



KIT SCIENTIFIC REPORTS 7728

# Neutronics Methods for Transient and Safety Analysis of Fast Reactors

Marco Marchetti



Marco Marchetti

**Neutronics Methods for Transient and Safety Analysis of Fast Reactors**

Karlsruhe Institute of Technology  
**KIT SCIENTIFIC REPORTS 7728**

# Neutronics Methods for Transient and Safety Analysis of Fast Reactors

by  
Marco Marchetti

Report-Nr. KIT-SR 7728

Dissertation, Karlsruher Institut für Technologie (KIT)  
Fakultät für Maschinenbau

Tag der mündlichen Prüfung: 14. Juli 2016

Referent: Prof.Dr.-Ing. Thomas Schulenberg

Korreferenten: Prof.Dr.-Ing. Piero Ravetto, Prof.Dr. Robert Stieglitz

#### Impressum



Karlsruher Institut für Technologie (KIT)  
KIT Scientific Publishing  
Straße am Forum 2  
D-76131 Karlsruhe

KIT Scientific Publishing is a registered trademark of Karlsruhe  
Institute of Technology. Reprint using the book cover is not allowed.

[www.ksp.kit.edu](http://www.ksp.kit.edu)



*This document – excluding the cover, pictures and graphs – is licensed  
under the Creative Commons Attribution-Share Alike 3.0 DE License  
(CC BY-SA 3.0 DE): <http://creativecommons.org/licenses/by-sa/3.0/de/>*



*The cover page is licensed under the Creative Commons  
Attribution-No Derivatives 3.0 DE License (CC BY-ND 3.0 DE):  
<http://creativecommons.org/licenses/by-nd/3.0/de/>*

Print on Demand 2017 – Gedruckt auf FSC-zertifiziertem Papier

ISSN 1869-9669

ISBN 978-3-7315-0611-9

DOI: 10.5445/KSP/1000063691







# Neutronics Methods for Transient and Safety Analysis of Fast Reactors

Zur Erlangung des akademischen Grades  
Doktor der Ingenieurwissenschaften

der Fakultät für Maschinenbau  
Karlsruher Institut für Technologie(KIT)

genehmigte

## Dissertation

von

Dottore Magistrale in Ingegneria Marco Marchetti

Tag der mündlichen Prüfung: 14. Juli 2016

Hauptreferent: Prof.Dr.-Ing. Thomas Schulenberg  
Karlsruher Institut für Technologie(KIT)

Korreferent: Prof.Dr.-Ing. Piero Ravetto  
Politecnico di Torino

Korreferent: Prof.Dr. Robert Stieglitz  
Karlsruher Institut für Technologie(KIT)



# Abstract

Modeling the evolution of possible or postulated accidents in nuclear reactors is fundamental in designing safe systems. For the next generation of reactors, in particular fast reactors, fuel movement during an accident can, in principle, drive an energetic event. Such is the issue of recriticality. The thermal energy produced during these events will, possibly, be converted into mechanical energy by some mechanisms.

For example, the nuclear heat deposited in the fuel could cause fuel vaporization and its subsequent expansion. This movement would accelerate the surrounding sodium: part of the initial energy in the fuel is thus converted into sodium kinetic energy. This mechanical energy will finally be absorbed, in some way or another, by the reactor vessel. Providing an accurate estimate for the maximum mechanical work that any accidental sequence can do onto the reactor vessel is an essential step in designing a reactor containment that would withstand any load generated by any accident. That would assure accident containment, without consequences for the general public.

Fast reactor accident modeling is a complicated task. The outcome of an accident is determined by different physical phenomena, all acting at almost the same time. Safety analysts must track all these different phenomena. Multi-physics codes have been developed for this task. They must contain accurate models for fluid-dynamics, neutronics, and structures. This work has to do with neutronics modeling of such accidents. Past and recent analyses have been limited to the approximate description of the neutronic field, for example by using a rough description of the energy and/or of the angular dependence of the neutron flux. In this work, different neutronic solvers are selected and coupled into a general multi-physics code for fast reactor accident analysis. Performances of each of them is then assessed. Some emphasis has been put also in assessing the speed of these solvers for determining the neutron flux. Recently, there has been a tendency to abandon two-dimensional reactor models in favor of full three-dimensional ones. Only the latter models are adequate to represent phenomena that are inherently three-dimensional. Because of the computational costs attached, three-dimensional accident simulation would become practical only if all the different modules applied for the safety analysis, and therefore the neutronics module as well, are sufficiently fast.

In this work, fast neutron solvers are found to reduce the total computational time even by 60%, paving thus the way to three-dimensional modeling, and to a more accurate description of the neutron field and its influence on the results of safety analyses.



# Zusammenfassung

Die Modellierung des zeitlichen Verlaufs möglicher Störfälle oder postulierter Unfälle von Kernreaktoren gehört zu den wichtigsten Aufgaben beim Entwurf sicherer Anlagen. Für die nächste Generation künftiger Reaktoren, insbesondere Schneller Reaktoren, kann die Bewegung von Brennstoff im Prinzip zu einem energiereichen Schadensfall führen.

Ein derartiges Ereignis kann Ergebnis einer Rekritikalität sein. Die dabei freigesetzte thermische Energie kann durch bestimmte Vorgänge zumindest teilweise in mechanische Energie umgewandelt werden. Beispielsweise kann die bei der Kernspaltung im Brennstoff erzeugte Wärme eine Verdampfung und dadurch eine anschließende Expansion des Brennstoffs verursachen. Diese Bewegung kann das umgebende Kühlmittel, Natrium, beschleunigen: auf diese Weise wird ein Teil der im Brennstoff enthaltenen thermischen Energie in kinetische Energie des Natriums konvertiert werden. Diese mechanische Energie wird schließlich auf irgendeine Art vom Reaktordruckbehälter absorbiert.

Die zuverlässige Abschätzung dieser maximalen mechanischen Energie, die durch irgendeinen Unfallablauf auf den Reaktordruckbehälter übertragen werden kann, ist ein entscheidender Schritt beim Entwurf eines derartigen Behälters, der der Belastung standhalten kann, die durch irgendeinen Unfall ausgelöst wird. Durch diese Begrenzung der Unfallfolgen auf das Innere des Behälters kann eine Gefährdung der Bevölkerung vermieden werden. Die Modellierung von Unfällen in Schnellen Reaktoren ist eine schwierige Aufgabe. Die Unfallfolgen werden durch verschiedenartige physikalische Phänomene beeinflusst, die nahezu zur gleichen Zeit ablaufen.

Sicherheitsanalytiker müssen diese unterschiedlichen Phänomene genau verfolgen. Dazu wurden sog. Multi-physics-Codes entwickelt, die zuverlässige Modelle für die Thermohydraulik, Neutronik sowie das Strukturmaterialverhalten enthalten. Die vorliegende Arbeit befasst sich mit der neutronischen Modellierung von Unfallabläufen. Bisherige Analysen waren beeinträchtigt durch die näherungsweise Bestimmung der Neutronenverteilung, z.B. hinsichtlich der Energie- und Winkel-Verteilung des Neutronenflusses.

In dieser Arbeit werden verschiedene Lösungsverfahren benutzt und mit einem allgemeinen Multi-physics-Code gekoppelt und ihr Leistungsvermögen festgestellt. Besondere Aufmerksamkeit wird der Bestimmung der Rechengeschwindigkeit zur Ermittlung des Neutronenflusses gewidmet. In letzter Zeit wird oft auf 2-dimensionale Reaktormodelle verzichtet zugunsten 3-dimensionaler Modelle, die eine zutreffende Modellierung der von Natur aus 3-dimensionalen Phänomene ermöglichen. Wegen der zugehörigen Rechenkosten können 3-dimensionale Unfallsimulationen nur dann durchgeführt werden, wenn alle Module, die in der Sicherheitsanalyse verwendet werden, also insbesondere auch das Neutronik-Modul, genügend schnell sind. In dieser Arbeit wurde festgestellt, dass schnelle Neutronik-

Lösungsverfahren die Gesamtrechenzeit sogar um 60% verringern können und damit eine 3-dimensionale Modellierung und eine genauere Bestimmung der Neutronenverteilung und deren Einfluss auf die Ergebnisse von Sicherheitsanalysen ermöglichen.

# Contents

<b>Abstract</b> . . . . .	<b>i</b>
<b>Zusammenfassung</b> . . . . .	<b>iii</b>
<b>1 Introduction</b> . . . . .	<b>1</b>
1.1 Fast Reactors . . . . .	2
1.2 Fast Reactors Plant System . . . . .	3
1.2.1 The Core . . . . .	4
1.2.2 Intermediate Heat Exchanger . . . . .	5
1.2.3 Steam Generator . . . . .	5
1.3 Fast Reactor Safety Considerations . . . . .	5
1.4 Overview of Hypothetical Core Disruptive Accidents . . . . .	6
1.4.1 Accident Initiators . . . . .	6
1.4.2 Transient Overpower Accidents . . . . .	7
1.4.3 Transient Undercooling Accidents . . . . .	9
1.5 Objective of the Research . . . . .	10
<b>2 Accidental Phases</b> . . . . .	<b>13</b>
2.1 Initiation Phase . . . . .	14
2.2 Transition Phase . . . . .	15
2.3 Core Disassembly Phase . . . . .	17
2.4 Expansion Phase . . . . .	18
2.5 Historical Background . . . . .	19
<b>3 Simulation Tools</b> . . . . .	<b>25</b>
3.1 Transition Phase Modeling: SIMMER Codes . . . . .	25
3.1.1 General Description . . . . .	25
3.1.2 SIMMER Neutronic Module . . . . .	29
3.1.3 Once-Through SIMMER Analysis . . . . .	36
3.2 PARTISN . . . . .	37
3.2.1 Iteration Strategy and Diffusion Synthetic Acceleration . . . . .	37
3.3 VARIANT: A Nodal Neutron Transport Solver . . . . .	39
3.3.1 The Variational Nodal Method . . . . .	40
3.4 KIN3D: A Kinetics Module for VARIANT . . . . .	45
3.4.1 Treatment of Flux Time Derivative . . . . .	45

3.4.2	Evolution in Time of Delayed Neutron Precursors . . . . .	46
3.4.3	An Equation for the Flux . . . . .	46
3.4.4	Even-parity Flux Equation . . . . .	47
3.4.5	Solution Schemes . . . . .	47
<b>4</b>	<b>SIMMER-PARTISN coupling . . . . .</b>	<b>49</b>
4.1	SIMMER-PARTISN Coupling Procedure . . . . .	50
4.1.1	Data Flow . . . . .	50
4.1.2	Additional Data Files . . . . .	51
4.1.3	Modifications in SIMMER . . . . .	52
4.1.4	Modifications in PARTISN . . . . .	53
4.2	Verification of SIMMER/PARTISN Coupling . . . . .	55
4.2.1	Two Dimensional Space-Time Neutronics Problem . . . . .	55
4.2.2	A Two-Dimensional Fast Reactor Unprotected Loss-of-Flow . . . . .	59
4.2.3	An ESFR Unprotected Loss of Flow . . . . .	66
4.2.4	Three-Dimensional Space-Time Neutronics Problem . . . . .	75
4.3	EBR-II Shutdown Heat Removal Experiments . . . . .	77
4.3.1	EBR-II Reactor Description . . . . .	77
4.3.2	EBR-II Shutdown Heat Removal Test SHRT-45R . . . . .	80
4.3.3	EBR-II SHRT-45R SIMMER-III Model . . . . .	81
4.3.4	EBR-II SHRT-45R Validation . . . . .	82
<b>5</b>	<b>VARIANT with Heterogeneous Nodes . . . . .</b>	<b>87</b>
5.1	Heterogeneous Node Technique . . . . .	88
5.1.1	Evaluation of Response Matrices . . . . .	88
5.1.2	A Simple Heterogeneous Node . . . . .	89
5.1.3	A New Set of Orthonormal Functions . . . . .	90
5.1.4	Evaluation of Response Matrices for Heterogeneous Node . . . . .	90
5.1.5	Evaluation of Coupling Coefficient Matrices . . . . .	90
5.2	Flux Reconstruction . . . . .	92
5.3	Intra-Nodal Power and Reactivity Calculations . . . . .	93
5.4	Numerical Results . . . . .	96
5.4.1	Reactor Scale Heterogeneities Modeling . . . . .	96
5.4.2	PWR Cell Calculations . . . . .	103
5.5	Some Remarks on the Heterogeneous Technique . . . . .	106
<b>6</b>	<b>SIMMER-KIN3D Coupling . . . . .</b>	<b>111</b>
6.1	Coupling Procedure . . . . .	112
6.1.1	Input KIN3D Perturbations: a Simple Scheme . . . . .	112
6.1.2	Input KIN3D Perturbations: a More Refined Approach . . . . .	113
6.1.3	Geometric Interface . . . . .	113
6.2	Verification of Coupling Approach . . . . .	114
6.2.1	ULOF Transient Fast Reactor . . . . .	114
6.3	Application of SIMMER/KIN3D . . . . .	116
6.3.1	Two HET Models . . . . .	119



<b>7 Conclusion . . . . .</b>	<b>125</b>
<b>Bibliography . . . . .</b>	<b>129</b>
<b>Nomenclature . . . . .</b>	<b>137</b>
<b>Acronyms . . . . .</b>	<b>143</b>



# Chapter 1

## Introduction

The ever-increasing human population is putting more and more strains onto the limited natural resources that our planet Earth can offer. By 2040, the human population is predicted to be around 8-9 billion; the corresponding electric energy demand by that time will have increased by  $\approx 80\%$ , as forecasted by the International Energy Agency (IEA) in its energy outlook IEA [2014]. Meeting the needs of that people is the challenge humanity will be facing in the next few decades. It is also more and more evident that the current energy policies cannot be pursued any longer. A drastic reduction of carbon dioxide emissions is necessary to avoid the direct consequences of climate changes, as highlighted in another IEA report IEA [2015]. Renewable energies must play a dominant role into the future energy mix. However, nuclear is one of the few other carbon-free energy options, and, as such, should not be completely ruled out of the energy mix. Nuclear energy has shown its ability to provide reliable baseload electricity and has avoided, since 1971, the release of an equivalent of two years of  $\text{CO}_2$  at present  $\text{CO}_2$  emission levels.

To investigate the most promising options for the next generation of nuclear reactors, the Generation-IV International Forum (GIF) was established. Six different reactor concepts emerged satisfying high safety, reliability, and sustainability criteria, as explained in GEN [2015]. Supercritical Water Reactor (SCWR) and Very High Temperature Reactor (VHTR) are two examples of thermal spectrum reactors. Fast Reactors (FRs) differentiate themselves from the coolant: Sodium Fast Reactor (SFR), Lead Fast Reactor (LFR), and Gas Fast Reactor (GFR) options are being investigated. Aside from the above mentioned reactors is the Molten Fuel Fast Reactor (MFR), in which the fuel is liquid. Fast spectrum reactors provide the best option to maximize the use of natural resources. Actual commercial reactors can use effectively only few fissionable isotopes – e.g.  $^{235}\text{U}$  – that made a small fraction of the total natural resources – i.e. natural uranium. Fast reactors, on the other hands, provide a way to use the natural resources in a much more efficient way.

Even if the nuclear option is to be ruled out of the energy mix in the future, fast reactors might still be used as radioactive waste burners. Past use of nuclear reactors has generated a large quantity of nuclear waste. No country has already a definitive plan on how to handle them. Due to their spectrum, fast reactors can effectively transmutate long-lived radioactive elements, reducing both mass and radio-toxicity of the wastes. Either way, future operation of fast reactors will be possible only if we can demonstrate that they

satisfy strict safety criteria. It is therefore of primary importance to understand the behavior of these machines under nominal and accidental conditions. This piece of research has to do with the simulation of transients and accidental sequences in fast reactors. In an area, in which little-to-no experimental campaigns are possible, computer simulation has become the “de facto” tool to test transient behavior of fast reactors, to indicate possible design flaws, to suggest accident-mitigating solutions to reactor designers. Developing a whole spectrum of high-fidelity, reliable, and fast simulation tools is fundamental in the advancement of fast reactor industry. In the last few years, there has been a trend to establish coupled multi-physics codes – i.e. fluid-dynamics and neutronics code coupling – to attempt a mechanistic simulation of accidental and transient sequences. Due to fast reactors peculiarities, neutronics plays an important role in the evolution of a transient. Here, several different neutronics solvers are investigated and coupled with a fluid-dynamics code. This chapter briefly introduces fast reactors, and sets the background for fast reactor accident analyses. It underlines in which aspect fast reactor modeling differs from commercial thermal reactors, it provides some typical reactor responses in the most commonly studied accidents, and then gives a more systematic view of the challenges of simulating FR accidental sequences.

## 1.1 Fast Reactors

Attractiveness for FRs has been recognized already in the infancies of nuclear industry. The key physical quantity is  $\eta$ , i.e. the average number of neutrons emitted per fission per neutron absorbed. For an isotope  $i$ , this quantity is defined as:

$$\eta_i(E) = \frac{\nu\sigma_f^i(E)}{\sigma_a^i(E)},$$

where  $\sigma_f^i(E)$  and  $\sigma_a^i(E)$  are the microscopic cross sections for fission and absorption, and  $\nu(E)$  is the average number of neutrons emitted in a fission event. All these quantities depend on the neutron energy  $E$ . It was discovered that this quantity is higher in a fast rather than in a thermal spectrum, as mentioned by Beynon [1974]. An higher  $\eta$  means more fission neutrons are available, per neutron absorbed, in a fast spectrum. It was then discovered that upon neutron capture, fertile isotope  $^{238}\text{U}$  is transformed into a fissile isotope:  $^{239}\text{Pu}$ . The neutron excess could then be used to convert non-fissile  $^{238}\text{U}$  into a fissile isotope, thus breeding fuel from the fertile isotope  $^{238}\text{U}$ . With a careful reactor design, a configuration in which more fissile isotopes would be produced than destroyed in the fission chain reaction could be achieved. It would then be possible to use the much more abundant natural resources of fertile isotopes to breed fissile fuel, rather than rely on scarce natural resources of fissile isotopes.

The Breeding Ratio (BR) renders this situation quantitative. It is defined as:

$$\text{BR} = \frac{\text{fissile material produced}}{\text{fissile material destroyed}} \quad (1.1)$$

If  $\text{BR} > 1$  than more fissile material is produced than destroyed: such a system is called breeder. An iso-generator is instead a  $\text{BR} = 1$  system. A neutron balance relates the BR to

$\eta$ . Consider a multiplicative system characterized by a single value of  $\eta$  – this value can be a convenient average over the different isotopes and over the energy variable. In such a system, every neutron absorbed in the fissile material generates  $\eta$  neutrons in the next generation. Of these  $\eta$  neutrons, one must be absorbed in the fissile material to sustain the fission chain reaction. Another fraction  $L$  either leaks out of the reactor or is absorbed by non fissile and non fertile materials. Therefore, the quantity

$$\eta - (1 + L), \tag{1.2}$$

represents the surplus of neutrons produced per neutron absorbed in the fissile material, or – since the two quantities coincide – per fissile material destroyed. If this surplus of neutrons is absorbed by some fertile material, and if, upon absorption, it is transformed into fissile material, then (1.2) identifies with the BR itself:

$$\text{BR} = \eta - (1 + L).$$

A situation, as the one described above, in which more fissile material is produced than destroyed, requires the BR to be greater than one, or in terms of  $\eta$ :

$$\eta > 2$$

The advantage – in terms of breeding only – of a fast over a thermal spectrum is evident when comparing averaged values of  $\eta$ . In a thermal spectrum,  $\eta$  values for isotopes  $^{239}\text{Pu}$  and  $^{235}\text{U}$  are respectively 2.04 and 2.06. These values make practically impossible to achieve breeding in a thermal spectrum. A fast spectrum shows  $\eta$  values of 2.43 and 2.10 for  $^{239}\text{Pu}$  and  $^{235}\text{U}$ , hence significative breeding ratio can then be achieved. A fast system with  $\text{BR} > 1$  is called Fast Breeder Reactor (FBR). On the other hand, the prevalence at high energies of fissions over captures could be used to fission Minor Actinides (MAs), that are responsible for the long-term radio-toxicity of nuclear spent fuel. This approach is known as transmutation, and can ease heat and radioactive loads onto geological repositories. Among the competing GIF technologies, the sodium fast reactor is maybe the most advanced one, as it can drawn from experience gained by several operating reactors: Experimental Breeder Reactor-I (EBR-I) and Experimental Breeder Reactor-II (EBR-II) in the US, PHENIX and SUPERPHENIX in France, and MONJU in Japan, just to name a few. A full list is in the International Atomic Energy Agency (IAEA) report [Int, 2007].

Several nations launched their own national fast reactor research program. For an historical view of these programs, see Cochran et al. [2010]. Within these programs, a lot of efforts had been dedicated to understand the dynamic behavior of fast reactors, as their response, should an accident happen, is fundamental in preventing any radiological release, and, as such, is being used in the reactor licensing process.

## 1.2 Fast Reactors Plant System

The principal objective of FRs is to produce electricity. Several basic components are present in every reactor: the core, the heat transfer system and a turbine-generator system. In this section, I provide a short description of each of these major components. Since the

sodium-cooled is the only Liquid Metal Fast Breeder Reactor (LMFBR) ever built and operated, examples in this section are mainly based on that technology. The core is the heat-producing unit. The energy is released locally (i.e. in the nuclear fuel) in form of thermal energy. It is then taken away by the coolant flowing through the core. The hot coolant enters the Intermediate Heat Exchanger (IHX) where this energy is transferred to an intermediate loop. The cold coolant enters again into the core. Circulation is provided by pumps. The intermediate loop simply transfers energy to the turbine-generator system. This arrangement corresponds to the loop-type design. In this design, each element (e.g. the core, the pumps, the heat exchangers) are located in separated cells and coolant flows via piping. A second approach features instead a large pool that contains not only the core but the pumps and the IHXs. Possibly, an intermediate loop, further separating the primary from the turbine loop, as in the loop-type plant, can be present.

### 1.2.1 The Core

The basic unit of the core is the fuel pin. It is essentially a slender, cylindrical rod containing the nuclear fuel and clad with steel. The clad separates the fuel from a direct contact with the sodium. The outer diameter of the clad is few millimeters, typically 6-8 mm, while the total height is several meters, e.g. 3 m.

Axially we can identify few different regions. The active region consists of a stack of fuel pellets, usually approximately 1 m high. Nuclear fuel is the form of U-Pu oxide. Below and above the active core region, there are fissile pellets, for a total blanket height (lower plus upper) of 0.3-0.4 m. This stack of pellets is kept in place by a spring located atop the pellet column. The pin provides also significative empty space that will accommodate the fission gases produced during irradiation. A wire is usually wrapped around the clad to separate pins from each others. Several pins are assembled in a cluster called assembly. Typically, one assembly contains circa 200 pins, arranged in a triangular lattice, to increase the fuel volume fraction. This pin lattice is wrapped by the assembly wall or duct. Coolant flows through lateral openings in the assembly duct (coolant inlet ports) and then upwards in the pin bundle section. The core is finally assembled by putting together several assemblies in an hexagonal lattice. A large reactor usually contains up to 3-4 t of fissile material.

A reactor contains different types of assemblies other than fuel assemblies. Control assemblies provide a means to control the fission chain reaction and therefore the reactor power. They are used for normal startup and shutdown procedures, as well as emergency shutdowns in accidental conditions. They are made with B which has a high cross section for neutron absorption (especially the isotope  $^{10}\text{B}$ ). This material is usually arranged in a pin bundle section. Shielding or reflector assemblies are used instead to shield the core surroundings from leaking gammas and neutrons. These assemblies are in the form of steel rod bundles, to allow sodium cooling. Particular important in FRs are the blanket assemblies. It is in this assemblies, in fact, that the fissile fuel is generated from fertile via neutron capture. They look very similar to a fissile assembly. However, the fuel pellets are replaced by fertile ones (e.g. made with depleted  $\text{UO}_2$ ). The fertile subassembly arrangement in the core defines two different concepts of breeding: homogeneous against heterogeneous breeding. In the first, blanket assemblies are mixed with fuel assemblies

in the core region. In the heterogeneous concept, blanket assemblies are placed together around the core region only. For loop-type reactors, a cylindrical steel shell provides further protection. This shell is known as reactor vessel. For pool type reactors, the vessel is usually called reactor tank. It has a bigger volume and contains the core, the pumps, and the intermediate heat exchangers. Another internal tank is necessary to separate the hot and the cold sodium legs in the pool design.

### **1.2.2 Intermediate Heat Exchanger**

The IHX receives sodium from the reactor hot leg (primary side) and transfers its energy to the intermediate sodium loop (secondary side). Almost all IHXs share the same design. They are vertical, counter-flow, shell-and-tube heat exchangers. Primary sodium enters at the top and flows downwards on the shell side. It then exits at the bottom of the IHX. The secondary sodium also enters at the top and flows downwards in the downcomer, it then reverse its direction, flowing upwards through parallel, vertical tubes. The sodium is collected then at the top before exiting the IHX. On the primary side, sodium enters the IHX at around 550 °C and it is discharged at 400 °C. On the secondary side, secondary sodium heats up from 350 °C to 500 °C or more.

### **1.2.3 Steam Generator**

Two types of steam generator are possible. In the integral type, all the major functions of a generator (i.e. economizer, evaporator, superheater) are included in an unique large unit. A modular steam generator has, instead, dedicated units for each of those functions. For sodium-cooled FRs, the steam generator is a critical point, because of the possible contact between sodium and water, should a pipe break happened. Several safety barriers have been considered, some design proposed the use of double-wall tubes. The relatively high sodium temperatures allow the use of a superheater, to reach a steam exit temperatures of around 520 °C and 160 bar. Thermal efficiency of such a steam loop is around 40%, higher than the typical 32% of Light Water Reactors (LWRs).

## **1.3 Fast Reactor Safety Considerations**

Every nuclear installation has the potential of radioactive release, and FRs are no exceptions. Safety analyses of FRs reflect in many ways that of LWRs, and indeed have benefitted from it. In LWRs, safety analysts have focused on accidental sequences known as Design Basis Accidents (DBAs). They are accidents that the plant must be designed and built to withstand without loss to the systems, structures, and components necessary to ensure public health and safety. In these accidents, it is usually postulated a complete failure of all the in-built active systems and installations by internal or external events of credible probability. The most investigated sequences follow a coolant pipe break in the cold leg between the coolant pump and the reactor vessel. The vessel is itself in pressure: a severe pipe break there leads to coolant flashing and blow-down (i.e. coolant being rapidly discharged from the primary system). Should such a situation occur, only intervention of the Emergency Core Cooling System (ECCS) might prevent a core melting. In contrast to

LWRs, mostly being moderated, a loss of coolant does not necessarily decline the reactor power in FRs. It may lead to power excursions necessitating a priori even higher quality design and safety measures in FRs compared to LWRs.

On the other hand, safety analysis for FRs has been mainly concerned with quantification of the maximum mechanical work that might be released on the core-surrounding structures during an accidental sequence. This energy release poses a challenge to the containment of the reactor. For example, a power burst might lead to violent sodium vaporization, and a subsequent pressure built up causing a failure of the primary system. Despite having catastrophic consequences, such accidents have very low probability to happen, even beyond that of DBAs. In fact, for such power bursts to occur, not only all plant protection systems must fail, not only the core must be melt (or at least a great portion of it), but an unlikely incident progression must be assumed. Such accidental sequences have been referred to as Hypothetical Core Disruptive Accidents (HCDAs), and their modeling is the major concern of FRs safety analyses.

## 1.4 Overview of Hypothetical Core Disruptive Accidents

It was recognized in the early stages of FR safety analysis [Waltar and Reynolds, 1981] that the dominant feature in accidental sequences is that of recriticality. As opposed to LWRs, fuel in FRs is not in the most reactive configuration: in other words, there is the possibility that a rearrangement of core geometry can lead to a prompt-criticality condition. For example, core structures may melt and collapse under gravity. This core compaction would insert positive reactivity into the system, power and fuel temperatures would rise until, finally, fuel vaporization and expansion would provide an inherent, negative feedback mechanism. By that time, the energy released might be higher than practically confineable. A typical FR accident comprises several recriticality events and power peaks, see figure 1.1. The first attempt to provide an upper limit on the energy release dates back to 1956 [Bethe and Tait, 1956]. The importance of that paper is not in the numbers, but rather in setting the framework for future FR safety analyses.

Since then, modeling improvements in HCDAs resulted in lower and lower predictions of the maximum energy in several FR systems. This reduction was achieved by a mixture of mechanistic modeling – using large computer codes to track the evolution of the accident from the start to the end –, of experimental campaigns – where possible –, and of phenomenological approach – simplifying the accident by excluding some paths based on general behavior principles that have been proved true on similar situations. Using this three-way approach, a fair characterization of HCDAs has been achieved. The next sections provide an overview of the principal phenomena involved in HCDAs.

### 1.4.1 Accident Initiators

Accident initiators are the causes that set the start of the accidental sequence. It is generally believed that, should the Plant Protection System (PPS) intervene, the accident can no longer evolve and the plant is brought into a safe, cold shut-down condition. These transients are indicated with the term “protected”.



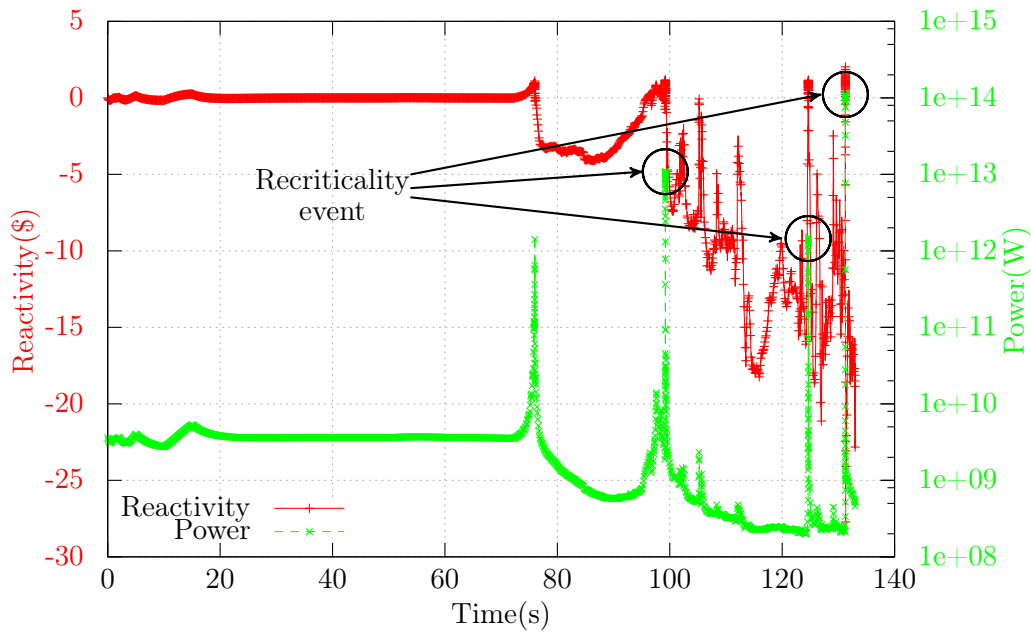


Figure 1.1: Schematical temporal evolution of reactivity (\$) and reactor power (W) of a generic Fast Reactor exhibiting several recriticality events.

Core disruption can only happen upon failure of PPS intervention, therefore HCDAs are sometimes referred to as “unprotected transients”. Since HCDAs involve an extensive core melting, accident initiators necessitate an imbalance between power produced and power removed by the coolant flow. They fall into two broad classes: Transient Overpower accident (TOP) and Transient Under Cooling accident (TUC).

The TOP refers to a situation, in which positive reactivity is inserted (e.g. withdrawn of one or more control rods), leading to power increase, while the coolant flow through the core is maintained. The TUC situation refers, conversely, to a reduction in the ability of the coolant to cool the core. These accidents can be initiated by a decreasing coolant mass flow rate through the core (e.g. lost power to the pumps). This class of accidents is sometimes indicated as Loss of Flow accidents (LOFs). Sometimes, to emphasize the failure of the PPS, the accident is indicated as Unprotected Loss of Flow accident (ULOF). Also to this class belongs the Loss of Heat Sink (LOHS), where a loss in the ability to remove the heat from the intermediate loop increases the core inlet coolant temperature and reduces the heat transfer between the fuel and the flow.

### 1.4.2 Transient Overpower Accidents

In the early phases of a TOP, the insertion of reactivity leads to an instantaneous rise in reactor power and thereby temperatures. This has several consequences: rising temperatures increase neutron absorption (i.e. Doppler effect), and expand fuel and steel bringing negative reactivity that counterbalance the reactivity insertion. However, temperature rise may result in fuel melting and eventually in clad failure: in such a scenario, melted fuel is discharged into the coolant flow and moves inside the reactor. This motion may lead to a further insertion of positive or negative reactivity, depending on the pattern of this

fuel movement. The dynamic response of the system depends therefore on the integral outcome of these effects: on the one hand, thermal expansion and Doppler effects tend to insert negative reactivity; on the other hand, fuel movement (relocation) may insert both positive or negative reactivity.

A typical response in case of a TOP event is shown in figure 1.2. There, insertion of positive reactivity leads to higher temperatures. At the start of the accident, only the Doppler effect is countering the positive insertion of reactivity, while fuel reactivity is zero. However, pins eventually fail: fuel becomes movable.

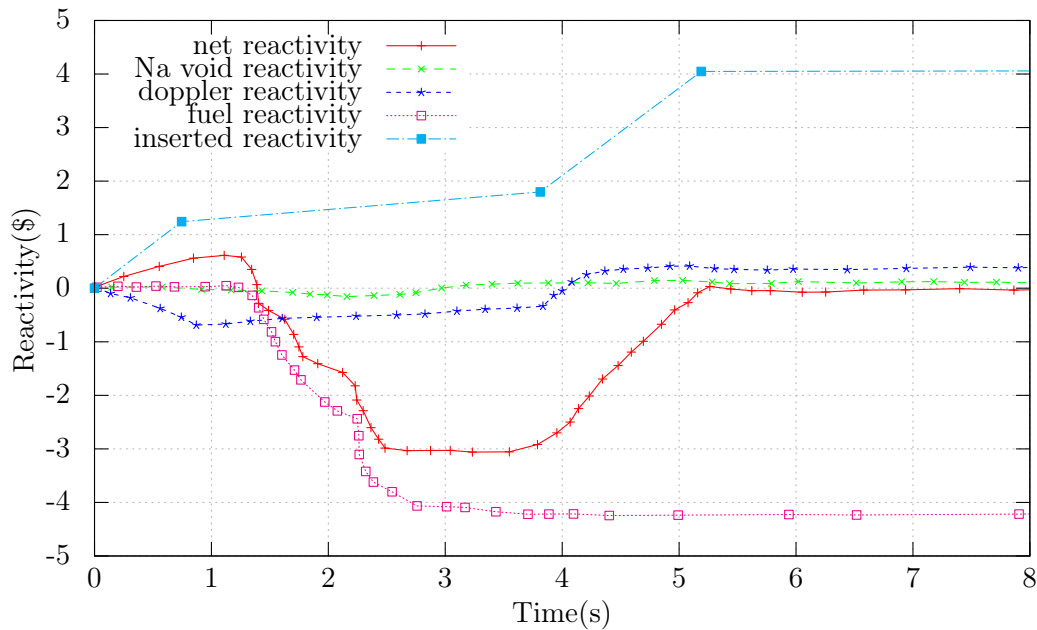


Figure 1.2: Temporal evolution of the individual reactivity contributions and the net reactivity in a generic TOP event.

This is reflected by the sudden change in fuel reactivity shortly after 1 s into the transient. In this accidental instance, fuel is pushed by the coolant flow out of the core, and the effect is a negative contribution. This negative fuel reactivity insertion is sufficient to bring the reactor into a subcritical condition, and then a new steady state is eventually achieved. One of the key aspects is the pin failure location. Fuel inside the pin may be melted and can move axially within the pin. Upon pin failure, the fuel moves towards the location of rupture: this fuel movement can bring positive or negative reactivity. If the rupture happens at the core mid-plane, then melted fuel moves from above and below towards the core center, therefore increasing reactivity. The opposite is true, if the rupture is away from the core mid-plane. The second question is that of heat transfer between fuel and coolant (e.g. sodium). Upon contact, sodium might vaporize almost instantly. This can bring flow reversal, even though this reversal might last a small spell of time, then the sodium pressure recedes, and the normal flow is restored. By that time, the melted fuel may be dispersed outside the core, a phenomena known as fuel sweep-out. This fuel movement may be enough to terminate the transient. In this respect, it is important to investigate the possible formation of blockages: melted fuel may solidify upon contact with colder

surfaces (e.g. upper or lower blankets) preventing any fuel discharge from the core and blocking any path for fuel cooling. In this accidental path, ever rising temperatures would cause more and more fuel melting: a pool consisting of a mixture of melted fuel and steel would form. In this scenario, an energetic disassembly of the core becomes very likely.

### 1.4.3 Transient Undercooling Accidents

In this type of incidents, the ability of the coolant flow to cool the core is reduced, while the power maintains its steady state level. A typical TUC response is shown in figure 1.3. The first effect is a rise in temperatures. Doppler and thermal expansion effects produce a mild reduction in power level. Rising sodium temperatures lead to a lower sodium density and this, in large cores, inserts positive reactivity. Eventually, the sodium boils, accompanied by a further reduction in its density and its heat transfer coefficient.

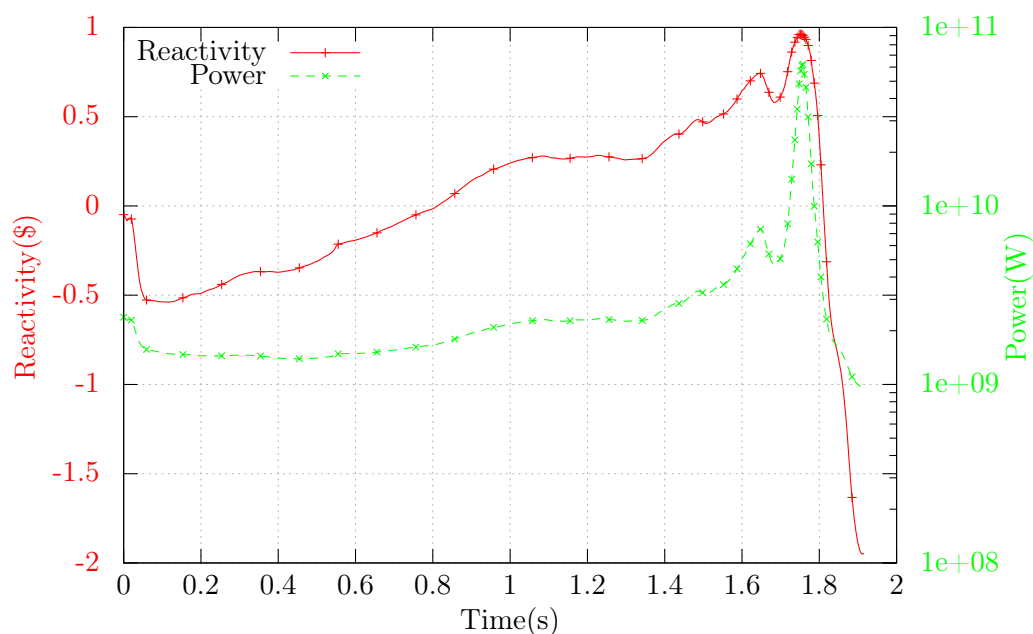


Figure 1.3: Temporal evolution of reactivity (\$) and power (W) of a generic FR in case of a Transient Undercooling accident.

As a consequence, clad temperatures rise to the melting point: steel leaves the core, shifting the neutron spectrum towards higher energies, thus inserting again a positive reactivity, and hence driving a mild, positive power rise. This energetic excursion causes several pin failures. Fuel is leaked into the coolant channel. It is then swept away from the core, terminating the power rise. In general, fuel movement has the potential to either terminate the transient – if the fuel moves out of the core – or to lead to a power burst.

In the latter case, the core melts throughout and a core-wide molten fuel pool would be formed. For FR accident analysis it is therefore crucial to represent as accurately as possible how the fuel is transported by the coolant flow.

## 1.5 Objective of the Research

As shown by this short overview, two aspects are essential to describe realistically Core Disruptive Accidents (CDAs) for fast reactors:

1. an accurate prediction of recriticality events, accompanied by
2. a faster and accurate simulation environments.

As I have already mentioned in section 1, recriticality events clearly dominate the behavior of any accidental sequence.

It is therefore of paramount importance to evaluate accurately reactivity insertions and ramp rates throughout the accident. This requires an accurate neutronic model. This model should evaluate the spatial distribution of the neutron flux by solving the neutron transport equation and use a perturbation theory module to evaluate the corresponding reactivity level. This is a critical point, as little differences in reactivity can strongly affect power, and consequently determine the rest of the transient – especially, when the reactor is near prompt-criticality, the power rises exponentially with reactivity. Discrete ordinates codes were, and are, used for neutron flux description in CDA analyses. Usually a low order discrete ordinates quadrature is used, such as  $S_4$ . That means that only 3 neutron streaming directions are considered for each octant: a rather simplified angular description, which in turn reflects into only approximate reactivity values. One straightforward solution is to use higher order quadratures to provide a better description of the neutron flux. On the other hand, richer neutron flux angular approximations can also be used, such as spherical harmonics  $P_N$  or simplified spherical harmonics  $SP_N$  methods. The same situation affects the energy dependence of the neutron flux, where a limited number of energy groups is usually used, typically eleven or eighteen.

Present and past CDA analyses have been mainly confined in two-dimensional domains. While two-dimensional accident modeling helps to understand the relevant phenomena at play in a fast reactor accident and has shaped the default approach to CDA studies, – i.e. division of the accident in phases, see figure 2.1 – they represent a gross simplification, as it forces a toroidal symmetry – i.e. symmetry with respect to the  $\theta$  angle – which can not be totally justified in reality. A full three-dimensional model would provide of course more realistic results and more realistic estimates of the work potential. Early examples of three-dimensional modeling – e.g. the ones done by Yamano et al. [2009] – exhibited a different sloshing behavior compared to two-dimensional cases. A full three-dimensional model in fact accounts for asymmetries in the toroidal direction, and they would likely hamper sloshing patterns. This results in milder reactivity insertion. However, the addition of one dimension requires more computational power: Yamano et al. [2009] reported a total computational time of more than two months for simulating only 5 s in the accident. Such time scales are clearly not practical. Moreover, very often sensitivity studies over unknown parameters – e.g. fuel particle radius upon pin breakup – are required to bound the behavior of the reactor within upper and lower “boundaries”. It is therefore clear that a better simulation of CDAs requires fast and accurate simulation tools. This research focuses on the neutronics part of CDA simulations. I compared different neutron solvers: coupling them with the SIMMER code and performing several accident and transient analyses. One

of the most obvious way to speedup a calculation is to use parallel codes. I attempted that way by using the parallel neutron transport solver PARTISN. This coupling is described in detail in chapter 4. The chapter introduces and explains the coupling procedure, focusing in particular on the physical quantities that need to be transferred from the fluid-dynamics module of SIMMER to the PARTISN neutron solver. Several applications of this coupling are then provided. They form a sound verification of this coupling against the standard SIMMER code. Simulation of an ULOF in the EBR-II provides also a first validation test of the coupled code.

As both PARTISN and the standard neutron solver already embedded in official releases of SIMMER are all discrete ordinates codes, I tried to use also other numerical methods in solving the transport equation. Nodal methods have shown their ability in predicting both integral parameters – such as k-effective, reactivity effects – and local quantities – such as neutron flux and power density distributions. I pursued this approach by using the nodal code VARIANT as an alternative neutron transport solver in SIMMER. Computational time in nodal methods is mainly determined by the number of computational nodes in the domain: keeping this number low is therefore essential, especially in transient simulations, where solution of the neutron transport problem might be required on a very small time scale. But few nodes actually imply low accuracy. A balance must therefore be found. For that purpose, I developed a VARIANT extension to heterogeneous nodes that might allow the user to use few big nodes in the neutronic mesh still retaining a good level of accuracy, as the heterogeneity of the nodes is still accounted for. This extension is the topic of chapter 5. There, the stationary (i.e. no time dependent) performances of this nodal technique are discussed. Several applications of this heterogeneous extension provide a first verification of the technique against the standard VARIANT code. The capability of this heterogeneous VARIANT extension for FR transient and safety analyses are investigated then in chapter 6, where the technique is coupled with the SIMMER code. After detailing of the coupling procedure, several verification cases are presented.



## Chapter 2

# Accidental Phases

It is evident from the short discussion in chapter 1 that an accident analyst is facing a rather complicated task. The outcome of an accidental sequence is the integral result of several different physical phenomena acting at the same time. Correct accident modeling requires to track these phenomena. To simplify the modeling, the approach used in literature is to divide the accident in phases and to focus on each of them singularly. This separation is essentially based on the different physical phenomena that are predominant at each stage of the accident. Coarsely speaking, four different phases are usually indicated in literature:

1. Initiation phase
2. Transition phase
3. Core disassembly phase
4. Expansion phase

During an accident, several different paths are possible. A complete characterization of HCDAs was achieved through several years of experimental campaigns and numerical simulations. It can be condensed into the figure 2.1 on the next page. Broadly speaking, two main outcomes are possible in an accident:

1. an energetic route (dashed line), which should be avoided
2. a less energetic route (solid line)

An accident starts in the initiation phase. An imbalance between reactor power and coolant flow would cause high temperatures, and would result in fuel pin damages. If pins fail at a high rate, the accident would energetically lead to the disassembly phase. This phase would inevitably lead to an expansion phase, in which the thermal energy that released in the fuel would be converted – by some mechanisms – into mechanical work. This work impacts onto the reactor surroundings, mainly the reactor vessel.

If, in the initiation phase, pin failures are rather gradual, than a non-energetic path into pre- and transition phases follows. A gradual core melting results in the formation of a pool of melted fuel and steel. Central to the transition phase is whether recriticality event occurs or not. These events might be triggered by disturbances such as sloshing, pool

collapse and compaction. In those cases, an energetic disassembly phase may probably follow. On the other hand, if fuel, in the transition phase, is gradually discharged from the active core region, and if recriticality events are not so severe leading to high reactivity insertion, a benign neutronic shutdown takes place. The accident finally terminates with no major energetic events. In the next few section, I describe in more details the main characteristics of each of these phases.

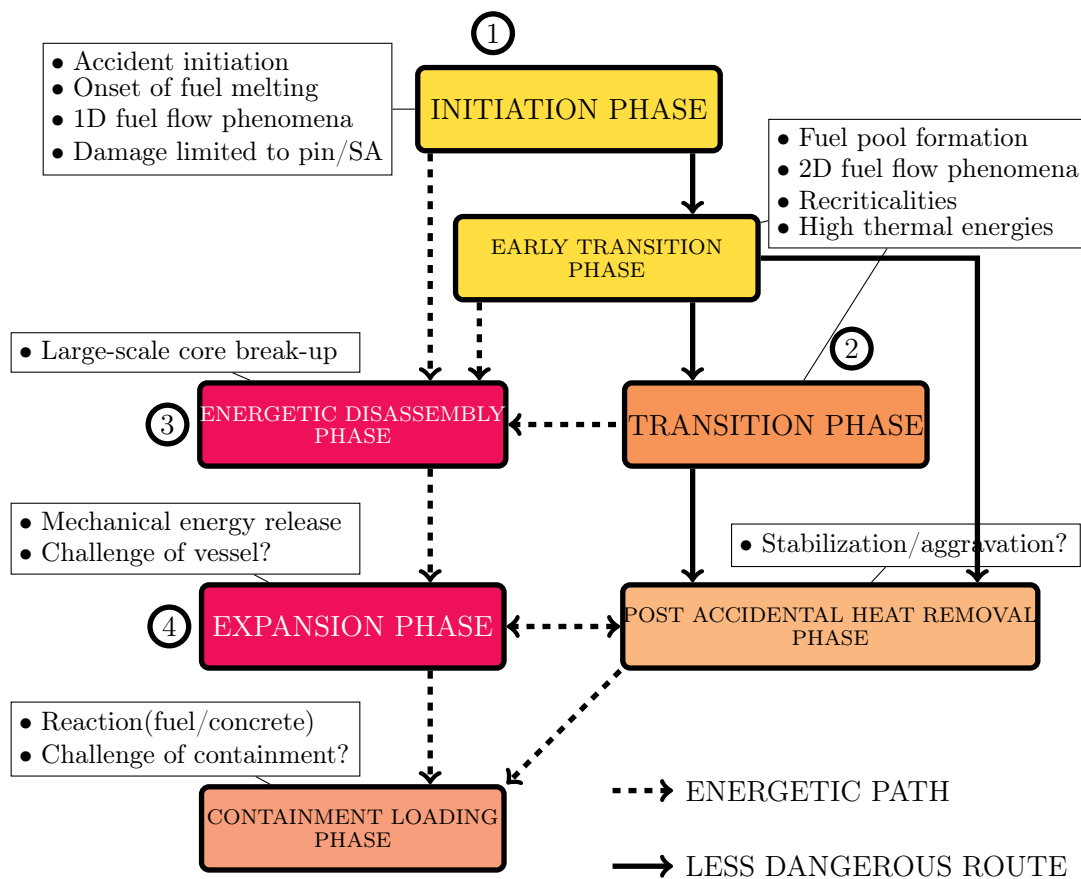


Figure 2.1: Possible paths of a HCDA accidental sequence and different conceivable routes possible.

## 2.1 Initiation Phase

The initiation phase covers the initiation of an accident and ends in one of the following scenarios:

1. neutron shutdown with an essentially intact core geometry or a
2. gradual core melting, leading to a transition phase or an
3. energetic core disassembly.

Steady-state conditions represent an important factor in this phase. Since the outcome of the modeling can be different, it is of vital importance to describe accurately the reactivity feedback coefficients. These comprise thermal expansion, void reactivity, – i.e. reactivity



associated with coolant voiding of the reactor – Doppler effect, fuel melting and fuel motion. Typically at the onset of the accident, the neutron flux shape changes only mildly with respect to the one present at steady-state conditions. In these cases, a point kinetics approach – as far as neutronic is concerned – can be considered to be the first order acceptable. Reactivity feedback coefficients are then evaluated on the basis of the steady-state neutron flux distribution. If the total reactivity feedback is negative enough, the transient ends with no or limited core damage. Alternatively, the accident might evolve into the transition phase. Analyses done by Nonaka and Sato [1992], Theofanous and Bell [1985] showed that for core with a moderate, positive void reactivity, the most common outcome of the initiation phase was indeed a progression into the transition phase.

Since subassemblies are separated by rigid can-walls, each subassembly is almost independent from the others. Mainly, subassemblies tend to behave differently according to their power-to-flow ratio. Key phenomena in high-power subassemblies comprise: coolant boiling, clad melting and relocation, fuel melting, fuel motion with freezing or dispersal. For low-power subassemblies, key phenomena are: pin failure mechanism, Fuel-Coolant Interactions (FCIs), fuel motion in the channel. Interaction between fuel and sodium may lead to sodium vaporization. This vaporization might increase sodium pressure and cause flow reversal. This can affect fuel relocation mechanism. The question of FCI has been studied intensively. It is generally believed that, should hot liquid fuel come into contact with coolant, the intense, intimate mixing and energy transfer required for a sodium vapor explosion appear very unlikely, see for example Fauske [1976]. Fuel and clad relocation have been studied by several in-pile experiments by Wright et al. [1985a,b]. Several modes of fuel pin disruption have been observed: solid fuel breakup in chunks or as a dust cloud, frothing of liquid fuel, melting and slumping. Driven by fission gases, each of these modes may enhance and/or hamper fuel dispersal. A more complete characterization of pin disruption and its impact on accidental behavior is found in Bandyopadhyay and Buzzell [1980]. Sodium boiling has been extensively investigated experimentally and theoretically. Several experiments targeted sodium boiling in heated subassemblies under LOF conditions, investigating the influence of parameters such as pump rundown, radial power distribution, and power level. Bottoni et al. [1990] performed several boiling sodium experiments for a 37-pin bundle mocking the SNR-300 Mark 1a fuel element. The campaign investigated the fuel behavior under different ULOF conditions and provided a rich experimental database for validation of sodium boiling phenomena in computer codes, such as the SAS-series codes developed by Tentner [1985].

## 2.2 Transition Phase

Negative feedback coefficients might avert a core disassembly during an initiation phase. However, it has been realized that this mechanism might not be strong enough to bring the reactor into a safe, cold shutdown: the accident evolves into the transition phase. This phase is characterized by a significant core melting and material relocation. Several power bursts are typical in this phase, followed either by a neutronic shutdown and an accident termination or by an energetic disassembly of the core. At early stages of this phase, fuel motion is confined to several adjacent subassemblies still on a subassembly

level. This incoherent motion prevents strong reactivity insertions: the net reactivity remains low and the power eventually rises slightly. Rising temperatures may then lead to extensive hex-can melt-through and a single, core-wise pool consisting of a mixture of molten steel and fuel can be formed. Coherent fuel motion is now possible and may lead to high reactivity insertions. In such a case, a prompt-criticality condition is achieved. It is therefore important to investigate the process, by which the pool is formed and how it behaves. As for pool formation, the question of fuel/steel blockages is of primary importance: upon contact with colder surfaces (e.g. upper and lower blankets) fuel might freeze, obstructing passages between pins and preventing a gross discharge of molten fuel from the active core region, as depicted in figure 2.2. Experimental and analytical studies show that significant quantities of molten steel and fuel relocate in the lower core region, freezing and blocking the coolant flow, see for example Jones [1990]. As for the upper region, the question is not so well defined, with several experiments showing discordant behavior. As a consequence, a situation in which a pool sustained by frozen fuel is envisaged in accidental analysis.

