computational speed to perform the computation cost effective.
				The first requirement is called accuracy control and the second time step optimization.
4/6	LVARIA	I6		
5/1	DTAUAU	L6		
5/2	DPTAUS	E12.5		
5/3	DPTAUI	E12.5		
5/4	ICHF	I6		
5/5	IPCHT	I6		
5/6	ISLIP0	I6		
5/7	LZYLIN	L6	—	=T Cylinder coordinates. =F Cartesian coordinates.
5/8	LPTU1	L6		
5/9	LPTU2	L6		
5/10	LPTU3	L6		

Further the following records are required:

6/1 (TITLE(I),I=1,20) 20A4 Title of the run.

The following 6 relaxation parameters should be set to 1. They are reserved for construction of successive relaxations methods. Only OMEGAP influences the iterative solution of the pressure equation in the present IVA3 version. As long as no better strategy for the IVA3 SOR is implemented this factor should be set to 1.

7/1	OMEGAV	E12.5	-	=1.
7/2	OMEGAP	E12.5	-	=1.
7/3	OMEGXS	E12.5	-	=1.
7/4	OMEGXN	E12.5	-	=1.
7/5	OMEGAS	E12.5	-	=1.
7/6	OMEGAR	E12.5	-	=1.

7/7	ISORMX	I4	-	Number of the block iterations during the solution of the pressure equation.
-----	--------	----	---	--

ISORMX=1 if line by line solution method for only one line is selected (1D flow) because the solution is obtained analytically and no iterations are needed. Similar is the situation if the plane by plane method is selected and the computational region consists of only one plane which coincides with the solution plane (2D). In this case ISORMX=1 because the pressures for all the cells belonging to the plane are obtained analytically and no iterations are needed. In any other case ISORMX > 1, e.g. 5 to 10. During the solution of the pressure equation the internal iterations are interrupted if the maximum pressure change from iteration to iteration is less than 0.0001 Pa. The reason not to set ISORMX a large number is the following. It is not necessary to solve exactly the pressure equation at the initial outer macro iterations of the method. The repetition of the outer iteration itself reduces also the errors in the pressure field.

7/8	ICOAMX	I4	-	=1	Reserved for possible implementation of coarse mesh rebalancing method.
-----	--------	----	---	----	---

At this place of the execution the main read program I2READ is called to read input and perform some additional data processing before starting the calculation. The following logical structure of the data is used:

- (a) geometry,
- (b) initial conditions, and
- (c) boundary conditions.

8/1	IMX	I5	-	Number of the real "radial" cells.
8/2	JMX	I5	-	Number of the real "angular" cells.
8/3	KMX	I5	-	Number of the real axial cells.
8/4	EQGRID	L5	-	=T Equidistant computational network is required.
			-	=F Non equidistant computational network is required

1.3.1 GEOMETRY

Before starting with the input defining the geometry we give some brief information about the way how the geometry is defined in IVA3.

The computational region is divided in $IM*JM*KM$ real computational cells. The computational region is surrounded by one layer of fictitious computational cells.

In general the boundaries of each real computational cell are defined with $RH, THH,$ and ZH having dimensions of

$$\begin{aligned} &RH (IM+1), \\ &THH(JM+1), \\ &ZH (KM+1), \end{aligned}$$

respectively. The sizes of the computational cells are $DRH, DTHH$ and DZH having dimensions

$$\begin{aligned} &DRH (IM+1), \\ &DTHH(JM+1), \\ &DZH (KM+1), \end{aligned}$$

respectively. The discrete coordinates of the centers of each computational cell are R, TH and Z having dimensions

$$\begin{aligned} &R (IM+2), \\ &TH(JM+2), \\ &Z (KM+2), \end{aligned}$$

respectively. The distance between them is DR, DTH and DZ having dimensions

$$\begin{aligned} &DR (IM+2), \\ &DTH(JM+2), \\ &DZ (KM+2), \end{aligned}$$

respectively. The computational cells are surrounded by surface permeabilities $GARE, GATHN$ and $GAZT$, having dimensions

$$\begin{aligned} &GARE (IM+2, JM+2, KM+2), \\ &GATHN(IM+2, JM+2, KM+2), \\ &GAZT (IM+2, JM+2, KM+2), \end{aligned}$$

respectively. The end characters stay for the orientation of each surface with respect to the center of the computational cell: E for east, N for north, and T for top, respectively. The surface permeabilities are defined as the ratio of the cross section available for the flow to the total surface of the computational cell in the corresponding plane.

Each computational cell is considered as occupied by flow and structure. The ratio of the volume available for the flow to the total volume of the computational cell is called volume porosity, GAV . The volume porosity has dimensions

$$GAV(IM+2, JM+2, KM+2).$$

Except for the boundary surfaces of the computational space the values for the surface permeabilities and volumetric porosities are initially set to one, which means no structural obstacles for the flow. The surface permeabilities at the boundaries of the computational regions are set to zero ($I=1,IM+1,J=1, JM+1,K=1, KM+1$), which means non permeable boundaries. The same is valid for the most other fictitious surfaces ($I=IM+2,J=JM+2,K=KM+2$). The user can replace these values by preparing appropriate input.

This way of defining the geometry successfully describes a complicated technical system. Moreover, the surface permeabilities in IVA3 are defined as time functions. So variable geometry problems can be analyzed too.

IVA3 describes the flow in a confined geometry with arbitrary internals as well pool flow. This feature needs the definition of space oriented hydraulic diameters. On each surface where a surface permeability is defined a corresponding hydraulic diameter is defined too. The hydraulic diameters DHYR, DHYTH and DHYZ have dimensions

$$\begin{aligned} & \text{DHYR (IM+2, JM+2, KM+2),} \\ & \text{DHYTH (IM+2, JM+2, KM+2),} \\ & \text{DHYZ (IM+2, JM+2, KM+2),} \end{aligned}$$

respectively. The default value of the hydraulic diameters is 1 m, which in fact did not influence the viscous flow in a fine mesh at all. If the flow is modelled by means of coarse mesh the viscous terms in the momentum equations did not influence the flow at all too. Therefore one can use the actual direction oriented hydraulic diameters to model the friction forces in porous structures by means of coarse mesh or use a fine mesh to model the viscous effects excluding the influence of the wall friction terms by using the default value of the hydraulic diameters which is very large.

In case of working with an equidistant computational network, EQGRID=T, the geometry input is reduced to the following:

9/1	RH(1)	E12.5	m	Radial coordinate at the left computational boundary.
9/2	RH(IMP1)	E12.5	m	Radial coordinate of the right computational boundary.
9/3	THH(1)	E12.5	rad (m)	Angular coordinate of east computational boundary.
9/4	THH(JMP1)	E12.5	rad (m)	Angular coordinate of the west computational boundary.
9/5	ZH(1)	E12.5	m	Axial coordinate of the top computational boundary.
9/6	ZH(KMP1)	E12.5	m	Axial coordinate of the bottom computational boundary.

In case of using a non equidistant computational network, EQGRID=F, the geometry input is:

Cell boundaries:

12 IVA3 Computer Code. Input description...

9/1	(RH (I),I=1,IMP1)	8F10.5	m	Radial coordinates.
9/2	(THH(J),J=1,JMP1)	8F10.5	rad (m)	Angular coordinates.
9/3	(ZH (K),K=1,KMP1)	8F10.5	m	Axial coordinates.

From this information IVA3 computes the coordinates of the cell centers, the sizes of the cells and the distance between the cell centers. In case of cylindrical coordinates, LCYLIN=T, the R values are set by IVA3 to one, and the computation uses DR and DRH values in the numerical method.

10/1	RAUZ	F10.7	m	Roughness of the structures.
10/2	DFB	F10.7	m	Default value of the solid particles diameter in the second velocity fields.
10/3	DTB	F10.7	m	Default value of the solid particles diameter in the third velocity field.

Reading permeabilities plane by plane:

11/1	NRFLR	I5		The information on the next statement is NRFLR times required (.GT.0).
------	-------	----	--	--

12/1	I	I4	—
12/2	J1	I4	—
12/3	J2	I4	—
12/4	K1	I4	—
12/5	K2	I4	—
12/6	GARE1	F12.4	—

The cylinder surface defined with I=const and limited by (J1,J2) and (K1,K2) has uniform radial surface permeabilities equal to GARE1.

13/1	NRFLTH	I5		The information on the next statement is NRFLTH times required (.GT.0).
------	--------	----	--	---

14/1	J	I4	—
14/2	I1	I4	—
14/3	I2	I4	—
14/4	K1	I4	—
14/5	K2	I4	—
14/6	GATHN1	F12.4	—

The rectangular surface defined with J=const and limited by (I1,I2) and (K1,K2) has uniform angular surface permeabilities equal to GATHN1.

15/1	NRFLZ	I5		The information on the next statement is NRFLZ times required (.GT.0).
------	-------	----	--	--

16/1	K	I4	—
16/2	I1	I4	—
16/3	I2	I4	—
16/4	J1	I4	—

16/5	J2	I4	—	
16/6	GAZT1	F12.4	—	The surface inside the circle segment defined with $K=const$ and limited by $(I1,I2)$ and $(J1,J2)$ has uniform axial surface permeabilities equal to GAZT1.

Reading volumetric porosities for volume regions:

17/1	NRZONG	I5		The information on the next statement is NRZONG times required (.GT.0).
18/1	I1	I5	—	
18/2	I2	I5	—	
18/3	J1	I5	—	
18/4	J2	I5	—	
18/5	K1	I5	—	
18/6	K2	I5	—	
18/7	GAV1	F12.4	—	For the region containing the cells with integer coordinates limited by $(I1,I2)$, $(J1,J2)$ and $(K1,K2)$ the volumetric porosities are uniform and equal to GAV1.

At this place of the execution IVA3 postulates for the geometry the cyclic condition as follows: The code sets the values of the surface permeabilities and the volumetric porosities for the vertical layer of fictitious cells defined with $J=JM+2$ equal to the values for the real vertical layer of cells defined with $J=2$. The same happens for the values of the permeabilities and porosities of the fictitious vertical layer defined with $J=1$, which obtains the corresponding values from the real vertical layer of cells defined with $J=JM+1$.

Reading hydraulic diameters.

19/1	NRZOND	I5		The information on the next statement is NRZOND times required (.GE.0).
20/1	I1	I5	—	
20/2	I2	I5	—	
20/3	J1	I5	—	
20/4	J2	I5	—	
20/5	K1	I5	—	
20/6	K2	I5	—	
20/7	DHYRE1	F5.3	m	
20/8	DHYTH1	F5.3	m	
20/9	DHYZT1	F5.3	m	For the region containing the cells with integer coordinates limited by $(I1,I2)$, $(J1,J2)$ and $(K1,K2)$ the radial, angular and axial hydraulic diameters at the increasing integer number side of the cells are uniform and equal to DHYRE1, DHYTH1 and DHYZT1, respectively. Note, hydraulic diameters are defined at the same cell surfaces where the velocity and the surface permeabilities are defined and follow the same convention of index organization.

14 IVA3 Computer Code. Input description...

Additional reading of hydraulic diameters: point by point.

21/1	NRPUNK	I5		The information on the next statement is NRPUNK times required (.GE.0).
22/1	I	I6	—	
22/2	J	I6	—	
22/3	K	I6	—	
22/4	DHYR (I,J,K)	E12.5	m	
22/5	DHYTH(I,J,K)	E12.5	m	
22/6	DHYZ (I,J,K)	E12.5	m	For the cell with integer coordinates (I,J,K) the radial, angular and axial hydraulic diameters at the positive side of the cells are equal to DHYR (I,J,K), DHYTH(I,J,K) and DHYZ (I,J,K), respectively.

