Development of a computer tool for in-depth analysis and postprocessing of the RELAP 5 thermalhydraulic code

CSN

Colección
Otros Documentos
31.2004

CSN

CONSEJO DE
SEGURIDAD NUCLEAR

Development of a computer tool for in-depth analysis and postprocessing of the RELAP5 thermalhydraulic code

Roberto Herrero Santos and José María Izquierdo Rocha
Consejo de Seguridad Nuclear

CSN CONSEJO DE SEGURIDAD NUCLEAR

Colección
Otros documentos CSN
Referencia: ODE-04.20

Publicado y distribuido por:
Consejo de Seguridad Nuclear
Justo Dorado, 11. 28040 - Madrid
http://www.csn.es
peticiones@csn.es

Maquetación: RGB Comunicación, S.L.
Imprime: Neografis, S.L.
Depósito legal: M-46623-2004

Contents

Presentación 5

I. Problem setup 9

I.1. Code problems 11

I.2. Objectives 15

II. Verifications implemented in the analysis tool 17

II.1. Introduction 18

II.2. Mass conservation equation 18

II.3. Mixture momentum equation 21

II.4. Pressure and volume equations 23

II.5. Non-equilibrium mass equation 27

II.6. Correspondence of variables 33

III. Description of the analysis tool 37

III.1. Introduction 39

III.2. Description of the programs used in the analysis tool 39

III.3. Linkage of the codes 41

III.4. Functioning of the analysis tool 43

IV. Application examples 47

IV.1. Introduction 49

IV.2. Compression caused by heating in a closed pipe 49

IV.3. Depressurization in a circuit with liquid and vapor 55

IV.4. Over specification of boundary conditions 66

IV.5. Full plant model. Flow mixing problem 77

V. Conclusions and recommendations for future developments 91

VI.1. Conclusions 93

VI.2. Recommendations for future develpments 94

Bibliography 99

Appendices 105

A. Nomenclature 107

B. Partial derivatives of thermodynamic properties 110

Presentación

Presentación

El Consejo de Seguridad Nuclear (CSN) ha promovido el desarrollo de una herramienta informática para el diagnóstico de errores y la interpretación de los transitorios de presión y nivel calculados con el código de simulación termohidráulica RELAP5/MOD3.2.

A tal fin se han deducido unas ecuaciones de balance a partir de las ecuaciones originales de masa y energía. Estos balances incluyen las ecuaciones de la presión, el nivel y la masa en desequilibrio. Los balances se emplean para comprobar la consistencia de la solución y para interpretar la evolución de la presión y el nivel en términos de variables físicas significativas.

La implantación en la herramienta de dichas verificaciones se ha conseguido mediante un script escrito en el lenguaje AWK y mediante el lenguaje de simulación de propósito general BABIECA. El script AWK genera automáticamente el fichero de entrada de BABIECA necesario para verificar el transitorio del código RELAP5. El intercambio de información entre RELAP5 y BABIECA tiene lugar mediante la librería de paso de mensajes PVM.

La herramienta se ha aplicado en algunos ejemplos demostrativos, que han revelado la existencia de algunos errores en el código RELAP5, tales como la sobre-especificación de condiciones de contorno y el mezclado de caudales.

Se sugieren finalmente las líneas maestras para el desarrollo de códigos con control de los errores.

Presentation

The Spanish Nuclear Safety Council (CSN) has sponsored the development of a computer tool for the error diagnosis and interpretation of the pressure and level transients computed with the code RELAP5/MOD3.2.

For that purpose some balance equations have been rigorously derived from the original mass and energy equations. Those balances include the pressure, level and non-equilibrium mass equations. The balances are used to check the consistency of the solution and to interpret the pressure and level evolution in terms of physically meaningful variables.

Verification of the equations is implemented in the tool with the aid of an AWK script and the general purpose simulation language BABIECA. The AWK script automatically generates the BABIECA input file necessary to check the RELAP5 transient. The execution of BABIECA with the corresponding input file implies the spawning of the RELAP5 transient. RELAP5 information is transferred to BABIECA through PVM, so that the balance equations can be verified.

The tool has been applied to some application examples, which have revealed some misconceptions in RELAP5, such as the overspecification of boundary conditions and the flow mixing problems.

Some guidelines for the development of future codes, with full control of the discretization errors, are pointed out.

I. Problem setup

I. Problem setup

I.1. Problems concerning the use of Thermalhydraulic codes in Nuclear Safety

I.1.1. Analysis of Thermalhydraulic codes

Nuclear Safety specialists face very often the need of simulating the transients expected to occur in nuclear power plants, in order to determine and quantify the associated risks. This need can be found in a wide variety of activities such as the conception and design of plant systems, optimization of procedures and set-points, operator training simulators, success criteria in PSAs, accident analyses, licensing tasks, etc.

The simulation of the operating and accidental transients that may happen in power plants based on a nuclear fission process requires, as a previous step, the description of the physical systems involved in the plant by means of mathematical models. A mathematical model could be defined as a set of, generally partial, differential equations capable of describing and predicting certain phenomena in a range of applicability. The predictive capability of the models must be considered as a mandatory requirement, since a set of equations that can only reproduce some transients by tuning certain parameters must never be regarded as a model. The predictive capability is inherent to the concept of model.

The solution of the equations involved in the models that describe a nuclear power plant is a very difficult problem. An analytical solution of the equations can never be achieved in practice. The most common approach consists in solving a dicretized version of the differential equations by means of a programmable computer algorithm. The resulting program is usually called a code. The nuclear industry counts on many codes focused on the different disciplines involved in Nuclear Safety: neutronics, containment building, structural analysis, etc. Thermalhydraulic codes, and specially the RELAP5 code [37], must not be forgotten.

The Thermalhydraulic codes simulate the behavior of water and steam circuits used to transport the energy generated in the reactor. Despite their undeniable advantages, the use of computer codes implies a number of problems which are very frequently understimated or not taken into account. For instance, the solution of the discretized equations is subject to errors that make very difficult to ensure how good the solution is. Interpretation of results is another remarkable problem. Very often the users regard the codes as black boxes that give an answer to the problems set through the input file, forgetting that they are formed by mathematical models and that they never substitute the plants perfectly. Other problems are related to the interaction between the codes and the users: build-up of the input file, correct interpretation of the data supplied to the code, nodalization, decisions about the different options available in the code, etc. As a consequence, the code may be told to solve a problem different from that the user would like to solve. On the other hand, the user may obtain wrong conclusions because of a misunderstanding of the output information computed by the code.

Some of the problems mentioned above are described in a greater detail in the following sections.

I.1.2. Verification of the solutions computed by the TH codes

Disagreement between experimental results and the output data computed by a code is not unusual at all. The causes of the discrepancies, usually referred to as *uncertainties,* may be classified as follows:

1. *External* sources, originated in the modelling process. Two different types can be defined in turn:
    - The discrepancies between the analytical equations of the models and the real world, which may be originated in deficiencies in the instrumentation used to obtain the experimental correlations, inapropriate curve fitting, wrong modelling hypotheses, etc.
    - Hypersensitivity of the model to certain parameters, initial conditions or boundary conditions, which cannot be measured with the required precision. As a result, some output variables may lie within a very broad range as the value of the above mentioned parameters is changed within the measuring error interval.
2. *Internal* sources, related to the discretization process of the analytical equations and the discrete nature of digital computers. Three types can be noticed:
    - Round-off errors, caused by the finite precision with which floating-point numbers are represented in a digital computer. The representation may be different depending on the machine and even on the compiler [39, 10]. Round-off errors are the root cause of the uncertainties attributed to machines and compilers in references such as [7, 27].
    - Truncation errors, which arise whenever a Taylor series is used. A discrete numerical method is a good example, since it implies in most cases a Taylor expansion up to a certain order. Truncation errors also appear when trying to represent in a digital computer non-rational numbers such as e, π, square roots, trigonometric functions, exponential functions, or any other trascendental function. Computation of that kind of numbers is always achieved by means of Taylor expansions. From the theoretical point of view, truncation errors can be as small as desired. For numerical methods, refinement of the spatio-temporal discretization mesh should decrease the error. In the case of the computation of non-rational numbers, calculation of higher order terms in the series should yield the same effect. But in real practice, the refinement of the mesh or the increment of the terms in the expansion results in more mathematical operations and hence in greater round-off errors. From a certain point on, round-off errors could even compensate for the decreasing truncation errors. The native floating-point precision with fixed finite length used by digital computers is then the main source of errors in computer simulations.

The so called external sources of uncertainty can be hardly suppressed, since they are closely related to the very nature of mathematical modelling. Ensuring that the analytical equations model the real world accurately is the process known as *validation.* This work does not pay attention to that kind of uncertainties.

On the other hand, internal uncertainties could be eliminated more easily. Round-off errors can be decreased as much as desired getting rid of machine native floating-point arithmetic, and using instead a software implemented multiple precision arithmetic, fitted to the particular needs of each problem [1]. Once the round-off errors are controlled, the solution can be improved by progressive refinement of the discretization mesh.

Assuming that current codes do not implement multiple precision arithmetic, the quality of the computed solution must be ensured by other methods. Reference [42] considers that the numerical solution computed by a code is uniquely related to the analytical equations as long as the following items are taken into account:

1. Estimation of the associated errors.
2. Comparison with well known analytical solutions.
3. Replacement of the solution into the original equations.

The two first items above imply an *a priori* verification of the solution. On the other hand, item 3 is an *a posteriori* method.

With respect to item 1, binding of the truncation error may be a cumbersome task. First of all, it may be difficult to find an analytical expresion for the error associated to a numerical method. Secondly, finding upper bounds of the error in certain ranges of the spatio-temporal domain might not be trivial.

Moreover, simulation codes result from coupling several sections, each of them accounting for certain physical models, yielding feedback loops. For instance, the RELAP5 code comprises a hydrodynamic section and a heat structure section among many others. Each of them is solved with its own numerical method. As a consequence, the boundary conditions of each section are already computed with a certain error, making it difficult to bind the error of the overall system.

With respect to item 2, the solution computed by the code can be compared to the analytical solution only in some cases. The water faucet and the oscillating manometer problems described in [32] are typical examples. Unfortunately, analytical solutions can be found in these cases at the expense of oversimplifying of the modelling hypotheses. For instance, constant and uniform pressure to avoid the use of water property tables, separated phases, no mass, momentum and energy interchange between phases, and incompressible and non viscid flows, are common hypotheses. The range of physical phenomena tested with those examples is then rather constrained, specially when compared to the range of all possible phenomena that may occur during a transient. Comparison with analytical solutions seems to be suitable just to test the numerical methods during the earlier stages of the developmental process. Fulfilment of those tests is anyway a necessary but not a sufficient condition to ensure the quality of the code.

An *a posteriori* verification of each particular calculation is much more advisable according to [35, 21, 29]. Substitution of the solution into the original analytical equations seems to be the natural way to check the results. However, this approach faces a great obstacle: solutions in the discrete domain must be substituted in equations defined in a continuous domain. Either continuous functions must be obtained from the discrete solutions, or the analytical equations must be discretized before replacement.

The first choice implies fitting the discrete results to continuous curves with a predefined shape. The curves should be then replaced into the analytical equations. The most important problem with this approach is that the higher the order of a derivative, the less reliable it is. The comparison could lead thus to negative conclusions even if the discrete solution is acceptable.

The second choice is the replacement of the discrete solution into a discretized form of the analytical equations. If the discretization is just the numerical method used to obtain the discrete solution, the problem becomes trivial. That is the reason why the discretized equations in this work are not the original ones but other rigorously derived from them. Chapter 2 is mainly devoted to the derivation of these new equations.

Verification of the computed solution is strongly encouraged in the last version of the US NRC Standard Review Plan [23], in the following terms: *The reviewers should confirm that the numerical solution conserves all important quantities. Even when the mathematical equations conserve mass, momentum, and energy, the numerical method used to solve the equations may not conserve any of these quantities. The reviewers should also confirm that all code options that are to be used in the accident simulation are appropriate and are not used merely for code tuning.*

1.1.3. Interpretation of results

Another problem that the code users face usually is the right interpretation of the results computed by the code [19]. The codes supply a large amount of output information. However, only a small fraction is used by the user to analyze the results. The right interpretation of the phenomena occurred during the transient may be hindered by several factors:

- Ignorance of the analysts about the models actually included in the code.
- Tendency of the analysts to explain the results in terms of models simpler than those embeded in the code (the homogeneous equilibrium model is perhaps the most representative example).
- Tendency of the analysts to explain the code results in terms of physical phenomena not modelled at all in the code.

1.1.4. User effect

Thermalhydraulic codes have several options, many of which are taken by default, that must be chosen by the users in the input file. These options allow for the selection of different models, solution procedures, convergence criteria, etc. Furthermore, different users may solve the same problem by means of different strategies and procedures (nodalization, time step, cross-flow or normal junctions, models, etc.), leading to dramatically different results. This is what is called the undesirable user effect, which has caused a great concern in the international community [4].

I.2. Objectives

The main objective of this work is the development of a computer tool to help the users of thermalhydraulic codes, and particularly of RELAP5, to solve or alleviate the problems described in section I.1. The proposed tool will be achieved by integrating the RELAP5 code in a higher level computing environment that allows for the on-line calculation of new variables, balances, verifications and information in addition to those available in the code output files. The tool will comprise a number of computer programs linked with each other according to a given standard for code connection. Such a standard is the object of reference [15].

In a finer detail, the tool will be capable of:

- Finding errors in the input file.
- Determining the local contribution of the several contributors to the global result of the code.
- Checking the consistency of the calculation with the original equations.
- Comparing different codes, models and even machines and compilers. This may help the user to ensure that his assumptions to interpret the code results are valid at least in the particular transient.

As a summary, the proposed tool will execute what we shall *call in-depth postprocessing* of the RELAP5 solutions, which must not be confused with the more traditional concept of postprocessing that makes reference to user friendly output visualization.

The analysis tool will comprise the following codes:

- The modular simulation language BABIECA [20], which have been coupled to RELAP5, providing new computation capabilities.
- The Parallel Virtual Machine (PVM) [12, 11] message passing library functions. These functions will perform the information exchange between RELAP5 and BABIECA at every time step, so the latter code can check the solution computed by the former one.
- The graphical visulization program XMGR5 [40, 22], which will be used to plot the results obtained by both RELAP5 and the analysis tool.
- The standard UNIX, non-interactive text processor AWK, which will build automatically the BABIECA input file needed to execute the analysis of a given RELAP5 case, once the user has specified the hydraulic subsystems and control volumes to be checked.

II. Verifications implemented in the analysis tool

II. Verifications implemented in the analysis tool

II.1. Introduction

This chapter shows the verifications and additional computations that have been implemented in tha analysis tool up to now. These verifications will enable us to achieve the goals stated in chapter 1, i.e., interpretation of results, assurance of the quality of the solution computed by RELAP5, fulfillment of the balance equations, comparison or different models and machines, etc.

The starting point for the verifications is the following set of time and space averaged equations for the hydrodynamic section of the RELAP5 code. The derivation of these equations can be found in [17, 33]. The nomenclature used hereinafter can be found in appendix A.

- Conservation of mass:

$$\frac{\partial}{\partial t}\left(\alpha_k A \rho_k\right)+\frac{\partial}{\partial x}\left(\alpha_k A \rho_k \upsilon_k\right)=A\Gamma_k \tag{2.1}$$

- Conservation of momentum:

$$\begin{aligned}&\frac{\partial}{\partial t}\left(\alpha_k A \rho_k \upsilon_k\right)+\frac{\partial}{\partial x}\left(\alpha_k A \rho_k \upsilon_k^2\right)=-\alpha_k A\frac{\partial P}{\partial x}+\alpha_k \rho_k B_x A\\&-\alpha_k\rho_k A\mathrm{FW}_k\upsilon_k+A\Gamma_k\upsilon_\mathrm{i}-\alpha_k\rho_k A\mathrm{FI}_k\left(\upsilon_k-\upsilon_{k'}\right)\\&-C\alpha_k\alpha_{k'}\rho_\mathrm{m}A\left[\frac{\partial}{\partial t}\left(\upsilon_k-\upsilon_{k'}\right)+\upsilon_{k'}\frac{\partial \upsilon_k}{\partial x}-\upsilon_k\frac{\partial \upsilon_{k'}}{\partial x}\right]\\&-\frac{1}{2}K\alpha_k\rho_k\left|\upsilon_k\right|\upsilon_k\delta\left(x-x_0\right)\end{aligned} \tag{2.2}$$

- Conservation of energy:

$$\begin{aligned}&\frac{\partial}{\partial t}\left(\alpha_k A \rho_k u_k\right)+\frac{\partial}{\partial x}\left(\alpha_k A \rho_k u_k \upsilon_k\right)=-AP\frac{\partial \alpha_k}{\partial t}-P\frac{\partial}{\partial x}\left(\alpha_k A\upsilon_k\right)\\&+AQ_{\mathrm{w}k}+AQ_{\mathrm{i}k}+A\Gamma_{\mathrm{i}k}h_k^*+A\Gamma_{\mathrm{w}k}h'_k+A\mathrm{DISS}_k\end{aligned} \tag{2.3}$$

In the following sections the new equations used to analyse the RELAP5 results, both in analytical and discretized form, will be derived.

II.2. Mass conservation equation

The RELAP5 code currently checks the conservation of the mixture mass in full thermalhydraulic systems. This way of controlling the calculation can fail because of two reasons:

- Positive mass errors in certain control volumes may be compensated by negative mass errors in other volumes. In other words, the mass may be globally well computed, although the local distribution may be wrong.
- Relatively large mass errors in the vapor phase may be masked when compared to the total mass, since the mass of liquid is in most cases much larger.

As a conclusion, checking of the mass errors in individual control volumes and small subsystems, for each pase, seems to be necessary. However, we must say that the first item has already been implemented in the 3.2.2beta version of RELAP5.

II.2.1. Analytical derivation

If we integrate equation (2.1) along a thermalhydraulic system of lenght L we obtain:

$$\frac{dm_k}{dt} = \int_0^L A\Gamma_k dx + W_k^{\text{in}} - W_k^{\text{out}} \tag{2.4}$$

where

- $m_k \equiv \int_0^L \alpha_k A \rho_k dx$ is the mass of phase k in the system and
- $W_k \equiv \alpha_k A \rho_k \upsilon_k$ is the mass flow.

Integrating with respect to time we obtain the integral form:

$$m_k(t) = m_k(0) + \int_0^t \left(\int_0^L A\Gamma_k dx + W_k^{\text{in}} - W_k^{\text{out}} \right) dt \tag{2.5}$$

II.2.2. Discretized form

The fulfillment of the mass conservation equation in a RELAP5 calculation must be checked using a discretized for of equation (2.5), since the output information is only available at given points of the spatio-temporal mesh (nodes of the spatial discretization and time steps.) The mass in the thermalhydraulic system to be checked is approximated as follows:

$$m_k \approx \sum_{l=1}^{N} \alpha_k^l V^l \rho_k^l \tag{2.6}$$

where the superindex l ranges over the control volumes in the subsystem. In a similar way the space integral in the right hand side of equation (5) can be approximated by

$$\int_0^L A\Gamma_k dx \approx \sum_{l=1}^{N} V^l \Gamma_k^l \tag{2.7}$$

The inlet mass flow is computed as

$$W_k^{\text{in}} \equiv \alpha_k^{\text{in}} A^{\text{in}} \rho_k^{\text{in}} \upsilon_k^{\text{in}} \tag{2.8}$$

If several junctions are attached to the inlet side of the control volume, the following definitions hold, according to the flow mixing scheme described on section III.1.7 of [37]:

$$\alpha_k^{\text{in}} \equiv \frac{\sum_{j=1}^{M} \alpha_k^j A^j}{\sum_{j=1}^{M} A^j} \tag{2.9}$$

$$\rho_k^{\text{in}} \equiv \frac{\sum_{j=1}^{M} \alpha_k^j A^j \rho_k^j}{\sum_{j=1}^{M} \alpha_k^j A^j} \tag{2.10}$$

$$\upsilon_k^{\text{in}} \equiv \frac{\left(\sum_{j=1}^{M} \alpha_k^j A^j \rho_k^j \upsilon_k^j\right)\left(\sum_{j=1}^{M} A^j\right)}{\left(\sum_{j=1}^{M} \alpha_k^j A^j \rho_k^j\right) A^{\text{in}}} \tag{2.11}$$

The superindex j ranges over all the junctions attached to the inlet face of control volume l. A^{in} is the inlet area. Similar definitions hold for the outlet face.

The discretized form of the mass conservation equation remains then as follows:

$$\frac{dm_k}{dt} \approx \sum_{l=1}^{N} V^l \Gamma_k^l + W_k^{\text{in}} - W_k^{\text{out}} \tag{2.12}$$

The right hand side of this equation may be integrated with respect to time with a trapezoidal rule. Its comparison with (2.6) provides the way to check the conservation of the mass of each phase.

II.3. Mixture momentum equation

Checking this equation could seem trivial at first sight, since it is one of the original equations solved by RELAP5. However, it will be useful to ensure that the pressure drop along a pipe equals the pressure difference between the inlet and outlet sides.

The sum of the two equations (2.2) yields:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(A\rho_{\text{m}}\upsilon_{\text{m}}\right) + \frac{\partial}{\partial x}\left(A\rho_{\text{m}}\upsilon_{\text{m}}^2\right) = -\frac{\partial}{\partial x}\left(\frac{\alpha_g \alpha_f \rho_g \rho_f A}{\rho_{\text{m}}}\left(\upsilon_g - \upsilon_f\right)^2\right) \\ &-A\frac{\partial P}{\partial x} + \rho_{\text{m}} A B_x - \alpha_g \rho_g A \text{FW}_g \upsilon_g - \alpha_f \rho_f A \text{FW}_f \upsilon_f \\ &-\frac{1}{2} K \alpha_g \rho_g \left|\upsilon_g\right| \upsilon_g \delta\left(x - x_0\right) - \frac{1}{2} K \alpha_f \rho_f \left|\upsilon_f\right| \upsilon_f \delta\left(x - x_0\right) \end{aligned} \tag{2.13}$$

where mixture density and mixture velocity are respectively defined as:

$$\rho_{\mathrm{m}} = \alpha_g \rho_g + \alpha_f \rho_f \tag{2.14}$$

$$v_{\mathrm{m}} = \frac{\alpha_g \rho_g v_g + \alpha_f \rho_f v_f}{\rho_{\mathrm{m}}} \tag{2.15}$$

and x_0 stands for the location of the local pressure drop.

The convective term $\frac{\partial}{\partial x}\left(\frac{\alpha_g \alpha_f \rho_g \rho_f A}{\rho_{\mathrm{m}}}\left(v_g - v_f\right)^2\right)$ accounts for the difference of the two fluid model with respect to the homogeneous equilibrium model.

Integrating (2.13) along the direction we get:

$$\begin{aligned}
&\frac{\partial}{\partial t}\int_0^L A\rho_{\mathrm{m}} v_{\mathrm{m}} dx + A\rho_{\mathrm{m}} v_{\mathrm{m}}^2\Big|^{\mathrm{out}} - A\rho_m v_m^2\Big|^{\mathrm{in}} \\
&\frac{\alpha_g \alpha_f \rho_g \rho_f A}{\rho_{\mathrm{m}}}\left(v_g - v_f\right)^2\Big|^{\mathrm{in}} - \frac{\alpha_g \alpha_f \rho_g \rho_f A}{\rho_{\mathrm{m}}}\left(v_g - v_f\right)^2\Big|^{\mathrm{out}} \\
&-\int_0^L A\frac{\partial P}{\partial x}dx + \int_0^L \rho_{\mathrm{m}} A B_x dx \\
&-\int_0^L \alpha_g \rho_g A\mathrm{FW}_g v_g dx - \int_0^L \alpha_f \rho_f A\mathrm{FW}_f v_f dx \\
&-\frac{1}{2}K\alpha_g \rho_g \left|v_g\right| v_g - \frac{1}{2}K\alpha_f \rho_f \left|v_f\right| v_f
\end{aligned} \tag{2.16}$$

We can assume that the cross-sectional is uniform within a control volume. Furthermore, if the integrands are also considered as uniform within the control volume, the previous equation becomes:

$$\begin{aligned}
&P^{\mathrm{in}} - P^{\mathrm{out}}\frac{\partial}{\partial t}\left(\rho_{\mathrm{m}} v_{\mathrm{m}} L\right) + \rho_{\mathrm{m}} v_{\mathrm{m}}^2\Big|^{\mathrm{out}} - \rho_{\mathrm{m}} v_{\mathrm{m}}^2\Big|^{\mathrm{in}} \\
&+\frac{\alpha_g \alpha_f \rho_g \rho_f}{\rho_{\mathrm{m}}}\left(v_g - v_f\right)^2\Big|^{\mathrm{out}} - \frac{\alpha_g \alpha_f \rho_g \rho_f}{\rho_{\mathrm{m}}}\left(v_g - v_f\right)^2\Big|^{\mathrm{in}} \\
&\rho_{\mathrm{m}} Hg + \alpha_g \rho_g L\mathrm{FW}_g v_g + \alpha_f \rho_f L\mathrm{FW}_f v_f \\
&-\frac{1}{2}\frac{K}{A}\alpha_g \rho_g \left|v_g\right| v_g - \frac{1}{2}\frac{K}{A}\alpha_f \rho_f \left|v_f\right| v_f
\end{aligned} \tag{2.17}$$

where the body force term has been transformed as follows:

$$\rho_{\mathrm{m}} L B_x = \rho_{\mathrm{m}} Lg\,cos(\theta) = \rho_{\mathrm{m}} Hg \tag{2.18}$$

The right hand side of the previous equation, computed at every time step with the output information provided by RELAP5, can be compared with the left hand side. Any discrepancy between both sides of the equation is an indicator of error in the solution.

II.4. Pressure and volume equations

These are new integral balances, derived from the mass and energy equations. It provides a new way of checking the RELAP5 results and interpreting the pressure and level evolution in TH systems.

II.4.1. Analytical derivation

First of all, we derive the enthalpy equation from the energy equation (2.3), taking into account that $h_k = u_k P \vartheta_k$ and neglecting hereinafter the dissipation term, which is taken into account by RELAP5 only if a control volume contains a pump:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(\alpha_k A \rho_k h_k\right) + \frac{\partial}{\partial x}\left(\alpha_k A \rho_k h_k \upsilon_k\right) = \alpha_k A \frac{\partial P}{\partial t} + \alpha_k A \upsilon_k \frac{\partial P}{\partial x} \\ &+ A Q_{\mathrm{w}k} + A Q_{\mathrm{w}k} + A Q_{\mathrm{w}k} + A \Gamma_{\mathrm{i}k} h_k^* + A \Gamma_{\mathrm{w}k} h'_k \end{aligned} \tag{2.19}$$

Taking into account the relationship

$$\frac{\partial h_k}{\partial t} + \upsilon_k \frac{\partial h_k}{\partial x} = \left[\frac{\partial \rho_k}{\partial t} + \upsilon_k \frac{\partial \rho_k}{\partial x}\right]\left(\frac{\partial h_k}{\partial \rho_k}\right)_P + \left[\frac{\partial P}{\partial t} + \upsilon_k \frac{\partial P}{\partial x}\right]\left(\frac{\partial h_k}{\partial P}\right)_{\rho_k} \tag{2.20}$$

the enthalpy conservation equation (2.19) is transformed into

$$\begin{aligned} &\alpha_k A \left[\frac{\partial \ln \rho_k}{\partial t} + \upsilon_k \frac{\partial \ln \rho_k}{\partial x}\right] = \\ &\alpha_k A \vartheta_k \left[\frac{\partial P}{\partial t} + \upsilon_k \frac{\partial P}{\partial x}\right]\left(\frac{\partial \rho_k}{\partial h_k}\right)_P \left[\vartheta_k - \left(\frac{\partial h_k}{\partial P}\right)_{\rho_k}\right] \\ &+ A \vartheta_k^2 \left(\frac{\partial \rho_k}{\partial h_k}\right)_P \left[Q_{\mathrm{w}k} + Q_{\mathrm{i}k} + \Gamma_{\mathrm{i}k} h_k^* + \Gamma_{\mathrm{w}k} h'_k - \Gamma_k h_k\right] \end{aligned} \tag{2.21}$$

The last two factors of the first term in the right hand side of equation (2.21) can be simplified as follows:

$$\begin{aligned} &\left(\frac{\partial \rho_k}{\partial h_k}\right)_P \left[\vartheta_k - \left(\frac{\partial h_k}{\partial P}\right)_{\rho_k}\right] = \vartheta_k \left(\frac{\partial \rho_k}{\partial h_k}\right)_P + \left(\frac{\partial \rho_k}{\partial P}\right)_{h_k} = \\ &\left(\frac{\partial h_k}{\partial P}\right)_{s_k} \left(\frac{\partial \rho_k}{\partial h_k}\right)_P + \left(\frac{\partial \rho_k}{\partial P}\right)_{h_k} = \left(\frac{\partial \rho_k}{\partial P}\right)_{s_k} \end{aligned} \tag{2.22}$$

where the following relatioships have been used:

$$\left(\frac{\partial x}{\partial y}\right)_z\left(\frac{\partial y}{\partial z}\right)_x\left(\frac{\partial z}{\partial x}\right)_y=-1$$

$$\left(\frac{\partial x}{\partial y}\right)_z=\left(\frac{\partial x}{\partial y}\right)_t+\left(\frac{\partial t}{\partial y}\right)_z\left(\frac{\partial x}{\partial t}\right)_y$$

$$\rho_k=\left(\frac{\partial P}{\partial h_k}\right)_{s_k}$$

If the *isobaric expansion coefficient* is defined as

$$\pi_k\equiv\left(\frac{\partial \vartheta_k}{\partial h_k}\right)_P \tag{2.23}$$

and the *isentropic compressibility coefficient* as

$$\gamma_k\equiv\left(\frac{\partial \ln P}{\partial \ln \rho_k}\right)_{s_k} \tag{2.24}$$

the final form of the enthalpy conservation equation is obtained:

$$\begin{aligned}&\alpha_k A\left[\frac{\partial \ln\rho_k}{\partial t}+v_k\frac{\partial \ln\rho_k}{\partial x}\right]=\frac{\alpha_k}{\gamma_k}A\left[\frac{\partial \ln P}{\partial t}+v_k\frac{\partial \ln P}{\partial x}\right]\\&A\pi_k\left[Q_{\mathrm{w}k}+Q_{\mathrm{i}k}+\Gamma_{\mathrm{i}k}h_k^*+\Gamma_{\mathrm{w}k}h'_k-\Gamma_k h_k\right]\end{aligned} \tag{2.25}$$

Moreover, if the derivatives in the mass conservation (2.1) are expanded, and the whole equation is divided by ρ_k, we obtain

$$\frac{\partial}{\partial t}(\alpha_k A)+\alpha_k A\frac{\partial \ln\rho_k}{\partial t}+\frac{\partial}{\partial x}(\alpha_k A v_k)+\alpha_k A v_k\frac{\partial \ln\rho_k}{\partial x}=A\frac{\Gamma_k}{\rho_k} \tag{2.26}$$

The substitution of (2.26) into (2.25) and the integration along the x axis yields the combined pressure and level equation for phase k:

$$\frac{dV_k}{dt}+C_k\frac{d\langle \ln P\rangle_k}{dt}=J_k+Sw_k+D_k+Ac_k \tag{2.27}$$

The following variables have been defined in the previous equation:

- V_k is the volume occupied by the phase k,
- $C_k\equiv\int_0^L A\frac{\alpha_k}{\gamma_k}dx$ is the phase k system compressibility,

- $\langle \ln P \rangle_k \equiv \dfrac{\int_0^L A \frac{\alpha_k}{\gamma_k} \ln P dx}{\int_0^L A \frac{\alpha_k}{\gamma_k} dx}$,

- $J_k \equiv \alpha_k A \upsilon_k \big|^{\text{in}} - \alpha_k A \upsilon_k \big|^{\text{out}}$ is the difference between the inlet and outlet flows in the TH system,

- $Sw_k \equiv \int_0^L A \frac{\Gamma_k}{\rho_k^{\text{sat}}} dx$ is the expansion/contraction due to phase k appearance/dissappearance,

- $D_k \equiv \int_0^L A\left[\pi_k\left(Q_{\text{w}k} + Q_{\text{i}k} + \Gamma_{\text{i}k} h_k^* + \Gamma_{\text{w}k} h'_k - \Gamma_k h_k\right) + \Gamma_k\left(\vartheta_k - \vartheta_k^{\text{sat}}\right)\right] dx$

  is the expansion/contraction of phase k due to enthalpy variations,

- $Ac_k \equiv \int_0^L A\left[\left(\ln P - \langle \ln P \rangle_k\right) \frac{\partial}{\partial t}\left(\frac{\alpha_k}{\gamma_k}\right) - \frac{\alpha_k}{\gamma_k} \upsilon_k \frac{\partial \ln P}{\partial x}\right]$ is a term dominated by pressure grandients.