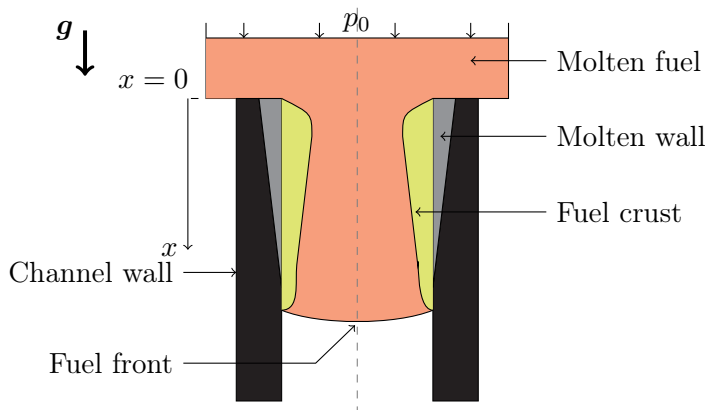


Figure 2.2: Sketch of frozen fuel due to colder steel surfaces (after core meltdown), eventually blocking fuel discharge from the core active region.

The pool is said to be open if no upper blockage is present, bottled-up otherwise. Different freezing models have been proposed in literature. Best et al. [1984] proposed a first model, the conduction limit model, in which heat is transferred from the fuel mixture to the steel walls. Frozen fuel would attach onto walls. This fuel layer would then increase as more and more fuel freezes and deposits, eventually leading to a total blockage of the sub-channel. Several arguments run against this model: the upper structure's wall can not provide a surface the fuel can attach to. Therefore, the slug of steel/fuel would freeze in a bulky-type fashion once the fusion heat has been drawn from it. This is the bulky-freezing model, see Epstein et al. [1973] for a more detailed description of the model. In general, experimental data are in favor of the bulky-freezing model, but even better agreement is achieved by allowing the ablation of molten steel into the flowing fuel/steel mixture. In this ablation model, the correct proportion of fuel and steel is accounted for.

The pool behavior depends, on which flow regime develops within the pool. As the melting temperature of the fuel is a few degrees below the steel vaporization temperature, melting fuel is always accompanied by a substantial steel boiling: the flow in the pool is

then of two-phase and multi-component type. Since flow regimes have potential for fuel dispersal, experimental and theoretical investigations of pool behavior have been performed in the past. As for flow regimes, analyses have identified three different flow regimes: bubbly, churn, and dispersed flow. Bubbly flow is prevalent at low power level showing signs of transition to churn and/or dispersed regimes as the power level is increased. A review of pool regimes can be found in Fauske [1977]. As for the effects of pool flow regimes on the accidental sequence, it was found by Ginsberg et al. [1979] that an intense pool boiling is generally regarded as a sufficient condition to prevent transition-phase recriticality events. Hakim et al. [1981] investigated, what would happen if a layer of molten material settles on a cold container. This situation is likely to happen in an Unprotected Loss of Flow accident. It was found that a crust would develop at the interface between molten fuel and steel container, isolating the fuel pool on one side while slowly heating and melting the container itself on the other.

Pool compaction has also been investigated, since it carries the risk of recriticality events. Compaction might happen in a bottled-up pool when the heat transfer with the structures surrounding the pool is less than the internal heat generation. In this case, pressure builds up inside the pool, reducing the vapor surface velocities. On the other hand, if the heat exchange with the surroundings is higher than the heat generation, pool boiling might stop, reducing the pool volume fraction, and causing fuel compaction. Therefore, heat transfer between the pool and the surrounding structure has been investigated in the past. Chawla and Chan [1982] showed that heat transfer to the side of a boiling pool takes place as a single phase liquid boundary layer, sustained by buoyancy forces. Fuel relocation is also a key aspect, since it determines the fuel inventory available for the pool. During transition phase, fuel escape paths include: intra-subassembly pin-bundle channels, inter-subassembly gaps, and control-rod assemblies. Downfall of upper blanket pellets into the pool has an appreciable negative reactivity feedback.

The SIMMER code, developed by Yamano et al. [2003a,b], represents maybe the first and most complete attempt to simulate the transition phase. Literature is rife with SIMMER calculations of FR transition phases, see for example Maschek and Asprey [1983], Bohl [1979], Wehner and Bell [1985], Luck et al. [1981]. Because of the outstanding contributions these analyses gave to the FR community, I judged helpful to describe some of the leading phenomena discovered by them in more detail, which is provided in section 2.5.

## 2.3 Core Disassembly Phase

In this phase, the core is composed as a mixture of steel and fuel materials. A positive reactivity insertion leads to an increased power level. As more energy is produced, the fuel temperatures will further rise. The Doppler effect will try to limit the power rise. However, at some point, the fuel will vaporize, building up pressure, that will then push the core apart. This core expansion introduces a strong, negative reactivity feedback, normally sufficient to bring the core into a subcritical condition, and to stop the neutron excursion and the core disassembly phase. The disassembly is simulated by an external insertion of reactivity; the magnitude of which is estimated based on the accidental progression up to the core disassembly phase. Normally, the reactor power is evaluated via point kinetics

equations, with Doppler and core material motion effects included into the equations via appropriate coefficients. The core is represented by an homogeneous mixture of core and steel materials. Several codes, based on the mentioned assumptions, have been developed to model this phase. Jankus [1962] compared the Bethe-Tait theory with results from the computer code AX-1. Sha and Huges [1970] developed the VENUS disassembly code, a two-dimensional code coupling point kinetic neutronics and fluid-dynamics together. Henninger et al. [1979] used SIMMER to simulate this phase, too.

## 2.4 Expansion Phase

The next and last step in HCDA analysis is the evaluation of the damage to the surrounding core structures. The thermal energy released during the disassembly phase is converted into mechanical energy, and might challenge the primary system structures, figure 2.3. Vaporizing fuel expands outwards, accelerating sodium and possibly generating a sodium plug that can finally impinge onto the vessel lid. Heat can also be transferred from fuel to sodium, if this process is rapid enough, sodium may vaporize violently, further accelerating the sodium expansion towards the vessel. The simplest way to estimate the potential mechanical work,  $W$ , that an expanding mass of fuel can exert, is to evaluate the integral of the pressure  $p$  in volume:

$$W = \int_{V_1}^{V_2} p dV, \quad (2.1)$$

from the start of the expansion – i.e. from some initial volume  $V_1$  – to some pre-defined post accidental thermodynamic state – i.e. to some final volume  $V_2$ . The assumption of an isentropic expansion provides a bounding estimate of the mechanical work  $W$ , and an easy way to evaluate it: in this case, work is done at the expenses of the internal energy of the expanding fuel.

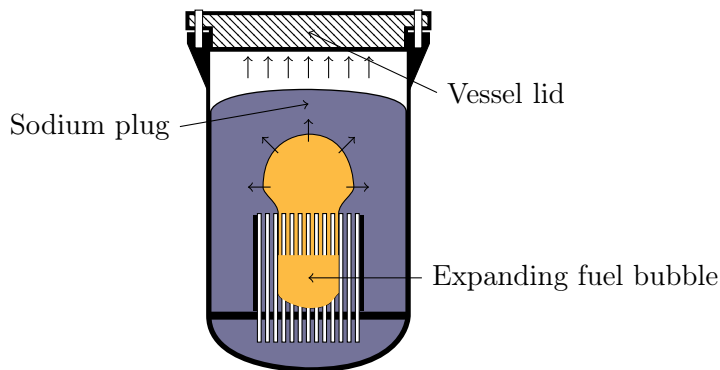


Figure 2.3: Illustration of phenomena occurring in the expansion phase of a FR HCDA, mass of expanding fuel and steel mixture accelerates surrounding sodium leading to pressure built up.

However, such an assumption is bound to provide a grossly overestimated value. Several phenomena contribute to reduce its value: friction with standing structures, in particular with the upper core structures, and heat transfer to the surroundings, just to name a few. All these mechanisms reduce the internal energy of the expanding fuel, and therefore, reduce the maximum energy available for the mechanical work. As modeling improved over

the years, a decreasing trend in estimates of the maximum mechanical work emerged. Bell et al. [1978], Theofanous and Bell [1985] are just few example of SIMMER modeling of the expansion phase. SIMMER can track the bubble of fuel while it expands and accelerates the surrounding sodium. When the sodium hits the vessel head, by knowing the velocities of the sodium, it is possible to evaluate the impact energy onto the lid.

## 2.5 Historical Background

As the comprehension of the fundamental mechanics at play in FR transients became more and more advanced, better estimations of the maximum work energy potential that can be done during HCDAs were possible. As a results, the trend has been historically a downward revision of the earlier estimations, as illustrated in figure 2.4.

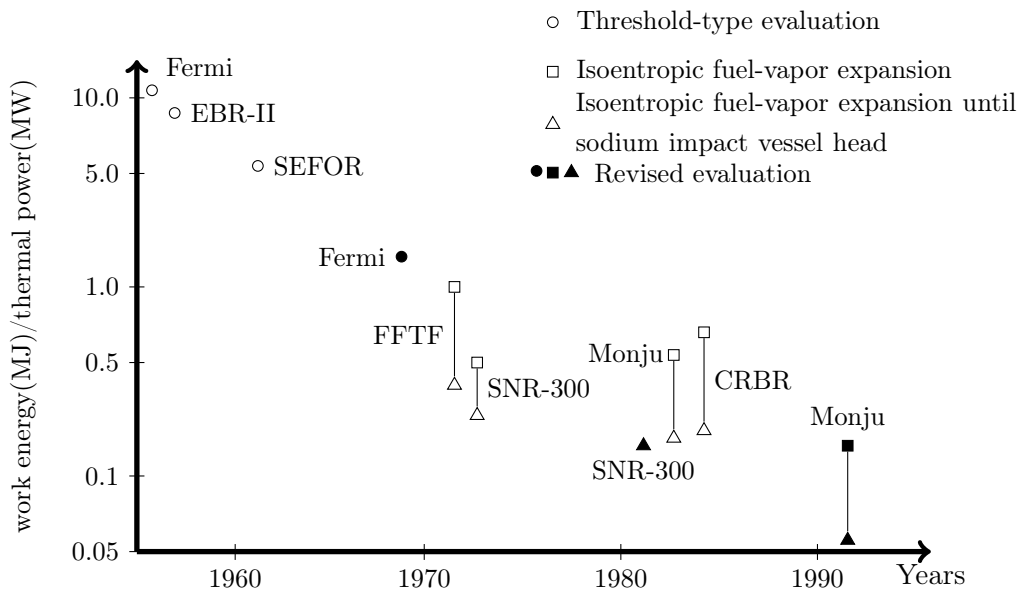


Figure 2.4: Maximum mechanical work potential for various fast reactors.

Earlier works are based on over-simplistic hypotheses. One typical mode of reactivity insertion was a core-wide collapse of the upper part of the core: it was first recognized that the loss of core structural integrity, because of melting, results in the loss of support of the upper core. Such a coherent core collapse introduces very high reactivity ramp rates, even as high as several hundreds dollars.

The energy developing in the subsequent energetic event is then evaluated based on some point kinetics model, until a pressure buildup begins to push the core apart, thus ending the power excursion. This thermal energy is then converted into mechanical energy. The first conversion approach uses an efficiency factor, that considers some analytical analyses and/or experimental data. This simple method is known as threshold approach. Such an the approach has been followed in safety assessments of earlier FR designs, such as the EBR-II, Southwest Experimental Fast Oxide Reactor (SEFOR), and Fermi Atomic Power Plant. These experimental and demonstrative facilities are described in Beyens-Henry [1966]. EBR-II was built to test the safety features of SFR design. It was a small core,

with a thermal power of 60 MW. A detailed reactor overview can be found in Loewenstein [1961]. Several ULOF tests were performed: for a detailed description of the program see Planchon et al. [1988]. The SEFOR reactor was a 20 MW SFR, with a characteristics similar to a soft spectrum FR. It was fueled with a  $\text{PUO}_2\text{-UO}_2$  mixture and was designed to measure the Doppler effect for oxide fuels. A description of the program is given in Horst [1964]. Koch et al. [1957] in their safety assessments of EBR-II assumed a ramp rate of 600\$/s. This rate was evaluated based on the following conservative hypotheses:

- no sodium is in the core,
- middle part of the core is molted and trickled at the bottom of the reactor and
- upper portion of the core falls down as a single unit.

With such a ramp rate level, the maximum mechanical work was estimated in circa ten times the reactor operational power. The authors indicate also that such high reaction rates are unlikely to happen in a real accident, they propose a more realistic mechanical work value of one tenth of the estimated value. Other works on the Fermi Atomic Plant suggest similar values. Wise et al. [1963] investigated the impact of such an energetic accident on the shield of the reactor. A more realistic work energy assessments became possible with the introduction of computer codes.

In FR, subassemblies are separated by rigid duct walls. Intact subassemblies are coupled then only through the inlet and outlet thermal-hydraulics and the reactor power. Subassemblies have therefore different power, mass flow, and temperature and, as a consequence, they tend to behave very differently from each other. This subassembly incoherence effect was understood, when the initiation phase modeling became possible, when so-called “channel” codes became available, like MELT-II [Waltar et al., 1971] and the SAS-family code [Tentner, 1985]. Such codes allow to model independent subassemblies. Subassemblies with the same power and mass flow rate are usually lumped together into a group called “channel”. All subassemblies in one channel are assumed to behave identically.

One of the earliest attempts to mechanistically model an accidental sequence was performed for the Fast Flux Test Facility (FFTF). It was a large core, with 400 MW of thermal power, cooled by sodium. Its main purpose was to test fuels and materials for the national FBR program, see Simpson et al. [1968]. Simpson et al. [1971] simulated several transient types, from TOP to LOF, with different ramp rates and flow coastdowns. The multi-channel code MELT-II developed by Waltar et al. [1971] was used to simulate the initiation phase of each accidental sequence. The modeling was stopped, when significative disassembly forces had began to push the core apart. The disassembly phase code VENUS, developed by Sha and Huges [1970], was then subsequently used to followup the accident. Advances were also made in modeling the mechanism that converts the thermal energy deposited in the fuel into mechanical work. A first modeling attempt was called “expansion model”. This approach is based on the recognition that the mechanical work is essentially done by a mass of fuel vaporizing and expanding. First models were based on isentropic fuel expansion, thus ignoring any energy dispersive phenomena, such as energy transfer to the surrounding structures. More sophisticated models were developed later. Hicks and Menzies [1965] included in their models energy transfer to the surrounding sodium. Later,

several transient thermodynamic codes were developed to model the heat transfer between molten fuel and sodium, see e.g. Padilla [1970]. This approach resulted in a further reduction of mechanical work estimates. In their FFTF analysis, using this more sophisticated expansion model, Simpson et al. [1971] showed that, out of a total of 580 MJ thermal energy deposited in molten fuel, only 150 MJ are available for mechanical work. The major outcome of these early works on reactor safety is the recognition of the fundamental role played by the sodium void reactivity during the initiation phase. The next generation of FRs was designed with the aim to have a small positive or even negative sodium void reactivity, to prevent an energetic disassembly occurring in the initiation phase. As a consequence, initial nuclear excursions are much milder: a vigorous disassembly excursion is found to be unlikely in this phase of the transient, [Jackson et al., 1974]. The most likely scenario instead is a gradual and mild core meltdown into the transition phase. One of the first reactors of this new generation was the Clinch River Breeder Reactor (CRBR): see for Jacobi and Tang [1974] for a full description. The reactor had a thermal power of 980 MW and an electric power of 380 MW. It is cooled by sodium, and the fuel is mixed Uranium-Plutonium oxide. The aim was to demonstrate a reasonable breeding capability and to prove the economic competitiveness of commercial LMFBRs. Theofanous and Bell [1985] in their CDA analysis of the reactor show that during initiation phase, energetic excursions are unlikely and a common accidental pattern has been identified:

- Sodium boiling due to high power and low mass flow rate.
- Sodium voiding of high power-to-mass ratio subassemblies,
- after voiding, clad melting results in fraction of seconds to a pin disruption

All cases show only mild energetic excursions, which increased the dispersive behavior of fuel motion. The author further identified two issues of concerns:

- so-called Loss of Flow driven Transient Overpower accident (LOF-d-TOP) and the
- role of upper plenum fission pressure.

The LOF-d-TOP is a situation typical of high sodium void reactivity reactors. In these reactors, a LOF induces a strong, positive reactivity insertion: a prompt-criticality condition is achieved with the majority of core subassemblies still wetted by the coolant. In this situation, a high power level, driven by reactivity, is accompanied by still relevant sodium flow rates. This scenario resembles more that of a typical TOP event, and that is why it is called LOF-d-TOP. However, this scenario must be avoided. High power in fact causes fuel pin failures. These pin failures probably happen near the core midplane, exhibiting the highest power densities, and movement of molten fuel inside the pin towards the location of failure drive a further positive reactivity feedback. The risk of autocatalytic behavior is then real. The authors showed that this scenario can be achieved only by unreasonable assumptions, such as neglecting the negative feedback due to axial fuel expansion, and assuming a non-dispersive fuel motion and a high sodium void reactivity (greater than 2\$ in this particular case). The second area of concern was individuated in the role of fission gases. In the upper plenum, gas pressure may arrive up to 3 MPa. Upon clad melting

and fuel failure, this pressure compacts the disrupted fuel column downwards, towards the centre. The resulting overpower causes further pin failures, and an autocatalytic effect is again possible. Based on their analysis, the authors bounded however the resulting reactivity ramp rate to 50 \$/s, a level that can barely qualify as energetic event. Incoherency effects were investigated for the SNR-300 reactor concept. The core had a thermal power of 730 MW, steam production is achieved by three steam generators, supplied by three primary circuits. Predicted electric power was 300 MW. A more detailed description of the core is found in Heusener et al. [1973]. Royl and Kuczera [1977] investigated the statistical nature of the pin failure. Several TOP accidents with different ramp rates were investigated for the SNR-300 Mark1A core. For each sequence, the standard deviation of pin failure times was evaluated. For the low ramp scenario, standard deviation of pin failure times of the order of 100 ms was evaluated. If pin failures are spread over such a large interval of time, a strong positive reactivity insertion becomes very unlikely. For mild ramp rate transients, a lower deviation of pin failure times was found, and for high ramp rate transients, it was not possible to exclude an energetic disassembly, as too many subassemblies tend to fail at the same time and coherency effects become possible.

Heusener et al. [1973] investigated several LOF sequences for the SNR-300. Their results predict that a coastdown of all primary pumps and a failure of the shut-down system do not cause energetic events that challenge the containment of the reactor. Separation of molten fuel and sodium limit the maximum work done onto the reactor containment. Only under very conservative assumptions, and during a 5\$ ramp rate insertion, the core mixes with the sodium. The resulting mechanical energy was estimated in 200 MWs, which represents a level of energetics that the containment would withstand. Investigation on a 700 MW(th)-class LMFBR in Japan by Kondo et al. [1985] showed that energetic events in the initiation phase are observed only under very conservative conditions that lead again to a LOF-d-TOP accident type. Even in that scenario, the maximum mechanical work on the surrounding was evaluated in 120 MJ, while only an energy of approximately 500 MJ would result in the failure of the reactor containment.

As the realization that a gradual core meltdown is the most probable outcome of the initiation phase, the focus of safety analysts moved gradually into the transition phase modeling. Inherent to the transition phase is the formation of a core-wide pool of molten fuel and steel. It was first recognized that several disturbances might induce recriticality events. Several modes of recriticality were individuated, according to Maschek [1982]. These can be condensed into:

- Pool collapse,
- coherent fuel motion,
- reentry of previously ejected core material and
- separation of control rod material.

Typically, disturbances, such as a mild neutronic excursion or a pressure buildup, push the liquid fuel apart, away from the center. As the fuel gathers again at the core center after hitting the lateral walls of the pool and rebounding, a critical condition can be achieved,



with a subsequent power excursion. Maschek et al. [1992] used the SIMMER code to predict a simple pool sloshing experiment. It was shown that a material accumulation due to sloshing was correctly predicted by the SIMMER code. The authors also considered that the existence of obstacles and asymmetries in a real reactor prevent a coherent pool sloshing motion and the insertion of high reactivity ramp rates.

Prior to the 80s, each of the mentioned recriticality modes were investigated assuming a reactivity ramp rate insertion and then using it as a driving factor in disassembly phase codes to compute the total mechanical energy released. Such analyses were conducted for each of the recriticality modes identified. Boudreau and Jackson [1974] considered for example a possible recriticality scenario following the return of expelled fuel for a small LMFBR. Several fuel reentry modes were investigated, coherent or incoherent reentry, and the influence of molten steel. Static neutronic analyses were used to assess ramp rates for each reentry mode. The result was a wide range of ramp rates, from 15 \$/s up to 700 \$/s, depending essentially on coherency or incoherency mode of reentry.

Mechanistic modeling of transition phase became possible only with the introduction of the SIMMER codes. Over the years, it seems that the standard route to model an accidental sequence is to use channel codes (for example the SAS code) to model the initiation phase, and then using SIMMER codes for the transition phase. The first type of this analysis was probably performed by Theofanous and Bell [1985] on CRBR. Maschek and Asprey [1983] analyzed a LOF accident for the SNR-300 core. They found that with a free fuel discharge from the active part of the core, the accident ends without formation of a bottle-up pool and with no energetic excursions. Pool formation became possible when fuel paths out of the core are blocked. Formation of fuel and steel blockages are therefore found to be one of the most important phenomena in determining the evolution of the accident. In case of pool formation, several recriticality events are unavoidable, with energy levels of the order of 1000 times the nominal reactor power. These energetic events melt a major portion of the residual standing structures with the subsequent fall of the upper blanket into the pool. Dilution of fuel with depleted uranium then terminates the transient. Fuel particle diameters upon fuel failure are found to have also a great influence on the accident. Larger fuel particles experience a lower drag force with the sodium vapor flow: fuel gravity-compaction mechanism are then faster, and the power excursion tends to happen earlier. These aspects were confirmed by analyses of a Japanese LMFBR by Tobita et al. [1999]. Parametrically studying different fuel particle sizes, and different axial fuel penetration lengths into the upper structures, they bounded the maximum mechanical energy to 16 MJ, a level unlikely to challenge the integrity of the reactor containment.

As the fuel penetration length determines the fuel inventory available to the pool, several design solutions are now being dedicated to assure a sufficient fuel removal path from the core. One solution envisages the use of a special subassembly design, in which a fraction of the hexagonal cross section is free from pins thus allowing an escape path for the molten fuel. This concept is known as FAIDUS, see for example Tobita et al. [2008]. Another option is to design control rod guide tubes increasing the fuel dispersal potential. This option was found to be efficient by Flad et al. [2013] in the safety assessment of the European Sodium Fast Reactor (ESFR) design . With no optimized control rod design, the ULOF transition phase exhibits several recriticalities events. Conversely, improving

the control rod guide tubes, shows a quiet transition phase, with no severe recriticality events. All the discussed SIMMER analyses were carried out using a two dimensional axis-symmetric r-z geometry. This geometry forces a toroidal symmetry in the computational models, which in real reactors is not necessarily justified. With increasing computational power, there has been recently a tendency to move to three-dimensional evaluation of CDAs. One of the first attempts is probably that by Yamano et al. [2009]. They analyzed a ULOF case for a large SFR using the three dimensional version of SIMMER, SIMMER-IV. The calculation showed a different fuel motion behavior compared to two dimensional cases. In the latter cases, the assumed toroidal symmetry tends to overestimate the fuel sloshing behavior. In the three-dimensional cases, instead, such a symmetric motion is hampered by obstacles and non-symmetric geometries. This results in a milder fuel sloshing and thereby milder recriticality events. The same conclusion was reached by other works, see for example Suzuki et al. [2015]. Three dimensional calculations require, however, a great amount of computational power. Yamano et al. [2009] used a SIMMER 35 x 35 x 48 computational mesh, containing almost 60000 computational nodes. The computational time was indeed enormous: two months only for modeling less than 5% of the accidental sequence. To render three-dimensional simulations of severe accidents practical, faster simulation tools are clearly needed.

# Chapter 3

## Simulation Tools

I am introducing here the various simulation tools I will be dealing with in this present work: SIMMER code for fast reactor transient analysis, VARIANT code as a nodal transport solver, KIN3D as a kinetics module for VARIANT, and PARTISN a parallel, discrete ordinates neutron transport solver.

### 3.1 Transition Phase Modeling: SIMMER Codes

Phenomena in the transition phase are very complex. Major advances in this area came, when a mechanistic calculation was possible. SIMMER was the first code to attempt a full mechanistic analysis of the transition phase. It was initiated at Argonne National Laboratory (ANL) in the 70s and has been updated ever since. Currently it stands to version three, described in Yamano et al. [2003a]. SIMMER codes represent probably the most ambitious attempt to model a core, in which an extensive material relocation (e.g. fuel and structure meltdown) occurs. It is a  $S_N$ , implicit, multi-velocity-field, multiphase, multicomponent, eulerian, fluid-dynamics code, coupled with a space- and energy- dependent neutron kinetics model. A three-dimensional version, version number four, has been also released by Yamano et al. [2003b].

I present here the general aspects of the SIMMER family of codes. I refer mainly to the SIMMER-III code mainly. However, almost all of what I am presenting here applies to the SIMMER-IV version as well. A more detailed tour of the neutronic module is provided in section 3.1.2, in order to facilitate the comprehension of the procedure adopted in coupling with the PARTISN code.

#### 3.1.1 General Description

SIMMER-III is composed of three modules: the fluid-dynamics, the neutronic, and the structure modules, as illustrated in figure 3.1. Each of them is independent of the others, and they work together simply by exchanging data.

The fluid-dynamics module evaluates fluid temperatures and masses distributions. The structure module evaluates fuel pins, clad, and can walls masses and temperatures. These informations are then processed in the neutronics module (i.e. computing a neutron flux), and a nuclear heat source is evaluated.

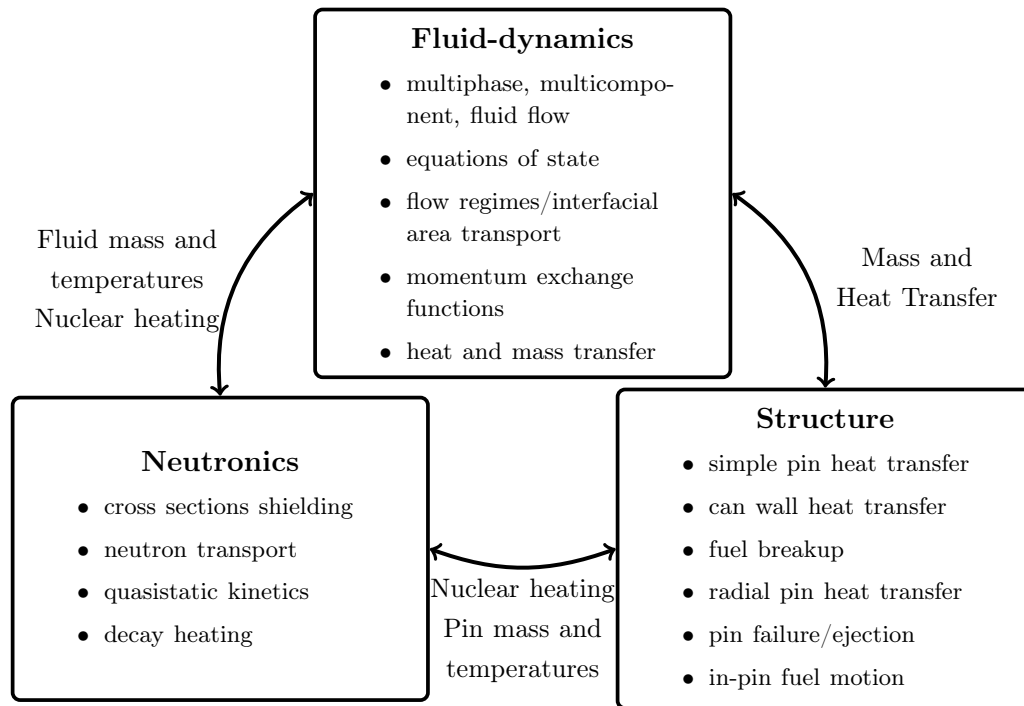


Figure 3.1: Sketch of the SIMMER code modular structure consisting of three major independent blocks.

This source is given back to the structure model, to evaluate the new temperature distribution in the fuel pins, clad, and can walls, and to the fluid-dynamics module, to evaluate new fluid temperature and flow distributions. Fluid-dynamics, heat, and neutron equations are not solved simultaneously but sequentially. For example, when the fluid-dynamics equations are solved, the neutron equations are not: the neutron flux behavior is only extrapolated in time, based on its previous history. Because of the many different time scales of the phenomena involved, equations are solved on different time steps, see figure 3.2.

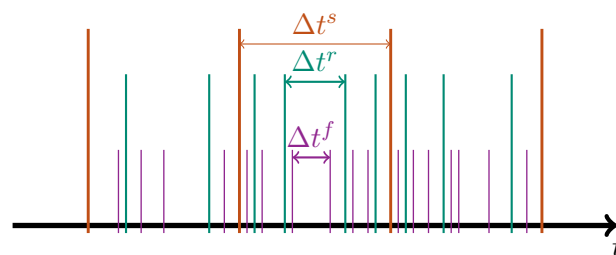


Figure 3.2: The three different time steps used in SIMMER: fluid-dynamics,  $\Delta t^f$ , reactivity,  $\Delta t^r$ , and shape,  $\Delta t^s$  steps.

On the one hand, the mean neutron lifetime in FRs is approximately  $10^{-6}$ s. Delayed neutron time is about  $10^{-1} - 10^2$ s. The fuel pin heat transfer time is assumed to be also  $10^{-6}$ s as the fuel temperature is strictly connected to the nuclear heating and hence to the neutronic time scales. Fluid-dynamics phenomena have characteristics time of the order of  $10^{-1} - 10^{-3}$ s. However, SIMMER restricts the fluid-dynamics time steps to the neutronic scale, in order to model the heat transfer between the fuel pin and the

coolant flow (as mentioned above, fuel pin temperatures are tightly connected with the nuclear power generation) and for numerical reasons. To guarantee the consistency of the computation, those time steps must be chosen carefully. Many different time steps criteria are implemented. They check both physical (e.g. excessive change in internal energy of fuel, excessive flux shape distortion) and numerical conditions (e.g. Courant condition for the fluid-dynamics equations). The basic geometry, for SIMMER-III is a  $r - z$  domain, as the one represented in figure 3.3. The domain is divided into several  $r - z$  cells. Of course, a similar arrangement is valid for  $x - y$  geometries and for three-dimensional domains.

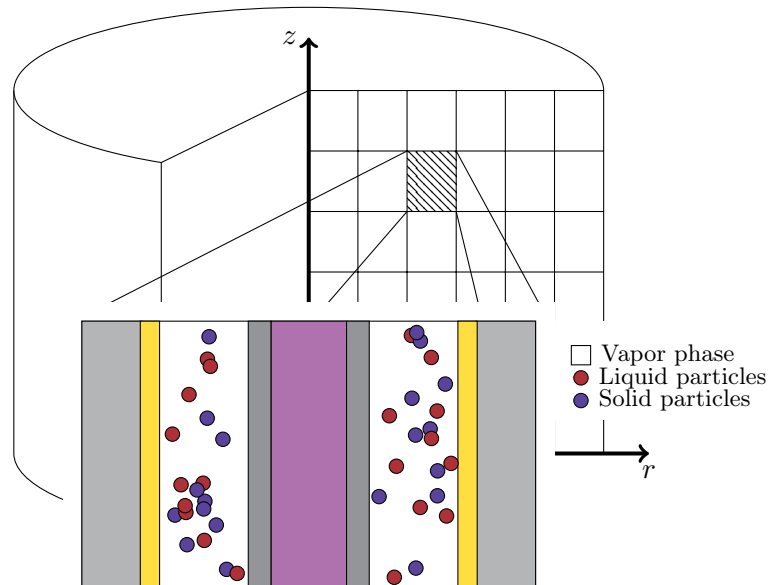


Figure 3.3: Basic SIMMER-III cell and overall geometry assignment for the reactor.

A basic SIMMER cell comprises several components, as depicted in figure 3.4: the fuel pin, its clad, and the left and the right can walls, that, together, confine the fluid flow. The fuel pin represents all the different fuel pins contained in the real subassembly. In this sense, it is an “average” fuel pin. Since fuel freezing on steel surfaces is important in the evolution of the accident, each can wall can have an additional node that represents the fuel crust. SIMMER comes with the sophisticated freezing model by Kenji [2003]. The flow can be multiphase, with both vapor and liquid phases. Solid particles may also be present in the flow, as shown in figure 3.3.

Mass, momentum, and energy conservation equations constitute the core of the fluid-dynamics module. A total of 35 different components are defined: from fertile and fissile liquid fuel, to liquid steel, liquid coolant, fuel particles, solid fuel and many more. Each of them belongs to one of three different “fields”: solid, liquid, and vapor field. To account for different enrichment zones, the fuel material is divided into a fissile and a fertile component. A control-material component, usually boron carbide ( $B_4C$ ), is added to include the neutron absorption effect of this material during the transient. A fission-gas component is also included to account for the presence of non-condensable gases in the vaporization-condensation model. For each of these components, a corresponding mass conservation equation is solved. Eight different velocity fields are available.

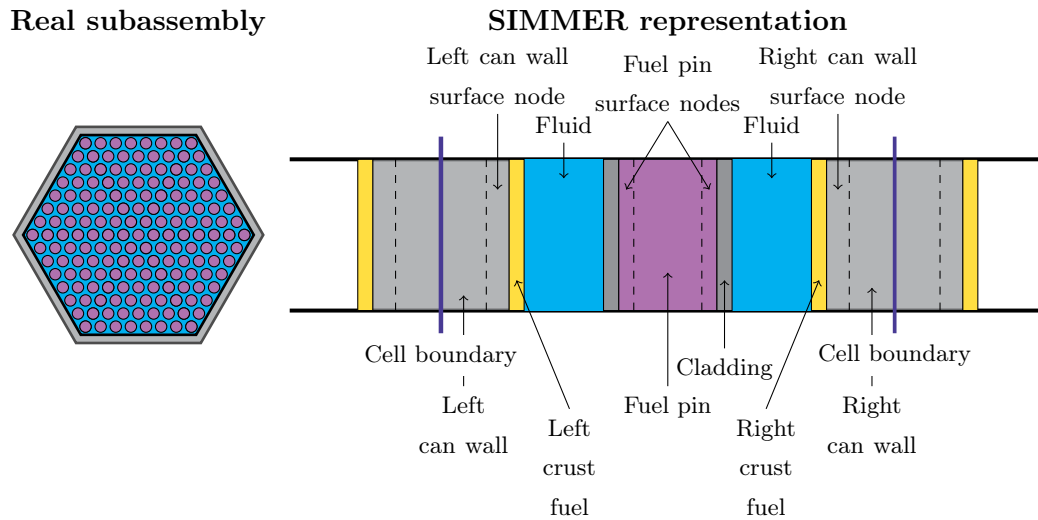


Figure 3.4: The real subassembly (left) is represented in SIMMER by a single basic cell with its components (right): a central fuel pin, left and right can walls and the flow area.

Components-to-velocity-field assignment can be chosen by the user. This freedom allows the investigation of the relative motion of the fuel with either the coolant or the steel. Seven of these momentum equations treat the dynamic behavior of the liquid components (e.g. liquid fuel, liquid steel, liquid control, and others). The last one is used to determine the behavior of the vapor field, which comprises, among others, fuel vapor, steel vapor, sodium vapor, and fission gas.

It is assumed that all these vapor field components are mixed together, so that a single velocity field is sufficient to model their dynamic behavior. A total of twenty-two energy conservation equations can be solved: as with the momentum equations, components-to-energy assignment can be selected by the user. A structure field is added to model the solid fuel, cladding, and subassembly walls (can-walls). This field is fixed in space and acts as a infinite momentum sink. SIMMER allows also the existence of solid particles into the liquid field. These can be either fuel or steel particles, resulting from molten fuel or steel freezing. For each component belonging to the structure field, a mass conservation equation is solved.

In general, a gain in a structure-field component comes from freezing and adhering of a liquid-field component; a loss comes, instead, from breakup and melting of a structure-field component. Several mass-transfer paths exist in SIMMER. So, for example, after a clad breakup, solid fuel in the structure field may begin to melt and passes from the structure fuel component into the liquid fuel component. The structure model also tracks in time the evolution of the geometric arrangement of the structure field, taking into account possible clad and fuel failures, as well as fuel crust formation onto can walls.

Equations of motion are not complete without equations of state relating densities and internal energies of the components. Equation-of-state information is required by the phase-transition model to predict the mass-transfer rates between different phases. Both analytical and tabular equation-of-state are available in SIMMER equation-of-state model, as described by Morita and Fischer [1998], Morita et al. [1999]. Thermophysical quanti-

ties, such as solid conductivities, liquid viscosities, surface tensions, and many others, are evaluated by a set of subroutines that constitutes the thermophysical module of SIMMER, details can be found in Morita et al. [2000].

### 3.1.2 SIMMER Neutronic Module

Here I give a detailed view of the neutron kinetics module. Its two main goals are: first, the evaluation of an up-to-date set of neutron cross sections, and second, the solution of the neutron kinetics equations. The actual coupling with TWODANT and THREEDANT codes was done in the past to overcome some limitations of the original neutronic scheme implemented in SIMMER. A detailed explanation of those is found in Buckel et al. [1999].

The fluid-dynamics and the neutron transport equations are partially decoupled, i.e. equations are not solved simultaneously. The neutronic state is updated at certain time steps and it is then extrapolated (based on actual and previous neutronic states) in time, while solving the fluid-dynamics equations. Cell-wise temperatures and masses are passed from the fluid-dynamics to the neutronic module. The inverse path is followed by cell-wise specific power and integral power level. The neutronic mesh is based on the fluid-dynamics mesh. Specific options allow restriction of the neutronic mesh to exclude zones where the neutron influence is deemed to be negligible, thus saving computational time. Each fluid-dynamics mesh within the neutronic domain is usually divided into sub-cells, in order to get meaningful neutronic results (e.g. avoid many negative fluxes in the  $S_N$  code).

#### Cross section model

The first task of the neutronic module is to evaluate cross sections at multiple times during the transient. These cross sections are based on masses and temperatures as they exist in the reactor at a generic time  $t$ . Cross sections are determined using the Bondarenko formalism, detailed in Segev [1975], Kidman et al. [1972]. In this procedure, the absorption, fission, scattering, and transport cross sections  $\bar{\sigma}_i$  for each isotope are evaluated by multiplying the infinitely dilute cross sections  $\sigma_i^\infty$  with a shelf-shielding factor  $f_s^i$ :

$$\bar{\sigma}_i = f_s^i \sigma_i^\infty. \quad (3.1)$$

The shielding factor  $f_s^i$  depends on the isotope temperature  $T_i$  and on a background cross section  $\sigma_{o,i}$  – i.e.  $f_s^i(\sigma_{o,i}, T_i)$  for isotope  $i$ . Because this background cross section depends on  $f_s^i$  itself, the process of determining the cross sections is iterative. During the time-dependent calculation, the background cross section is determined for each isotope for each energy group as:

$$\sigma_{o,i} = \frac{1}{N_i} \sum_{j \neq i} N_j \bar{\sigma}_{t,j}, \quad (3.2)$$

where  $N_i$  denotes the atom density for isotope  $i$  and  $\bar{\sigma}_{t,j}$  denotes the shielded total cross section for isotope  $j$ . It is clear from equation (3.2) that for an isotope  $i$ , its background cross section represents the total cross section of all the other isotopes present in the mixture, except for the very isotope  $i$ . The value of the shielding factor  $f_s^i(\sigma_{o,i}, T_i)$  for isotope  $i$  is evaluated by interpolating on a table of pre-calculated shielding factors. The effective cross section is then evaluated for each isotope according to equation (3.1). These self

shielding tables are evaluated externally and are passed to SIMMER as input data. The usual iterative procedure starts by assuming all  $f_s^i(\sigma_{o,i}, T_i)$  as one. Then, with equation (3.2), background cross sections  $\sigma_{o,i}$  for each isotope  $i$  are evaluated. Using then the tabulated shielding factors  $f_s^i(\sigma_{o,i}, T_i)$ , new shielding factors can be estimated, by interpolating these tables on the isotope temperature  $T_i$  and on the new isotope background cross section  $\sigma_{o,i}$ . These new shielding factors are then used in equation (3.1) to obtain new total cross sections  $\bar{\sigma}_{t,i}$ . These are, in turn, used in equation (3.2) to evaluate new background cross sections  $\sigma_{o,i}$ , and another interpolation on the shielding factor tables starts a new iteration. This procedure terminates when changes in shielding factors between two consecutive iterations are below a certain error, whose value can be set by the user.

### The Quasistatic approach

The second task of the neutronic module is the solution of the time-dependent neutron transport problem. The SIMMER neutronic model is based on the Improved Quasistatic Method (IQS) approach, first proposed by Henry [1958]. Sometimes, it is referred to simply as Quasistatic Method (QS). The full time-dependent transport problem to be solved is:

$$\frac{1}{v} \frac{\partial \phi}{\partial t} = \mathbf{F}_p[\phi] - \mathbf{M}[\phi] + \mathbf{S}_d[C_k] + S(\mathbf{r}, \boldsymbol{\Omega}, E, t) \quad (3.3a)$$

$$\frac{\partial C_k(\mathbf{r}, t)}{\partial t} = -\lambda_k C_k(\mathbf{r}, t) + \beta_k \int_0^\infty dE' \nu \Sigma_f(\mathbf{r}, E', t) \Phi(\mathbf{r}, E', t) \quad (3.3b)$$

where  $\mathbf{M}[\phi]$  is the transport operator,  $\mathbf{F}_p[\phi]$  is the prompt fission operator,  $\mathbf{S}_d[C_k]$  is the delayed neutron source operator, and  $\mathbf{F}_d[\phi]$  is the quasi-stationary delayed neutron source operator. They are defined as:

$$\begin{aligned} \mathbf{M}[\phi] = & \nabla \cdot \boldsymbol{\Omega} \phi(\mathbf{r}, \boldsymbol{\Omega}, E, t) + \Sigma_t(\mathbf{r}, E, t) \phi(\mathbf{r}, \boldsymbol{\Omega}, E, t) - \\ & + \int_{4\pi} d\boldsymbol{\Omega}' \int_0^\infty dE' \Sigma_s(\mathbf{r}, E' \rightarrow E, \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, t) \phi(\mathbf{r}, \boldsymbol{\Omega}', E', t) \end{aligned} \quad (3.4a)$$

$$\mathbf{F}_p[\phi] = \frac{\chi_p(E)(1-\beta)}{4\pi} \int_{4\pi} d\boldsymbol{\Omega}' \int_0^\infty dE' \nu \Sigma_f(\mathbf{r}, E', t) \phi(\mathbf{r}, \boldsymbol{\Omega}', E', t) \quad (3.4b)$$

$$\mathbf{F}_{dk}[\phi] = \frac{\chi_k^d(E) \beta_k}{4\pi} \int_{4\pi} d\boldsymbol{\Omega}' \int_0^\infty dE' \nu \Sigma_f(\mathbf{r}, E', t) \phi(\mathbf{r}, \boldsymbol{\Omega}', E', t) \quad (3.4c)$$

$$\mathbf{F}_d[\phi] = \sum_{k=1}^{\text{ID}} \mathbf{F}_{dk}[\phi] \quad (3.4d)$$

$$\mathbf{F}[\phi] = \mathbf{F}_p[\phi] + \mathbf{F}_d[\phi] \quad (3.4e)$$

$$\mathbf{S}_d[C_k] = \sum_{k=1}^{\text{ID}} \frac{\chi_k^d(E)}{4\pi} \lambda_k C_k(\mathbf{r}, t) \quad (3.4f)$$

where integer ID represents the number of delayed neutron families. It is worth mentioning that the quasi-stationary delayed neutron source  $\mathbf{F}_d[\phi]$  does not represent the actual delayed neutron source at time  $t$ . This quantity is  $\mathbf{S}_d[C_k]$ . The quasi-stationary delayed neutron source  $\mathbf{F}_d[\phi]$  represents instead the delayed neutron source that would be produced



in a stationary reactor with cross sections and fluxes as they exist at time  $t$ . The physical parameters in the definitions (3.4) are:

- $\phi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  is the angular neutron flux,
- $\Phi(\mathbf{r}, E, t) = \int_{4\pi} d\Omega \phi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  is the scalar flux,
- $S(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  is the external neutron source,
- $\Sigma_t(\mathbf{r}, E, t)$  is the total neutron cross section,
- $\Sigma_f(\mathbf{r}, E, t)$  is the fission cross section,
- $\nu$  is the number of neutrons emitted per fission event,
- $\Sigma_s(\mathbf{r}, E' \rightarrow E, \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', t)$  is the double-differential scattering cross section,
- $\chi^p(E)$  is the prompt fission spectrum,
- $\chi_k^d(E)$  is the delayed fission spectrum for delayed family  $k$ ,
- $C_k(\mathbf{r}, t)$  is the spatial neutron precursor concentration for family  $k$ ,
- $\lambda_k$  is the decay constant for delayed family  $k$ ,
- $\beta_k$  is the delayed neutron fraction for family  $k$ ,
- $\beta$  is the total delayed neutron fraction and
- $E$  is the neutron energy

The IQS method is based on the assumption that the flux  $\phi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  can be factorized into two time-depending functions: the amplitude  $N(t)$ , which is varying on a fast time scale, and the flux shape  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$ , which is varying on a slower time scale:

$$\phi(\mathbf{r}, \boldsymbol{\Omega}, E, t) = N(t) \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t). \quad (3.5)$$

As it stands, the factorization (3.5) is clearly not unique. To render it unique, another condition is imposed:

$$I = \iint \varphi^*(\mathbf{r}, E) \frac{1}{v} \varphi(\mathbf{r}, E, t) d\mathbf{r} dE = \gamma = \text{const}, \quad (3.6)$$

where  $\varphi^*(\mathbf{r}, E)$  is the stationary adjoint flux and  $\varphi(\mathbf{r}, E, t)$  is the angular-integrated shape function. Condition (3.6) is referred to as the quasistatic constraint, and means:

$$\frac{d}{dt} \gamma = 0.$$

Assumption (3.6) is not imposing that the shape  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  must be constant in time, but rather a milder condition: that the weighted integral of the shape must not change in time. Therefore, equations (3.5) and (3.6) offer an opportunity to recalculate the flux shape  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  relatively seldom compared to the amplitude  $N(t)$ .

Substituting equations (3.5) and (3.6) into the time-dependent transport problem (3.3), one gets an equation for updating the flux shape  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$ :

$$\frac{1}{v} \frac{\partial \varphi}{\partial t} + \frac{1}{v N} \frac{dN}{dt} \varphi = \mathbf{F}_p[\varphi] - \mathbf{M}[\varphi] + \frac{\mathbf{S}_d[C_k]}{N} + \frac{S}{N}. \quad (3.7)$$

Equations for the amplitude  $N(t)$  are instead obtained by multiplying equation (3.3) by the adjoint flux and taking the integral over space, angular and energy variables. That leads to the so-called point kinetics equations:

$$\frac{dN(t)}{dt} = \frac{\rho(t) - \beta^{eff}(t)}{\Lambda(t)} N(t) + \sum_{k=1}^{ID} \lambda_k c_k(t), \quad (3.8a)$$

$$\frac{dc_k(t)}{dt} = \frac{\beta_k^{eff}(t)}{\Lambda(t)} N(t) - \lambda_k c_k(t), \quad (3.8b)$$

where  $\rho(t)$  is the reactivity,  $\beta_k^{eff}(t)$  are the effective delayed neutron fractions,  $\Lambda(t)$  is the neutron generation time and  $c_k(t)$  are the effective delayed neutron precursor concentrations. Together, they are known as kinetics parameters, and are evaluated as integrals over space, energy, and angular variables:

$$\rho(t) = \frac{1}{F(t)} \langle \varphi^*, \frac{1}{k_0} \mathbf{F}[\varphi] - \mathbf{M}[\varphi] \rangle, \quad (3.9a)$$

$$\beta_k^{eff}(t) = \frac{1}{F(t)} \langle \varphi^*, \frac{1}{k_0} \mathbf{F}_{dk}[\varphi] \rangle, \quad (3.9b)$$

$$\Lambda(t) = \frac{1}{F(t)} \langle \varphi^*, \frac{1}{v} \varphi \rangle, \quad (3.9c)$$

$$c_k(t) = \frac{1}{F(t) \Lambda(t)} \langle \varphi^*, \chi_k^d C_k \rangle, \quad (3.9d)$$

where  $F(t)$  is the weighted total fission source production:

$$F(t) = \langle \varphi^*, \frac{1}{k_0} \mathbf{F}[\varphi] \rangle.$$

The multiplication factor at steady-state  $k_0$  is used in SIMMER to normalize all fission operators. The brackets  $\langle \cdot, \cdot \rangle$  signify an integration over space, energy, and angular variables. It is important to note that integrals (3.9) require the shape  $\varphi$  as a weighting function. This shape comes from solution of the shape equation (3.7). Therefore, in the QS method, first the shape equation is solved for the shape  $\varphi$ . Then, this shape is used in integrals (3.9) to evaluate the kinetics parameters. The point kinetics equations (3.8) are solved for several time steps, until a new shape  $\varphi$  is evaluated again, and the process is repeated up to the end of the transient.

### Initialization of spatial neutronics

Initialization of space neutronics is complicated by the fact that, usually, SIMMER is used to investigate the transition phase of an accident: a phase, in which material movement and core disruption is likely to occur. The initiation phase of the accident involves no core disruption and, therefore, a point kinetics model is usually employed to evaluate this first phase of the accident up to pin failures. As a result, SIMMER starts from a transient reactor state. From the point kinetics modeling of the initiation phase, amplitude  $N(t)$ , effective precursor concentrations  $c_k(t)$ , reactivity  $\rho(t)$ , effective delayed neutron fractions  $\beta_k^{eff}(t)$ , and the neutron generation time  $\Lambda(t)$  are all known. These quantities

are now inputs for SIMMER. However, no shape fluxes  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  and spatial precursor concentrations  $C_k(\mathbf{r}, t)$  are available. These variables are linked to the input quantities by equations (3.9). Therefore, at the start of the SIMMER computation, we are in a situation, in which we know the left-hand sides of equations (3.9), but not the right-hand sides.

Initial shape flux and spatial precursor concentrations are evaluated by solving equations (3.7) and (3.3b) at time equal zero. As a simplification, the time derivative of the shape is assumed to be zero. Time derivative of the amplitude is instead available as an input from the previous point kinetics modeling of the initiation phase. Spatial precursor concentrations are assumed to be proportional to the delayed neutron fission source. With these assumptions, equations (3.7) and (3.3b) can be solved for the initial shape and precursor concentrations. However, if we use these quantities to evaluate the kinetics parameters with equations (3.9), we will not get the prescribed input values. To deal with this situation, data adjustment factors for the effective delayed neutron fractions  $h_k$ , for the group velocities  $f_v$ , and for the spatial precursor concentrations  $g_k$  are defined. Correspondingly, the equations (3.9) are modified:

$$\rho(t) = \frac{1}{F(t)} \langle \varphi^*, \frac{1}{k_0} (\mathbf{F}_p[\varphi] + \sum_k^{\text{ID}} h_k \mathbf{F}_{dk}[\varphi]) - \mathbf{M}[\varphi] \rangle, \quad (3.10a)$$

$$\beta_k^{eff}(t) = \frac{1}{F(t)} \langle \varphi^*(\mathbf{r}, E), \frac{h_k}{k_0} \mathbf{F}_{dk}[\varphi] \rangle, \quad (3.10b)$$

$$\Lambda(t) = \frac{1}{F(t)} \langle \varphi^*, \frac{f_v}{v} \varphi \rangle, \quad (3.10c)$$

$$c_k(t) = \frac{1}{F(t) \Lambda(t)} \langle \varphi^*, g_k \chi_k^d C_k \rangle, \quad (3.10d)$$

$$F(t) = \langle \varphi^*, \frac{1}{k_0} (\mathbf{F}_p[\varphi] + \sum_k h_k \mathbf{F}_{dk}[\varphi]) \rangle. \quad (3.10e)$$

These data adjustment factors are evaluated so that the calculated kinetics parameters match the prescribed input ones. As can be seen from equations (3.10),  $h_k$  factors act on the delayed fission source shape operators to match the initial values of  $\rho(t)$  and  $\beta(t)$ . The  $f_v$  factor acts instead on the velocities, while the  $g_k$  factors multiply the spatial delayed neutron precursor concentrations so to match the input  $c_k(t)$ . Evaluation of these factors is done during the initial shape calculation.