Reading information for a nuclear reactor core if required.

23/1	SZTRUE	L5	=T	Nuclear reactor core will be simulated. (The discretization in this case only in the angular direction is restricted to equidistant (use EQGRID=F). This limitation can be easily removed by modifying the statements computing the number of the fuels in a cell belonging to the core in I2READ).
			=F	Nuclear reactor core will not be simulated.
23/2	WIRE	L5	=T	Wire wrap spacer.
			=F	No wire wrap spacer.

The following statements 24 through are required only if nuclear reactor core should be simulated, SZTRUE=T.

The default values for all heat sources and the heated diameters in the core are zero.

24/1	ICORE1	I5	—	
24/2	ICORE2	I5	—	
24/3	JCORE1	I5	—	
24/4	JCORE2	I5	—	
24/5	KCORE1	I5	—	
24/6	KCORE2	I5	—	A nuclear reactor core is defined in the computational cells having integer coordinates within the region (ICORE1, ICORE2), (JCORE1, JCORE2) and (KCORE1, KCORE2).
24/7	DBE	F10.5	m	Outer cladding diameter.
24/8	STEPBE	F10.5	m	Pitch of the equidistant triangular fuel rod array arrangement.
24/9	DELTAH	F10.5	m	Cladding thickness.
24/10	DBS	F10.5	m	Fuel (pellet) diameter.
25/1	NRBE	I10	—	Number of the rods in the core.
25/2	QREACT	E10.0	W	Reactor thermal power.

Using the above information the code computes the averaged power per unit fuel volume as follows

$$QBS3M = QREACT / ((FLOAT(NRBE) * 3.1415 * DBS ** 2 / 4. * (ZH(KCORE2) - ZH(KCORE1 - 1))))$$

Next the number of the fuel rods in the core cells, NRBEIJ(I,J), is computed by the assumption of triangular grid with pitch equal to STEPBE. If ICORE2 is equal ICORE1 all fuel rods are assumed to be in the one vertical column of computational cells limited in axial direction by KCORE1 and KCORE2. If ICORE2 is greater than ICORE1 the other core layer of computational cells, defined by ICORE2, is assumed to be a bypass without fuel rods.

After computing the number of the fuel rods in the core cells the code computes automatically the surface permeabilities, the hydraulic and the heated diameters in the vertical directions, and the volumetric porosities for the core region. For some changes the following statements are used.

Heated diameters in volume regions.

26/1	NRZONG	I5		The information on the next statement is NRZONG times required (.GE.0).
27/1	I1	I4	—	
27/2	I2	I4	—	
27/3	J1	I4	—	
27/4	J2	I4	—	
27/5	K1	I4	—	
27/6	K2	I4	—	
27/7	DHEAT	F12.4	m	For the region containing the cells with integer coordinates limited by (I1,I2), (J1,J2) and (K1,K2) the heated diameters are uniform and equal to DHEAT.

Heated diameters: point by point.

28/1	NRPUNK		I5	The information on the next statement is NRPUNK times required (.GE.0).
29/1	I		I6	—
29/2	J		I6	—
29/3	K		I6	—
29/4	DHEAT(I,J,K)		F12.5 m	For the cell with integer coordinates (I,J,K) the heated diameters is DHEAT(I,J,K).

The next statement is needed only if WIRE=T.

30/1	DWIRE	E12.5	m	Wire wrap diameter.
30/2	WIREP	E12.5	m	Wire wrap pitch.

The above statement is reserved for future improvement of the friction models in fast breeder reactors.

The space variation of the generated thermal power is defined by means of the next

statements. Note that the thus defined space form is conserved during the transients.

31/1	(FACR (I), I=1,ICM)	(16F5.3)	—	Radial distribution factors.
32/1	(FACTH(J), J=1,JCM)	(16F5.3)	—	Angular distribution factors.
33/1	(FACZ (K), K=1,KCM)	(16F5.3)	—	Axial distribution factors.

The integer counter increase in the positive direction. The numbers of the points approximating the factors are

```

ICM=ICORE2-ICORE1+1,
JCM=JCORE2-JCORE1+1,
KCM=KCORE2-KCORE1+1.

```

for each direction respectively. The thermal power per unit volume of the fuel, QP3, and per unit volume of the fluid, QP3KM, are computed as follows:

```
FACT=1.E+00
```

```

ICOR21=ICORE2-1
IF(ICORE2.EQ.ICORE1) ICOR21=ICORE2
DO 159 I=ICORE1,ICOR21
DO 159 J=JCORE1,JCORE2
DO 159 K=KCORE1,KCORE2

```

```

IC=I-ICORE1+1
JC=J-JCORE1+1
KC=K-KCORE1+1

```

```

159 QP3 (I,J,K)=QBS3M*FACR(IC)*FACTH(JC)*FACZ(KC)*FAKT
QP3KM(I,J,K)=QP3(I,J,K)*DBS**2/(DBE*DHEAT(I,J,K))

```

A final possibility is provided in IVA3 to correct additionally some surface permeabilities and porosities point by point.

34/1	NRPUNK	I5		The information on the next statement is NRPUNK times required (.GE.0).
35/1	I	I5	—	
35/2	J	I5	—	
35/3	K	I5	—	
35/4	GARE (I,J,K)	F12.4	—	
33/5	GATHN(I,J,K)	F12.4	—	
33/6	GAZT (I,J,K)	F14.4	—	
33/7	GAV (I,J,K)	F12.4	—	

For the cell with integer coordinates (I,J,K) the radial, angular and axial permeabilities at the positive side of the cells and the volumetric porosity are equal to GARE(I,J,K), GATHN(I,J,K), GAZT(I,J,K) and GAV(I,J,K), respectively.

After having defined the geometry the code computes the frequently used geometrical BETA — groups once at the very beginning of the simulation as follows

```

DO 456 I=2,IMP1
DO 456 J=2,JMP1

```

```

DO 456 K=2,KMP1
BETA(1,I,J,K)=GARE (I ,J,K)*RH(I )/(DR(I)*R(I))
BETA(2,I,J,K)=GARE (I-1,J,K)*RH(I-1)/(DR(I)*R(I))
BETA(3,I,J,K)=GATHN(I,J ,K)/(R(I)*DTH(J))
BETA(4,I,J,K)=GATHN(I,J-1,K)/(R(I)*DTH(J))
456 BETA(5,I,J,K)=GAZT (I,J,K )/DZ(K)
BETA(6,I,J,K)=GAZT (I,J,K-1)/DZ(K).

```

In general the surface permeability in a given point is a tensor with nine components. We define in IVA3 three of them at the surface of the computational cell. The other three GAUC, GAVC and GAWC, are defined in the center (C) of the cell using the following procedure (convention).

```

DO 457 I=2,IMP2
DO 457 J=2,JMP2
DO 457 K=2,KMP2

IF((GARE (I,J,K)+GARE (I-1,J,K)).GT.0.0001) THEN
GAUC(I,J,K)=2.*GARE(I,J,K)*GARE(I-1,J,K)/
(GARE(I,J,K)+GARE(I-1,J,K))
ELSE
GAUC(I,J,K)=0
END IF

IF((GATHN(I,J,K)+GATHN(I,J-1,K)).GT.0.0001) THEN
* GAVC(I,J,K)=2.*GATHN(I,J,K)*GATHN(I,J-1,K)/
(GATHN(I,J,K)+GATHN(I,J-1,K))
ELSE
GAVC(I,J,K)=0.
END IF

IF((GAZT(I,J,K)+GAZT(I,J,K-1)).GT.0.0001) THEN
* GAWC(I,J,K)=2.*GAZT(I,J,K)*GAZT(I,J,K-1)/
(GAZT(I,J,K)+GAZT(I,J,K-1))
ELSE
GAWC(I,J,K)=0.
END IF
457 CONTINUE

```

This is in fact a harmonic averaging in case of one of the two permeabilities differs from zero. It provides zero permeability in the center in the corresponding direction if at least one of the permeabilities is equal to zero.

The residual three permeabilities are computed using some averaging procedure.

In order to save CPU time in the course of the simulation the above procedure of computing of BETA and GA groups is repeated only if changes of the geometry are introduced with the variable geometry boundary conditions.

1.3.2 INITIAL CONDITIONS

The initial conditions for all of the dependent variables should be defined at the very beginning of the analysis. Next we describe briefly the set of the dependent variables and how they are defined in the geometry defined in Chapter 1.3.1. The dependent variables describing the three phase flow behavior are defined in the next listed common blocks:

```

COMMON /I2VAR1/ P ( IMP2,JMP2,KMP2),
* PA( IMP2,JMP2,KMP2),
* XN (LM,IMP2,JMP2,KMP2),
* S (LM,IMP2,JMP2,KMP2),
* T (LM,IMP2,JMP2,KMP2),
* AL (LM,IMP2,JMP2,KMP2),
* XNA(LM,IMP2,JMP2,KMP2),
* SA (LM,IMP2,JMP2,KMP2),
* TA (LM,IMP2,JMP2,KMP2),
* ALA(LM,IMP2,JMP2,KMP2)

COMMON /I2VAR2/ UU (LM,IMP2,JMP2,KMP2),
* VV (LM,IMP2,JMP2,KMP2),
* WW (LM,IMP2,JMP2,KMP2),
* UUA(LM,IMP2,JMP2,KMP2),
* VVA(LM,IMP2,JMP2,KMP2),
* WWA(LM,IMP2,JMP2,KMP2)

COMMON /I2VAR6/ PART (LM,IMP2,JMP2,KMP2),
* PARTA(LM,IMP2,JMP2,KMP2),
* PNUC (LM,IMP2,JMP2,KMP2),
* PNUCA(LM,IMP2,JMP2,KMP2)

```

The meaning of the variables is:

P Pressure in Pa.

For each of the velocity fields $L=1\dots LM$ ($LM=3$ in this IVA3 version):

XN Inert mass concentration – dimensionless.

S Specific field entropy – $J/(kgK)$.

T Temperature – K (only as a input instead of S, used in IVA3 as an auxiliary variable).

AL Volume fraction (only two of the three volume fractions are "independent" variables) – dimensionless.

UU Radial velocity in m/s.

VV Angular velocity in m/s.

WW Axial velocity in m/s.

PART Particle number density in numbers per m^3 of the flow.

PNUC Nucleation density in numbers per unit volume of the velocity field (reserved for future development).

All above mentioned variables are defined in the actual, new time plane. The ending A means the old time plane value.

All dependent variables must be initialized before starting any simulation. IVA3 provides the following procedure for this purpose.

The default value for all velocity components is zero.

The default value for the number of the particle per unit volume is zero.

The default value for the number of the kinetic nuclei is 1.E+09.

36/1	FELDER	L5	=F	Dummy parameter.
36/2	NRSPEC	I5	1	Dummy parameter.

Pressures in volume regions:

37/1	NRZONG	I5		The information on the next statement is NRZONG times required (.GT.0).
------	--------	----	--	---

38/1	I1	I5	—
38/2	I2	I5	—
38/3	J1	I5	—
38/4	J2	I5	—
38/5	K1	I5	—
38/6	K2	I5	—
38/7	P1	F10.5	Pa
38/8	P2	F10.5	Pa

In the volume region (I,J,K) defined with (I1,I2), (J1,J2) and (K1,K2) the pressure changes linearly between the values P1 and P2 in the vertical direction only. This way of initialization of pressure fields is appropriate for steady state fluids in gravitational field.

Pressures initialization plane by plane:

39/1	NRZONG	I5		The information on the next statement is NRZONG times required (.GE.0).
------	--------	----	--	---

40/1	K	I5	—
40/2	I1	I5	—
40/3	I2	I5	—
40/4	J1	I5	—
40/5	J2	I5	—
40/6	P1	F10.5	Pa

In the horizontal layer of computational cells (I,J,K) defined with K=const, (I1,I2) and (J1,J2), the pressure is set uniform equal to P1.