The relation

$$\int_0^L A \frac{\alpha_k}{\gamma_k}\left(\frac{\partial \ln P}{\partial t} + \upsilon_k \frac{\partial \ln P}{\partial x}\right) dx = C_k \frac{d\langle \ln P \rangle_k}{\partial t} - Ac_k \tag{2.28}$$

has been also taken into account.

If the two equations (2.27) are summed, one for each phase, the following pressure equation is obtained:

$$\frac{d\langle \ln P \rangle}{\partial t} = \frac{\sum_k J_k + Sw_k + D_k + Ac_k}{\sum_k C_k} \tag{2.29}$$

where

$$\frac{d\langle \ln P \rangle}{\partial t} = \frac{\sum_k C_k \frac{d\langle \ln P \rangle_k}{dt}}{\sum_k C_k} \tag{2.30}$$

Equation (2.29) shows that the pressure evolution in a TH system is the result of the accomodation of volume changes. The greater the compressibility of the system, the smaller the impact of those changes in the pressure. The sources of volume change are the inlet and outlet volumetric flows, the swelling of the phases, the thermal expansions and changes caused by the pressure distribution along the system.

Assuming that $\frac{d\langle \ln P \rangle_k}{dt} \approx \frac{d\langle \ln P \rangle}{dt}$, we obtain the following level equation:

$$\frac{dV_f}{dt} = J_f + Sw_f + D_f + Ac_f - C_f \frac{d\langle \ln P \rangle}{dt} \tag{2.31}$$

The equation above clearly shows that variations of the volume of liquid in the system, which is closely related to the level, can be caused by a mismatch between the inlet and outlet volumetric flows, by the appearance/dissapearance of liquid phase, by the thermal expansion of the liquid, by spatial variations of the pressure and, finally, by temporal variations of the global pressure. The greater the compressibility of the liquid, the greater the influence of the variations of the global pressure in the evolution of the level.

II.4.2. Discretized form

The terms in the right hand side of the pressure and level equation can be discretized as follows:

$$C_k \approx \sum_{l=1}^{N} \frac{\alpha_k^l}{\gamma_k^l} V^l \tag{2.32}$$

$$Sw_k \approx \sum_{l=1}^{N} V^l \frac{\Gamma_k^l}{\rho_k^{\mathrm{sat}^l}} \tag{2.33}$$

$$D_k \approx \sum_{l=1} V^l A\left[\pi_k^l\left(Q_{\mathrm{w}k}^l + Q_{\mathrm{i}k}^l + \Gamma_{\mathrm{i}k}^l h_k^{*^l} + \Gamma_{\mathrm{w}k}^l h'^{\,l}_k - \Gamma_k^l h_k^l\right) + \Gamma_k^l\left(\vartheta_k^l - \vartheta_k^{\mathrm{sat}^l}\right)\right] \tag{2.34}$$

$$Ac_k \approx \sum_{l=1}^{N} V^l\left(\ln P^l - \langle \ln P \rangle_k\right)\frac{\partial}{\partial t}\left(\frac{\alpha_k^l}{\gamma_k^l}\right) + \sum_{l=1}^{N} A^l \frac{\alpha_k^l}{\gamma_k^l} \upsilon_k^l \ln\left(\frac{P^{\mathrm{in}^l}}{P^{\mathrm{out}^l}}\right) \tag{2.35}$$

The volumetric flow J_k is computed with the equations (2.9) and (2.11).

The partial derivative with respect to time in equation (2.35) is estimated by finite differences:

$$\frac{\partial}{\partial t}\left(\frac{\alpha_k^l}{\gamma_k^l}\right) \approx \frac{1}{\Delta t}\left(\frac{\alpha_k^{l,n}}{\gamma_k^{l,n}} - \frac{\alpha_k^{l,n-1}}{\gamma_k^{l,n-1}}\right) \tag{2.36}$$

where the superindices n and n–1 stand for two successive time steps.

The right hand side of equation (2.29) is integrated with respect to time with a BABIECA module that implements the trapezoidal rule. The result is compared to

$$\langle \ln P \rangle_k \approx \frac{\sum_{l=1}^{N} V^l \frac{\alpha_k^l}{\gamma_k^l} \ln P^l}{\sum_{l=1}^{N} V^l \frac{\alpha_k^l}{\gamma_k^l}} \tag{2.37}$$

Similarly, the right hand side of equation (2.31) is numerically integrated and compared with

$$V_f \approx \sum_{l=1}^{N} \alpha_k^l V^l \tag{2.38}$$

II.5. Non-equilibrium mass equation

This integral balance, derived also from the mass and energy equations, has been developed with two purposes:

- Provide an alternative way to check the consistency of the solution with the conservation equations.
- Under certain hipoteses, a form of the conservation equations that shows the tendence of two-phase systems towards thermal equilibrium can be obtained. This tendence is specially strong in the emerging phase, even if the constitutive equations do not indicate this behavior in principle. If this tendence towards equilibrium in the emerging phase is confirmed, simplified models can take advantage of that feature. This assumption is used, for instance, in the HIPA code [43], which leads to more simplified models. Checking of the equilibrium hipothesis in the emerging phase in a wide variety of transients would confirm the validity of those simplified models, in which the two phases are decoupled from each other. More efficient numerical schemes can then be devised.

II.5.1. Analytical derivation

The continuity equation can be rewritten as follows:

$$\frac{\partial}{\partial t}(\alpha_k A\rho_k)+\frac{\partial}{\partial x}(\alpha_k A\rho_k v_k)=\rho_k\left[\frac{\partial}{\partial t}(\alpha_k A)+\frac{\partial}{\partial x}(\alpha_k A v_k)\right]+\alpha_k A\left[\frac{\partial \rho_k}{\partial t}+v_k\frac{\partial \rho_k}{\partial x}\right]=A\Gamma_k \tag{2.39}$$

Defining

$$\delta\rho_k^{\mathrm{nq}} \equiv \rho_k^{\mathrm{sat}} - \rho_k \tag{2.40}$$

$$A\Delta\Gamma_k \equiv \frac{\partial}{\partial t}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}v_k\right) \tag{2.41}$$

and taking into account the continuity equation (2.1), the saturation density fulfills the following conservation equation:

$$\begin{aligned}&\frac{\partial}{\partial t}\left(\alpha_k A\rho_k^{\mathrm{sat}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\rho_k^{\mathrm{sat}}v_k\right)=\frac{\partial}{\partial t}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}v_k\right)\\&+\frac{\partial}{\partial t}\left(\alpha_k A\rho_k\right)+\frac{\partial}{\partial x}\left(\alpha_k A\rho_k v_k\right)=A\left(\Gamma_k+\Delta\Gamma_k\right)\end{aligned} \tag{2.42}$$

If we expand the left hand side of the previous equation we get:

$$\rho_k^{\mathrm{sat}}\left[\frac{\partial}{\partial t}\left(\alpha_k A\right)+\frac{\partial}{\partial x}\left(\alpha_k A v_k\right)\right]+\alpha_k A\left[\frac{\partial \rho_k^{\mathrm{sat}}}{\partial t}+v_k\frac{\partial \rho_k^{\mathrm{sat}}}{\partial x}\right]=A\left(\Gamma_k+\Delta\Gamma_k\right) \qquad (2.43)$$

Substracting equations (2.43) and (2.39), previously divided by ρ_k^{sat} and ρ_k, respectively, we obtain

$$\alpha_k A\left[\frac{\partial \ln\rho_k^{\mathrm{sat}}}{\partial t}+v_k\frac{\partial \ln\rho_k^{\mathrm{sat}}}{\partial x}+\frac{\partial \ln\rho_k}{\partial t}-v_k\frac{\partial \ln\rho_k^{\mathrm{sat}}}{\partial x}\right]=A\Gamma_k\left(\vartheta_k^{\mathrm{sat}}-\vartheta_k\right)+A\frac{\Delta\Gamma_k}{\rho_k^{\mathrm{sat}}} \qquad (2.44)$$

If we define the compressibility along the saturation line as

$$\lambda_k^{\mathrm{sat}}\equiv\frac{d\ln P}{d\ln\rho_k^{\mathrm{sat}}} \qquad (2.45)$$

and we use equations (2.25) and (2.41), we get

$$\frac{\partial}{\partial t}\left(\alpha_k A\delta\rho_k^{\mathrm{np}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\delta\rho_k^{\mathrm{np}}v_k\right)=\rho_k^{\mathrm{sat}}\left[\left(\frac{1}{\lambda_k^{\mathrm{sat}}}-\frac{1}{\gamma_k}\right)\alpha_k A\left(\frac{\partial \ln P}{\partial t}+\frac{\partial \ln P}{\partial x}\right)+dx\right] \qquad (2.46)$$

where d_k is the integrand of the thermal expansion term, i.e., $d_k\equiv\frac{\partial D_k}{\partial x}$.

Defining

$$d_k^{\mathrm{eq}}\equiv-\left(\frac{1}{\lambda_k^{\mathrm{sat}}}-\frac{1}{\gamma_k}\right)\alpha_k A\left(\frac{\partial \ln P}{\partial t}+\frac{\partial \ln P}{\partial x}\right) \qquad (2.47)$$

and

$$d_k^{\mathrm{nq}}\equiv d_k-d_k^{\mathrm{eq}} \qquad (2.48)$$

we obtain the following differential form of the non-equilibrium mass conservation equation:

$$\frac{\partial}{\partial t}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}v_k\right)=\rho_k^{\mathrm{sat}}d_k^{\mathrm{nq}} \qquad (2.49)$$

Integrating along the x axis:

$$\frac{dm_k^{\mathrm{nq}}}{\mathrm{d}t}=\int_0^L\rho_k^{\mathrm{sat}}d_k^{\mathrm{nq}}dx+W_k^{\mathrm{nq}}\Big|^{\mathrm{in}}-W_k^{\mathrm{nq}}\Big|^{\mathrm{out}} \qquad (2.50)$$

where the *non-equilibrium mass* m_k^{nq} has been defined as

$$m_k^{\mathrm{nq}} \equiv \int_0^L \alpha_k A \delta\rho_k^{\mathrm{nq}} dx \tag{2.51}$$

and the *non-equilibrium mass flow* as

$$W_k^{\mathrm{nq}} \equiv \int_0^L \alpha_k A \delta\rho_k^{\mathrm{nq}} v x \tag{2.52}$$

The definitions (2.47) and (2.48) are physically meaningful. If phase k is in equilibrium, the left hand side of equation (2.49) vanishes, yielding

$$d_k = d_k^{\mathrm{eq}} = -\left(\frac{1}{\lambda_k^{\mathrm{sat}}} - \frac{1}{\gamma_k}\right)\alpha_k A\left(\frac{\partial \ln P}{\partial t} + v_k \frac{\partial \ln P}{\partial x}\right) \tag{2.53}$$

This is the reason for the definition of (2.47). The difference between the actual term d_k and that defined as the equilibrium term is just the definition (2.48).

When phase k is in thermal equilibrium, the terms that include any of the Γ variables in the expresion of d_k vanish. d_k becomes then

$$d_k^{\mathrm{eq}} = A\pi_k\left(Q_{\mathrm{w}k} + Q_{\mathrm{i}k}^{\mathrm{eq}}\right) \tag{2.54}$$

where Q_{ik}^{eq} is the interfacial heat in those conditions. The heat transferred to phase is then donnored to the other phase k through the interfacial surface, subtracting the heat necessary to accomodate the new saturation conditions imposed by the spatio-temporal variations of the pressure. The contribution of the pressure is accounted for with the term d_k^{eq}.

II.5.2. Discretized form

The integral in the right hand side of equation (2.50) is approximated, within a control volume l, as follows:

$$\begin{aligned}
&\int_0^{L^l} \rho_k^{\mathrm{sat}} d_k^{\mathrm{nq}} dx \approx \rho_k^{\mathrm{sat}^l} D_k^l + \int_0^{L^l} \rho_k^{\mathrm{sat}}\left(\frac{1}{\lambda_k^{\mathrm{sat}}} - \frac{1}{\gamma_k}\right)\alpha_k A\left(\frac{\partial \ln P}{\partial t} + v_k \frac{\partial \ln P}{\partial x}\right)dx \approx \\
&\rho_k^{\mathrm{sat}^l}\left[D_k^l + \gamma_k^l\left(\frac{1}{\lambda_k^{\mathrm{sat}^l}} - \frac{1}{\gamma_k^l}\right)\int_0^{L^l} A\frac{\alpha_k}{\gamma_k}\left(\frac{\partial \ln P}{\partial t} + v_k \frac{\partial \ln P}{\partial x}\right)dx\right] = \\
&\rho_k^{\mathrm{sat}^l}\left[D_k^l + \gamma_k^l\left(\frac{1}{\lambda_k^{\mathrm{sat}^l}} - \frac{1}{\gamma_k^l}\right)\left(J_k^l + Sw_k^l + D_k^l - \frac{dV_k^l}{dt}\right)\right]
\end{aligned} \tag{2.55}$$

The relations (2.28) and (2.31) have been used in the last part of the previous equation.

The right hand side of the non-equilibrium mass equation (2.50) is approximated in a whole TH system by the sum of all the equations (2.55) that represent each control volume:

$$\begin{aligned}\int_0^L \rho_k^{\mathrm{sat}} d_k^{\mathrm{nq}} dx \approx \sum_{l=1}^{N} \rho_k^{\mathrm{sat}^l} \frac{\gamma_k^l}{\lambda_k^{\mathrm{sat}^l}} \left[\left(J_k^l + Sw_k^l + D_k^l - \frac{dV_k^l}{dt}\right)\right] \\ -\sum_{l=1}^{N} \rho_k^{\mathrm{sat}^l} \left[J_k^l + Sw_k^l - \frac{dV_k^l}{dt}\right] + W_k^{\mathrm{nq}}\Big|^{\mathrm{in}} - W_k^{\mathrm{nq}}\Big|^{\mathrm{out}}\end{aligned} \tag{2.56}$$

The saturation density in the definition of the non-equilibrium mass flows is that corresponding to the inlet (or outlet) pressure of the system. The equation (2.56), integrated with respect to time, is compared then with the following approximation of the non-equilibrium mass of phase *k* in the whole system:

$$m_k^{\mathrm{nq}} \approx \sum_{l=1}^{N} V_k^l \left(\rho_k^{\mathrm{sat}^l} - \rho_k^l\right) \tag{2.57}$$

The comparison of the two sides of equation (2.50) can be used to check the consistency of the solutions provided by the RELAP5 code, as already declared at the beginning of this section.

II.5.3. Uncoupling of the non-equilibrium mass equations

The thermal expansion term d_k in the non-equilibrium mass equation can be simplified under certain hipotheses, leading to the uncoupling of the two phases. If those hipotheses are confirmed, the uncoupled equations could be used as the basis for more efficient numerical schemes.

The term can be rewritten as follows:

$$\begin{aligned}d_k = A\left[\pi_k\left(Q_{\mathrm{w}k} + Q_{\mathrm{i}k} + \Gamma_{\mathrm{i}k} h_k^* + \Gamma_{\mathrm{w}k} h'_k - \Gamma_k h_k\right) + \Gamma_k\left(\vartheta_k - \vartheta_k^{\mathrm{sat}}\right)\right] = \\ A\pi_k\left(Q_{\mathrm{w}k} + Q_{\mathrm{i}k}^{w} + Q_{\mathrm{i}k}^{\mathrm{b}}\right) + A\Gamma_{\mathrm{i}k}\left[\pi_k\left(h_k^* - h_k\right) + \left(\vartheta_k - \vartheta_k^{\mathrm{sat}}\right)\right] \\ + A\Gamma_{\mathrm{w}k}\left[\pi_k\left(h'_k - h_k\right) + \left(\vartheta_k - \vartheta_k^{\mathrm{sat}}\right)\right]\end{aligned} \tag{2.58}$$

where both the interfacial heat and the generation rate of phase *k* have been split into their contributions in the wall and the bulk of the fluid, according to the equations

$$Q_{\mathrm{i}k} = Q_{\mathrm{i}k}^{\mathrm{w}} + Q_{\mathrm{i}k}^{\mathrm{b}} \tag{2.59}$$

$$\Gamma_k = \Gamma_{\mathrm{w}k} + \Gamma_{\mathrm{i}k} \tag{2.60}$$

If $\Gamma_{\mathrm{i}k} < 0$ then $h_k^* = h_k$ (see the definition of h_k^* in section 2.6), and the term $\Gamma_{\mathrm{i}k}$ with in the left hand side of equation (2.58) remains in the form

$$A\Gamma_{ik}\left[\pi_k\left(h_k^* - h_k\right) + \left(\vartheta_k - \vartheta_k^{\text{sat}}\right)\right] = A\Gamma_{ik}\left(\vartheta_k - \vartheta_k^{\text{sat}}\right) \tag{2.61}$$

In this case the two phases cannot be uncoupled, since the expresion of the generation rate of vapor in the bulk

$$\Gamma_{ik} = -\frac{H_{ik}\left(T^{\text{sat}} - T_k\right) + H_{ik'}\left(T^{\text{sat}} - T_{k'}\right)}{h_k^* - h_{k'}^*} \tag{2.62}$$

contains variables relative to the other phase.

On the other hand, if $\Gamma_{ik} > 0$ then $h_k^* = h_k^{\text{sat}}$, and the term Γ_{ik} with in the left hand side of equation (2.58) takes the form

$$A\Gamma_{ik}\left[\pi_k\left(h_k^* - h_k\right) + \left(\vartheta_k - \vartheta_k^{\text{sat}}\right)\right] = A\Gamma_{ik}\left[\pi_k\left(h_k^{\text{sat}} - h_k\right) + \left(\vartheta_k - \vartheta_k^{\text{sat}}\right)\right] \tag{2.63}$$

If the specific volumes are expanded in a Taylor series:

$$\vartheta_k^{\text{sat}} = \vartheta_k + \left(\frac{\partial \vartheta_k}{\partial h_k}\right)_P \left(h_k^{\text{sat}} - h_k\right) + \frac{1}{2}\left(\frac{\partial^2 \vartheta_k}{\partial h_k^2}\right)_P \left(h_k^{\text{sat}} - h_k\right)^2 + \ldots \tag{2.64}$$

Neglecting the second and higher order terms in equation (2.64), and taking into account the definition of π_k, equation (2.63) becomes trivial, simplifying the non-equilibrium mass equation.

Moreover, if $\Gamma_{wk} > 0$ then $h'_k = h_k^{\text{sat}}$ and the term Γ_{wk} in the right hand side of equation (2.58) can also be neglected. In this case the non-equilibrium mass equation remains as follows:

$$\frac{\partial}{\partial t}\left(\alpha_k A \delta\rho_k^{\text{nq}}\right) + \frac{\partial}{\partial x}\left(\alpha_k A \delta\rho_k^{\text{nq}} v_k\right) = \rho_k^{\text{sat}} A \pi_k \left(Q_{wq} + Q_{iq}^{\text{w}} + Q_{iq}^{\text{b}}\right) - \rho_k^{\text{sat}} d_k^{\text{eq}} \tag{2.65}$$

Since we assume that $\Gamma_{wk} > 0$, this phase appears in equilibrium in the wall, and hence $Q_{ik}^{\text{w}} = 0$. The interfacial heat in the bulk is defined by the expresion:

$$Q_{ik}^{\text{b}} = H_{ik}\left(T^{\text{sat}} - T_k\right) \tag{2.66}$$

Expanding the previous equation by a Taylor series up to the first order:

$$Q_{ik}^{\text{b}} \approx H_{ik} \delta\rho_k^{\text{nq}} \left(\frac{\partial T}{\partial \rho}\right)_P^{\text{sat}} = -\frac{H_{ik}}{\rho_k^{\text{sat}} \beta_k^{\text{sat}}} \delta\rho_k^{\text{nq}} \tag{2.67}$$

The final form of the non-equilibrium mass equation, assuming that $\Gamma_{ik} > 0$ y $\Gamma_{wk} > 0$, is:

$$\begin{aligned} \frac{\partial}{\partial t}\left(\alpha_k A \delta\rho_k^{\text{nq}}\right) + \frac{\partial}{\partial x}\left(\alpha_k A \delta\rho_k^{\text{nq}} v_k\right) &\approx -\frac{H_{ik}}{\beta_k^{\text{sat}}} \pi_k A \delta\rho_k^{\text{nq}} + \rho_k^{\text{sat}} A \pi_k Q_{wk} \\ &+ \rho_k^{\text{sat}} \left(\frac{1}{\lambda_k^{\text{sat}}} - \frac{1}{\gamma_k}\right) \alpha_k A \left(\frac{\partial \ln P}{\partial t} + v_k \frac{\partial \ln P}{\partial x}\right) \end{aligned} \tag{2.68}$$

where all the variables are defined for the phase k itself, demonstrating the uncoupling from the other phase.

Let's remind that the validity of this model is guarantedd only under the following assumptions:

- $\Gamma_{ik}\left[\pi_k\left(h_k^{\mathrm{sat}}-h_k\right)+\left(\vartheta_k-\vartheta_k^{\mathrm{sat}}\right)\right]\approx 0$
- $\Gamma_{wk}\left[\pi_k\left(h_k^{\mathrm{sat}}-h_k\right)+\left(\vartheta_k-\vartheta_k^{\mathrm{sat}}\right)\right]\approx 0$
- $T_k^{\mathrm{sat}}-T_k \approx -\dfrac{\delta\rho_k^{\mathrm{nq}}}{\rho_k^{\mathrm{sat}}\beta_k^{\mathrm{sat}}}$

On the other hand, if $\Gamma_{wk}<0$ then $h'_k=h_k$ and the terms that involve the interfacial heat in the wall and the generation rate of phase k in the wall become

$$\begin{aligned}&A\left[\pi_k Q_{ik}^{\mathrm{w}}+\Gamma_{wk}\left(\vartheta_k-\vartheta_k^{\mathrm{sat}}\right)\right]=A\Gamma_{wk}\left(\left(\vartheta_k-\vartheta_k^{\mathrm{sat}}\right)-\pi_k\left(h'_k-h'_{k'}\right)\right)=\\&A\Gamma_{wk}\left(\left(\vartheta_k-\vartheta_k^{\mathrm{sat}}\right)-\pi_k\left(h_k-h_{k'}^{\mathrm{sat}}\right)\right)=\end{aligned}\tag{2.69}$$

where the following relation taken from the page 4-130 of reference [38] has been used:

$$\Gamma_{wk}=-\frac{Q_{ik}^{\mathrm{w}}}{h'_k-h'_{k'}}\tag{2.70}$$

The reference [38] also suggests on page 4-132 the following expresions for the generation rate of vapor and liquid in the wall:

$$\Gamma_{wg}=\frac{Q_{wg}}{\max\left(h_g-h_f^{\mathrm{sat}},10^4\,J/Kg\right)}\qquad\text{(wall condensation)}\tag{2.71}$$

$$\Gamma_{wf}=-\frac{Q_{wf}}{\max\left(h_g^{\mathrm{sat}}-h_f,10^4\,J/Kg\right)}\mathrm{Mul}\qquad\text{(wall boiling)}\tag{2.72}$$

Assuming that $\left|h_k-h_{k'}^{\mathrm{sat}}\right|>10^4\,\mathrm{J/Kg}$, and taking into account that $\vartheta_k-\vartheta_k^{\mathrm{sat}}\approx\dfrac{\delta_l^{\mathrm{nq}}}{\delta_l^{\mathrm{sat}^2}}$, the following expresion of the non-equilibrium mass equation is obtained:

$$\begin{aligned}&\frac{\partial}{\partial t}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}\right)+\frac{\partial}{\partial x}\left(\alpha_k A\delta\rho_k^{\mathrm{nq}}\upsilon_k\right)\approx\left[\frac{Q_{wk}\mathrm{Mul}}{\rho_k^{\mathrm{sat}}\left(h_k-h_{k'}^{\mathrm{sat}}\right)}-\pi_k\frac{H_{ik}}{\beta_k^{\mathrm{sat}}}\right]A\delta\rho_k^{\mathrm{nq}}\\&+\rho_k^{\mathrm{sat}}A\pi_k Q_{wq}\left(1-\mathrm{Mul}\right)+\rho_k^{\mathrm{sat}}\left(\frac{1}{\lambda_k^{\mathrm{sat}}}-\frac{1}{\gamma_k}\right)\alpha_k A\left(\frac{\partial\ln P}{\partial t}+\upsilon_k\frac{\partial\ln P}{\partial x}\right)\end{aligned}\tag{2.73}$$

Like in the case in which $\Gamma_{wk} > 0$, the previous equation is decoupled from the other phase, since $h_{k'}^{\text{sat}}$ depends only on the pressure. The multiplier depends only on the pressure as well. This multiplier equals 1 for the vapor phase. In this case the first term with the variable Q_{wk} in equation (2.73) vanishes.

It is convenient to remind here that these conclusions are valid only for the phase in which $\Gamma_{ik} > 0$ and $\Gamma_{wk} < 0$, under the following assumptions:

- $\Gamma_{ik}\left[\pi_k\left(h_k^{\text{sat}} - h_k\right) + \left(\vartheta_k - \vartheta_k^{\text{sat}}\right)\right] \approx 0$
- $T_k^{\text{sat}} - T_k \approx -\dfrac{\delta\rho_k^{\text{nq}}}{\rho_k \beta_k}$
- $\vartheta_k - \vartheta_k^{\text{sat}} \approx \dfrac{\delta_l^{\text{nq}}}{\rho_k^{\text{sat}^2}}$

Confirmation of these assumptions in a wide range of transients will allow to validate the new model based on the non-equilibrium mass equations.

II.6. Correspondence with the RELAP5 output variables

To evaluate the integral balances described in the sections II.2 to II.5 it is necessary to associate the variables involved in those balances with the output variables provided by the RELAP5 output files. The partial derivatives are obtained from the three partial derivatives obtained in RELAP5, i.e.:

- $C_P \equiv \left(\dfrac{\partial h}{\partial T}\right)_P$
- $\beta \equiv \dfrac{1}{\vartheta}\left(\dfrac{\partial \vartheta}{\partial T}\right)_P$
- $\eta \equiv \left(\dfrac{\partial \rho}{\partial P}\right)_u$

On the contrary, the partial derivatives available in the BABIECA module `r5prop`, needed to evaluate the properties along the saturation line, are the following:

- $C_P \equiv \left(\dfrac{\partial h}{\partial T}\right)_P$
- $C_\vartheta \equiv \left(\dfrac{\partial u}{\partial T}\right)_\vartheta$
- $\beta \equiv \dfrac{1}{\vartheta}\left(\dfrac{\partial \vartheta}{\partial T}\right)_P$

Several derivatives of thermodynamic properties, computed as function of the two groups on variables above, are shown in Appendix B.

The correspondences are shown below, with the RELAP5 variables on the right hand side of the equivalences.

Variables concerning control volumes

$$C_{Pg}^{l} \equiv \mathtt{csubpg}$$

$$C_{Pf}^{l} \equiv \mathtt{csubpf}$$

$$\beta_g^l \equiv \mathtt{betagg}$$

$$\beta_f^l \equiv \mathtt{betaff}$$

$$\eta_g^l \equiv \mathtt{drgdp}$$

$$\eta_f^l \equiv \mathtt{drfdp}$$

$$V^l \equiv \mathtt{vvol}$$

$$A^l \equiv \mathtt{avol}$$

$$P^l \equiv \mathtt{p}$$

$$\alpha_g^l \equiv \mathtt{voidg}$$

$$\alpha_f^l \equiv \mathtt{voidf}$$

$$\rho_g^l \equiv \mathtt{rhog}$$

$$\rho_f^l \equiv \mathtt{rhof}$$

$$\vartheta_g^l \equiv \frac{1}{\mathtt{rhog}}$$

$$\vartheta_f^l \equiv \frac{1}{\mathtt{rhof}}$$

$$u_g^l \equiv \mathtt{ug}$$

$$u_f^l \equiv \mathtt{uf}$$

$$v_g^l \equiv \mathtt{velg}$$

$$v_f^l \equiv \mathtt{velf}$$

$$\Gamma_{ig}^l \equiv \mathrm{gammai}$$

$$\Gamma_{if}^l \equiv -\mathrm{gammai}$$

$$\Gamma_{wg}^l \equiv \mathrm{gammaw}$$

$$\Gamma_{wf}^l \equiv -\mathrm{gammaw}$$

$$\Gamma_g^l \equiv \mathtt{gammaw} + \mathtt{gammai}$$

$$\Gamma_f^l \equiv -\left(\mathtt{gammaw} + \mathtt{gammai}\right)$$

$$h_g^l \equiv \mathtt{ug} + \frac{\mathtt{p}}{\mathtt{rhog}}$$

$$h_f^l \equiv \mathtt{uf} + \frac{\mathtt{p}}{\mathtt{rhof}}$$

$$h_g^{*l} \equiv \begin{cases} h_g^{\mathrm{sat}^l}(\mathtt{p}) \\ h_g^l \end{cases}$$

$$h_f^{*l} \equiv \begin{cases} h_f^{\mathrm{sat}^l}(\mathtt{p}) \\ h_f^l \end{cases}$$

$$h'^{l}_{g} \equiv \begin{cases} h_g^{\mathrm{sat}^l}(\mathtt{p}) \\ h_g^l \end{cases}$$

$$h'^{l}_{f} \equiv \begin{cases} h_f^{\mathrm{sat}^l}(\mathtt{p}) \\ h_f^l \end{cases}$$

$$H_{ig}^l \equiv \mathtt{hig}$$

$$H_{if}^{l} \equiv \texttt{hif}$$

$$T^{\mathrm{sat}^{l}} \equiv \texttt{satt}$$

$$T_{g}^{l} \equiv \texttt{tempg}$$

$$T_{f}^{l} \equiv \texttt{tempf}$$

$$Q_{wf}^{l} \equiv \frac{\texttt{q}-\texttt{qwg}}{\texttt{vvol}}$$

$$Q_{ig}^{\mathrm{w}l} \equiv \begin{cases} -\texttt{gammaw}\left(h'^{l}_{g} - h'^{l}_{f}\right) & \text{if } \texttt{gamma} < 0 \\ 0 & \text{if } \texttt{gamma} < 0 \end{cases}$$

$$Q_{if}^{\mathrm{w}l} \equiv \begin{cases} -\texttt{gammaw}\left(h'^{l}_{g} - h'^{l}_{f}\right) & \text{if } \texttt{gamma} > 0 \\ 0 & \text{if } \texttt{gamma} < 0 \end{cases}$$

$$Q_{ig}^{\mathrm{b}l} \equiv \texttt{hig}\left(\texttt{sattemp}-\texttt{tempg}\right)$$

$$Q_{if}^{\mathrm{b}l} \equiv \texttt{hif}\left(\texttt{sattemp}-\texttt{tempf}\right)$$

$$\mathrm{FW}_{g}^{l} \equiv \texttt{voidg} * \texttt{rhog} * \texttt{fwalg}$$

$$\mathrm{FW}_{f}^{l} \equiv \texttt{voidf} * \texttt{rhof} * \texttt{fwalf}$$

Variables concerning junctions

$$v_{g}^{j} \equiv \texttt{velgj}$$

$$v_{f}^{j} \equiv \texttt{velfj}$$

$$\alpha_{g}^{j} \equiv \texttt{voidgj}$$

$$\alpha_{f}^{j} \equiv \texttt{voidfj}$$

$$\rho_{g}^{j} \equiv \texttt{rhogj}$$

$$\rho_{f}^{j} \equiv \texttt{rhofj}$$

$$A^{j} \equiv \texttt{ajun}$$

$$K_{f}^{j} \equiv \texttt{formgj}$$

$$K_{r}^{j} \equiv \texttt{formfj}$$

III. Description of the analysis tool

III. Description of the analysis tool

III.1. Introduction

This chapter describes the codes and computer programs used to build the proposed analysis tool and the way they are connected to perform the in-depth postprocess of the results obtained in RELAP5 calculations.

III.2. Description of the programs used in the analysis tool

III.2.1. The RELAP5 code

RELAP5 is the thermalhydraulic code whose results will be checked with the analysis tool. Thus, it is the central code of it. Please refer to [37] for details on the code.