Rigorously, the adjoint flux  $\varphi^*$  must be the adjoint for the stationary reactor state, which, again, is not known. However, since small perturbations usually occur in the initiation phase of the transient, the use of the adjoint  $\varphi^*$  calculated from the material and temperature distributions provided by the initiation phase modeling is usually a good approximation. After the adjoint and the initial shape calculations, the usual normalization is used: the flux shape is normalized to match the reactor power and the adjoint flux,  $\varphi^*$ , is normalized so that  $\gamma = 1.0$  in (3.6).

### Calculational flow

As already mentioned above, fluid-dynamics, neutronic, and heat equations are not solved simultaneously. Three different time steps are used, as illustrated in figure 3.2. The

smallest time step is the fluid-dynamics time step. Fluid-dynamics equations are solved on this fine time mesh. Solution of the fluid-dynamics equations requires the spatial power distribution, i.e. the amplitude  $N(t)$  and the scalar flux shape  $\varphi(\mathbf{r}, E, t)$ . These quantities are available only for the previous time steps. The solution adopted is then to extrapolate them in time, while solving the fluid motion. This extrapolation is based on the previous neutronic history. The SIMMER calculational flow is shown in figure 3.5.

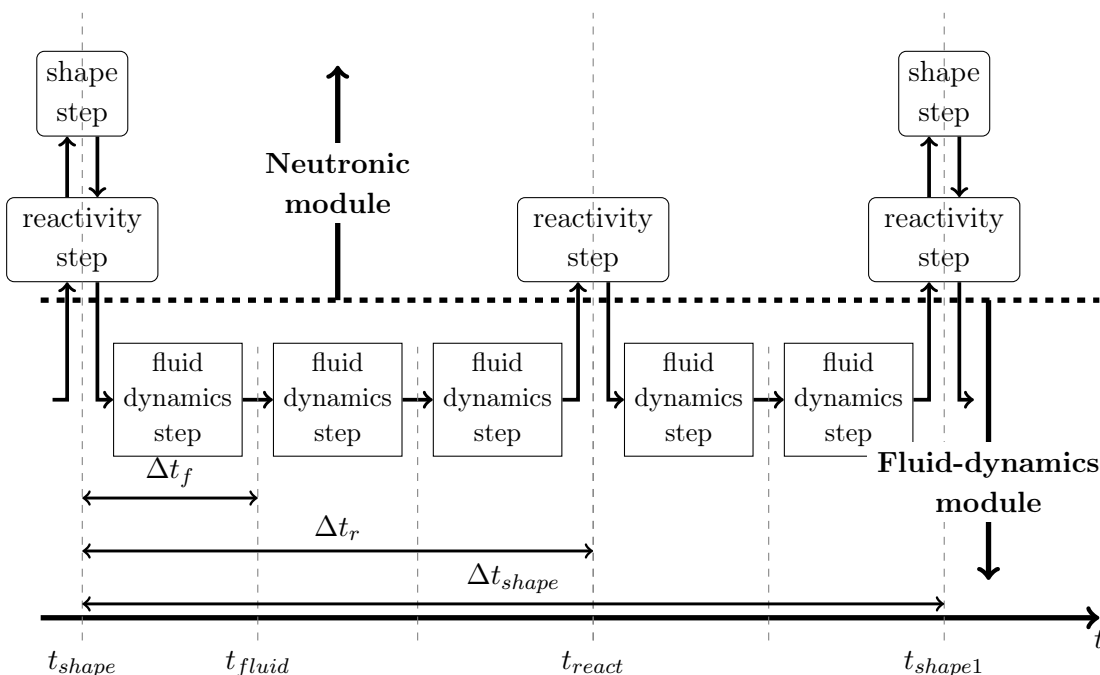


Figure 3.5: Sequence of shape, reactivity, and fluid-dynamics steps during a SIMMER computation. Fuel pin heat transfer time step are done at each reactivity step.

Suppose we have just evaluated a new shape  $\varphi(t_{shape})$  at a generic time  $t_{shape}$ . Several fluid-dynamics steps are performed first. During these steps, the code performs several checks. When one of these checks is satisfied, then the neutronic state must be updated: either a reactivity or a shape step is performed. Suppose a reactivity step is performed at time  $t_{react}$ . At this time, the instantaneous mass and temperature distributions evaluated in the fluid-dynamics steps are passed to the neutronic module. The reactivity step is controlled by the neutronic module, which does the following steps:

- it evaluates a new set of cross-sections, based on masses and temperatures,
- it evaluates the kinetics parameters (3.9),
- it solves the amplitude equation and
- extrapolates the up-to-date amplitude to the next reactivity step

Solution of the amplitude equation is performed on the smallest time step, the amplitude time step. A great level of sophistication is involved in solving for the amplitude. Kinetics parameters are in fact not considered constant within a reactivity step, but assumed to vary

in time with a second degree polynomial. The coefficients of this parabola are evaluated by using values at the current time point and at two preceding ones. It is worth mentioning also, that during these reactivity and fluid-dynamics steps, the flux shape is constant and equal to the shape evaluated at the start of the shape step, that is  $\varphi(t_{shape})$ .

There is, however, a little problem. At this point, the new amplitude  $N(t_{react})$  coming from solution of point kinetics equations (3.8) will certainly differ from the extrapolated amplitude used in the previous fluid-dynamic steps. A rigorous treatment of this inconsistency would require an iterative procedure. However, it is deemed too expensive to re-do the fluid-dynamics steps. A simpler solution is adopted: a correction term is taken into account while extrapolating the amplitude to the next reactivity step. After several reactivity steps, it may be necessary to update the flux shape at time  $t_{shape1}$ . Material motion and temperature changes may alter the flux shape. Whenever a shape update is necessary, the neutronic module performs first a reactivity step, and then solves the shape equation (3.7). Computationally speaking, this is the most heavy part of the neutronic module. After a new shape  $\varphi(t_{shape1})$  is obtained, kinetics parameters are re-evaluated using this new shape, a new amplitude at this time point  $N(t_{shape1})$  is computed and extrapolated for the next round of fluid-dynamics steps.

### Gamma iteration

Among the numerous specificities of SIMMER, one worth mentioning is the so called gamma iteration, or  $\gamma$  iteration. It is an iterative procedure, inherently linked with the QS method, and devised for improving the consistency between the amplitude  $N(t)$  and the shape  $\varphi(\mathbf{r}, \mathbf{\Omega}, E, t)$ . At the end of each shape step, a new shape flux is evaluated by solving the flux shape equation (3.7). In the QS scheme, the flux shape is assumed fixed within a shape step, and the kinetics parameters are evaluated using this fixed shape. However, in reality, the shape changes within a shape step, although mildly. Because shape variations can be taken into account only at the end of the shape step (that is, when a new shape equation is solved), the kinetics parameters evaluated within a shape step are therefore an approximation, and, as a consequence, so is the amplitude  $N(t)$ . Since the shape equation has a term that involves the amplitude itself (the time absorption term, see equation (3.7)), using an inaccurate amplitude when solving the shape equation might lead to a different shape compared to the real shape. This deviation is easily verified by evaluating the value of quasistatic constraint – i.e.  $\gamma$  – with the new flux shape. If the new  $\gamma$  is far from one, than this is an indication that the new flux shape is not fully correct.

To solve this problem, an iterative procedure is used. All the reactivity steps within a shape step are evaluated again – i.e. by solving for the point kinetics amplitude  $N(t)$ . But this time, the kinetics parameters  $\rho(t)$ ,  $\beta_k^{eff}(t)$ ,  $\Lambda(t)$  are corrected. This correction is built from both the initial and the final shape. That is to say, kinetics parameters now feel the effect that the shape is changing within a shape step. After reactivity steps are done – i.e. at the end of the shape step – a new flux shape is evaluated again. This newly-evaluated shape should be more close with the real shape, and the  $\gamma$  value would be closer to one than at the previous iteration. This procedure is repeated until the quasistatic constraint (3.6) at the end of the shape step is satisfied within an user-defined criteria.

### Some programming informations

Data exchange between fluid-dynamics and neutronic module is ensured both via external binary files and with `COMMON` blocks. In the first case, a `SIMMER` module – i.e. `LINKM` – writes the needed input files, while another module – in the `TWODANT` part of the code – reads them. Geometry information and cross sections are passed this way, using external binary files. Scalar and angular neutron fluxes are also passed this way. A schematic view of the procedure is shown in figure 3.6. The scalar fluxes are needed to compute the cell-wise specific power, while the angular fluxes are requested by the `QS` scheme to treat rigorously the time derivative of the flux shape.

Other informations are passed directly via `COMMON` blocks. Among them, the most important ones are the delayed neutron precursor concentrations and the neutron currents. The first ones are needed to evaluate the delayed neutron source, as in equation (3.7). Neutron currents are instead required by the point kinetics module to evaluate the leakage component of the reactivity.

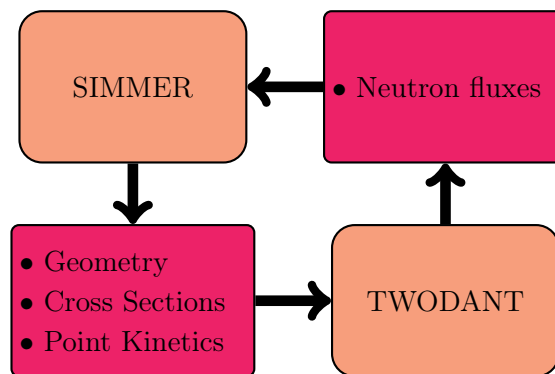


Figure 3.6: Exchange of data between fluid-dynamics and neutronics module.

### 3.1.3 Once-Through SIMMER Analysis

The standard route to compute an accidental sequence has been to use specific channel codes – for example like `SAS4A` – to model the initiation phase, and then have `SIMMER` simulate later accidental phases. The usual approach is to follow initiation phase up to some time point after pin failures.

By that time, fuel is discharged into the coolant channel, and, as a result, significant distortion in the neutron distribution are to be expected – especially if can walls melt and fuel can travel radially. In this scenario, channel-code-modeling assumptions become questionable – specifically the use of a point kinetics model. The solution has been to use `SIMMER` from that time on. Code coupling has been historically a source of problems. First of all, selection of the coupling point is somewhat arbitrary. It is up to the user to deem when `SIMMER` should take over the modeling. Different codes are also based on different assumptions, and use different solution techniques: in some cases, a smooth transition to `SIMMER` was not possible at all. All these reasons have spurred efforts to extend `SIMMER`'s range of applicability, with the aim to model the whole accidental sequence: from steady-state up to accident termination. This `SIMMER` “once-through”

approach presents several challenges though. One of them is the correct modeling of thermal expansion reactivity feedbacks. This is one of the main driver during the initiation phase, and it determines the later evolution of the transient. As of now, SIMMER totally ignores any reactivity contribution due to thermal expansion. Implementation of such effects can be also problematic due to geometric code constrains, which is based on the use of a structured Eulerian mesh.

## 3.2 PARTISN

PARTISN is a code designed to solve the neutron transport equation. It is developed by Los-Alamos National Laboratory (LANL), see Alcouffe and Baker [2009] and provides support for multi-processor computations via the MPI standard.

It is organized in modules. Three modules separate input, solution, and edit part of a computation. These modules are independent of each other, and data are passed by means of external binary files.

The code deals with one-, two-, and three-dimensional domains, can solve both the static and the time-dependent transport equations using a standard multigroup approach to treat the energy variable and a discrete ordinate method for the angular dependence.

### 3.2.1 Iteration Strategy and Diffusion Synthetic Acceleration

PARTISN iteration strategy employs the so-called Diffusion Synthetic Acceleration (DSA), as described by Alcouffe [1977], Alcouffe and Baker [2009]. The iteration procedure starts with a multigroup diffusion calculation:

$$\begin{aligned} -\nabla \cdot D_g(\mathbf{r})\nabla \Phi_g^{k+1}(\mathbf{r}) + \Sigma_{R,g}(\mathbf{r}) \Phi_g^{k+1}(\mathbf{r}) = \\ = \frac{\chi_g^t}{k_0} F^k(\mathbf{r}) + \sum_{g'=1}^{g-1} \Sigma_{s,gg'}(\mathbf{r}) \Phi_{g'}^{k+1}(\mathbf{r}) + \sum_{g'=g+1}^G \Sigma_{s,gg'} \Phi_{g'}^k(\mathbf{r}), \end{aligned} \quad (3.11)$$

where  $g$  is the energy group,  $G$  is the total number of energy groups,  $\Sigma_{R,g}(\mathbf{r})$  is the removal cross section for energy group  $g$ ,  $D_g(\mathbf{r})$  is the group diffusion coefficient,  $\chi_g^t$  is the total fission spectrum, and  $F^k(\mathbf{r})$  is the fission source spatial distribution at iteration  $k$ :

$$F^k(\mathbf{r}) = \sum_{g'=1}^G (\nu \Sigma_f(\mathbf{r}))_{g'} \Phi_{g'}^k(\mathbf{r}). \quad (3.12)$$

A guessed flat flux is used for the first iteration, i.e.  $k = 1$ . Solution of equation (3.11) is itself an iterative procedure. When the scalar fluxes  $\Phi_g^{k+1}(\mathbf{r})$  for each energy group are evaluated, a new fission source  $F^{k+1}(\mathbf{r})$  is evaluated with equation (3.12) and used to start a new iteration. This iterative procedure stops when flux convergence is deemed satisfactory. Each of these iterations is called diffusion sub-outer. After that, the next outer iteration begins with the so-called inner iteration, in which the following transport problem is solved – isotropic scattering is used here just to simplify the notation –

$$\Omega \cdot \nabla \tilde{\Psi}_g^l(\mathbf{r}, \Omega) + \Sigma_g(\mathbf{r}) \tilde{\Psi}_g^l(\mathbf{r}, \Omega) = \Sigma_{s,gg}(\mathbf{r}) \Phi_g^{l-1}(\mathbf{r}) + S_g(\mathbf{r}), \quad (3.13)$$

where  $\tilde{\Psi}_g^l(\mathbf{r}, \boldsymbol{\Omega})$  is the transport angular flux for group  $g$  at inner iteration number  $l$ . The tilde is used in the following to indicate quantities evaluated using the angular transport fluxes. Note that in equation (3.13) the scalar flux  $\Phi_g^{l-1}(\mathbf{r})$  is coming from the previous inner iteration (i.e.  $l-1$ ). The neutron source  $S_g(\mathbf{r})$  contains the fission contribution (this term comes from the last diffusion sub-outer iteration, and, therefore, is fixed during the inner iterations), and the in-scattering source from all energy groups  $g'$  other than group  $g$ . Solution of equation (3.13) is done using the standard discrete ordinate approach. A neutron direction is selected. Then, starting from the boundaries, angular fluxes  $\tilde{\Psi}_g^l(\mathbf{r}, \boldsymbol{\Omega})$  are evaluated for each spatial node following the selected direction. This procedure is called transport sweep.

When the angular fluxes  $\tilde{\Psi}_g^l(\mathbf{r}, \boldsymbol{\Omega})$  for all angular directions are known, the scalar transport fluxes  $\tilde{\Phi}_g^l(\mathbf{r})$  are evaluated, and accelerated using the DSA technique. This procedure involves solution of a modified diffusion-like equation. Three different correction schemes are used: the source correction, the diffusion correction and the removal correction.

In the source correction scheme, a correction term  $R^l(\mathbf{r})$  modifies the neutron source. The diffusion equation solved is:

$$-\nabla \cdot D_g(\mathbf{r}) \nabla \Phi_g^{l+1}(\mathbf{r}) + \Sigma_{R,g}(\mathbf{r}) \Phi_g^{l+1}(\mathbf{r}) = S_g(\mathbf{r}) - R_g^l(\mathbf{r}) \quad (3.14)$$

where the correction term is evaluated using the transport angular fluxes as

$$R_g^l(\mathbf{r}) = \nabla \cdot \tilde{\mathbf{J}}_g^l(\mathbf{r}) + \nabla \cdot D_g(\mathbf{r}) \nabla \tilde{\Phi}_g^l(\mathbf{r}), \quad (3.15)$$

where:

$$\begin{aligned} \tilde{\Phi}_g^l(\mathbf{r}) &= \int_{4\pi} d\Omega \tilde{\Psi}_g^l(\mathbf{r}, \boldsymbol{\Omega}), \\ \tilde{\mathbf{J}}_g^l(\mathbf{r}) &= \int_{4\pi} d\Omega \boldsymbol{\Omega} \tilde{\Psi}_g^l(\mathbf{r}, \boldsymbol{\Omega}). \end{aligned}$$

If the procedure converges, it converges to the transport solution.

When equation (3.14) is solved, the new scalar flux  $\Phi_g^{l+1}(\mathbf{r})$  is used again in equation (3.13), and a new inner iteration is performed. The inner iterations stop, when some convergence criteria are met.

When the inner iterations are converged for each energy group, new scalar fluxes  $\Phi_g^l(\mathbf{r})$  are available, and a new fission source  $F^k(\mathbf{r})$  is evaluated and used to start the first iteration of the sub-outer part of the algorithm. A modified multigroup diffusion equations is then solved:

$$\begin{aligned} -\nabla \cdot D_g(\mathbf{r}) \nabla \Phi_g^{k+1}(\mathbf{r}) + \Sigma_{R,g}(\mathbf{r}) \Phi_g^{k+1}(\mathbf{r}) &= \\ = \chi_g^t F^k(\mathbf{r}) + \sum_{g'=1}^{g-1} \Sigma_{s,gg'}(\mathbf{r}) \Phi_{g'}^{k+1}(\mathbf{r}) + \sum_{g'=g+1}^G \Sigma_{s,gg'} \Phi_{g'}^k(\mathbf{r}) + S_g(\mathbf{r}) - R_g^k(\mathbf{r}). \end{aligned} \quad (3.16)$$

This source correction scheme is effective for inhomogeneous problems. For eigenvalue problems, the diffusion scheme is used. The idea behind this scheme is simple. A new



diffusion coefficient  $\bar{D}_g^l(\mathbf{r})$  is evaluated by using the scalar fluxes  $\tilde{\Phi}_g^l(\mathbf{r})$  and neutron currents  $\tilde{\mathbf{J}}_g^l(\mathbf{r})$  coming from the solution of the transport problem:

$$\bar{D}_g^l(\mathbf{r}) = -\frac{\tilde{\mathbf{J}}_g^l(\mathbf{r})}{\nabla \tilde{\Phi}_g^l(\mathbf{r})}, \quad (3.17)$$

and then accelerating the solution by solving the diffusion equation:

$$-\nabla \cdot \bar{D}_g^l(\mathbf{r}) \nabla \Phi_g^{l+1}(\mathbf{r}) + \Sigma_{R,g}(\mathbf{r}) \Phi_g^{l+1}(\mathbf{r}) = S_g(\mathbf{r}). \quad (3.18)$$

The outer iteration proceeds with the multigroup diffusion calculation (sub-outers) where the diffusion coefficient is substituted with the corrected one (3.17):

$$\begin{aligned} -\nabla \cdot \bar{D}_g(\mathbf{r}) \nabla \Phi_g^{k+1}(\mathbf{r}) + \Sigma_{R,g}(\mathbf{r}) \Phi_g^{k+1}(\mathbf{r}) &= \\ &= \frac{\chi_g^t}{k_0} F^k(\mathbf{r}) + \sum_{g'=1}^{g-1} \Sigma_{s,gg'}(\mathbf{r}) \Phi_{g'}^{k+1}(\mathbf{r}) + \sum_{g'=g+1}^G \Sigma_{s,gg'} \Phi_{g'}^k(\mathbf{r}). \end{aligned} \quad (3.19)$$

The last correction scheme is the removal correction scheme. It is analogous with the previous scheme, provided that the removal cross section is corrected, instead of the diffusion coefficient. The new removal cross section is defined as:

$$\tilde{\Sigma}_{R,g}^l(\mathbf{r}) = \Sigma_{R,g}(\mathbf{r}) + \frac{R_g^l(\mathbf{r})}{\tilde{\Phi}_g^l(\mathbf{r})}. \quad (3.20)$$

The fluxes are accelerated solving then the modified diffusion equation:

$$-\nabla \cdot D_g(\mathbf{r}) \nabla \Phi_g^{l+1}(\mathbf{r}) + \tilde{\Sigma}_{R,g}^l(\mathbf{r}) \Phi_g^{l+1}(\mathbf{r}) = S_g(\mathbf{r}), \quad (3.21)$$

and the sub-outers diffusion iterations are:

$$\begin{aligned} -\nabla \cdot D_g(\mathbf{r}) \nabla \Phi_g^{k+1}(\mathbf{r}) + \tilde{\Sigma}_{R,g}^l(\mathbf{r}) \Phi_g^{k+1}(\mathbf{r}) &= \\ &= \frac{\chi_g^t}{k_0} F^k(\mathbf{r}) + \sum_{g'=1}^{g-1} \Sigma_{s,gg'}(\mathbf{r}) \Phi_{g'}^{k+1}(\mathbf{r}) + \sum_{g'=g+1}^G \Sigma_{s,gg'} \Phi_{g'}^k(\mathbf{r}). \end{aligned} \quad (3.22)$$

### 3.3 VARIANT: A Nodal Neutron Transport Solver

Nodal methods have been highly successful in reducing the cost of large multidimensional neutron diffusion and transport calculations by using coarse spatial mesh grids without a commensurate increase in computational effort per node as shown by Lawrence [1986].

They begin with a statement of a nodal neutron balance, and employ transverse integration procedures to obtain approximate quasi-one-dimensional equations. While highly successful in obtaining fast, coarse-mesh solutions, these approaches have been confounded to some extent by the complexity of the space-angle coupling present in the transport equation. Another approach to nodal transport methods is based on the so-called Variational Nodal Method (VNM), as described by Dilber and Lewis [1985], Carrico et al. [1992]. The defining feature of the method is a variational principle for the even-parity form of the transport equation, which guarantees neutron conservation for each node. This

variational formulation allows computational algorithms to be derived using the classical Ritz procedure: known trial functions in space and angle are used to approximate the neutron flux within each node and to obtain sets of linear algebraic equations for each node. This method has been implemented into the FORTRAN module called VARIANT by Palmiotti et al. [1995]. VARIANT is part of the DIF3D code system developed by Derstine [1984]. Recently, VARIANT has been embedded into the ERANOS code package system, described in Doriath [1993].

VARIANT performs multigroup neutron transport calculations in both cartesian and hexagonal geometries in two and three dimensions. Both direct and adjoint calculations may be performed. Spherical harmonics are employed to treat the angular dependence of the flux, actually  $P_1$ ,  $P_3$  and  $P_5$  approximations are implemented in all geometries, and include both within-group and group-to-group anisotropic scattering. The spatial dependence of the flux is represented by complete polynomials within node volumes, and along inter-node surfaces. Polynomials up to sixth order for both cartesian and hexagonal geometries are implemented.

Up to now, the VNM has been formulated under the assumption that each node is homogeneous. However, this is not a fundamental restriction, since the Ritz procedure, by which the equations are derived, can, in principle, accommodate changes of cross sections within the nodes.

### 3.3.1 The Variational Nodal Method

In this section, the theory behind the VNM is described. I follow the notation used by Palmiotti et al. [1995]. For simplicity, the problem here is formulated only with isotropic scattering. The starting point is the within-group transport equation with isotropic scattering and source:

$$\boldsymbol{\Omega} \cdot \nabla \Psi_g(\mathbf{r}, \boldsymbol{\Omega}) + \Sigma_{t,g}(\mathbf{r}) \Psi_g(\mathbf{r}, \boldsymbol{\Omega}) = \int_{4\pi} d\Omega' \Sigma_{s,g}(\mathbf{r}) \Psi_g(\mathbf{r}, \boldsymbol{\Omega}') + S_g(\mathbf{r}), \quad (3.23)$$

where  $\Sigma_{t,g}(\mathbf{r})$  is the total cross section,  $\Sigma_{s,g}(\mathbf{r})$  is the within-group scattering cross section,  $\Psi_g(\mathbf{r}, \boldsymbol{\Omega})$  represents the angular flux,  $S_g(\mathbf{r})$  is the group source,  $\mathbf{r}$  and  $\boldsymbol{\Omega}$  are the neutron position and direction of travel.

The definitions of even-  $\Psi_g^+(\mathbf{r}, \boldsymbol{\Omega})$  and odd-parity  $\Psi_g^-(\mathbf{r}, \boldsymbol{\Omega})$  fluxes are:

$$\Psi_g^+(\mathbf{r}, \boldsymbol{\Omega}) = \frac{1}{2} [\Psi_g(\mathbf{r}, \boldsymbol{\Omega}) + \Psi_g(\mathbf{r}, -\boldsymbol{\Omega})], \quad (3.24)$$

$$\Psi_g^-(\mathbf{r}, \boldsymbol{\Omega}) = \frac{1}{2} [\Psi_g(\mathbf{r}, \boldsymbol{\Omega}) - \Psi_g(\mathbf{r}, -\boldsymbol{\Omega})]. \quad (3.25)$$

From this point on and for clarity, I will drop the subscript  $g$ : fluxes and cross sections are to be intended as energy group quantities.

To formulate the problem variationally, I must obtain the even-parity equation with isotropic scattering and sources. This is accomplished by evaluating equation (3.23) at  $\boldsymbol{\Omega}$  and at  $-\boldsymbol{\Omega}$ , adding the results together, and then multiplying by one half:

$$\boldsymbol{\Omega} \cdot \nabla \Psi^-(\mathbf{r}, \boldsymbol{\Omega}) + \Sigma_t(\mathbf{r}) \Psi^+(\mathbf{r}, \boldsymbol{\Omega}) = \Sigma_s(\mathbf{r}) \int_{4\pi} d\Omega' \Psi^+(\mathbf{r}, \boldsymbol{\Omega}') + S(\mathbf{r}). \quad (3.26)$$

Likewise, evaluation of equation (3.23) at  $\mathbf{\Omega}$  and  $-\mathbf{\Omega}$  and a subsequent subtraction yields an equation that relates the odd and even components of the neutron flux:

$$\mathbf{\Omega} \cdot \nabla \Psi^+(\mathbf{r}, \mathbf{\Omega}) + \Sigma_t(\mathbf{r}) \Psi^-(\mathbf{r}, \mathbf{\Omega}) = 0. \quad (3.27)$$

The even parity equation:

$$-\mathbf{\Omega} \cdot \nabla \Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi^+(\mathbf{r}, \mathbf{\Omega}) + \Sigma_t \Psi^+(\mathbf{r}, \mathbf{\Omega}) = \Sigma_s \Phi(\mathbf{r}) + S(\mathbf{r}) \quad (3.28)$$

is then obtained by using equation (3.27) to express the odd-parity flux  $\Psi^-(\mathbf{r}, \mathbf{\Omega})$  in terms of the even-parity  $\Psi^+(\mathbf{r}, \mathbf{\Omega})$  as:

$$\Psi^-(\mathbf{r}, \mathbf{\Omega}) = -\Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi^+(\mathbf{r}, \mathbf{\Omega}), \quad (3.29)$$

and then eliminating it from (3.26). Angular integration of the even-parity flux  $\Psi_g^+(\mathbf{r}, \mathbf{\Omega})$  has a physical significance. It is not difficult to show, that the result represents the scalar flux  $\Phi(\mathbf{r})$ :

$$\Phi(\mathbf{r}) = \int_{4\pi} d\Omega \Psi^+(\mathbf{r}, \mathbf{\Omega}). \quad (3.30)$$

Multiplying the odd-parity flux  $\Psi^-(\mathbf{r}, \mathbf{\Omega})$  by  $\mathbf{\Omega}$  and integrating over the angle, leads instead to the neutron current  $\mathbf{J}(\mathbf{r})$ :

$$\mathbf{J}(\mathbf{r}) = \int_{4\pi} d\Omega \mathbf{\Omega} \Psi^-(\mathbf{r}, \mathbf{\Omega}). \quad (3.31)$$

The even-parity transport equation (3.28) may be formulated as a variational principle in terms of a global functional  $F$ , which is a superposition of volume and surface contributions  $F_\kappa$  from  $\kappa$  spatial nodes and  $\gamma$  nodal interfaces comprised in the problem domain:

$$F[\Psi^+, \Psi^-] = \sum_{\kappa} F_\kappa[\Psi^+, \Psi^-]. \quad (3.32)$$

For clarity, the position and the angular dependence of  $\Psi^+(\mathbf{r}, \mathbf{\Omega})$  and  $\Psi^-(\mathbf{r}, \mathbf{\Omega})$  will be dropped in the following. The contribution from each node  $\kappa$  is:

$$F_\kappa[\Psi^+, \Psi^-] = \int_{\kappa} dV \left\{ \int_{4\pi} d\Omega \left[ \Sigma_t^{-1} (\mathbf{\Omega} \cdot \nabla \Psi^+)^2 + \Sigma_t (\Psi^+)^2 \right] + \right. \\ \left. - \Sigma_s \Phi^2 - 2\Phi S \right\} + 2 \int_{\gamma} d\Gamma \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_\gamma \Psi^+ \Psi^-, \quad (3.33)$$

where  $\gamma$  indicates a generic node surface,  $\mathbf{n}_\gamma$  its outward directed normal vector, and  $d\Gamma$  an infinitesimal surface. Requiring this functional to be stationary with respect to variations in  $\Psi^+$  and  $\Psi^-$  leads to the even-parity equation within each node, and to the continuity of both even-  $\Psi^+$  and odd-  $\Psi^-$  parity fluxes across node interfaces.

Suppose  $\Psi_\kappa^+$  be the reference even parity flux for  $\mathbf{r} \in V_\kappa$ , and  $\Psi_\gamma^-$  the corresponding reference odd parity flux for  $\mathbf{r}_\gamma \in \Gamma_\gamma$ . We examine the effect of taking arbitrary variations about these reference functions:

$$\Psi^+ = \Psi_\kappa^+ + \delta\Psi^+, \quad (3.34a)$$

$$\Psi^- = \Psi_\gamma^- + \delta\Psi^-. \quad (3.34b)$$

Substituting equations (3.34) into (3.33), we may express the value of the functional  $F_\kappa$  in terms of the arbitrary variations  $\delta\Psi^-$  and  $\delta\Psi^+$ :

$$F_\kappa[\Psi_\kappa^+ + \delta\Psi^+, \Psi_\kappa^- + \delta\Psi^-] = F_\kappa[\Psi_\kappa^+, \Psi_\kappa^-] + \delta F_\kappa[\Psi^+, \Psi^-] + \delta^2 F_\kappa[\Psi^+, \Psi^-]. \quad (3.35)$$

where the three terms on the right are referred to as the zero, the first, and the second variations with respect to  $\Psi^+$  and  $\Psi^-$ . The second variation  $\delta^2 F_\kappa[\Psi^+, \Psi^-]$  contains only products of variations  $\delta\Psi^+$  and  $\delta\Psi^-$ , so we can neglect it. The first variation  $\delta F_\kappa[\Psi^+, \Psi^-]$  can be written, using the divergence theorem and rearranging terms:

$$\begin{aligned} \delta F_\kappa = & 2 \int_\kappa dV \int_{4\pi} d\Omega \delta\Psi^+ [-\mathbf{\Omega} \cdot \nabla \Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi_\kappa^+ + \Sigma_t \Psi_\kappa^+ - \Sigma_s \Phi_\kappa - S] + \\ & + 2 \int_\gamma d\Gamma \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_\gamma \delta\Psi^+ (\Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi_\kappa^+ + \Psi_\gamma^-) + 2 \int_\gamma d\Gamma \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_\gamma \Psi_\kappa^+ \delta\Psi^- \end{aligned} \quad (3.36)$$

We require now that the global functional, equation (3.32), is stationary. This in turn postulates that the first order variation vanishes. This yields:

$$\delta F[\Psi^+, \Psi^-] = \sum_\kappa \delta F_\kappa[\Psi^+, \Psi^-] = 0. \quad (3.37)$$

Since variations  $\delta\Psi^+$  and  $\delta\Psi^-$  are arbitrary, equation (3.37) is satisfied only if the first volume term in (3.36) is zero for each node. This requirement is exactly the even-parity equation (3.28). Thus within each node, the even-parity transport equation is the Euler-Lagrange equation of the global functional (3.32).

The surface integrals over nodal interfaces in (3.36) must be treated differently. Consider the interface between two adjacent nodes  $\kappa$  and  $\kappa'$ . The contribution of this interface to the variation of the global functional  $\delta F[\Psi^+, \Psi^-]$ , equation (3.37), may be written, since in this case  $\mathbf{n}_{\gamma'} = -\mathbf{n}_\gamma$ , as the sum of just two integrals:

$$\begin{aligned} & 2 \int_\gamma d\Gamma \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_\gamma \delta\Psi^+ (\Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi_\kappa^+ - \Sigma_t^{-1} \mathbf{\Omega} \cdot \nabla \Psi_{\kappa'}^+) + \\ & + 2 \int_\gamma d\Gamma \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_\gamma \delta\Psi^- (\Psi_\kappa^+ - \Psi_{\kappa'}^+). \end{aligned} \quad (3.38)$$

For the second integral to vanish for arbitrary variations of  $\delta\Psi^-$ , the even parity flux  $\Psi_\kappa^+$  must be continuous across the interface. Likewise, for the first integral to vanish, the flux gradient terms, which as we have already shown in equation (3.29) are the odd-parity fluxes  $\Psi_\gamma^-$ , must also be continuous across the interface.

Now we can use the variational formulation to discretize the transport equation and obtain a set of linear equations suitable for numerical computations. The classical Ritz procedure is used. We adopt here the classical convention that repeated English subscripts imply summation over the index.

The even- and odd- parity fluxes are expanded in terms of known trial functions. The even-parity fluxes  $\Psi^+(\mathbf{r}, \mathbf{\Omega})$  are expanded within the node volume, the odd-parity fluxes  $\Psi^-(\mathbf{r}, \mathbf{\Omega})$  on each node surface:

$$\Psi^+(\mathbf{r}, \mathbf{\Omega}) = d_i(\mathbf{r}) g_m^+(\mathbf{\Omega}) \psi_{im}^+, \quad (3.39a)$$

$$\Psi^-(\mathbf{r}, \mathbf{\Omega}) = h_{j_\gamma}(\mathbf{r}) k_{n_\gamma}(\mathbf{\Omega}) \psi_{jn_\gamma}^-, \quad (3.39b)$$

where  $\psi_{im}^+$  and  $\psi_{jn_\gamma}^-$  are arrays of unknown coefficients,  $d_i(\mathbf{r})$  and  $h_{j_\gamma}(\mathbf{r})$  are the spatial basis functions over the node and over its surfaces respectively, and  $g_m^+(\mathbf{\Omega})$  and  $k_{m_\gamma}(\mathbf{\Omega})$  are the angular basis functions for the even- and odd- parity fluxes. More specifically,  $\psi_{im}^+$  represents the even-parity flux moment onto spatial function  $d_i(\mathbf{r})$  and angular function  $g_m^+$ . Of course the same is true for the odd-parity flux. The even-parity group sources  $S(\mathbf{r})$  are represented as:

$$S(\mathbf{r}) = d_i(\mathbf{r})s_i, \quad (3.40)$$

where  $s_i$  are the expansion coefficients onto the basis function set. The scalar flux  $\Phi(\mathbf{r})$  is, in view of equation (3.30):

$$\Phi(\mathbf{r}) = d_i(\mathbf{r})\delta_{1m}\psi_{im}^+. \quad (3.41)$$

The  $d_i(\mathbf{r})$  and  $h_{j_\gamma}(\mathbf{r})$  are complete polynomials. They are orthonormal over the node volume and surfaces, meeting the conditions:

$$\int_{\kappa} d_i(\mathbf{r})d_{i'}(\mathbf{r})dV = \delta_{ii'}, \quad (3.42)$$

and

$$\int_{\gamma} h_{j_\gamma}(\mathbf{r})h_{j'_\gamma}(\mathbf{r})d\Gamma = \delta_{jj'}. \quad (3.43)$$

The angular basis functions,  $g_m^+(\mathbf{\Omega})$ , are even-parity spherical harmonics meeting the orthonormal conditions:

$$\int_{4\pi} g_m^+(\mathbf{\Omega})g_{m'}^+(\mathbf{\Omega})d\Omega = \delta_{mm'}. \quad (3.44)$$

The odd-parity basis functions,  $h_{n_\gamma}(\mathbf{\Omega})$ , consist of odd-spherical harmonics. Inserting expansions of the even- and odd- parity fluxes, and of the source into equation (3.33) results in the reduced functional  $F[\psi_{im}^+, \psi_{jn_\gamma}^-]$ :

$$F[\psi_{im}^+, \psi_{jn_\gamma}^-] = \psi_{im}^+ A_{ii'}^{mm'} \psi_{i'm'}^+ - 2\psi_{im}^+ s_{im} + 2 \sum_{\gamma} \psi_{im}^+ M_{ij_\gamma}^{mn} \psi_{jn_\gamma}^-, \quad (3.45)$$

where, for convenience, I have defined  $s_{im} = \delta_{0m}s_i$ . The matrices are defined as:

$$A_{ii'}^{mm'} = \Sigma_t^{-1} P_{ii}^{kl} H_{kl}^{mm'} + V_{\kappa} \delta_{ii'} (\Sigma_t \delta_{mm'} - \Sigma_s \delta_{1m} \delta_{1m'}), \quad (3.46a)$$

and

$$M_{ij_\gamma}^{mn} = D_{ij_\gamma} E_{mn_\gamma}, \quad (3.46b)$$

where matrices  $P_{ii'}^{kl}$ ,  $D_{ij_\gamma}$ ,  $E_{mn_\gamma}$  and  $H_{kl}^{mm'}$  involve integrations over known basis functions:

$$P_{ii'}^{kl} = \int_V dV \nabla_k d_i(\mathbf{r}) \nabla_l d_{i'}(\mathbf{r}), \quad (3.47a)$$

$$H_{kl}^{mm'} = \int_{4\pi} d\Omega \Omega_k \Omega_l g_m^+(\mathbf{\Omega}) g_{m'}^+(\mathbf{\Omega}), \quad (3.47b)$$

$$D_{ij_\gamma} = \int_{\gamma} d\Gamma d_i(\mathbf{r}) h_{j_\gamma}(\mathbf{r}), \quad (3.47c)$$

$$E_{mn_\gamma} = \int_{4\pi} d\Omega \mathbf{\Omega} \cdot \mathbf{n}_{\gamma} g_m^+(\mathbf{\Omega}) k_n(\mathbf{\Omega}). \quad (3.47d)$$

Each of the elements of matrices (3.47) is given in terms of integrals of known spatial or angular trial functions, and they can be evaluated, once the set of these trial functions is chosen. Equation (3.45) can be written in a more compact form by defining vector  $\boldsymbol{\psi}^+$  and  $\boldsymbol{\psi}_\gamma^-$  as partitioned vectors formed from the successive columns of  $\psi_{im}^+$  and  $\chi_{jn_\gamma}$ , and  $\mathbf{s}$  as a vector containing source moments:

$$F_\kappa[\boldsymbol{\psi}^+, \boldsymbol{\psi}_\gamma^-] = (\boldsymbol{\psi}^+)^t \mathbf{A} \boldsymbol{\psi}^+ - 2(\boldsymbol{\psi}^+)^t \mathbf{s} + 2(\boldsymbol{\psi}^+)^t \mathbf{M} \boldsymbol{\psi}_\gamma^-. \quad (3.48)$$

Requiring the functional to be stationary with respect to variations in  $\boldsymbol{\psi}^+$  yields:

$$\boldsymbol{\psi}^+ = \mathbf{A}^{-1} \mathbf{s} - \mathbf{A}^{-1} \mathbf{M} \boldsymbol{\psi}_\gamma^-. \quad (3.49)$$

The variation with respect to  $\boldsymbol{\psi}_\gamma^-$  requires continuity of the quantity

$$\boldsymbol{\psi}_\gamma^+ = \mathbf{M}^t \boldsymbol{\psi}^+ \quad (3.50)$$

across nodal interfaces.  $\boldsymbol{\psi}_\gamma^+$  is the vector of even-parity flux moments on the nodal interface  $\gamma$ . Combining equations (3.50) and (3.49) we have:

$$\boldsymbol{\psi}_\gamma^+ = \mathbf{M}^t \mathbf{A}^{-1} \mathbf{s} - \mathbf{M}^t \mathbf{A}^{-1} \mathbf{M} \boldsymbol{\psi}_\gamma^-. \quad (3.51)$$

To obtain a response matrix in the conventional form we introduce a change of variables:

$$\mathbf{j}_\gamma^\pm = \frac{1}{4} \boldsymbol{\psi}_\gamma^+ \pm \frac{1}{2} \boldsymbol{\psi}_\gamma^-, \quad (3.52)$$

where  $\mathbf{j}_\gamma^\pm$  are respectively outgoing and incoming partial current moments on a nodal interface  $\gamma$ . Combining equation (3.52) and (3.51) we have:

$$\mathbf{j}_\gamma^+ = \mathbf{B} \mathbf{s} + \mathbf{R} \mathbf{j}_\gamma^-, \quad (3.53)$$

where  $\mathbf{R}$  and  $\mathbf{B}$  are defined in terms of  $\mathbf{A}$  and  $\mathbf{M}$  matrices, see equations (3.46). The response matrix equations, (3.53), are then solved, and the even-parity flux moments  $\boldsymbol{\psi}^+$  can be computed by:

$$\boldsymbol{\psi}^+ = \mathbf{A}^{-1} \mathbf{s} - \mathbf{A}^{-1} \mathbf{M} (\mathbf{j}_\gamma^+ - \mathbf{j}_\gamma^-). \quad (3.54)$$

The VNM has been implemented into the VARIANT code. It solves equation (3.53) for the partial currents  $\mathbf{j}_\gamma^+$  and  $\mathbf{j}_\gamma^-$ , and then reconstructs even-parity flux moments on a node volume,  $\boldsymbol{\psi}^+$ , from equation (3.54).

Matrices  $\mathbf{R}$  and  $\mathbf{B}$  are called the response matrices of a node, since they describe how a node responds, the outgoing neutron current  $\mathbf{j}_\gamma^+$ , to an incoming neutron current  $\mathbf{j}_\gamma^-$  and an internal neutron source  $\mathbf{s}$ . These matrices are evaluated once integrals (3.47) are known. In the actual version of VARIANT, these integrals are available only for homogeneous nodes, so are the response matrices  $\mathbf{R}$  and  $\mathbf{B}$ . I will describe later, see chapter 5, how we can drop this restriction and extend this algorithm to heterogeneous nodes as well.

### 3.4 KIN3D: A Kinetics Module for VARIANT

Solution of time-dependent neutron transport problem requires solution of equations (3.3). The usual approach is to transform the time-dependent problem into a quasi-static one, and to solve it at each time step. Therefore, every kinetics code relies on a static neutron solver. Transient problems differ from stationary ones in two aspects mainly: the presence of time derivative of the neutron flux and the addition of the delayed neutron source.

KIN3D is a kinetics module written by Rineiski [1997] on top of the VARIANT code. It can solve the kinetics problem with different solution schemes, e.g. direct method or space-time flux factorization options. It has the advantage of using the VNM, and this allows the use of a coarser neutronic meshes compared to other neutron transport solution techniques. A perturbation theory module is also included in KIN3D, since perturbation theory integrals – of the type of (3.9) – are needed for space-time flux factorization options. A perturbation theory based on even-parity flux moments is already described in literature by Laurin-Kovitz and Lewis [1995].

As a kinetics code, KIN3D must be given the driving perturbation as an input. Two kinds of perturbations are possible. The first one is a cross section change. Reactivity insertion is in this case determined by changing cross sections in some part of the reactor between two different time points. This mechanism can effectively be used to simulate the movement of a Control Rod (CR) or an increase in temperatures in some region of the core. The second class of perturbations is induced by changing the external neutron source – applicable for example to Accelerator Driven System (ADS). This section provides a brief description of the mathematical theory on which KIN3D is based.

Consider a time step between time points  $t_0$  and  $t_1$ . We are considering now how quantities (e.g. neutron fluxes) at the begin of the time step can be related to quantities at the end of the time step.

#### 3.4.1 Treatment of Flux Time Derivative

The simplest way to treat the flux time derivative is to assume a particular time dependence for the flux within the time step. We assume that within this time step, flux dependence in time can be factorized as:

$$\Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t) \approx P(\mathbf{r}, t) \{ \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_0) + \alpha_g(\mathbf{r}, \boldsymbol{\Omega}) (t - t_0) \}, \quad (3.55)$$

where  $\alpha_g(\mathbf{r}, \boldsymbol{\Omega})$  is an unknown variable and  $P(\mathbf{r}, t)$  is a fitting function. We can eliminate this unknown variable by evaluating (3.55) at time  $t_1$  and rearranging the relation to:

$$\alpha_g(\mathbf{r}, \boldsymbol{\Omega}) = \frac{1}{\Delta t} \left\{ \frac{1}{P(\mathbf{r}, t_1)} \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_1) - \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_0) \right\}. \quad (3.56)$$

The flux time derivative then can be written as:

$$\frac{\partial}{\partial t} \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_1) = \left( \frac{1}{P(\mathbf{r}, t_1)} \frac{\partial P(\mathbf{r}, t_1)}{\partial t} + \frac{1}{\Delta t} \right) \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_1) - \frac{P(\mathbf{r}, t_1)}{\Delta t} \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_0). \quad (3.57)$$

### 3.4.2 Evolution in Time of Delayed Neutron Precursors

Similarly, we can suppose that the fission source  $F(\mathbf{r}, t)$  behaves in a similar manner as the neutron flux, equation (3.55):

$$\beta_i(\mathbf{r}, t) F(\mathbf{r}, t) \approx P(\mathbf{r}, t) (\beta_i(\mathbf{r}, t_0) F(\mathbf{r}, t_0) + a_i(\mathbf{r}) (t - t_0)). \quad (3.58)$$

Equation (3.58) can be inserted into equation (3.3) and time-integrated to get an equation for the precursor concentration  $C_i(\mathbf{r}, t)$  at time  $t_1$ , knowing its value at time  $t_0$ :

$$\begin{aligned} C_i(\mathbf{r}, t_1) = & C_i(\mathbf{r}, t_0) e^{-\lambda_i \Delta t} + \beta_i(\mathbf{r}, t_0) F(\mathbf{r}, t_0) \int_0^{\Delta t} P(\mathbf{r}, \tau + t_0) e^{-\lambda_i(\Delta t - \tau)} \left(1 - \frac{\tau}{\Delta t}\right) d\tau + \\ & + \beta_i(\mathbf{r}, t_1) F(\mathbf{r}, t_1) \frac{1}{P(\mathbf{r}, t_1)} \int_0^{\Delta t} P(\mathbf{r}, \tau + t_0) e^{-\lambda_i(\Delta t - \tau)} \frac{\tau}{\Delta t} d\tau. \end{aligned} \quad (3.59)$$

### 3.4.3 An Equation for the Flux

An equation for the neutron flux at the end of the time step  $t_1$  can be obtained substituting (3.57) and (3.59) into equation (3.3):

$$\begin{aligned} \boldsymbol{\Omega} \cdot \nabla \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_1) + \tilde{\Sigma}_{t,g}(\mathbf{r}, t_1) \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_1) = & \frac{1}{4\pi} \sum_{g'} \Sigma_{s,gg'}(\mathbf{r}, t_1) \phi_{g'}(\mathbf{r}, t_1) + \\ & + \frac{1}{4\pi} \tilde{\chi}(\mathbf{r}, t_1) F(\mathbf{r}, t_1) + \frac{1}{4\pi} Q_g(\mathbf{r}, t_1) + \tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1). \end{aligned} \quad (3.60)$$

$Q_g(\mathbf{r}, t)$  represents the external neutron source into energy group  $g$ , and, for simplicity, is assumed isotropic. Equation (3.60) has now the same structure as a steady-state transport problem with an external, anisotropic source  $\tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$ , and relates the flux at the start of the time step, i.e.  $t_0$ , to the flux at the end of the time step, i.e.  $t_1$ .

Quantities with a tilde are defined as ‘‘artificial’’, as they contain terms, which are not physical but related to the time discretization method. More specifically, we introduce an artificial macroscopic cross section  $\tilde{\Sigma}_{t,g}(\mathbf{r}, t_1)$ , an artificial fission prompt spectrum  $\tilde{\chi}_g(\mathbf{r}, t_1)$ , and an artificial external neutron source  $\tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$ , which are given by:

$$\tilde{\Sigma}_{t,g}(\mathbf{r}, t_1) = \Sigma_{t,g}(\mathbf{r}, t_1) + \frac{1}{vP(\mathbf{r}, t_1)} \frac{\partial P(\mathbf{r}, t_1)}{\partial t} + \frac{1}{v\Delta t}, \quad (3.61a)$$

$$\tilde{\chi}_g(\mathbf{r}, t_1) = \chi_g^p(\mathbf{r}, t_1) - \sum_i \chi_{i,g}^d \beta_i(\mathbf{r}, t_1) \left\{ 1 - \frac{\lambda_1}{P(\mathbf{r}, t_1)} \int_0^{\Delta t} P(\mathbf{r}, \tau + t_0) e^{-\lambda_i(\Delta t - \tau)} \frac{\tau}{\Delta t} d\tau \right\}, \quad (3.61b)$$

$$\begin{aligned} \tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1) = & \frac{P(\mathbf{r}, t_1)}{v\Delta t} \Psi_g(\mathbf{r}, \boldsymbol{\Omega}, t_0) + \sum_i \lambda_i \chi_{i,g} C_i(\mathbf{r}, t_0) e^{-\lambda_i \Delta t} + \\ & \sum_i \lambda_i \chi_{i,g}^d \beta_i(\mathbf{r}, t_0) F(\mathbf{r}, t_0) \int_0^{\Delta t} P(\mathbf{r}, \tau + t_0) e^{\lambda_i(\Delta t - \tau)} \left(1 - \frac{\tau}{\Delta t}\right) d\tau. \end{aligned} \quad (3.61c)$$

where  $\chi_{i,g}^d$  is the delayed fission spectrum for family  $i$  onto energy group  $g$ . All these quantities can be computed once a particular fitting function  $P(\mathbf{r}, t)$  is selected.



### 3.4.4 Even-parity Flux Equation

Equation (3.60) is the equation we must solve. Since our tool is called VARIANT, we must recast that equation into its even-parity form. The procedure parallels the one already presented in section 3.3.1, we report here only the final results. The even-parity version of equation (3.60) is:

$$\begin{aligned} \boldsymbol{\Omega} \cdot \nabla \Psi_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_1) + \tilde{\Sigma}_{t,g}(\mathbf{r}, t_1) \Psi_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_1) &= \frac{1}{4\pi} \sum_{g'} \Sigma_{s,gg'}(\mathbf{r}, t_1) \Phi_{g'}(\mathbf{r}, t_1) + \\ &+ \frac{1}{4\pi} \tilde{\chi}_g(\mathbf{r}, t_1) F(\mathbf{r}, t_1) + \frac{1}{4\pi} S_g(\mathbf{r}, t_1) + \frac{1}{4\pi} \tilde{Q}_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1), \end{aligned} \quad (3.62a)$$

and the odd-parity equation is instead:

$$\boldsymbol{\Omega} \cdot \nabla \Phi_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_1) + \tilde{\Sigma}_{t,g}(\mathbf{r}, t_1) \Psi_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_1) = \tilde{Q}_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1), \quad (3.62b)$$

where  $\tilde{Q}_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$  and  $\tilde{Q}_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$  are the even- and odd- parity components of the “artificial source”  $\tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$ . To solve these equations, we follow the same approach used for solving transient diffusion equations, in which Fick’s Law is assumed valid under transient condition. At present, that translates equation (3.62b) into:

$$\boldsymbol{\Omega} \cdot \nabla \Psi_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_1) + \Sigma_{t,g}(\mathbf{r}, t_1) \Psi_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_1) \approx 0, \quad (3.63)$$

where the total cross section is the physical cross section  $\Sigma_{t,g}(\mathbf{r}, t_1)$  and not the “artificial” one  $\tilde{\Sigma}_{t,g}(\mathbf{r}, t_1)$  as according to equation (3.61a), and the odd-parity sources  $\tilde{Q}_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$  are ignored. With hypothesis (3.63) equation (3.62a) becomes:

$$\begin{aligned} -\boldsymbol{\Omega} \cdot \nabla \Sigma_{t,g}^{-1}(\mathbf{r}, t_1) \boldsymbol{\Omega} \cdot \nabla \Psi_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_1) + \tilde{\Sigma}_{t,g}(\mathbf{r}, t_1) \Psi_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_1) &= \frac{1}{4\pi} \sum_{g'} \Sigma_{s,gg'} \Phi_{g'}(\mathbf{r}, t_1) + \\ &+ \frac{1}{4\pi} \tilde{\chi}_g(\mathbf{r}, t_1) F(\mathbf{r}, t_1) + \frac{1}{4\pi} S_g(\mathbf{r}, t_1) + \tilde{Q}_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1). \end{aligned} \quad (3.64)$$

This is the final equation solved by VARIANT at each time step. The accuracy of the method is founded on the approximations (3.58), (3.55), and (3.63) to be valid within the time interval considered.