Additional pressure initialization point by point:

41/1	NRZONG	I5		The information on the next statement is NRZONG times required (.GE.0).
------	--------	----	--	---

42/1	I	I5	—
------	---	----	---

42/2	J	I5	—
42/3	K	I5	—
42/4	P(I,J,K)	F10.5	Pa

Initialization of the temperatures, inert mass concentrations and volumetric fractions for each velocity field.

Usually the engineer prefers to work with the temperature of the fluid rather than with the specific entropy. That is why we provide the possibility to initialize the temperatures instead of the specific entropies. The code automatically computes the corresponding initial values of the specific entropies of the velocity fields.

Only one convention should be taken into account if the velocity field three consists only of an inert component:

(a) If the temperature is higher than the solid–liquid saturation temperature the liquid state is assumed. Appropriate to model liquid metals.

(b) If the temperature is lower than the solid–liquid saturation temperature the complete solidification is assumed.

Note, that for this case (inert component only) no solid liquid two phase mixture can be initialized within the velocity field three. Nevertheless IVA3 can cross this two phase region during the simulation, modelling it under the assumption of thermodynamic and mechanical equilibrium inside the field.

43/1	NRZONG I5	The information on the next statement is NRZONG times required (.GT.0).
------	-----------	---

44/1	I1	I4	—
44/2	I2	I4	—
44/3	J1	I4	—
44/4	J2	I4	—
44/5	K1	I4	—
44/6	K2	I4	—
44/7	T1	F8.5	K
44/8	T2	F8.5	K
44/9	XN1	F8.5	—
44/10	XN2	F8.5	—
44/11	AL1	F8.5	—
44/12	AL2	F8.5	—

In the volume region (I,J,K) defined with (I1,I2), (J1,J2),and (K1,K2) the temperature T, the inert mass concentrations XN, and the volumetric concentrations AL of the velocity field L=1 change linearly between the values T1 and T2 etc. in the vertical direction only.

The same procedure is repeated for fields L=2 and field L=3.

Velocity field L=2:

45/1	NRZONG I5	The information on the next statement is NRZONG times required (.GT.0).
------	-----------	---

46/1	I1	I4	—
------	----	----	---

46/2	I2	I4	—
46/3	J1	I4	—
46/4	J2	I4	—
46/5	K1	I4	—
46/6	K2	I4	—
46/7	T1	F8.5	K
46/8	T2	F8.5	K
46/9	XN1	F8.5	—
46/10	XN2	F8.5	—
46/11	AL1	F8.5	—
46/12	AL2	F8.5	—

Velocity field L=3:

47/1	NRZONG	I5	—	The information on the next statement is NRZONG times required (.GT.0).
48/1	I1	I4	—	
48/2	I2	I4	—	
48/3	J1	I4	—	
48/4	J2	I4	—	
48/5	K1	I4	—	
48/6	K2	I4	—	
48/7	T1	F8.5	K	
48/8	T2	F8.5	K	
48/9	XN1	F8.5	—	
48/10	XN2	F8.5	—	
48/11	AL1	F8.5	—	
48/12	AL2	F8.5	—	

Additional temperature, inert mass concentration and volumetric fraction initialization point by point:

Velocity field L=1:

49/1	NRZONG	I5	—	The information on the next statement is NRZONG times required (.GE.0).
50/1	I	I5	—	
50/2	J	I5	—	
50/3	K	I5	—	
50/4	T(1,I,J,K)	F10.5	K	
50/5	XN(1,I,J,K)	F10.5	—	
50/6	AL(1,I,J,K)	F10.5	—	

The same procedure is required for the other two velocity fields.

Velocity field L=2:

51/1	NRZONG	I5	—	The information on the next statement is NRZONG times required (.GE.0).
52/1	I	I5	—	
52/2	J	I5	—	

22 IVA3 Computer Code. Input description...

52/3	K	I5	—
52/4	T (2,I,J,K)	F10.5	K
52/5	XN(2,I,J,K)	F10.5	—
52/6	AL(2,I,J,K)	F10.5	—

Velocity field L=3:

53/1	NRZONG I5	The information on the next statement is NRZONG times required (.GE.0).
------	-----------	---

54/1	I	I5	—
54/2	J	I5	—
54/3	K	I5	—
54/4	T (3,I,J,K) F10.5		K
54/5	XN(3,I,J,K) F10.5		—
54/6	AL(3,I,J,K) F10.5		—

The initialization of the fuel rod temperatures in the radial direction in case of a modeling of nuclear reactor core (SZTRUE=T) is performed at this place automatically by setting the temperatures (a) equal to the liquid temperature if liquid is available in the cell or (b) equal to the gas temperature if only gas is available in the cell (AL(1,I,J,K).GT.=0.999). The real initial steady state temperatures in the fuel rods are computed at the first time step of the transient automatically. In case of a restart the fuel temperature fields are read from the last restart file.

Initialization of the field velocities:

55/1	NRZONG I5	The information on the next statement is NRZONG times required (.GT.0).
------	-----------	---

56/1	I1	I4	—
56/2	I2	I4	—
56/3	J1	I4	—
56/4	J2	I4	—
56/5	K1	I4	—
56/6	K2	I4	—
56/7	U1	F8.5	m/s
56/8	U2	F8.5	m/s
56/9	V1	F8.5	m/s
56/10	V2	F8.5	m/s
56/11	W1	F8.5	m/s
56/12	W2	F8.5	m/s

In the volume region (I,J,K) defined with (I1,I2), (J1,J2), and (K1,K2) the velocities in the radial (U), in the angular (V), and in the axial (W), directions of the velocity fields L=1,2,3 change linearly between the values U1 and U2 etc. in the vertical direction only. Note the homogeneity assumption is introduced only to simplify the input. In fact during the computation in IVA3 the velocities develop in accordance with the assumption of non homogeneity.

Additional input for the velocities point by point:

57/1	NRZONG I5	The information on the next statement is
------	-----------	--

NRZONG times required (.GE.0).

58/1	I	I5	—
58/2	J	I5	—
58/3	K	I5	—
58/4	UU(L,I,J,K)	F10.5	m/s
58/5	VV(L,I,J,K)	F10.5	m/s
58/6	WW(L,I,J,K)	F10.5	m/s

where L=1,2,3.

1.3.3 BOUNDARY CONDITIONS (BC)

1.3.3.1 DEPENDENT VARIABLES

In case of simulating a nuclear reactor core (SZTRUE=T) the variation of the thermal power with the time is needed. In IVA3 the input of the relative power as a time function is required. The power is normalized with the initial power at time zero.

The next statement is required only if SZTRUE=T.

59/1	NRFACT	I5		The information on the next statement is NRFACT times required (GT.0, LE.50, I=1,NRFACT)
60/1	TAUFAC(I)	F10.6	s	Time.
60/2	FACTAU(I)	F10.6	-	Normalized thermal power.

The boundary conditions for the dependent variables are gathered in groups which are physically meaningful. The number of these groups cannot exceed 20.

61/1	NRRB	I5		Number of the boundary conditions groups (GE.0, LE.20).
------	------	----	--	---

No boundary conditions are specified by setting NRRN=0. There are many situations where transients in a closed space are caused by geometry changes or by unstable initial conditions. Even in such cases one does not need boundary conditions.

The statements 61 through 76 are NRRB times required (I=1,NRRB).

62/1	IRB(I,1)	I5	-
62/2	JRB(I,1)	I5	-
62/3	KRB(I,1)	I5	-
62/4	IRBTYP(I,1)	I5	-
62/5	IRBTYP(I,2)	I5	-

IRB,JRB and KRB are the integer coordinates of the cell where the boundary condition with number I is applied. IRBTYP(I,1) specifies the type of the boundary condition.

IVA3 provides the following conventions:

IRBTYP(I,1)	=1	P, XN ,AL, T and PART values as time functions.
	=2	Radial velocities U as time functions.
	=3	Angular velocities V as time functions.
	=4	Axial velocities W as time functions.
	=5	Pressure P as time function.
	=6	Critical discharge.

The integration region is defined by the 6 surfaces namely:

$$I=\text{const}=1, I=\text{const}=IM+1,$$

J=const=1, J=const=JM+1,
K=const=1, K=const=KM+1.

At each of them velocities as boundary conditions (3,4,5) can be defined.

Only the pressure boundary condition, BC Nr.5, is needed to specify outflow from the integration region. The auxiliary layers of cells defined with

I=const=IM+1,
J=const=JM+1,
K=const=KM+1,

are allowed. In such cases one should define the surface permeability between the integration region and the pressure boundary condition cell different from zero.

The BC Nr.1 must be specified if flow is expected to leave the BC cell and enters the integration region (the pressure in this case is needed only for the computation of the thermodynamic and transport properties in the BC cell).

Note: Outflow from the computational region into the auxiliary layer of cells defined with

I=const=1,
J=const=1,
K=const=1,

or inflow from the auxiliary layer defined above into the computational region is not allowed in IVA3 unless the BC.2 and/or 3 and/or 4 are not supplementary defined. This is not a limitation of the code because in such cases the internal cells belonging to the computational region can be used for this purpose.

IRBTYP(I,2) specifies the number of the same boundary conditions from the I group. If IRBTYP(I,2)=1 the particular boundary condition already specified by IRBTYP(I,1) is applied only at the cell defined with IRB,JRB and KRB. In this case the next statement is not required. If IRBTYP(I,2) is greater than one, IRBTYP(I,2)-1 additional integer coordinates must be specified for the cells where the same boundary condition acts too (repetitions of the boundary conditions).

Note IRBTYP(I,2).LT.20.

The next record is IRBTYP(I,2)-1 times required.

63/1	IRB(I,LVID)	I5	-
63/2	JRB(I,LVID)	I5	-
63/3	KRB(I,LVID)	I5	-

Integer coordinates defining the cell where the already chosen boundary condition I acts too.

The next records specify the time functions and depend on the value of IRBTYP(I,1).

The records 63 through 68 are required only if IRBTYP(I,1)=1.

Number of points for P,XN,AL,T and PART as time functions (GT.0, LE.50).

64/1	NRPSP (I)	I5	—
64/2	NRXNSP(I)	I5	—
64/3	NRALSP(I)	I5	—
64/4	NRTSP (I)	I5	—
64/5	NRPART(I)	I5	—

The next record is NRPSP (I) times required.

65/1	TAUP(I,J)	F10.6	s	Time.
65/2	PRB (I,J)	F10.6	Pa	Pressure.

The next record is NRXNSP(I) times required.

66/1	TAUXN(I,J)	F10.5	s	Time.
66/2	XNRB (I,1,J)	F10.5	—	XN(,L=1,) values.
66/3	XNRB (I,2,J)	F10.5	—	XN(,L=2,) values.
66/4	XNRB (I,3,J)	F10.5	—	XN(,L=3,) values.

The next record is NRALSP(I) times required.

67/1	TAUAL(I,J)	F10.5	s	Time.
67/2	ALRB(I,1,J)	F10.5	—	AL(,L=1,) values.
67/3	ALRB(I,2,J)	F10.5	—	AL(,L=2,) values.
67/4	ALRB(I,3,J)	F10.5	—	AL(,L=3,) values.

The next record is NRTSP (I) times required.

68/1	TAUT(I,J)	F10.5	s	Time.
68/2	TRB(I,1,J)	F10.5	K	Temperature T(,L=1,).
68/3	TRB(I,2,J)	F10.5	K	Temperature T(,L=2,).
68/4	TRB(I,3,J)	F10.5	K	Temperature T(,L=3,).