III.2.2. The BABIECA simulation language

BABIECA is the driver of the continuous, general purpose simulation language integrated in the package CAMPEADOR [20]. BABIECA will be the core of the analysis tool. It is the central program to which the rest of the codes will be connected. It performs the additional calculations described in chapter II.

The BABIECA simulation language is modular, which means that the problem to be solved is defined as a block diagram. Each block may be considered as a multi input-multi output relation between time-evolving variables, and represents a physical system or a mathematical procedure. The input-output relation consists of a set of differential equations plus a numerical solution algorithm (implicit). There is no common solution algorithm for all the blocks, as it is the norm in most simulation languages. Blocks are particular instances of computational entities called modules. The set of private data of the block defines it, while it is the connection among the blocks and the overall computation order that constitute the global numerical scheme. Each block has a number, defined by the user in the input file. The block outputs are numbered successively starting from the block number. The user must take care of avoiding output number overlaping. The driver routine of BABIECA manages the time step control and calls the modules sequentially in the user-defined order. When all the blocks have been successfully computed the time step is considered to be finished, and a new time step calculation begins.

Modules cover a broad range of mathematical tools, physical models and special components, and are mainly oriented, but not limited, to the solution of large thermal hydraulic networks like those appearing in nuclear power plants. Among all the modules available, only few will be used in this tool. The most important one is the module called `FUNIN`, which is a parser that processes, at each time step, a FORTRAN-like mathematical expresion dependent upon the block inputs. Other important modules are `LOGATE`, which obtains logical variables from analogical inputs, `R5STEAM`, which contains the RELAP5 water properties tables, `FINT`, which implements time dependent functions, WRITES, used to write column formatted output files and finally SNDCODE and RCVCODE, used for the connection with other codes.

III.2.3. Parallel Virtual Machine (PVM)

Parallel Virtual Machine (PVM) [11, 12] is a software for the parallelization and exchange of information between different computer processes. It runs on top of the UNIX sockets. PVM is interoperable, i.e., the computer processes to be connected can run even on different machines of different nature. It is only required that the machines be connected through the TCP/IP protocol.

PVM comprises a library of C and FORTRAN functions and a daemon program called pvmd. The library provides functions to spawn a new process from a running one, to pack data in buffers, to send these buffers, to receive them and to unpack the data. These functions, when properly inserted in the source code of the original programs, allow the developers to make the programs work concurrently to perform an overall task.

It is convenient here to make a reference to the Message Passing Interface (MPI) software [13]. It is more oriented towards parallelization of codes running in multiprocesor machines than towards distributed computing. It only can work between machines of the same type. Another drawback is the incapability to spawn processes. On the other hand, it shows a better performance in the exchange of information between processes. The University of Tennessee and the Oak Ridge National Laboratory are planning to develop a product that would support all the capabilities of both PVM and MPI, called PVMPI [9]. If this product is finally released, it is almost sure that it will become the new standard for parallelization and connection of programs.

III.2.4. The text editor AWK

AWK is a non-interactive text editor included in all standard distributions of UNIX-like operating systems. Details and reference manual can be found in any UNIX reference manual. Generally speaking, AWK is a C-like interpreted language that allows users to use and operate variables on the fly without previous definition. The AWK command are usually written in a file. These commands read information from a set of files, process the information, and generate another set of files. The file containing the AWK commands comprises three parts:

- A section called BEGIN, where the user can initialize variables and define parameters used along the rest of the file.
- A main body, in which AWK searchs for lines containing given patterns in a set of files specified by the user. Depending on the matching of these patterns, and the information contained in the files, AWK can generate new information, in form of new variables, operations between variables, etc.
- An END section, in which AWK can generate new files depending on the information generated in the main body.

Due to its special features, AWK is specially useful for automatic generation of particular purpose input files of general purpose programs.

III.2.5. XMGR5

XMGR [40] is a program for the treatment and graphical display of scientifical data. The XMGR5 program [22] is an extension of the previous one, which is capable of reading the information stored in the RELAP5 restart-plot files. In addition, XMGR5 can access the RELAP5 water property tables. For these reasons, XMGR5 has been chosen to display the results of the RELAP5 in-depth postprocessing.

III.3. Linkage of the codes

In this section we describe how the previous codes are linked to build the postprocessing tool.

III.3.1. The BABIECA-RELAP5 coupling

BABIECA is the central program of the postprocessing tool. It will be connected to RELAP5 via the PVM library functions, allowing the exchange of information between them. Output information from RELAP5 will be transferred to properly built BABIECA blocks in order to perform additional verifications of the RELAP5 calculations.

The coupling between BABIECA and RELAP5 was established in a general way that could be used to couple any other code. This may allow to postprocess other codes in the future, aside from RELAP5, or just to run combined simulations where each code is responsible for the calculation of certain phenomena. The first step towards generalization of code coupling was the definition of a standard specifying how a scientific code must be written to facilitate the connection to a general purpose modular simulation language such as BABIECA. A full description of this standard will be the object of another report. A description of the code connection philosophy can be found in [16]. However, some general ideas will be given here.

The proposed standard assumes that the codes will be coupled to a modular simulation language with capabilities similar to those of BABIECA. The connected codes will behave then as blocks of the modular simulation language. The user can now connect the inputs and outputs of the codes by defining a connection topology in the modular language.

Instead of writing a different module for each code coupled to the system, the standard establishes that the modular language will link to any other code by means of two modules called respectively `sndcode` and `rcvcode`. When using the module `sndcode`, the user must specify in the BABIECA input file the code to be used, with the proper command line including options, and the names of the variables that will be modified externally by BABIECA. These variables can be boundary or initial conditions. The `sndcode` spawns the remote code, using the command line defined in the input file, and sends it the names of the boundary and initial conditions that will be supplied by BABIECA. At each time step, BABIECA also sends the remote code the value of the step, the values of the boundary conditions for the current time step, and the values of the initial conditions if some flag is activated. `rcvcode` is the counterpart of `sndcode`. The user must specify in the BABIECA input file the names of the variables whose values will be received from the remote file at the end of each time step. `rcvcode` receives from the corresponding sndcode the process identification number of the remote code, provided by PVM at the spawning time.

The standard also determines how the codes to be coupled must be written in order to facilitate a proper inclusion of PVM functions, that will allow a synchronized information exchange with the modules `sndcode` and `rcvcode`. The PVM library functions will be executed only if the code detects that it has a parent process, and hence it has been spawned by BABIECA. This allows to use the same executable file of the code for coupling with BABIECA and for standalone calculations. After being spawned, and after reading the corresponding input file, the remote code will receive from `sndcode` the names of the boundary and initial conditions to be transferred by BABIECA. These names must be in agreement with a established nomenclature, which is code dependent. With these names the coupled code must be capable of finding a pointer to the values of the corresponding boundary and initial conditions. The code also receives from `rcvcode` the names of the output variables to be sent to BABIECA, in order to obtain pointers to the memory allocation of those variables. At each time step, the code receives from `sndcode` the time step size and the values of the boundary conditions. Initial conditions are also received if necessary. The received values are used to overwrite those computed from the code input file and stored in the memory positions pointed by the pointers mentioned above. Once the solution is advanced one time step, the code sends to `rcvcode` the values of the demanded outputs.

Although the RELAP5 code does not comply with the standard, PVM routines have been properly inserted into the source code to make it behave as if it met the requirements. Not all the capabilities specified in the standard have been implemented in the RELAP5-BABIECA coupling. For instance, only boundary conditions specified by means of time dependent volume or junctions can be overwritten. No initial conditions can be supplied, since this feature is not necessary for postprocessing purposes.

III.3.2. The role of the AWK editor

The non-interactive text editor AWK is used to build the BABIECA input file necessary to check the results of a given RELAP5 case and once the user has specified the control volumes to which the checks must be applied. The checks are hence written in a file called `babpost.awk`, which contains AWK commands. With this file, and depending on the particular RELAP5 calculation and the volumes to be checked, the AWK editor builds the corresponding BABIECA input file.

III.3.3. The role of the XMGR5 program

XMGR5 is just used to graphically display the RELAP5 and the postprocessing results after the end of the calculations. The tool takes advantage of the XMGR5 capabilities to read RELAP5 restart-plot files.

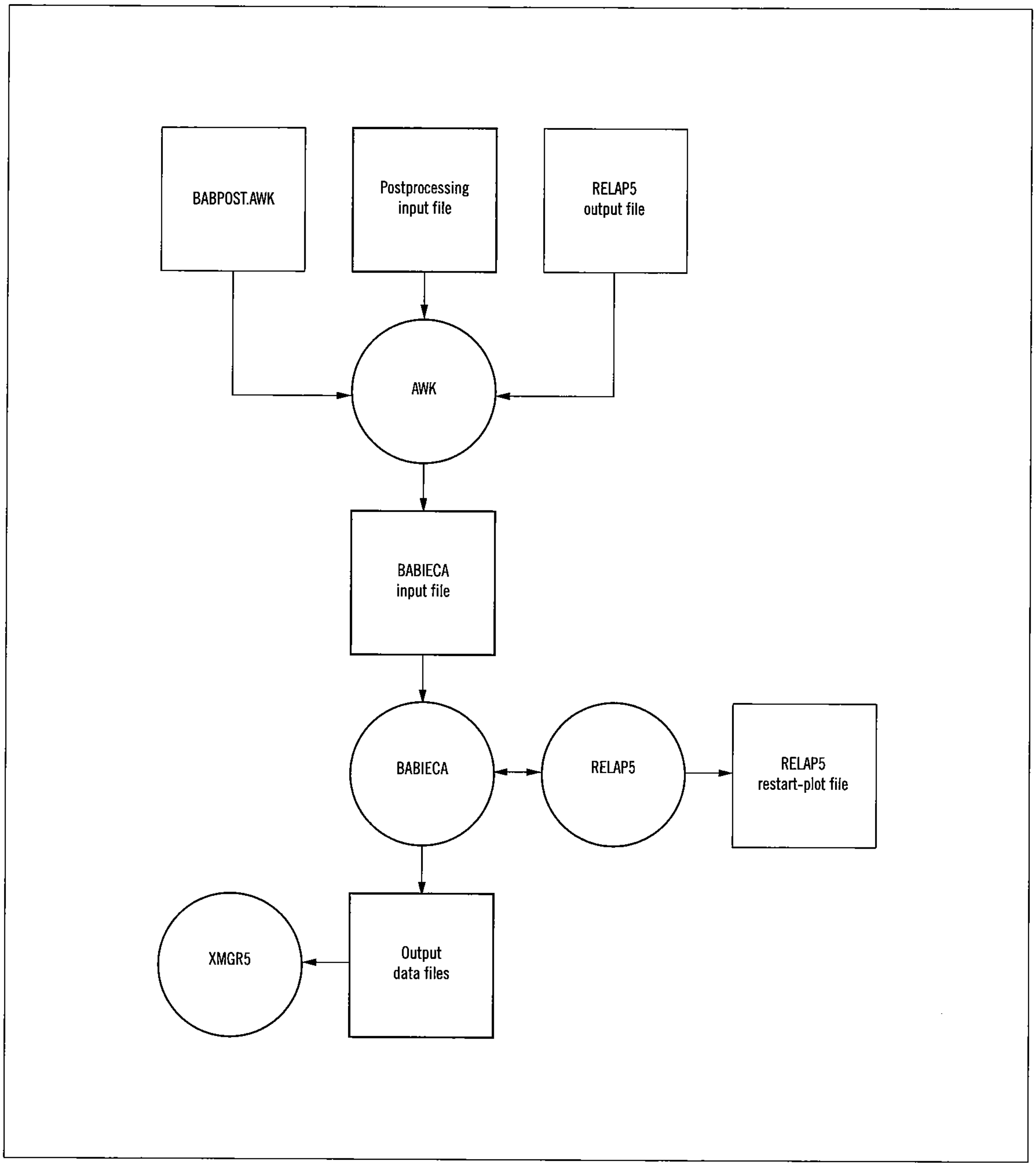

Figure 3.1. Functioning diagram of the analysis tool

III.4. Functioning of the analysis tool

The analysis tool described in this report works as depicted in figure 3.1.

First of all, the user must write an ASCII file containing the following information in successive lines:

Each control volume to be postprocessed must be also specified with:

```
FILE = rootword
HOST = host
INITTIME = inittime
STARTTIME = starttime
ENDTIME = endtime
FREQ = freq
NSYSTEMS = nsystems
For each thermalhydraulic subsystem the file must contain also:
SYSNAME = sysname
NVOLS = nvols
cccvv0000
```

The words in typewriter type must be written literally. The words in italics have the following meaning:

- *rootword* is the root name of the RELAP5 case to be checked. It is assumed that the input file has the extension `.i` appended to the root name. The RELAP5 output file will be called *rootword*`.o` and the restart-plot file will be *rootword*`.r`.
- *host* is the name of the host where the RELAP5 executable file is allocated.
- *inittime* is a real number specifying the transient instant, in seconds, at which RELAP5 will be spawned by BABIECA. It is used to synchronize the times of BABIECA and RELAP5, specially when the RELAP5 calculation is a restart case.
- *starttime* is a real number specifying the transient instant, in seconds, at which the checks begin.
- *endtime* is a real number specifying the transient instant, in seconds, at which the checks end.
- *freq* is an integer number specifying the frequency used to write output data, in time steps.
- *nsystems* is the number of thermalhydraulic subsystems to be checked. A TH subsystem is a set of RELAP5 control volumes defined by the user. They may be connected or not.
- *sysname* is a name given to each TH subsystem. The results of the verifications in each subsystem will be written in a file called *rootword_sysname*`.dat`.
- *nvols* is the number of control volumes in the current TH subsystem.
- *cccvv*`0000` is a nine digit string denoting each control volume, according to the RELAP5 nomenclature. The three digit substring *ccc* refers to the RELAP5 hydrodynamic component and the two digit substring *vv* refers to the number of the volume within the component (`01` for a single volume). The last four digits are always `0000`.

In the following lines an example of a file of this type is shown:

```
FILE = liq_pres
HOST = spp2k
INITTIME = 0.0
STARTTIME = 200.0
ENDTIME = 400.0
FREQ = 5
NSYSTEMS = 1
SYSNAME = mainpipe
NVOLS = 10
404010000
404020000
404030000
404040000
404050000
404060000
404070000
404080000
404090000
404100000
```

After writing a file like this, the user must execute from the UNIX command prompt the following sentence:

`awk -f babpost.awk` *inpfile rootword*`.o > `*babfile*`.bab`

The previous line performs the following actions: the AWK editor reads the file of AWK commands contained in the file `babpost.awk` and looks for the information in the file *inpfile.* Once this information has been read, and AWK knows about the control volumes to be checked, the AWK editor looks into the RELAP5 output file *rootword*`.o` to identify the junctions connecting the control volumes to be checked. This file must be available by running the RELAP5 input file *rootword*`.i`, preferably in check mode. Finally, AWK writes the BABIECA input file *babfile*`.bab`, which performs the ckekings of the RELAP5 calculation.

The following step consists in running PVM with a configuration file specifying the hosts that will constitute the virtual machines, executable paths, working directories, etc. Here is an example of such a file:

```
# Configuration file for pvm3
#
spp2k dx=/opt/pvm3/lib/pvmd
ep=$HOME/bin:$HOME/relap/pvmcomp/selap
bx=/users/rhs/casos/testpvm/debugger
wd=$HOME/relap/postcases/liq_pres
```

The PVM daemon is activated just by typing

`pvm` *pvm_conf_file*

from the command prompt, where *pvm_conf_file* is the configuration file.

The postprocess of the RELAP5 case is then executed by typing

`BABIECA -n` *babfile*`.bab`

BABIECA spawns then a RELAP5 process and sets the communication between them. After finishing a time step calculation, RELAP5 sends the required variables to BABIECA. The latter computes then the proper verifications. Once the calculation is finished, PVM must be deactivated. The results of each TH subsystem are stored in files named *rootword_sysname*`.dat`. The results concerning individual volumes are stored in files *rootword_cccvv*`.dat`. The variables in these files can be displayed with XMGR5.

IV. Application examples

IV. Application examples

IV.1. Introduction

This chapter illustrates the capabilities of the analysis tool by means of some application examples.

Example 4.2 reveals an error in the input file due to the user. Section IV.3 is a good example of interpretation of the results computed by the code, which demonstrates a fake argument commonly used to explain the different depressurization rates obtained when a breach appears in the vapor or liquid region of a hydraulic system. The same example is used to verify the non-equilibrium model derived in section II.5. Example 4.4 points out the boundary condition overspecification problem in several TH codes. Finally, application example 4.5 is devoted to other problem usually found in TH codes: flow mixing and splitting in converging pipes.

IV.2. Compression caused by heating in a closed pipe

IV.2.1. Example description

This example consists in an 18 node horizontal pipe. Allthough all the nodes are equal, it is convenient to discriminate between the ten central nodes and the two groups of 4 nodes each one located at the tips of the pipe. The noding is shown in figure 4.1. The pipe is closed in both sides, and initially filled with liquid and vapor in equilibrium conditions, in a horizontally stratified regime. The pressure is similar to that of the primary circuit in a PWR. After running a null transient, to let the system reach a steady state, a heat power is introduced in each of the ten central nodes. The total power is represented in figure 4.2.

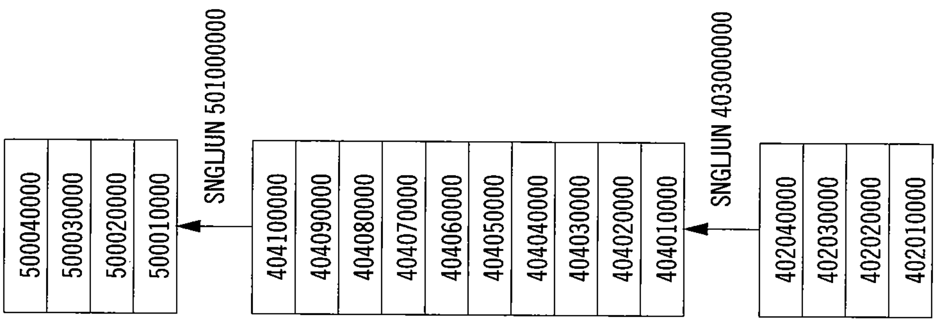

Figure 4.1. Nodalization used in the example

As a result of the heating, and being the pipe closed in both sides, we expect the pressure to rise, along with the void fraction. Since the problem shows a clear spatial simmetry, simmetric results are so expected.

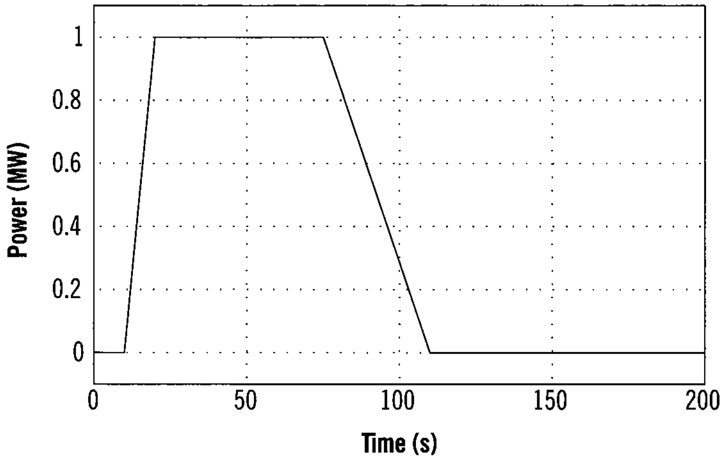

Figure 4.2. Power introduced in the pipe

IV.2.2. Analysis of results

Figure 4.3 shows the two sides of the pressure equation (2.29), in the TH subsystem defined by the ten central nodes that constitute the pipe 402. Both sides of the equation are identical. No anomaly in the solution can be inferred from this results. The figure also shows the expected increase of the system pressure.

The comparison of the two sides of the level equation is fully satisfactory either, according to figure 4.4. Nevertheless, an increase of the liquid level in the system is observed, contrary to what is expected. This observation suggest an error in the simulation of the example.

Figure 4.5 shows the expected increase of the vapor mass, even though great discrepancies between the two sides of the equation can be noticed. Mass errors in the vapor phase are very frequently observed in RELAP5 calculations. The mass error is neglibible in the liquid phase, represented in figure 4.6. This figure also shows, contrary to what is expected, an increase of the liquid mass.

The two sides of the non-equilibrium mass equation, for the vapor phase, shown in figure 4.7, are not coincidental because of the mistmach of the vapor mass. On the other hand, the verification of the liquid non-equilibrium mass equation is satisfactory, as shown in figure 4.8.

The terms taking part in the pressure and level equations are displayed in figure 4.9. The only non-zero terms are the volumetric ones, which have surprisingly the same shape in both phases.

Figures 4.10 to 4.16 illustrate the same analysis in pipe 500. The postprocessing of the pressure equation leads to results identical to those obtained in the pipe 402. However, the liquid volume decreases during the transient, unlike in pipe 402. The simmetry of the problems does not seem to allow a result like this, reenforcing the suspects of an error in the solution. The two sides of the vapor mass equation are quite diferent either.