### 3.4.5 Solution Schemes

Depending on the choice of the fitting function  $P(\mathbf{r}, t)$ , we can have different solution schemes. The most simple of them is probably the “direct scheme”. It relies on one time mesh, and equation (3.64) is solved at each time step. We can select different fitting functions, the most simplest one is of course:

$$P(\mathbf{r}, t) = 1,$$

which implies, see (3.55), a linear dependence on time within one time step. More complicated function however can be adopted. One may argue for example, that between two time points, the flux is better approximated by an exponential function such as:

$$P(\mathbf{r}, t) = e^{\omega(\mathbf{r})(t-t_0)}.$$

The frequency  $\omega(\mathbf{r})$  can be estimated by looking at the previous history of the flux. At the other side of the scale, in terms of computational efforts, there is the point kinetics method, based on a flux factorization as such:

$$\Psi_g(\mathbf{r}, \mathbf{\Omega}, t) = P(t)\Psi_g(\mathbf{r}, \mathbf{\Omega}, 0). \quad (3.65)$$

The selection of this flux separation leads to the point kinetics equations, and the fitting function  $P(t)$  corresponds entirely to the point kinetics amplitude.

Assumption (3.65) effectively “freezes” the flux distribution at time zero, permitting variations only in flux level. This can be too restrictive for transients in which localized spatial effects might severely distort the flux distribution compared to the stationary one. A QS scheme can be implemented, if the flux function is defined as:

$$\Psi_g(\mathbf{r}, \mathbf{\Omega}, t) = P(t)\tilde{\Psi}_g(\mathbf{r}, \mathbf{\Omega}, t_{sh}), \quad (3.66)$$

where the flux  $\tilde{\Psi}_g(\mathbf{r}, \mathbf{\Omega}, t)$  is now evaluated – by solving equation (3.64) – at time points  $t_{sh}$ , selected on a relatively coarse time mesh.

## Chapter 4

# SIMMER-PARTISN coupling

As neutron transport solvers, the SIMMER codes use the  $S_N$  neutron transport approach from the DANTSYS code package described in Alcouffe et al. [1995]. More specifically, TWODANT and THREEDANT are the neutron transport solvers applied in for the two- and three- dimensional SIMMER versions.

The ever-increasing computational power has prompted the parallelization of the fluid-dynamics module of SIMMER. However, the official releases of the code still rely on the rather old TWODANT and THREEDANT codes, which are not parallelized. I felt it was worth trying to parallelize the neutronic module as well. Despite a large majority of FR CDA simulations show that most of the computational time is employed in solving the fluid-dynamics equations, some peculiar cases may feature a very high total neutronic time (i.e. the time spent in solving the time-dependent neutron transport equations). In those cases, a parallel neutron solver substantially reduces the computational time. Moreover, we can benefit from a parallel neutron solver by using more detailed neutronic models without increasing much the total time required for the computations. This last aspect is important especially for CDA simulations for FRs that require a detailed neutronic description in order to assess whether recriticality might or might not occur during the evolution of the accident. In selecting a candidate for a parallel neutronic module, the  $S_N$  PARTISN code is a natural choice Alcouffe and Baker [2009]. The name stands for PArallel TIme-dependent  $S_N$ . It is an evolution of the DANTSYS code package and provides support for multiple-processor calculations through the MPI standard.

In this chapter, the coupling between SIMMER codes and PARTISN is described. As the coupling procedure is almost independent of the SIMMER version in use (i.e. SIMMER-III or SIMMER-IV), I will be referring to it generally as SIMMER/PARTISN coupling. The chapter is organized in this way: first a detailed description of the coupling procedure is given in section 4.1. Later, the performances and the parallel speedup of the coupled code are analyzed in section 4.2. This section is also verifying the approach by means of a code to code comparison with the traditional SIMMER results. A validation through a comparison with experimental data coming from of a series of test that were performed in the EBR-II reactor is provided in section 4.3.

## 4.1 SIMMER-PARTISN Coupling Procedure

The easiest way to couple SIMMER and PARTISN is to keep the two executables separated and exchange data via external files, as explained in Marchetti et al. [2014]. In the procedure, PARTISN can be regarded as an external program called by the controlling program (i.e. SIMMER) each time a new solution of the neutron transport equation is required. A SIMMER subroutine (i.e. subroutine LINKM) writes down all the input files needed by PARTISN, and a subsequent system call starts the PARTISN computation upon these SIMMER-written input files.

When the PARTISN computation is done, SIMMER takes back control of the computation. The data exchange between the two codes is achieved via external binary files, which are in turn read and written by each of the two codes.

Of course, writing and reading of external files is slower than a direct access to the memory<sup>1</sup>. But, by approach the controlling program (i.e. SIMMER) masters the PARTISN process. However, this solution is the simplest and, in particular, allows a “decoupling” between SIMMER and PARTISN parallelization. This means, for example, that SIMMER can be parallelized independently (e.g. with OpenMP, or MPI, or even both) of the parallelization in PARTISN.

### 4.1.1 Data Flow

The flow of data between the two codes is depicted in figure 4.1. SIMMER holds almost all quantities in FORTRAN COMMON blocks (e.g. scalar and angular fluxes are stored in common blocks, as do many other quantities, such as cross sections, delayed neutron data, etc. . . ), as shown in step one in figure 4.1.

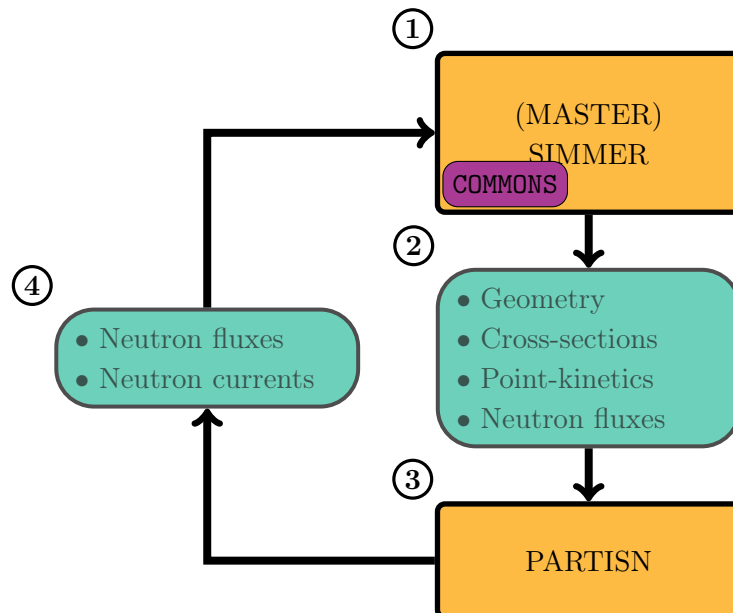


Figure 4.1: Sketch of the flow of data between SIMMER-III and PARTISN.

<sup>1</sup>This can be solved by using techniques that allow two processes to share a common memory area

This data is used in the SIMMER subroutine LINKM to prepare (i.e. to write) the needed input files for PARTISN calculations, this corresponds to step 2 in the figure. More specifically, these input files contain all the input data needed to start a PARTISN calculation, they comprise among others:

1. geometry data, that defines the arrangement of the core,
2. cross-sections for each material in the domain,
3. point-kinetics quantities, and
4. scalar and angular fluxes to provide the neutronic state at the previous time point.

All these data are read by the PARTISN code and a neutron flux calculation is initiated, see step 3 in figure 4.1. The geometric data comprise, essentially, the neutronic mesh boundaries and the material distribution in each computational node. Each of them is assigned to a particular set of cross-sections, contained in a separate input file. The scalar neutron fluxes on each neutronic node are used as an initial flux guess for non-stationary (i.e. shape update) PARTISN computations to accelerate the convergence.

The angular fluxes at the previous time step are needed because to evaluate numerically the neutron flux time derivative in the transport equation, see equations (3.3). Point-kinetics quantities are passed via another binary file (called SIMDAT). This file is specific to the SIMMER/PARTISN procedure, and it will be described later in section 4.1.2. On the other side, PARTISN also outputs binary files, as indicated in step 4 in figure 4.1. They contains the following up-to-date quantities:

1. Scalar and angular fluxes,
2. neutron currents on each node surface, and
3. updated value of the QS constraint  $\gamma$ .

This data is read back by the SIMMER code and a new cycle can start with a new step 1. The up-to-date neutron currents and the QS constraint  $\gamma$  are passed to SIMMER via an additional binary file (called PASDUT in the following). This file is also specific to the SIMMER/PARTISN coupling. A short description of its contents is given in section 4.1.2

### 4.1.2 Additional Data Files

Because of the QS approach of SIMMER, additional data must be passed to PARTISN. This data exchange is done via the two additional files mentioned above:

1. SIMDAT: from SIMMER to PARTISN,
2. PASDUT: from PARTISN to SIMMER.

The SIMDAT file is written by SIMMER as an additional input for PARTISN. It contains some parameters that control, what kind of computation is required (i.e. whether it is a real steady-state, an adjoint steady-state, or a shape update calculation). It contains data that relates to the QS method itself: the number of the gamma iteration, the amplitude  $N(t)$ ,

the reactor period, the initial steady-state eigenvalue  $k_0$ , and, eventually, any external reactivity insertion. Several other convergence criteria (e.g. convergence criteria on the value of the QS constraint  $\gamma$ , see equation (3.6)) are also included in the file. Finally, delayed neutron data are also stored in this file, as the delayed neutron concentrations and their decay constants.

On the other hand, that is from PARTISN back to SIMMER, `PASDUT` file is written by PARTISN and read back by SIMMER. It contains the new value of the QS constraint  $\gamma$  as evaluated with the new flux shape, the number of outer iterations done, and the total computational time. Also x-, y-, and z- directed neutron currents are contained in the file. These currents are collected during PARTISN computation (i.e. more specifically during the last transport sweep). In line with the original SIMMER code, these currents are needed to evaluate the leakage component of the reactivity.

### 4.1.3 Modifications in SIMMER

Several SIMMER subroutines are modified for the coupling. To provide a full list of those modifications is of minor interest. Therefore, only the major modifications are described here, auxiliary subroutines (such as the ones needed for initialization) are omitted.

#### Modifications in subroutine LINKM

This is probably the most modified subroutine. The code writing the PARTISN input files is designed completely new. Also, some old files are no longer needed in PARTISN and are therefore suppressed. Two additional subroutines are added: one writing the `SIMDAT` file and one reading the `PASDUT` file. The latter, after reading, performs several other tasks: most importantly, it checks that the PARTISN computation ends without errors and/or convergence mismatches, and sets the value of the new QS constraint  $\gamma$  in SIMMER.

#### Modifications in subroutine GRIND

SIMMER `GRIND` subroutine is the main driver of the neutronic module. It calls the cross section-generating module, it calls the point kinetics module, and it calls the neutron transport solver. Calls to the original `TWODANT` code are substituted by calls to a new subroutine. This subroutine oversees the PARTISN computation:

1. it checks that all needed input data files are present,
2. it checks that the PARTISN executable is available,
3. it starts the PARTISN computation,
4. it checks that the output files are produced,
5. it returns control to `GRIND`.

#### 4.1.4 Modifications in PARTISN

As before, only the most important modifications of the code are described. Taking advantage of the FORTRAN 95 language, a new module keeping together all additional new subroutines and variables is established.

##### Delayed neutrons

The currently available version of PARTISN has no option to account for delayed neutrons. Therefore, the first significant PARTISN modification is the inclusion of delayed neutron support. As the adjoint computation is concerned, the problem is solved by modifying the total fission spectrum  $\chi_g^t$  for each energy group  $g$  as:

$$\chi_g^t = (1 - \beta)\chi_g^p + \sum_k \beta_k \chi_{k,g}^d, \quad (4.1)$$

where  $\beta_k$  is the delayed neutron fraction of precursor family  $k$ ,  $\chi_g^p$  is the prompt fission spectrum,  $\chi_{k,g}^d$  is the delayed fission spectrum of precursor family  $k$ , and  $\beta$  is the total delayed neutron fraction:

$$\beta = \sum_k \beta_k.$$

For direct calculations (i.e. both the initial real calculation and the subsequent shape calculations) the problem requires coding in PARTISN. The fission source is divided into the prompt and the delayed contribution:

$$\chi_g^p \sum_k (1 - \beta_k) F(\mathbf{r}, t) + \sum_k \chi_{k,g}^d \lambda_k C_k(\mathbf{r}, t), \quad (4.2)$$

where  $F(\mathbf{r}, t)$  is the total fission source at  $\mathbf{r}$ ,  $\lambda_k$  is the decay constant of precursor family  $k$ ,  $C_k(\mathbf{r}, t)$  is the spatial precursor concentration of family  $k$ .

The initial precursor concentrations  $C_k^{initial}(\mathbf{r}, t_0)$  are determined in the PARTISN code during the real steady-state calculation as:

$$C_k^{initial}(\mathbf{r}, t_0) = \frac{\beta_k}{\lambda_k} F(\mathbf{r}, t_0). \quad (4.3)$$

New initial precursor concentrations  $C_k^{initial}(\mathbf{r}, t_0)$  are evaluated using equation (4.3) at the start of each outer iteration, as soon as a new flux (and therefore a new fission source  $F(\mathbf{r}, t_0)$ ) is available. Their contribution is added to the fission source, according to equation (4.2), and the calculation proceeds until convergence. These initial precursor concentrations are then passed to SIMMER via the PASDUT file and stored in appropriate common blocks. During transient calculations, precursor concentrations are updated at each reactivity step in the SIMMER kinetics module by assuming a linear variation in time of the fission source  $F(\mathbf{r}, t)$  within the reactivity time step itself, and then solving the precursor equations (3.3b). At each shape step, precursor concentrations are written into the SIMDAT file, and are used by PARTISN to compute and to add the delayed neutron source to the prompt contribution, again according to equation (4.2).

**Time derivative of the flux**

During transient calculations, the time-dependent neutron transport problem is solved using a fully-implicit time scheme. Therefore, the discretization of the time derivative of the neutron flux shape  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$  is:

$$\frac{1}{v} \frac{\partial \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)}{\partial t} \Big|_{t_i} \approx \frac{1}{v \Delta t} \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t_i) - \frac{1}{v \Delta t} \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t_{i-1}), \quad (4.4)$$

This results in the term  $\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t_{i-1})$ , which represents the angular neutron flux at the previous time step. This is equivalent to an anisotropic source term. This term is taken into account in PARTISN by adding a term to the neutron source vector during the transport sweep. A new PARTISN subroutine is tasked with reading the angular flux vector at the previous time step, storing it in a local array and then adding it to the neutron source.

**Time absorption cross section**

Because of the quasistatic scheme, the additional “absorption” term:

$$\left( \frac{1}{v N(t_i)} \frac{d}{dt} N(t_i) \right) \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t_i), \quad (4.5)$$

appears also in the shape update equation. This term is dealt by modifying the total cross section  $\Sigma_t(\mathbf{r}, E, t_i)$  as:

$$\bar{\Sigma}_t(\mathbf{r}, E, t_i) = \Sigma_t(\mathbf{r}, E, t_i) + \frac{1}{v N(t_i)} \frac{d}{dt} N(t_i). \quad (4.6)$$

This additional factor to the total cross section  $\Sigma_t(\mathbf{r}, E, t_i)$  in equation (4.6) acts therefore as an additional absorption cross section: it is usually referred to as “time absorption” cross section since it involves time derivative of the amplitude  $N(t)$ . A new PARTISN subroutine adds this term to the total cross section. Note that the amplitude  $N(t)$  and its derivative with respect to time originate from the point kinetics module of SIMMER, and are passed to PARTISN via the SIMDAT file.

**Neutron currents**

As already mentioned before, the point kinetics module of SIMMER requires the neutron currents on each node surface, in order to evaluate the leakage component of the reactivity. These currents are already evaluated in PARTISN during the computations, however, in the current version of the code, there is no option to output them.

An additional subroutine stores these surface neutron currents of the final PARTISN iteration (i.e. when the fluxes are well converged) and writes them into the PASDUT file.

**New options for the PARTISN iteration scheme**

PARTISN has been modified to use the QS constraint  $\gamma$  to accelerate the convergence of the shape equation. In the TWODANT environment, the constraint is used to divide the prompt fission source. A new QS constraint  $\gamma$  is evaluated after each transport inner iteration and after each sub-outer iteration. When the flux is converged, also is  $\gamma$ , but it



may be far from one, as required by the QS scheme. In such cases, a repetition of the reactivity steps and a solution of a new shape flux is initiated to converge the value of  $\gamma$  to one. This is the gamma iteration as described in section 3.1.2.

The same scheme is replicated in PARTISN. A new PARTISN subroutine takes the fluxes as an argument and evaluates  $\gamma$ , after each transport inner and each diffusion sub-outer iteration. As in TWODANT, additional convergence checks on the  $\gamma$  value are introduced into the PARTISN inner/outer iteration strategy. The computed  $\gamma$  is finally passed to SIMMER using the PASDUT file, when the shape update converged.

### Evaluation of data adjustment factors

In case of initial transient conditions, a correct initialization of the space neutronics requires the evaluation of so-called data adjustment factors, introduced in section 3.1.2. SIMMER defines three different classes of these factors. Each of them modify a certain physical quantity in order to have consistency between the neutron spatial distribution and the point kinetics quantities provided as initial conditions to the SIMMER code.

Factor  $f_v$  modifies neutron velocities. Factor  $h_k$  adjusts delayed neutron yields  $\beta_k$  for each delayed neutron family  $k$ . Factor  $g_k$  provides a proportionality factor between the initial delayed neutron concentration  $C_k^{initial}(\mathbf{r}, t_0)$  and the initial fission neutron source  $F(\mathbf{r}, t_0)$ , see equation (4.3). In the original TWODANT scheme, these factors are evaluated (i.e.  $h_k$ ,  $g_k$ , and  $f_v$ ) during the initial steady state calculation. At each outer iteration (i.e. when new fluxes become available) these factors are evaluated and used in the computation (e.g. by multiplying the delayed neutron fractions  $\beta_k$  in the delayed neutron source term of transport equation). To be as consistent as possible, the same scheme is replicated in PARTISN. As the outer iterations proceed, a new subroutine evaluates the data adjustment factors, given the new fluxes and the physical quantities needed (e.g. delayed neutron fractions  $\beta_k$ ). Upon convergence of the calculation, this data is written on the additional output file PASDUT and transferred back to SIMMER.

## 4.2 Verification of SIMMER/PARTISN Coupling

In this section, several applications of the SIMMER/PARTISN coupling are presented starting from two dimensional cases (SIMMER-III/PARTISN) and then presenting three dimensional ones (SIMMER-IV/PARTISN).

The aim is to provide a verification of the SIMMER/PARTISN coupling. Results are first compared against the standard SIMMER version. Later on, one case will provide also a first validation of the coupling against experimental data, section 4.3.

### 4.2.1 Two Dimensional Space-Time Neutronics Problem

I start here from the Space-Time Neutronics Problem (STN) for SIMMER-III. It is one of the sample problem distributed with the SIMMER-III code, see Yamano et al. [2003a].

### Description of the problem

The problem consists of a transient induced by a lump of fuel being injected into a LMFBR-type core from below. As the fuel is injected into the core the reactivity rises accompanied by a neutronic feedback. The problem is tailored so that, when the fuel lump reaches core mid-plane, the reactor becomes prompt-critical. After that, the fuel travels away from the core, bringing the reactivity down again.

The corresponding SIMMER-III r-z model is shown in figure 4.2. The complementing fluid-dynamics mesh is composed of 12 radial and 16 axial meshes. The neutronic mesh is divided into 30 radial and 40 axial meshes. Eighteen energy groups are used, while the discrete ordinate order is set to 4.

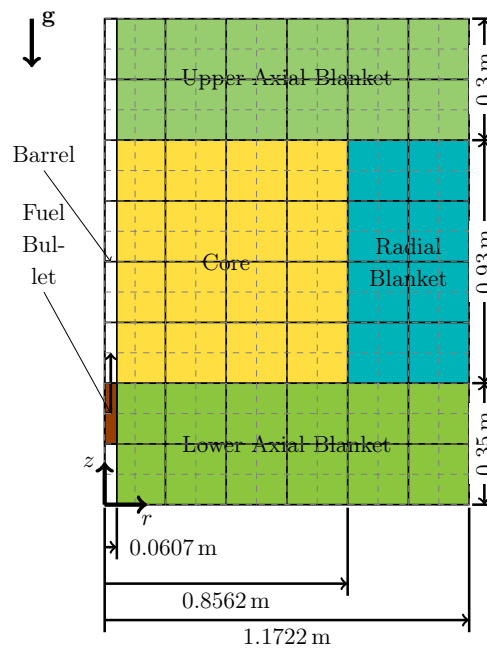


Figure 4.2: Schematical representation of the Space-Time Neutronics (STN) problem for a fuel bullet injection in a LMFBR reactor from below.

### Steady-state results

First of all, the data coming from the initial steady-state calculation are compared. Figure 4.3 shows the differences in the fast flux (i.e. energy group number 1) between SIMMER-III and SIMMER-III/PARTISN codes at core mid-plane (i.e. at axial mesh number 20). Relative differences do not exceed  $10^{-4}$ . The  $k_{eff}$ s are identical within the convergence required ( $10^{-5}$ ). The same order of errors is found for the adjoint fluxes.

### Transient results

With an initial velocity of 100 m/s, the fuel bullet reaches the core midplane in  $\approx 5$  ms and the top of the reactor in  $\approx 10$  ms. The highest reactivity value is reached when the bullet is at core midplane (since the neutron importance is higher at this axial position).

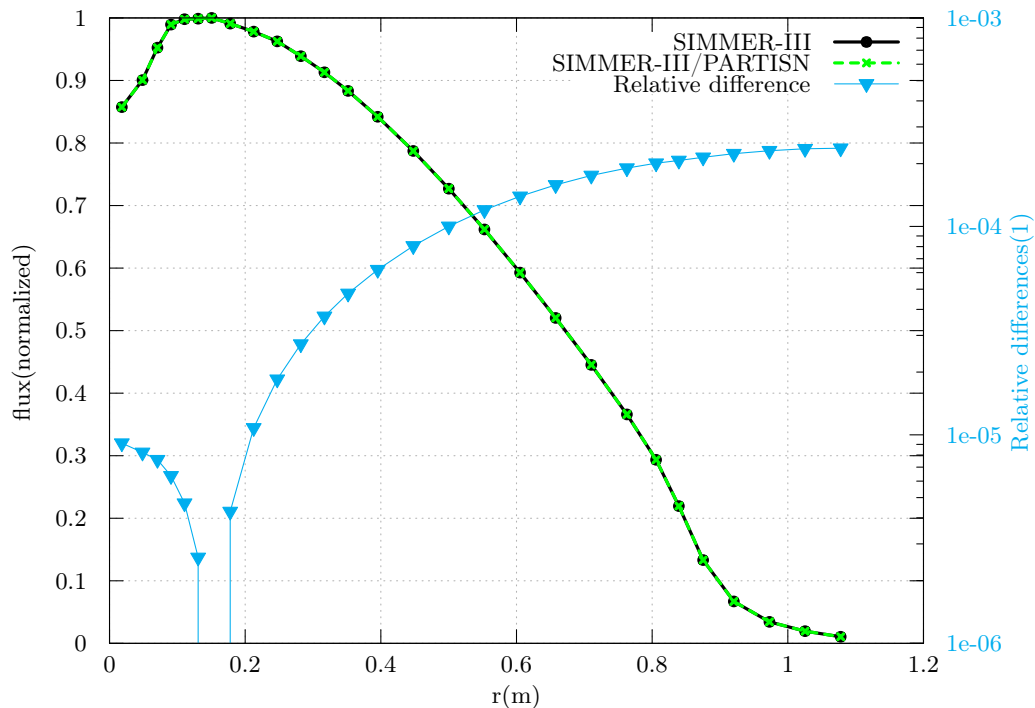


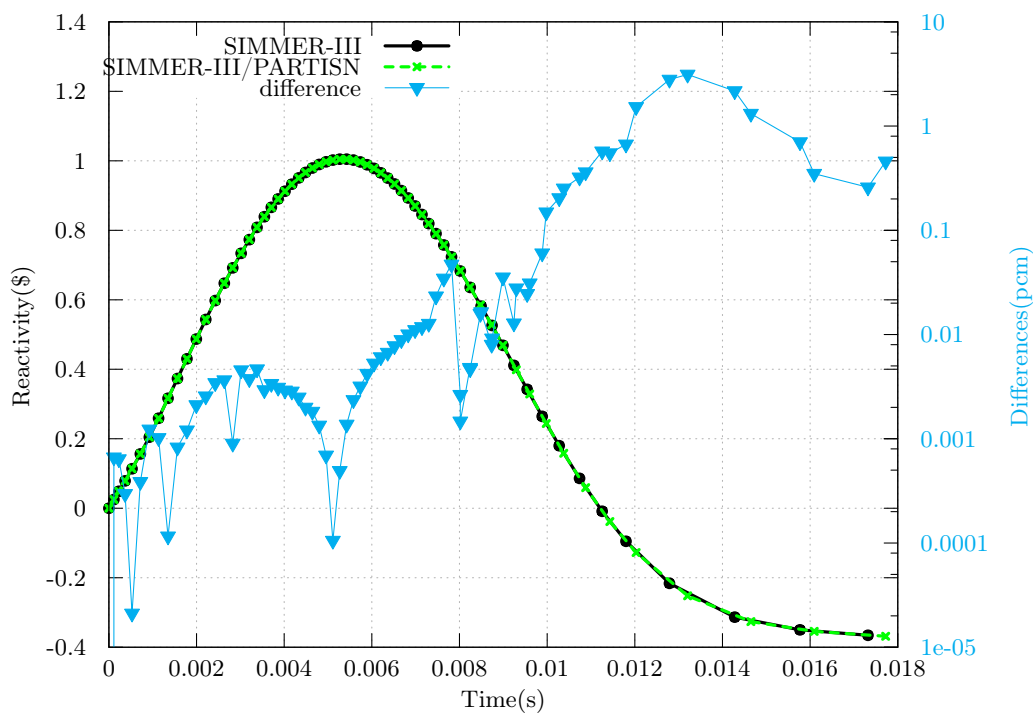
Figure 4.3: Comparison of the radial fast flux steady-state distribution between SIMMER-III and PARTISN of the LMFBR problem of figure 4.2.

After that point, in fact, the upwards movement of the fuel bullet inserts negative reactivity. A peak in the amplitude (and, therefore, in the power of the reactor) follows closely the reactivity peak. The power excursion in this case is mild (i.e. only 20 times the nominal power level). As a consequence, only a limited amount of energy is deposited in the fuel, hence no other reactivity feedbacks (except the feedback caused by the movement of the fuel bullet itself) play a significative role in the transient.

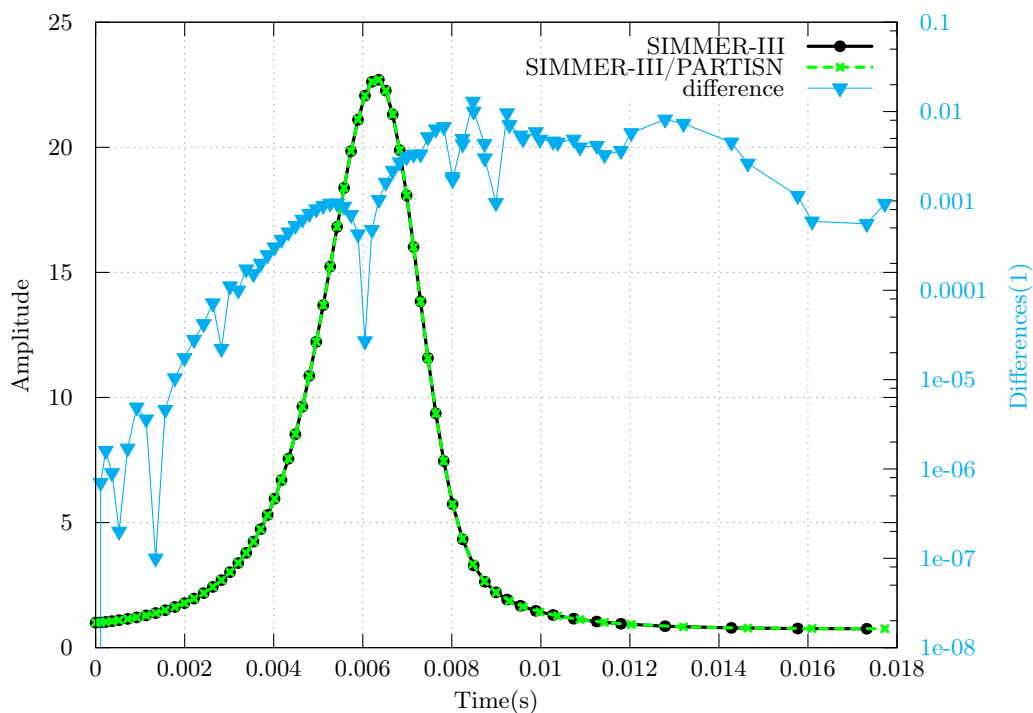
Transient results comprise reactivity and amplitude traces, as illustrated in figure 4.4. The agreement between SIMMER and SIMMER-III/PARTISN is very good. Differences between the two codes are negligible for this case. Reactivity values differ by no more than 1 pcm, see figure 4.4(a). This agreement brings very small differences in the amplitude, with relative amplitude differences smaller than 1%, see figure 4.4(b). The neutron solver is called at the very same times in the two computations for most of the transient. Towards the end, however, that is no longer true. This may stem from small flux differences accumulating during the transient and determining different time shape steps later on.

### Investigation of parallel option

The total computational time for SIMMER-III is 25.2 s, 68% of which (i.e.  $\approx 17$  s) is spent for the neutron solver TWODANT, as indicated in table 4.1. These  $\approx 17$  s represent the effective part of the computation I can accelerate with the coupled code. First I run the serial option for the SIMMER-III/PARTISN coupled code, where similar data compared to SIMMER are obtained: a total computational time of 26.6 s, with approximately 64% spent into the PARTISN code.



(a) Reactivity.



(b) Amplitude.

Figure 4.4: Comparison between SIMMER-III and SIMMER-III/PARTISN for STN benchmark of a LMFBR. (a) temporal evolution of the reactivity and (b) amplitude as a function of time.

I then investigated the speedup  $S$  in case of multiple processors, see table 4.1. The speedup is defined as the ratio of the serial computational time  $T_{serial}$  over the parallel one  $T_{parallel}$ , that is:

$$S = \frac{T_{serial}}{T_{parallel}}.$$

In this case, a fraction  $f_{npar} \approx 1 - 0.64$  of the code is not parallelized (i.e. the fluid-dynamics part) so the speedup can be expressed as:

$$S = \frac{T_{serial}}{f_{npar} T_{serial} + (1 - f_{npar}) \frac{T_{serial}}{PE}}, \quad (4.7)$$

where  $PE$  is the number of processors used in the parallel part of the code (in our case only in the PARTISN neutron solver). Equation (4.7) is the well-known Amdahl's Law, and it limits the maximum achievable speedup  $S_{max}$  to:

$$S_{max} = \frac{1}{f_{npar}},$$

as the number of processors  $PE$  tends to infinity. The maximum achievable speedup in this case is therefore  $S_{max} \approx 2.78$ . As it can be seen from table 4.1, the highest speedup achieved is  $\approx 1.42$ , rather far from the maximum one. Also, increasing further the number of processors (i.e. going to eight processors) seems not to improve performances. This may be due to additional computation and communication overhead caused by spawning more and more processes. The mesh used in this case is in fact small (only 30x40). Moreover, thanks to acceleration techniques in use in both TWODANT and PARTISN (such as gamma iteration and diffusion synthetic acceleration), updates of the flux shape take, for this particular case, only a fraction of a second in both codes (i.e.  $\approx 0.3$ s for the first flux shape update in TWODANT and in the one-processor PARTISN computation). Parallel PARTISN computations are faster. However, a comparison between them reveals that the first flux shape update in the two-, four-, and eight-processor PARTISN calculations takes the same amount of time (i.e. approximately 0.2s). Therefore, the gain in the computational time due to multiple processes is offset by the communication overhead between the different processes and additional computations needed in the parallel cases, resulting in no significant speedup as the number of processors is increased further. However, with 4 processors, the SIMMER-III/PARTISN code is still able to reduce the total computational time by a considerable amount of 30%, table 4.1.

#### 4.2.2 A Two-Dimensional Fast Reactor Unprotected Loss-of-Flow

The coupled SIMMER-III/PARTISN code is verified here for a ULOF accident in a pool-type LMFBR core. In this example, the transition phase is modeled: a major portion of the core is already melted and forms three different pools, separated by the Control Rods Guide Tubes. It is because of that pool separation that a rather mild transient evolves. The initiation phase is modeled with a point kinetics code. Therefore, the SIMMER calculation starts with a reactor in a transient state. This example provides an opportunity to validate the initialization procedure of the space neutronics implemented in PARTISN, see section 3.1.2.

	SIMMER-III	SIMMER-III/PARTISN			
		1pe	2pe	4pe	8pe
Computational time(s)	25.2	26.6	19.4	18.7	19.3
Neutronics time(s)	17.2	17.0	10.3	9.72	10.0
Fraction(%)	68	64	53	52	52
Speed-up	-	-	1.37	1.42	1.38

Table 4.1: Investigation of speedup for STN benchmark. Maximum theoretically achievable speedup is 2.78, as according to equation (4.7).

### Case Description

The reactor is composed of 14 rings of hexagonal subassemblies with a 11.65 cm-long pitch. The core comprises the innermost 9 rings, surrounded by 3 rings of fuel blanket subassemblies and 2 further rings of radial shield subassemblies. Control rods are located in the inner core, within the first seven rings. There are in total 19 control rods: one located at the center of the reactor, 6 located in ring number 3 and the rest located in ring number 7. The SIMMER-III geometric model of the reactor is a two-dimensional r-z model with 39 radial and 40 axial fluid-dynamic meshes. Radially, it extends to  $\approx 3$  m with the last node representing the sodium pool. The neutronic mesh extends radially to include two rings of radial shield subassemblies for a total of 71 radial meshes. Axially, the neutronic mesh comprises 94 meshes, stretching from the lower shield up to the upper gas plenum.

The SIMMER-III model of the core is shown in figure 4.5. A major portion of the pins are already disrupted. However, the can walls of each subassembly are still physically intact, preventing any radial fuel motion. No liquid sodium is present in the core, except for the three CR subassemblies. An eleven energy-group library is used, with a  $S_N$  order equal to 4. As usual, six delayed neutron families are used.

### Steady-state results

Since the SIMMER calculation starts from a transient state, data adjustment factors, defined in section 3.1.2, are evaluated during the first neutron flux calculation. These factors are compared between SIMMER-III and SIMMER-III/PARTISN in table 4.2. Relative errors against SIMMER-III calculation are reported there. Data adjustment factors for delayed-neutron fractions  $h_k$  differ by only one part in one million. For precursor factors  $g_k$ , relative errors are of the order of  $10^{-8}$  and for the velocity factor  $f_v$  of  $10^{-5}$ . Cell-wise comparison of initial delayed neutron precursor concentrations  $C_k^{initial}(\mathbf{r}, t_0)$  reveals the same order of error, that is  $10^{-5}$ .

The fast flux at core mid-plane (i.e. axial mesh number 47) for the initial real flux calculation is compared in figure 4.6. Relative errors are within  $10^{-5}$ . We can conclude that the initialization procedure implemented in SIMMER-III/PARTISN is equivalent to that of the SIMMER-III code.

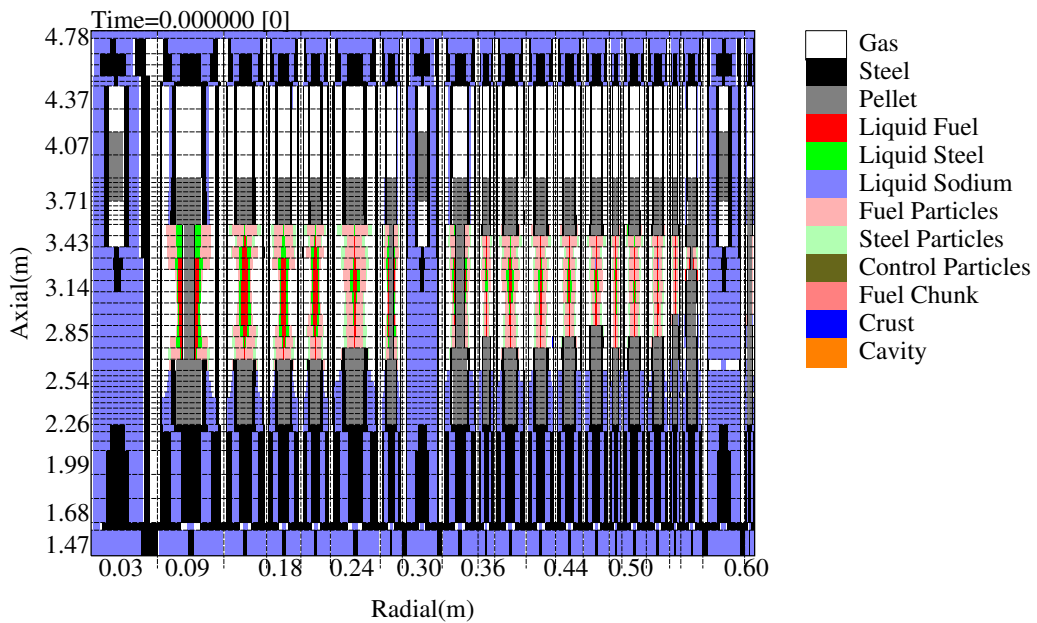


Figure 4.5: Initial state of ULOF calculation. A significant portion of the reactor is already disrupted. Here only the core domain is shown.

### Transient results

The transient is dominated by the fuel motion reactivity feedback. It is mainly fuel compaction that drives an insertion of positive reactivity up to almost the prompt-critical state. The subsequent power burst pushes then the fuel out of the active core region, hence the reactivity quickly becomes negative.

Figure 4.7 shows the temporal behavior computed by SIMMER-III of the fuel mass in the lower part of the core (i.e. between the bottom of the core at  $z = 2.26$  m and the core midplane at  $z = 3.06$  m, with reference to figure 4.5) and in the upper portion (i.e. from the core midplane up to the core top at  $z = 3.8$  m).

At first, this downwards relocation has a negative effect on reactivity. However, as the fuel accumulates in the bottom part of the core, fuel density increases, hence the reactivity rises, driving the first energetic event. The power burst vaporizes then the liquid fuel, generating a pressure build-up that disperse the fuel. As a consequence, this dispersive motion causes the last, sharp drop in the reactivity.

Reactivity traces between the two codes are compared in figure 4.8(a). As mentioned above, the initial reactivity drop is caused by fuel relocating into the lower part of the reactor. As this contribution turns positive, reactivity rises up to prompt-critical condition, where the subsequent power burst terminates the transient.

Comparison between the two codes is good. At the start of the transient, the two computations are very close, with the same shape and reactivity time steps being taken by the two codes. Reactivity differences are then as low as few pcm. However, during the transient, small differences in the fluxes (because of the different neutron solvers) inevitably accumulate and this leads to different shape and reactivity time steps. As a consequence, the two computations begin to diverge.

	SIMMER-III	SIMMER-III/PARTISN (relative error * 10 <sup>5</sup> )		SIMMER-III	SIMMER-III/PARTISN (relative error * 10 <sup>7</sup> )
$h_1$	1.2538213	0.319	$g_1$	0.30499946	0.984
$h_2$	1.1943112	0.586	$g_2$	0.33310310	0.600
$h_3$	1.2207454	0.492	$g_3$	0.51049876	0.784
$h_4$	1.2117959	0.495	$g_4$	0.87938245	0.682
$h_5$	1.2262910	0.408	$g_5$	2.2637115	0.442
$h_6$	1.2281781	0.489	$g_6$	3.8223214	0.785

(a) Data-adjustment factors for delayed-neutron fractions  $h_k$  and their relative errors.

(b) Data-adjustment factors for delayed-neutron precursors  $g_k$  and relative errors.

	SIMMER-III	SIMMER-III/PARTISN (relative error * 10 <sup>5</sup> )
$f$	1.0164349	1.28

(c) Velocity factors  $f_v$ .

Table 4.2: Comparison for data-adjustment factors for ULOF case.

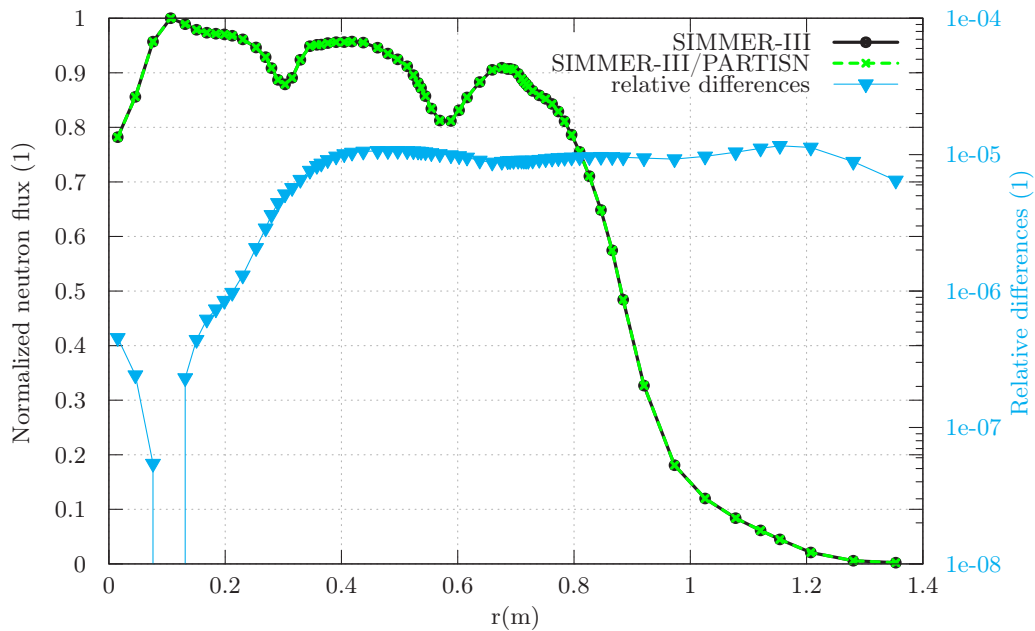


Figure 4.6: Comparison of the radial steady state fast flux distribution at core mid-plane ( $z \approx 3.06$  m) at the starting time of the LMFBR problem with a ULOF. Fluxes are normalized to the maximum SIMMER-III neutron flux value.



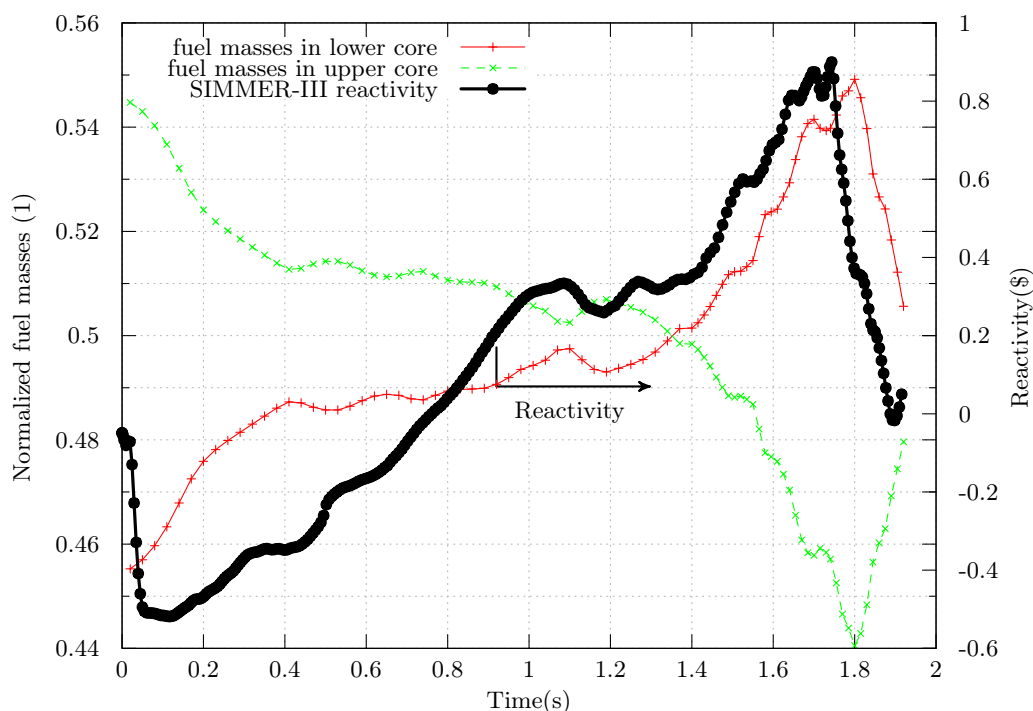
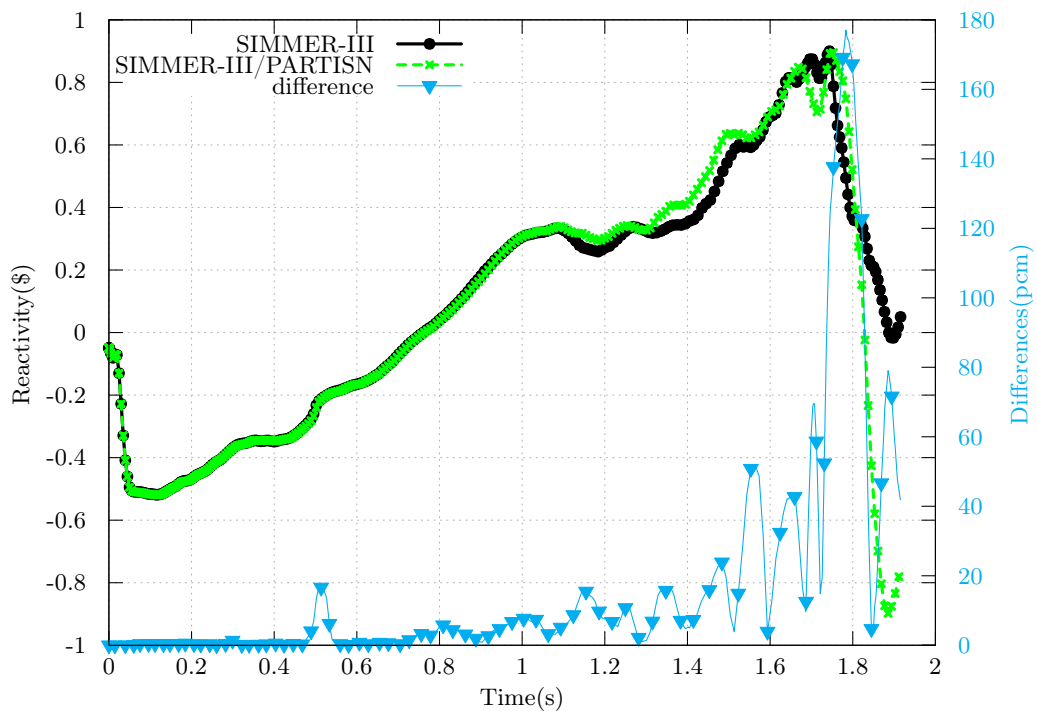


Figure 4.7: Temporal evolution of total fuel masses in the upper and lower part of the core for the ULOF case shown in figure 4.5. SIMMER-III reactivity as a function of time (black line) is plotted on the secondary y-axis.

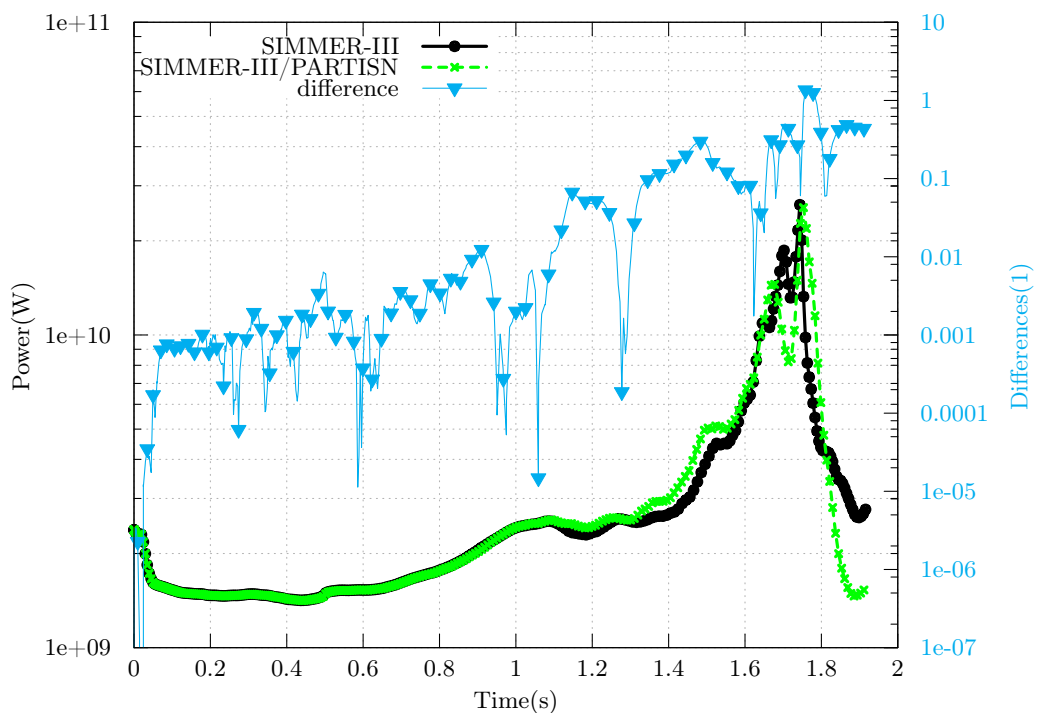
Reactivity differences reach values higher than 180 pcm. These differences seem to be caused by the different times at which reactivity peaks are predicted by the two codes, while the reactivity levels themselves are comparable. However, even at later stages of the transient, the main characteristics are reproduced: a rise in reactivity up to prompt-criticality with one sharp peak followed by a sudden reactivity decrease.

Agreement on the power is also good. Relative differences before the power peak are lower than 1%, see figure 4.8(b). At prompt-critical condition those differences inevitably become large, as the power peak is predicted at slight different times by the two codes. At this time, the difference rises to 100%.

The power level is however comparable: SIMMER-III calculates a maximum value of approximately 10.9 of the initial power; SIMMER-III/PARTISN a value of  $\approx 10.7$ . Axial flux traverses during the transient for ring number two are shown in figure 4.9(a). Values are normalized to the maximum value of the flux at time zero. At 1.0 s, SIMMER-III/PARTISN and SIMMER-III flux differences are still negligible and under 1%, see figure 4.9(b). After the power burst, fluxes decrease as fuel is discharged from the core. The instantaneous value of the flux depends on the whole reactivity history. Differences, especially after the power peak (i.e.  $t > 1.0$  s), are then caused by the slightly different reactivity history. Both codes show, however, a similar axial flux distortion, as fuel is relocated downward during the first part of the transient.

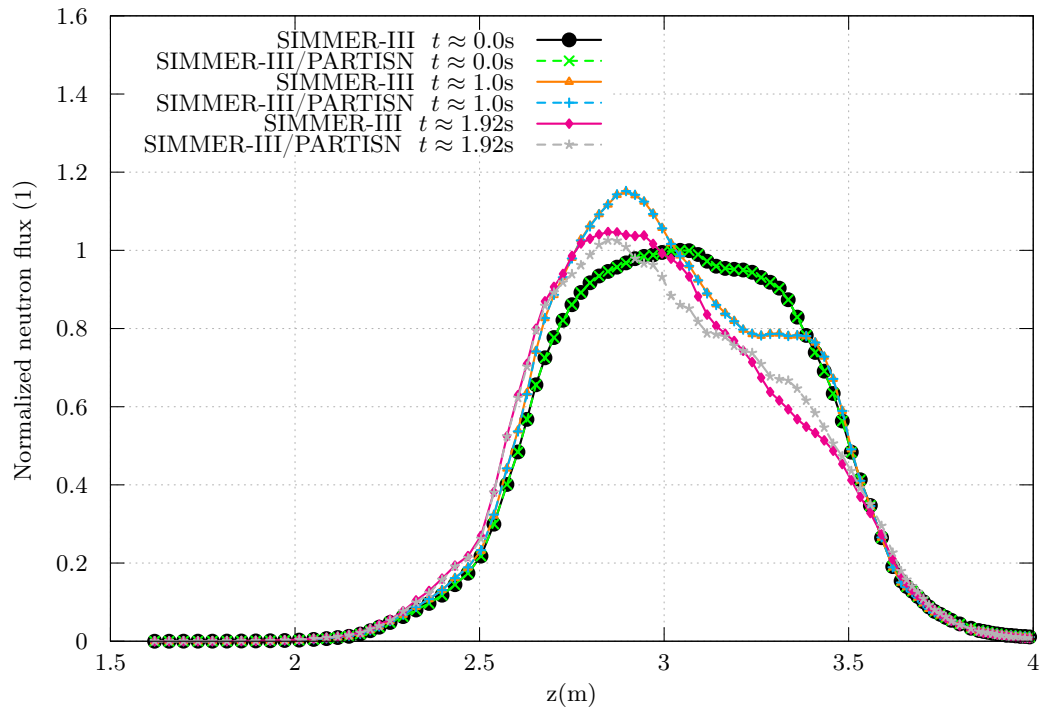


(a) Reactivity.