The next record is NRPART(I) times required.

69/1	TAUPAR(I,J)	F10.5	s	Time.
69/2	PARTRB(I,1,J)	F10.5	m ⁻³	Partical number density for L=1.
69/3	PARTRB(I,2,J)	F10.5	m ⁻³	Partical number density for L=2.
69/4	PARTRB(I,3,J)	F10.5	m ⁻³	Partical number density for L=3.

The records 70 and 71 are required only if IRBTYP(I,1)=2.
Radial velocities U as time functions.

70/1	NRVSP(I)	I5	—	Number of the points for the radial velocity (U) boundary condition (GT.0, LE.50).
------	----------	----	---	--

The next record is NRVSP(I) times required.

71/1	TAUV(I,J)	F10.5	s	Time.
71/2	DUERB(I,1,J)	F10.5	m/s	Radial velocity L=1.
71/3	DUERB(I,2,J)	F10.5	m/s	Radial velocity L=2.
71/4	DUERB(I,3,J)	F10.5	m/s	Radial velocity L=3.

This group of boundary conditions must be used in cases where the mass is expected to leave the computational cell. Even for inflow in this cell and occasional outflow this boundary condition should be specified. If one specifies only the pressure boundary condition in the cell of the auxiliary boundary cell layer and mass is leaving the cell, the properties of the cell are automatically set by IVA3 equivalent to the properties of the neighboring internal cell of the computational region.

The records 72 and 73 are required only if IRBTYP(I,1)=3.
Angular velocities V as time function.

72/1 NRVSP(I) I5 The same as 70/1.

The next record is NRVSP(I) times required.

73/1 TAUV(I,J) F10.5 s The same as 71/1.
73/2 DVERB(I,1,J) F10.5 m/s Angular velocity L=1.
73/3 DVERB(I,2,J) F10.5 m/s Angular velocity L=2.
73/4 DVERB(I,3,J) F10.5 m/s Angular velocity L=3.

The records 74 and 75 are required only if IRBTYP(I,1)=4.
Axial velocities W as a time functions.

74/1 NRVSP(I) I5 s The same as 70/1.

The next record is NRVSP(I) times required.

75/1 TAUV(I,J) F10.5 s The same as 71/1.
75/2 DWERB(I,1,J) F10.5 m/s Angular velocity L=1.
75/3 DWERB(I,2,J) F10.5 m/s Angular velocity L=2.
75/4 DWERB(I,3,J) F10.5 m/s Angular velocity L=3.

The records 76 and 77 are required only if IRBTYP(I,1)=5.

Pressure P as time function:

76/1 NRPSP(I) I5 Number of points for the pressure as time function (GT.0, LE.50).

The next statement is NRPSP(I) times required.

77/1 TAUP(I,J) F10.6 s Time.
77/2 PRB (I,J) F10.6 Pa Pressure.

In case of IRBTYP(I,1)=6 which means critical flow boundary condition is required no additional statements are needed.

1.3.3.2 VARIABLE GEOMETRY

As already mentioned the IVA3 method define the surface permeabilities as time functions.

78/1	LVARG	L5	=T	Variable surface permeabilities with the time are required.
			=F	The surface permeability remain constant during this run.

IVA3 provides the option to include or not to include boundary conditions in the restart input data set. If new information is read as input it overrides the old one. Thus only the last input recorded in the restart files and read by the restart subroutine is available if one omits to include new data for the boundary conditions in the restart data set.

The statements 79 through 87 are required only if LVARG=T.

79/1	NRRBG	I5		Number of the boundary conditions for variable geometry (GE.0, LE.20). The statements 80 through 87 are NRRBG times required (I=1,NRRBG).
80/1	IRBG(I,1)	I5	-	
80/2	JRBG(I,1)	I5	-	
80/3	KRBG(I,1)	I5	-	
80/4	IRGTYP(I,1)	I5	-	
80/5	IRGTYP(I,2)	I5	-	

IRBG,JRBG and KRBG are the integer coordinates of the cell where the boundary condition with number I is applied. IRGTYP(I,1) specifies the type of the boundary condition.

IVA3 provides the following conventions for the variable permeabilities:

IRGTYP(I,1)	=1	Radial.
	=2	Angular.
	=3	Axial.

IRGTYP(I,2) specifies the number of the same boundary conditions from the I group. If IRBTYP(I,2)=1 the particular boundary condition already specified by IRGTYP(I,1) is applied only at the cell defined with IRBG,JRBG and KRBG. In this case the next statement is not required. If IRGTYP(I,2) is greater than one, IRBTYP(I,2)-1 additional integer coordinates must be specified for the cells where the same boundary condition acts too (repetitions of the boundary conditions).

Note IRGTYP(I,2).LT.20.

The next record is IRGTYP(I,2)-1 times required.

81/1	IRBG(I,LVID)	I5
81/2	JRBG(I,LVID)	I5
81/3	KRBG(I,LVID)	I5

Integer coordinates defining the cell where the already chosen boundary condition I acts

too.

The next record specifying the time functions depend on the value of IRGTYP(I,1).

The records 82 through 83 are required only if IRGTYP(I,1)=1.

82/1	NRGAR(I)	I5		Number of points for the radial porosities time functions (GT.0, LE.50).
as				

The next record is NRGAR(I) times required.

83/1	TAUGAR(I,J)	E10.3	s	Time.
83/2	GAREV (I,J)	E10.3	—	Radial permeability.

The records 84 through 85 are required only if IRGTYP(I,1)=2.

84/1	NRGATH(I)	I5		Number of points for the angular porosities as time functions (GT.0, LE.50).
------	-----------	----	--	---

The next record is NRGATH(I) times required.

85/1	TAUGAT(I,J)	E10.3	s	Time.
85/2	GATHNV(I,J)	E10.3	—	Angular permeability.

The records 86 through 87 are required only if IRGTYP(I,1)=3.

86/1	NRGAZ(I)	I5		Number of points for the radial porosities as time functions (GT.0, LE.50).
------	----------	----	--	--

The next statement is NRGAZ(I) times required.

87/1	TAUGAZ(I,J)	E10.3	s	Time.
87/2	GAZTV (I,J)	E10.3	—	Radial permeability.

1.3.3.3 EXTERNAL MASS SOURCES

88/1	NRRBQ	I5		Number of the boundary conditions for external mass sources (GE.0, LE.20). The statements 88 through 98 are NRRBQ times required (N=1,NRRBQ).
89/1	IRBQ(N,1)	I5	—	
89/2	JRBQ(N,1)	I5	—	
89/3	KRBN(N,1)	I5	—	
89/4	IRBQV(N)	I5	—	

IRBQ,JRBQ and KRBN are the integer coordinates of the cell where the boundary condition with number N is applied. IRBQV(N) specifies the number of the same boundary conditions from the N group. If IRBQV(N)=1 the particular boundary condition as already specified is applied only at the cell defined with IRBQ,JRBQ and KRBN. In this case the next statement is not required. If IRBQV(N) is greater than one, IRBQV(N)-1 additional integer coordinates must be specified for the cells where the same boundary condition acts too (repetitions of the boundary conditions).

Note IRBQV(N).LT.20.

The next record is IRBQV(N)-1 times required.

90/1	IRBG(I,LVID)	—	
90/2	JRBG(I,LVID)	—	
90/3	KRBG(I,LVID)	—	

Integer coordinates defining the cell where the already chosen boundary condition N acts too.

Number of points for P,XN,AL,T and PART as time functions (GT.0, LE.50).

91/1	NRPQ(N)	I5	—
91/2	NRXNQ(N)	I5	—
91/3	NRALQ(N)	I5	—
91/4	NRTQ(N)	I5	—
91/5	NRPARQ(N)	I5	—

The next record is NRPQ(N) times required.

92/1	TAUPQ(N,J)	F10.6	s	Time.
92/2	PRBQ(N,J)	F10.6	Pa	Pressure.

The next record is NRXNQ(N) times required.

93/1	TAUXNQ(N,J)	F10.5	s	Time.
93/2	XNRBQ(N,1,J)	F10.5	—	XN(,L=1,) values.
93/3	XNRBQ(N,2,J)	F10.5	—	XN(,L=2,) values.
93/4	XNRBQ(N,3,J)	F10.5	—	XN(,L=3,) values.

The next record is NRALQ(N) times required.

94/1	TAUALQ(N,J)	F10.5	s	Time.
94/2	ALRBQ(N,1,J)	F10.5	-	AL(,L=1,) values.
94/3	ALRBQ(N,2,J)	F10.5	-	AL(,L=2,) values.
94/4	ALRBQ(N,3,J)	F10.5	-	AL(,L=3,) values.

The next record is NRTQ(N) times required.

95/1	TAUTQ(N,J)	F10.5	s	Time.
95/2	TRBQ(N,1,J)	F10.5	K	Temperature T(,L=1,).
95/3	TRBQ(N,2,J)	F10.5	K	Temperature T(,L=2,).
95/4	TRBQ(N,3,J)	F10.5	K	Temperature T(,L=3,).

The next record is NRPARQ(N) times required.

96/1	TAUPAQ(N,J)	F10.5	s	Time.
96/2	PARRBQ(N,1,J)	F10.5	m ⁻³	Partical number density for L=1.
96/3	PARRBQ(N,2,J)	F10.5	m ⁻³	Partical number density for L=2.
96/4	PARRBQ(N,3,J)	F10.5	m ⁻³	Partical number density for L=3.

Mass source term as a function of time:

97/1	NRQM(N)	I5	Number of the points of external mass sources (GE.0, LE.50).	
98/1	TAUM(N,J)	F10.5	s	Time.
98/2	QMRB(N,J)	F10.5	kg/s	Mass source.

The dimension of kilogram per second convenient because it does not need correction if one changes the discretization. The external mass source term as needed in the code in kg/(m³s) is recomputed in the code by dividing with the cell volume:

$$QM = QM / (R(I) * DR(I) * DTH(J) * DZ(K))$$

The external mass source terms in kg/(m³s) for the separated velocity fields, XSAQ(L)*QM, and for the inert component inside the velocity fields, XNAQ(L)*XSAQ(L)*QM, are added to the computed in the codes as follows:

$$\begin{aligned} XSMUE(L,I,J,K) &= XSMUE(L,I,J,K) + XSAQ(L)*QM \\ XNMUE(L,I,J,K) &= XNMUE(L,I,J,K) + XNAQ(L)*XSAQ(L)*QM \end{aligned}$$

2. CHECK LIST FOR PREPARING AN IVA3 LOAD MODULE

Ensure that the following steps are performed before producing an executable load module appropriate for the particular case you are going to analyze:

(1) Did you include the appropriate equation of state package for the "solid" component of the third velocity field ? The following sets are available:

I3EOSUO2	Properties for liquid, equilibrium liquid – solid, and solid uranium dioxide.
I3EOSSS	Properties for liquid, equilibrium liquid – solid, and solid stainless steel.
I4EOSCOR	Properties for liquid, equilibrium liquid – solid, and solid corium mixture consisting of 60% UO ₂ , 16% ZrO ₂ , and 24% stainless steel (67% Fe, 21%Cr, 12% Ni) by weight.

(2) Did you change the parameter statement

PARAMETER (LM= ,IM= ,JM= ,KM=)

in MAIN IVA3, I2READ, INTGRA, WRESTA, RRESTA.

(3) Did you change the parameter statement

PARAMETER (IAM= , JAM=)

in MAIN IVA3.

(4) Did you change the parameter statement

PARAMETER (N= ,IMMM=)

in MAIN IVA3.