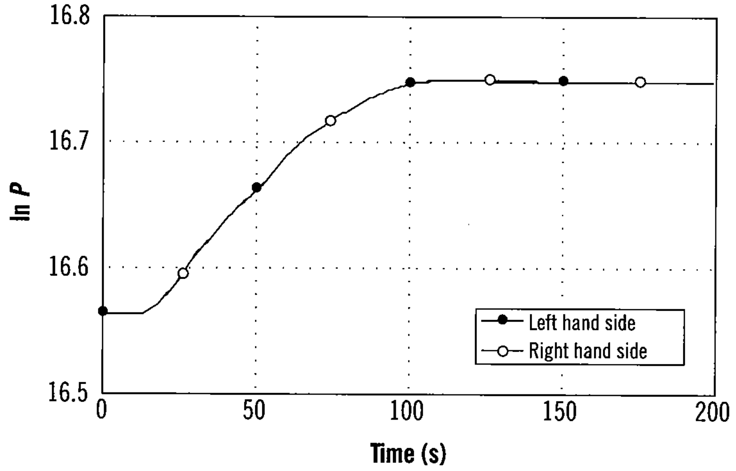

Figure 4.3. Verification of the pressure equation. Pipe 402

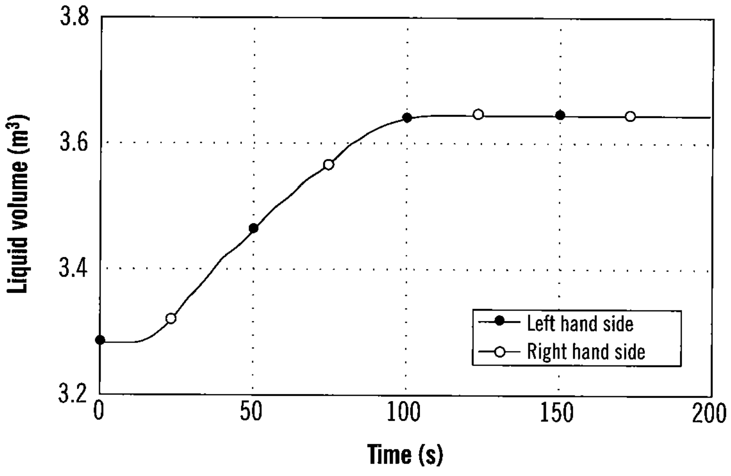

Figure 4.4. Verification of the level equation. Pipe 402

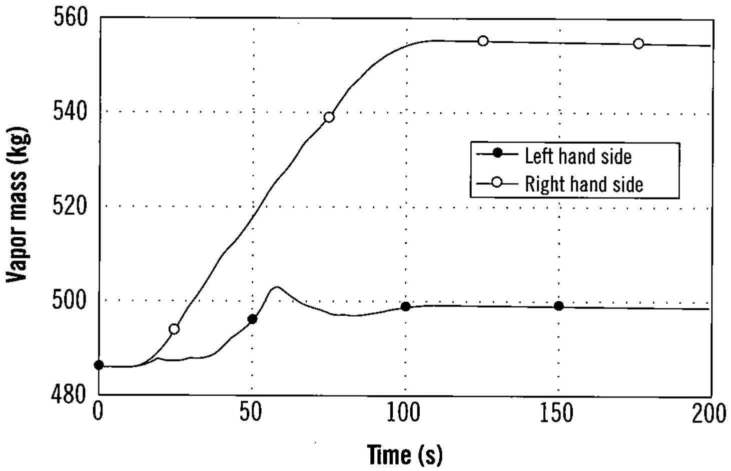

Figure 4.5. Verification of the vapor mass. Pipe 402

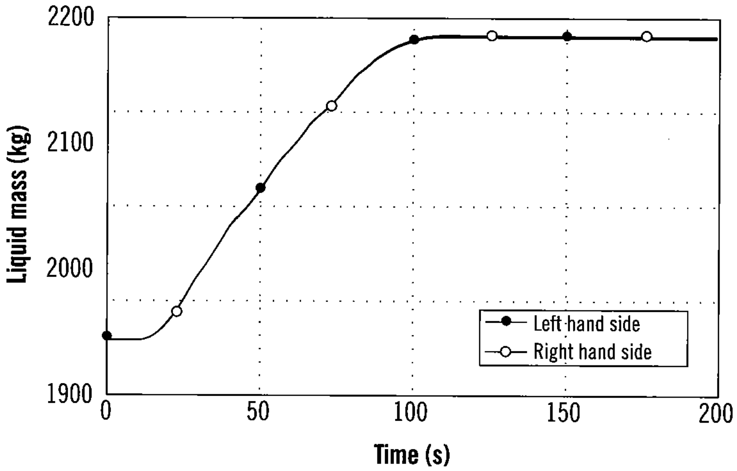

Figure 4.6. Verification of the liquid mass. Pipe 402

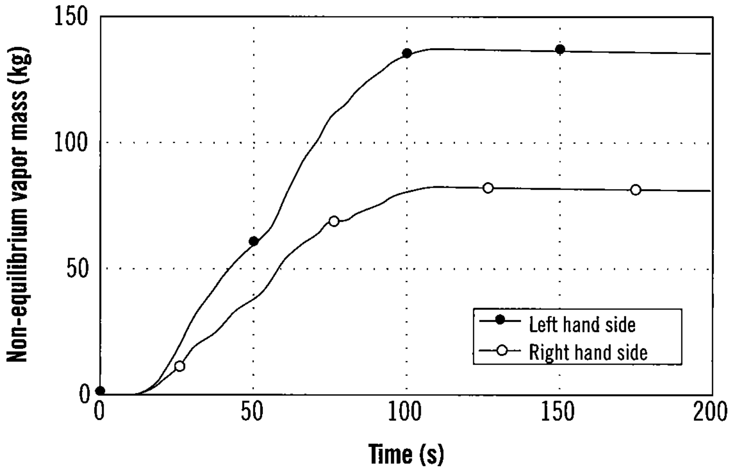

Figure 4.7. Non-equuilibrium vapor mass. Pipe 402

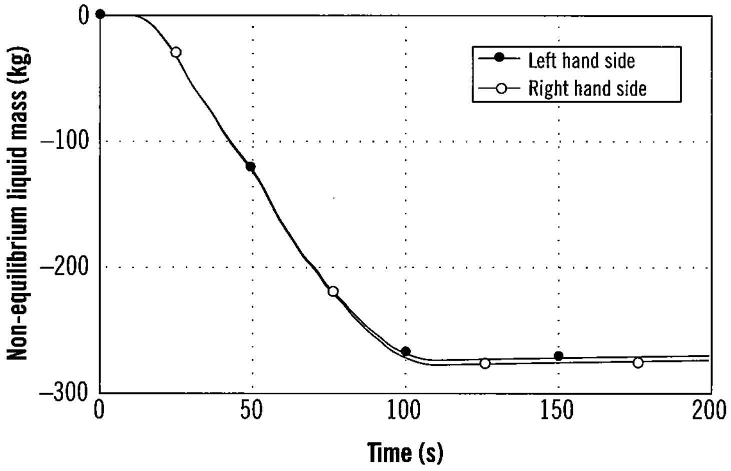

Figure 4.8. Non-equuilibrium liquid mass. Pipe 402

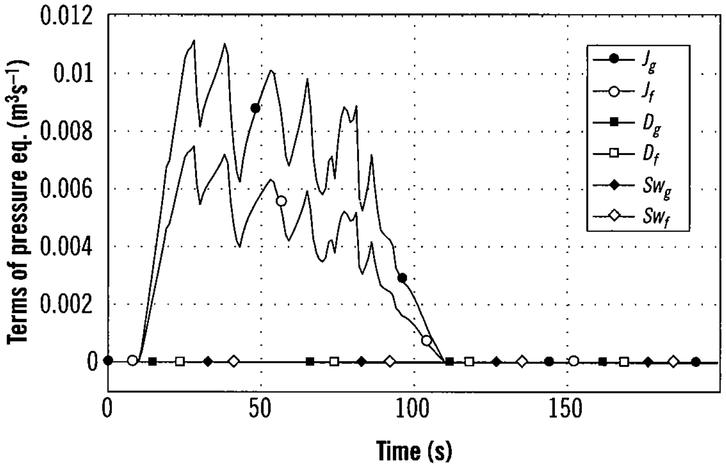

Figure 4.9. Terms of the pressure equation. Pipe 402

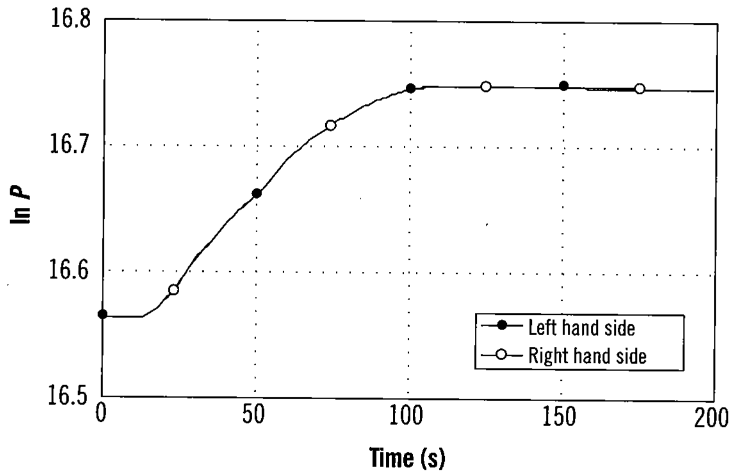

Figure 4.10. Verification of the pressure equation. Pipe 500

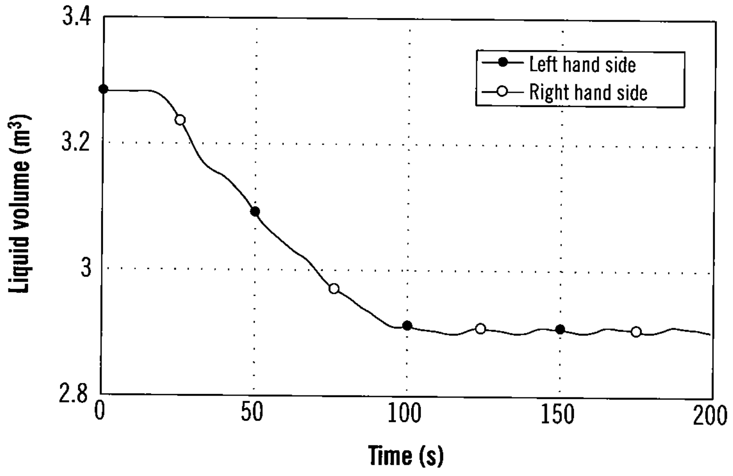

Figure 4.11. Verification of the level equation. Pipe 500

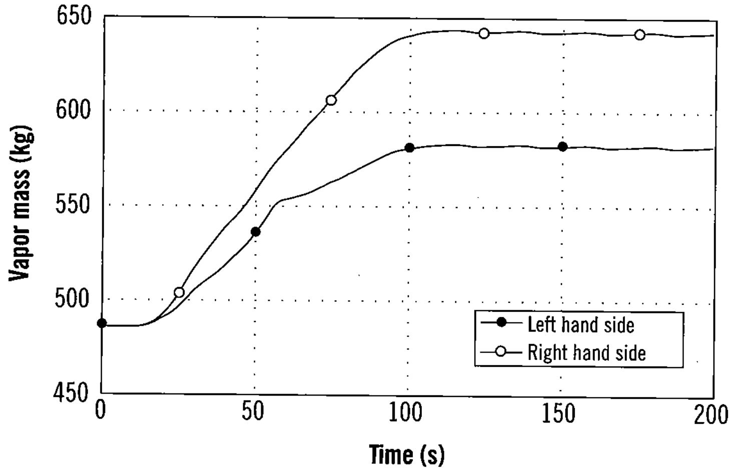

Figure 4.12. Verification of the vapor mass. Pipe 500

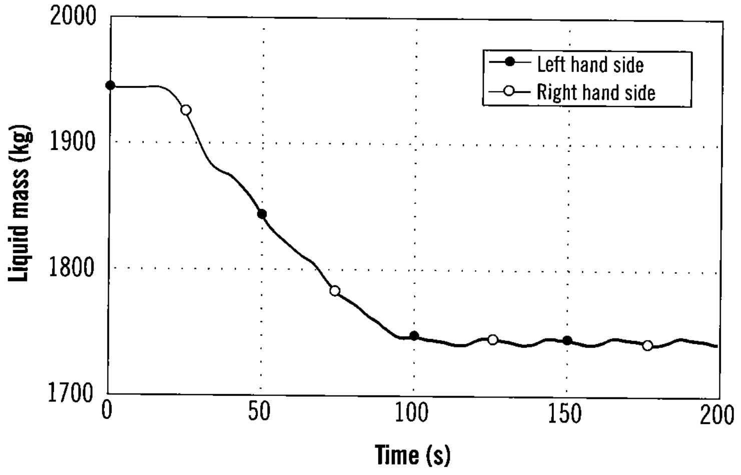

Figure 4.12. Verification of the liquid mass. Pipe 500

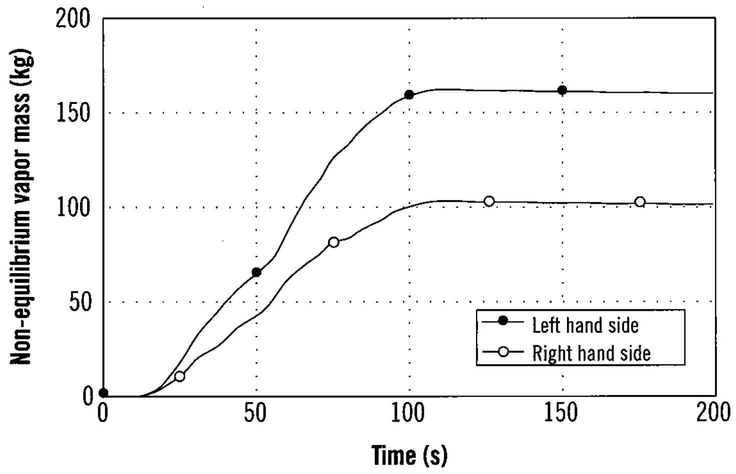

Figure 4.14. Non-equuilibrium vapor mass. Pipe 500

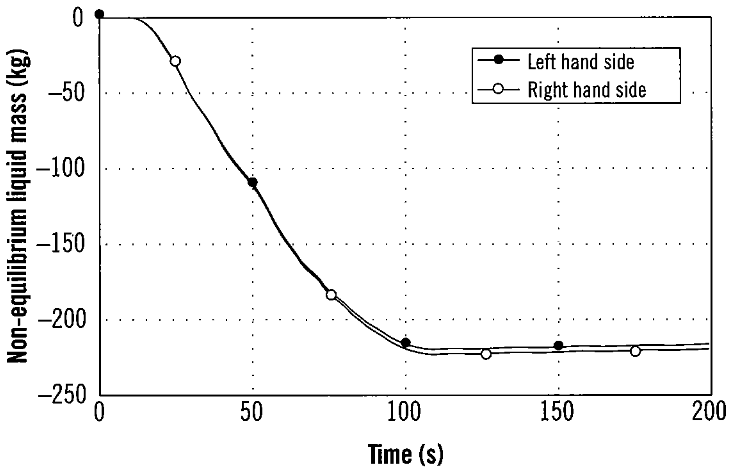

Figure 4.15. Non-equuilibrium liquid mass. Pipe 500

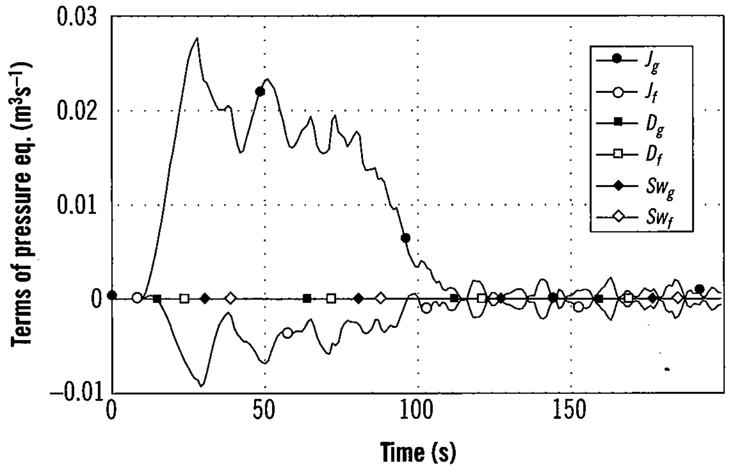

Figure 4.16. Terms of the pressure equation. Pipe 500

It can also be noticed that the mass vapor is different in pipes 402 and 500. The liquid mass decreases as well, unlike what happens in pipe 402. Discrepancies in the non-equilibrium mass equation, caused by the errors in the vapor mass, are still observed.

The only non-negligible terms in the pressure and volume equations are still the volumetric ones (see figure 4.16). However, the shapes of the vapor and liquid volumetric terms are quite different, contrary to pipe 402. This additional element of assimmetry is another indicator of an error in the definiton of the problem, very probably induced through the input file.

Looking thoroughly into the input file, it was found out that the homegeneous equilibrium model had been in specified in the junction 403, very probably as a result of a *cut and paste* operation. This specification introduces an element of assimetry in the problem, which is enough to justify the different results obtained in the pipes 402 and 500. The inadverted use of the homegeneous equilibrium model in the junction 403 explains why the volumetric flows in figure 4.9 have the same shape. In a homogeneous model the velocity of each phase is identical. Thus the difference in the volumetric flows is caused by the scaling of the void fraction. Once the error in junction 403 has been fixed, the example and the postprocessing tool have been rerun.

The figure 4.17 shows that, once more, the analysis of the pressure equation is satisfactory. Both sides of the level equation (see figure 4.18) are also in good agreement. It must be realize that the level decreases as expected. Significant mass errors (figure 4.19) are still observed in the vapor phase, suggesting that the solution is not completely right even when the homogeneous model flag has been corrected. on the other hand, the liquid mass shown in figure 4.20 is properly preserved. As a consequence of the mass errors, the non-equilibrium vapor mass analysis is not satisfactory either. The non-equilibrium liquid mass is preserved, because the liquid mass is. The terms of the pressure and level equations are plotted in figure 4.23. The only non-negligible terms are the volumetric ones. It is very curious that in the permanent regime at the end of the transient the vapor volumetric flow is compensated by the vapor flow, resulting in a steady pressure. On the contrary, the level never reaches a steady state, remaining in an oscillating permanent regime, since only the liquid volumetric flow plays a role in the corresponding equation.

The results in the pipe 500 are identical to those obtained in pipe 402, as shown in figures 4.24 to 4.30. This confirms that, as expected, the results are symmetric with respect to the spatial variable *x*.

IV.3. Depressurization in a circuit with liquid and vapor

IV.3.1. Justification of the example

The example shown in this section illustrates the use of the analyisis tool to interpret the results computed by the RELAP5 code. It will be also used to check the hypotheses that lead to the simplified non-equilibrium mass equations derived in section II.5. The example consists in a closed hydraulic circuit, nodalized as represented in figure 4.31, initially filled with liquid until

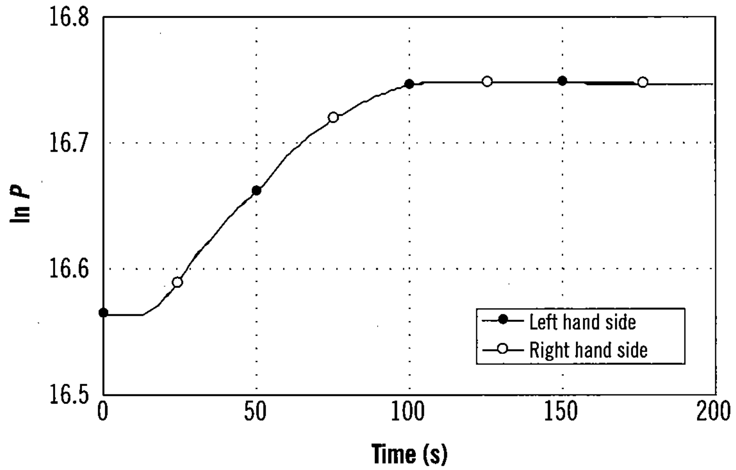

Figure 4.17. Verification of the pressure equation. Pipe 402. Revised example

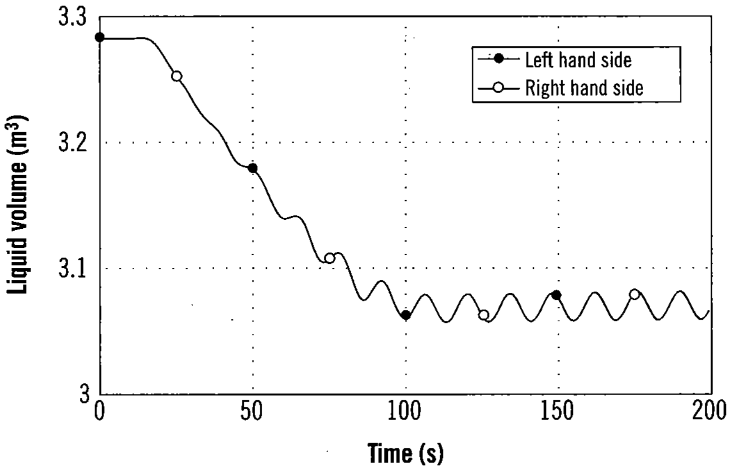

Figure 4.18. Verification of the level equation. Pipe 402. Revised example

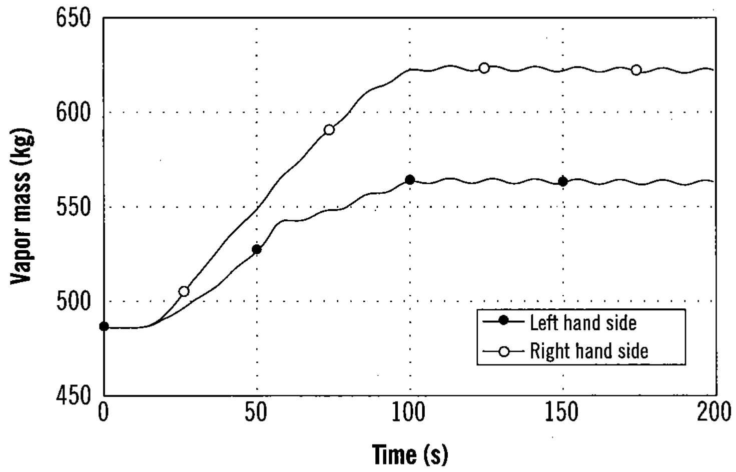

Figure 4.19. Verification of the vapor mass. Pipe 402. Revised example

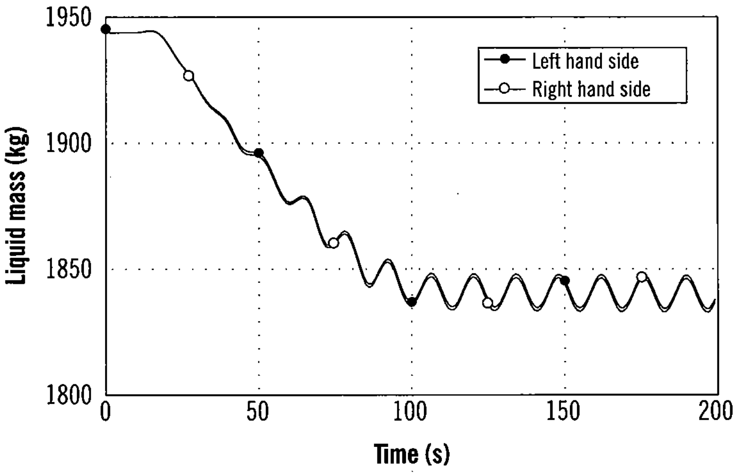

Figure 4.20. Verification of the liquid mass. Pipe 402. Revised example

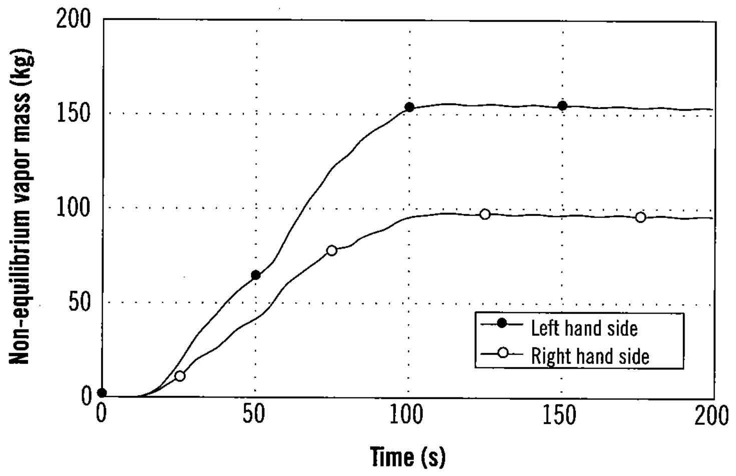

Figure 4.21. Non-equilibrium vapor mass. Pipe 402. Revised example

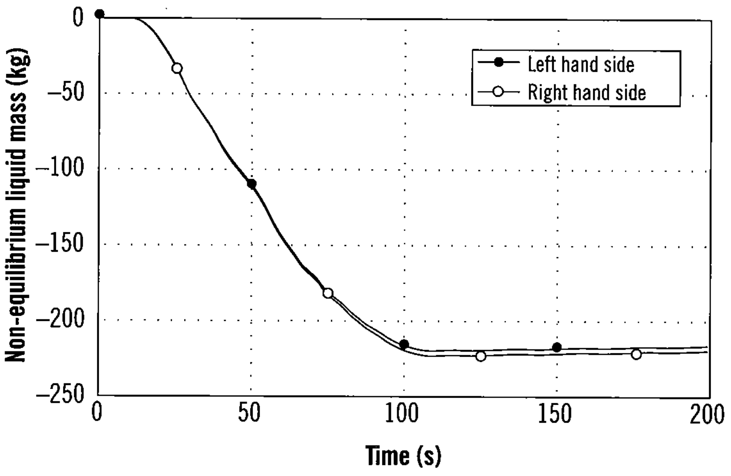

Figure 4.22. Non-equilibrium liquid mass. Pipe 402. Revised example

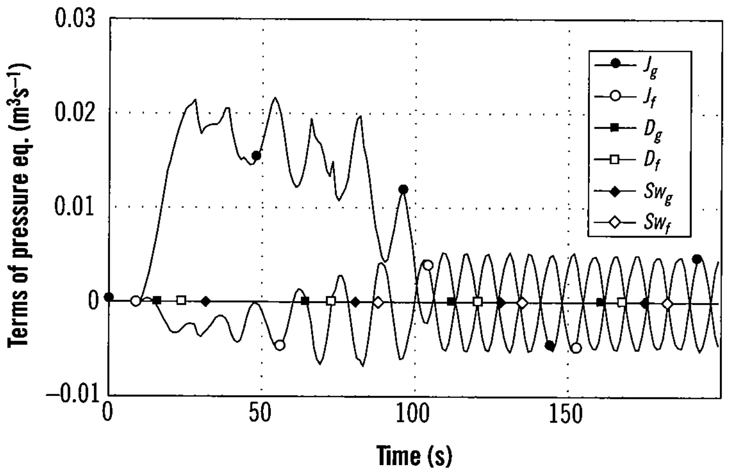

Figure 4.23. Terms of the pressure eq. Pipe 402. Revised example

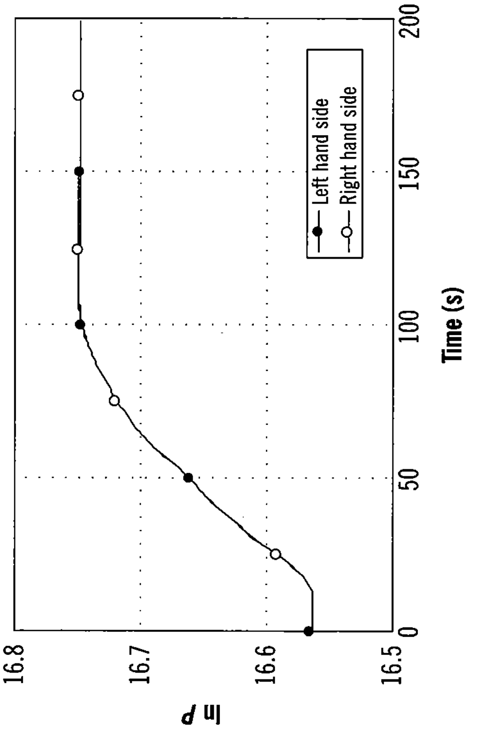

Figure 4.24. Verification of the pressure equation. Pipe 500. Revised example

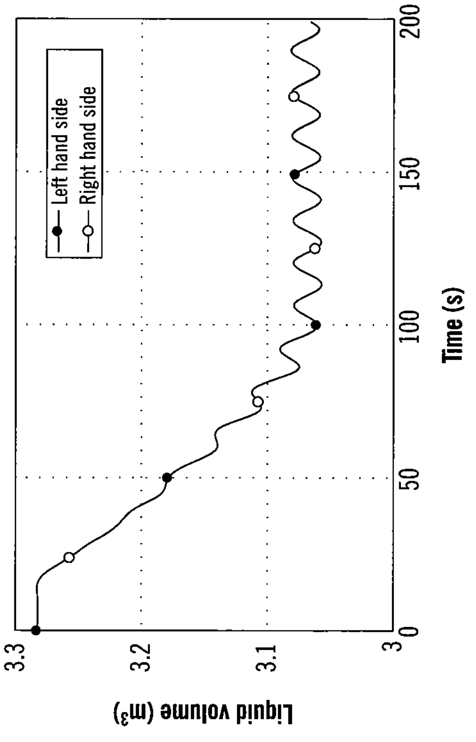

Figure 4.25. Verification of the level equation. Pipe 500. Revised example

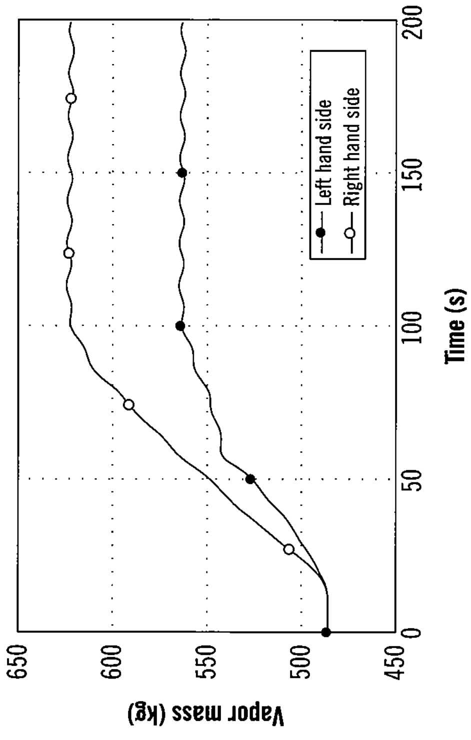

Figure 4.26. Verification of the vapor mass. Pipe 500. Revised example

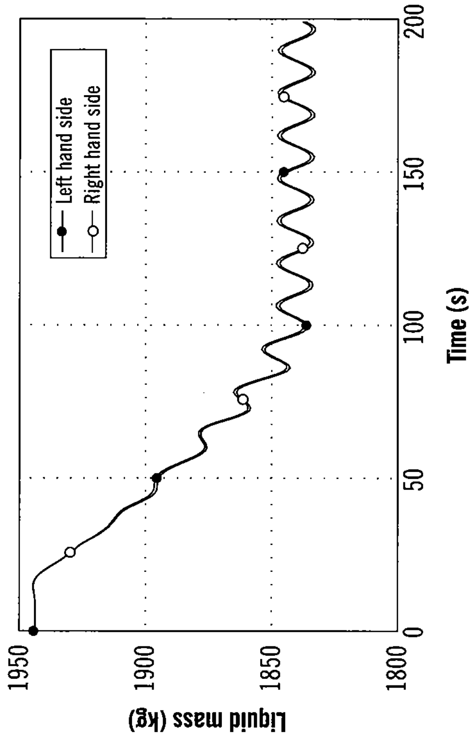

Figure 4.27. Verification of the liquid mass. Pipe 500. Revised example

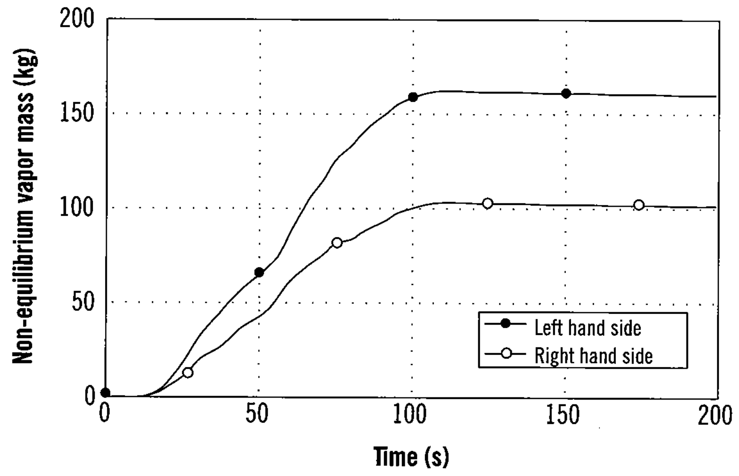

Figure 4.28. Non-equilibrium vapor mass. Pipe 500. Revised example

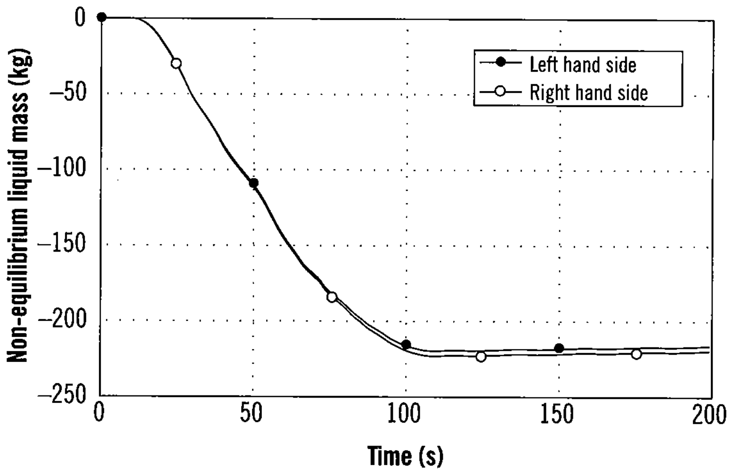

Figure 4.29. Non-equilibrium liquid mass. Pipe 500. Revised example

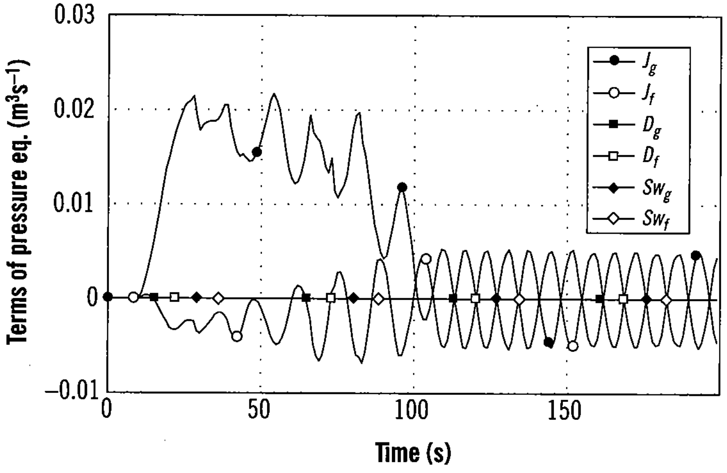

Figure 4.30. Terms of the pressure eq. Pipe 500. Revised example

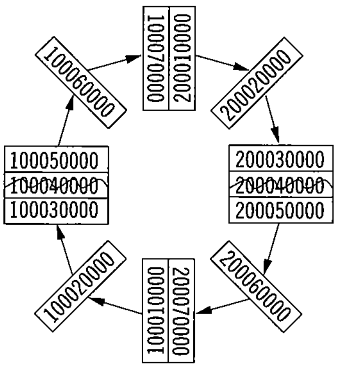

Figure 4.31. Circuit nodalization

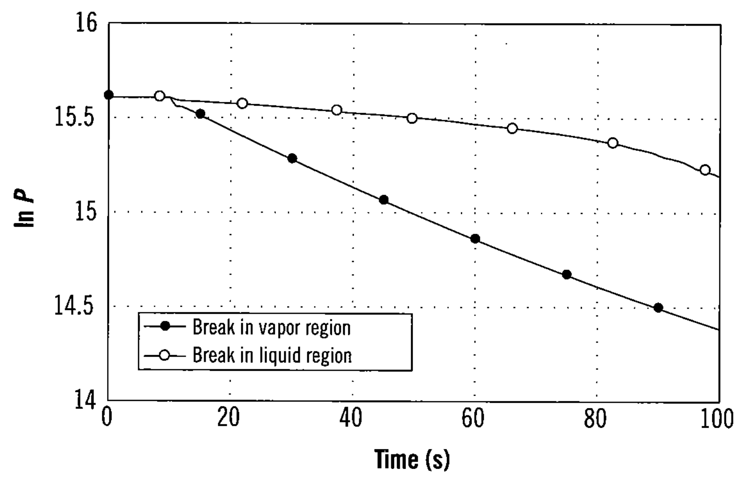

Figure 4.32. Pressure evolution

approximately half the level of the circuit. The other half is filled with vapor. The thermodynamic conditions are similar to those of the secondary circuit of a PWR. Being in stagnant conditions, the liquid occupies the lower half of the circuit and the vapor occupies the upper half. The exercise is focused in the comparison of the depressurization rates obtained when a breach of the circuit takes place either at the uppper part or at the lower part. It is well known that a break in the steam line of a secondary circuit leads to a greater depressurization rate that a break in the feedwater line. The explanation given by many authors ([3] among them), is that the energy released through the break is greater if steam leaks. This argument assumes a direct relationship between the specific enthalpy of the fluid leaking through the break and the system pressure which is not demonstrated.

Figure 4.32 shows how, as predicted, the depressurization rate is grater if the break is induced in the upper part of the circuit, filled with vapor.

IV.3.2. Verification of the balance equations

Before proceeding to interpret and explain the results, the fulfillment of the mass, pressure and level equations will be checked, according to the equations (2.1), (2.29) and (2.31).

Figure 4.33 shows the two sides of the pressure equation, when the break takes place in the vapor region. The agreement between them is fair. No objections can be made to the results from this viewpoint at least.

The two sides of the level equation (2.31), shown in figure 4.34, are also in good agreement.

On the contrary, the postprocess of the vapor and liquid mass equations, shown respectively in figures 4.35 and 4.36, leads to dissapointing results, questioning the quality of the solution. Note that the errors in the vapor and the liquid phase are of similar order, but with opposite sign. This is a good example of compensation of the mass errors between the phases, which is not detected by the RELAP5 mass verification.

If the break takes place in the liquid region, the analysis of the pressure, level and vapor and liquid mass, represented respectively in figures 4.37, 4.38, 4.39 and 4.40, leads to similar results. Error compensation between the liquid and vapor phases is observed as well, although it is smaller than when the break occurs in the vapor region.

IV.3.3. Interpretation of results

Although the specific internal energy and enthalpy are greater for the vapor phase than for the liquid phase, the energy and enthalpy flows released through the break are greater when the break takes place in the liquid region, because of the higher density of this phase. This is confirmed in figures 4.41 and 4.42. As a conclusion, the higher depressurization rate when the break occurs in the vapor region cannot be explained in terms of the released energy. Another explanation must be found.

The results computed by RELAP5 can be interpreted according to the pressure equation (2.29). Figure 4.43 shows the terms in the numerator of the pressure equation, except the negligilbe Ac term, when the break is induced in the vapor region. The same terms, for the case of break in the liquid region, are shown in figure 4.44.

It is easy to realize that the depressurization is mainly driven by two term, in the first part of the transient at least. Those are the volumetric term J (for the vapor phase in the first case and for the liquid phase in the second one) and the swelling term Sw_g(caused by the appearance of vapor). Both terms tend towards mutual compensation, although the volumetric term is always the dominant one. The sum of the two terms is greater when the break occurs in the vapor region, leading in this case to a greater numerator in the pressure equation. Moreover, the system compressibility, which acts as the denominator of the pressure equation, is greater when the break takes place in the liquid, as shown in figure 4.45. The global effect is that the depressurization rate is greater for a break in the vapor region.