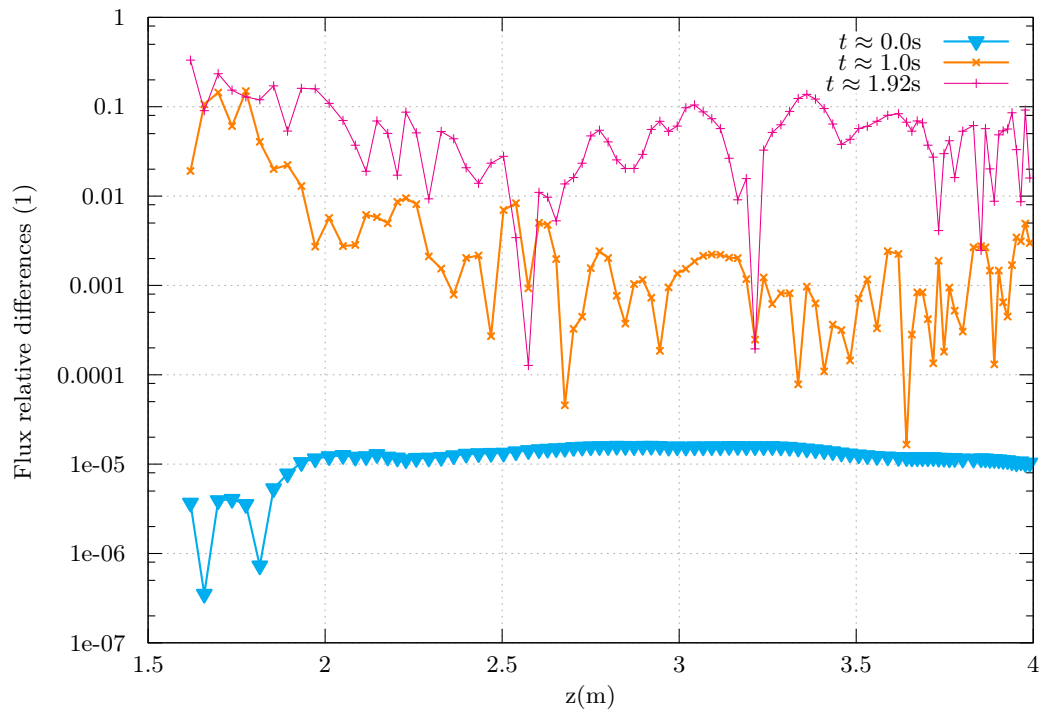


(b) Power.

Figure 4.8: SIMMER-III and SIMMER-III/PARTISN computed reactivity and power as a function of time for ULOF case.



(a) Axial flux traverses for ring number 2 at different times. Values are normalized to the maximum flux at time  $t = 0$ s. .



(b) Axial flux differences at different time steps. .

Figure 4.9: Comparison of calculated axial distributions for a ULOF case at several time steps (a). (b) Relative differences for SIMMER vs. SIMMER-III/PARTISN.

	SIMMER-III	SIMMER-III/PARTISN			
		1pe	2pe	4pe	8pe
Computational time( $10^4$ s)	1.638	1.650	1.530	1.562	1.574
Neutronics time(s)	240	232	200	166	147

Table 4.3: Comparison of parallel calculation for ULOF case.

### Investigation of speed up for parallel option

In this case it is difficult to draw any conclusion about the speedup and to perform a comparison. Here, in fact, differences in the fluxes between different parallel computations (e.g using two processors and then four for the same case) grow to a point where the time step selection algorithm is influenced. As a consequence, using different number of processors leads to different time steps: the total number of calls to the fluid-dynamics module and to the neutronic solver are different in each computation. In a strict sense, we are comparing two different computational problems.

We can get however a ballpark figure about the speedup by comparing the total computational time for the transient and the total time spent into the neutronic module. Those are reported in table 4.3. Serial SIMMER-III/PARTISN and SIMMER-III are almost equivalent. Only  $\approx 240$  s are spent into the neutronic module out of a total computational time of more than 4 h: we expect therefore little to no substantial improvement by going to the parallel version. Moreover, comparing a single shape update between serial and parallel versions, we find that they require the same amount of time: approximately 0.3 s. The small reduction in the total computational time found in table 4.3 seems to be mainly due to a lower number of fluid-dynamics, reactivity, and shape steps between the various calculations rather than a gain due to the parallel neutronic solver.

### 4.2.3 An ESFR Unprotected Loss of Flow

Here the focus is again on an ULOF transient but this time it is modeled from the initiation phase. The reactor is the ESFR reactor.

The accidents has already been investigated in the past with the standard SIMMER-III code by Chen et al. [2016].

#### Description of the European Sodium Fast Reactor (ESFR)

ESFR is a large sodium-cooled fast reactor concept being investigated by the European Union. Strong emphasis has been put into reducing the sodium void effect. The reactor underwent several design modifications aimed at mitigating this effect. Interested readers can refer to Rineiski et al. [2011] to get a detailed description of those modifications. The ESFR model described here represents the final design stage of that process.

The core consists of two parts – inner and outer cores – with different fuel enrichment, figure 4.10. Fuel is a mixture of U- and Pu- oxide. The fuel subassembly is a hexagonal tube containing 271 fuel pins, arranged in a triangular mesh. The control system includes

9 Diverse Shutdown Devices (DSDs) and 24 Control and Shutdown Devices (CSDs). They both contain natural boron carbide  $B_4C$ , with a different enrichment of  $B^{10}$ . The core is surrounded by several rings of reflector subassemblies. Above the core is a large Na plenum followed by a neutron-absorber layer. There is no radial blanket but only an axial one, placed below the core. It is made of depleted uranium oxide (95% vol. fraction) and americium oxide (5% vol. fraction). The evaluated average power density is  $206 \text{ W/cm}^3$ .

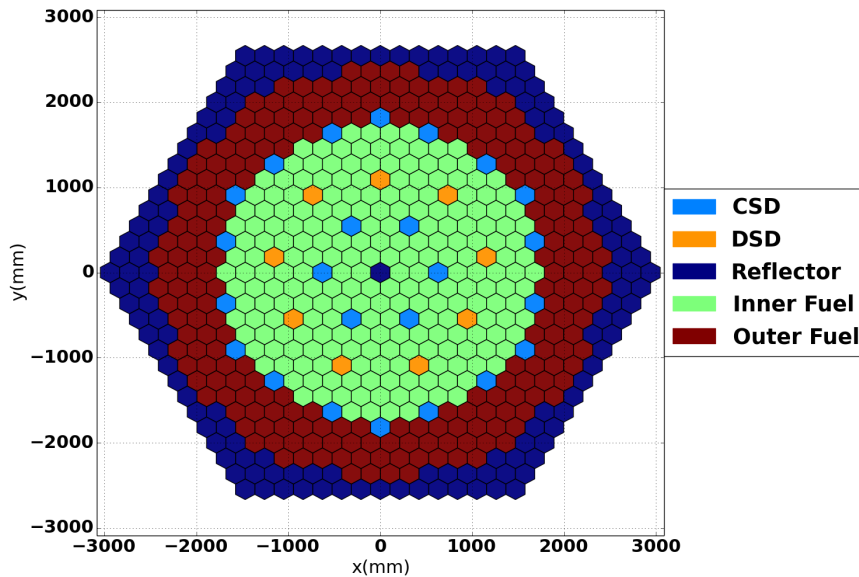


Figure 4.10: Radial configuration of the ESRF reactor. (DSD Diverse Shutdown Device, CSD Control Shutdown Device).

The two-dimensional cylindrical ( $r$ - $z$ ) SIMMER-III model is depicted in figure 4.11. The core has a height of 1 m, extending from  $z = 5 \text{ m}$  to  $z = 6 \text{ m}$ . Radially, the core zone ends at  $r = 2.32 \text{ m}$ , followed by a reflector ring and two further rings for shielding. Control rods are represented by three absorbing rings located at 0.61, 1.17, and 1.7 m from core center.

### Description of the Accident

The accident is initiated by tripping the primary pumps. The imposed pump coastdown is shown in figure 4.12, along with the computed sodium mass flow rate.

Since the reactor has a positive sodium void effect, the reduction of the mass flow rate inserts positive reactivity, increases the reactor power and hence the sodium and fuel temperatures. Rising temperature inserts then a negative reactivity, because of the Doppler effect. For the first few seconds of the transient, these opposing contributions balance each other. As a consequence, the net reactivity remains low and around zero.

However, because of the continuous reduction of the mass flow rate, the sodium reaches its boiling temperature. This phenomena reduces further the sodium density, and therefore causes a strong, positive reactivity insertion. The Doppler effect can no longer counterbalance this effect and the consequence is the first energetic event.

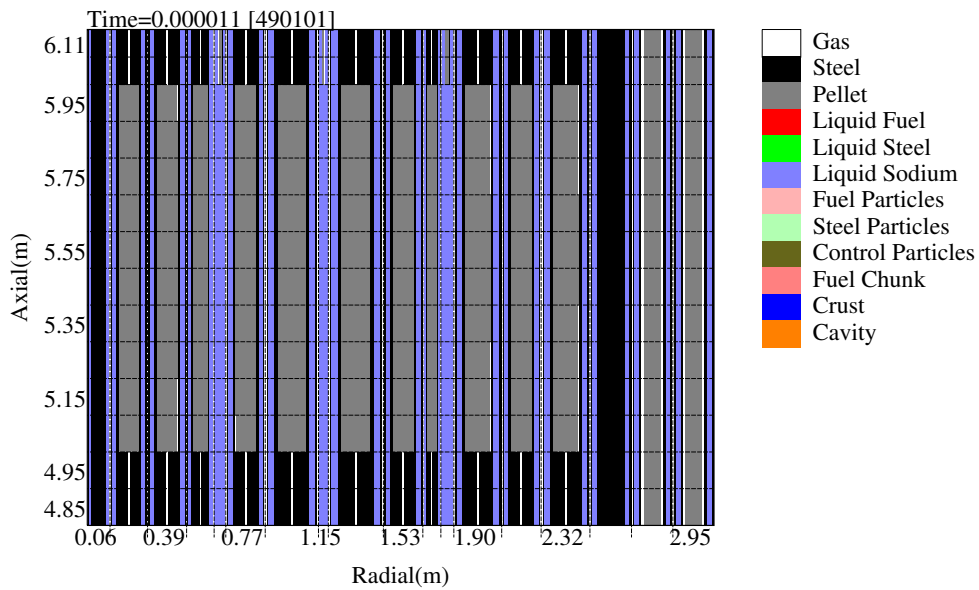


Figure 4.11: Two-dimensional SIMMER-III core model of the ESFR used for simulating the Unprotected Loss of Flow transient.

The figure 4.13 shows the time behavior of the reactivity and the (normalized) reactor power as computed by the two codes. At the reactivity peak, difference in the reactivity level is only 10 pcm, with the SIMMER-III/PARTISN predicting the higher value. However, since the reactor is prompt-critical, even such a small deviation leads to 30% difference in the power level.

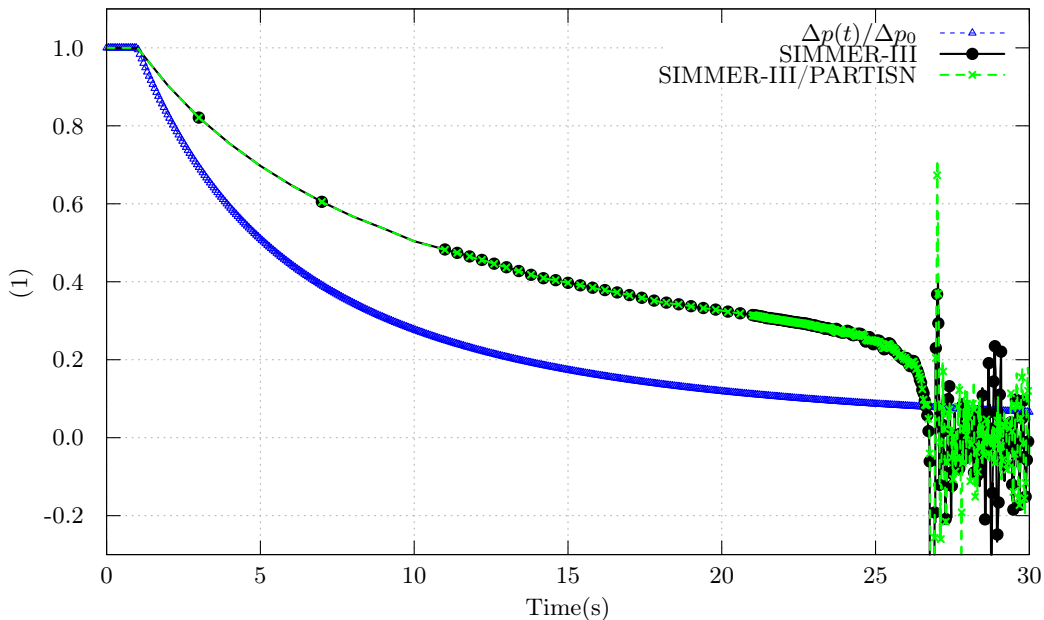


Figure 4.12: Imposed pressure coastdown and computed mass flow rate (normalized) for the ESFR Unprotected Loss of Flow case.

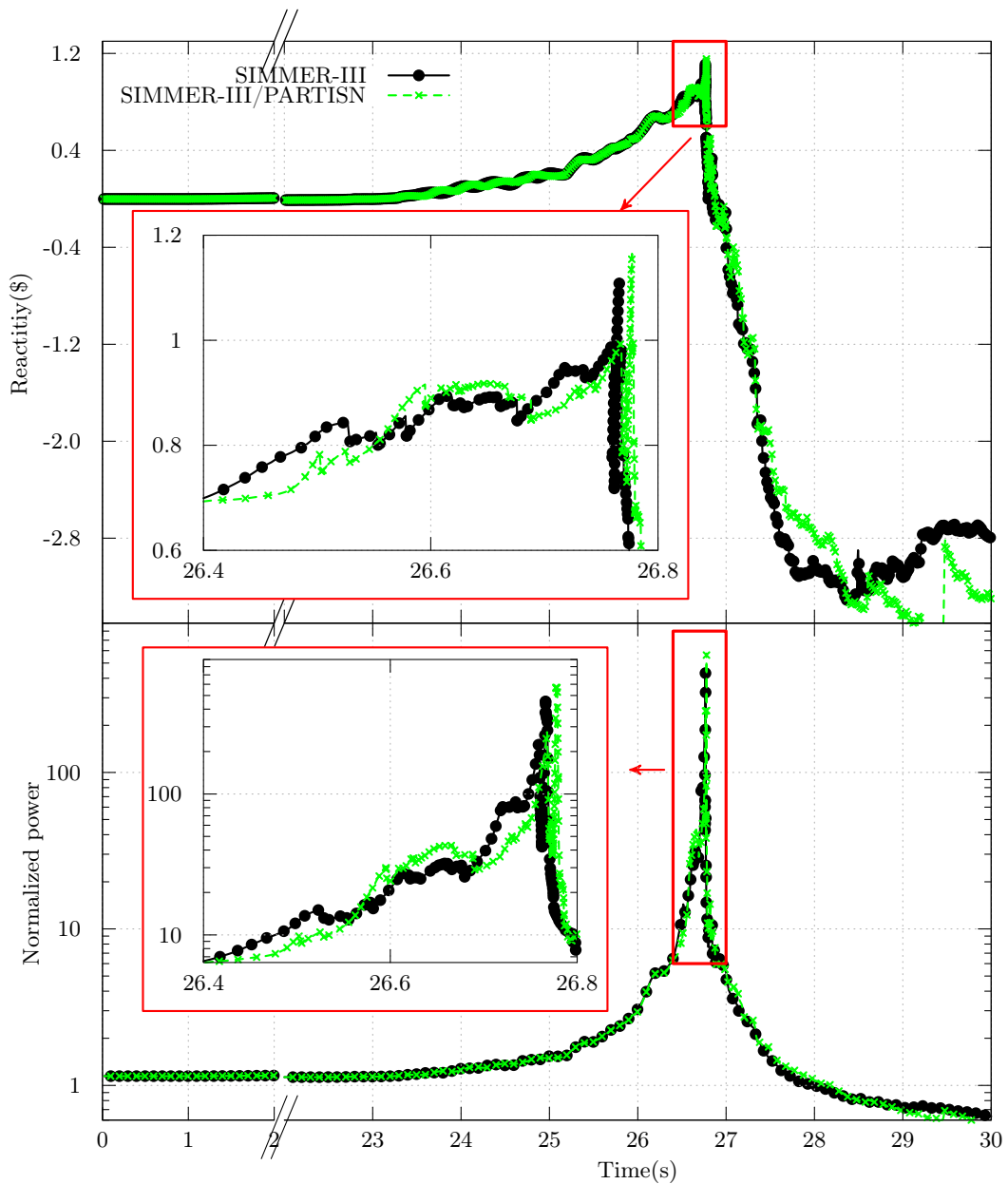


Figure 4.13: Temporal evolution of reactivity (top) and power (bottom) as evaluated by the two codes for the ESRF Unprotected Loss of Flow case.

### Sodium boiling

The SIMMER-III code predicts the onset of sodium boiling in channel 3 approximately at the exit of the core (axial elevation  $z = 6$  m). This is clearly indicated by the sharp rise in the void fraction  $\alpha$  (i.e. the fraction of the cell volume that is occupied by the gas phase) at that location between 21 and 22 s in the top graph in figure 4.14. At this time, the sodium temperature is almost 1200 K, as indicated by the second graph in the same figure. Results from the coupled code predicts exactly the same pattern: void fractions and sodium temperatures are replicated with no significant differences compared to SIMMER-III.

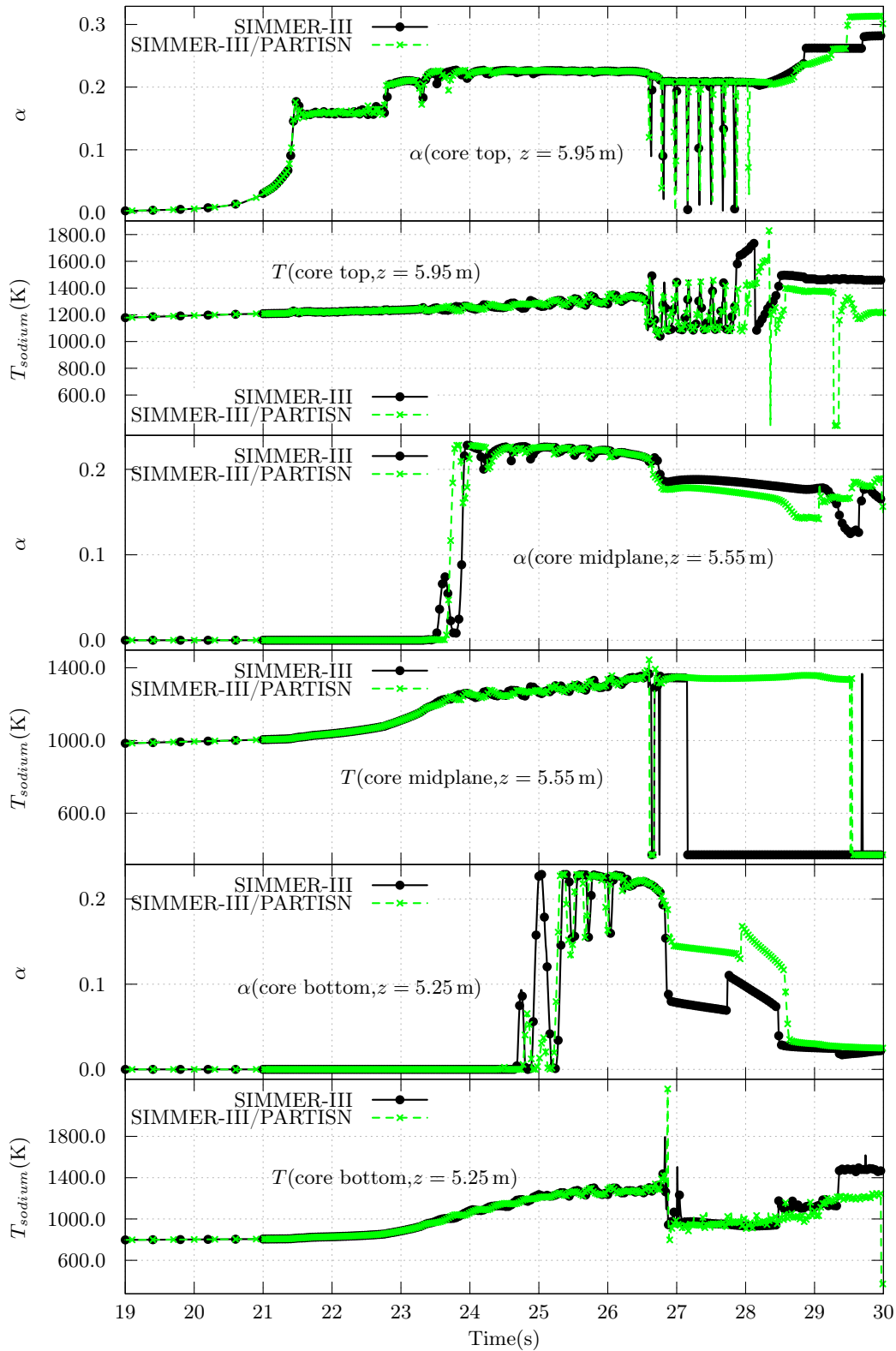


Figure 4.14: Temporal evolution of void fraction  $\alpha$  and sodium temperatures  $T_{sodium}$  in ring 3 at three different axial elevations (i.e. at core top, at core midplane and at core bottom) for the ESRF Unprotected Loss of Flow case.



The boiling region tends then to extend axially. Between 23 and 24 s, it reaches the core midplane at an elevation of  $z = 5.55$  m, as it is indicated by the sharp rise in the void fraction  $\alpha$  in the third graph of figure 4.14. The bottom of the core is then reached between 24 and 25 s, see fifth and sixth graphs in figure 4.14.

On the other hand, the upwards extension of the boiling region is somewhat hampered by the sub-cooled sodium presents above the core. The axial extension of the boiling region in ring 3 as a function of time is depicted in figure 4.15. The same pattern is predicted by both SIMMER-III and SIMMER-III/PARTISN.

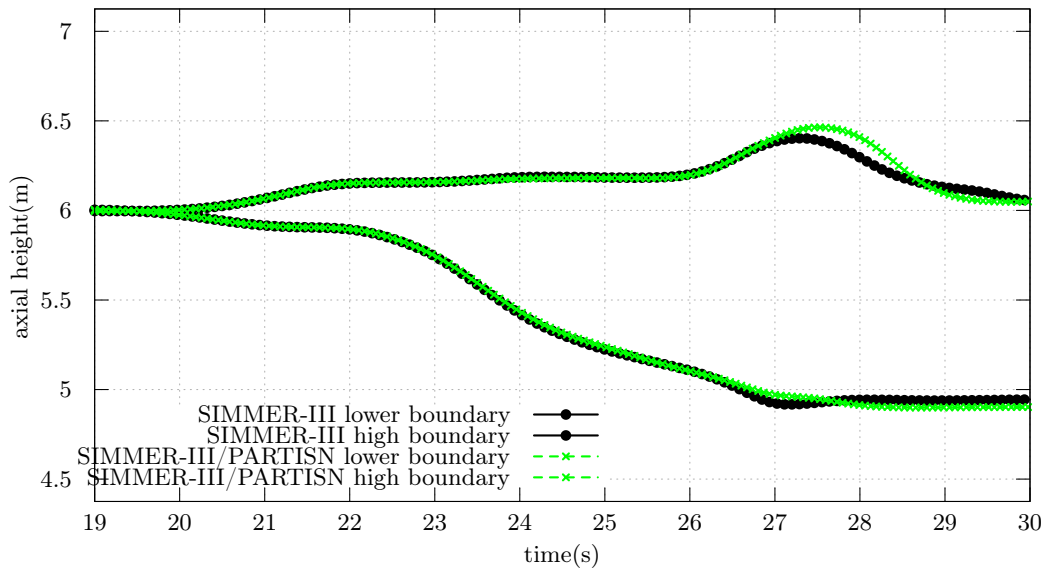


Figure 4.15: Axial extension of the boiling region in ring 3 as computed by the two codes for the ESRF Unprotected Loss Of Flow case.

The boiling extends then radially to neighboring rings. A confront between SIMMER-III and SIMMER-III/PARTISN at the power peak reveals that sodium boiling is occurring in every ring in the inner core, except in the control rod guide tubes. This comparison shows that SIMMER-III/PARTISN can well represent the onset of sodium boiling and the subsequent expansion of the boiling region when compared to SIMMER-III.

### Core disruption

The power increase leads to pin failures. The first pin to fail is in ring 3 at core midplane (i.e. elevation  $z = 5.6$  m). It happens just before the power peak is reached: SIMMER-III predicts it at  $t = 26.76$  s, SIMMER-III/PARTISN at  $t = 26.77$  s, when the local clad temperature exceed the melting temperature of steel (peak clad temperature 1720 K). After this point, disruption quickly spreads to other sub-assemblies.

A sketch of the status of the reactor at three different time points, as evaluated by the SIMMER-III code, is in figure 4.16(a), (b), and (c). This sequence is reproduced by the coupled SIMMER-III/PARTISN code with minor deviations only. Fuel is discharged into the coolant flow and can move axially (the can walls are still standing, thus preventing any radial movement of the fuel).

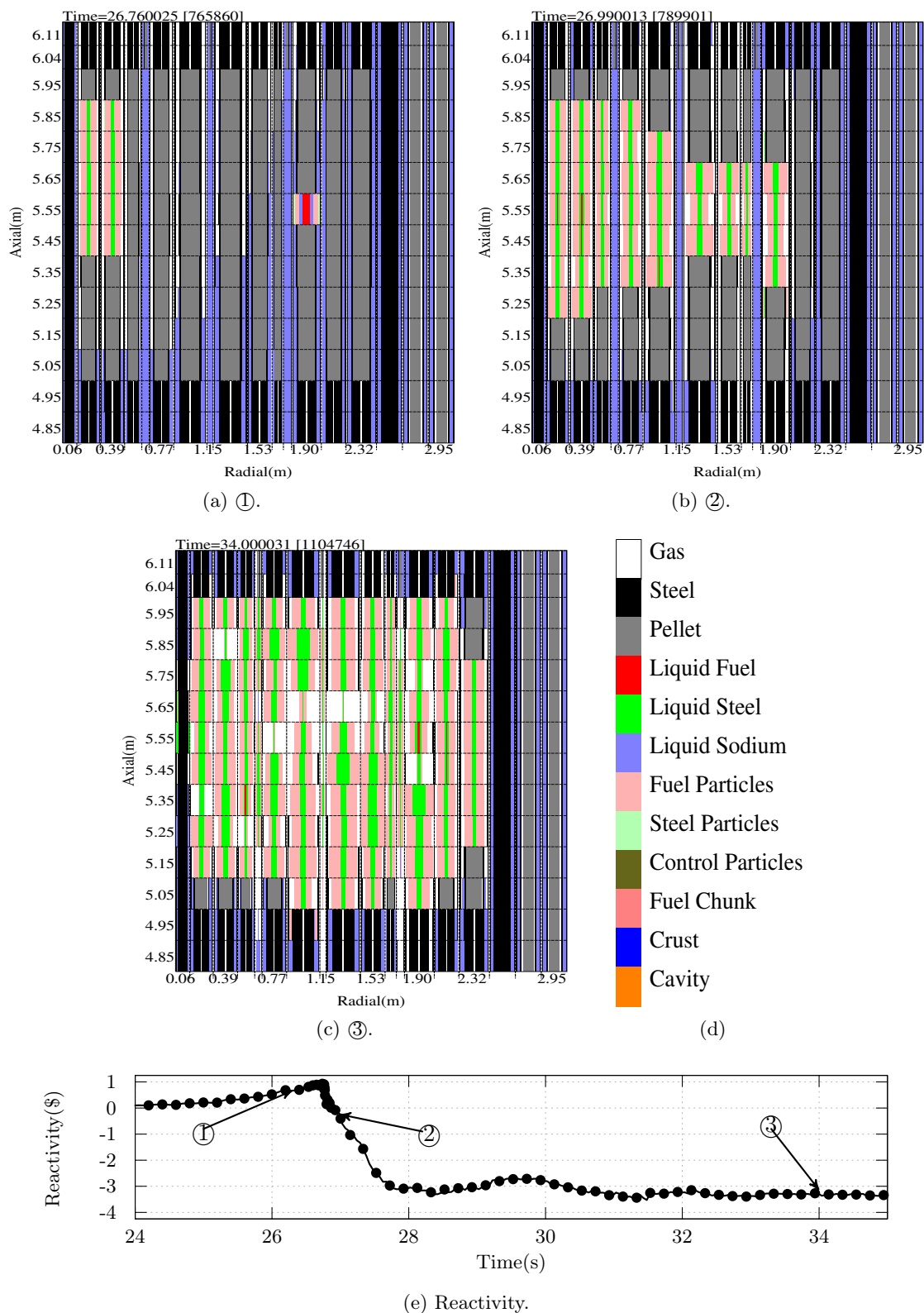


Figure 4.16: Material distributions at various times as evaluated by the SIMMER-III code.

Fuel movement brings a strong, negative reactivity insertion that terminates the first power excursion. Investigation of fuel relocation patterns indicates that fuel is discharged (both downwards and upwards) from the core. Figure 4.17 shows the fractional change in the fuel mass contained in the core. From the graph, it is clear that no sooner the fuel is discharged from the core, the reactivity drops. The same pattern is calculated by the two codes.

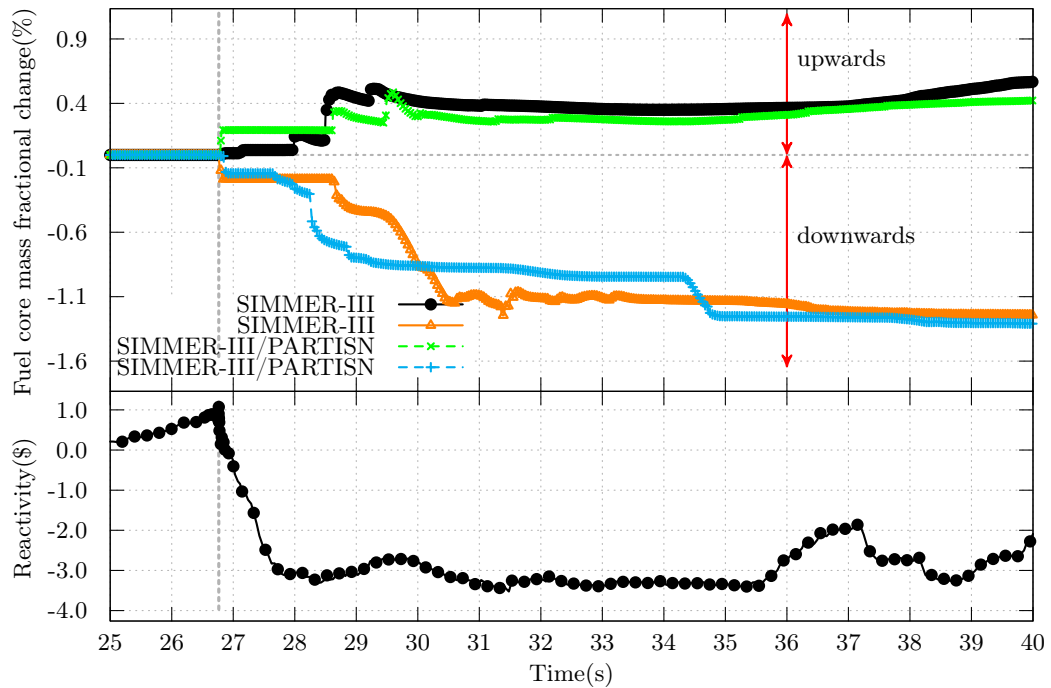


Figure 4.17: Fractional change of core fuel mass as a function of time during ESFR ULOF case.

This is an important point, since the reactivity associated with fuel movement can be quite large, and a correct prediction of fuel movement is essential in CDAs. At this point, no liquid sodium is present anymore in the reactor and cooling is prevented: the can walls fail one after the other. A sketch of the reactor state at  $t = 34$  s, figure 4.16(c) reveals that a core-wide pool, consisting of liquid fuel and steel, is formed.

### Transition Phase

Both codes predict that only 2% of the initial fuel inventory has been discharged from the core region at the start of the Transition Phase (TP). This fuel is present in form of solid fuel particles, as its temperature is still lower than its melting point (at  $\approx 3040$  K). As fuel solid fuel particles have a high density, they relocate mainly to the bottom of the core. Also, because of their viscosity, they effectively block any downwards path available for fuel discharge through the lower reactor structures.

Even control rod guide tubes, which have a large cross-sectional area for fuel to flow are blocked. As a result, fuel inventory remains constant for the first seconds in the TP. In such a scenario, with a large fuel pool inventory and no discharge paths, recriticality events are very likely. The evolution in time of reactivity and power in the TP is illustrated in the

first two graphs in figure 4.18. As can be seen, SIMMER-III predicts a recriticality event, at  $t = 45.11$  s. These are actually 3 power bursts in a very short sequence. The first results from a spike in the pressure ( $p = 22$  bar) caused by an increase in the partial sodium vapor pressure. At this time, reactivity is small ( $-1.4\%$ ) so is the power. This pressure is generated at ring 10 and therefore pushes the solid fuel particles towards the core center, inserting positive reactivity and causing the first power peak, as illustrated in figure 4.19(a).

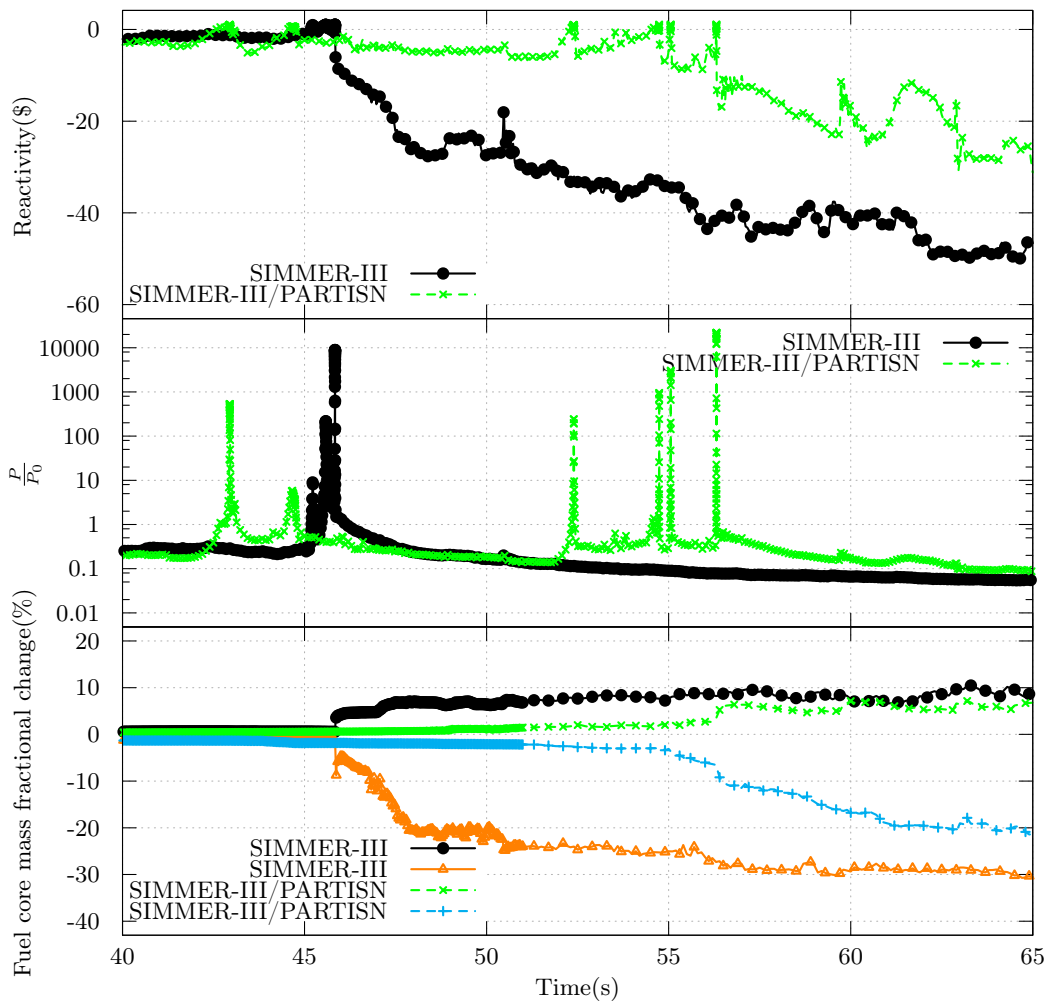


Figure 4.18: Reactivity (top), normalized power level (middle), and fractional change in the fuel core mass (bottom) computed by the two codes as function of time for the TP of the ESRF case.

Only a modest quantity of energy is produced, since, as the fuel particles rebound on the core center and reverse their motion (i.e. outwards again), reactivity and power quickly decrease again. Fuel particle temperatures are still below the melting temperature of fuel, and, essentially, no fuel is yet in liquid form. Another inward movement, see figure 4.19(b), causes then the second power peak with a power level of 1000. The energy released is sufficient to melt almost all of the solid fuel particles, turning them into liquid fuel. Compared to solid particles, liquids have a smaller viscosity and this enhances the sloshing. The outwards motion is however still not able to discharge fuel. In fact, the layer of solid fuel particles is still blocking any discharging path. After rebounding against the

core surroundings, this liquid fuel gathers again at the center of the core. This time the recriticality event is much more severe, with a power level of 10000. The power burst is able to melt the blocking layer of solid fuel particles and the resulting outwards motion can finally discharge fuel from the core, figure 4.19(c). Fuel escapes mainly through the lower portions of the control rod guide tubes, which, compared to other sub-assemblies, contain a small quantity of steel and thus have a larger cross-sectional area for liquid fuel to flow through. Specifically, fuel is discharged from ring 8 and ring 12. As a results, 20% of the total fuel mass is displaced downwards and 10% upwards, as depicted in the third graph of figure 4.18. The fuel inventory is then reduced, reactivity drops to  $-40\%$  and no further recriticality events are computed in the accidental sequence.

SIMMER-III/PARTISN shows instead a different transient. Several recriticality events are computed as shown in figure 4.18. These events have a power level of 530, 6, 240, 1000, 3000, and 22000. Only the last one is able to discharge fuel and the transient terminates.

### Comments

This ULOF case for ESFR reactor shows that the agreement between the two code is rather good for the initiation phase of the accident. The coupled SIMMER-III/PARTISN code is able to predict the same sequence of events as SIMMER-III: first the onset of sodium boiling, then the power increase that leads to pin disruptions, hence the fuel relocation feedback that terminates that energetic event, and the final formation of a core-wide pool. The case highlights, however, how sensible is the outcome of a HCDA to variations in the value of reactivity. Here, a deviation of only 18 pcm at the prompt-critical state determines a 30% difference in the power level predicted by the two codes. Such different energetic events lead to different temperatures and spatial fuel distributions at the start of the TP, and hence to a diverse evolution of the pool behavior.

### 4.2.4 Three-Dimensional Space-Time Neutronics Problem

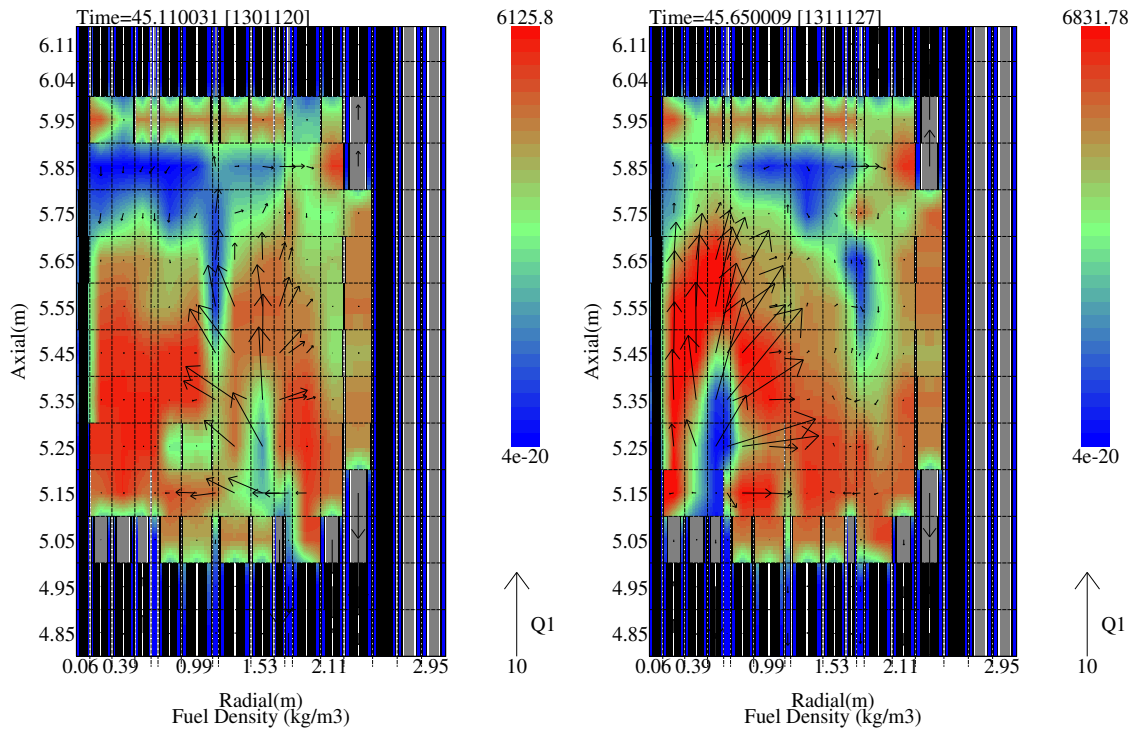
The focus is here directed to verify the SIMMER-IV/PARTISN code on the Cartesian, three-dimensional version of the STN problem, see Yamano et al. [2003b].

#### Description of the case

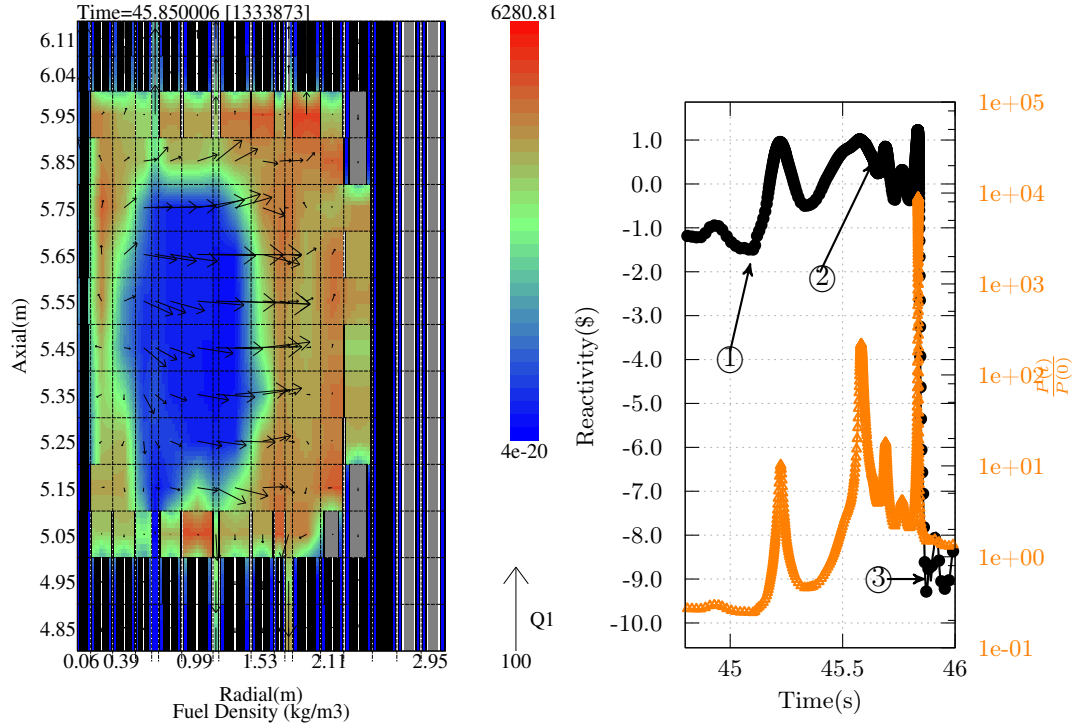
The set up of the case is similar to the two-dimensional version presented in section 4.2.1. However, the SIMMER-IV model has a three dimensional Cartesian geometry with a  $42 \times 32 \times 42$  neutronic mesh, 18 neutron energy groups, and  $S_4$ .

#### Transient results

As described before, the fissile lump brings the reactor rapidly to a prompt-critical state, from which it returns to sub-criticality as the fissile lump travels away from the core. A comparison between SIMMER-IV and SIMMER-IV/PARTISN shows again a perfect agreement, as illustrated in figure 4.20. There we note again that towards the end of the transient, the time steps between the two computations are slightly different.



(a) ① Fuel density distribution before the first recriticality event ( $t = 45.11$  s), fuel is pushed towards core center .  
 (b) ② Fuel density distribution at the second power peak ( $t = 45.65$  s), fuel is being pushed towards core periphery .



(c) ③ Fuel density distribution after the third power peak ( $t = 45.85$  s), as the fuel is pushed outwards, it escapes through CRCTs .  
 (d) Reactivity and normalized power level .

Figure 4.19: ESRF ULOF case, TP phase: fuel density and fuel velocity field .

Reactivity differences are smaller than a few pcm and increase only towards the end of the transient, when the time steps between the two computations start to differ, as shown in figure 4.20(a). Amplitude relative differences are well below a few percent for most of the transient duration, as depicted in figure 4.20(b).

### Investigation of speedup

SIMMER-IV and serial (i.e. one processor) SIMMER-IV/PARTISN calculations are almost equivalent, see table 4.4. The fraction of the time spent in the neutron solver is  $\approx 55\%$ , so the maximum achievable speedup is  $\approx 2.2$ . I got 1.64 with eight processors, reducing the total computational time by 40%: from 1680s to 1000s.

Interestingly, the performance with 16 processors is worse. It must be noted however that the machine I used had exactly 16 processors. It is therefore a situation in which there are more processes (i.e. one for SIMMER-IV plus 16 for PARTISN against 16 core available) than processors, and that might cause a bottleneck: had I had more processors available, I might have achieved higher speedups.

	SIMMER-IV	SIMMER-IV/PARTISN				
		1pe	2pe	4pe	8pe	16pe
Computational time( $10^3$ s)	1.77	1.68	1.27	1.13	1.02	1.04
Neutronics time( $10^2$ s)	11.9	9.20	4.69	3.39	2.91	2.81
Fraction(%)	67	55	37	30	28	27
Speed-up	-	-	1.32	1.49	1.64	1.62

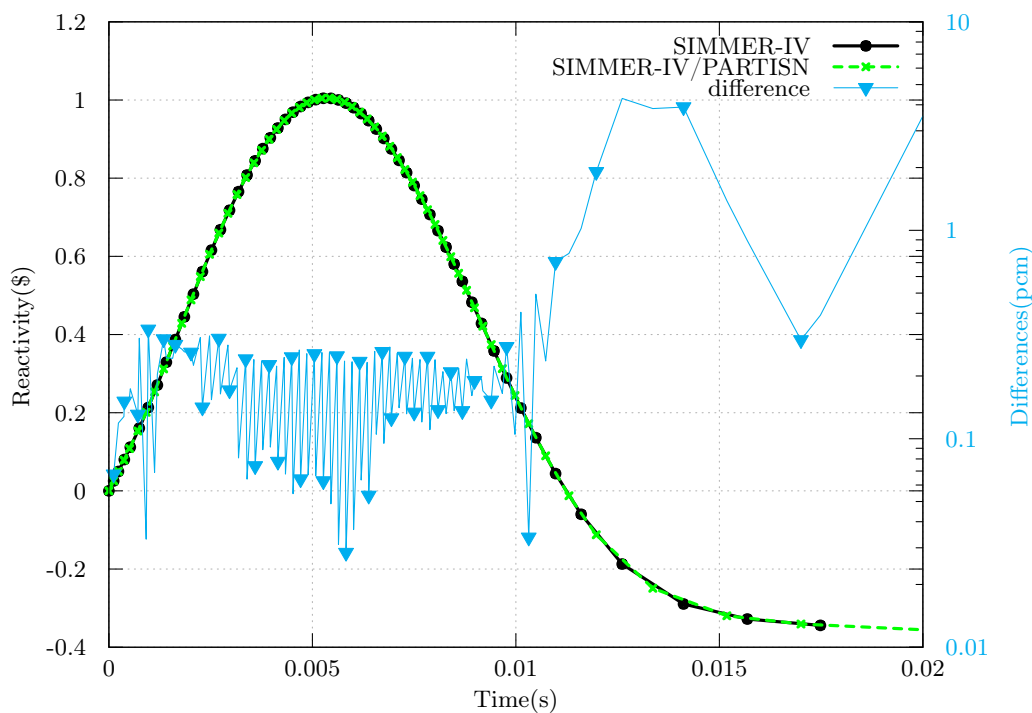
Table 4.4: Investigation of speedup for the three-dimensional STN case.

## 4.3 EBR-II Shutdown Heat Removal Experiments

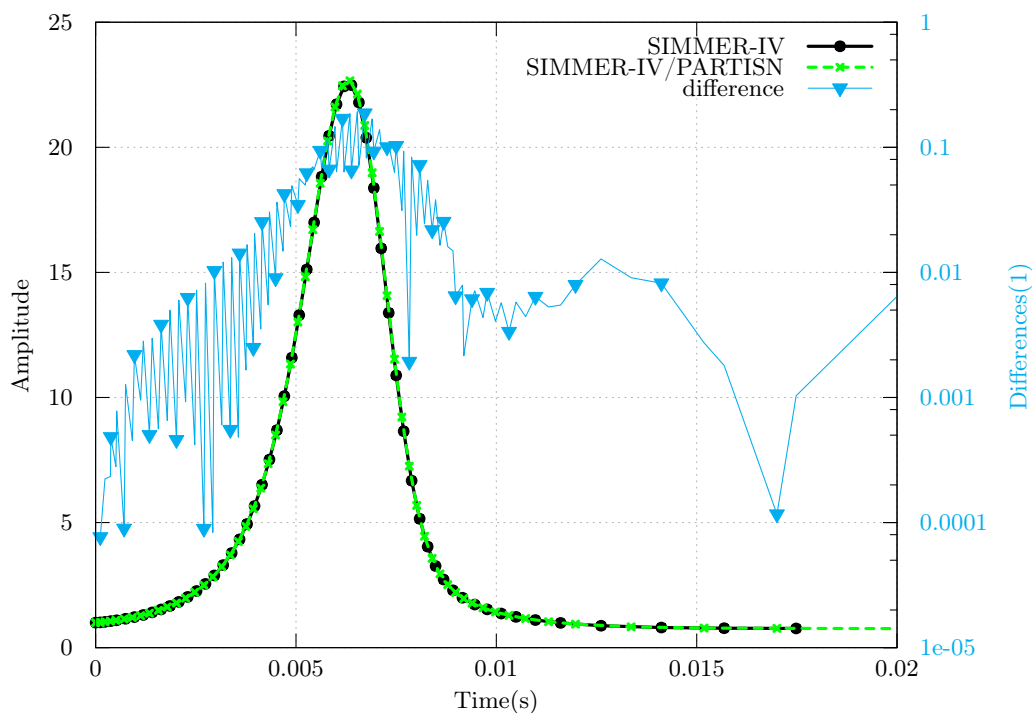
Up to this point, the SIMMER/PARTISN coupling has been verified against the normal SIMMER code. Validation against experimental data is necessary for any new code development. As KIT, I participated into the IAEA Collaborative Research Project “Benchmark Analysis of an EBR-II Shutdown Heat Removal Test” described by Briggs et al. [2013]. Set-up by IAEA and ANL, the project invited organizations to simulate some specific EBR-II transients with their state-of-the-art codes, providing then experimental data to validate different simulation tools. Its major aim was to verify, to validate, and to quantify uncertainties on neutronic and thermal-hydraulics codes used for fast reactor modeling.

### 4.3.1 EBR-II Reactor Description

The EBR-II reactor had been operated by ANL from 1964 to 1994, a detailed description of the reactor is available in Koch [2008]. It is a heterogeneous, sodium-cooled, fast breeder reactor. Thermal power is approximately 60 MW with an electric output of 20 MW.



(a) Reactivity.



(b) Amplitude.

Figure 4.20: Comparison between SIMMER-IV and SIMMER-IV/PARTISN for the three-dimensional STN case. Temporal evolution of reactivity (a) and amplitude (b).



The core can host 637 hexagonal subassemblies arranged in three regions: inner core, outer core, and blanket region. The fuel is metallic: a special alloy composed of 95% in weight of Uranium plus other additional metals. A view of the reactor is shown in figure 4.21.