The following JCL set can be used for creating a load module:

```
//INR341A3 JOB (0341,104,P0N1Q),KOLEV,REGION=3999K,
// NOTIFY=INR341,MSGCLASS=H,TIME=(1,59)
// *MAIN LINES=20
// EXEC ASC
// A.SYSIN DD DSN=TSO341.IVA3.FORT(TLEFT),DISP=SHR
// EXEC F7CL,PARM.C='NOMAP,DEBUG',PARM.L='LIST,MAP',
// LIB=DUMMY
// C.SYSIN DD DSN=TSO341.IVA3.FORT(IVA3),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I4RES),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I4ALENT),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3RESTAR),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3PLOTI),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3PRINT),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3HILPR),DISP=SHR
```

```
// DD DSN=TSO341.IVA3.FORT(I3VELO),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3COEF1),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3ENTROP),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3PART),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3ALPHA),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3MASCON),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3POIS1),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3POIS2),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3POIS3),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3POIS4),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3MATRI),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3RAND),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I4CONSTI),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3SOURCE),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3MOMENT),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3DPDZ),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3LINKHM),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3FRAG),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3REGIM),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3FUEL),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3MASTRA),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3HEATTR),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3EOS),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I4EOSCOR),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3READ),DISP=SHR
// DD DSN=TSO341.IVA3.FORT(I3TURBO),DISP=SHR
//L.SYSLMOD DD DISP=SHR,UNIT=INR,VOL=SER=INR003,
// DSN=INR341.I14128.LOAD,SPACE=(TRK,(90,15,2),RLSE),
// DCB=(RECFM=U,BLKSIZE=24300)
//L.SYSIN DD *
ENTRY MAIN
NAME I14128(R)
/*
//
```


3. OUTPUT DESCRIPTION

All input data are printed for control purposes immediately after being read. So input errors are easily localized.

During the integration each IPRINT time steps a large edit is printed out. The format of the large edit is self explaining and will not be described here. Only the keys for the flow regimes addressed in the documented time step are given below.

The flow regime in each computational cell is postulated to be one of the following regimes given in Table 1. The corresponding number of the regimes are assigned to the vector IREGIM(I,J,K). The vector IREGIA(I,J,K) contains the flow regime identification in the old time level. By comparing both parameters IVA3 identifies the moment where the actual structure of the computational cells changes. This information is further used to model fragmentation processes of non-solid velocity fields.

Table 1 Postulated flow regimes in IVA3

IREGIM

	One velocity field only:
17	Velocity field 1 (gas) only.
1	Velocity field 2 (water) only.
22	Velocity field 3 (liquid metal) only.
	Two velocity fields only:
4	Two phase bubble flow – continuous velocity field 2 and discrete velocity field 1 (gas).
6	Two phase slug flow – continuous velocity field 2 and discrete velocity field 1 (gas).
7	Two phase churn turbulent flow – continuous velocity field 2 and discrete velocity field 1 (gas).
9	Two phase dispersed flow – continuous velocity field 1 (gas), discrete velocity field 2 (droplets) for pool flow.
8	Annular flow – continuous velocity field 1 (gas) and continuous velocity field 2 (water film), identified only for flow in a confined geometry.
3	Dispersed velocity field 3 (liquid metal droplets or solid particles) in continuous velocity field (2) (water).
2	Dispersed water droplets (2) in continuous liquid metal (3).
13	Continuous velocity field 1 (gas) and dispersed velocity field 3 (water droplets).

- 14 Continuous velocity field (1) (gas) and dispersed velocity field (3) (solid particles).
- 15 Dispersed velocity field 1 (gas bubbles) 1 in continuous velocity field 3 (liquid metal).
- 16 Continuous velocity field 1 (gas) and dispersed velocity field 3 (solid particles or liquid metal droplets).

Three velocity fields:

- 5 Three phase bubble flow – dispersed velocity field 1 (gas bubbles), continuous velocity field 2 (water) and dispersed velocity field 3 (liquid metal or solid particles).
- 21 Continuous velocity field 1 (gas), dispersed velocity field 2 (water droplets) and dispersed velocity field 3 (solid particles or liquid metal droplets or porous velocity field 3) – for pool flow only.
- 23 Continuous velocity field 1 (gas), disperse velocity field 2 (water droplets) surrounded by continuous velocity field 3 (porous liquid metal with large particle diameters).
- 11 Disperse velocity field 1 (gas bubbles), continuous velocity field 2 (water) and continuous velocity field 3.
- 10 Continuous velocity field 1 (gas), continuous velocity field 2 (water) and discrete velocity field (3) (water droplets) – only for flow in confined geometry.
- 12 Continuous velocity field 1 (gas), continuous velocity field 2 (water) and discrete velocity field (3) (microscopic solid particles) – only for flow in confined geometry.
-

You can suppress the large edit by setting IPRINT equal to a large number.

In each outer iteration some information about the convergence, accuracy and performance is printed out as follows. The term "maximum" is used below for the absolute value of all possible realizations of the physical quantity under consideration during the iteration – e.g. for all cells, all velocity fields, all directions.

ISTAZI	Number of the actual iteration.
DTAU	Time step used.
DTAUMI	The maximum possible time step computed by division of the mech length scale by the corresponding velocity. This is the so called Courant criterion.
DVMAX	The absolute value of the maximum velocity change for each particular iteration. Zero means the iteration does not contribute to

further increase the accuracy.

DELPMA	The absolute value of the maximum pressure change for each particular iteration. The cell where it occurs is defined by $MXDP(I,J,K)$ – designation, where the three integer numbers corresponds to I, J, and K, respectively.
DELRMA	The absolute value of the maximum mass conservation error as defined in the previous chapter for each particular iteration. The cell where it occurs is defined by $MDR(I,J,K)$, where the integer numbers below corresponds to I, J, and K, respectively.
DELTAS	Maximum error in the energy balance in entropy units.
DELTXS	Maximum error of the local mass conservation.
INTGR	CPU time for processing all constitutive relations.
I2GES	CPU time for preparing the coefficients of the momentum equations and the partial velocity decoupling.
POISO	CPU time for solving the POISSON equation.

After each time step a protocol of some important parameter is printed as follows:

MAKROI	Number of the successfully ended time step.
TAU	Physical process time.
CPU PRO MAKROI	CPU time per successfully finished time step in s.
DVMAXA(I,J,K)	For the successfully ended time step the absolute value of the maximum velocity change, DVMAXA, was observed for the velocity defined by I, J, and K integer coordinates.

4. COMMON BLOCKS

The contents of most of the COMMON blocks is described in Chapter 1. Here we give the explanation of the contents of only those COMMON blocks which are not described so far.

4.1 ALPHABETIC LIST OF THE COMMON BLOCKS

/BETA1 /	Contents described in Chapter 1.
/DIFUSI/	Contains the transport properties and densities of each velocity field:
RNUE	kinematic viscosity,
RL	thermal conductivity,
DDI	diffusion constant of the inert component,
RHOI	the best available estimation for the density at the new time level,
RHOIA	density at the old time level.
/DTAINT/	Contains auxiliary information for the time step control:
DTAUEI	Store the recommended time step as read per input.
DTAUMA	The absolute maximum of the possible time step without any restrictions.
DTAUAL	Time step recommended to overcome negative volume fractions without loss of mass if such situation happens.
/I2GEOM/	Contents described in Chapter 1.
/I2LOGI/	Contents described in Chapter 1.
/I2RAN1/	Content described in Chapter 1.
/I2RAN2/	Contents described in Chapter 1.
/I2RAN3/	Contents described in Chapter 1.
/I2RAN4/	Contents described in Chapter 1.
/I2SPZ1/	Contents described in Chapter 1.
/I2SPZ2/	Contents described in Chapter 1.
/I2SPZ3/	Contents described in Chapter 1.
/I2SPZ4/	
QP3	Power per m ³ of the fuel rod pellets.
QP3KM	Power inserted into one m ³ of the flow.

QP2	Heat flux in J per m ² at the cladding outer surface.
QP2QKR	Critical heat flux ratio, dimensionless.
BETE	Fuel temperature in different radial positions controlled by the first index:
n=7	Cladding surface temperature.
n=6	Gap temperature.
n=5	Pellet temperature at $R=RBS/2.*SQRT(1./2.)$, where RBS is the pellet radius.
n=4	Pellet temperature at $R=RBS/2.*SQRT(3./2.)$.
n=3	Pellet temperature at $R=RBS/2.*SQRT(5./2.)$.
n=2	Pellet temperature at $R=RBS/2.*SQRT(7./2.)$.
n=1	Maximum pellet temperature at the pellet axis.

/I2SPZ5/ Heat flux at the surface of the fuel rods and the fuel temperatures at the old time levels.

QP2A	Heat flux.
BETEA	Distributed fuel temperatures at the old time level.

/I2SPZ6/ Contains heat transfer coefficients at the surface of the fuel rods and the reference temperature for computation of the corresponding heat transfer coefficients:

HTC	heat transfer coefficients at the surface of the fuel rods,
TBEZUG	reference temperature for computation of the corresponding heat transfer coefficients.

/I2TITL/ Contents described in Chapter 1.

/I2VAR1/ Contents described in Chapter 1.

/I2VAR2/ Contents described in Chapter 1.

/I2VAR4/

RHO	Density of the three fluid mixture in the new time plane.
RHOA	Density of the three fluid mixture in the old time plane.
A	Velocity of sound of the homogeneous three phase mixture.
TSA	Derivative of the temperature with respect to the entropy of each velocity field at constant pressure and inert mass concentrations.
TP	Derivative of the temperature with respect to the pressure of each velocity field at constant entropy and inert mass concentrations.
TXN	Derivative of the temperature with respect to the inert mass concentration of each velocity field at constant entropy and pressure.

/I2VAR5/	Contains density derivatives of the velocity field:	
RHOP	with respect to pressure at constant specific entropy and inert concentration,	
RHOS	with respect to specific entropy at constant pressure and inert concentration, and	
RHOXN	with respect to the inert concentration at constant pressure and specific entropy.	
/I2VAR6/		
PART	Particle number density in the new time plane.	
PARTA	Particle number density in the old time plane.	
PNUC	Nucleation number density in the new time plane.	
PNUCA	Nucleation number density in the old time plane.	
/I2VEL2/	Contains the linearized coefficients of the nine momentum equations.	
	The best evaluation of the implicit velocities in case of no pressure difference across the control volume surface:	
DUU	in radial direction,	
DVV	in azimuthal direction, and	
DWW	in axial direction.	
	Pressure difference multipliers:	
RUU	in radial direction,	
RVV	in azimuthal direction, and	
RWW	in axial direction.	
/I3ALEP/		
ALEPS(L)	For volume fractions greater than ALEPS(L) the velocity field is considered as existent.	
/I4INCO/		
DTMAXI	Maximum allowed temperature increment inside one time step.	
DPMAXI	Maximum allowed pressure increment inside one time step.	
/I4SOL1/		
DP	pressure differences from iteration to iteration. Not used in IVA3.	

/IDENTI/

IREGIM Flow regime in the particular computational cell in the new time plane.
 IREGIA Flow regime in the particular computational cell in the old time plane.

/ISTOER/ Contents described in Chapter 1.

/MASGE1/ Contains the six positive defined mass source terms in $\text{kg}/(\text{m}^3 \text{ s})$:

MUE(1, mass source for velocity field l=1,
 MUE(2, mass sink for velocity field l=1,
 MUE(3, mass source for velocity field l=2,
 MUE(2, mass sink for velocity field l=2,
 MUE(5, mass source for velocity field l=3,
 MUE(2, mass sink for velocity field l=3.