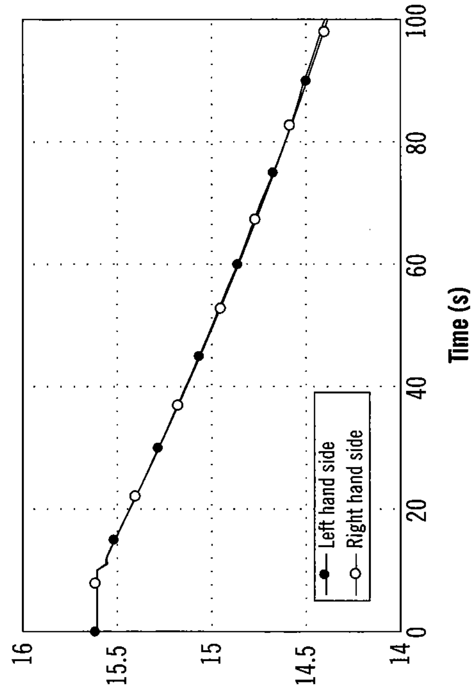

Figure 4.33. Pressure analysis. Break in vapor region

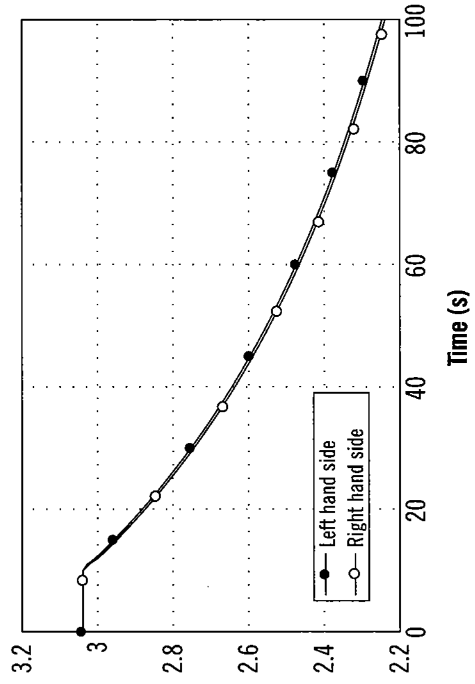

Figure 4.34. Liquid volume analysis. Break in vapor region

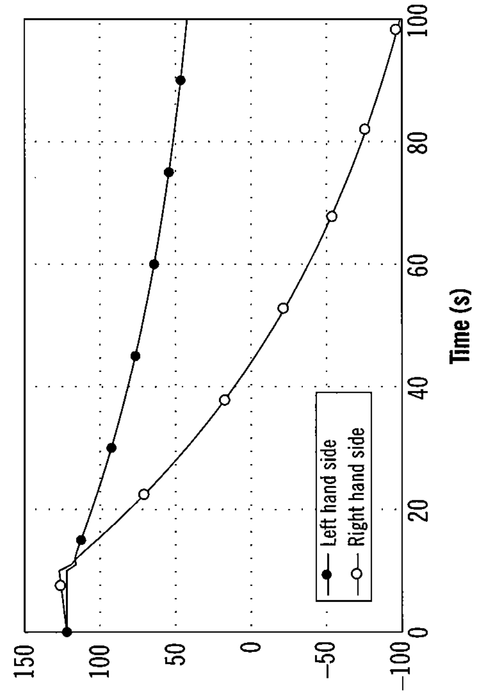

Figure 4.35. Vapor mass analysis. Break in vapor region

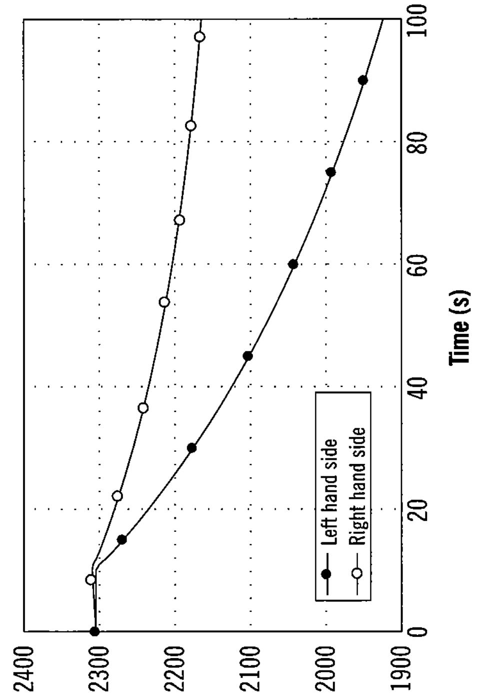

Figure 4.36. Liquid mass analysis. Break in vapor region

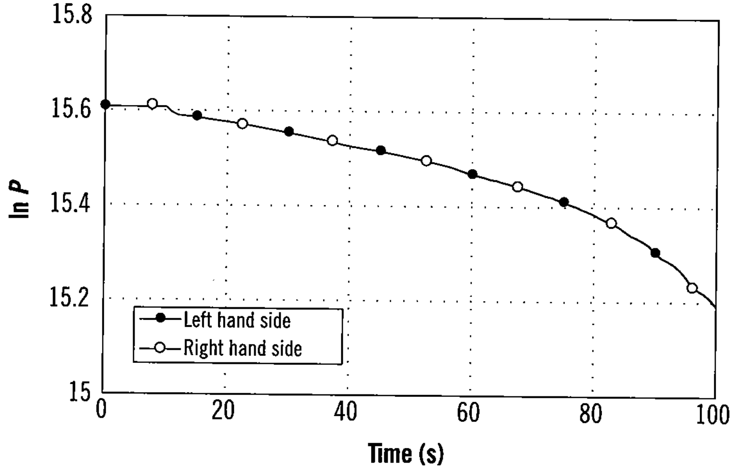

Figure 4.37. Pressure analysis. Break in liquid region

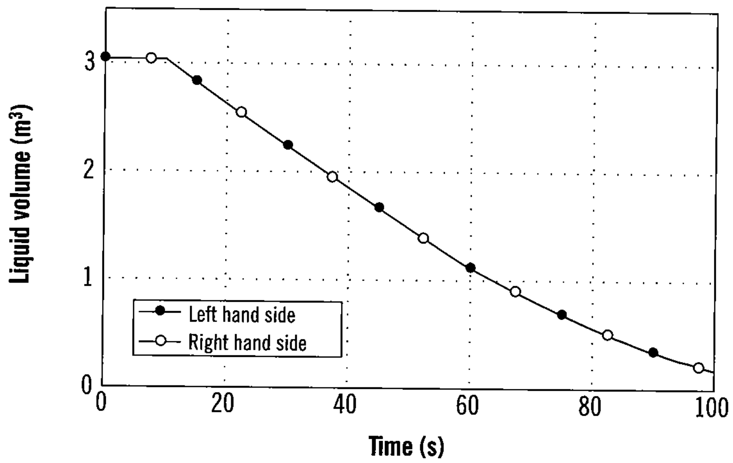

Figure 4.38. Level analysis. Break in liquid region

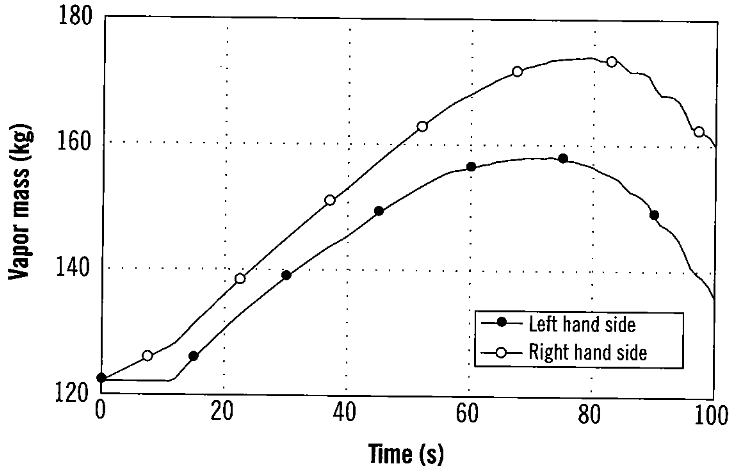

Figure 4.39. Vapor mass analysis. Break in liquid region

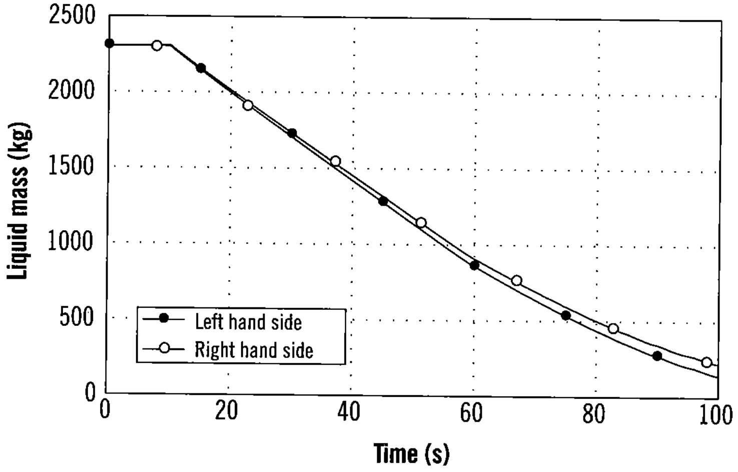

Figure 4.40. Liquid mass analysis. Break in liquid region

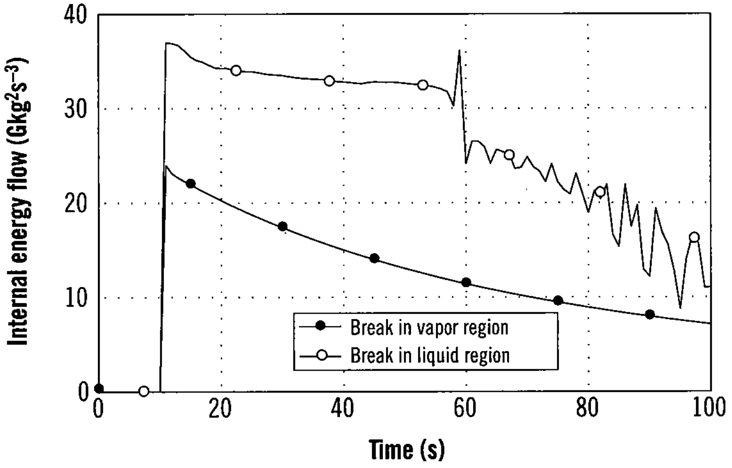

Figure 4.41. Energy flow released through the breaks

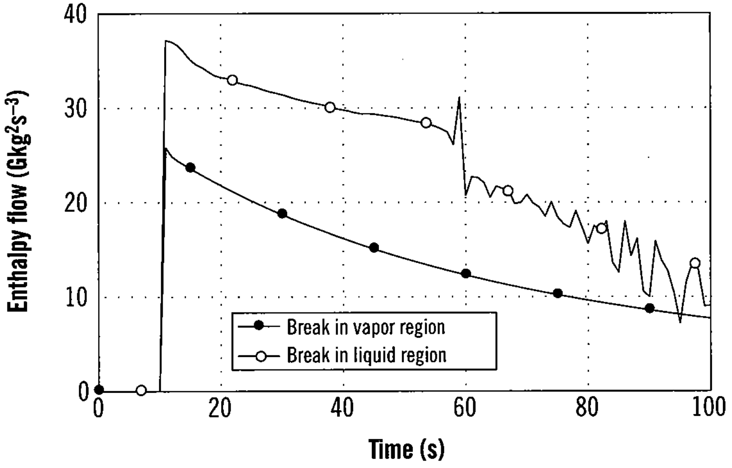

Figure 4.42. Enthalpy flow released through the breaks

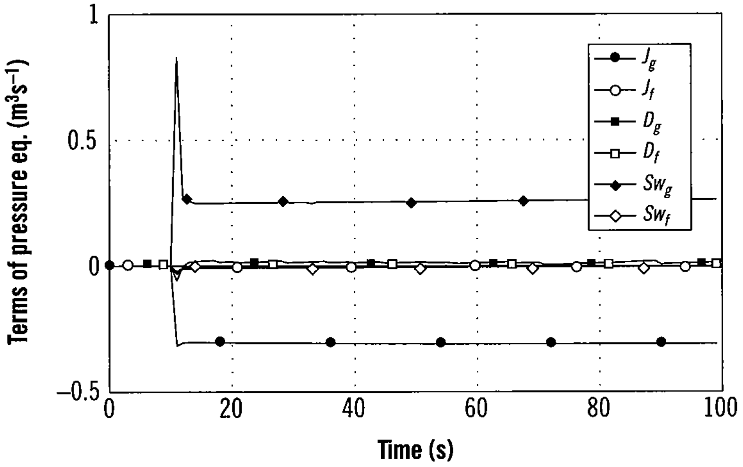

Figure 4.43. Terms of pressure eq. Break in vapor region

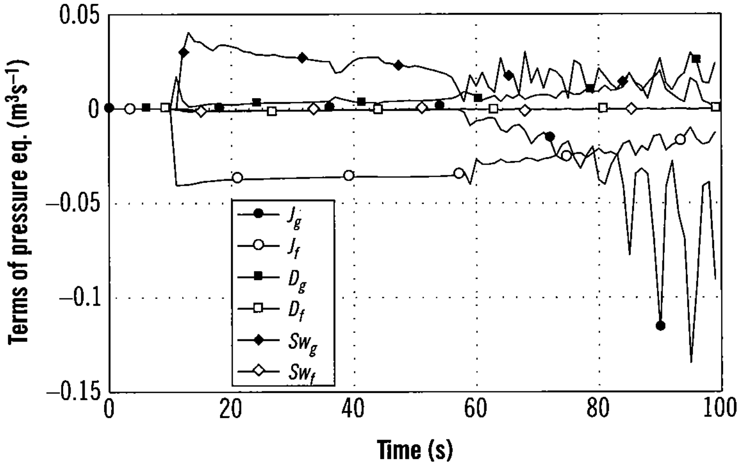

Figure 4.44. Terms of pressure eq. Break in liquid region

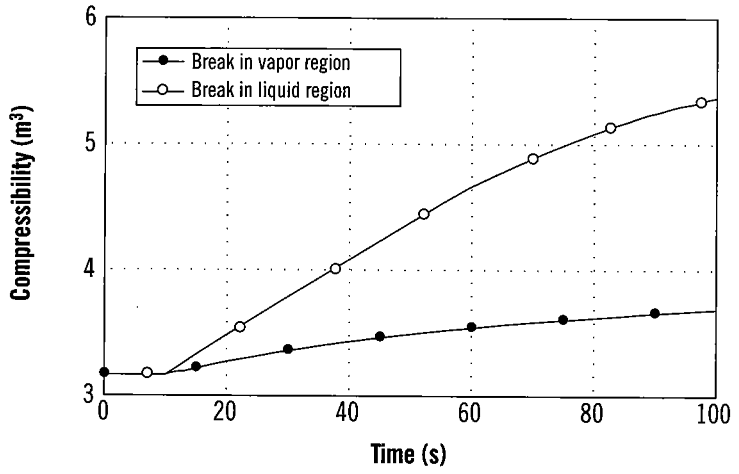

Figure 4.45. System compressibility

IV.3.4. Verification of the simplified non-equilibrium model

This section is devoted to the verification of the hypotheses under which the simplified non-equilibrium mass equations described in section 5 are valid and the two phases remain decoupled from each other. Fulfillment of equations (2.68) and (2.73) will then be checked in the current example. For the sake of brevity, only the control volumes 10001 (in the upper part of the circuit) and 10001 (in the lower part) will be checked. We remind that this is an exercise to illustrate how a code can be used to validate, as if it were an experimental facility, simpler models. We do not intend to validate the above mentioned hypotheses at the sight of this exercise. A true validation would require a huge number of transients and calculations.

If the break occurs in the vapor region, the vapor generation rate in volume 10001 is positive during the whole transient. Hence the verifications must be performed in this phase.

Figure 4.46 represents the right hand side of equation (2.49) and the term of that side including the dependency on Γ_{ig}. It is clear that this term contributes very weakly to the right hand side of equation (2.49).

The terms that depends on Γ_w is zero, since no heat is transferred through the wall.

Figure 4.47 displays the bulk interfacial heat versus its approximation according to equation (2.67). The agreement is excellent.

The liquid generation rate is positive in volume 20001 during all the transient, with the exception of the first instants. The right hand side of equation (2.49) is much larger than the term with Γ_{if}, as shown in figure 4.48. The bulk interfacial heat for the liquid phase is almost identical to its approximation (see figure 4.49). During the first instants of the transient the interfacial heat is zero according to the figure. This only means that Γ_{if} is negative during those instants and the simplified model is not applicable for the liquid phase. It must not be understood as a true zero value.

If the break occurs in the liquid region we come out with the similar conclusions, as shown in the figures 4.50 and 4.53.

It can be inferred at the sight of these results that the simplifying hypotheses of the non-equilibrium mass equation could be applied in this example at least.

IV.4. Over specification of boundary conditions

IV.4.1. Motivation and background

This example tries to show how the post-processing techniques can be used to identify potential mistakes committed during the development of a TH computer code, which can lead to unphysical systems even if the problem is well posed from the mathematical point of view. A historical review of the boundary condition over specification problem is pertinent here.

During the 1996 Fall CAMP Meeting [14] the overspecification of boundary conditions in the RELAP5 code when using time dependent junctions was already pointed out. The problem was illustrated with an example based on a simplified pressurizer model. No correcting actions were taken at that time, probably because the root cause of the problem had not been identified yet.

During the 2000 Fall CAMP Meeting, Dr. Jingzao Zhang from Tractebel presented a simple RELAP5 case which led to surprising and unphysical results. It will be demonstrated here that the cause of these results is, like in the pressurizer case presented in [14], the overspecification of boundary conditions. Zhang's case will be finally used to illustrate the problem and explain the root causes, since it is much simpler and leads to more dramatic results than the pressurizer case.

Zhang's case, depicted in figure 4.54, comprises a time dependent volume (TDV) at the inlet side and another at the outlet side, one single volume, one time dependent junction (TDJ) that connects the inlet TDV to the single volume to specify the inlet flow, and a single junction at the outlet side. The pressure in the inlet TDV is set to 4 MPa and the temperature to 553 K. The pressure of the outlet TDV is set 2.1 MPa, and the temperature, which does not affect the result since the volume is placed downstream, to 523 K. The mass flow in the TDJ is kept constant at a value of 10 kg s^{-1}. The initial conditions in the single volume are those of the outlet TDV. The initial velocity in the single junctions is zero. In the conditions described above, the system contains superheated steam. Since the inlet TDJ forces the fluid from the inlet TDV to replace the fluid of the single volume, we expect a increase of its temperature. In the final steady-state, the thermodynamic properties of the single volume should be very similar to the boundary conditions specified in the inlet TDV.

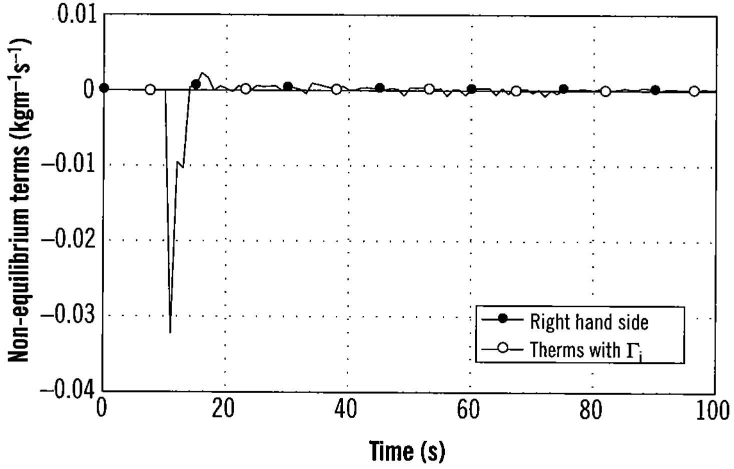

Figure 4.46. Neglection of term with Γ_i in the non-equilibrium eq. Vapor phase. Volume 10001. Break in vapor region

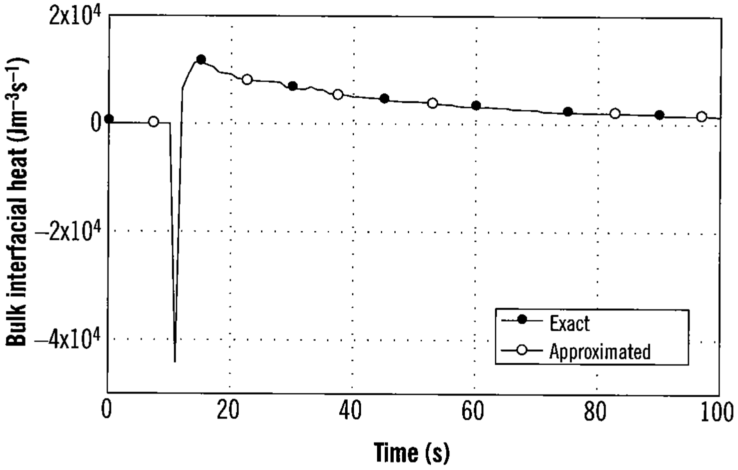

Figure 4.47. Approximation of the interfacial heat power in the vapor phase. Volume 10001. Break in vapor region

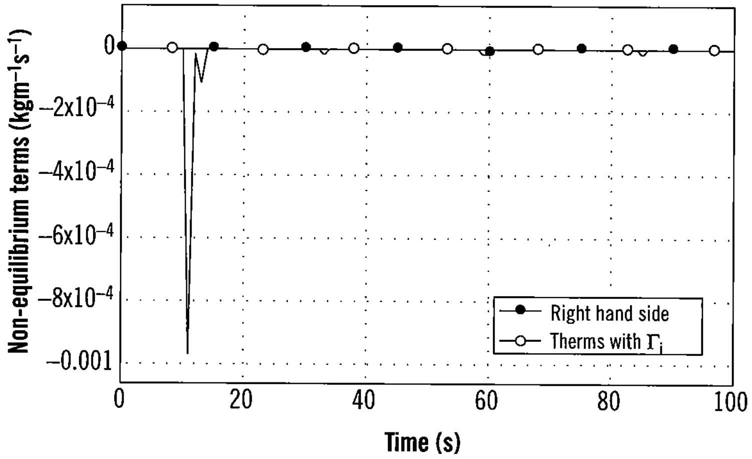

Figure 4.48. Neglection of term with Γ_i in the non-equilibrium eq. Liquid phase. Volume 20001. Break in vapor region

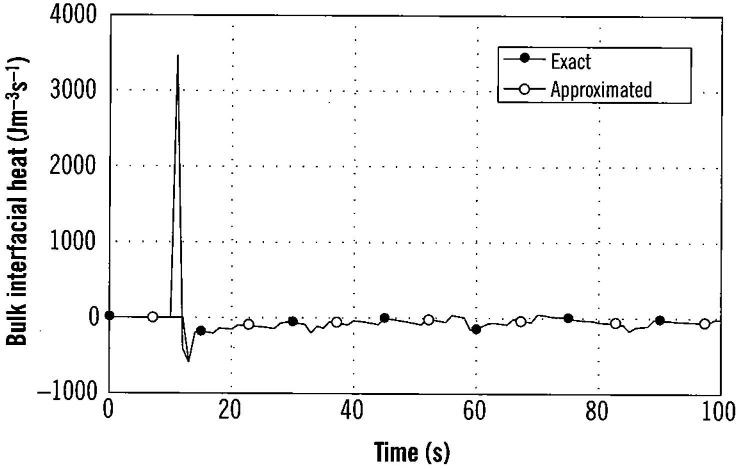

Figure 4.49. Approximation of the interfacial heat power in the liquid phase. Volume 20001. Break in vapor region

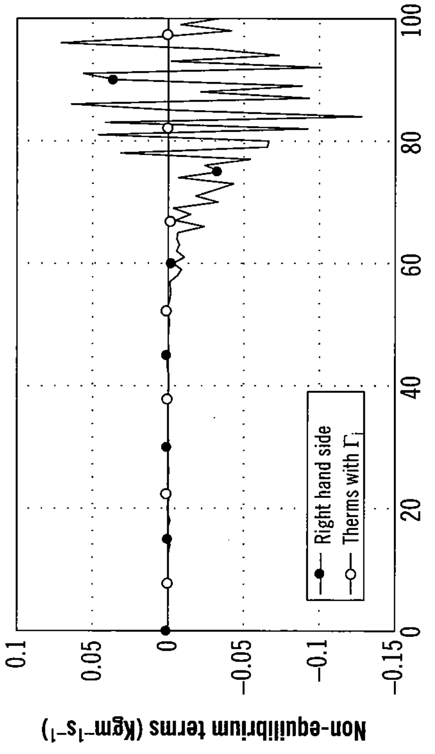

Figure 4.50. Neglection of term with Γᵢ in the non-equilibrium eq. Vapor phase. Volume 10001. Break in liquid region

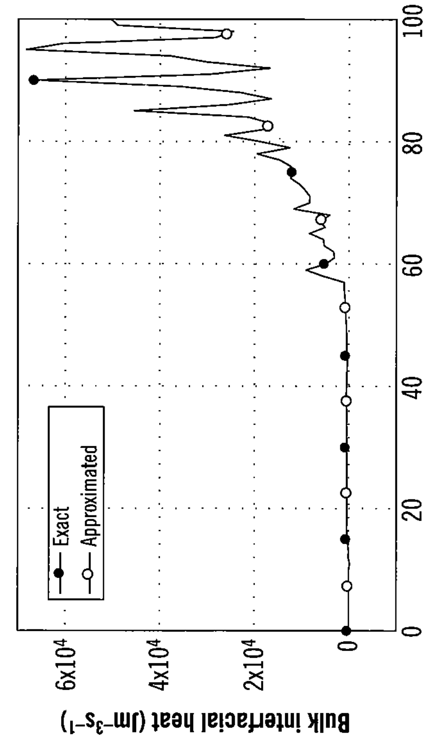

Figure 4.51. Approximation of the interfacial heat power in the vapor phase. Volume 10001. Break in liquid region

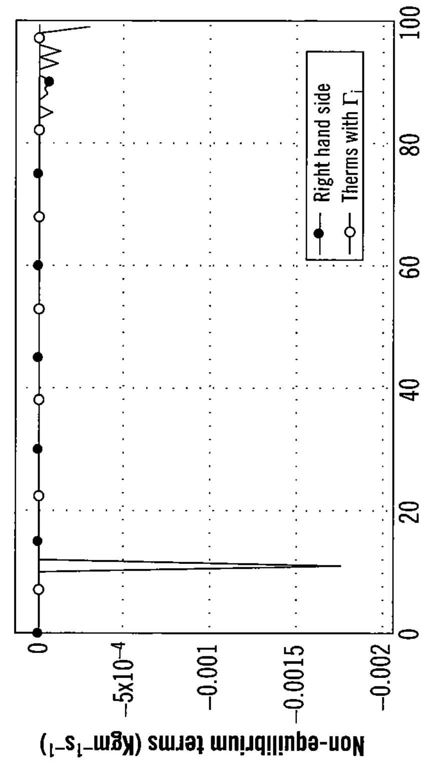

Figure 4.52. Neglection of term with Γᵢ in the non-equilibrium eq. Liquid phase. Volume 20001. Break in liquid region

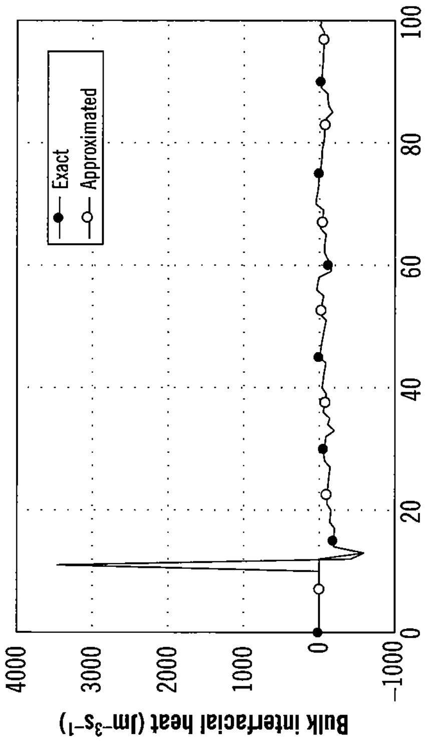

Figure 4.53. Approximation of the interfacial heat power in the liquid phase. Volume 20001. Break in liquid region

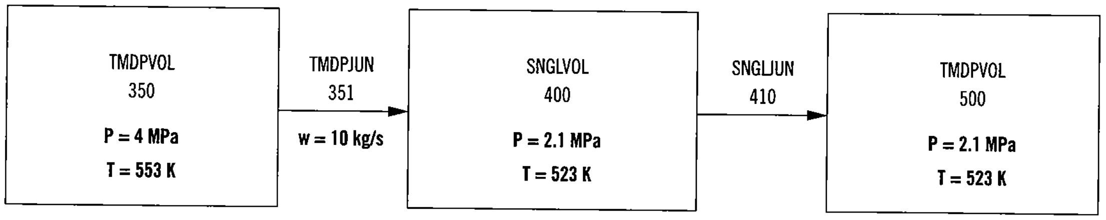

Figure 4.54. Zhang's problem

IV.4.2. Original case

The postprocessing of Zhang's example reveals some inconsistencies. First of all, the global pressure equation is not fully satisfied, as shown in figure 4.55. Figure 4.56 shows that, although a little liquid fraction appears in volume 400 during the transient, that fraction is not accounted for in the right hand side of the level equation. The vapor mass, represented in 4.57, is properly preserved. The non-equilibrium vapor mass balance in figure 4.58 is not fulfilled. The pressure drop derived from the momentum equation, and caused mainly by the friction in volume 400, has been represented against the difference P_{350}–P_{500} in figure 4.59. The mismatch between the two computed variables is evident.

Direct observation of some physical variables provides evidences of unphysical results as well. Figure 4.60 shows that the temperature in volume 400 decreases during the transient, contrary to expectations. The vapor enthalpy in figure 4.61 undergoes an unexpected variation too. The temperature and the enthalpy were just the two variables that moved Dr. Zhang to present his results. Other variables present an abnormal behavior, such as the density in figure 4.62, and the pressure in figure 4.63.

IV.4.3. Zhang's problem with PV term

Reference [41] tries to explain the results obtained in Zhang's problem and to provide a solution that leads to physical results.

That reference states that the internal energy equation (2.3) is not able to preserve the enthalpy, since its discretization leads to the following convective terms upstream and downstream, in the single phase case:

$$\left(\dot{\rho}_{j+1}\dot{u}_{j+1}+P_L\right)v_{j+1}A_{j+1} \qquad (4.1)$$

$$-\left(\dot{\rho}_j\dot{u}_j+P_L\right)v_jA_j \qquad (4.2)$$

where L stands for the single volume 400. The term (4.1) represents a donnor cell formulation of the enthalpy equation convective term in the downstream junction, while the term (4.2) does not represent an equivalent upstream term. For this to occur, the donnor cell pressure P_K should appear in the term instead of the downstream pressure P_L.

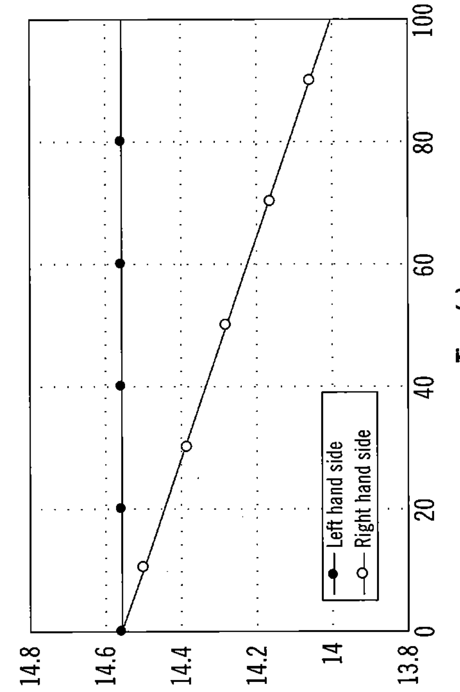

Figure 4.55. Zhang's problem: pressure postprocessing

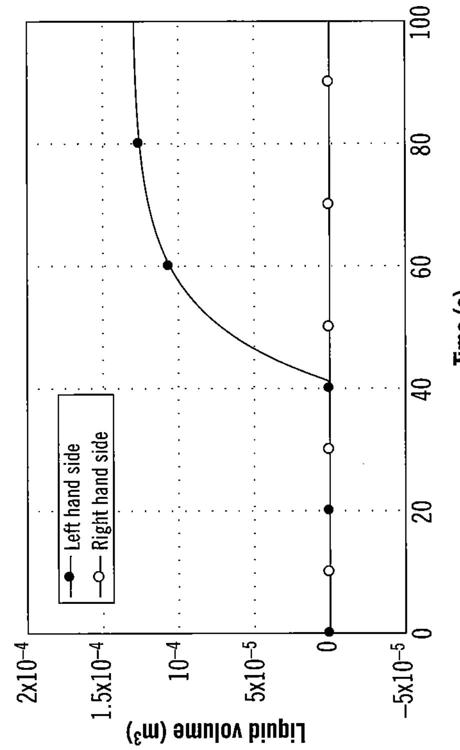

Figure 4.56. Zhang's problem: level postprocessing

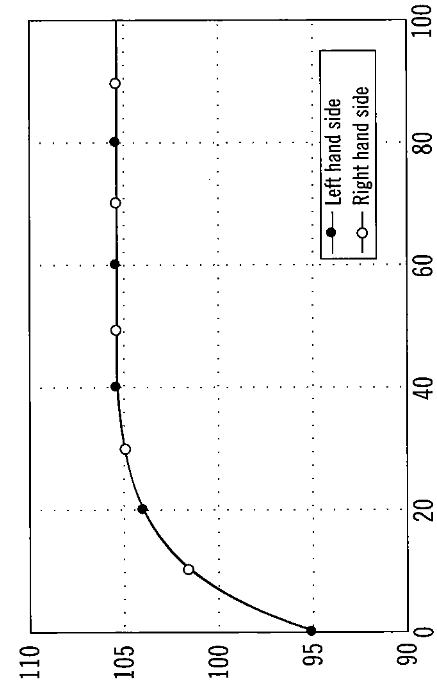

Figure 4.57. Zhang's problem: vapor mass postprocessing

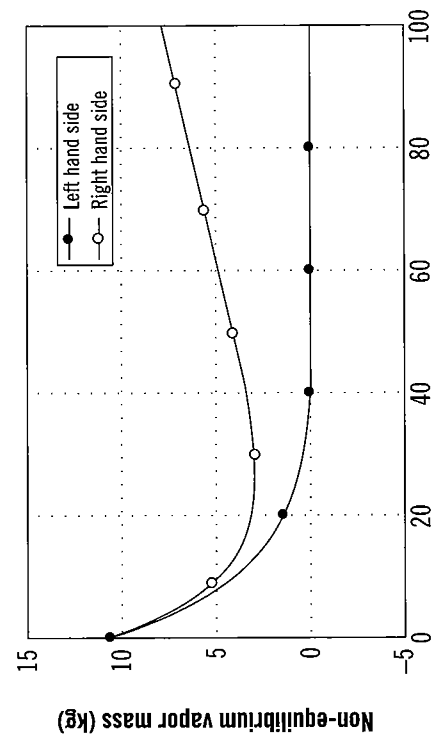

Figure 4.58. Zhang's problem: non-equilibrium vapor mass postprocessing

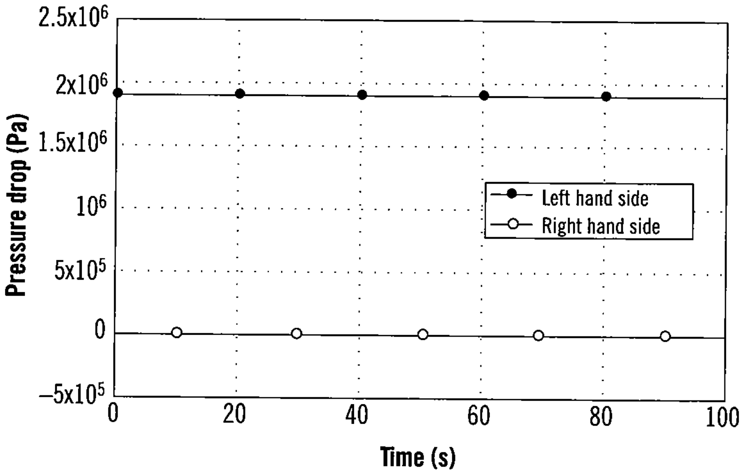

Figure 4.59. Zhang's problem: pressure drop postprocessing

To force this behavior, a new option has been added to the junctions of a developmental version of the RELAP5 code. Upon activation of this option, called the *e-flag*, the code adds the term $-(P_K-P_L)v_jA_j$ to the energy equation in the corresponding junction, where stands for he upstream TDV 350. This term is referred to as the PV term in the RELAP5 documentation.