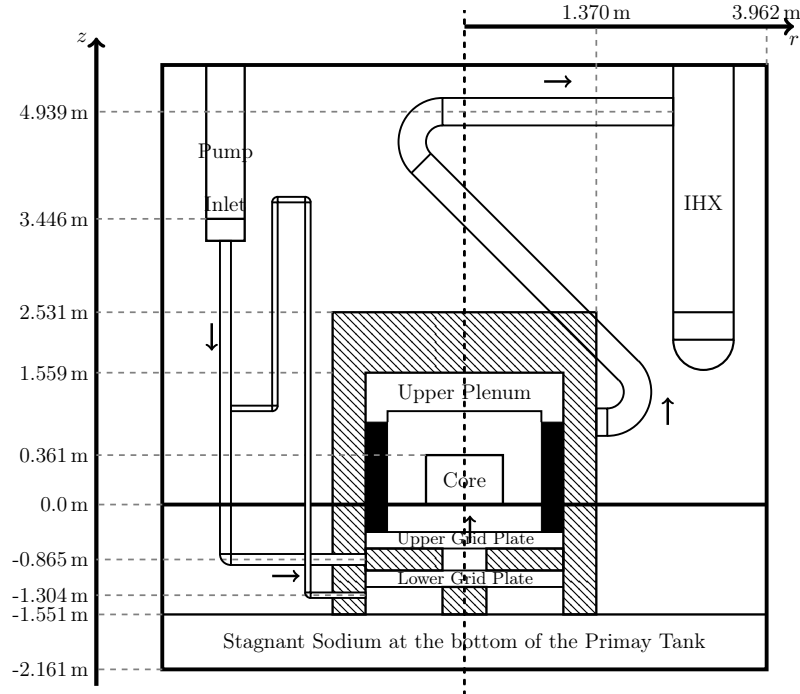


Figure 4.21: Schematic vertical cut through the EBR-II reactor building. Major components and their dimensions are also shown.

Sodium flows through the core and collected in the upper plenum. Then, through a z-shaped pipe, it reaches the IHX. After the IHX, cold sodium is discharged into the sodium pool. Two primary pumps provide sodium circulation. Pump inlets are high in the sodium pool, sodium is pumped into the core via the high and low pressure inlet plena. All is immersed in a rather capable sodium pool.

Several tests had been conducted under the EBR-II Shutdown Heat Removal Testing (SHRT) program between 1984 and 1986. The objective of the program was to demonstrate the passive reactor shutdown in response to protected and unprotected transients.

I will focus in particular on one ULOF test proposed within the benchmark and identified as SHRT-45R. From full power and full flow, this transient is initiated by tripping all primary pumps to simulate the loss of flow. The plant protection system had been deactivated to prevent it from intervening. Temperatures rose but to an acceptable level, before passive reactivity feedbacks – mainly the sodium void effect, thermal expansion, and Doppler effect – were sufficient to bring the reactor to a safe, cold condition. No damage to any fuel pin was recorded. In term of accidental phases terminology, see section 2, this test is an example of an initiation phase modeling.

### 4.3.2 EBR-II Shutdown Heat Removal Test SHRT-45R

A full description of the SHRT-45R neutronics benchmark is given by Briggs et al. [2013]. The planar core layout of this test is shown in figure 4.22. A great variety of subassemblies is loaded into the core. The reactor is composed of 16 rings of hexagonal subassemblies, with lattice pitch  $\approx 5.9$  cm. The core region comprises the first seven rings. The equivalent outer core radius is only  $\approx 35$  cm, the active core height is  $\approx 36$  cm.

Two different types of fuel subassemblies are used: full driver ones, with 91 pins filled with fuel, and half-driver ones, in which about half of the 91 pins are filled with steel instead of fuel. Driver fuel subassemblies are loaded into the innermost seven rings.

Metallic fuel – a so-called U-fissium alloy, see Briggs et al. [2013] – with highly enriched Uranium (67% in  $U^{235}$ ) is employed. Ring 3 hosts two safety rods (S), eight control rods (C) are in ring number 5.

Control rods (C) contain a fuel section in the central part, below a boron carbide pin bundle section. Therefore, when a control rod is lowered, the fuel section is pushed out of the core, while the upper neutron poison section enters the core. The axial position of the control rods is such that the bottom of their fuel section is 28.97 cm below that of the other fuel driver subassemblies. Similarly, the bottom of the fuel section of the safety rods is 23.89 cm below that of the fuel driver subassemblies. Rings 8 to 16 contain outer blanket and reflector subassemblies.

The reactor was heavily instrumented. Two instrumented subassemblies are located in ring 5, they are called XX09 and XX10. These subassemblies have mass flow meters and several thermocouples at different axial heights. Prediction of their signals was one of the main task proposed in the benchmark.

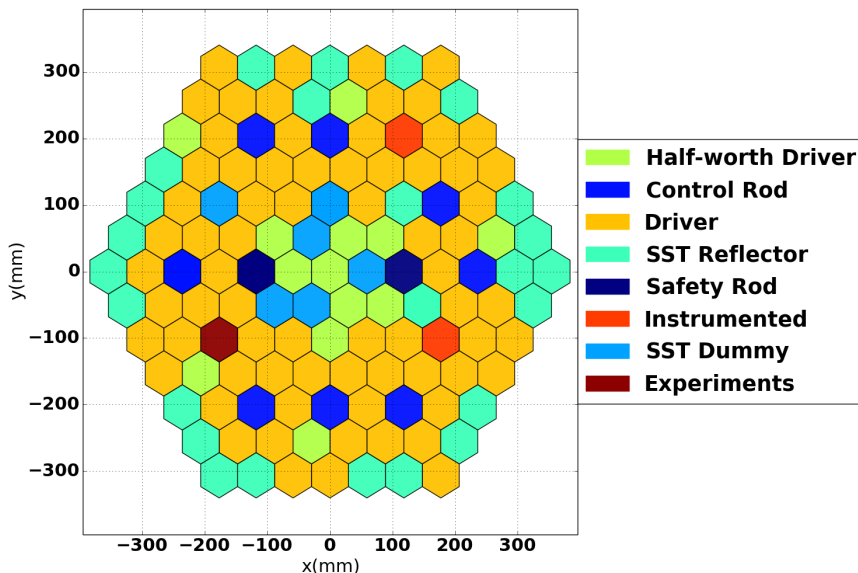


Figure 4.22: Schematic drawing of the EBR-II SHRT-45R core layout (only the first seven rings) for the validation case in section 4.3.1.

### 4.3.3 EBR-II SHRT-45R SIMMER-III Model

The transient is modeled with SIMMER-III. A two-dimensional cylindricized version of the reactor is shown in figure 4.23. The core extends radially between fluid-dynamics cell 1 to cell 36, i.e. from core centerline up to the radial neutron shield. Axially there are 42 cells, modeling from the upper grid plate all the way up to the upper plenum. The neutronic mesh is very fine. That is necessary to avoid many flux fixups in the neutronic solver. In total, the neutronic mesh has 116 radial and 138 axial meshes.

All reactor pipings are modeled with “virtual walls”. They are virtual boundaries: they have no mass themselves, but they allow no mass, momentum, and energy flux across. The z-pipe is modeled by a succession of horizontal and vertical tracts. The simplest heat exchanger model is used, discharging cold sodium at the same temperature of the sodium pool. Because of SIMMER-III geometric limitations, I could use only one pump. I defined therefore an “equivalent” pump placed at the core inlet. This equivalent pump is defined as to provide the same total pressure-flowrate characteristics as provided by the two primary pumps. As the core presents no sign of any azimuthally symmetry, a two dimensional cylindric model can yield only a gross approximation.

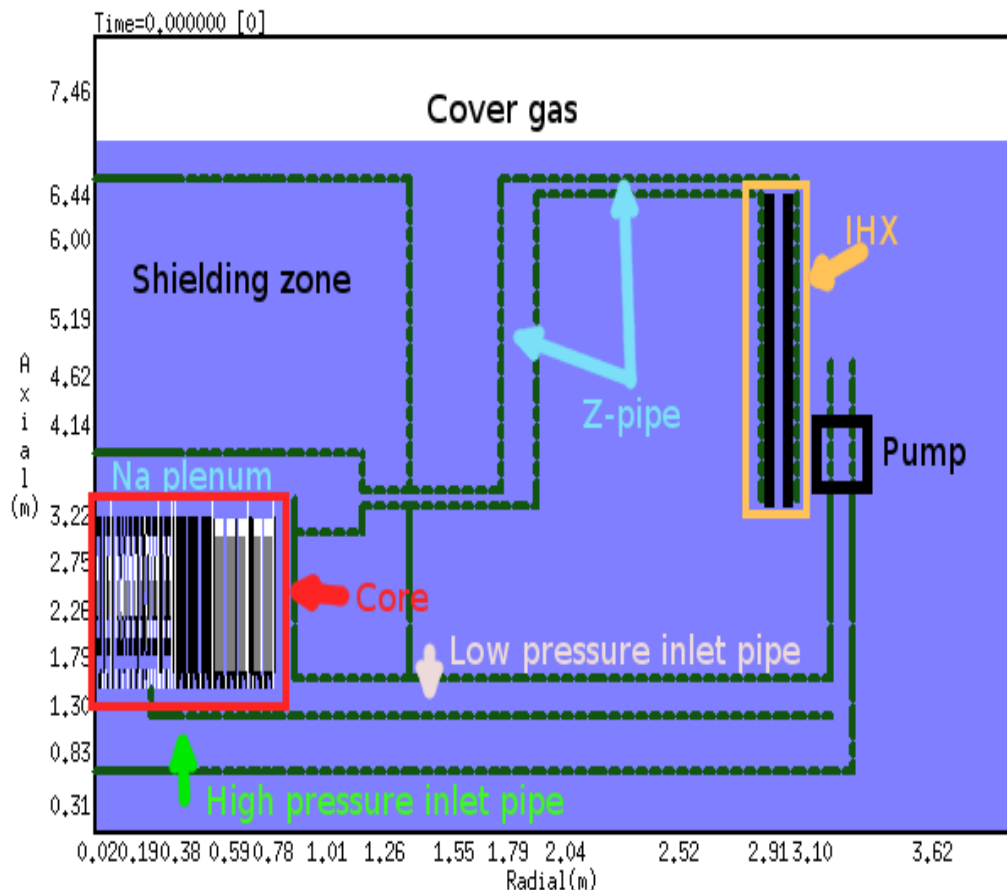


Figure 4.23: SIMMER-III two dimensional model (r-z) for the EBR-II SHRT-45R validation.

### 4.3.4 EBR-II SHRT-45R Validation

Here I show the results of the computations. Before running the transient, I studied some neutronic parameters at the initial steady-state configuration, in order to characterize reactivity feedbacks of the reactor. Those are presented in the next section. Next are results from the ULOF calculation, see section 4.3.4.

#### Steady-State results

The transient is run from steady-state condition, therefore the initial multiplication factor is one. Computed multiplication factor in the SIMMER-III model is just 35 pcm above unity, as indicated in table 4.5.

Reactivity effect	value
Multiplication factor	1.00036
Doppler constant (pcm)	-60.0
Sodium density coefficient (pcm)	-489.5
Radial expansion coefficient (pcm)	-3146.8
Axial expansion coefficient (pcm)	-1002.7

Table 4.5: Reactivity coefficients computed at steady-state for the EBR-II SHRT-45R.

Doppler effect is negative, as expected, but it is rather small, possibly due to high  $^{235}\text{U}$  and relatively low  $^{238}\text{U}$  content. This effect is evaluated by an exact perturbation theory doubling the fuel temperature in the core and in the blanket.

The sodium density coefficient is evaluated for a 10% reduction of sodium density within the core plus blanket regions. The effect is strongly negative: as the reactor is small, a reduction of sodium density leads to a rising neutron leakage, which in turn poses a negative effect. Other positive effects, such as a reduced neutron absorption by sodium and a hardening of the neutron spectrum, are outweighed by this negative leakage contribution. Expansion coefficients are evaluated by increasing mesh dimensions and at the same time reducing densities, so to keep masses constant between the two configurations. Effects are evaluated considering a 10% axial and a 1% radial expansions with exact perturbation theory. I must warn again, however, that in the transient simulations, expansion effects are not taken into account due to the absence of an expansion modeling in SIMMER.

Using PARTISN with 4 processors in parallel, the computational time for one eigenvalue solution is  $\approx 3.1$  s, reduced by a half compared with the standard neutron solver.

#### Transient results

Figure 4.24 shows the temporal evolution of the computed reactivity. Pumps are powered off at time zero. The immediate response of the reactor is a drastic reactivity reduction, as the reactor itself has a negative sodium void effect. As a consequence, power decreases, see figure 4.25. Fission power is slightly overestimated through the transient. This might be due to ignoring thermal expansion negative feedback.

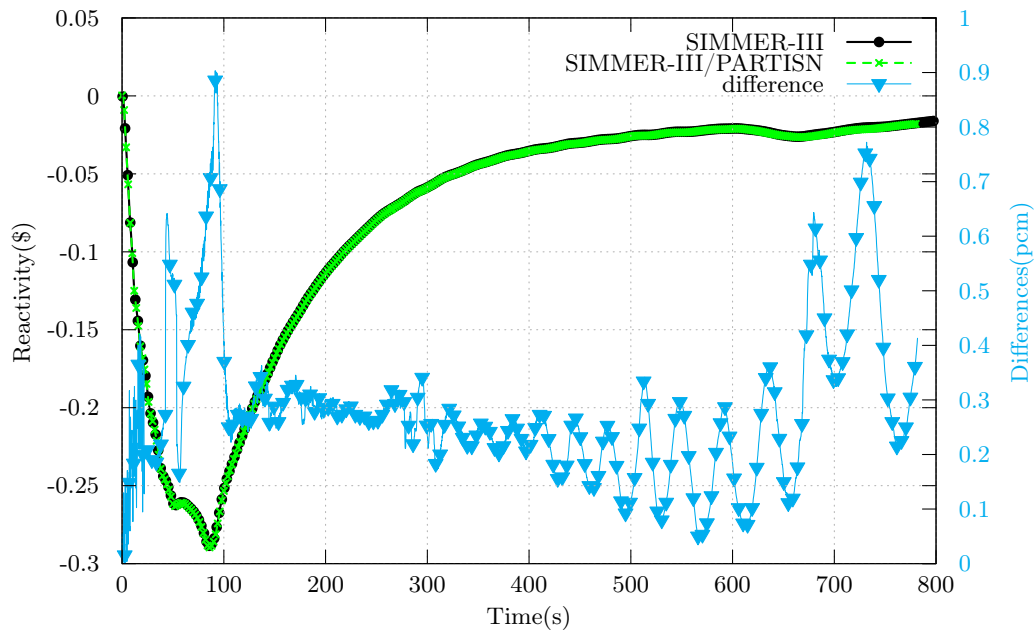


Figure 4.24: Computed reactivity as a function of time for EBR-II SHRT45-R validation case.

The total inflow of sodium decreases also, figure 4.26. However, the ratio of power to mass flow rate of sodium increases, indicating an imbalance between power produced and power removed from the core, figure 4.27. This leads to increasing temperatures in the core. As the fuel temperature rises, the Doppler effect introduces further a negative reactivity. But, due to the huge thermal inertia of the sodium pool, the inlet sodium temperature remains stationary at its steady-state value of 616 K during the whole transient.

At approximately 100 s in the transient, the power drops to such a low level that the small residual coolant flow is now sufficient to cool effectively the core. Temperatures start then to decrease. This effect is also accompanied by an increase in reactivity, declining temperatures narrows the absorption spectrum and thereby introduce a positive reactivity. The transient proceeds then to a steady-state condition, with a level of natural circulation sufficient to cool the residual decay heat in the core. The sodium mass flow rate in the core is well predicted by the code, as shown in figure 4.26. Only at  $\approx 80$  s the computed curve shows a steeper decrease than the experimental one, reaching a “plateau” value some twenty seconds before. This mass flow rate drop causes power-to-mass-flow-rate ratio to rise sharply, figure 4.27, affecting, as I will show shortly, all computed temperature traces.

After  $\approx 100$  s, the pump inertia is almost lost and the sodium flow rate is mainly driven by thermal gradients, meaning natural circulation only. The predicted natural circulation flow rate is circa 10 kg/s lower than the experimental value, which is attributed to the fact the flow rate is obtained by energy and momentum conservation. Temperature traces show also a good agreement. Figure 4.28 compares sodium temperature above the core in the instrumented subassembly XX09.

The computed temperature has the same behavior of the power-to-mass-flow-rate curve. The maximum computed temperature is less than 50 K higher than the experimental data, even though it is recorded circa 50 s later than experiment.

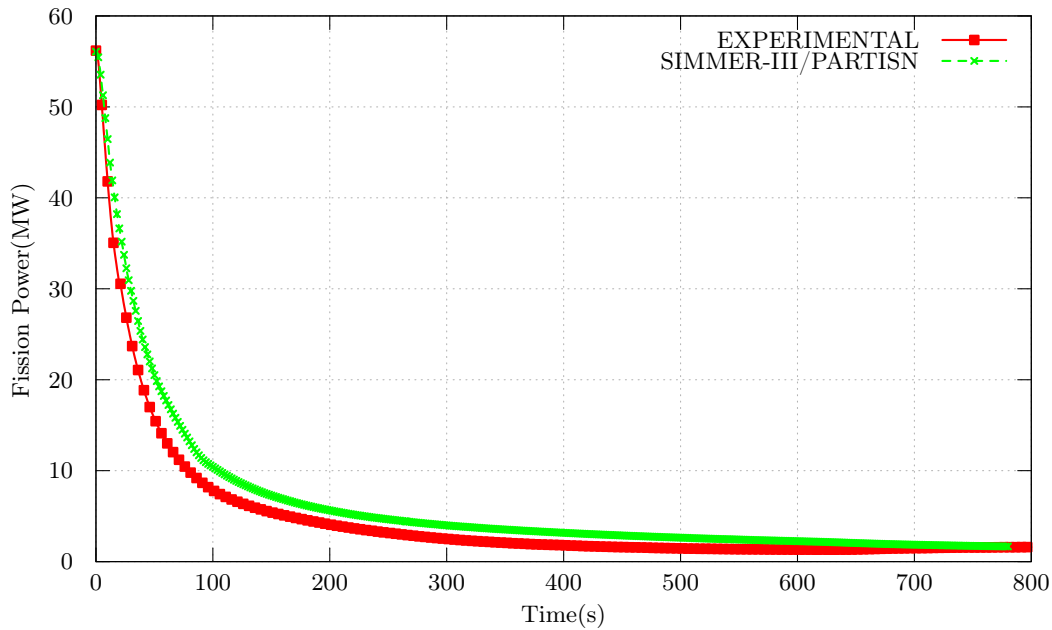


Figure 4.25: Temporal variation of the fission power for the EBR-II SHRT-45R case.

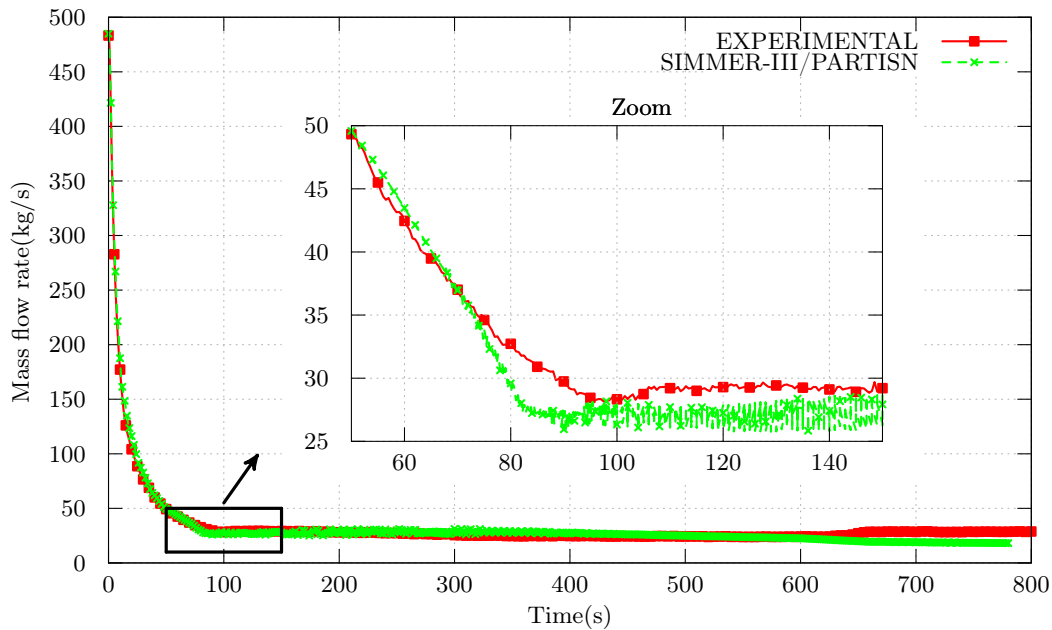


Figure 4.26: Time behavior of the sodium mass flow rate for the EBR-II SHRT-45R case.

The temperature then decreases and stabilizes at a value 50 K hotter than experimental data, due probably to the overestimation of the power level and underestimation of the natural circulation mass flow rate I already mentioned.

Other “global” temperatures are also well predicted. Figure 4.29 shows the inlet temperature at the z-pipe. Its maximum value is overestimated by  $\approx 30$  K and is predicted again some time later than recorded data. The temperature seems then to stabilize at a level circa 40 K hotter than experimental data.

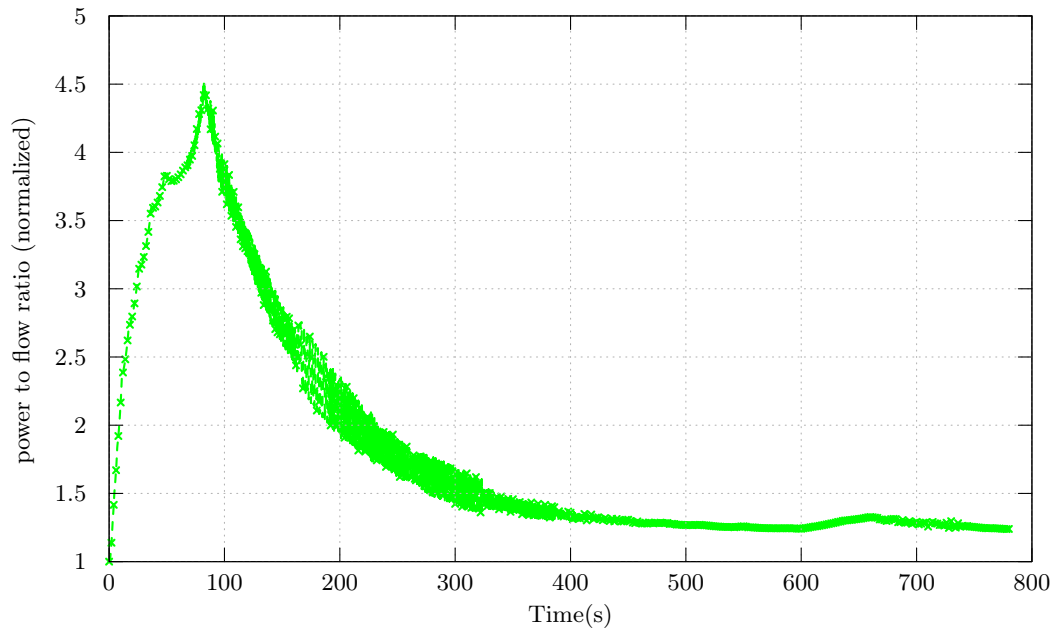


Figure 4.27: The power to flow rate ratio as a function of time as computed by the SIMMER-III/PARTISN code for the EBR-II SHRT-45R validation case.

### Comparison of coupled code with SIMMER-III

The case has been also run with the SIMMER-III standard code. The reactivity differences are below one pcm for the whole transient, as shown in figure 4.24. The same good agreement is extended to other traces as well, such as e.g. temperatures, power, mass flow rates. As for the computational time, SIMMER-III is much slower. The total computational time for SIMMER-III is  $\approx 7$  days – i.e. 0.61 Ms – 65% of which are spent in TWODANT subroutines. Using 4 processors, SIMMER-III/PARTISN computational time is reduced to  $\approx 2.9$  days – i.e. 0.25 Ms –, with only 8% spent in PARTISN calculations. This is a remarkable result, and it shows that, even in two dimensional cases, SIMMER-III/PARTISN code can provide a significant reduction in computational times.

This example provides also a first validation of the coupled code with experimental data, and shows that a more accurate neutronics modeling of transients – e.g. using higher order discrete ordinates, tighter convergence requirements, more energy groups – can be performed keeping the computational time to a practical level.

### Remarks on transient results

SIMMER-III/PARTISN provides a reasonable agreement with experimental data. Sodium mass flow rate and fission power are in acceptable accordance with the recorded data. Temperature signals from different reactor locations are well predicted within a range of  $\approx 50$  K. Thermal expansion modeling might improve the results. As a negative feedback mechanism, it would provide a faster power decrease at the start of the transient, and, in general, a lower power level. Temperatures would then be also lower and closer to experimental values in this benchmark.

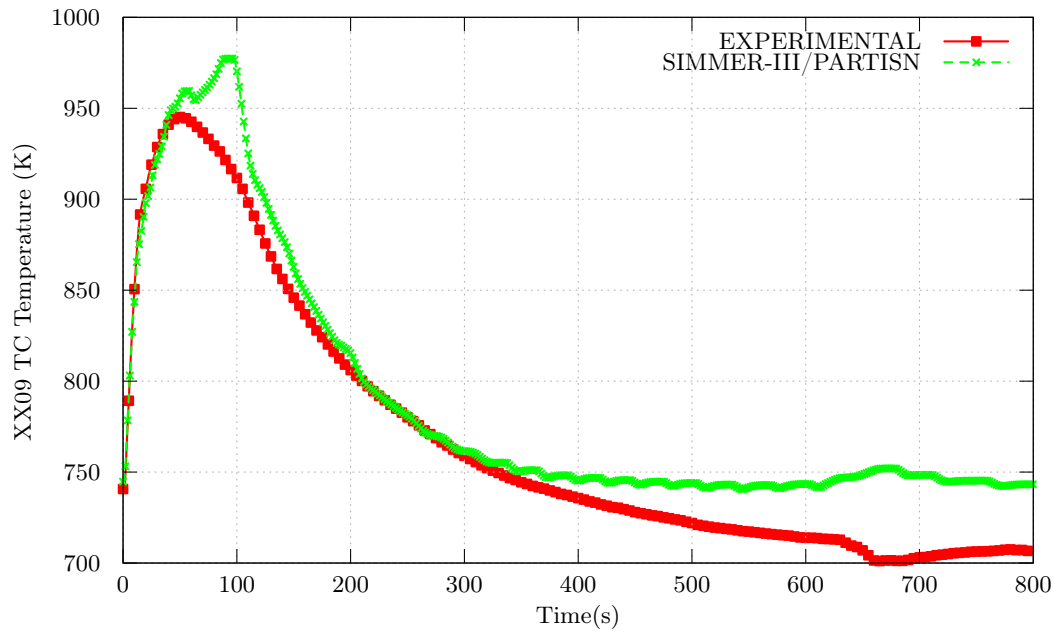


Figure 4.28: Confront between the experimental and computed sodium temperature in the instrumented subassembly XX09 at the core exit for the EBR-II SHRT-45R case.

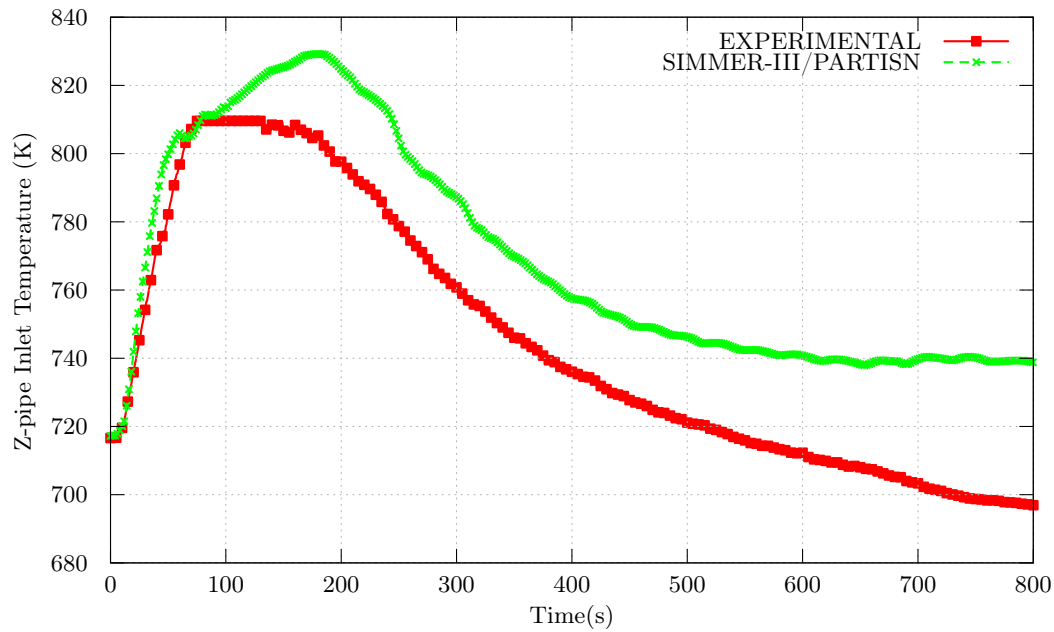


Figure 4.29: Comparison of experimental and computed sodium temperature at the z-pipe inlet as a function of time for the validation case EBR-II SHRT-45R.



## Chapter 5

# VARIANT with Heterogeneous Nodes

As I have mentioned in the previous section 3.3, the VNM has been formulated under the assumption that each node is homogeneous. However, this is not a fundamental restriction, since the Ritz procedure, by which the governing equations are derived, can in principle accommodate changes of cross sections within the nodes.

The VNM solution scheme employs response matrices computed for homogeneous nodes from pre-calculated large arrays of integrals over known basis functions. Fanning and Palmiotti [1997] demonstrated that for heterogeneous nodes, these arrays can be evaluated by numerical integration, using some known quadratures. This approach tends to slow down the computation, especially in transients, where these numerical integrals must be performed at every time step. For that purpose, I have developed a technique that allows, at least in some simple geometrical configurations, an analytical evaluation of these response matrices, thus avoiding any numerical integration.

The possibility to use heterogeneous nodes in the nodal transport theory is very attractive. It may be possible to drastically reduce the total number of nodes in the computational domains, reducing therefore the total computational cost of the calculation without sacrificing much accuracy. Moreover, it may come into hand to avoid the use of small nodes (e.g. to exactly match interfaces between different materials, such as between a control rod and its follower) that are known to hamper the numerical convergence of the solution.

Attracted by these above-mentioned advantages, I implemented this response matrix evaluation technique into the VARIANT code, verified it and assessed its performance on several cases. The same technique can be used to reconstruct the spatial dependence of any quantity (such as the power density or the reactivity worth) within the heterogeneous node, once its moments are evaluated.

I extended then this heterogeneous node technique to time dependent problems, by implementing it in the KIN3D code, the kinetics module of VARIANT. This method is useful to simulate some transients, such as control rod movement. By now in KIN3D, the motion of a control rod is simulated by smearing cross sections between the bottom of the control bundle and the follower in the node that at that time contains the bottom of the moving control rod. The motion of the control rod can now be simulated, instead, by using a heterogeneous node in axial direction, in which the upper part of the node is occupied by the control rod while the bottom is assigned to the follower. Smearing of cross sections

is then no longer needed and therefore a better accuracy can be expected. More generally, this technique can be used in every dynamics code – SIMMER to name one – to provide a flux shape update required during transient calculations.

Another development is cell calculations. In principle, one could avoid smearing of cross sections over the cell, as it is usually done now, by directly computing the response matrix of the cell taking into account all its heterogeneity.

This chapter details these efforts. It is organized as follows. First the mathematical is introduced, then its implementation into the VARIANT code is discussed and, finally, some results of its application are shown.

## 5.1 Heterogeneous Node Technique

In this section, the heterogeneous node technique is presented. Viewed from a height, this technique is a method to evaluate response matrices for heterogeneous nodes.

To simplify notation, I will not differentiate between spatial and angular basis functions, using  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  to indicate a generic basis function, which, according to equation (3.39), would then be defined as:

$$f_i(\mathbf{r}, \boldsymbol{\Omega}) = d_{l(i)}(\mathbf{r}) g_{m(i)}^+(\boldsymbol{\Omega}), \quad (5.1)$$

where  $l(i)$  and  $m(i)$  are arbitrary integer functions, that only define the ordering of the  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  functions.

### 5.1.1 Evaluation of Response Matrices

In the VNM, integrals of the following type must be evaluated for each node and energy group, see equations (3.46) in section 3.3.1:

$$A'_{t,i,j} = \int_{4\pi} \int_V \Sigma_t^{-1}(\mathbf{r})(\boldsymbol{\Omega} \cdot \nabla f_i(\mathbf{r}, \boldsymbol{\Omega}))(\boldsymbol{\Omega} \cdot \nabla f_j(\mathbf{r}, \boldsymbol{\Omega}))d\Omega dV, \quad (5.2a)$$

$$A_{t,i,j} = \int_{4\pi} \int_V \Sigma_t^{-1}(\mathbf{r})f_i(\mathbf{r}, \boldsymbol{\Omega})f_j(\mathbf{r}, \boldsymbol{\Omega})d\Omega dV, \quad (5.2b)$$

$$A_{x,i,j} = \int_V \Sigma_x(\mathbf{r})f_i(\mathbf{r})f_j(\mathbf{r})dV, \quad (5.2c)$$

where

$$f_i(\mathbf{r}) = \int_{4\pi} f_i(\mathbf{r}, \boldsymbol{\Omega})d\Omega.$$

$\Sigma_t(\mathbf{r})$  is the total cross section,  $\Sigma_x(\mathbf{r})$  is a partial cross section for reaction  $x$  and  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  are orthonormal basis functions on space and angle:

$$\frac{1}{V} \int_{\Omega} \int_V f_i(\mathbf{r}, \boldsymbol{\Omega})f_j(\mathbf{r}, \boldsymbol{\Omega})d\Omega dV = \delta_{i,j}.$$

These matrices are then used to evaluate response matrices  $\mathbf{R}$  and  $\mathbf{B}$  for each node, see equation (3.53). When nodes are homogeneous, integrals (5.2) are calculated analytically, by taking into account ortho-normality of the basis functions  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  within each node. Integrals (5.2) are evaluated for each geometry and for a standard node with a volume of  $1 \text{ cm}^3$ . These values are included as FORTRAN DATA statements into the VARIANT

code, and are used to evaluate integrals (5.2) for a node with arbitrary dimensions by using a change of variables.

If one has to calculate integrals (5.2) for heterogeneous nodes, more complex calculations are required. For 2D XY geometries, Fanning and Palmiotti [1997] proposed to compute these integrals numerically in space, Gaussian quadrature to be used. They showed that for these heterogeneous models, taking directly into account the intra-nodal cross section spatial dependence may provide better results than cross section smearing within a node, these smeared cross sections then being used for VNM calculations.

Direct numerical integration makes the computation more complicated and time consuming. I will show how I can avoid this numerical integration for some simple cases. The results may be generalized for more complicated models.

### 5.1.2 A Simple Heterogeneous Node

Let us consider a 3D (HEX-X or XYZ) reactor model in which some nodes are heterogeneous in axial direction only, as illustrated in figure 5.1. We will suppose further that the node is subdivided into  $N$  axial parts, within each part  $(z_k, z_{k+1}), k = 1, \dots, N$ , the cross sections being homogeneous.

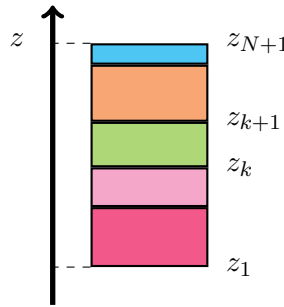


Figure 5.1: Geometry model.

This may be, for example, a node which includes the bottom of a CR, the lower part of this node being assigned to the CR follower. Then integrals (5.2) may be calculated as sums of volume integrals over each sub-node volume  $V_k$ :

$$A'_{t,i,j} = \sum_{k=1}^N \Sigma_{t,k}^{-1} \int_{V_k} \int_{4\pi} (\boldsymbol{\Omega} \cdot \nabla f_i(\mathbf{r}, \boldsymbol{\Omega})) (\boldsymbol{\Omega} \cdot \nabla f_j(\mathbf{r}, \boldsymbol{\Omega})) d\boldsymbol{\Omega} dV, \quad (5.3a)$$

$$A_{t,i,j} = \sum_{k=1}^N \Sigma_{t,k} \int_{V_k} \int_{4\pi} f_i(\mathbf{r}, \boldsymbol{\Omega}) f_j(\mathbf{r}, \boldsymbol{\Omega}) d\boldsymbol{\Omega} dV, \quad (5.3b)$$

$$A_{x,i,j} = \sum_{k=1}^N \Sigma_{x,k} \int_{V_k} f_i(\mathbf{r}) f_j(\mathbf{r}) dV. \quad (5.3c)$$

Within each sub-node  $k$ , the functions  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  are no longer orthonormal. Therefore, integrals (5.3) are of no easy solution.

### 5.1.3 A New Set of Orthonormal Functions

Let us introduce suitable functions  $f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})$  which would be orthonormal there. The subscript  $k$  indicates here the particular sub-node. We assume that both  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  and  $f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})$  are constructed from the same set of spatial and angular polynomials. One can then expand  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  within each sub-node according to:

$$f_i(\mathbf{r}, \boldsymbol{\Omega}) = \sum_{m=1}^I C_{k,m,i} f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}), \quad (5.4)$$

where  $f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})$  are orthonormal on the sub-node i.e.:

$$\frac{1}{V_k} \int_{V_k} \int_{4\pi} f_{k,i}(\mathbf{r}, \boldsymbol{\Omega}) f_{k,j}(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV = \delta_{i,j}, \quad (5.5)$$

and  $C_{k,m,i}$  is the projection of function  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  on the basis function  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$  of the generic sub-node  $k$ :

$$C_{k,m,i} = \frac{1}{V_k} \int_{V_k} \int_{4\pi} f_i(\mathbf{r}, \boldsymbol{\Omega}) f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV.$$

Knowledge of the coupling coefficient matrix  $C$  is sufficient to pass between the original set of basis functions  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  and the new one  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$ .

### 5.1.4 Evaluation of Response Matrices for Heterogeneous Node

We have now all the ingredients to evaluate the integrals (5.3). For example, employing equation (5.4), equation (5.3a) transforms into:

$$A'_{t,i,j} = \sum_{k=1}^N \sum_{m=1}^I \sum_{l=1}^I C_{k,m,i} C_{k,l,j} A'_{t,k,m,l}, \quad (5.6)$$

where

$$A'_{t,k,i,j} = \Sigma_{t,k}^{-1} \int_{V_k} \int_{4\pi} (\boldsymbol{\Omega} \cdot \nabla f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})) (\boldsymbol{\Omega} \cdot \nabla f_{k,j}(\mathbf{r}, \boldsymbol{\Omega})) d\Omega dV. \quad (5.7)$$

Equations (5.3b) and (5.3c) can be transformed in the same way. The values defined by equation (5.7) may be computed in the usual way, since they involve integration on a homogeneous sub-node. Therefore, knowing matrices  $C_{k,m,i}$  for each sub-node  $k$ , one can obtain matrices  $A$  by a simple summation through equation (5.6), thus avoiding any numerical integration. The application of this method seems justified, when the flux in a heterogeneous node can be sufficiently approximated in space by the few first basis functions, because otherwise it might be better to deal with finer nodes.

### 5.1.5 Evaluation of Coupling Coefficient Matrices

The method requires evaluation of the coupling coefficients  $C_{k,m,i}$  for each sub-node  $k$ . To evaluate them, we must take a closer look at the basis functions  $f_i(\boldsymbol{\Omega}, x, y, z)$ , that are actually used in VARIANT. They have the form:

$$f_i(\boldsymbol{\Omega}, x, y, z) = g_i(\boldsymbol{\Omega}) W_i(x, y) Z_i(z). \quad (5.8)$$

The axial basis functions  $Z_i(z)$  form an orthogonal base on the axial direction and are defined by:

$$Z_i(z) = \sqrt{2l(i) + 1} P_{l(i)} \left( \frac{2z - z_N - z_1}{z_N - z_1} \right),$$

where  $P_i(x)$  is the Legendre polynomial of order  $i$ ,  $l(i)$  is an integer function defined by the orderings of the basis functions;  $z_N$  and  $z_1$  are the lower and upper node boundary positions (indeed the axial axis is inverted in VARIANT). The new set of orthonormal functions on each sub-node  $k$  can be defined analogously as:

$$f_{k,i}(\mathbf{\Omega}, x, y, z) = g_i(\mathbf{\Omega}) W_i(x, y) Z_{k,i}(z), \quad (5.9)$$

where the axial functions  $Z_{k,i}(z)$ :

$$Z_{k,i}(z) = \sqrt{2l(i) + 1} P_{l(i)} \left( \frac{2z - z_{k+1} - z_k}{z_{k+1} - z_k} \right),$$

now form an orthogonal base over the sub-node  $k$ . Using equation (5.8) and (5.9), coupling coefficients  $C_{k,m,i}$  are evaluated as:

$$C_{k,m,i} = G_{m,i} D_{m,i} E_{k,l(m),l(i)}, \quad (5.10)$$

where

$$G_{m,i} = \int g_i(\mathbf{\Omega}) g_m(\mathbf{\Omega}) d\mathbf{\Omega}, \quad (5.11)$$

$$D_{m,i} = \frac{z_{k+1} - z_k}{V_k} \int W_i(x, y) W_m(x, y) dx dy, \quad (5.12)$$

$$E_{k,m,i} = \frac{\sqrt{(2m+1)(2i+1)}}{z_{k+1} - z_k} \int_{z_k}^{z_{k+1}} P_m \left( \frac{2z - z_{k+1} - z_k}{z_{k+1} - z_k} \right) P_i \left( \frac{2z - z_N - z_1}{z_N - z_1} \right) dz. \quad (5.13)$$

The matrix  $G_{m,i}$  consists of zero and unity entry and may be defined easily once the ordering of the basis functions is known. Also function  $W_i(x, y)$  are orthonormal, therefore the matrix  $D_{m,i}$  has also only zero and unity entries. Hence, equation (5.13) may be rewritten as:

$$E_{k,m,i} = \sqrt{\frac{2i+1}{2m+1}} F_{k,m,i},$$

where

$$F_{k,m,i} = \frac{2m+1}{2} \int_{-1}^1 P_m(u) P_i(u \Delta u_k - \bar{u}_k) du,$$

$$\bar{u}_k = \frac{z_N + z_1 - 2z_k}{z_N - z_1},$$

$$\Delta u_k = \frac{z_{k+1} - z_k}{z_N - z_1}.$$

Taking into account that a Legendre polynomial is orthogonal to any polynomial of lower order, one may calculate these values for each sub-node  $k$  by the recurrence formulæ:

$$\begin{aligned}
 F_{k,0,0} &= 1, \\
 F_{k,0,1} &= -\bar{u}_k, \\
 F_{k,1,1} &= \Delta u_k, \\
 F_{k,m,i+1} &= \frac{2i+1}{i+1} \left\{ \frac{(m+1)\Delta u_k}{2m+3} F_{k,m+1,i} - \bar{u}_k F_{k,m,i} + \right. \\
 &\quad \left. + \frac{m\Delta u_k}{2m-1} F_{k,m-1,i} - \frac{i}{2i+1} F_{k,m,i-1} \right\}.
 \end{aligned}$$

## 5.2 Flux Reconstruction

I show here that this technique, and especially expansion (5.4), can be conveniently used to pass between the fine and the coarse mesh (i.e. from moments evaluated on the big heterogeneous node to moments on each sub-node and vice versa). Typically, one is confronted with the following situation: VARIANT would compute flux moments onto the coarse grid. However, one might be interested in knowing the flux onto each fine mesh. This can be achieved with little extra computation.

So, suppose we have the flux moments computed onto the big coarse node and we want to evaluate flux moments onto each sub-node. For a generic function  $\psi(\mathbf{r}, \boldsymbol{\Omega})$ , its moments  $\xi_{k,m}$  in a generic sub-node  $k$  onto basis function  $m$  are, as usual, projections of the function onto the basis functions  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$ , i.e.:

$$\xi_{k,m} = \frac{1}{V_k} \int_{4\pi} \int_{V_k} \psi(\mathbf{r}, \boldsymbol{\Omega}) f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV_k. \quad (5.14)$$

Now,  $\psi(\mathbf{r}, \boldsymbol{\Omega})$  is our flux, and it behaves on the coarse node as:

$$\psi(\mathbf{r}, \boldsymbol{\Omega}) = \sum_i \xi_i f_i(\mathbf{r}, \boldsymbol{\Omega}). \quad (5.15)$$

Substituting equation (5.15) into equation (5.14) and rearranging, we obtain:

$$\xi_{k,m} = \frac{1}{V_k} \sum_i \xi_i \int_{4\pi} \int_{V_k} f_i(\mathbf{r}, \boldsymbol{\Omega}) f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV_k. \quad (5.16)$$

We can simplify the integrals (5.16) by expanding each of the  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  into the new set of orthogonal functions  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$ , see equation (5.4), to obtain:

$$\xi_{k,m} = \frac{1}{V_k} \sum_i \xi_i \sum_n C_{k,n,i} \int_{4\pi} \int_{V_k} f_{k,n}(\mathbf{r}, \boldsymbol{\Omega}) f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV_k. \quad (5.17)$$

But  $f_{k,n}(\mathbf{r}, \boldsymbol{\Omega})$  and  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$  are orthonormal on the sub-node  $k$  (see equation (5.5)), so we get a simple expression for flux moment  $m$  and sub-node  $k$ :

$$\xi_{k,m} = \sum_i \xi_i C_{k,m,i}. \quad (5.18)$$

Equation (5.18) means that sub-node moments  $\xi_{k,m}$  can be evaluated through summation once the coupling coefficients  $C_{k,m,i}$  and moments  $\xi_i$  are known.

We might be interested in the inverse problem: computing flux moments onto the coarse mesh, from the knowledge of flux moments on each sub-node of the fine mesh. In this case, the flux moments  $\xi_i$  on the coarse mesh are simply defined as:

$$\xi_i = \frac{1}{V} \int_{4\pi} \int_V \psi(\mathbf{r}, \boldsymbol{\Omega}) f_i(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV. \quad (5.19)$$

Splitting the domain of integration into each individual sub-node  $k$ , we obtain:

$$\xi_i = \frac{1}{V} \sum_k \int_{4\pi} \int_{V_k} \psi(\mathbf{r}, \boldsymbol{\Omega}) f_i(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV_k. \quad (5.20)$$

Now expanding the flux  $\psi(\mathbf{r}, \boldsymbol{\Omega})$  into each sub-node  $k$  onto the sub-node set of basis functions  $f_{k,m}(\mathbf{r}, \boldsymbol{\Omega})$ :

$$\xi_i = \frac{1}{V} \sum_k \sum_m \xi_{k,m} \int_{4\pi} \int_{V_k} f_{k,m}(\mathbf{r}, \boldsymbol{\Omega}) f_i(\mathbf{r}, \boldsymbol{\Omega}) d\Omega dV_k. \quad (5.21)$$

Expanding now also the  $f_i(\mathbf{r}, \boldsymbol{\Omega})$  function in integral (5.21) and taking advantage of the orthogonality, equation (5.21) mutates to:

$$\xi_i = \sum_k \sum_m \xi_{k,m} C_{k,m,i}. \quad (5.22)$$

Equation (5.22) means that the knowledge of moments  $\xi_{k,m}$  and coupling coefficients  $C_{k,m,i}$  on each sub-node  $k$  is sufficient to evaluate moments  $\xi_i$  on the big heterogenous node. Only one set of coupling coefficients  $C_{k,m,i}$  is therefore needed to pass from a coarse to a fine mesh, see equation (5.18), and from a fine to a coarse one, see equation (5.22).

### 5.3 Intra-Nodal Power and Reactivity Calculations

This technique can be applied to calculate intra-nodal power density and reactivity contributions, see Rineiski [2001]. This application is briefly discussed here.

Let us consider a problem where one has to calculate power density and reactivity contributions for small sub-nodes of rather big nodes. This situation may arise when coupling neutronic codes with thermo-hydraulics codes, which may require a finer spatial mesh compared to the neutronic nodal model. We suppose that the even-parity real and adjoint flux expansion coefficients have already been obtained for the whole node. The power density  $p_k$  in a sub-node  $k$  is therefore,

$$p_k = \frac{1}{V_k} \Sigma_{\text{power},k} \int_{4\pi} \int_{V_k} \psi(\mathbf{r}, \boldsymbol{\Omega}) d\Omega, dV \quad (5.23)$$

where  $\Sigma_{\text{power},k}$  is the power cross section for sub-node  $k$ . Expanding again the flux in equation (5.23), we have:

$$p_k = \Sigma_{\text{power},k} \sum_{i=1}^I C_{k,1,i} \xi_i = \Sigma_{\text{power},k} \xi_{k,1}. \quad (5.24)$$

The power can then be evaluated in every sub-node once  $\xi_{k,1}$  are computed. For perturbation theory calculations with VNM, the interface partial current moments are not needed, as shown by Laurin-Kovitz and Lewis [1995].

The perturbation theory integrals for entirely homogeneous nodes can be computed in two steps. During the first step, the scalar products of the real and adjoint flux moments are computed from the expansion coefficients. During the second step, the integrals (reactivity effect contributions,  $\beta^{eff}$ ,  $\Lambda$ , etc..) are computed combining those scalar products and nuclear data. For each node, two different types of scalar products are needed:

$$\langle \mathbf{\Omega} \cdot \nabla \psi, \mathbf{\Omega} \cdot \nabla \psi^* \rangle = \sum_{i=1}^I \sum_{j=1}^I \xi_i B_{i,j} \xi_j^*, \quad (5.25a)$$

$$\langle \psi, \psi^* \rangle = V \sum_{i=1}^I \xi_i \xi_i^*, \quad (5.25b)$$

where

$$B_{i,j} = \int_V \int_{4\pi} (\mathbf{\Omega} \cdot \nabla f_i(\mathbf{r}, \mathbf{\Omega})) (\mathbf{\Omega} \cdot \nabla f_j(\mathbf{r}, \mathbf{\Omega})) d\Omega dV. \quad (5.26)$$

Matrix  $B_{i,j}$  is already available in VARIANT for a standard node. For a generic node, its value is evaluated by employing a transformation of variables. Reactivity worths are then evaluated multiplying scalar products (5.25) with nodal cross sections.

Having defined the sub-node expansion coefficients, equation (5.18), it is possible to retain this two-step scheme for intra-nodal reactivity worth calculations. Sub-node scalar products are evaluated as:

$$\langle \mathbf{\Omega} \cdot \nabla \psi, \mathbf{\Omega} \cdot \nabla \psi^* \rangle_k = \sum_{i=1}^I \sum_{j=1}^I \xi_{k,i} B_{k,i,j} \xi_{k,j}^*, \quad (5.27a)$$

$$\langle \psi, \psi^* \rangle_k = \sum_{i=1}^I V_k \sum_{i=1}^I \xi_{k,i} \xi_{k,i}^*, \quad (5.27b)$$

where

$$B_{k,i,j} = \int_{V_k} \int_{4\pi} (\mathbf{\Omega} \cdot \nabla f_i(\mathbf{r}, \mathbf{\Omega})) (\mathbf{\Omega} \cdot \nabla f_j(\mathbf{r}, \mathbf{\Omega})) d\Omega dV. \quad (5.28)$$

Since each sub-node  $k$  is homogeneous, matrices  $B_{k,i,j}$  are evaluated from the pre-calculated  $B_{i,j}$  using a change of variables.

This technique was implemented into the KIN3D code. To verify this new option, calculations of nodal/sub-nodal contributions to the control rod reactivity effect are performed for the Takeda 1 benchmark, described in Takeda and Ikeda [1991].

The Takeda 1 model, for which a cross sectional cut is shown in figure 5.2 on the facing page, is a thermal reactor. The core has 30 cm in height; two 10 cm high axial reflectors are above and below the core. A uniform 5 cm spatial meshing is used in the planar  $x - y$  directions, assuming axis symmetry with respect to  $x = 0$  and  $y = 0$ .

Two energy groups are used: one for fast and the other for thermal neutrons. The benchmark requires evaluation of the effect associated with the CR extraction. In the reference configuration, the CR is inserted into the core. It is then extracted from below.