/MASGEN/

XNMUE Source term for the halfe conservative form of the mass conservation equation of the inert component of the velocity field.
 XSMUE Source term for the conservative form of the mass conservation equation for each velocity field.
 XSMUEP Source term for the halfe conservative form of the mass conservation equation for each velocity field.
 DS Source term for the halfe conservative form of the entropy equation for each velocity field.
 DPART Source term for the conservative form of the particle number conservation equation for each velocity field.
 DIAMP Particles diameter of the existing velocity field.

/MATRI1/ Contains the coefficients and the right hand side of the pressure equations

$$\text{CJM } p_{j-1} + \text{CIM } p_{i-1} + \text{CKM } p_{k-1} + \text{C } p + \text{CJP } p_{j+1} + \text{CIP } p_{i+1} + \text{CKP } p_{k+1} = \text{D}.$$

/OMEGA/ Contents described in Chapter 1.

/SWICH1/

KBREAK Controls the initialization of the particle size at the very beginning of the computation with value equal to zero.

4.2 COMMON BLOCKS STORED IN RESTART FILES

The restart files are produced in IVA3 by the SUBROUTINE WRESTA for two reasons: (a) for successive restart by the SUBROUTINE RRESTA and/or (b) for post processing of the results, e.g. graphical representation, computation of integral parameters for the computational regions etc. The following common blocks form a IVA3 restart file:

BETA1,
I2GEOM,
I2LOGI,
I2RAN1,
I2SPZ1,
I2SPZ2,
I2SPZ3,
I2SPZ4,
I2TITL,
I2VAR1,
I2VAR2,
I2VAR4,
I2VAR6,
IDENTI,
ISTOER,
MASGEN.

5. TEST EXAMPLE

In what follows a short example is given how the code operates for a pre test simulation for an SACLE experiments being in preparation at the time of formulation of the task (the input data for this example for high pressure corium water interaction are formulated by H. Jacobs).

In a vertical vessel of 0.47 m I.D and 4.97 m high a corium melt mass of 4 x 12.85 kg is dropped into a one meter deep water pool having initially 4 x 33.50 kg. The water and the steam above the water are in saturation temperature corresponding to 50 bar pressure. The initial distribution of the melt is shown in fig. 5.1, and the initial velocity of the melt gas mixture in the cells containing melt was set arbitrary to -1.4 s in mechanical non equilibrium with the environment not satisfying any conservation law initially. The initial melt temperature was 3000 K, larger than the 2920 K liquidus temperature of the model corium included in this IVA3 load module. The simulation performed with IVA3 was 2D. The necessary JCL's and the input data prepared by H. Jacobs are attached to this note. The simulation takes 100 CPU min on an IBM 3090. Some 1.8 physical seconds are simulated for illustration.

The main results are presented in Figures 5.1 through 5.4. The accuracy requirements are chosen so as to ensure satisfactory mass conservation. The fuel mass conservation is demonstrated in fig.5.2. It ensures that no substantial complementing effects are expected from the tolerated conservation imbalance. In the same figure the corium being in bubble three phase flow is depicted too. Somewhere after 0.3 s the corium reaches the water surface and an intensive evaporation starts. In the same figure, for information the core mass which can be theoretically surrounded by liquid in granular stagnant situation – the absolute maximum of the premixing, is entered too. Fig. 5.3 illustrates the total mass of water. Until the evaporation starts it is indication also for the water mass conservation, which is obviously satisfactory. The void fraction distribution is depicted in fig.5.1 and the corresponding pressure change in 8 different positions in integration space is presented in fig.5.4. The initial approximately linear pressure increase is associated with heat transfer from the fragmenting corium into the environmental steam. The gradient of the pressure increase is changed exactly after the corium reaches the water surface. Further intermixing, as illustrated in fig. 5.1, is responsible for the intensive evaporation and the pressure increase until enough steam blankets the settled corium. The following small condensation at the liquid due to the fact that the liquid remains partially subcooled by the increased pressure is responsible for the low pressure decrease after the maximum was reached. Obviously after 1 s a very small amount of the corium is in bubble three phase flow. An extremely interesting evaporation mechanism is taking place – drops reach the corium and produce steam which enters the bubble flow region above the stagnant bottom region.

Remark: Due to careful time averaging of the source terms IVA3 is able to tolerate large prescribed errors. This is important for the development stage of the code but must be used extremely carefully in production calculations without any changes in the integration method. The accuracy of the numerical integration is the most important thing if one does not need the numerical noise of the code but the reaction of the conservation of the modeling equations and constitutive relationships.

Appendix 1: JCL's and input data for start and restart of IVA3 for the example discussed in Chapter 5.

JCL for executing IVA3:

```
//INR341S2 JOB (0341,104,P0N1Q),KOLEV,
// TIME=(,19),NOTIFY=INR341,MSGCLASS=H,REGION=3999K
//*MAIN LINES=5
// EXEC F7G,LIB=DUMMY,NAME=I14128
//G.STEPLIB DD DSN=INR341.I14128.LOAD,DISP=SHR
//G.FT06F001 DD DISP=SHR,DSN=INR341.OUTPUT.DATA1
//G.FT11F001 DD DISP=SHR,VOL=SER=INR003,UNIT=INR,
// SPACE=(TRK,(50,10),RLSE),DSN=INR341.VECCWT01.DATA,
// DCB=(RECFM=VBS,BLKSIZE=3476)
//G.FT13F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU101.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
//G.FT14F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,
// DSN=INR341.MAYER01.IVA2,DCB=(RECFM=VBS,BLKSIZE=23476)
//G.SYSIN DD DSN=TSO341.MELT.TEST(SACLES),DISP=SHR
```

The input deck is in:

```
//G.SYSIN DD DSN=TSO341.MELT.TEST(SACLES),DISP=SHR
```

The resulting listing is redirected to:

```
//G.FT06F001 DD DISP=SHR,DSN=INR341.OUTPUT.DATA1
```

The first restart file is created in

```
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU101.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
```

The output files

```
//G.FT11F001 DD DISP=SHR,VOL=SER=INR003,UNIT=INR,
// SPACE=(TRK,(50,10),RLSE),DSN=INR341.VECCWT01.DATA,
// DCB=(RECFM=VBS,BLKSIZE=3476)
```

and

```
//G.FT14F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,
// DSN=INR341.MAYER01.IVA2,DCB=(RECFM=VBS,BLKSIZE=23476)
```

are reserved for writing user defined output information during the integration.

JCL for restart of IVA3:

```
//INR341S2 JOB (0341,104,P0N1Q),KOLEV,
// TIME=(9,59),NOTIFY=INR341,MSGCLASS=H,REGION=3999K
// *MAIN LINES=5
// EXEC F7G,LIB=DUMMY,NAME=I14128
// G.STEPLIB DD DSN=INR341.I14128.LOAD,DISP=SHR
// G.FT06F001 DD DISP=SHR,DSN=INR341.OUTPUT.DATA10
// G.FT11F001 DD UNIT=INR,VOL=SER=INR002,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,
// DSN=INR341.VECCWT01.DATA,
// DCB=(RECFM=VBS,BLKSIZE=3476)
// G.FT12F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU109.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
// G.FT13F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU110.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
// G.FT14F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,
// DSN=INR341.MAYER09.IVA2,
// DCB=(RECFM=VBS,BLKSIZE=23476)
// G.SYSIN DD DSN=TSO341.MELT.TEST(SACLER),DISP=SHR
//
```

The input data are read from

```
//G.SYSIN DD DSN=TSO341.MELT.TEST(SACLER),DISP=SHR
```

The resulting listing is redirected to

```
//G.FT06F001 DD DISP=SHR,DSN=INR341.OUTPUT.DATA10
```

The preceding restart file is read from

```
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU109.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
```

At the end of the computation the following restart file is created

```
//G.FT13F001 DD UNIT=INR,VOL=SER=INR003,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,DSN=INR341.KWU110.IVA3,
// DCB=(RECFM=VBS,BLKSIZE=23476)
```

The output files

```
//G.FT11F001 DD UNIT=INR,VOL=SER=INR002,
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,
// DSN=INR341.VECCWT01.DATA,
// DCB=(RECFM=VBS,BLKSIZE=3476)
```

and

```
//G.FT14F001 DD UNIT=INR,VOL=SER=INR003,  
// SPACE=(TRK,(50,10),RLSE),DISP=SHR,  
// DSN=INR341.MAYER09.IVA2,  
// DCB=(RECFM=VBS,BLKSIZE=23476)
```

are reserved for writing user-defined output information during the integration.

Appendix 2: Summary of the figures discussed in Chapter 5

Fig. 5.1 Volume fraction of corium (black), water (dashed) and steam (transparent) as a function of space for different times.

Fig. 5.2 Pressure versus time. 8 curves in different positions.

Fig. 5.3 Limits of the corium water intermixing as a function of time (devided by 4). IVA3 2D prediction of the SACLE experiment.

Fig. 5.4 Water mass distribution versus time (divided by 4). IVA3 2D prediction of the SACLE experiment.

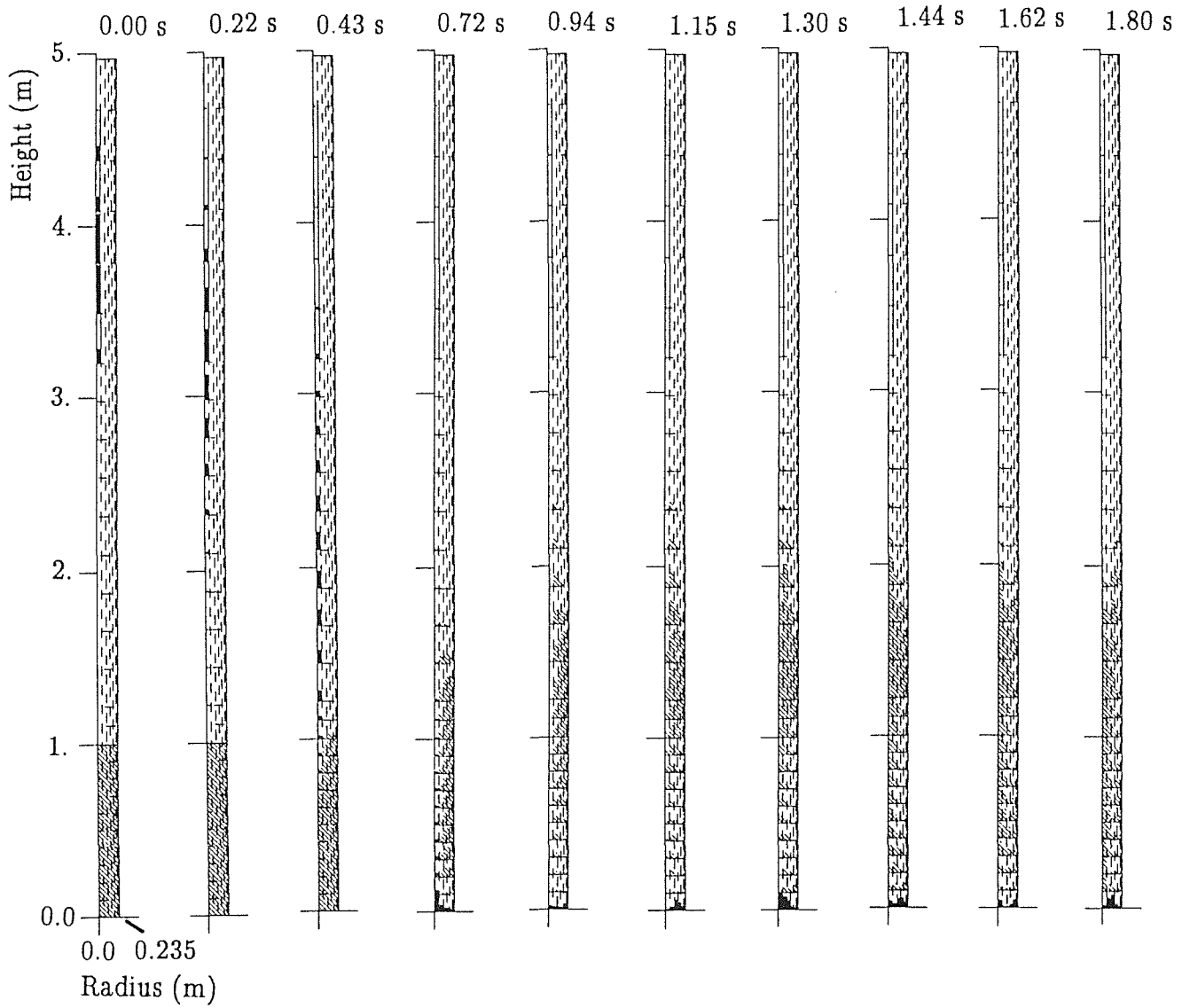


Fig. 5.1 Volume fraction of corium (black), water (dashed) and steam (transparent) as a function of space for different times.