The results obtained with the activation of the e-flag in the inlet time dependent junction couldn't have been postprocessed as the developmental version of RELAP5 including the PV term in the TDJs has not been available. Physical inconsistencies still arise anyway. Although the vapor enthalpy approaches asymptotically the value defined in volume 350 (see figure 4.65), the temperature does not show the same behavior (see figure 4.64). The temperature is increased during the transient, but it never reaches the value in volume 350. The density, represented in figure 4.66, remains constant in its initial value. The pressure, shown in figure 4.67, is kept in its initial value as well.

IV.4.4. Over specification suppression with pressure feedback

Having proved the inefficiency of the PV term, it is necessary to explain the cause of the unphysical results obtained with Zhang's problem and why the PV term fixes the enthalpy balance even when other variables such as the temperature and the density do not behave as expected. Single phase flow will be assumed in the following, which does not result in a lack of generality.

When using time dependent volumes only, without time dependent junctions, the following boundary conditions are specified:

- the upstream pressure,
- a thermodynamic property defined upstream (typically the temperature or the internal energy)
- the downstream pressure (the rest of variables defined in the downstream time dependent volume are not donnored and have no influence in the code results, although the code requests the user to input those values)

The number of boundary conditions is then three. The RELAP5 solves in this case the following set of linearized equations (see [37] for details):

$$Ax = b + g_1 v_{j+1}^{n+1} + g_2 v_j^{n+1} \tag{4.3}$$

where the matrices are defined as follows:

$$x = \begin{bmatrix} \tilde{u}_L^{n+1} - u_L^n \\ P_L^{n+1} - P_L^n \end{bmatrix} \tag{4.4}$$

$$A = \begin{bmatrix} \left(\dfrac{\partial \rho}{\partial u}\right)_L^n & \left(\dfrac{\partial \rho}{\partial P}\right)_L^n \\ u_L \left(\dfrac{\partial \rho}{\partial u}\right)_L^n + \rho_L^n & u_L^n \left(\dfrac{\partial \rho}{\partial P}\right)_L^n \end{bmatrix} \qquad b = \begin{bmatrix} 0 \\ \left(Q_{\mathrm{w},L}^n + \mathrm{DISS}_L^n\right)\Delta t \end{bmatrix} \tag{4.5}$$

$$g_1 = \begin{bmatrix} \dot{\rho}_{j+1}^n A_{j+1} \Delta t / V_L \\ -\left(\dot{\rho}_{j+1}^n \dot{u}_{j+1}^n + P_L^n\right) A_{j+1} \Delta t / V_L \end{bmatrix} \qquad g_2 = \begin{bmatrix} -\dot{\rho}_j^n A_j \Delta t / V_L \\ \left(\dot{\rho}_j^n \dot{u}_j^n + P_L^n\right) A_j \Delta t / V_L \end{bmatrix} \tag{4.6}$$

It is assumed that the density in matrix A depends on the pressure P and the internal energy u. When a partial derivative of the density is present in that matrix, with respect to a given variable, it must be understood that the other variable is kept constant.

The two linear equation system (4.3) is obtained from the mass and energy equations, discretized in the control volume L. The whole system is obtained by adding the two equations obtained from the discretization of the momentum equation in the junctions j and $j+1$:

$$\begin{aligned} -\Delta t\left(P_L^{n+1} - P_j^{n+1}\right) &= -\rho_j^n B_x \frac{\Delta x_L}{2} \Delta t \\ &+ \frac{1}{2}\dot{\rho}_j^n \left[\left(v^2\right)_L^n - \left(v^2\right)_j^n\right] \Delta t - \rho_j^n \frac{\Delta x_L}{2} v_j^n \\ &+ \left[\rho_j^n \frac{\Delta x_L}{2}\left(1 + \mathrm{FW}_j^n \Delta t\right) + \dot{\rho}_j^n \mathrm{HLOSS}_j^n \Delta t\right] v_j^{n+1} \end{aligned} \tag{4.7}$$

$$\begin{aligned} -\Delta t\left(P_{j+1}^{n+1} - P_L^{n+1}\right) &= -\rho_{j+1}^n B_x \frac{\Delta x_L}{2} \Delta t \\ &+ \frac{1}{2}\dot{\rho}_{j+1}^n \left[\left(v^2\right)_{j+1}^n - \left(v^2\right)_L^n\right] \Delta t - \rho_{j+1}^n \frac{\Delta x_L}{2} v_{j+1}^n \\ &+ \left[\rho_{j+1}^n \frac{\Delta x_L}{2}\left(1 + \mathrm{FW}_{j+1}^n \Delta t\right) + \dot{\rho}_{j+1}^n \mathrm{HLOSS}_{j+1}^n \Delta t\right] v_{j+1}^{n+1} \end{aligned} \tag{4.8}$$

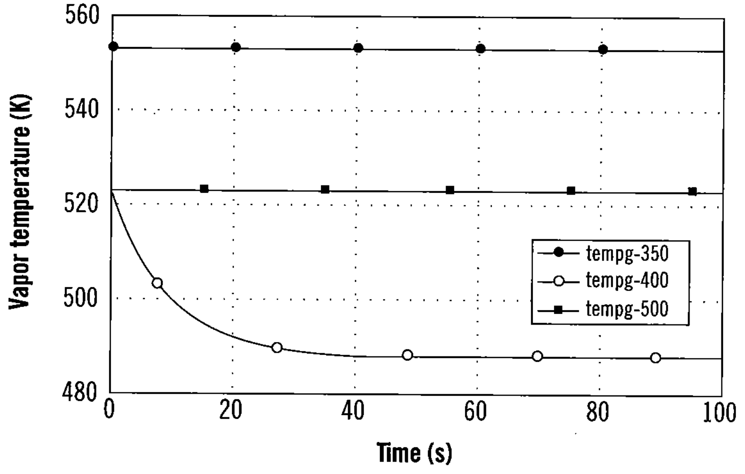

Figure 4.60. Zhang's problem: vapor temperature

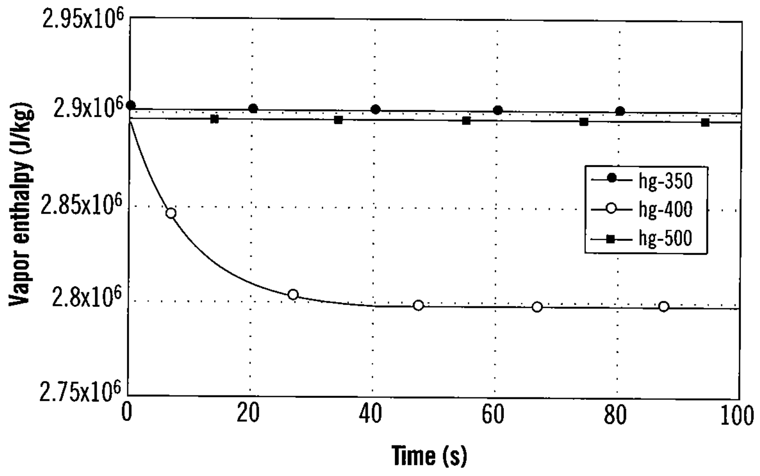

Figure 4.61. Zhang's problem: vapor enthalpy

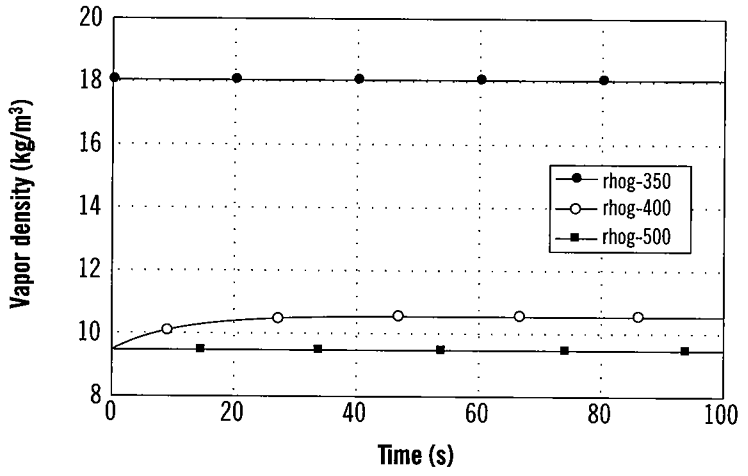

Figure 4.62. Zhang's problem: vapor density

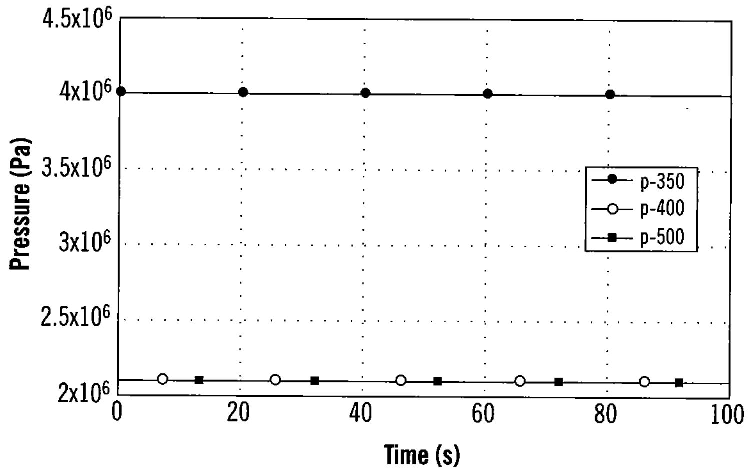

Figure 4.63. Zhang's problem: pressure

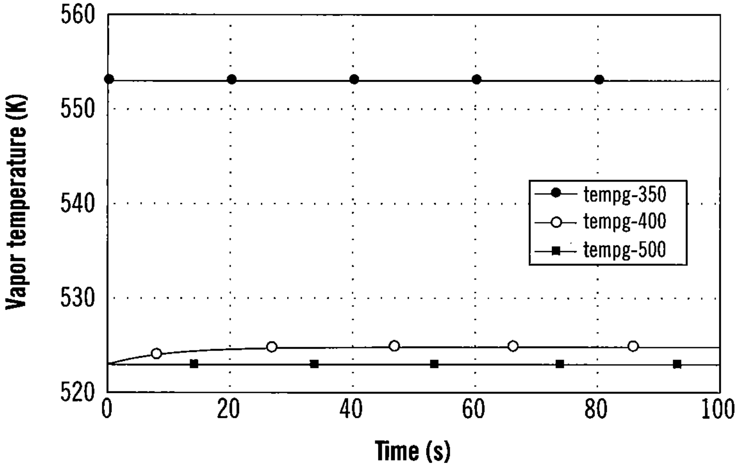

Figure 4.64. Vapor temperature with PV term

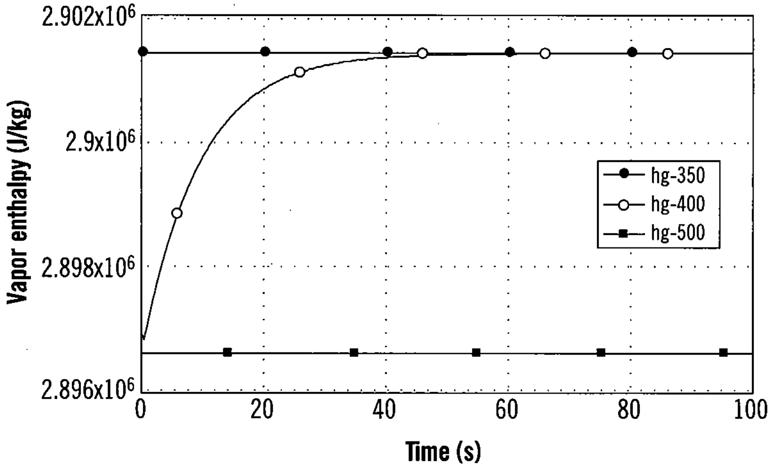

Figure 4.65. Vapor enthalpy with PV term

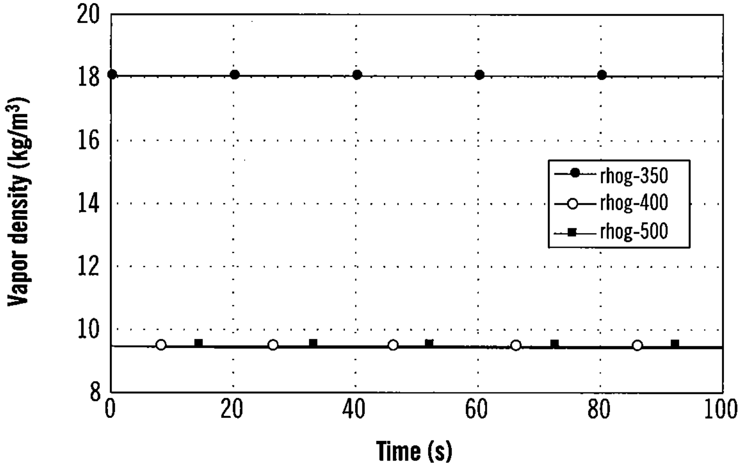

Figure 4.66. Vapor density with PV term

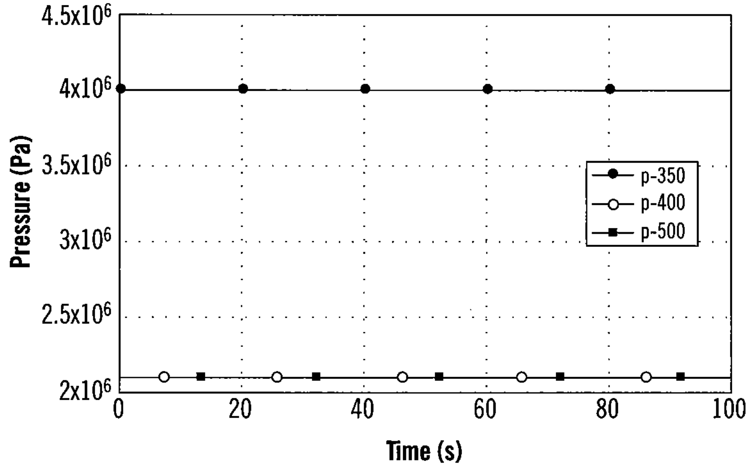

Figure 4.67. Pressure with PV term

The previous four equation set has the following unknowns: P_L^{n+1} , u_L^{n+1}, υ_j^{n+1} y υ_{j+1}^{n+1} . The variable P_j^{n+1} in equation (4.7) refers to the pressure in the upstream TDV.

When a time dependent junction is used, a new boundary condition, the inlet velocity, is added. This makes a total of four boundary conditions. This increase of the number of degrees of freedom is a clear index of error, since that number is inherent to the physical system and should not be changed by the choice of the set of variables that will act as boundary conditions. The increase of boundary conditions is caused by the non-solving of the momentum equation in the TDJ. It's mistakenly believed that this equation is not necessary, since inlet velocities are provided. As a consequence, the discrete momentum equation in the inlet junction is suppresed, and the number of equations decreases from four to three. The mathematical problem is still consistent, since the unknown variables are also three $\left(P_L^{n+1}, u_L^{n+1}\text{, and } \upsilon_{j+1}^{n+1}\right)$. However, it is not consistent from the physical point of view, since the number of equations cannot be arbitrarily decreased. Once the momentum equation (4.8) is not solved, the pressures P_L^{n+1} and P_j^{n+1} are no longer linked by any phisical principle. This observation modifies all the forms of the energy equation, since the upstream pressure P_j^{n+1} which should be computed by the code defines, along with the upstream temperature, a wrong density and internal energy. These two variables are introduced in the energy equation through the convective terms.

If the momentum equation were solved in the inlet junction, the system of linear equations would be:

$$Ax = c + g_1 \upsilon_{j+1}^{n+1} \tag{4.9}$$

where

$$c = \begin{bmatrix} -\dot{\rho}_j^n A_j \Delta t / V_L \\ \left(Q_{\mathrm{w},L}^n + \mathrm{DISS}_L^n\right)\Delta t + \left(\dot{\rho}_j^n \dot{u}_j^n + P_L^n\right) A_j \Delta t / V_L \upsilon_j^{n+1} \end{bmatrix} \tag{4.10}$$

Note that c is the result of adding b and $g_2 \upsilon_j^{n+1}$, since the latter term is no longer an unknown. The momentum equations (4.7) and (4.8) are also a part of the system. Nevertheless, the unknown variable in equation (4.7) is now P_j^{n+1} instead of υ_j^{n+1}. With this system the number of equations (four) and unknowns (P_L^{n+1}, $u_L^{n+1}, \upsilon_{j+1}^{n+1}$ y P_j^{n+1}) is preserved.

Once the root cause of the abnormal results obtained in Zhang's problem has been found, it is necessary to explain why the PV term is enough to preserve the enthalpy, even though the momentum equation is not solved in the inlet junction yet.

As the spatial mesh is finer and finer, the PV term tends to $\upsilon A \dfrac{\partial P}{\partial x}$.

Adding that term to the analytical form of the internal energy equation (2.3), assuming single phase regime, unheated pipe, and neglecting the energy dissipated by friction, such an equation becomes:

$$\frac{\partial}{\partial t}\left(A\rho u\right)+\frac{\partial}{\partial x}\left(A\rho u v\right)+vA\frac{\partial P}{\partial x}=-P\frac{\partial Av}{\partial x} \tag{4.11}$$

Grouping some terms, and taking into account that the enthalpy is expressed as $h=u+\frac{P}{\rho}$, we obtain

$$\frac{\partial}{\partial t}\left(A\rho h\right)+\frac{\partial}{\partial x}\left(A\rho h v\right)=\frac{\partial}{\partial t}\left(AP\right) \tag{4.12}$$

Integrating (4.12) in space and time:

$$m_k \approx \sum_{l=1}^{N}\alpha_k^l V^l \rho_k^l \tag{4.13}$$

If the temporal changes of the pressure are negligible as compared to the terms carrying the enthalpy, the previous equation becomes an enthalpy balance with no influence of the upstream and downstream pressures. Even if the pressure gradient is very large, as it is the case in Zhang's problem, the balance (4.13) will be preserved anyway. This the reason why the enthalpy balance is fulfilled when the PV term is introduced.

It must be highlighted that equations (4.11) and (4.12) do not represent any physical principle, since they are derived from adding a ficticious term to a physically meaningfull equation.

It has been proven that the root cause of Zhang's results is the over specification of boundary conditions, which derives in turn from skipping the computation of the momentum equation in the time dependent junctions. The ideal solution would be the modification of the RELAP5 code, to apply a different solution algorithm when TDJs are used. However, a less traumatic solution can be devised. It consists in computing the momentum equation in the inlet junction through the control variables supplied by the code.

The pressure in the upstream time dependent volume is no longer set by the user. It equals the pressure in the centre of volume 400, plus the pressure drop due to the momentum equation. This approach leads to the following equation, which is valid for single phase flow and a pipe with uniform cross-section:

$$\rho\left[\frac{\partial v}{\partial t}+\frac{1}{2}\frac{\partial v^2}{\partial x}\right]=-\frac{\partial P}{\partial x}-\mathrm{FWAL}v \tag{4.14}$$

Denoting the inlet time dependent junction by j, the volume number 400 by L and the inlet time dependent junction by , the semi-implicit discretization of equation (4.14) yields:

$$\rho_j \frac{\partial v_j}{\partial t} + \frac{1}{2}\rho_j \frac{v_L^2 - v_j^2}{\Delta x_j} = -\frac{P_L - P_K}{\Delta x_j} - \mathrm{FWALL_L} v_L \qquad (4.15)$$

Taking into account that $\Delta x_j = \frac{\Delta x_L}{2}$, the upstream pressure can be computed as:

$$P_K = P_L + \rho_j \frac{\partial v_j}{\partial t}\frac{\Delta x_L}{2} + \frac{1}{2}\rho_j\left(v_L^2 - v_j^2\right) + \mathrm{FWAL_L} v_L \frac{\Delta x_L}{2} \qquad (4.16)$$

If the boundary condition in the inlet time depedent junction is specified as a constant velocity, instead of as a constant mass flow, the term with the time derivative in equation (4.16) is cancelled. The computation of a numerical derivative is then avoided. This little change does not affect the essence of Zhang's problem.

The momentum equation (4.16) was implemented through RELAP5 control variables to feedback the pressure into the upstream TDV. The following results were obtained.

Figures 4.68 to 4.71 show the results of postprocessing the pressure, the volume, the vapor mass and the vapor non-equilibrium mass respectively. The agreement between both sides of the respective equations is excellent. A discrepancy between both sides of the pressure drop equation, represented in figure 4.72, is observed during the first time step. The cause is the joint effect of the discretization and the initialization of the velocities to zero values. The system spend this first time step to reach the inlet velocity specified through the TDJ.

The quality of the solution obtained by feeding the pressure back is confirmed by the good behavior of other variables, such as the temperature in volume 400 shown in figure 4.73. This temperature tends to that in the inlet TDV as expected. This behavior is not observed when using the PV term, as shown in figure 4.64. The enthalpy in volume 400 also approaches the value in the inlet TDV, according to figure 4.74. The density in figure 4.75 behaves in a similar way. It can be realized in figure 4.76 that the pressure is almost identical in the three volumes, because the frictional pressure drop is negligible as compared to the global system pressure.

IV.5. Full plant model. Flow mixing problem

IV.5.1. Introduction

In this last example a transient in a full plant model is analyzed. The transient, which is one of the sequences obtained in a Probabilistic Safety Analysis, consists in a steam generator tube rupture with HPIS failure, CVCS failure, accumulator success, steam dump success and RHRS success.

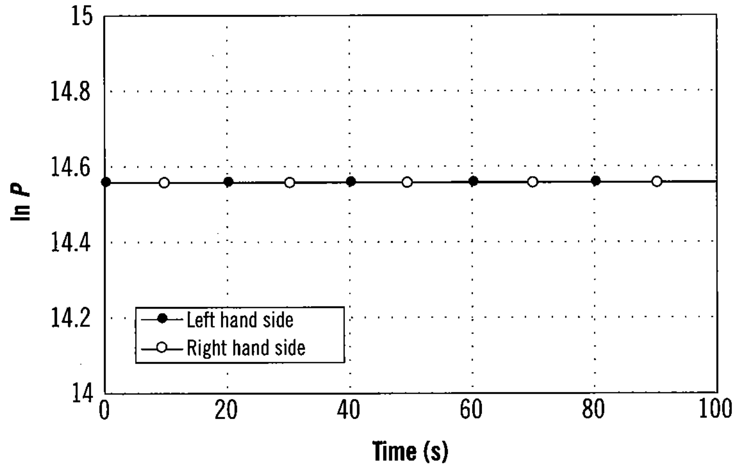

Figure 4.68. Pressure equation with momentum calculation

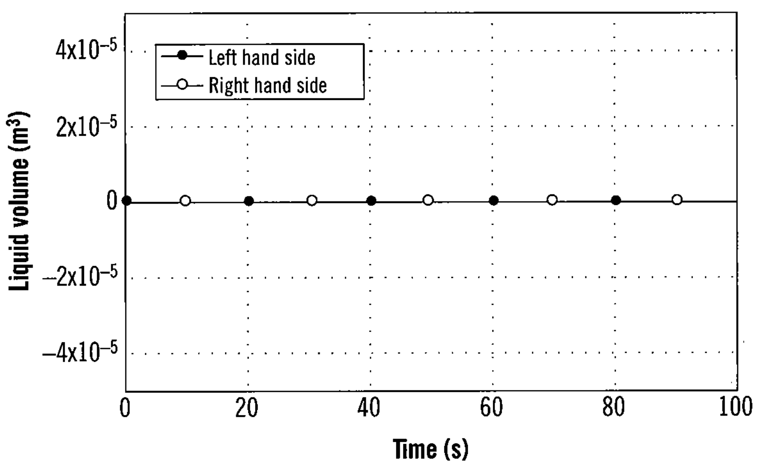

Figure 4.69. Liquid volume equation with momentum calculation

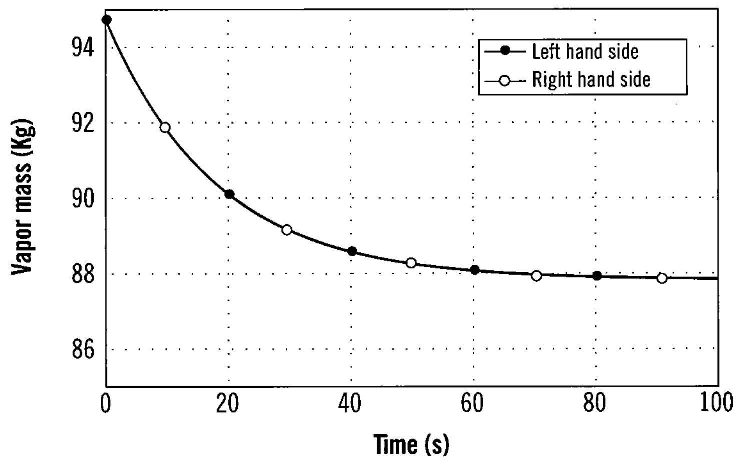

Figure 4.70. Mass equation with momentum calculation

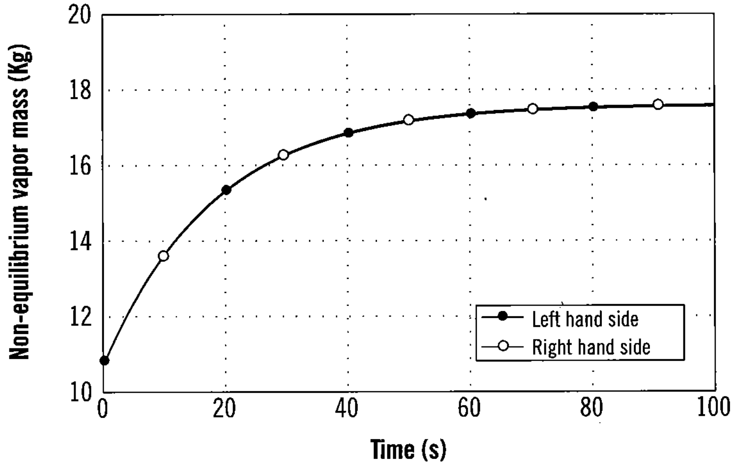

Figure 4.71. Non-equilibrium mass equation with momentum calculation

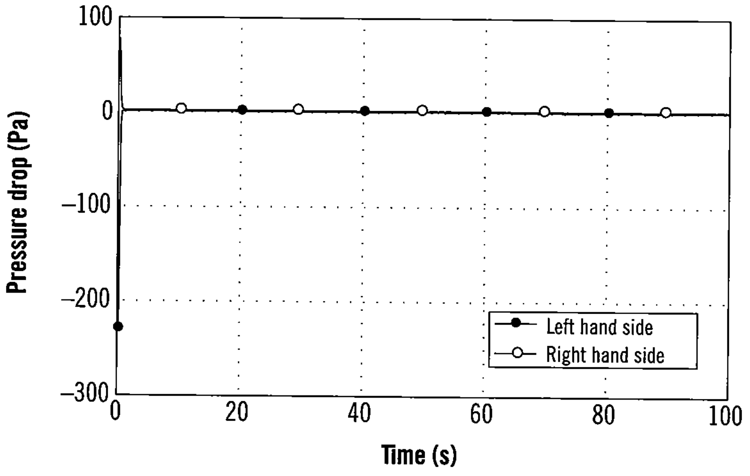

Figure 4.72. Pressure drop equation with momentum calculation

The subsystem to be analyzed is the downcomer in one of the steam generators, since the transient seems to be very demanding in that component. Flow reversal can be observed in figure 10.

Figure 4.78 shows the analysis of the global pressure equation in the downcomer. The comparison between both sides of the equation is completely unsatisfactory. On the other hand, the analysis of the liquid volume equation (see figure 4.79), is acceptable. The analysis of the vapor and liquid mass equations (figures 4.80 and 4.81) is also satisfactory except for a certain offset in the vapor, which can be attributed to a certain spike in the derivative of that quantitie. A direct comparison of the derivatives of both sides of the equation would be more convincing. The analyses of the non-equilibrium vapor and liquid mass equations (figures 4.82 and 4.83) is not satisfactory either, as they are affected by the pressure equation. Figure 4.84 shows the terms of the pressure and liquid volume equations. The inlet vapor volumetric flow and the vapor swelling are the dominant terms. Both terms tend to compensate each other.

A possible explanation for the results obtained in the analysis of the pressure equation is the flow mixing that takes place at the downcomer inlet. The feedwater, separator and recirculation flows are mixed at that point. The three flows have very different thermodynamic properties. To prove this hypothesis, the same analysis have been undertaken again excluding the first control volume in the downcomer. The resulting subsystem is strictly one-dimensional, with one inlet junction and one outlet junction.