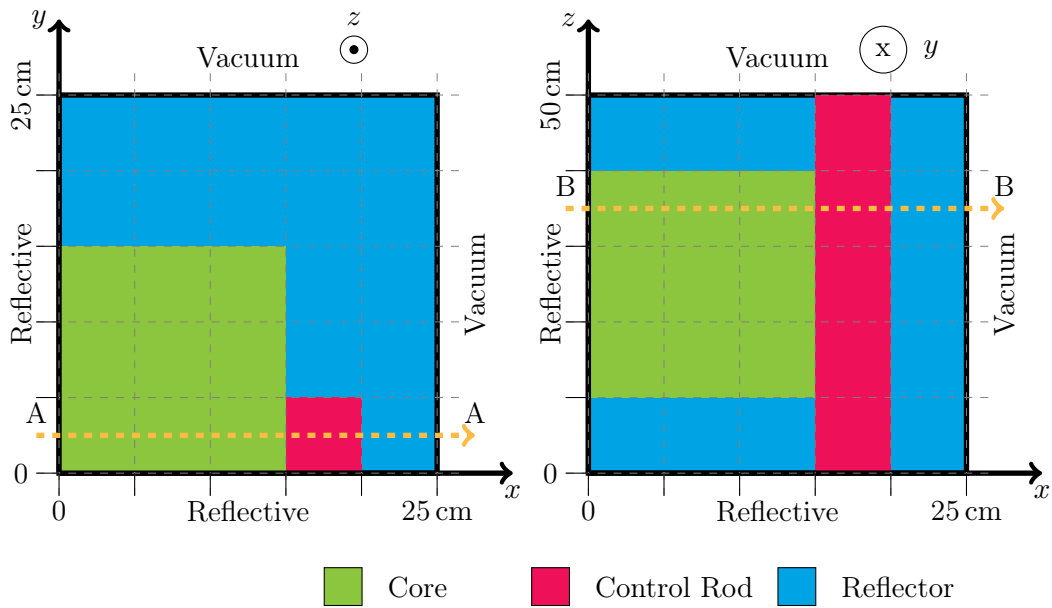


Figure 5.2: Core configuration in the horizontal cross sections of the Takeda 1 model in the axial mid-plane.

Therefore, in the perturbed configuration, the top two nodes of the CR are substituted with the CR follower. The reactivity contributions are computed in three different ways:

1. for a coarse axial mesh,
2. for the same coarse mesh, but then remapping to a finer axial mesh or
3. directly for the fine mesh.

The calculations are performed in a three dimensional cartesian geometry by means of a  $P_3$  angular approximation and a spatial approximation of the 4-th order. The coarse mesh size in axial direction is 10 cm, and the fine mesh size is 2.5 cm.

Of course, extraction of the CR inserts positive reactivity. No contribution is coming from the lowest part (i.e. the first 10 cm of the reactor), which is occupied by the CR both in the perturbed and unperturbed configurations. Removal of absorbing material causes a distortion of the neutron flux distribution in the nodes surrounding the perturbed region. Hence higher reactivity values are found as  $z$  increases towards the bottom of the CR. The results of the model are presented in figure 5.3. Of course, the coarse model can only represent (for each coarse node) the mean value of the reactivity contribution.

Comparison between the fine mesh calculation and the remapped model, reveals that this method of calculation provides reasonable sub-nodal reactivity values and preserves the coarse nodal reactivity. The computing time usually increases linearly with increasing number of nodes. That is why employing intra-nodal reactivity calculations with coarse direct and adjoint fluxes may save computer time compared to direct fine flux and reactivity calculations. This successful benchmark is conceived as the verification of this technique.

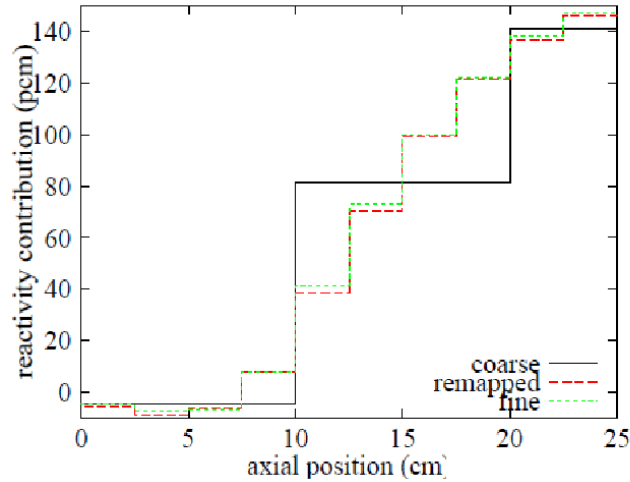


Figure 5.3: Reactivity contribution to control rod withdrawal for Takeda 1 model.

## 5.4 Numerical Results

Applications of this technique will be presented here. Two further different kinds of verification tests are performed, depending on the spatial scale of heterogeneities being considered.

The first series involves a reactor-size scale. A full reactor is modeled with heterogeneous nodes. Both integral and local quantities (e.g. multiplication factor, reactivity effects, power density distribution) are evaluated and compared with the standard model (i.e. the standard version of VARIANT with homogeneous node only). Such tests are discussed in section 5.4.1.

The second series of tests involves heterogeneities on a much smaller scale: at assembly-scale level. The idea is to validate the performance of the extension to cell calculations, where the complexity of the geometry and the smallness of the resulting computational nodes discourage – if not make impossible – the use of nodal transport methods in favor of more suitable ones, like Monte Carlo, Collision Probability, and  $S_N$  methods. These results are discussed in section 5.4.2.

### 5.4.1 Reactor Scale Heterogeneities Modeling

As for the first series, a reactor model is chosen to calculate some integral quantities such as multiplication factor and reactivity effects, comparing then the results with the standard model. In doing so, local quantities – such as the power density – can be compared with reference values. Investigated reactivity effects comprised by CR movement and sodium boiling, see Marchetti and Rineiski [2012], are additionally studied. The reactor model is the European Sodium Fast Reactor.

As a second example, the same reactor is modeled using very large heterogeneous nodes. The objective here is to try to reduce as much as possible the number of nodes while still maintaining a sufficient accuracy in the calculation.

### Description of the European Sodium Fast Reactor (ESFR) core

The ESFR reactor has been introduced previously in section 4.2.3. Here, I established also an ERANOS2.2 model for the ESFR reactor. The actual axial structure of the ESFR core is illustrated in figure 5.4.

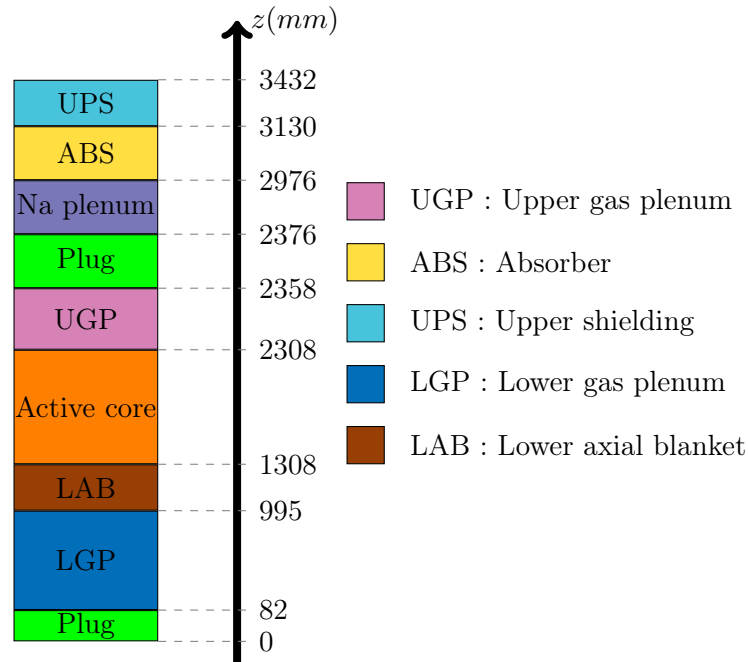


Figure 5.4: Schematic axial configuration for the ESFR core.

In the reference reactor configuration, all CRs are assumed fully withdrawn from the core. Effective cross sections are prepared subassembly-wise using the ECCO cell code, see Rim-pault [1997], with a heterogeneous subassembly geometry and a fine energy group nuclear data library – JEFF-3.1 library with 1968 energy groups. These cross sections are then collapsed to 33 energy groups for flux evaluation. Transport calculation are done with VARIANT code using the SP<sub>3</sub> angular option and a 4th order spatial approximation of the flux. I performed calculations using a coarse mesh (that included some heterogeneous nodes) and then compared with traditional VARIANT calculations, performed on a finer mesh that included all sub-nodes boundaries. A simplified option – in which cross sections are smeared over the heterogeneous node according to its sub-nodes' volume fraction – is also considered. All reactivity effects are evaluated by direct calculations.

### Control Rod movement

The first example is the evaluation of reactivity change caused by insertion of CRs. In the reference configuration, all CRs are assumed out of the core: in the perturbed configuration, CRs are inserted 5 cm into the core, as depicted in figure 5.5(b). Axially, the core is modeled with five 20 cm-long nodes. The insertion of CRs makes the top core node heterogeneous, the upper portion of which (i.e. the top 5 cm) belongs to the absorber section of the CR, and the lower part (i.e. lower 15 cm) to the follower, see figure 5.5(b).

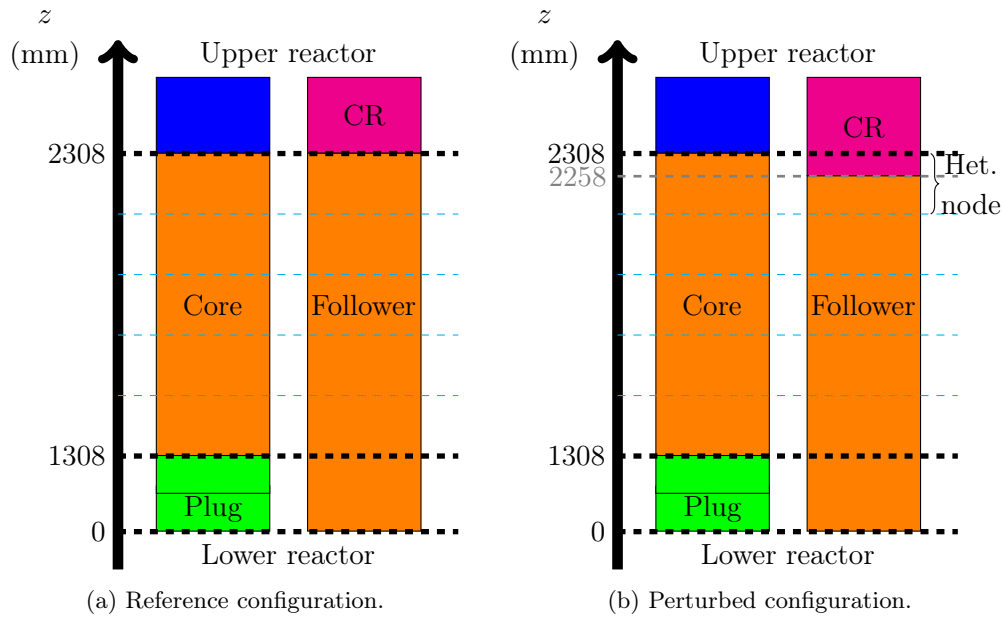


Figure 5.5: Schematical geometric configuration for a CR movement in the ESFR core.

I was obliged to add an additional plane in the perturbed configuration to match the interface between CR and follower when using the standard VARIANT code. This resulted in a greater number of computational nodes. On the other hand, my extended VARIANT could directly use the coarse mesh, without increasing the number of nodes with respect to the reference configuration. As a consequence, the computational time is  $\approx 5\%$  lower, as it is described in table 5.1.

	VARIANT	VARIANT HET	VARIANT SMP
No. of nodes	5187	4914	4914
Computational time(s)	825	784	825
No. of outer iterations	17	17	18
$k_{eff}$	1.009976	1.009989	1.008569
$\Delta\rho$ (pcm)	-119	-118	-257

Table 5.1: Computed reactivity effect  $\Delta\rho$  for the 5 cm Control Rod insertion in the ESFR reactor using different methods.

The multiplication factors  $k_{eff}$  evaluated by the standard VARIANT (VARIANT) and by my extension (VARIANT HET) are equivalent, so similar as the calculated reactivity variations  $\Delta\rho$ s. In the simplified model (VARIANT SMP) cross sections in the heterogeneous nodes are averaged according to sub-nodes volume fractions. I could therefore use the same coarse mesh as VARIANT HET calculation but the results could not predict the correct reactivity change, as the table 5.1 shows.

Cross section averaging process, in a sense, moves the absorber part of the CR— which is in reality confined to the top 5 cm of the core — down into the whole node, where the neutron importance is higher — since it is more deep into the core.

This results in an overestimation of the absorption and in a much lower multiplication factor, see table 5.1. The power density is also compared between the various models. Figure 5.6 shows the power density – for the perturbed case – in a fuel subassembly located in the first ring of the core. The heterogeneous node is composed by follower and the CR absorber section (CSD): it is treated as a single coarse node in the VARIANT HET calculations. For this case, individual (i.e. sub-nodes) power density values are calculated by the reconstruction technique presented in section 5.3. Differences between VARIANT and VARIANT HET calculations are of the order of  $10^{-3}$ . Results from the VARIANT SMP model may still be acceptable, but they are worse than VARIANT HET results.

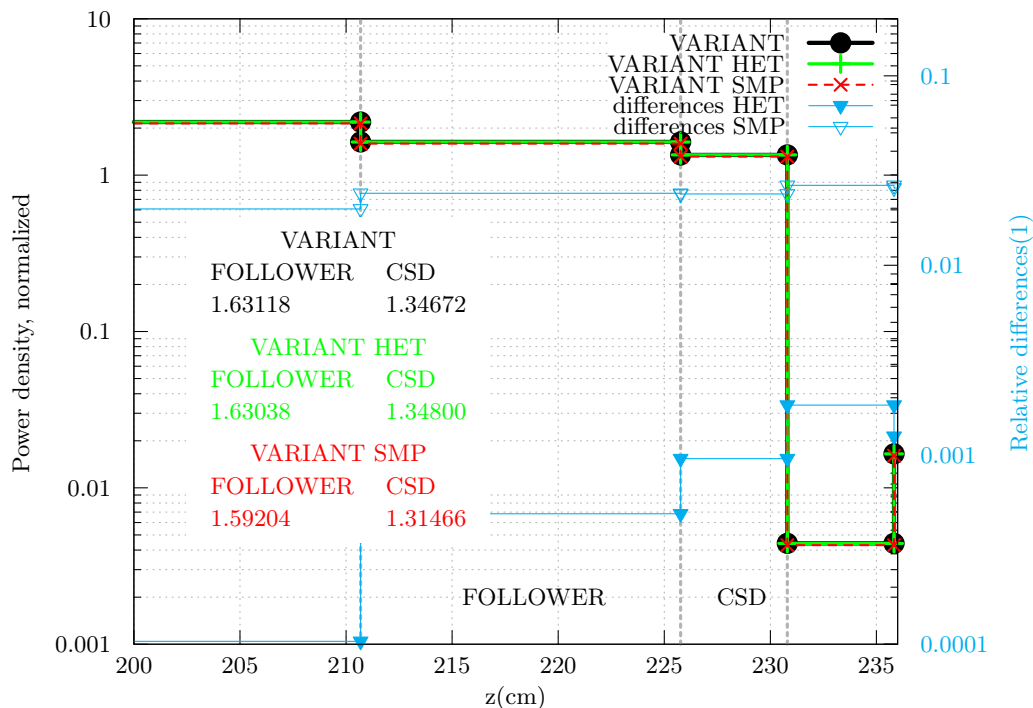


Figure 5.6: Comparison of calculated axial normalized power density in a fuel subassembly in the first ring of the ESRF core, for a 5 cm CR insertion using different techniques. Relative differences on the secondary  $y$ -axis are evaluated versus VARIANT.

### Sodium Void effect

Imagine a scenario, in which sodium evaporation happens in the upper reactor, say in the top 10 cm of the active core. This situation is depicted in figure 5.7. Evaporation is replicated in my ERANOS model by removing all the sodium from the top 10 cm of fuel subassemblies. Cross sections for this voided case are re-evaluated with ECCO and used in the transport calculation.

The top core node becomes once again an heterogeneous node, composed of 10 cm of wetted fuel (lower part) and 10 cm of voided fuel (upper part). Again, the voided scenario requires to introduce a new plane in the standard VARIANT model, resulting in a higher computational time, as shown in table 5.2. The multiplication factors between VARIANT and VARIANT HET are equivalent.

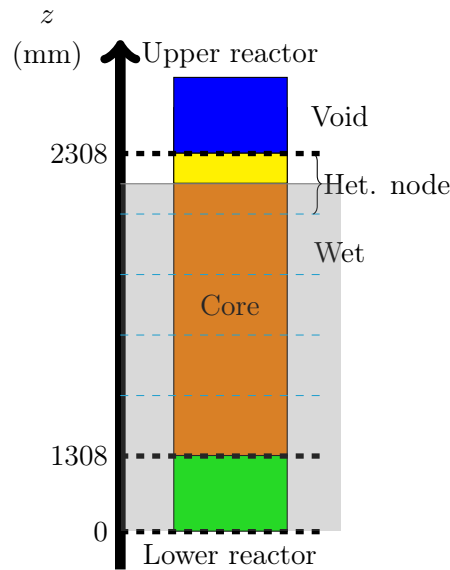


Figure 5.7: Perturbed geometrical configuration, in which sodium void is considered in a layer of 10 cm above the active ESFR core.

	VARIANT	VARIANT HET	VARIANT SMP
No. of nodes	5187	4914	4914
Computational time(s)	828	796	787
No. of outer iterations	17	17	17
$k_{eff}$	1.002905	1.002903	1.003395
$\Delta\rho$ (pcm)	-817	-817	-768

Table 5.2: Comparison of the evaluated sodium void effect  $\Delta\rho$  for the ESFR core.

The predicted reactivity change is the same between the two models. Simplified VARIANT SMP model is again not able to match the reference calculation, even though it is not as wrong as in the previous CR example. In this case, the two sub-nodes that make up heterogeneous nodes (i.e. wetted and voided fuel) are not so different in composition, and cross sections averaging over the node is not expected to introduce a large error. Power density comparison reveals differences of the order of  $10^{-3}$  between VARIANT and VARIANT HET. Figure 5.8 on the facing page shows the power density into a fuel subassembly on the first ring of the core. Simplified model results may still be acceptable, VARIANT SMP model matches the reference model with differences of the order of  $10^{-2}$ .

### Using coarse nodes

The focus of these studies is directed to model a reactor using few big heterogeneous nodes only, and thereby to reduce as much as possible the computational time without sacrificing accuracy. This is a critical aspect in view of a possible, future coupling of VARIANT with fluid-dynamics codes (e.g. SIMMER)

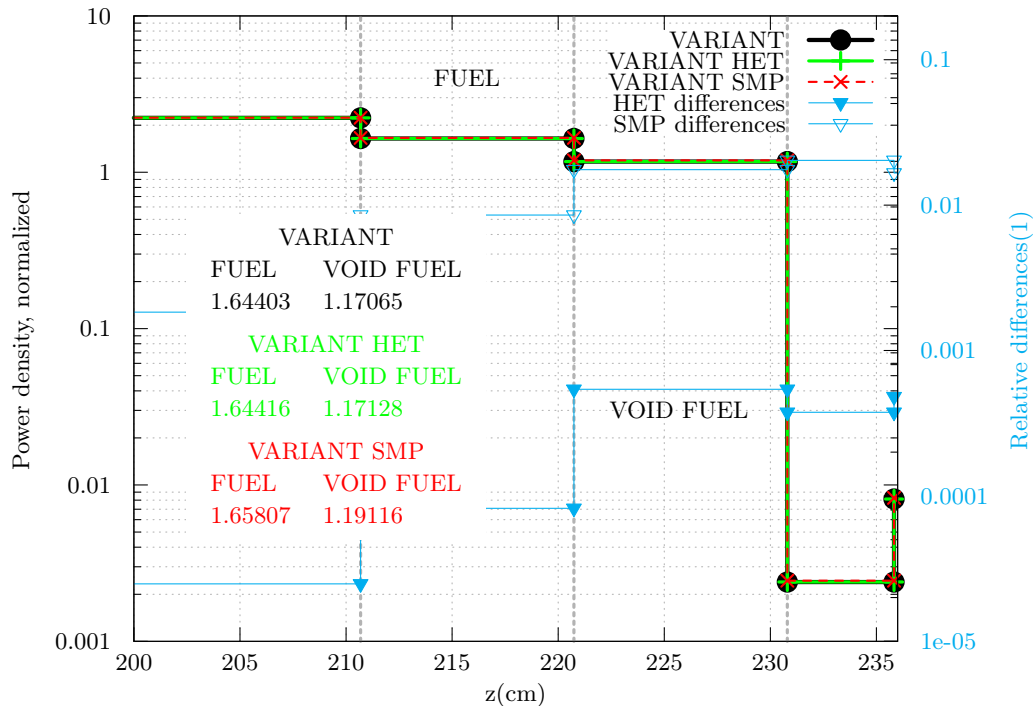


Figure 5.8: Normalized power density axial distribution in a subassembly located in the first ring of the ESFR reactor, for the scenario, in which sodium boiling is considered in a 10 cm layer at the top of the core. Relative differences versus the VARIANT model are plotted on the right y-axis.

Usually, fluid-dynamics codes use very fine meshes; with my heterogeneous extension, I may gather several of those fine fluid-dynamics meshes into one big heterogeneous node for neutronic computations, and still be able to model heterogeneity effects (e.g. caused by a temperature gradient or different node compositions). Here the performances of the method in a static calculation are investigated.

As a first simple example, I decided to model the ESFR reactor by only three nodes in axial direction: one for the lower part of the reactor, one for the whole core, and the last for the upper part. The lower node comprises the plug, the lower plenum, and the fertile blanket. The upper node is made of the upper plenum, the plug, the sodium plenum, the absorber layer, and the upper shield, figure 5.9(a).

A second, more refined, model uses 6 nodes instead: it is derived from the 3-node model by further splitting each node into two, as depicted in figure 5.9(b). Calculations are performed with the same options used in section 5.4.1:  $SP_3$ , 4th order spatial approximation of the flux, and 33 energy groups. The obtained results are compared with the reference ESFR model previously described. The evaluated multiplication factors are shown in table 5.3. The number of nodes is only 819 and 1638 for the HET models: computational time is reduced by  $\approx 75\%$  in one case and by  $\approx 42\%$  in the other. Nevertheless, the  $k_{eff}$ s are still acceptable: only a 75 pcm difference exists for the 3-node HET model, while the 6-node model predicts the value correctly – calculation accuracy on the  $k_{eff}$  is  $10^{-5}$ . Reconstructed power density profiles in the lower and upper nodes of a fuel subassembly located in the second ring of the reactor are shown in figure 5.10.

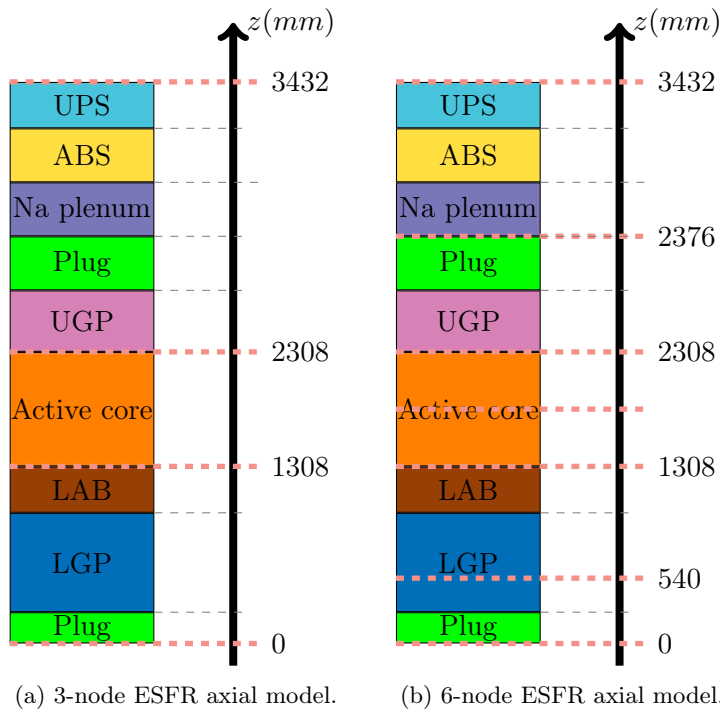


Figure 5.9: 3-node (a) and 6-node (b) axial representations for the ESFR reactor. Thick red dashed lines mark the mesh boundaries for the heterogeneous geometries.

Significant deviations exist for the 3-node model. In the lower node, figure 5.10(a), differences are of the order of  $\approx 10\%$ , but with very high values where the fluxes are low:  $\approx 52\%$  in the lower plug and  $\approx 30\%$  in the lower plenum. The situation is similar into the upper node, figure 5.10(b).

Differences of  $\approx 5 \div 6\%$  are in the upper plug, plenum, and sodium plenum, but these values rise to 30% and even to 200% in the absorber and in the upper shield. In particular, in the absorber, the reconstructed power density is negative. In the end, the 3-node HET model gives poor results, when it comes to predict the power profile within the node. Even if the neutron flux behaves smoothly within a heterogeneous node, the cross-section for the power changes in a step-like manner across each sub-node boundary. As a result, the power density changes also spatially in a step-like manner.

	VARIANT	VARIANT HET	
		3-NODE	6-NODE
No. of nodes	5187	819	1638
Computational time(s)	782	203	325
No. of outer iterations	17	20	18
$k_{eff}$	1.011193	1.011964	1.011131
$\Delta\rho$ (pcm)	–	75	-6

Table 5.3: Comparison of the  $k_{eff}$  computed with the heterogeneous 3-node and 6-node models against the homogeneous VARIANT code for the ESFR case.



Representing such a discontinuous function with a continuous polynomial can give only a first order approximation of its spatial shape. This might result in negative fluxes, especially when the flux value is, in absolute terms, very small. Increasing the number of nodes to 6 is beneficial: differences in the power density do not exceed 3% in any of the nodes. Such values may still be acceptable. This example shows also the consistency of my extension: by increasing the number of nodes, the solution approaches the fine mesh calculation. Moreover, the 3-node and 6-node models are used to determine reactivity parameters – as section 5.4.1. Results from a 20 cm CRs insertion are reported in table 5.4(a). Even in this case, predicted values of  $\Delta\rho$  are acceptable, differing by only  $\approx 45$  pcm and  $\approx 20$  pcm with the VARIANT reference calculation. Good results are found also for the sodium void scenario, table 5.4(b), where differences are  $\approx 50$  pcm and  $\approx 65$  pcm.

	VARIANT	VARIANT HET	
		3-NODE	6-NODE
No. of nodes	5187	819	1638
Computational time(s)	828	235	358
No. of outer iterations	17	23	21
$k_{eff}$	1.004050	1.004348	1.003771
$\Delta\rho$ (pcm)	-703	-749	-725

(a) CRs insertion.

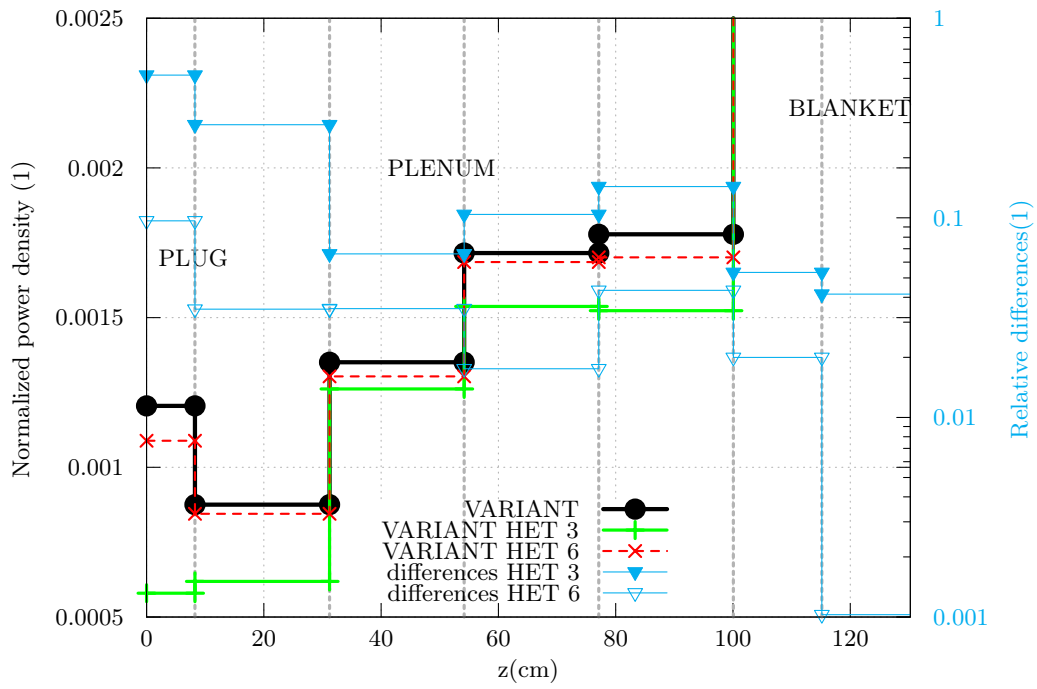
	VARIANT	VARIANT HET	
		3-NODE	6-NODE
No. of nodes	5187	819	1638
Computational time(s)	828	187	335
No. of outer iterations	18	20	18
$k_{eff}$	1.002905	1.003151	1.002196
$\Delta\rho$ (pcm)	-817	-868	-882

(b) Sodium Void Effect.

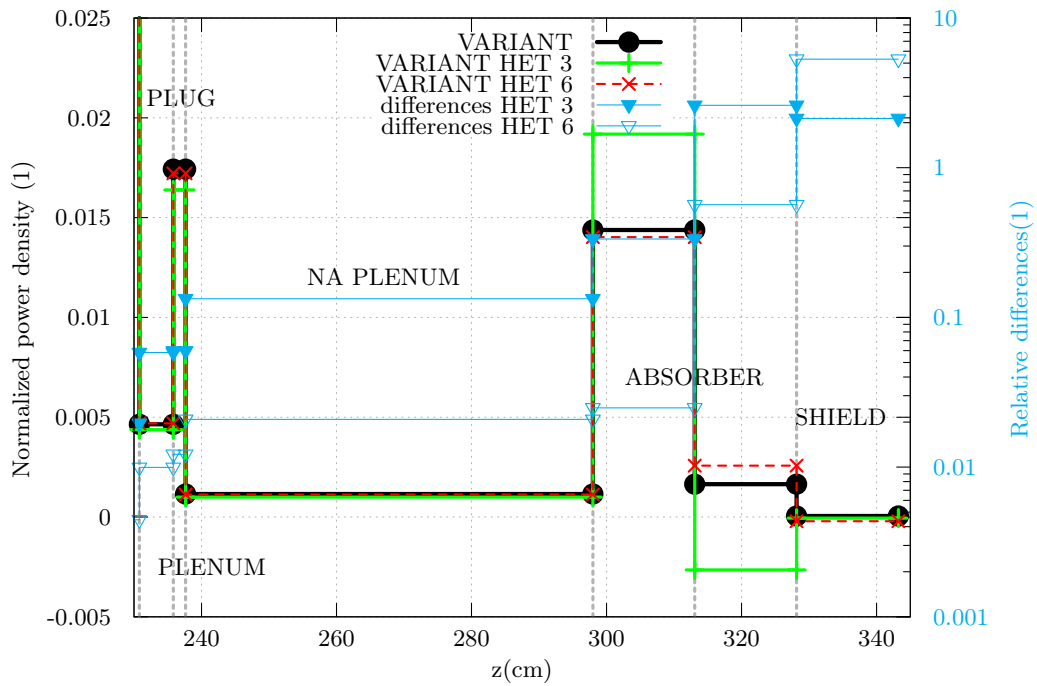
Table 5.4: Reactivity effects evaluated with the heterogeneous 3-node and 6-node VARIANT models and comparison with the homogeneous VARIANT code for the ESRF case.

### 5.4.2 PWR Cell Calculations

The second series of validations is focused to model small-scale heterogeneities. I selected a PWR 17x17 fuel assembly and tried to model it using heterogeneous nodes, comparing then multiplication factor and fluxes to a reference model. As a reference, I selected a two-dimensional PARTISN model. The aim here is to model the assembly using a few coarse nodes while still being able to model the geometrical complexity of the assembly.



(a) Lower reactor.



(b) Upper reactor.

Figure 5.10: Spatial distribution of the computed power density in the lower (a) and (b) part of the ESRF reactor. Differences, against the VARIANT model, are shown on the secondary y-axis.

### Description of the model

The assembly itself consists of a rectangular lattice of pin cells. Each pin cell consists of a fuel pin of 0.41 cm surrounded by a clad with an external radius of 0.46 cm and its moderator. The total pin cell measures 1.26 cm. To model such a setup, I defined first a cartesian two dimensional model of the pin cell, as illustrated in figure 5.11. Each pin cell is modeled by a 12x12 fine mesh, represented by the fine dashed gray lines in that figure. PARTISN calculations are done on this fine mesh. In the VARIANT HET model, each pin cell is divided instead into a coarser 4x4 mesh with heterogeneous nodes, as represented by thick black dashed lines in figure 5.11. VARIANT HET calculations also use this coarse mesh. As a result the VARIANT HET model has a total of  $70 \times 70 = 4900$  spatial nodes, while PARTISN model has  $206 \times 206 = 42436$  spatial nodes.

As the angular approximation is concerned, PARTISN used a  $S_N$  order equal to 16, VARIANT HET used a  $P_5$  approximation. Four energy group are applied.

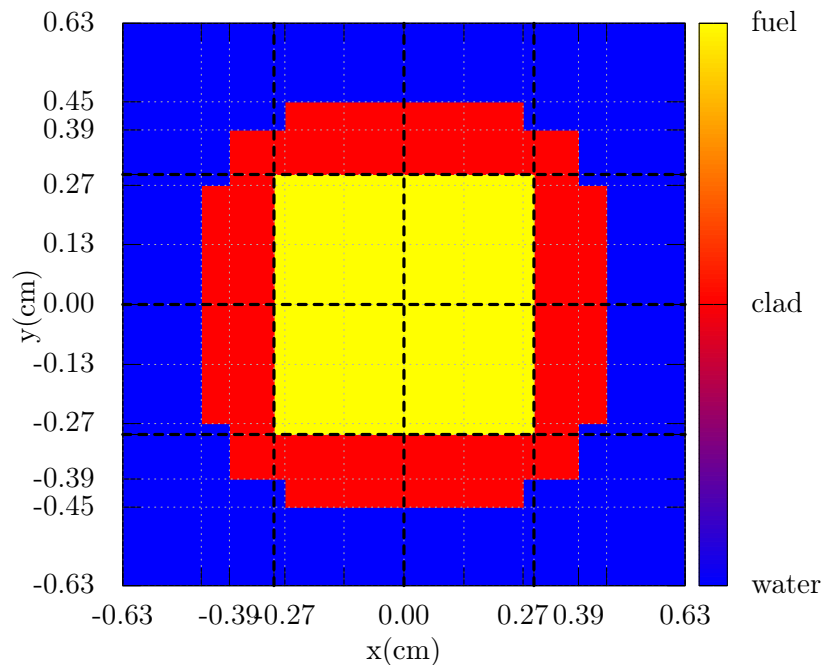


Figure 5.11: Schematic representation of a spatial mesh for one pin of a PWR 17x17 assembly. Black lines identify coarse nodes, while gray lines represent the fine mesh.

### Results

The multiplication factors are listed in table 5.5. They differs by  $\approx 281$  pcm. Despite the large number of nodes in the PARTISN model, the computational time is comparable to that of VARIANT HET. I must however note that the latter uses a full  $P_5$  angular approximations, resulting in a very large linear system.

The VARIANT HET model, the fine-mesh fluxes are reconstructed from the coarse ones by means of the reconstruction technique discussed in section 5.2. These VARIANT HET fine-mesh fluxes are compared with the PARTISN fluxes.

	PARTISN	VARIANT HET
$k_{eff}$	1.321142	1.326078
Computational time(s)	115	137

Table 5.5: Evaluated multiplication factor ( $k_{eff}$ ) between the PARTISN and the heterogeneous VARIANT (HET VARIANT) model for a PWR 17x17 assembly.

The fast flux is expected to be high in the fuel pin and low in the moderator, the opposite holds for the thermal flux. Fast neutrons are produced in the fuel and slow down in the moderator, to build-up the thermal flux. Thermal neutrons diffuse instead from the moderator to the fuel pin, where they are absorbed. For the fuel pin located in the center of the assembly, VARIANT HET fluxes are shown in figure 5.12, where the just-mentioned characteristics can be appreciated.

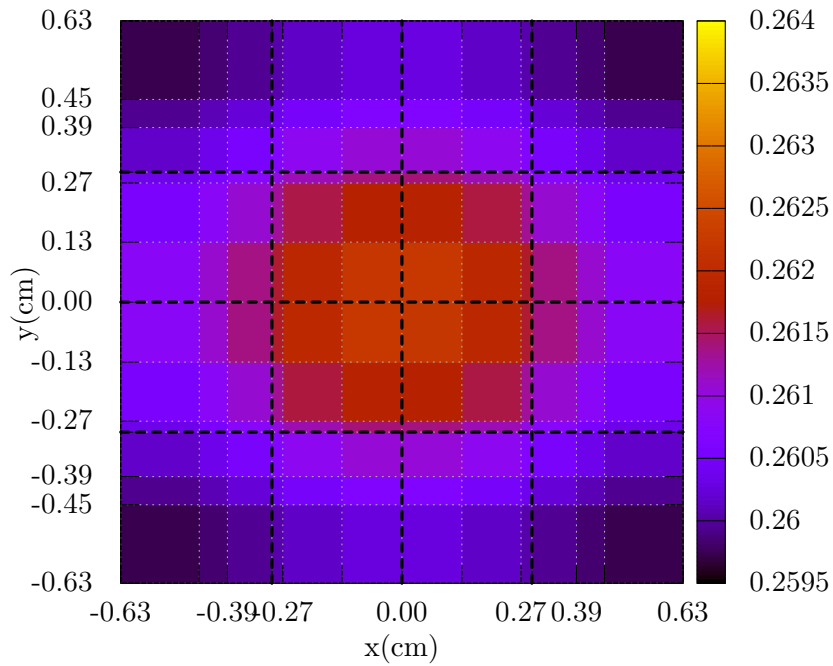
Comparison of the evaluated fluxes reveals a rather good agreement between PARTISN and VARIANT HET computations. For the pin cell located at the center of the assembly, the fluxes on the centerline are compared in figure 5.13. Relative differences are below 1% for the fast flux, while they are below 3% for the thermal flux. These higher values are located at the boundary of the cell.

## 5.5 Some Remarks on the Heterogeneous Technique

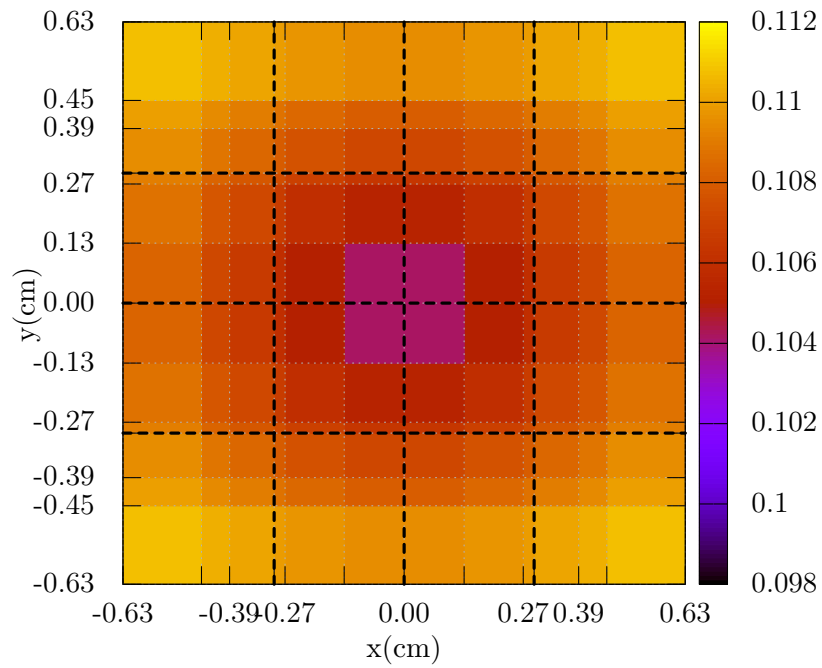
Examples presented in section 5.4 showed a good performance for this technique. It can speedup the computations, with a loss of accuracy in predicting reactivity effects or local quantities – such as power levels – that can still be considered acceptable.

However, there are cases, in which it is better not to use this technique, and to revert to the standard homogeneous calculations. I am talking of cases, in which the cross sections variations within an heterogeneous node are “too strong”. In these cases, the flux within the node is going to behave rather roughly, and even a fourth order spatial polynomial expansion might not match at all the spatial behavior of the real flux within the node.

I put such a situation to a test with the Takeda 1 benchmark. An heterogeneous node comprising the last 5 cm of the core and the 5 cm of the CR are used. A fast flux traverse in the  $x$  direction cutting through the control rod – more specifically as indicated by the dashed orange arrow in figure 5.2 – is shown in figure 5.14. One would expect a cosine-like shape of the flux in the core material – i.e. between 10 cm and 15 cm – and an exponential decrease of the flux in the absorber of the CR– i.e. between 15 cm and 20 cm. Indeed this is what I find in the homogeneous model. But such a different behavior – i.e. cosine shape and negative exponential – can not be well represented by a forth order polynomial expansion, therefore the heterogeneous model is providing completely wrong results.

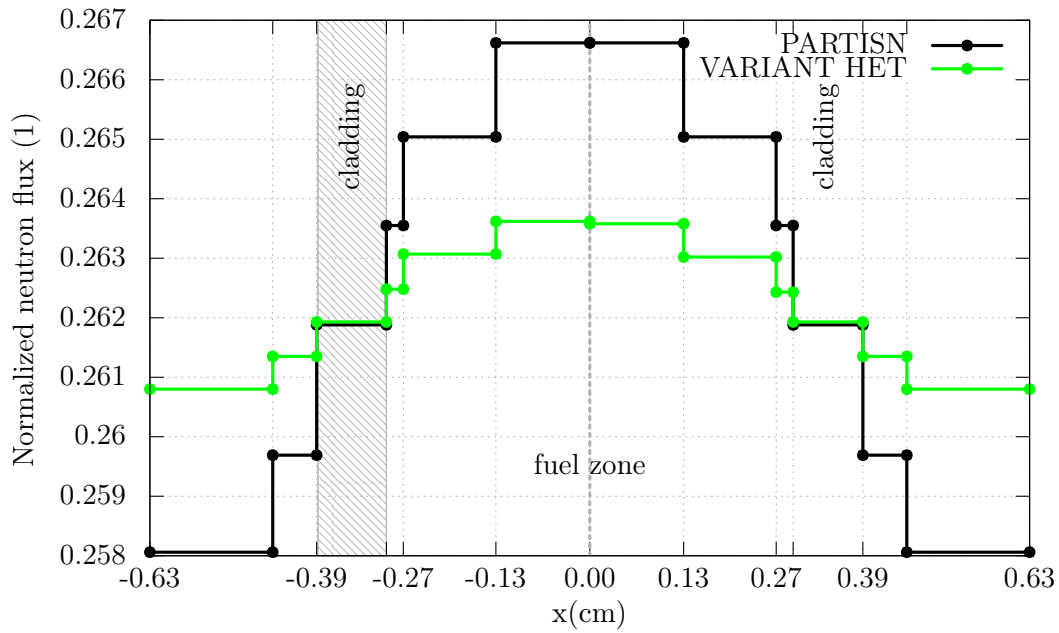


(a) Fast flux.

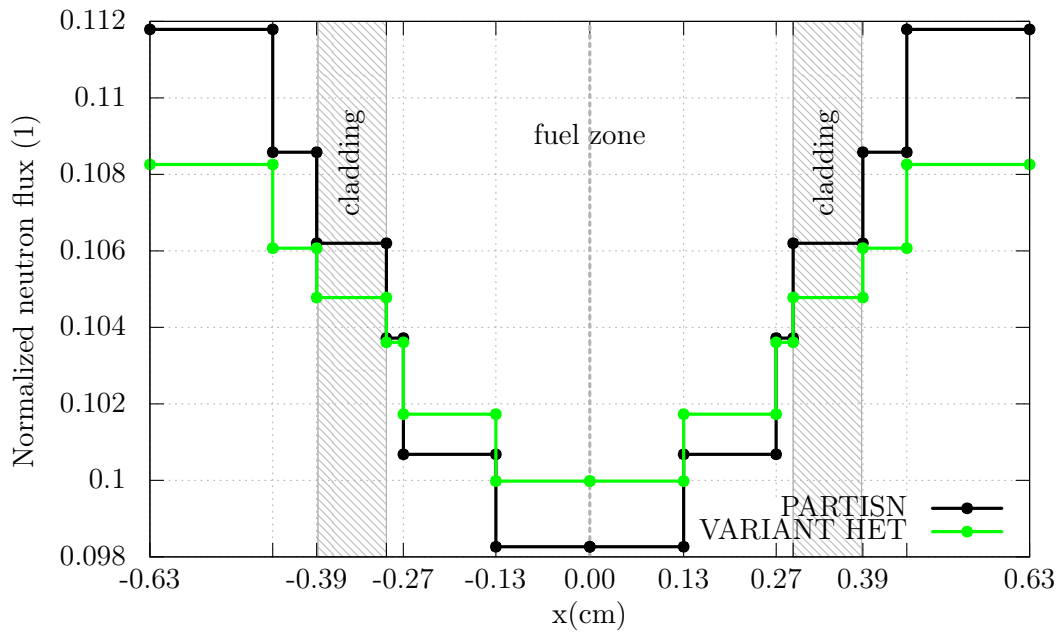


(b) Thermal flux.

Figure 5.12: Calculated neutron fluxes evaluated by the VARIANT HET model for the central fuel pin of 17x17 PWR assembly.



(a) Fast flux.



(b) Thermal flux.

Figure 5.13: Comparison of fast (a) and thermal (b) fluxes on the centerline of the central pin cell of a 17x17 PWR assembly.

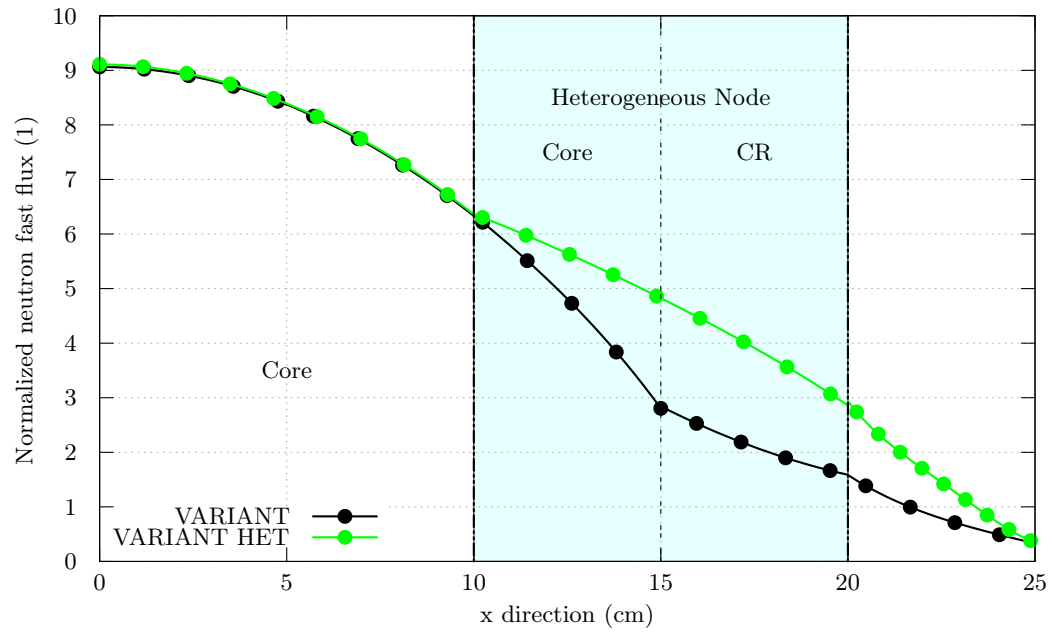


Figure 5.14: Fast neutron flux for the Takeda Benchmark 1 case, see section 5.3. The case is here modeled with an heterogeneous node comprising a portion of the core and of the CR.





## Chapter 6

# SIMMER-KIN3D Coupling

Current neutronic solvers for SIMMER (i.e. TWODANT and THREEDANT) use finite difference, discrete ordinates methods. These methods are known to be mainly subjected to two kind of problems: ray-effects and negative flux fixups.

Ray-effects phenomena is inherent to the  $S_N$  method itself, as described by Lathrop [1968]. It is caused by the restriction of the neutron streaming to a discrete and finite set of directions that is adopted by the  $S_N$  method. Situations arise, where at a certain point in space no streaming directions are available for neutrons (coming from a nearby source for example) to reach that position. As a consequence, the neutron flux at this point is very low. By contrast, a nearby point might instead have a direct streaming direction from the source: at this point the flux is high, because of the direct contribution of the source. Typically, spatial oscillations in scalar fluxes are therefore observed, as one passes alternatively through points that might or might not show a direct streaming direction to the source, as explained by Lathrop [1971]. Briggs et al. [1975] showed that increasing the number of streaming directions ameliorates but does not solve the problem. By contrast,  $P_N$  or  $SP_N$  methods do not suffer from this effect.

Flux fixups have to do with the spatial discretization of the problem. In  $S_N$  methods, the angular neutron flux is usually assumed to vary linearly within one spatial cell. This linear behavior is then used to evaluate (i.e. extrapolate) the flux downstream at the further boundary of the cell from the knowledge of the flux at two different upstream points (usually at the cell center and at the other cell boundary). This linear extrapolation is clearly only an approximation. Think for example of a pure absorbing media. There, the flux is behaving exponentially in space as  $e^{-\Sigma \Delta x}$ . Extrapolated values might therefore differ significantly from the real physical ones. In low-flux regions, this extrapolation might cause negative fluxes to appear. As these negative fluxes are propagated to neighboring cells, numerical stability of the solution can be disrupted. The usual approach is to set negative fluxes to zero as they appear during computation. Refinement of the spatial discretization is one possible approach to avoid flux fixups. In fact, a linear approximation of the exponential behavior is accurate as long as  $\Sigma \Delta x$  is low, i.e.  $\Delta x \ll \Sigma^{-1}$ , where  $\Delta x$  is the dimension of the cell. This criteria can be easily satisfied in steady-state cases. However, in accidental sequences, materials can relocate: situations in which the original spatial meshing is too coarse might arise during the transient. It is difficult to know a

priori a spatial neutronic mesh suitable for the whole transient. Ray-effects and flux fixups are two problems demanding other methods to tackle the solution of the neutron transport problem. As a nodal,  $P_N$  or  $SP_N$  method, VARIANT is free of those problems.

This chapter details the status of the coupling between the VARIANT code and the SIMMER-III. The coupling is not complete. As opposed to SIMMER/PARTISN coupling, KIN3D is not used as a substitute for the SIMMER neutron solver, but rather in parallel. The normal SIMMER computational flow is unchanged: KIN3D-evaluated fluxes are not read back in SIMMER which still relies to TWODANT for flux shape updates. The aim is to show, whether KIN3D is able to reproduce accurately the SIMMER transient, given the appropriate input data.

## 6.1 Coupling Procedure

The coupling procedure is somewhat similar to the previous SIMMER/PARTISN coupling. We can consider SIMMER as the driving program, preparing input data files for KIN3D calculations, calling the external code via a system call, waiting for computations, and then resuming the standard SIMMER execution flow. This procedure is replicated each time the neutron transport solution is required by SIMMER, i.e. at each time shape. KIN3D performs the kinetics calculations within this shape step.

### 6.1.1 Input KIN3D Perturbations: a Simple Scheme

SIMMER provides a set of cross sections describing the reactor state at the start of the shape step, time  $t_1$  in figure 6.1, and another one for the end of the shape step, time  $t_2$ . The KIN3D driving perturbation is then this cross sections change during time, see figure 6.1. cross sections are interpolated linearly in time between  $t_1$  and  $t_2$ .

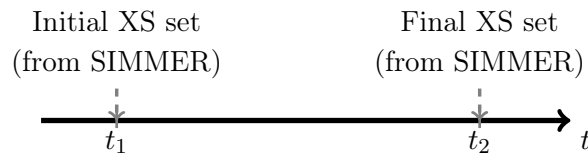


Figure 6.1: SIMMER/KIN3D time scheme. Initial and final XS sets are produced by SIMMER, KIN3D solves then the spatial kinetic equations using these XS sets to build the perturbation driving the transient.

Mass and temperature distributions in each SIMMER-III cell change during transient, so the whole set of cross sections is changing continuously in time – i.e. cross sections on a generic  $i,j$  cell at time  $t_1$  are different to cross sections on the same  $i,j$  cell but at a later time  $t_2$ . With the current KIN3D scheme, that would enforce us to specify as many KIN3D “composition change” cards as cells, and to put all compositions – i.e. actual and at next time – into one single cross section file would be an infeasible approach.

To avoid the problem, the original KIN3D input perturbation scheme is extended: it can now handle multiple cross sections input data files. Each of them represents a reactor state at a specific time point. Perturbations in this case are automatically built by the

code assuming that the whole set of cross sections changes between two consecutive time points. All cross sections are linearly interpolated in time between time points.

### 6.1.2 Input KIN3D Perturbations: a More Refined Approach

It is worth mentioning that during a shape step, i.e. between  $t_1$  and  $t_2$ , SIMMER performs several reactivity steps – those  $t_r$  in figure 6.2. At each of those, cross sections in SIMMER are recalculated based on the new, instantaneous mass and temperature distributions. In the coupling procedure above instead, the KIN3D transient is driven only by cross sections variations between time  $t_1$  and  $t_2$ , thus ignoring cross section changes on the reactivity-step time scale. This simplification seems justified as long as the shape step is not too “long”.