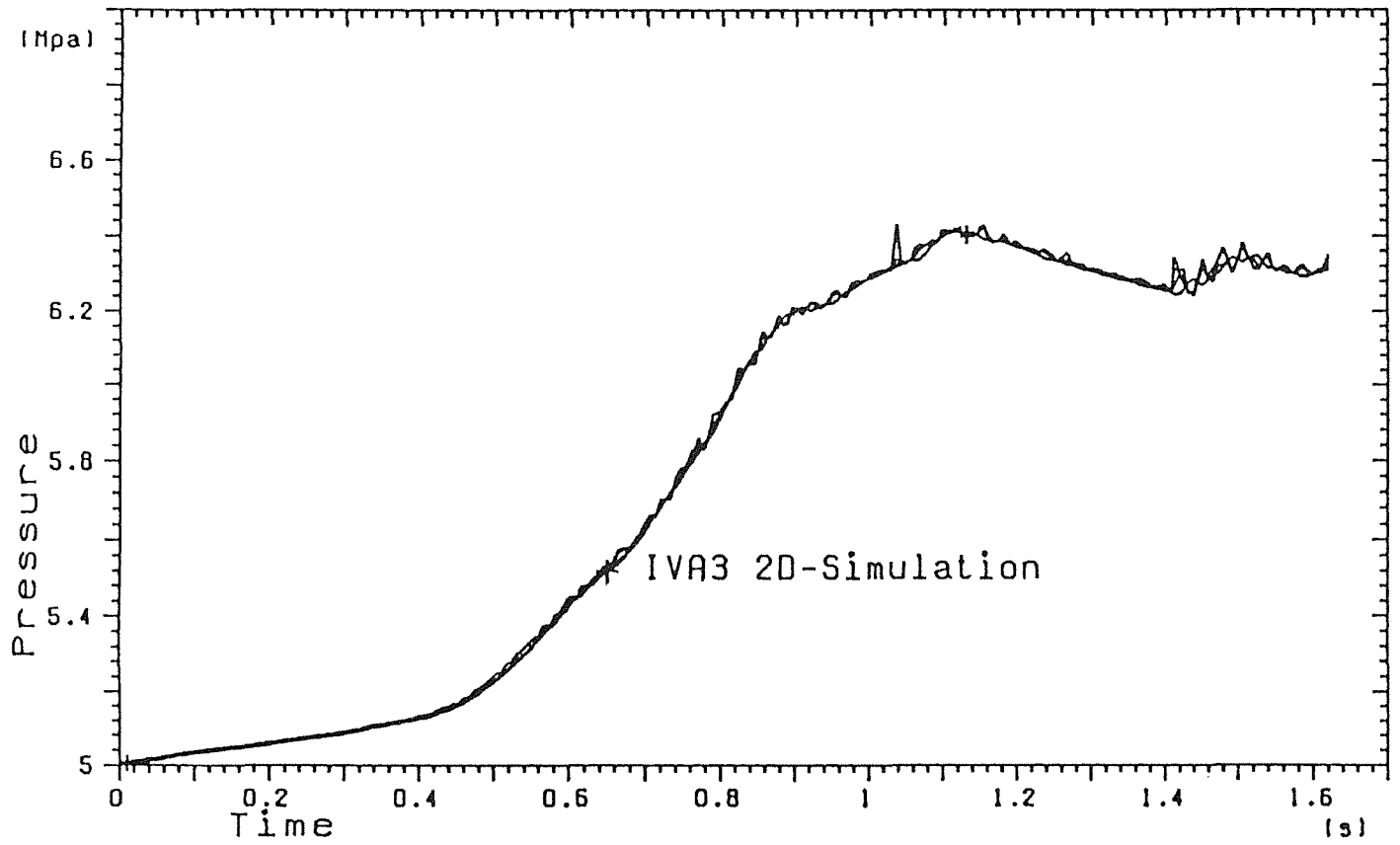


Fig. 5.2 Pressure versus time. 8 curves in different positions.

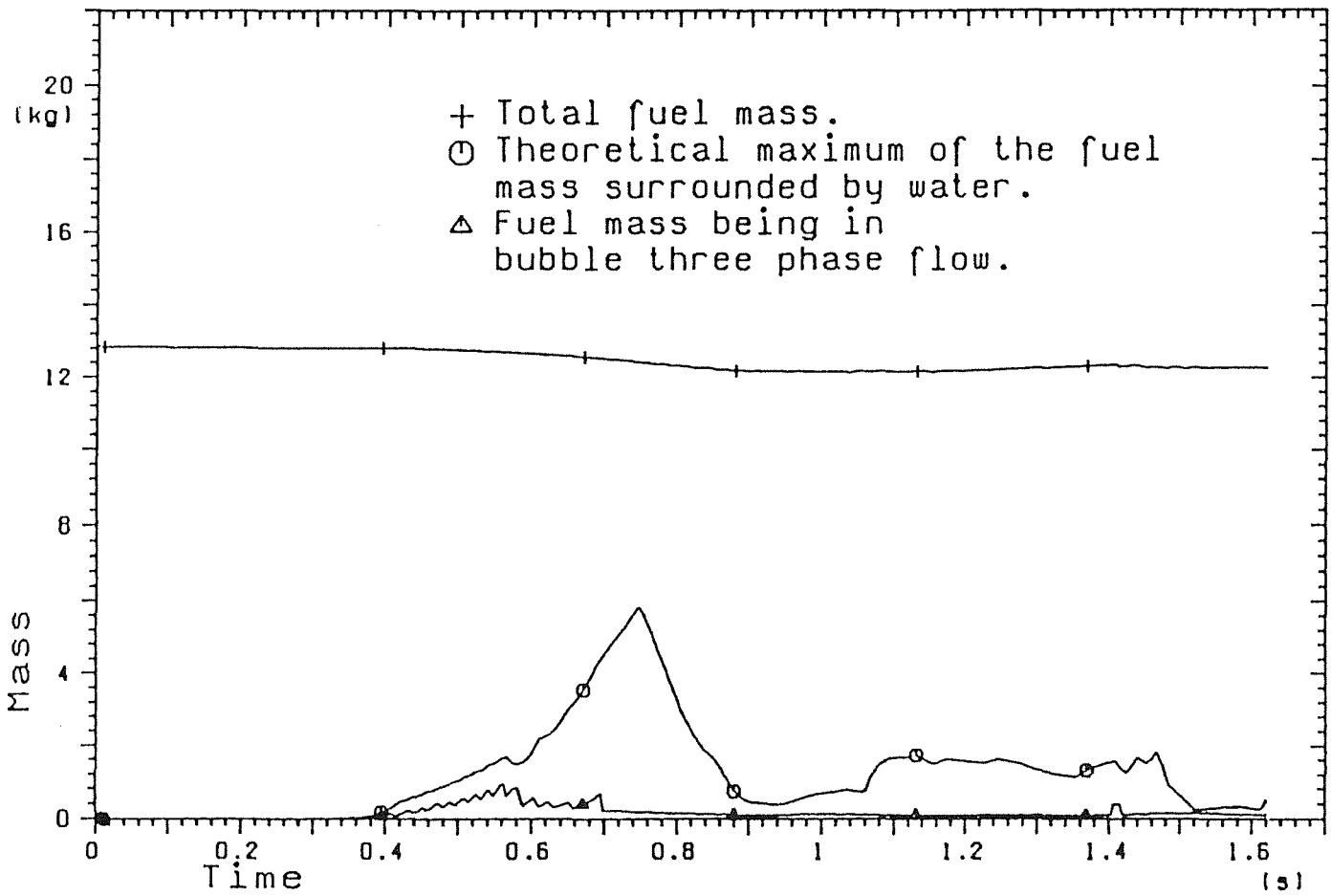


Fig. 5.3 Limits of the corium water intermixing as a function of time (devided by 4). IVA3 2D prediction of the SACLE experiment.

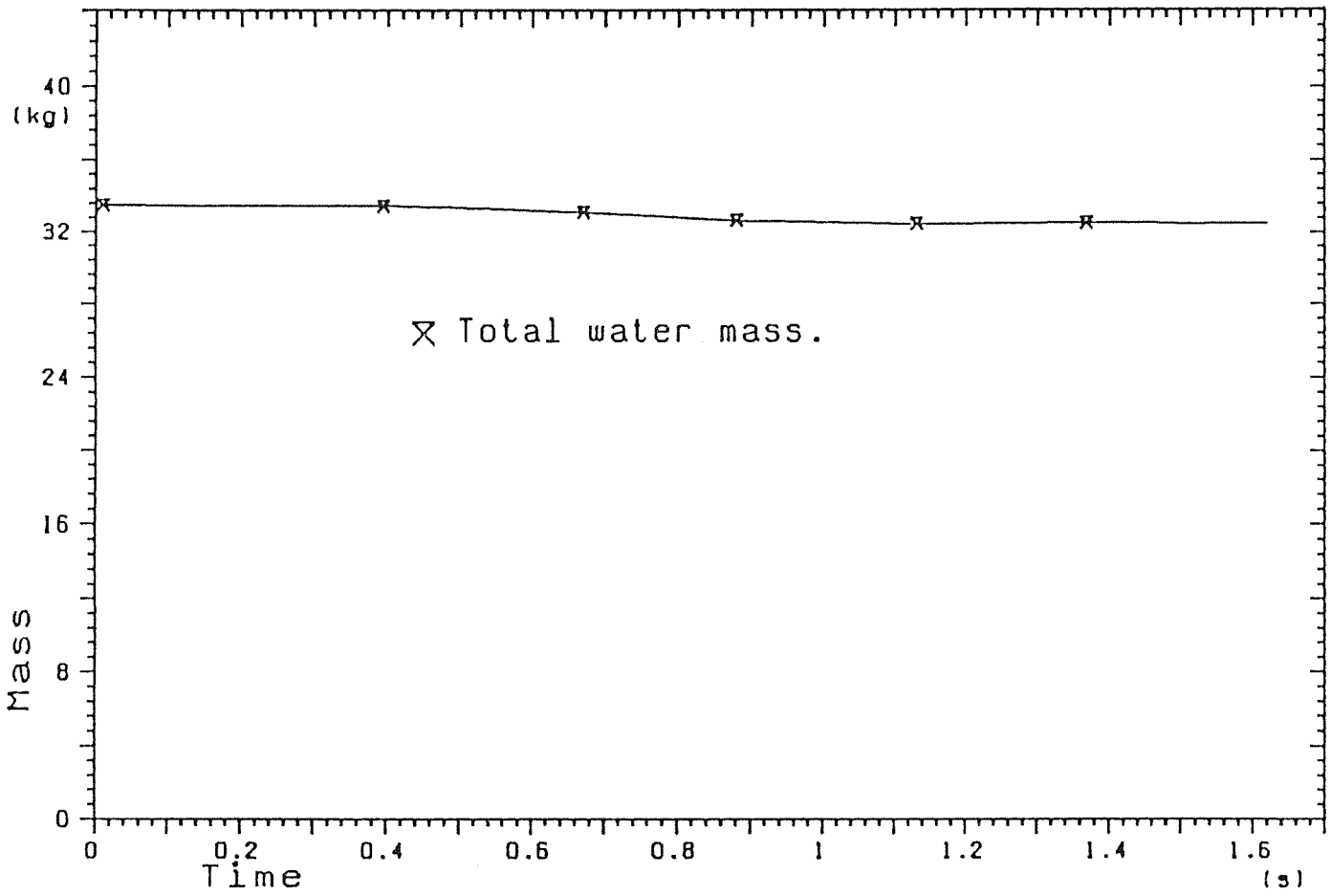


Fig. 5.4 Water mass distribution versus time (divided by 4). IVA3 2D prediction of the SACLE experiment.