Figure 4.85 shows that the pressure analysis is more satisfactory when removing the first control volume. Similar comments apply to the liquid volume and vapor and liquid mass equations (figures 4.86 to 4.88). However, fullfilment of the non-equilibrium mass equations (figures 4.89

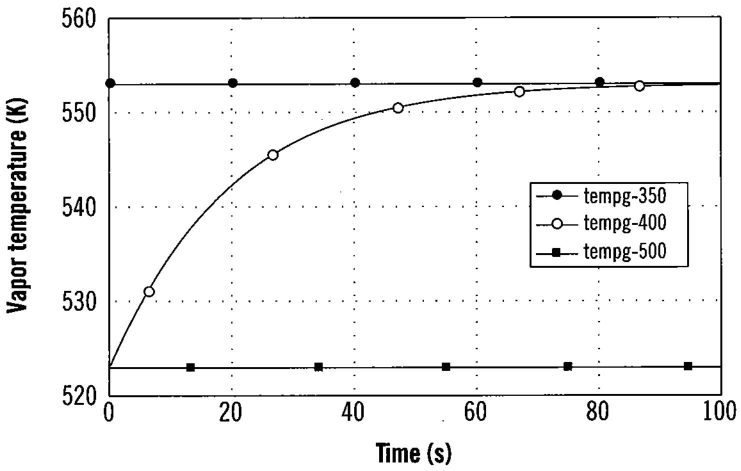

Figure 4.73. Vapor temperature with momentum calculation

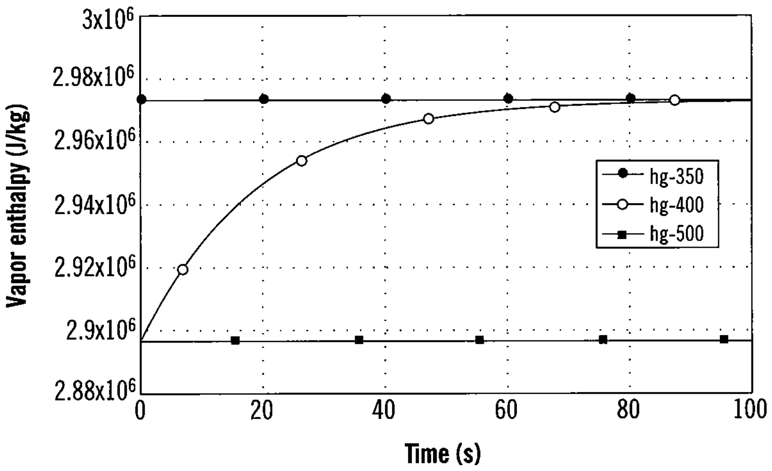

Figure 4.74. Enthalpy with momentum calculation

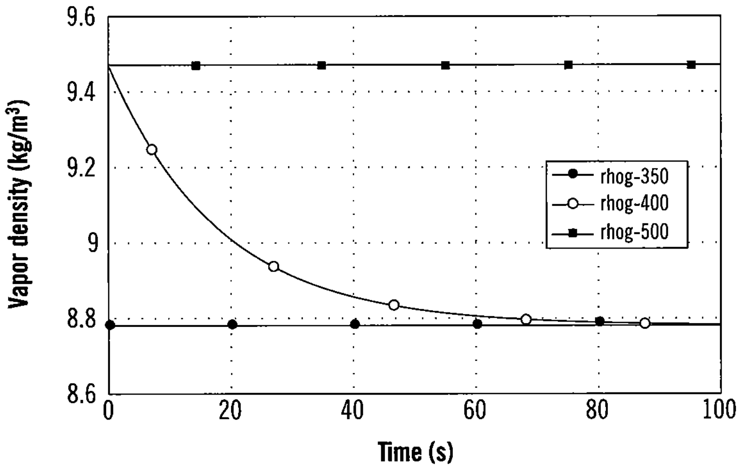

Figure 4.75. Vapor density with momentum calculation

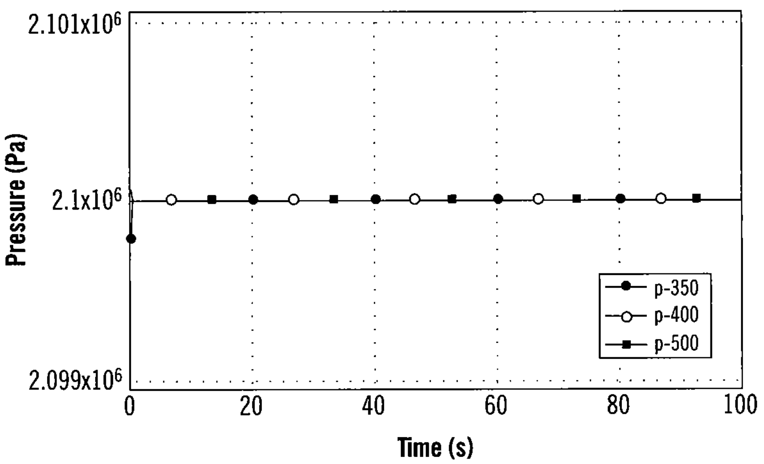

Figure 4.76. Pressure with momentum calculation

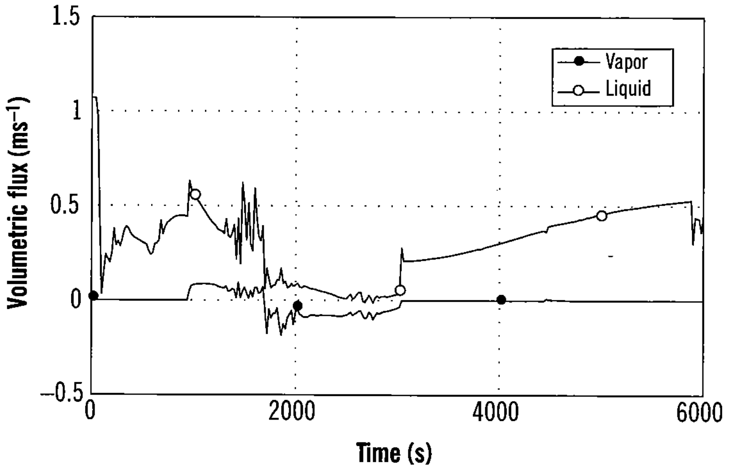

Figure 4.77. Volumetric flow in junction 5 of the downcomer

and 4.90) is not properly achieved yet. The pressure equation terms (figure 4.91) resemble those of the analysis with the whole downcomer.

The results seem to confirm that flow mixing is not properly treated by the RELAP5 code. The following lines will try to explain why the RELAP5 treatment of flow mixing is not satisfactory.

The one-dimensional formulation of two-phase flow dynamics allows to solve problems in pipes with two well defined ends (i.e., inlet and outlet). In this kind of pipes the direction along the pipe plays the role of the independent variable in a parametric representation, ranging from 0 to L, where the latter is the whole length of the pipe. In the case of flow mixing, where pipes converge into a single one as described in figure 4.92, the overall problem comprises actually n+1 one-dimensional subproblems, each one defined in a range $[0,L_i]$. The n+1 are coupled among them. They are linked in such a way that the boundary conditions supplied to the pipe on the right hand side in figure 4.92 fullfil the conservation equations. Those boundary conditions must be obtained by integrating on a null volume (i.e., with zero length) the conservation equations (2.1), (2.2) and (2.3). Since the integral of the volumetric sources is considered to be zero in such a null volume, only the convective terms, i.e. those affected by partial derivatives with respect to the spatial variable, contribute to the equations. The integration of a convective term leads to a difference between the sums of inlet and outlet flows. If a convective terms has the form $\frac{\partial \Psi v}{\partial x}$, where Ψ is a generic variable and v is the flow velocity, the integral on a zero volume results in $\Psi v\big|^{\text{out}} - \Psi v\big|^{\text{in}}$. Note that integration on a zero volume requires that the partial derivative with respect to time affect all the factors, because terms such as would not have a well defined integral. As a conclusion, integration on a null volume must only be applied to the so called *conservative form* of the conservation equations.

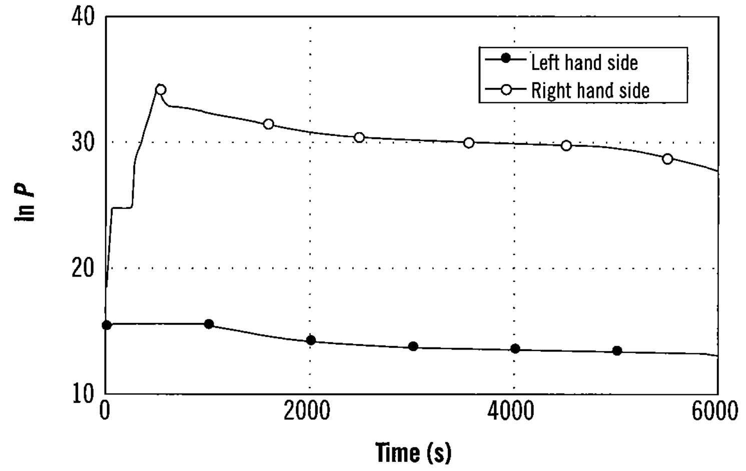

Figure 4.78. Analysis of the pressure equation in the downcomer

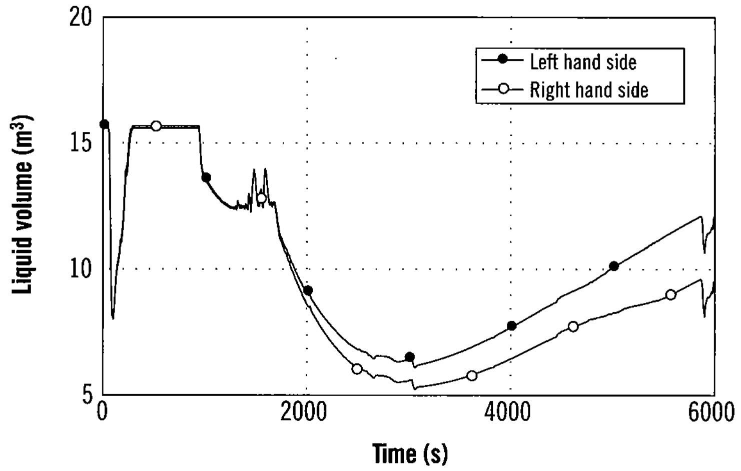

Figure 4.79. Analysis of the liquid volume equation in the downcomer

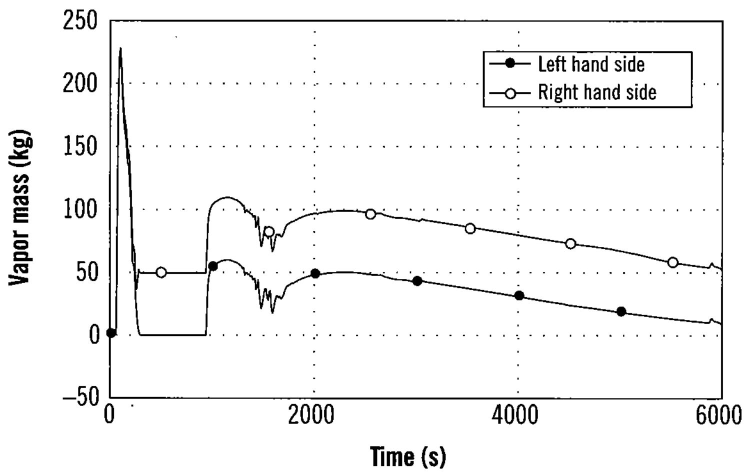

Figure 4.80. Analysis of the vapor mass equation

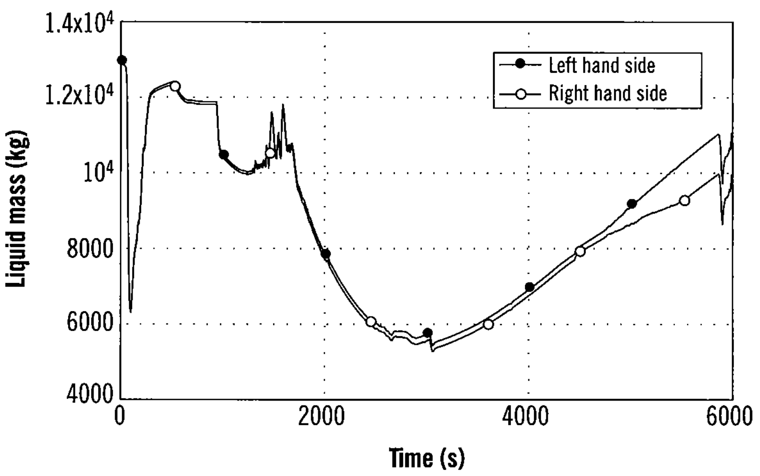

Figure 4.81. Analysis of the liquid mass equation in the downcomer

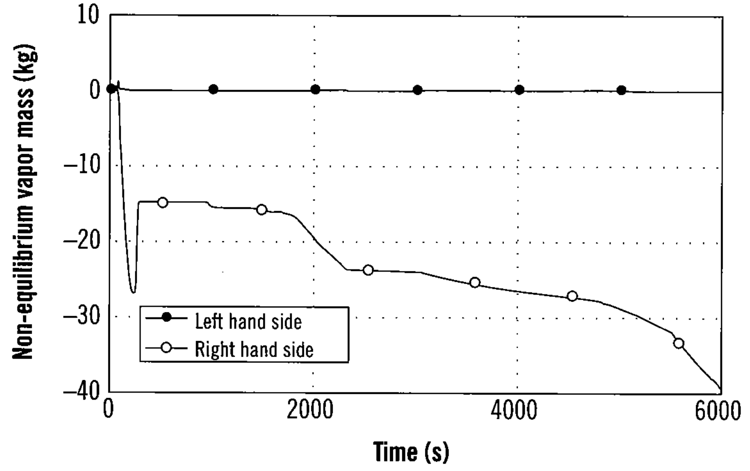

Figure 4.82. Analysis of the non-equilibrium vapor mass equation in the downcomer

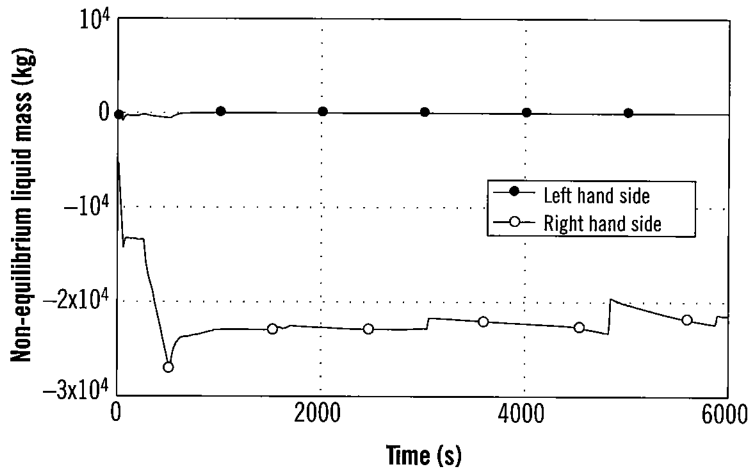

Figure 4.83. Analysis of the non-equilibrium liquid mass equation in the downcomer

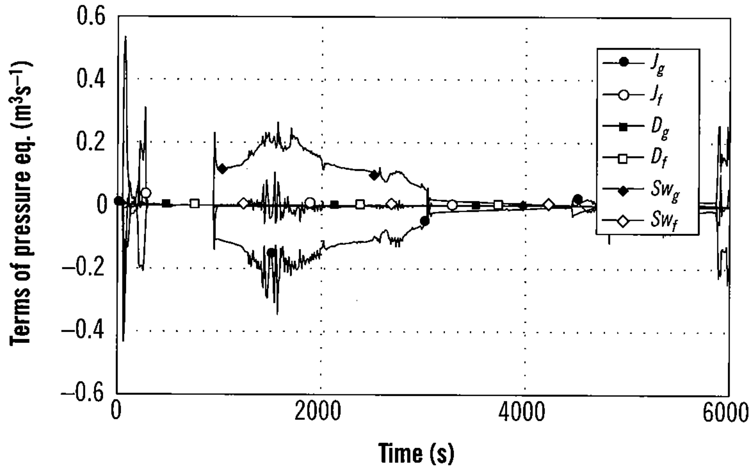

Figure 4.84. Terms of the pressure equation in the downcomer

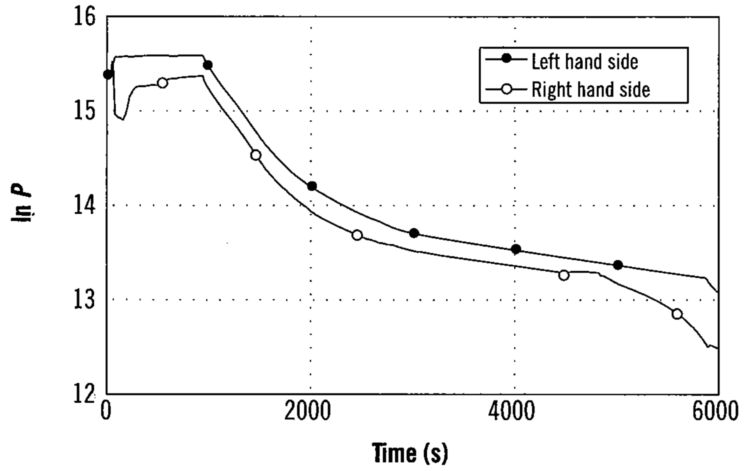

Figure 4.85. Analysis of the pressure equation in the downcomer subsystem

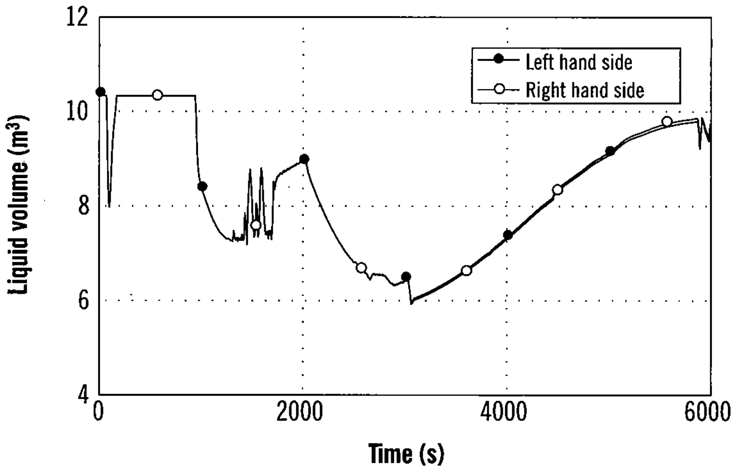

Figure 4.86. Analysis of the level equation in the downcomer subsystem

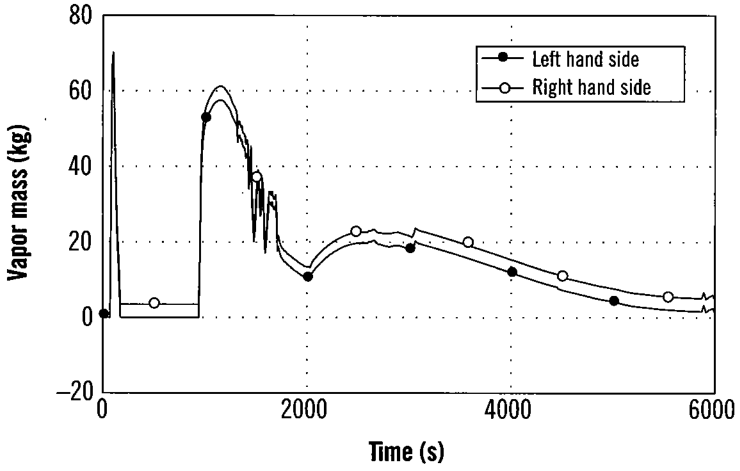

Figure 4.87. Analysis of the vapor mass equation in the downcomer subsystem

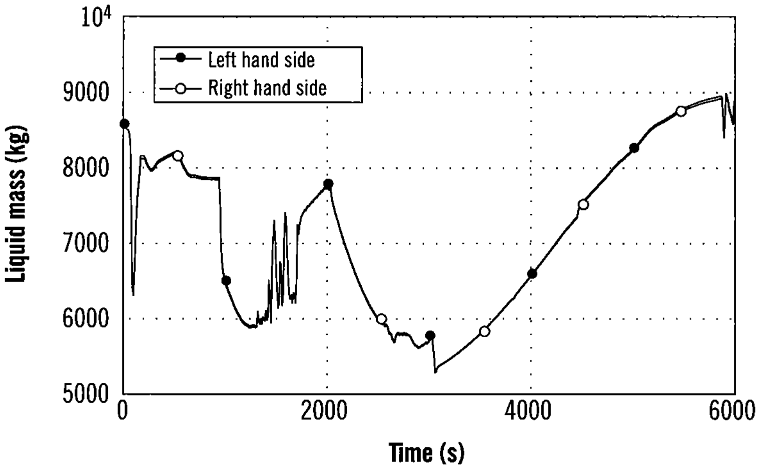

Figure 4.88. Analysis of the liquid mass equation in the downcomer subsystem

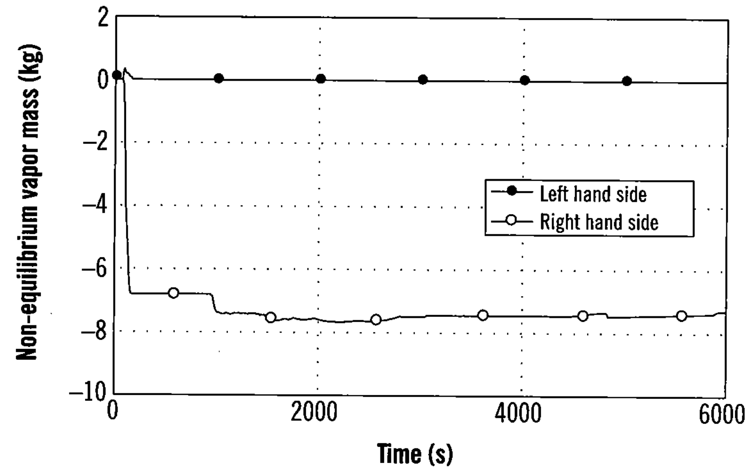

Figure 4.89. Analysis of the non-equilibrium vapor mass equation in the downcomer subsystem

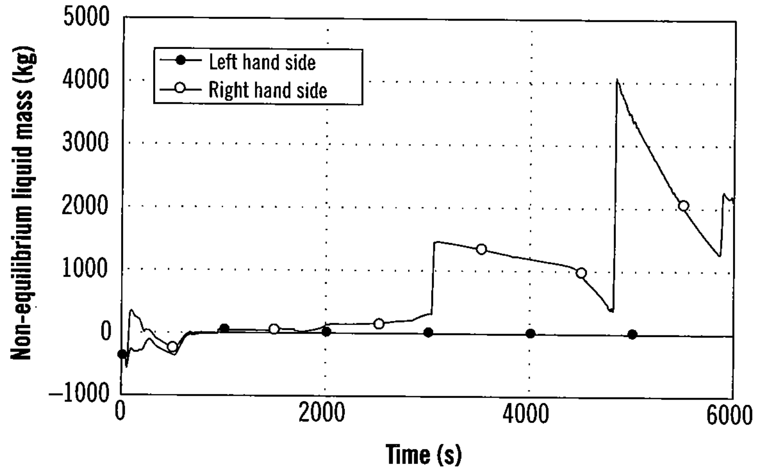

Figure 4.90. Analysis of the non-equilibrium liquid mass equation in the downcomer subsystem

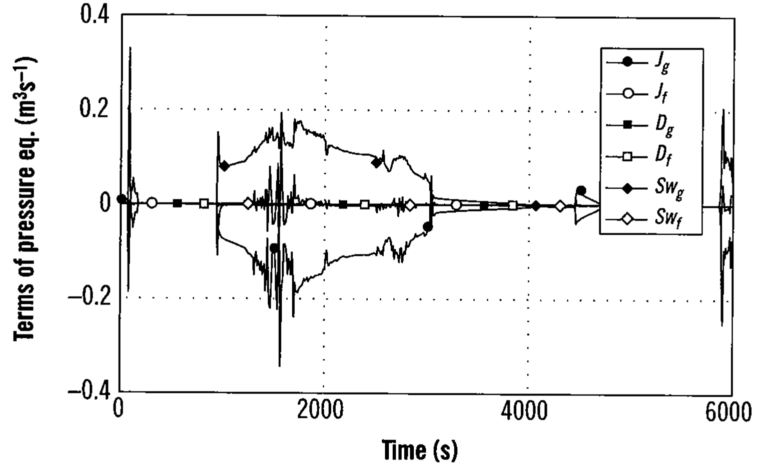

Figure 4.91. Pressure equation terms in the downcomer subsystem

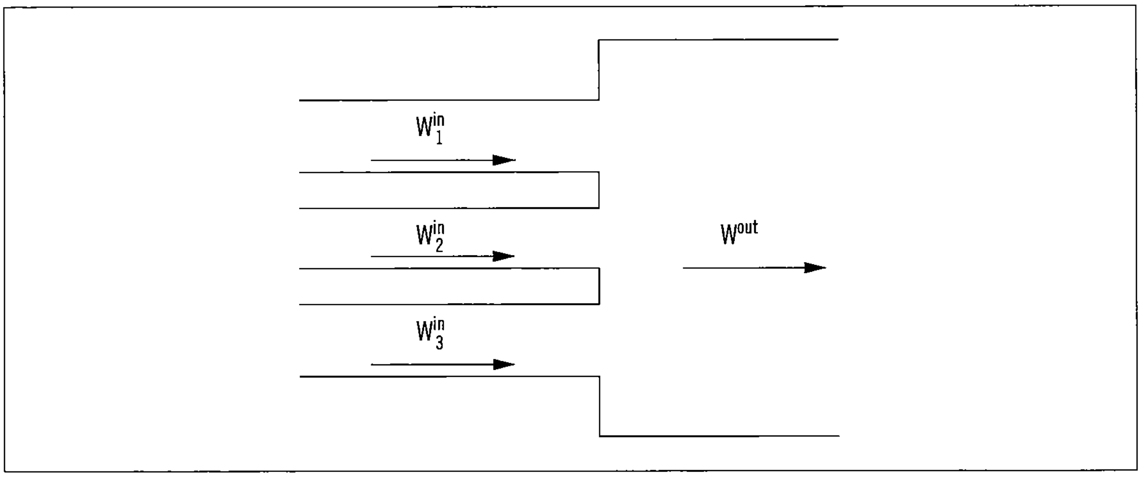

Figure 4.92. One-dimensional flow mixing

For the mass equation, the convective terms meet that requirement to yield

$$\sum \alpha_k A\rho_k \upsilon_k \Big|^{\text{in}} = \alpha_k A\rho_k \upsilon_k \Big|^{\text{out}} \tag{4.17}$$

The internal energy equation includes a term of the form $-P\frac{\partial}{\partial x}\left(\alpha_k A\upsilon_k\right)$, which cannot integrated directly. I.e. the internal energy equation is not a conservative form. In order to introduce the pressure into the gradient, the kinetic energy equation is added to the internal energy equation, which results from multiplying the momentum equation by the velocity. The kinetic energy equation is as follows:

$$\begin{aligned}
&\frac{1}{2}\alpha_k A\rho_k \frac{\partial \upsilon_k}{\partial t} + \frac{1}{2}\alpha_k A\rho_k \frac{\partial \upsilon_k^2}{\partial x} = -\alpha_k A\upsilon_k \frac{\partial P}{\partial x} + \alpha_k \rho_k B_x A\upsilon_k \\
&-\alpha_k \rho_k A\mathrm{FW}_k \upsilon_k^2 + A\Gamma_k \upsilon_\mathrm{i} \upsilon_k - \alpha_k \rho_k A\mathrm{FI}_k \left(\upsilon_k - \upsilon_{k'}\right)\upsilon_k \\
&-C\alpha_k \alpha_{k'} \rho_\mathrm{m} A\left[\frac{\partial}{\partial t}\left(\upsilon_k - \upsilon_{k'}\right) + \upsilon_{k'}\frac{\partial \upsilon_k}{\partial x} - \upsilon_k \frac{\partial \upsilon_{k'}}{\partial x}\right]\upsilon_k \\
&-\frac{1}{2}K\alpha_k \rho_k \upsilon_k^3 \delta\left(x - x_0\right)
\end{aligned} \tag{4.18}$$

where the continuity equation (2.1) has been taken into account.

Adding (4.18) to (2.3), the term with the pressure takes the integrable form $-\frac{\partial}{\partial x}\left(\alpha_k PA\upsilon_k\right)$. The mixing condition is then:

$$\sum \alpha_k A\rho_k \left(u_k + \frac{1}{2}\upsilon_k^2\right)\upsilon_k \Bigg|^{\text{in}} - \alpha_k A\rho_k \left(u_k + \frac{1}{2}\upsilon_k^2\right)\upsilon_k \Bigg|^{\text{out}} = -\sum \alpha_k PA\upsilon_k \Big|^{\text{in}} + \alpha_k PA\upsilon_k \Big|^{\text{out}} \tag{4.19}$$

Taking into account that $h = u + P\vartheta$, the previous equation can be written as

$$\sum \alpha_k A\rho_k\left(h_k+\frac{1}{2}\upsilon_k^2\right)\upsilon_k\Bigg|^{\text{in}} = \alpha_k A\rho_k\left(h_k+\frac{1}{2}\upsilon_k^2\right)\upsilon_k\Bigg|^{\text{out}} \tag{4.20}$$

The momentum equation (2.2) cannot be directly integrated on a null volume, since a term of the form $-\alpha_k A\frac{\partial P}{\partial x}$ is found on the right hand side. The factors α_k and A, which lie inside the partial derivative in the original three-dimensional equation, are factored out in the one-dimensional form by using the Gauss divergence theorem:

$$\alpha_k A\frac{\partial P}{\partial x} = \frac{\partial}{\partial x}\left(\alpha_k AP\right) + \lim_{\Delta x\to 0}\frac{1}{\Delta x}\int_{A_I} Pn_x dA + \lim_{\Delta x\to 0}\frac{1}{\Delta x}\int_{A_W} Pn_x dA \tag{4.21}$$

where A_I is the interfacial area within the pipe interval Δx, is the wall area in the same interval, and n_x is the projection of the unit normal vector to the surface (either interface or wall) onto the axis. Even assuming that the interfacial term is zero, the wall term would still remain. In case of flow mixing, or discontinuous area change, the computation of that term would remain undetermined. If the discontinuity is seen as the limit case of a series of problems with ever increasing slope in the representation of the cross-sectional area with respect to the x direction, the final result would depend on the infinite possible ways of approximation. A relationship between inlet and outlet properties can be found for the mass and total energy equations because the terms with spatial derivative represent fluxes (i.e. they include the factor $A\upsilon_k$). Since the velocity is parallel to the wall, the flux through the wall is always zero. On the other hand, the term with the spatial derivative in the momentum equation is not a flux and thus the corresponding wall integral is not zero.

As a conclusion, it is impossible to rigorously describe flow mixing or abrupt area change problems with the one-dimensional formulation. The momentum equation has not been used in the derivation of the global pressure equation anyway, so it does not justify the results obtained in the analysis of the downcomer. The previous study just tries to highlight that extreme care must be taken with flow mixing in order to achieve proper conservation of the thermodynamic variables. In the case of the momentum equation, the hypotheses to calculate the local pressure drop in a flow mixing or abrupt area change must be clearly specified. Current codes do not pay much attention to this issue, since the local pressure drops are used to tune the circuit flow.

It is time to explain now why the pressure analysis is not satisfactory when flow mixing is present. The RELAP5 manual does not refer anywhere to flow mixing. The discontinuities caused by merging flows with different thermodynamic properties at a sharp point, or by abrupt area changes, are not taken into account. When a control volume is attached to several inlet junctions, RELAP5 computes the internal energy convective term by adding the energy fluxes in each junction. Each of these fluxes has the form . The donnor cell quantities are referred to the upstream side of the junction. In order to avoid the discontinuities in the junctions, they should be referred to the downstream side. Of course, the conservation equations should be used to compute the downstream variables as a function of the upstream values. This treatment of the convective terms does not preserve the energy balance rigorously, what justifies the results obtained in the pressure analysis.

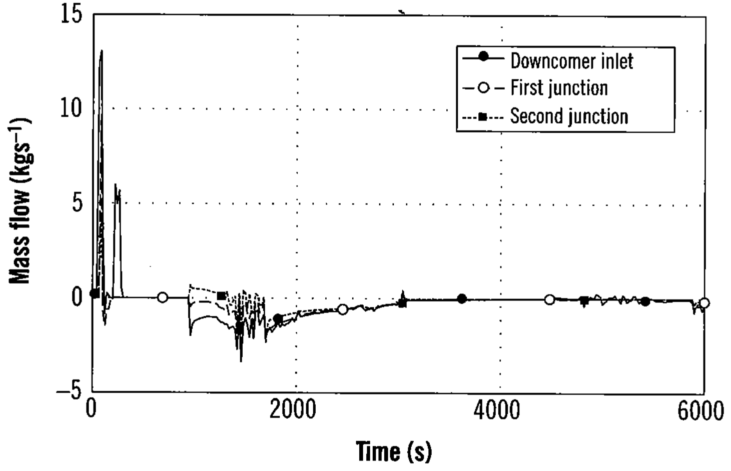

Figure 4.93. Vapor mass flows in the downcomer

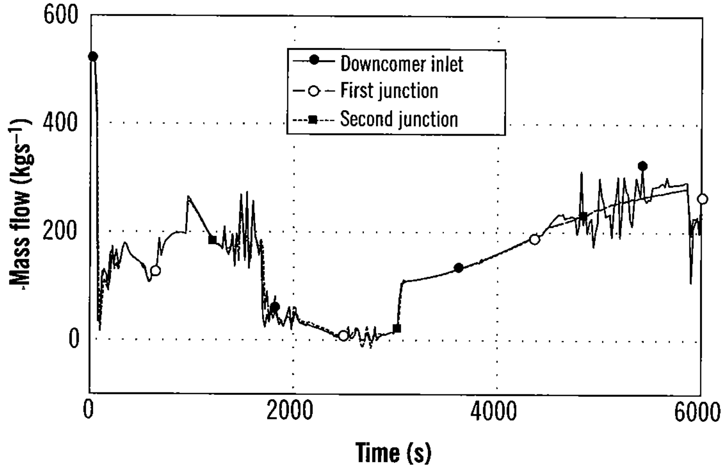

Figure 4.94. Liquid mass flows in the downcomer

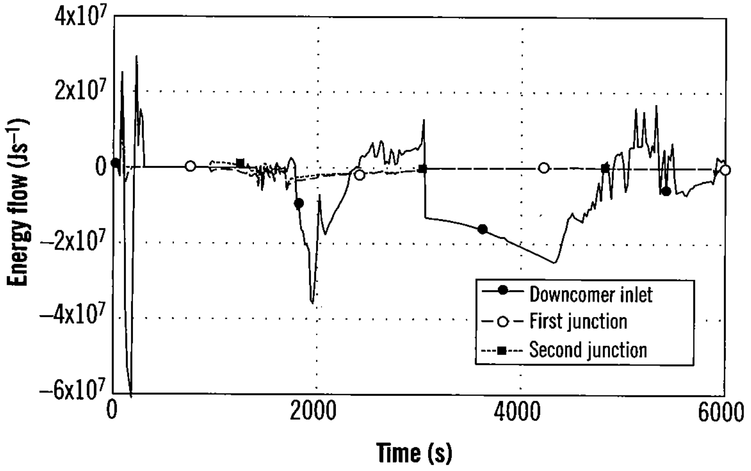

Figure 4.95. Vapor energy flows in the downcomer

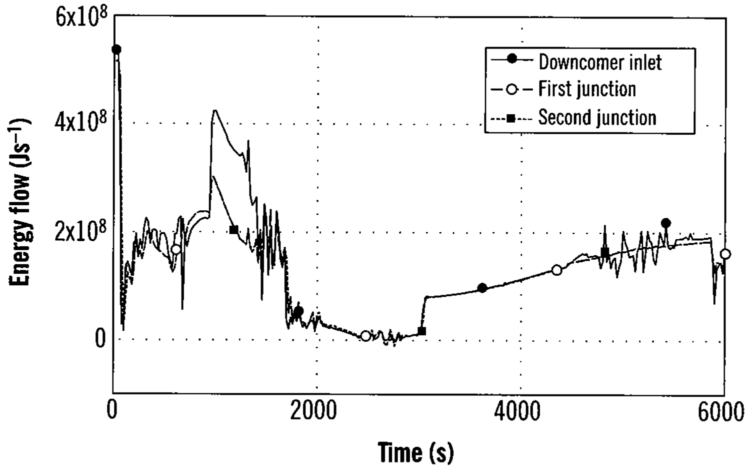

Figure 4.96. Liquid energy flows in the downcomer

V. Conclusions and recommendations for future developments

V.1. Conclusions

This report demonstrates the convenience of utilities for analysis and postprocess of computer codes. The utility has been particularly successful in the following areas:

- Correct interpretation of the results in terms of models more suitable to explain the evolution of certain variables, such as the global pressure (2.29) and the liquid volume (2.31) equations. These equations have been used in application example 4.3 to refutate a classical argument to explain the results obtained when the secondary circuit of a PWR is broken.
- Diagnosis of errors, which may be due to several causes. Example 4.2 shows how the analysis tool has been used to identify a user error in the input file. Case 4.4 illustrates how the tool can reveal a code misconception that affects not only RELAP5 but also TRAC-P [36] and TRAC-B [34]. RETRAN03 [6] developers seem to have solved that error successfully, since the input file only requires the velocity and the enthalpy of the fluid as inlet boundary conditions. Furthermore, the tools pointed out in example 4.5 the problems that may arise when mixing flows with different thermodynamic properties.
- Validation of alternative models. Those models may be more simplified, with the aim of speeding up the overall calculation or simplifiyng the resolution scheme, as demonstrated in example 4.3.4 with the non-equilibrium mass model derived in section II.5.

As a conclusion, the development of a new generation of computer codes with estimation of error upper bounds is highly recommended. This is the only way to scientifically ensure the quality of the results.

V.2. Recommendations for future developments

V.2.1. Analysis of other codes and models

Verification of the solutions computed by simulation codes by means of alternative balance equations is a subject open to research. Any other equation rigorously derived from the original equations can be added to enhance the tool. Balance equations derived in chapter 2 are just a sample of the large number of possibilities offered by this technique.

The heat structures and the power generation section seem to be natural candidates for the extension of the postprocessing ideas.

V.2.2. Identification of lower order models

If transient dynamics is very involved, a previous simplification of the models may be necessary. Section II.5.3 is a good example of this kind of simplifications. It shows how simpler models, with an uncoupled phase, can be derived from the non-equilibrium mass equations. Many other simplified models, using a number of equations between 3 and 6, are found in the history of this discipline [31]. Most of them can be demonstrated to be valid under some assumptions. Validation of those models in certain regions of operation can be performed with techniques similar to those shown in this work.

A typical application of simplified models is the Probabilistic Safety Assessment (PSA) of nuclear power plants. The multifarious sequences that can stem from an initiating event in a level 1 PSA demand the use of simplified, conservative models, in order to decrease the overall computation time [18].

However, some other applications require even more simplified thermalhydraulic dynamics, so it can be integrated with other type of phenomena. For instance, new phenomena arise in level 2 PSA, derived from the degraded core as a consequence of level 1 PSA. Those phenomena are very often of stochastic, barely known nature, subject to a high degree of uncertainty, resulting in a wide variety of possibilities for further evolution of the sequence. That kind of dynamics can hardly be incorporated into the risk calculations, and very specially to the delineation of the so called Accident Progression Event Trees (APET) [8, 5].

Techniques to be used in those cases consist mainly in model identification. A general mathematical form, dependent on some cefficients, is assumed for the dynamic evolutions. The coefficients are tuned to fit a sufficient number of transients precalculated with *best estimate* codes such as RELAP5 or MELCOR. This process can be termed as identification or training of the system. Once the system is trained, it is used to compute non precalculated transient evolutions. Choice of format as physically meaningful as possible is essential. At this point, Dynamic Systems Theory may be helpful.

For example, system dynamics can be regarded as comprising several time intervals, within each one the system can be linearized (piecewise linearization). It can be demonstrated that the behavior of any system can be approximated this way, with the possible exceptions of chaotic systems or presence of discontinuities caused by explosive phenomena. The main drawback of this approach is that the order of the linearized system may eventually be too high and the range of validation very narrow. The main advantage is that the previous history of the system is collected through the state vector. Some integral representations may obviate the high order problem, at the expense of dealing with the previous history explicitly. Some examples can be found in reference [30].

Rigorous inclusion of those generic PSA models can be done with the help of the latest trends in dynamic reliability [25, 26, 24].

Model identification, in the terms explained above, can be another application of the postprocessing techniques shown in this work.

V.2.3. Development of codes with error control

The postprocessing tools has proven to be more effective in the interpretation of results and identification of programming errors than in the diagnosis of errors due to an inappropriate nodalization or time step choice. Identification of such a type of errors would imply a comparison of the discrete solution provided by the code with an hypothetical analytical solution. The latter is of course unknown. Verification of the discrete solution by replacement into a discretized set of equations seems to be close to a tautology. It would mean that the solution is checked versus the same premises with wich it has been obtained. It is not surprising that the solution fulfills the balance equations in every node, although very probably the global analytical equations would not be satisfied.

The discretization, which depends strongly on the code user, is still one of the main contributors to the overall code uncertainty. Moreover, discretization errors can make the *verification* and *validation* processes merge inappropriately. Figures 6.1 and 6.2 illustrate this statement.

Verification is the process which ensures that the discrete solution approaches the true analytical solution within a certain tolerance margin. Verification implies matching discrete with continuous mathematical forms. It must be noted that the verification process does not have to do with the capability of the analytical models to represent the real world. Verification of the solution is attempted by very few codes.

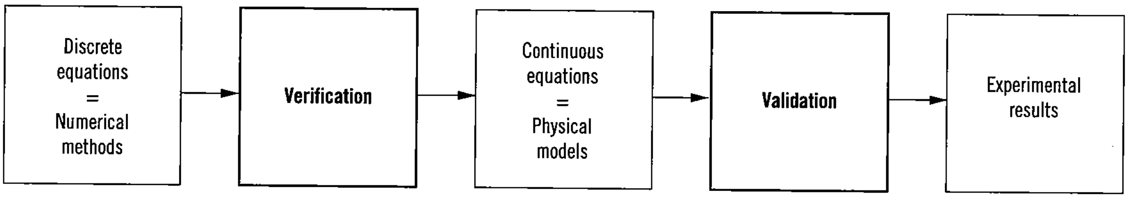

Figure 5.1. Verification and validation processes

The *validation* process can be launched only after a proper verification of the solution. It implies a checking of the capability of the models included in the code to reproduce a set of experimental data. The process consists in matching continuous domain maths with the physical reality. Moreover, the term validation makes reference to a certain experiment for set of experiments. It is not an absolute term as it is often understood. Figure 5.1 emphasizes that the verification and validation processes must be clearly separated from each other.

Since the verification process is seldom undertaken, it is usually merged with validation as shown in figure 5.2. This means that the solutions computed by the code are directly compared with the experimental data, which is a risky although common practice. If the comparison between code results and experimental data is not satisfactory, a good model could be rejected even if the cause of discrepancies lies in an inadequate nodalization. On the contrary, the analyst may try to improve the solution by refining the noding mesh, although the analytical model might be the cause of the insatisfactory results. It is difficult to accurately determine the source of errors if the verification and validation processes are merged.

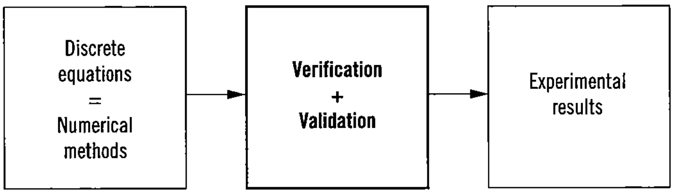

Figure 5.2. Collapsed verification and validation processes

The verification task can be undertaken by means of the so called precise numerical methods [2, 1]. These methods take into account the intrinsic discrete nature of both the numerical methods and the computing machine.

The problem to be solved no longer consists in solving the stages of the numerical method in the user defined computational mesh. On the contrary, the user must specify the discrete points at which the solution is to be observed, as well as the required precision. Note that the observation mesh may not be coincident with the computational mesh necessary to find a solution within the desired tolerance margin. Moreover, the latter depends strongly on the former. The precision of the solution can be known only if an upper bound of the truncation error is estimated. If the error associated to any of the output variables under observation is greater than the desired precision, the transient must be recomputed with a finer spation-temporal nodalization. Although current codes usually consider just one kind of computational mesh, The following types must be discriminated:

- The spatial mesh used to define certain input parameters. For instance, the cross-sectional areas of the pipes. The greater the spatial variations of those parameters are, the finer this mesh must be.

- The spatial mesh used to define the initial conditions at discrete points, which must be finer as long as the initial conditions contain high frequency spatial components.
- The temporal mesh used to define the boundary conditions. The higher the time frequency of the boundary conditions is, the finer the mesh must be.
- The spatio-temporal mesh formed by the points at which the user desires to observe certain output variables.
- The calculational mesh used to solve the numerical method.

For most current codes all the types of discretization described above are the same, even though each of them accounts for very different aspects. If the error associated to any of the desired output variables is larger than the required precision a new calculation must be attempted with a finer spatio-temporal nodalization. If the numerical method is proven to be both consistent and convergent, the new noding scheme should yield smaller truncation errors. The proposed approach needs of course a certain method for automatic mesh generation. The numerical discretization recovers then its strict algorithmical meaning. It does not depend on the user's choice but on the desired output variables and the required precision. The plant nodalization is no longer a contributor to the so called *user efect*. This is restricted just to the choice of optional models.

The estimation of the truncation errors associated to a certain numerical method usually requires the use of higher order derivatives of mathematical functions. An analytical expression of those derivatives is usually unknown when the code is programmed. Derivation of the analytical expressions demands the use of symbolic calculus, which adds an extra degree of complexity into the overall problesm.

Besides the truncation error associated to the numerical methods, round-off errors must be controlled as well. Digital computers, being discrete machines, can represent only a subset of the rational numbers and none of the irrational numbers with their native floating-point arithmetic. Although some authors such as [42] state that round-off errors are negligible as compared to truncation errors, several evidences on the contrary have been found. Another evidence of the importance of round-off errors is provided in [28], which shows that different results can be obtained, even with the same machine, when the thermal conductivity of water in presence of non-condensible gases is calculated by linear interpolation based either on the leftmost point,

$$y(x) = y_0 + \frac{y(x_1) - y(x_0)}{x_1 - x_0}(x - x_0)$$

or the rightmost point,

$$y(x) = y_1 + \frac{y(x_1) - y(x_0)}{x_1 - x_0}(x - x_1)$$

Both formulations are completely equivalent. Any discrepancy in the results can only be explained by the different effects of each formulation in the round-off errors.

The *precise numerical methods* focus on the control of any source of error. They require the use of a special type of floating-point arithmetic called *range arithmetic.* Real numbers are represented by data structures such as `struct` in C language or `TYPE` in FORTRAN 90. Every digit in the mantissa is represented by a native integer type. Another native integer data type accounts for the exponent. The number of digits in the mantissa is not fixed, as it is in ordinary native floating-point arithmetics. An extra integer digit represents an upper bound of the error associated to the floating-point number. For instance, the number 1.5984336~2E-2 means that its value lies in the interval $1.5984336\text{x}10^{-2}\pm2\text{x}10^{-9}$.

Whenever an operation between numbers is performed, the round-off error is accounted for either incrementing the error digit or, if the latter is greater than 9, decrementing the number of digits in the mantissa. It is clear than the greater the number of operations used to compute a number, the smaller the precision is. If the error associated to any of the user required variables is greater than the maximum allowed, the transient must be computed from the beginning, starting with a larger initial precision. The number of digits in the mantissa can be changed by using the dynamic memory allocation capabilities provided by advanced programming languages. With range arithmetic the precision is no longer limited by any machine word length (usually 64 bits for double precision) but by the memory available in the computer. Note that, since the precision is progressively decreased during the course of the transient, a very large initial precision might be required to obtain an output precision of just a few digits.

The use of precise numerical methods yields the following cascade effects:

- Since range arithmetic is implemented through integer data types, and the result of operations between integer numbers is always exact, the results are machine independent. The so called *machine-compiler* effect dissapears.
- Since the working precision is unlimited (although it is always finite), the round-off errors can be as small as desired.
- As a consequence of the item above, nodalizations can be refined as much as necessary to obtain the required precision in the output variables. The refinement implies a greater number of operations and thus greater round-off errors, but they can be compensated by a larger initial precision. The nodalization needed to obtain the desired final precision is automatically chosen by the code, not by the user. The nodalization is not then a contributor to the *user effect.*

Bibliography

Bibliography

[1] O. Aberth. *Precise Numerical Methods Using C++*. Academic Press, 1998.

[2] O. Aberth and M. J. Schaefer. Precise computation using range arithmetic, via C++. *ACM Transactions on Mathematical Software,* 18(4):481–491, December 1992.

[3] P. Abramson. *Guidebook to Light Water Reactor Safety Analysis.* Hemisphere Publishing Corporation.

[4] R. Ashley, M. El-Shanawany, F. Eltawila, and F. D'Auria. Good practices for user effect reduction. Status report. Working draft 2. Technical report OECD/NEA/CSNI/PWG2 TG on THA, OECD/NEA, April 1998.

[5] A. S. Benjamin, V. Behr, D. M. Kunsman, and W. B. Murfin. Containment event analysis for postulated severe accidents: Surry power station, Unit 1. Technical report NUREG/CR-4700, SAND 86-1135, US Nuclear Regulatory Commission, February 1987.

[6] Computer Simulation Flow Systems. RETRAN-03: A program for tran-sient thermal-hydraulic analysis of complex uid ow systems. volume 1: Theory and numerics. Technical report NP-7450, Volume 1. Research Project 889-10, Electric Power Research Institute, May 1992.

[7] F. D'Auria, E. Cojnack, H. Glaeser, C. Lage, and T. Wickett. Overview of uncertainty issues and methodologies. In *OECD/CSNI Seminar on 'Best Estimate Methods in Thermal-Hydraulics Safety Analysis', Ankara.* June 1998.

[8] J. Evrard, J. Mireau, P. Lardjane, and B. Lucas. KANT: a probabilistic software for level 2 PSA. In *Proceedings of the Fourth International Conference on Probabilistic Safety Assessment and Management, New York.* September 1998.

[9] G. E. Fagg, J. J. Dongarra, and A. Geist. Heterogeneous MPI application interoperation and process management under PVMPI. In *Euro PVM-MPI conference, Cracow, Poland.* November 1997. `http://www.netlib.org/utk/papers/pvmmpi97.ps.`

[10] L. R. Feinauer. Compiler issues associated with safety-related software. *Nuclear Technology,* 93:116–122, January 1991.

[11] A. Geist et al. *PVM, Parallel Virtual Machine. A User's Guide and Tutorial for Networked Parallel Computing.* The MIT press; Scientific and Engineering Computation Series, 1994. `ftp://netlib2.cs.utk.edu/pvm3/book/pvm-book.ps.`

[12] A. Geist et al. *PVM3 Users Guide and Reference Manual.* Oak Ridge National Laboratory, September 1994. `ftp://netlib2.cs.utk.edu/pvm3/ug.ps.`

[13] B. Gropp, R. Lusk, and A. Skjellum. *Using MPI.* MIT press, 1994.

[14] R. Herrero. Specification of initial and boundary conditions in RELAP5/ Mod3.2. In 9th CAMP Fall Meeting. Santa Fé. October 1996.

[15] R. Herrero. A standardized methodology for the linkage of computer codes. Application to RELAP5/Mod3.2. Technical report NUREG/IA-0179, US Nuclear Regulatory Commission. Office of Nuclear Regulatory Research, March 2000.

[16] R. Herrero and E. Meléndez. Linkage and parallelisation of scientific programs with a modular simulation language. Submitted to Transactions of the Society for Computer Simulation.

[17] E. Hughes. An evaluation of state-of-the-art two-velocity two-phase flow models and their applicability to nuclear reactor transient analysis. Technical report NP-143, EPRI, 1976.

[18] J. Izquierdo, C. Queral, R. Herrero, J. Hortal, M. Sánchez, and R. Muñoz. Role of fast running codes and their coupling with PSA tools. In *OECD-CSNI Workshop on 'Advanced Thermal-Hydraulic and Neutronic Codes: Current and Future Applications'. Barcelona.* April 2000.

[19] J. M. Izquierdo, J. Hortal, and L. Vanhoenacker. Merits and limits of thermalhydraulic plant simulations - towards a unified approach to qualify plant models. *Nuclear Engineering and Design,* 145:175–205, 1993.

[20] J. M. Izquierdo and M. Sánchez Perea. CAMPEADOR: An implicit simulation language for continuous sytems combined with discrete events able to apply protection theory. In *1994 European Simulation Multiconference (ESM'94). Barcelona, Spain.* June 1994.

[21] C. Johnson, R. Rannacher, and M. Boman. Numerics and hydrodynamic stability: Toward error control in computational uid dynamics. *SIAM J. Numer. Anal.,* 32(4):1058–1079, August 1995.

[22] K. Jones. XMGR5 extensions. Idaho National Engineering Laboratory, 1995.

[23] M. Kowan and J. Staudenmaier. Draft Standard Review Plan Section 15.0.2. Review of transient and accident analysis methods. Technical report NUREG-0800, United States Nuclear Regulatory Commission, January 2003.

[24] P. E. Labeau, J. M. Izquierdo, and E. Meléndez. Stimulus-driven branchings in continuous event trees. A transport model for the estimation of the rupture probability of a single-compartment containment under H_2 laminar de agration. 2002. Publication pending.

[25] P. E. Labeau, J. M. Izquierdo, and E. Meléndez. Stimulus-driven branchings in continuous event trees. I. Extending the theory of probabilistic dynamics. 2002. Publication pending.

[26] P. E. Labeau, J. M. Izquierdo, and E. Meléndez. Stimulus-driven branchings in continuous event trees. II. A cell-to-cell transport theory approach. 2002. Publication pending.

[27] A. López and J. M. Sierra. Contraste de resultados obtenidos con RELAP5/ Mod3.2 en diferentes entornos de computación. In *First Technical Session CAMP Spain. Barcelona.* April 1997.

[28] G. A. Mortensen. One root cause of numerical problems that occur in large codes. In *Fall CAMP Meeting.* October 1999.

[29] W. L. Oberkampf and F. G. Blotner. Issues in Computational Fluid Dynamics. Code verification and validation. Technical report SAND95-1352, Sandia National Laboratory, 1997.

[30] M. S. Perea. *Análisis y simulación de redes de subsistemas lineales mediante desarrollos en serie de funciones de Laguèrre.* PhD Thesis, Universidad Pontificia de Comillas. Escuela Técnica Superior de Ingenieros Industriales (ICAI). Dpto. de Electrónica y Automática, 1996.

[31] C. Queral. *Algoritmo para la resolución de las ecuaciones de conservación de uidos bifásicos. Aplicación a centrales nucleares.* PhD Thesis, Universidad Nacional de Educación a Distancia. Facultad de Ciencias. Dpto. de Física Fundamental, 1995.

[32] V. Ransom. Faucet flow, oscillating manometer. In G. Hewitt, J. Delhaye, and N. Zuber, editors, *Multiphase Science and Technology,* pages 465- 470. Hemisphere Publishing Corporation, 1987.

[33] V. H. Ransom. A numerical model of two-phase ows. In *Ecole d'Ete d'Analyse Numerique.* May 1989. EGG-EAST-8546.

[34] W. H. Rettig and N. L. Wade. TRAC-BF1/MOD1: An advanced best-estimate computer program for BWR accident analysis. User's guide. Technical report NUREG/CR-4356. EGG-2626. Vol. 2 R4, Idaho National Laboratory, June 1992.

[35] P. Roache. Quantification of uncertainty in computational uid dynamics. *Annual Review of Fluid Mechanics,* 29:123–160, 1997.

[36] J. W. e. a. Spore. TRAC- PF1/MOD2. Volume I . Theory manual. Technical report LA-12031-M, Vol. I. NUREG/CR-5673, Los Alamos National Laboratory, 1993.

[37] The RELAP5 Code Development Team. RELAP5/MOD3 code manual, volume I: Code structure, system models and solution methods. Technical report NUREG/CR-5535, INEL-95/0174, Idaho National Engineering Laboratory, June 1995.

[38] The RELAP5 Code Development Team. RELAP5/MOD3 code manual, volume IV: Models and correlations. Technical report NUREG/CR-5535, INEL-95/0174, Idaho National Engineering Laboratory, June 1995.

[39] K. Trambauer. Computer and compiler effects on code results. Technical report (97)6, OECD/CSNI, 1997.

[40] P. J. Turner. *ACE/gr Users Manual. Graphics for exploratory data analysis. Software Documentation Series, SDS3, 91-3.* Center for Coastal and Land-Margin Research. Oregon Graduate Institute of Science and Technology, 19600 NW von Neumann Dr., Beaverton, Oregon, 97006-1999, 1993.

[41] D. P. Weidong Wang, Glen Mortensen. Energy conservation in RELAP5. Technical report SCIE-NRC-392-99, SCIENTECH, Inc., November 1999.

[42] W. Wulff. Computational methods for multiphase ow. In *Second International Workshop on Two-Phase Flow Fundamentals. Rensselaer Polytechnic Institute.* March 1987.

[43] W. Wulff, H. Cheng, S. Lekach, and A. Mallen. The BWR plant analyzer. Technical report NUREG/CR-3943. BNL-NUREG-51812, Brookhaven National Laboratory, August 1984.

Appendices

Appendix A. Nomenclature

Appendix B. Partial derivatives of thermodynamic properties

Appendix A. Nomenclature

Latin letters

t	time (s).
x	spatial variable (m).
v	fluid velocity (ms^{-1}).
v_i	interphase velocity (ms^{-1}).
A	cross-sectional area (m^2).
V	volume (m^3).
m	mass (Kg).
W	mass flow ($Kg\ s^{-1}$).
P	pressure (Pa).
B_x	projection of gravity onto the pipe direction ($m\ s^{-2}$).
FW	wall friction coefficient (s^{-1}).
FI	interphase friction coefficient (s^{-1}).
C	virtual mass coefficient (non-dimensional). compressibility (m^3).
K	local pressure drop coefficient (non-dimensional).
HLOSS	form loss coefficient (ms^{-1}). HLOSS = $1/2K\|v\|$
u	specific internal energy ($J\ Kg^{-1}$).
H	heat transmission coefficient per unit volumen ($Js^{-1}m^{-3}K^{-1}$), control volume height (m).
Q	heat power per unit volume ($Js^{-1}m^{-3}$).
h	specific enthalpy (JKg^{-1}).
T	temperature (K).
s	specific entropy ($JKg^{-1}K^{-1}$).
DISS	dissipation power per unit volume ($Js^{-1}m^{-3}$).
L	length of a one-dimensional TH system (m).
N	number of control volumes in a TH system.
M	number of inlet or outlet junctions in a TH system.
J	difference between inlet and outlet volumetric flows (m^3s^{-1}).
Sw	expansion/contraction due to appearance/disappearance of phase (m^3s^{-1}).
D	dilation of phase, caused by energy interchanges (m^3s^{-1}).
Ac	term dominated by pressure gradients (m^3s^{-1}).
C_P	constant pressure specific heat capacity ($JKg^{-1}K^{-1}$).
C_ϑ	constant volume specific heat capacity ($JKg^{-1}K^{-1}$).
g	gravity acceleration (ms^{-2}).

Greek letters

α	fraction of phase (non-dimensional).
ρ	density (Kgm^{-3}).
Γ	phase generation rate per unit volume (Kgm^{-3}s^{-1}).
$\delta(x-x_0)$	Dirac's delta function, centered around the point x_0 (m^{-1}).
ϑ	specific volume (m^{-3}Kg^{-1}).
π	thermal expansion coefficient (m^{-3}J^{-1}).
γ	isentropic compressibility coefficient(non-dimensional).
λ	compressibility coefficient along the saturation line (non-dimensional).
β	isobaric thermal expansion coefficient (K^{-1}).
μ	$\left(\dfrac{\partial\rho}{\partial P}\right)_u$ (m^{-2}s^{2}).
θ	angle between a control volume and the vertical direction (non-dimensional).

Subindices

k	phase (liquid or vapor).
k'	phase opposite to phase k.
i	relative to the interphase.
w	relative to the wall.
b	relative to the fluid bulk.
m	relative to the mixture.
g	relative to vapor.
f	relative to liquid.

Superindices

in	relative to the inlet face.
out	relative to the outlet face.
nq	non- equilibrium.
eq	equilibrium.
n	time step.
sat	relative to the saturation state.

RELAP5 output variables: control volumes

`csubpg`	C_{Pg} (JKg^{-1}K^{-1}).
`csubpf`	C_{Pf} (JKg^{-1}K^{-1}).
`betagg`	β_g (K^{-1}).
`betaff`	β_f (K^{-1}).
`drgdp`	μ_g (m^{-2}s^{2}).

drfdp	μ_f(m^{-2}s^{2}).
vvol	volume (m^{3}).
avol	cross-sectional area (m^{2}).
p	control volume pressure (Pa).
voidg	vapor fraction (non-dimensional).
voidf	liquid fraction (non-dimensional).
rhog	vapor density (Kgm^{-3}).
rhof	liquid density (Kgm^{-3}).
ug	vapor specific internal energy (JKg^{-1}).
uf	liquid specific internal energy (JKg^{-1}).
gammai	vapor generation rate per unit volume in the fluid bulk (Kgm^{-3}s^{-1}).
gammaw	vapor generation rate per unit volume in the wall (Kgm^{-3}s^{-1}).
velg	vapor velocity (ms^{-1}).
velf	liquid velocity (ms^{-1}).
q	total heat power transferred to the fluid through the wall (Js^{-1}).
qwg	heat power transferred to vapor through the wall (Js^{-1}).
hig	vapor heat transfer coefficient per unit volume (Js^{-1}m^{-3}K^{-1}).
hif	liquid heat transfer coefficient per unit volume (Js^{-1}m^{-3}K^{-1}).
satt	saturation temperature at total pressure, as an output variable (K).
tsatt	saturation temperature at total pressure, as a boundary condition (K).
quale	equilibrium qualtiy (non-dimensional).
boron	boron conentration (parts of boron per parts of water).
tempg	vapor temperature (K).
tempf	liquid temperature (K).
fwalgx	wall friction coefficient for vapor phase (Kgm^{-3}s^{-1}).
fwalfx	wall friction coefficient for liquid phase (Kgm^{-3}s^{-1}).

RELAP5 output variables: junctions

velgj	vapor velocity (ms^{-1}).
velfj	liquid velocity (ms^{-1}).
mflowgj	vapor mass flow, as a boundary condition (Kgs^{-1}).
mflowfj	liquid mass flow, as a boundary condition (Kgs^{-1}).
voidgj	vapor fraction (non-dimensional).
voidfj	liquid fraction (non-dimensional).
rhogj	vapor density (Kgm^{-3}).
rhofj	liquid density (Kgm^{-3}).
formgj	local pressure drop coefficient for vapor phase (non-dimensional).
formfj	local pressure drop coefficient for liquid phase (non-dimensional).

Appendix B. Partial derivatives of thermodynamic properties

Partial derivatives as a function of C_P, C_ϑ y β

$$\left(\frac{\partial u}{\partial T}\right)_P = C_P - P\vartheta\beta$$

$$\left(\frac{\partial u}{\partial \vartheta}\right)_P = \frac{C_P - P\vartheta\beta}{\vartheta\beta}$$

$$\left(\frac{\partial T}{\partial s}\right)_\vartheta = \frac{T}{C_\vartheta}$$

$$\left(\frac{\partial T}{\partial s}\right)_P = \frac{T}{C_P}$$

$$\left(\frac{\partial \vartheta}{\partial s}\right)_P = \frac{T\vartheta\beta}{C_P}$$

$$\left(\frac{\partial T}{\partial P}\right)_s = \frac{T\vartheta\beta}{C_P}$$

$$\left(\frac{\partial u}{\partial s}\right)_P = \left(C_P - P\vartheta\beta\right)\frac{T}{C_P}$$

$$\left(\frac{\partial s}{\partial P}\right)_T = -\vartheta\beta$$

$$\left(\frac{\partial P}{\partial T}\right)_\vartheta = \frac{C_P - C_\vartheta}{T\vartheta\beta}$$

$$\left(\frac{\partial P}{\partial \vartheta}\right)_T = \frac{C_\vartheta - C_P}{T\vartheta^2\beta^2}$$

$$\left(\frac{\partial \vartheta}{\partial P}\right)_s = \frac{C_\vartheta T\vartheta^2\beta^2}{C_P\left(C_\vartheta - C_P\right)}$$

$$\left(\frac{\partial u}{\partial \vartheta}\right)_T = \frac{C_P - C_\vartheta - P\vartheta\beta}{\vartheta\beta}$$

$$\left(\frac{\partial \vartheta}{\partial h}\right)_P = \frac{\vartheta\beta}{C_P}$$

$$\frac{1}{\gamma} \equiv \left(\frac{\partial \ln\rho}{\partial \ln P}\right)_s = \frac{PC_\vartheta T\vartheta\beta^2}{C_P\left(C_P - C_\vartheta\right)}$$

$$\frac{1}{\lambda} \equiv \left(\frac{d\ln\rho}{d\ln P}\right)_{\text{sat}} = -PT\beta\left[\frac{\vartheta\beta}{C_\vartheta - C_P} + \frac{\vartheta_{fg}}{h_{fg}}\right]$$

Partial derivatives as a function of μ, C_P y β

$$\left(\frac{\partial \rho}{\partial u}\right)_P = \frac{\rho^2\beta}{P\beta - \rho C_P}$$

$$\left(\frac{\partial u}{\partial P}\right)_\rho = \eta\frac{\rho C_P - P\beta}{\rho^2\beta}$$

$$\left(\frac{\partial P}{\partial s}\right)_\vartheta = \frac{T\beta\rho^2}{\eta\left(\rho C_P - P\beta\right)}$$

$$\left(\frac{\partial \vartheta}{\partial P}\right)_s = \frac{\eta\left(P\beta - \rho C_P\right)}{C_P\rho^3}$$

$$\left(\frac{\partial T}{\partial s}\right)_\vartheta = \frac{T}{C_P}\left(1 - \frac{P\beta^2 T}{\eta\left(P\beta - \rho C_P\right)}\right)$$

$$\left(\frac{\partial \vartheta}{\partial s}\right)_u = \frac{T}{P}$$

$$\left(\frac{\partial T}{\partial \vartheta}\right)_s = \frac{T\beta\rho^2}{\eta\left(P\beta - \rho C_P\right)}$$

$$\left(\frac{\partial \vartheta}{\partial s}\right)_T = \frac{\eta\left(P\beta - \rho C_P\right) - \rho\beta^2 T}{\rho\beta^2 C_P}$$

$$C_\vartheta = \frac{\eta C_P\left(P\beta - \rho C_P\right)}{\eta\left(P\beta - \rho C_P\right) - \rho\beta^2 T}$$