Appendix 3: IVA3 interface for thermodynamic and thermo –physical properties.

SATURATED LINE WATER – STEAM

COMMON /SAET/ contains quantities describing water steam transition, namely:

TS	K	saturated temperature at system pressure.
VW	m ³ /kg	water specific volume at the saturation line.
VD	m ³ /kg	steam specific volume at the saturation line.
HW	J/kg	specific water enthalpy at the saturation line.
HD	J/kg	specific steam enthalpy at the saturation line.
HWD	J/kg	latent heat of vaporization.
SW	J/(kgK)	specific water entropy at the saturation line.
SD	J/(kgK)	specific steam entropy at the saturation line.
TSP	K/Pa	derivative of the temperature with respect to pressure at the saturation line.
CPW	J/(kgK)	water specific heat at constant pressure at the saturation line.
CPD	J/(kgK)	steam specific heat at constant pressure at the saturation line.
EW	kg/(ms)	water dynamic viscosity at the saturation line.
ED	kg/(ms)	steam dynamic viscosity at the saturation line.
LW	kg/(ms)	water thermal conductivity at the saturation line.
LD	kg/(ms)	steam thermal conductivity at the saturation line.
SIGMAS	N/m	surface tension water steam at the saturation line.

The above mentioned properties are computed by SUBROUTINE SAET01 either as a function of temperature TS in K, if the logical switch K=0, or as a function of pressure PS in Pa, if K=1, respectively.

The thermodynamic and thermo – physical properties of water are computed in SUBROUTINE WASSER as a function of temperature in K and system pressure in Pa. The results are:

RF	kg/m ³	density.
HF	J/kg	specific enthalpy.
SF	J/(kgK)	specific entropy.
CPF	J/(kgK)	specific heat at constant pressure.
ASF	m/s	velocity of sound.
EF	kg/(ms)	dynamic viscosity.
LF	W/(mK)	thermal conductivity.
PRF	–	Prandtl number.
HFP	(J/kg)/Pa	specific enthalpy derivative with respect to pressure at constant temperature.
RFT	kg/(m ³ K)	density derivative with respect to the temperature at constant pressure.
RFP	(kg/m ³)/Pa	density derivative with respect to the pressure at constant temperature.
SIG	N/m	surface tension metallic phase – gas.

The logical parameter LCATAS is set true if routines are addressed outside their applicability limits. This information is used further for localization of time and place where this occurs.

The thermodynamic and thermo – physical properties of air are computed in SUBROUTINE LUFT as a function of temperature TG in K and partial pressure PL in Pa. The results are:

RL	kg/m^3	density.
HL	J/kg	specific enthalpy.
SL	J/(kgK)	specific entropy.
CPL	J/(kgK)	specific heat at constant pressure.
ASL	m/s	velocity of sound.
EL	kg/(ms)	dynamic viscosity.
LL	W/(mK)	thermal conductivity.
PRL	–	Prandtl number.

The thermodynamic and thermo – physical properties of steam are computed in SUBROUTINE DAMPF as a function of temperature T in K and of the partial pressure PDD in Pa. The results are:

RD	kg/m^3	density.
HD	J/kg	specific enthalpy.
SD	J/(kgK)	specific entropy.
CPD	J/(kgK)	specific heat at constant pressure.
ASD	m/s	velocity of sound.
ED	kg/(ms)	dynamic viscosity.
LD	W/(mK)	thermal conductivity.
PRD	–	Prandtl number.
HDP	(J/kg)/Pa	specific enthalpy derivative with respect to pressure at constant temperature.
RDT	$\text{kg}/(\text{m}^3\text{K})$	density derivative with respect to the temperature at constant pressure.
RDP	$(\text{kg}/\text{m}^3)/\text{Pa}$	density derivative with respect to the pressure at constant temperature.

The final properties of the velocity fields are computed in the subroutines FLU and GAS.

SUBROUTINE FLU compute properties of a binary mixture consisting of inert and non inert components as a functions of the field temperature TF in K, system pressure P in Pa and inert mass concentration XBS. The results are:

SF	J/(kgK)	specific entropy.
HFF	J/kg	specific water enthalpy.
RF	kg/m^3	density.
AFS	m/s	velocity of sound.
RFSF	$(\text{kg}/\text{m}^3)/(\text{J}/\text{kgK})$	density derivative with respect to the specific entropy at constant pressure and inert mass concentration.
RFXBS	(kg/m^3)	density derivative with respect to the inert mass concentration at

		constant pressure and specific entropy.
TFSF	$K/[J/(kgK)]$	temperature derivative with respect to the specific entropy at constant pressure and inert mass concentration.
TFP	K/Pa	temperature derivative with respect to the pressure at constant specific entropy and inert mass concentrations.
TFXBS	K	temperature derivative with respect to the inert mass concentration at constant pressure and specific entropy.
CPF	$J/(kgK)$	specific heat at constant pressure.
EF	$kg/(ms)$	dynamic viscosity.
LF	$W/(mK)$	thermal conductivity.
PRF	$-$	Prandtl number.
SIGF	N/m	surface tension metallic phase – gas.
SB	$J/(kgK)$	specific entropy of the inert component.
SFF	$J/(kgK)$	specific entropy of the water component.
HB	J/kg	specific enthalpy of the inert component.

The logical parameter LCATAS is set true if routines are addressed outside their applicability limits. This information is used further for localization of time and place where this occurs.

SUBROUTINE GAS computes properties of a binary gas mixture consisting of inert component (air) and non inert component (steam) as a functions of the gas temperature TG in K, if the system pressure P in Pa and of the inert mass concentration XLS. The results are:

SG	$J/(kgK)$	specific entropy.
HGD	J/kg	specific steam enthalpy.
RG	kg/m^3	density.
AGS	m/s	velocity of sound.
RGSG	$(kg/m^3)/(J/kgK)$	density derivative with respect to the specific entropy at constant pressure and inert mass concentration.
RGXLS	(kg/m^3)	density derivative with respect to the inert mass concentration at constant pressure and specific entropy.
TGSG	$K/(J/(kgK))$	temperature derivative with respect to the specific entropy at constant pressure and inert mass concentration.
TGP	K/Pa	temperature derivative with respect to the pressure at constant specific entropy and inert mass concentrations.
TGXL	K	temperature derivative with respect to the inert mass concentration at constant pressure and specific entropy.
CPG	$J/(kgK)$	specific heat at constant pressure.
EG	$kg/(ms)$	dynamic viscosity.
LG	$W/(mK)$	thermal conductivity.
PRG	$-$	Prandtl number.
DDL	m^2/s	diffusions constant air in steam.
PL	Pa	partial pressure of the inert component.
SL	$J/(kgK)$	specific entropy of the inert component.
SD	$J/(kgK)$	specific entropy of the steam component.
HL	J/kg	specific enthalpy of the inert component.

The logical parameter LCATAS is set true if routines are addressed outside their applicability limits. This information is used further for localization of time and place

where this occurs.

Liquid metal, solid liquid two phase equilibrium mixture, frozen particles

COMMON /SAET3/ contains quantities describing solid-liquid transition, namely:

T3S	K	liquidus temperature.
R3S	kg/m ³	saturated solid phase density.
R3L	kg/m ³	saturated liquid density.
R3TS	kg/(m ³ K)	density derivative with respect to temperature at the two phase/solid transition line.
R3TL	kg/(m ³ K)	density derivative with respect to temperature at the liquid/two phase transition line.
H3S	J/kg	saturated solid phase specific enthalpy.
H3L	K/kg	saturated liquid phase specific enthalpy.
H3LS	J/kg	latent heat of solidification.
S3S	J/(kgK)	saturated solid phase specific entropy.
S3L	J/(kgK)	saturated liquid phase specific entropy.
S3LS	J/(kgK)	latent specific solidification entropy.
CP3S	J/(kgK)	specific heat at constant pressure – saturated solid phase.
CP3L	J/(kgK)	specific heat at constant pressure – saturated liquid phase.
E3S	kg/(ms)	saturated solid phase dynamic density.
E3L	kg/(ms)	saturated liquid phase dynamic viscosity.
L3S	W/(mK)	saturated solid phase thermal conductivity.
L3L	W/(mK)	saturated liquid phase thermal conductivity.
SIGMA3	N/m	surface tension liquid metal – gas.

For description of the thermo – physical properties of solid and liquid materials the following functions are necessary:

TUO2L	K	temperature of the liquid metal as a function of the specific entropy in J/(kgK).
SIGUO2	N/m	surface tension of the liquid metal as a function of the temperature in K.
ETAUO2	kg/(ms)	liquid metal dynamic viscosity as a function of the temperature in K.
HUO2	J/kg	specific enthalpy of the solid phase as a function of the temperature in K.
HUO2L	J/kg	specific liquid metal enthalpy as a function of the temperature in K.
CPUO2	J/(kgK)	specific heat at constant pressure for solid phase as a function of the temperature in K.
CPUO2L	J/(kgK)	specific heat at constant pressure for liquid metal as a function of the temperature in K.
SUO2	J/(kgK)	specific solid phase entropy as a function of the temperature in K.

SUO2L $J/(kgK)$ specific liquid metal entropy as a function of the temperature in K.

RHO2L kg/m^3 solid phase density as a function of temperature in K.

RTUO2 $kg/(m^3K)$ solid density derivatives with respect to the temperature as a function of the temperature K.

RHO2L kg/m^3 liquid metal density as a function of the temperature in K.

RTUO2L $kg/(m^3K)$ liquid metal density derivatives with respect to the temperature as a function of the temperature in K.

WLUO2 $W/(mK)$ solid phase thermal conductivity as a function of temperature in K.

WLUO2L $W/(mK)$ liquid metal thermal conductivity as a function of temperature.

SUBROUTINE EOSUO2 checks in which state the metallic velocity field is and computes properties either for liquid state or for two phase liquid – solid equilibrium state or for solid state. The following properties as a function of temperature in K and/or of the specific entropy in $J/(kgK)$ are the result:

R	kg/m^3	density.
H	J/kg	specific enthalpy.
CP	$J/(kgK)$	specific heat at constant pressure.
AS	m/s	velocity of sound.
E	$kg/(ms)$	dynamic viscosity.
L	$W/(mK)$	thermal conductivity.
PR	–	Prandtl number.
HP	$(J/kg)/Pa$	specific enthalpy derivative with respect to pressure at constant temperature.
RT	$kg/(m^3K)$	density derivative with respect to the temperature at constant pressure.
RP	$(kg/m^3)/Pa$	density derivative with respect to the pressure at constant temperature.
SIG	N/m	surface tension metallic phase – gas.

Postulated flow regimes in IVA3