A later development allows to take into account these reactivity-step cross section variations. In this case, SIMMER produces also cross section data files at each reactivity step, i.e. at times  $t_{r1}$ ,  $t_{r2}$ , and  $t_{r3}$  as indicated in figure 6.2 – and not only at times  $t_1$  and  $t_2$ . KIN3D uses then the same cross section variations as SIMMER. All tests performed, however, showed no substantial differences between the two coupling schemes. Therefore, in the following, I restrict myself to the first and simpler scheme.

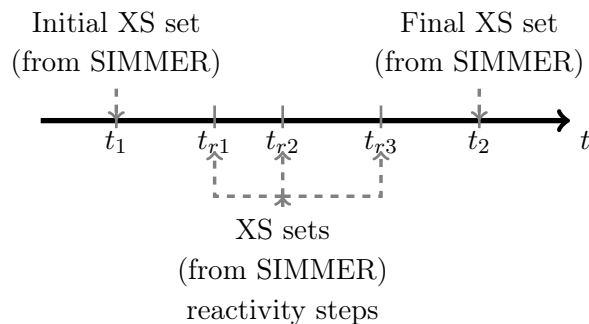


Figure 6.2: In a second coupling scheme, cross sections sets are produced by SIMMER also at each reactivity step (e.g.  $t_{r1}, t_{r2}$ ).

### 6.1.3 Geometric Interface

A geometry interface between the two codes is also necessary: SIMMER-III targets  $r$ - $z$  geometries with azimuthal symmetry, while VARIANT/KIN3D treats only Cartesian and hexagonal geometries – either two- or three- dimensional.

Usually in SIMMER-III, one or more radial rings are used to represent a ring of hexagons in the real reactor, so that the user can try to model the temperature gradient within subassemblies. In other words, each SIMMER-III ring is assigned to a particular ring of hexagons in the VARIANT/KIN3D model, see figure 6.3. The interface allows the user to specify this assignment. To maintain azimuthal symmetry, hexagons in one VARIANT/KIN3D ring share the same composition.

Cross sections for each ring of hexagons in KIN3D are evaluated by volume-averaging all SIMMER-III rings that are assigned to that ring of hexagons. KIN3D models a 60-degree sector of the reactor, as the SIMMER-III model shows azimuthal symmetry. Such a symmetry might however be broken by “special” subassemblies – e.g. typically control rods.

Users can specify which SIMMER-III rings represent a special composition and assign them particular positions in the VARIANT/KIN3D hexagonal lattice. These compositions are special in the sense they are not averaged with all other compositions that belong to that particular ring of hexagons.

Most notably, this prevents smearing of control rods and fuel subassemblies together and the subsequent assignment of that smeared composition to the whole ring of hexagons in VARIANT/KIN3D.

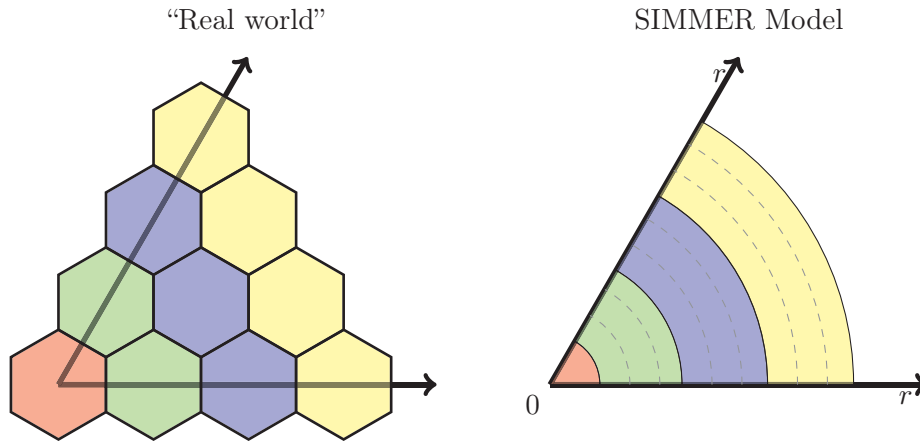


Figure 6.3: Schematics for a conversion of a hexagonal geometry into a cylindrical geometry..

## 6.2 Verification of Coupling Approach

I applied this coupling procedure to several cases, and here the results are shown. The following examples show that KIN3D is able to reproduce SIMMER transients.

### 6.2.1 ULOF Transient Fast Reactor

I apply this coupling to the same ULOF case for the medium-sized, sodium-cooled FR analyzed with SIMMER-III/PARTISN in section 4.2.2. The reactor has a thermal power of 1000 MW. It is divided into inner and outer regions.

A radial blanket surrounds the outer core region, followed by several rings of reflector subassemblies. The equivalent core diameter is 2 m, active core height is 1 m. Control system is formed by 19 CR bundles.

KIN3D models a 60-degree sector of the reactor in 3D hexagonal geometry, see figure 6.4. It consists of 13 rings. Each ring has an unique composition, determined by radial-averaging SIMMER-III cells that belong to that specific ring.

One control rod is positioned in ring 4 and two more are in ring 6 and ring 7. Axially, the VARIANT/KIN3D mesh is the same in the SIMMER-III one. This VARIANT/KIN3D model is labeled as VARIANT/KIN3D HOMO model.

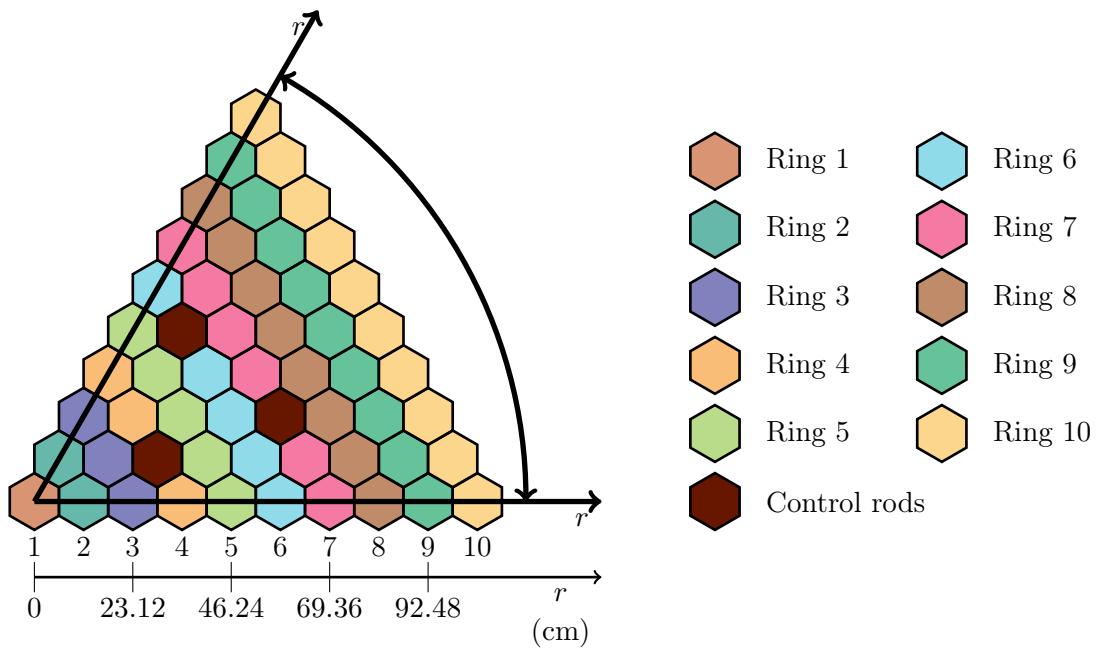


Figure 6.4: VARIANT/KIN3D hexagonal model, only first 10 rings (core zone) are shown. In dark brown are Control Rod subassemblies.

### Steady-State comparison

At first multiplication factor and flux distribution are compared. Table 6.1 compares the multiplication factor for different angular discretizations. Agreement is good, in particular, as the angular discretization is refined, KIN3D and SIMMER results seem to converge to the same value within the convergence criteria.

The SIMMER-III computation is very fast, requiring only 4 s with  $S_4$  and 10 s with  $S_8$ . VARIANT computations are slower:  $SP_3$  option takes more than 30 s, while with  $P_3$  the increase in the number of unknowns makes this option unrealistic for transient analysis. It requires more than 700 s.

	SIMMER		VARIANT	
	( $S_4$ )	( $S_8$ )	( $SP_3$ )	( $P_3$ )
$k_{eff}$	1.07077	1.07000	1.06786	1.07030
$\Delta k(\text{pcm})$	-	67	250	41
Computational time (s)	3.9	10.2	33.4	709.7

Table 6.1: Comparison between SIMMER and VARIANT computed multiplication factors and the effects of the angular discretization.

Figure 6.5 shows flux distributions in axial and radial direction for fast fluxes, i.e. energy group one with energy boundaries 10.5 MeV and 6.5 MeV. An 11-group library is used in this case. Specifically, figure 6.5(a) shows an axial traverse and figure 6.5(b) shows a radial one. Excellent agreement between the two codes is found. Differences are well below 1% within the inner and outer core regions. Only in blanket and reflector subassemblies, i.e.

toward periphery of the core, we observe few values with 10% difference. However, in these regions, fluxes are low, even one hundredth of the flux at core center, so these differences are very small in absolute terms.

### Transient results

The transient involves the compaction of fuel, which raises the reactivity level up to prompt-critical condition. The power peak generates a fuel movement in the core region that causes reactivity to rapidly drop to sub-critical level. Comparison of reactivity traces is illustrated in figure 6.6(a). The agreement between both codes is indeed good, differences are few pcm, with a maximum value of  $\approx 40$  pcm. These differences are caused by the different treatment in VARIANT/KIN3D, where two or more axial nodes, and therefore cross-sections, are smeared together. However, such small deviations might become important close to the prompt-critical conditions, i.e. when  $\rho \approx \beta$ . This is what I found in this transient and it is well reflected by the power behavior depicted in figure 6.6(b). The power peak is overestimated by VARIANT/KIN3D by a factor of  $\approx 10^3$ , while the agreement is generally good before that peak. A closer analysis exhibits that the reactivity during this power excursion is above prompt-critical for VARIANT/KIN3D, but in SIMMER-III this condition is never reached, figure 6.7. Above prompt-critical, the time behavior of the reactor is determined exclusively by the prompt neutrons, and the power will behave as:

$$N(t) \approx N_0 e^{\alpha(t-t_0)}, \quad (6.1)$$

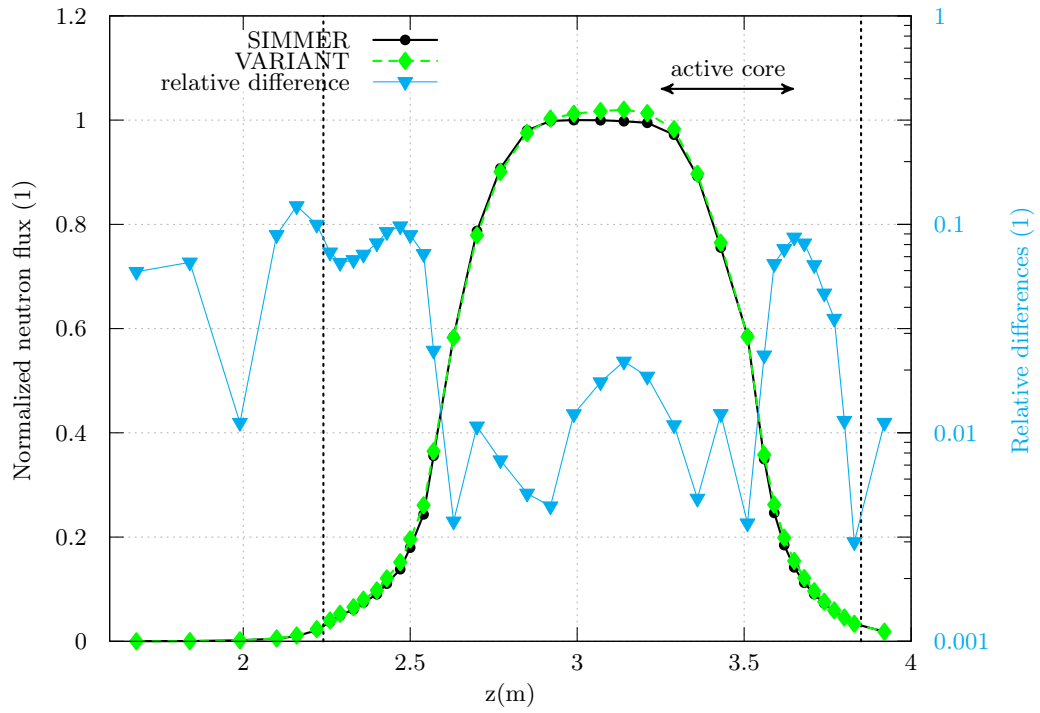
where  $\alpha$  is the inverse of the reactor period:

$$\alpha = \frac{\rho - \beta}{\Lambda}.$$

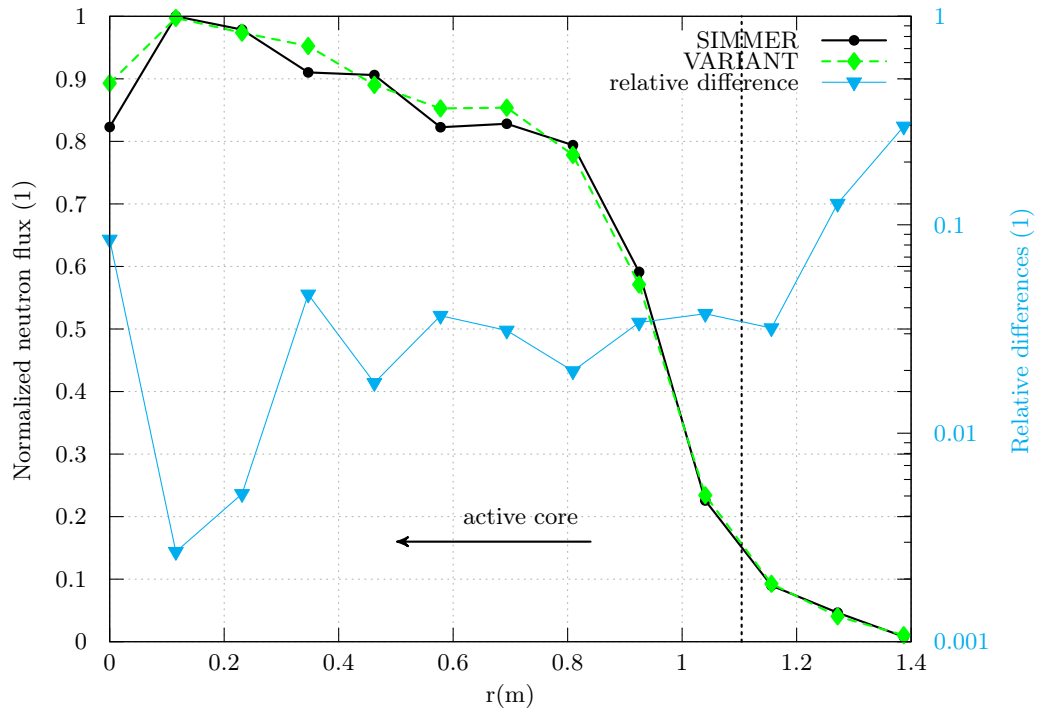
For fast reactors, the neutron generation time  $\Lambda$  is of the order of  $3.0 e^{-7}$  s: therefore even a small reactivity excess over  $\beta$  may lead to very small reactor periods, that is, the power changes on a very small time scale. Power estimated with formula (6.1) – with kinetics parameters taken from VARIANT/KIN3D calculation – is shown in figure 6.7, and fits well with the computed VARIANT/KIN3D overall power. Hence to conclude, the VARIANT/KIN3D power peak is determined by the reactor reaching prompt-criticality due to a small overestimation of the reactivity by the VARIANT/KIN3D code. As for the computational burden, a SIMMER-III shape update takes approximately just one second, while a single VARIANT flux shape computation takes a little less than 10 s.

## 6.3 Heterogeneous Extension to SIMMER/KIN3D Coupling

The previous case shows the ability of VARIANT/KIN3D to model consistently SIMMER-III transients. There is though one aspect that may limit the application of the present SIMMER/KIN3D coupling, and that is computational time. Compared to TWODANT or PARTISN, VARIANT is slower. I must add that the VARIANT/KIN3D model is usually a three-dimensional model, while SIMMER-III is limited to two dimensions. This makes the VARIANT neutronic computation particularly daunting. One possible way to tackle the problem is to reduce the number of nodes in the neutronic computations.

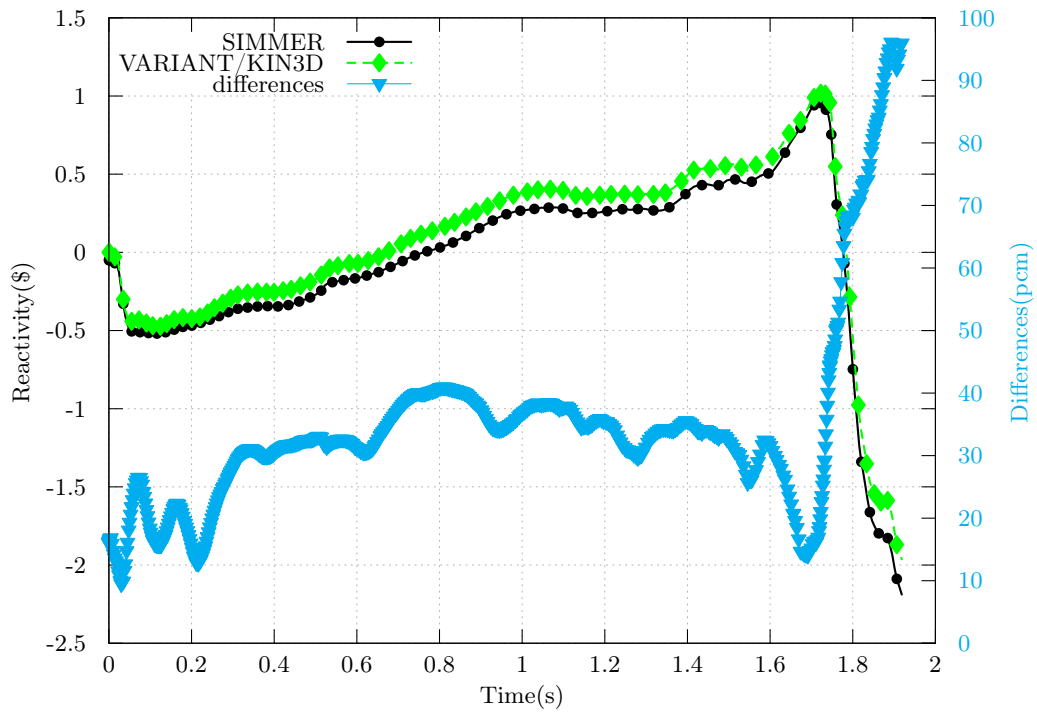


(a) Axial flux traverse for fast flux in second ring ( $r \approx 0.11$  m).

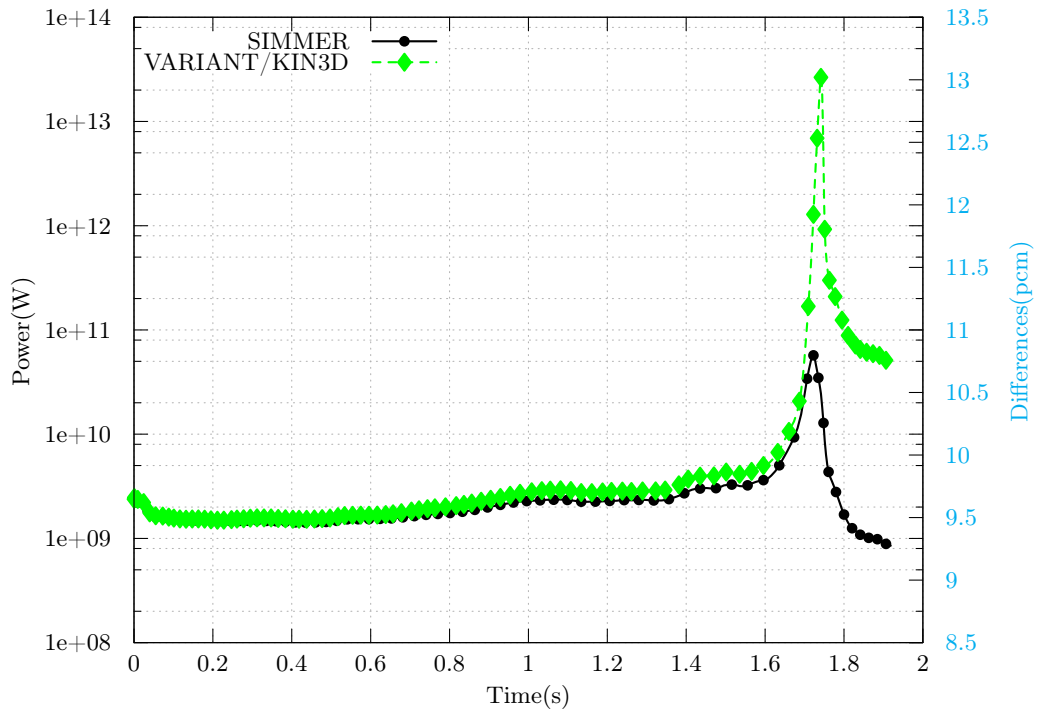


(b) Radial flux traverse for fast flux at core midplane ( $z \approx 3.1$  m).

Figure 6.5: Comparison between computed initial fast fluxes for the ULOF case.



(a) Reactivity traces as evaluated by SIMMER and VARIANT/KIN3D .



(b) Power traces as evaluated by SIMMER and VARIANT/KIN3D .

Figure 6.6: Temporal evolution of reactivity (a) and power (b) for the ULOF case transient computed by SIMMER and by VARIANT/KIN3D.



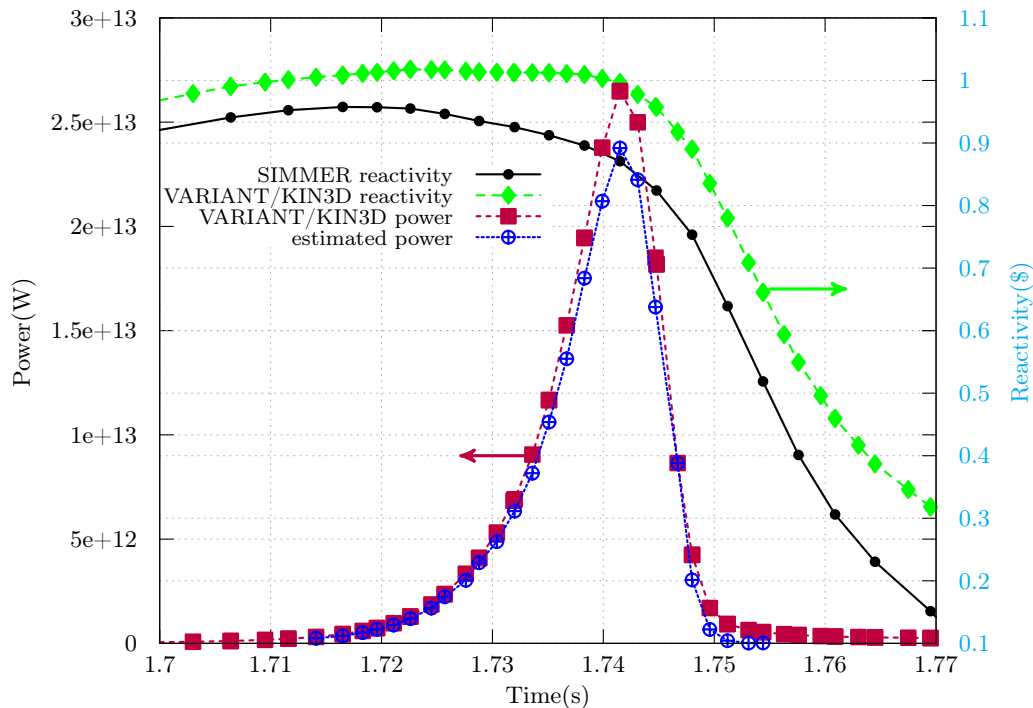


Figure 6.7: VARIANT/KIN3D power excursion is caused by reaching prompt-critical condition, while in SIMMER-III this condition is never reached, thereby the predicted power excursion is milder.

The application of the VARIANT heterogeneous extension I described before in section 5 could be used to achieve this objective. Moreover, this seemed a particularly favorable situation to apply it. Typically, along the axial direction, material densities change mainly at specific heights only: for example at the start and at the end of fuel pins, at lower and upper plugs.

Figure 6.9 shows fuel densities and temperatures along the axial direction for one sub-assembly for the same FR case analyzed in section 6.2.1. It is evident how fuel density changes in a step-like manner, with density almost constant on each interval. Only temperatures change continuously with height. This leads to small cross section variations between axially-adjacent cells (provided that they belong to the same density interval) and I explained already that the VARIANT heterogeneous extension works very well, when cross section variations within an heterogeneous node are mild.

The basic idea is to build heterogeneous nodes assembling two or more SIMMER-III fluid-dynamics cells, see figure 6.8. Only heterogeneities in axial direction are considered here. Neutron fluxes on each SIMMER-III fluid-dynamics cell can be reconstructed from the computed VARIANT flux moments on the heterogeneous neutronic cell, as shown in section 5.2. In the following section, this heterogeneous VARIANT extension is applied to the same ULOF case analyzed in section 6.2.1.

### 6.3.1 Two HET Models

I investigated two different heterogeneous VARIANT models. They differ by the way fine SIMMER-III planes are gathered together to build heterogeneous nodes.

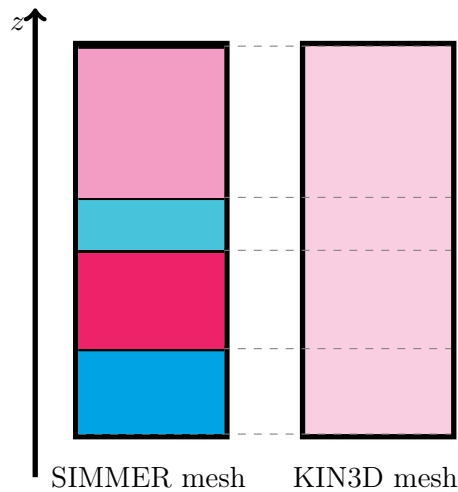


Figure 6.8: SIMMER fluid-dynamics mesh and VARIANT coarse mesh: several fluid-dynamics planes are gathered together to make a single heterogeneous VARIANT node.

The first model represents only the core active height with heterogeneous nodes. Every heterogeneous node is composed of only two SIMMER-III fine nodes. The total number of axial planes for this model is 32, down from 40 fine planes in the SIMMER-III model. This model is further labeled as VARIANT/KIN3D HET 1.

The second model uses coarser axial meshes. It has only 16 planes, heterogeneous nodes are bigger, consisting in some cases of 6 SIMMER-III fine planes. This model is called VARIANT/KIN3D HET 2. The homogeneous VARIANT model discussed in section 6.2.1 is referred to as VARIANT/KIN3D HOMO.

In assembling the two models, I tended to put cells belonging to the same density interval – see figure 6.9 – in the same heterogeneous node. However, that is not always possible and in some subassemblies those density intervals cannot be respected.

Results for all the models are compared, whenever possible, on the fine SIMMER-III mesh. This means for example, that coarse-node neutron fluxes coming from heterogeneous models are reconstructed on the fine SIMMER-III mesh and then compared.

### Steady-State results

Both models predict the multiplication factor ( $k_{eff}$ ) with small errors, as listed in table 6.2. Differences between VARIANT HET 1 and HOMO models are within the convergence criteria. However, as for the computational time, no significant difference is observed between VARIANT/KIN3D HOMO and HET 1 models. Using a heterogeneous node requires more computations per node than using a standard homogeneous node. It seems in this case that this additional “heterogeneous” computational effort offsets the advantage of having fewer nodes in the domain: it is expected, however, that a further reduction in the number of nodes must at some point reduce the total computational burden.

This trend is indeed confirmed by the second heterogeneous model VARIANT HET 2, in which the total computational time is halved compared to the VARIANT/KIN3D HOMO model, paying to accuracy only  $\approx 6$  pcm.

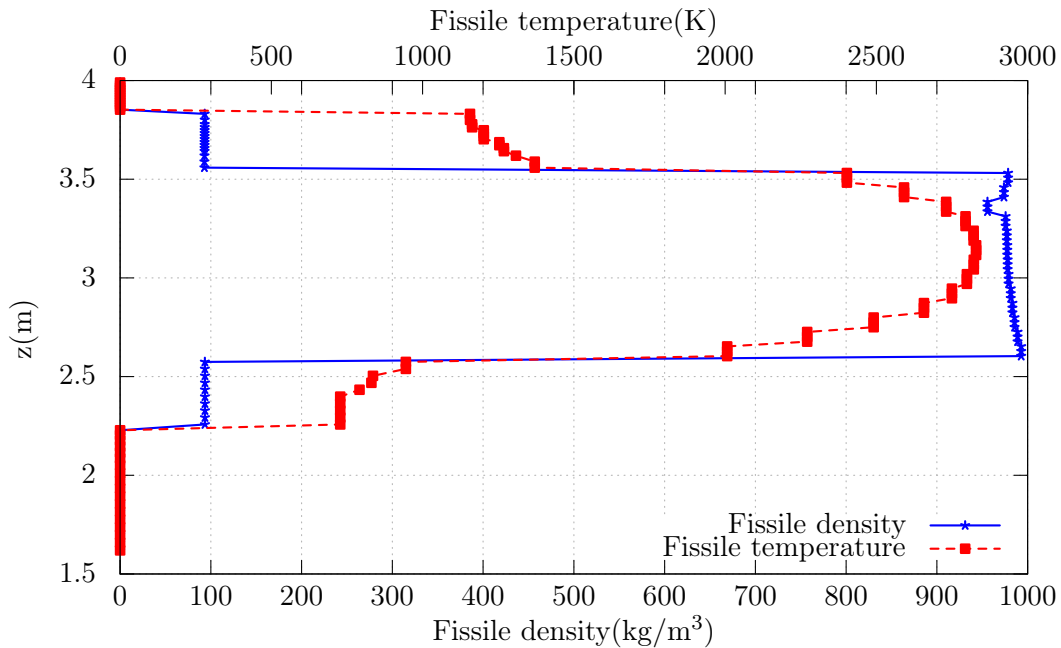


Figure 6.9: Axial distribution of fissile density and temperature on one subassembly for the ULOF case analyzed in section 6.2.1 at initial condition.

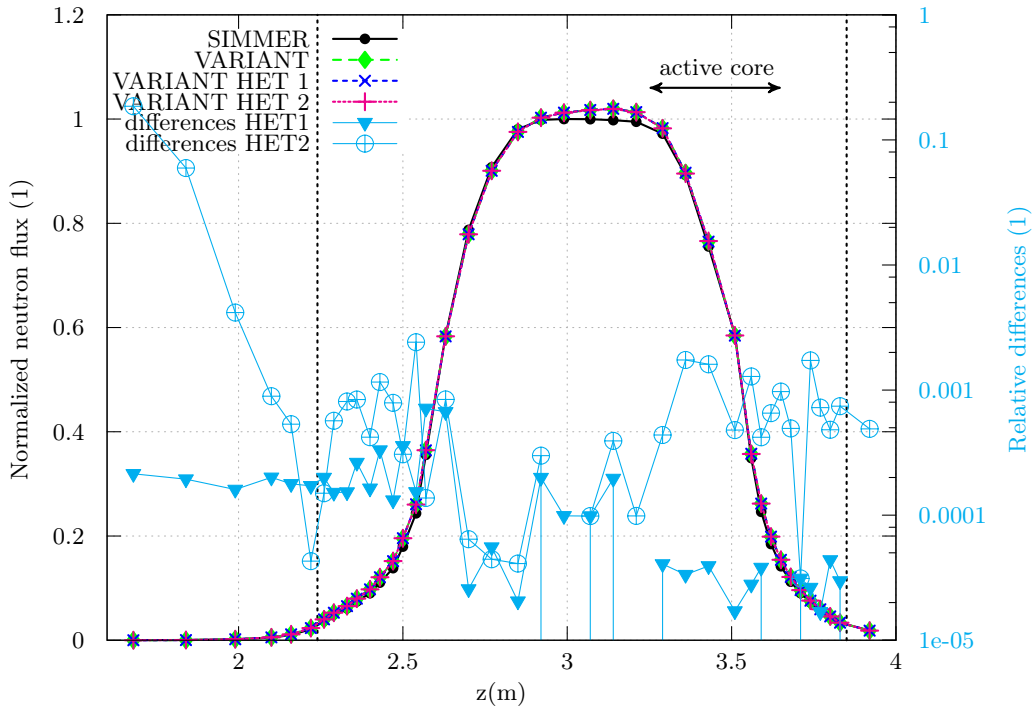
	HOMO	HET 1	HET 2
$k_{eff}$	1.06786	1.06785	1.06792
$\Delta k$ (pcm)	-	0.97	5.62
Computational time(s)	33.4	34.4	14.9

Table 6.2: Multiplication factors ( $k_{eff}$ ) evaluated with different models for the ULOF case.

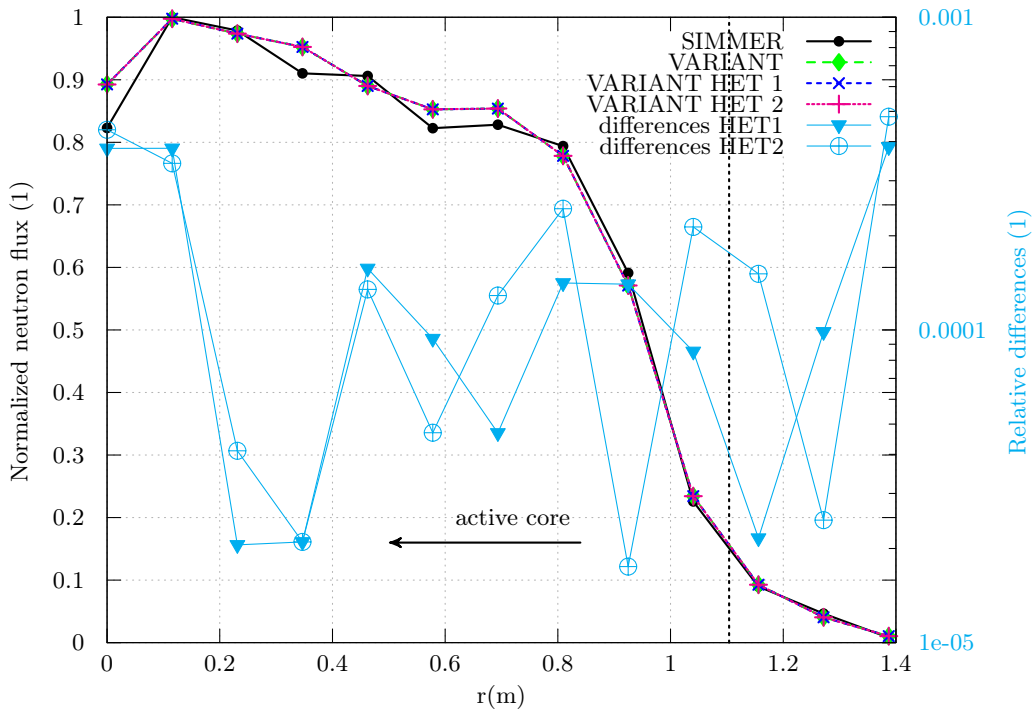
Axial and radial fast flux traverses are shown in figure 6.10. The agreement is indeed very good. Pointwise relative differences on the flux between VARIANT/KIN3D HOMO and HET 1 models are lower than 0.07% through the reactor. The second heterogeneous model is less precise, with a maximum relative error of 0.2%, if we exclude the very first axial mesh. There, errors are as high as 18%. I have already shown in section 5.2 that this heterogeneous VARIANT technique tends to lose efficiency, where the flux level is low and where the node is rather big. In this case, the first axial mesh belongs to a 0.5 m high heterogeneous node in a region, in which the flux is  $10^{-4}$  lower than the flux at the core center. I deem, however, that the accuracy of the VARIANT/KIN3D HET 2 model is still acceptable, as these high relative errors occur in low-importance, low-flux regions. A higher accuracy is always attainable by refining the heterogeneous mesh.

### Transient results

Reactivity traces are shown in figure 6.11. Good agreement is found passing to HET models. Throughout the transient, the maximum evaluated difference in reactivity between VARIANT/KIN3D HOMO and both HET models is less than 4 pcm.



(a) Axial flux traverse for fast flux for ring number two ( $r \approx 0.11$  m).



(b) Radial flux traverse for fast flux at core midplane ( $z \approx 3.1$  m).

Figure 6.10: Axial (a) and radial (b) fast flux distribution at the initial state evaluated by SIMMER and VARIANT models for the ULOF case. Relative differences are computed against VARIANT.

In particular, the HET 1 model, being more detailed, shows differences below 1.5 pcm. The same good agreement is found for the power traces. A single shape update in VARIANT/KIN3D HET 1 takes approximately 14s, and is even slower than the homogeneous VARIANT model at  $\approx 10$ s. Only 3.5s are instead needed for a VARIANT/KIN3D HET 2 flux shape update. That figure is still much higher than SIMMER-III but it is a major reduction with respect to the HOMO model, and with little loss of accuracy. This first example then shows, how the application of the VARIANT heterogeneous extension opens the possibility to use VARIANT as a neutronic solver for FR transient analyses.

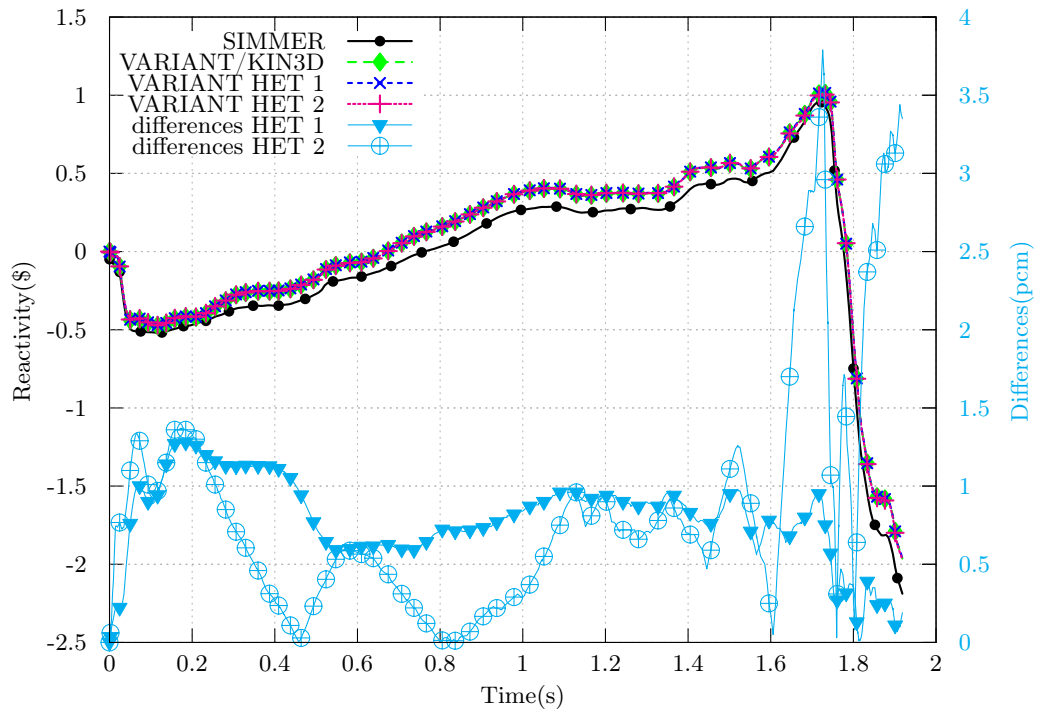


Figure 6.11: Transient reactivity (\$) as function of time evaluated with different models. Differences (pcm) are indicated on the secondary y-axis, they are computed against the VARIANT/KIN3D.



## Chapter 7

# Conclusion

HCDAs have been playing a major role in licensing of fast reactors. Despite their low probability, – in fact they are usually referred to as hypothetical – such accidents have been extensively investigated, as their consequences can be very severe.

As in fast reactors fuel is not in the most reactive configuration, core disruption and rearrangement during a transient – e.g. in case of fuel melting – might lead to a critical condition accompanied by introduction of a high reactivity rates and an increase of the reactor power by several orders of magnitude. Such a power burst generates an expanding mass of steel and fuel, in turn, accelerating the surrounding sodium, creating a sodium slug finally hitting the reactor vessel. An accurate prediction of such conceivable transients is of vital importance to the safety assessment of FRs.

In modeling such types of transients, it is therefore of primary importance to accurately model the neutron field, in order to have accurate estimates of reactivity and power density distribution during the transient.

To improve our understanding of fast reactor accidents, it is therefore necessary to use full three-dimensional models. This however requires fast computational tools.

The obvious approach is to speed up calculation by using parallel codes. That is why I coupled the actual version of SIMMER to a parallel neutron transport solver. I described the new coupled SIMMER/PARTISN code, and then I put it to test. The coupled code is first verified against existing SIMMER analyses. I showed the ability of the coupled code to correctly describe the neutron flux distribution. Differences with SIMMER code are negligible. Integral quantities, such as reactivity and power, also show a good agreement with SIMMER. I then validated the coupled code against experimental data for an ULOF accident in the EBR-II reactor. The results exhibit a good agreement with local – e.g. thermocouple signals – and global parameters – e.g. reactor power, mass flow rate.

Investigation of computational speedup shows mixed results. On the one hand, two-dimensional cases with a rather approximate neutronic description – i.e. reduced number of neutronic meshes, few energy groups, low order discrete ordinates – allow only mild speedups. Two-dimensional computations with a more refined neutronic description and/or three-dimensional geometry models benefitted the most from the parallel neutronic solver. Analysis of the EBR-II transient conducted with very fine neutronic meshes and with a discrete ordinate order of 8 is achieved. Total computational time for the SIMMER/PARTISN

coupled code is lower than 3 days, only 3% of which is spent into PARTISN computations. Normal SIMMER takes instead more than 7 days.

Several examples show the effectiveness of the new parallel option SIMMER/PARTISN. Moreover, it is not only possible to save computational time but the coupled code allows for a more detailed neutronic description of accidents.

Another approach is to use different type of neutron solvers. Nodal methods have shown their ability to solve the neutron transport equation with a sufficient accuracy on coarse neutronic meshes, e.g. the VARIANT code. Coupling SIMMER with VARIANT code has been achieved via the VARIANT kinetics module KIN3D. The nodal code is found to predict neutron fluxes with relative errors below 3% in the core region. Application of the coupled code to a transient analysis showed some differences in reactivity levels. These differences are below 30 pcm before the power peak. However, towards prompt criticality, even these small differences become important in case of ULOF studied. After the peak, differences increase up to 100 pcm.

As for the computational time, VARIANT code is found to be slower than SIMMER. Application of this coupling is only practical if computing times are reduced.

Using heterogeneous nodes has beneficial effects on computing times. Stationary results of this technique are promising. Evaluation of reactivity effects for the ESRF shows no deviations with the standard VARIANT code. Control rod insertions and sodium void reactivity effects differ by less than 1 pcm when using the heterogeneous extension. I found also that this method is much more accurate than the usual procedure to smear the different cross sections on one heterogeneous node. Furthermore, I showed that this extension can not only describe the mean power density on a heterogeneous node, but can also predict its spatial variation within the heterogeneous node itself. Differences in power level with the VARIANT code are less than few percent.

I obtained good results also when using a very coarse node to model the same reactor. Reactivity effects differ by less than 60 pcm. Power levels are in agreement with VARIANT within 10%. This level of accuracy is acceptable, and obtained with a reduction of the computational time of 60% compared to VARIANT.

This is a critical point for a possible application of the VARIANT code as neutron solver for SIMMER. The coupling with the heterogeneous extension of VARIANT exhibits no significative difference compared to standard VARIANT. Heterogeneous models predict the same neutron flux distribution. Reactivity values for heterogeneous models are in good agreement with maximum deviations of 1-4 pcm. Computational time for one, single shape update is reduced to one third compared to normal VARIANT. These cases show that VARIANT can be effectively used as neutron transport solver for accident analyses, and the heterogeneous node extension can provide quite a good speedup of the calculation.

VARIANT can provide a detail angular description of the neutron flux. The next step would be to investigate the influence of the various angular options the code can offer (e.g. higher order  $SP_N$ ,  $P_3$ ,  $P_5$ ) on HCDAs modeling. In this work, the heterogeneous VARIANT extension has been applied in the axial direction only. However, the extension is capable to model heterogeneities in the planar direction as well. With this option, coarse neutronic meshes in the radial direction might be used, further reducing the number of computational nodes in the neutronic domain and accelerating the computation. Full, three-dimensional



HCDAs modeling might then become possible with the SIMMER/KIN3D coupling. The parallel capability of the PARTISN code can also be investigated further. PARTISN allows the possibility to decompose the neutron energy variable as well. Each processor will then be assigned a spatial portion of the neutron mesh and only a limited number of energy groups, instead of the whole energy domain. In this way, the code can treat the energy variable in parallel. This option may then leads to even higher speedups.



# Bibliography

- Generation IV International Forum, 2015. URL [https://www.gen-4.org/gif/jcms/c\\_9260/public](https://www.gen-4.org/gif/jcms/c_9260/public).
- R. E. Alcouffe. Diffusion Synthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations. *Nuclear Science and Engineering*, 64:344–355, 1977.
- R. E. Alcouffe and R. S. Baker. *PARTISN: A Time-Dependent, Parallel Neutral Particle Transport Code System*. LA-UR-08-07258, March 2009. Revised March 2009.
- R.E. Alcouffe, R.S. Baker, F.W. Brinkley, D.R. Marr, R.D. O'Dell, and W.F. Walters. DANTSYS: A Diffusion Accelerated Neutral Particle Code System. Report LA-12969-M, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, 1995.
- G. Bandyopadhyay and J. A. Buzzell. The Role of Fission Gas and Fuel Melting in Fuel Response During Simulated Hypothetical Loss-of-Flow Transients. *Nuclear Technology*, 41(1):91–109, January 1980.
- Charles R. Bell, Jay E. Boudreau, Robert D. Burns III, and Larry B. Luck. SIMMER-II Analysis of LMFBR Postdisassembly Expansion. In *International Meeting on Nuclear Power Reactor Safety Topical Meeting*, October 16-19 1978. Brussels, Belgium.
- Frederick R. Best, David Wayne, and Carl Erdman. A Fuel Freezing Model for Liquid-Metal Fast Breeder Reactor Hypothetical Core Disruptive Accidents. *Nuclear Science and Engineering*, 89(1):49–60, January 1984.
- H. A. Bethe and J. H. Tait. An Estimate of the Order of Magnitude of Vigorous Interaction Expected Should the Core of a Fast Reactor Collapse. Report, U.K.A.E.A. Reactor Group, April 1956.
- M. Th. Beyens-Henry. Characteristics of Fast Neutron Power Reactors. Technical report, EUROPEAN ATOMIC ENERGY COMMUNITY - EURATOM, 1966.
- T D Beynon. The nuclear physics of fast reactors. *Reports on Progress in Physics*, 37(8): 951, 1974. URL <http://stacks.iop.org/0034-4885/37/i=8/a=001>.
- W. R. Bohl. Some Recriticality Studies with SIMMER-II. In *International Meeting on Fast Reactor Safety Technology*, August 19-23 1979. Seattle, Washington.

- Maurizio Bottoni, Burkhardt Dorr, Christoph Homann, Franz Huber, Karl Mattes, F. W. Pepler, and Dankward Struwe. Experimental and Numerical Investigations of Sodium Boiling Experiments in Pin Bundle Geometry. *Nuclear Technology*, 89(1):56–82, January 1990.
- J. E. Boudreau and J. F. Jackson. Recriticality Considerations in LMFBR Accidents. LA-UR 74-520, Los Alamos Scientific Laboratory, 1974.
- L. Briggs, C. Choi, W. Hu, L. Maas, W. Maschek, B. Merk, and K. Mikityuk. Benchmark Analyses of the Shutdown Heat Removal Tests Performed in the EBR-II Reactor. In *Proc. Int. Conf. FR13*, March 4-7 2013. Paris, France.
- L. L. Briggs, Jr. W. F. Miller, and E. E. Lewis. Ray-Effect Mitigation in Discrete Ordinate-Like Angular Finite Element Approximations in Neutron Transport. *Nuclear Science and Engineering*, 57:205–217, 1975.
- G. Buckel, E. Hesselschwerdt, E. Kiefhaber, S. Kleinheins, and W. Maschek. A new SIMMER-III Version with Improved Neutronics Solution Algorithms. Report FZKA 6290, Forschungszentrum Karlsruhe, June 1999.
- C.B. Carrico, E.E. Lewis, and G. Palmiotti. Three-Dimensional Variational Nodal Transport Methods for Cartesian, Triangular, and Hexagonal Criticality Calculations. *Nuclear Science and Engineering*, 111:168–179, 1992.
- T. C. Chawla and S. H. Chan. Heat Transfer From Vertical/Inclined Boundaries of Heat-Generating Boiling Pools. *Journal Heat Transfer*, 104(3):465–473, August 1982.
- X.-N. Chen, A. Rineiski, F. Gabrielli, L. Andriolo, B. Vezzoni, R. Li, W. Maschek, and E. Kiefhaber. Fuel-steel mixing and radial mesh effects in power excursion simulations. *Annals of Nuclear Energy*, 90:26 – 31, 2016. ISSN 0306-4549. doi: <http://dx.doi.org/10.1016/j.anucene.2015.11.041>. URL <http://www.sciencedirect.com/science/article/pii/S0306454915300013>.
- Thomas B. Cochran, Harold A. Feiveson, Walt Patterson, Gennadi Pshakin, M.V. Ramana, Mycle Schneider, Tatsujiro Suzuki, and Frank von Hippel. Fast Breeder Reactor Programs: History and Status. Report, International Panel on Fissile Materials, February 2010. URL [http://fissilematerials.org/library/2010/02/fast\\_breeder\\_reactor\\_programs\\_.html](http://fissilematerials.org/library/2010/02/fast_breeder_reactor_programs_.html).
- K.L. Derstine. DIF3D: A Code to Solve One-, Two- and Three-Dimensional Finite Difference Diffusion Theory Problems. Report ANL-82-64, Argonne National Laboratory, 1984.
- I. Dilber and E. E. Lewis. Variational Nodal Methods for Neutron Transport. *Nuclear Science and Engineering*, 91:132, 1985.
- J.Y. Doriath. ERANOS1: the Advanced European System of Codes for Reactor Physics. In *Proc. Int. Conf. on Mathematical Methods and Supercomputing in Nuclear Application*, Karlsruhe, Germany, 1993.

- Michael Epstein, Michael A. Grolmes, Robert E. Henry, and Hans K. Fauske. Transient Freezing of a Flowing Ceramic Fuel in a Steel Channel. *Nuclear Science and Engineering*, 61(3):310–323, November 1973.
- T. H. Fanning and G. Palmiotti. Variational Nodal Transport Method with Heterogeneous Nodes. *Nuclear Science and Engineering*, 127:154, 1997.
- H. K. Fauske. Assessment of accident energetics in LMFBR core-disruptive accidents. *Nuclear Engineering and Design*, 42(1):19 – 29, 1977. ISSN 0029-5493. doi: [http://dx.doi.org/10.1016/0029-5493\(77\)90058-9](http://dx.doi.org/10.1016/0029-5493(77)90058-9). URL <http://www.sciencedirect.com/science/article/pii/0029549377900589>.
- Hans K. Fauske. The Role of Energetic Mixed-Oxide-Fuel-Sodium Thermal Interaction In Liquid Metal Fast Breeder Reactor Safety. Report, Reactor Analysis and Safety Division, Argonne National Laboratory, Argonne, Illinois, USA, 1976.
- M. Flad, D. Zhang, C. Matzerath Boccaccini, F. Gabrielli, B. Vezzoni, W. Maschek, G. Brilant, and H. Bonneville. ESFR Sever Accidents Analyses with SIMMER-III. In *FAST REACTORS AND RELATED FUEL CYCLES: SAFE TECHNOLOGIES AND SUSTAINABLE SCENARIOS (FR13)*, Paris, France, March 2013. International Atomic Energy Agency.
- T. Ginsberg, Jr. O. C. Jones, and J. C. Chen. Flow Behavior of Volume-Heated Boiling Pools: Implications with Respect to Transition Phase Accident Conditions. *Nuclear Technology*, 46(3):391–398, December 1979.
- S. J. Hakim, T. H. Bauer, and R. O. McNary. An Analysis of the Thermal Behavior of a Stagnant Pool for Fast Reactor Safety. *Nuclear Technology*, 52(1):43–56, January 1981.
- R.J. Henninger, R.E. Alcouffe, U.S. Nuclear Regulatory Commission. Office of Nuclear Regulatory Research, and Los Alamos Scientific Laboratory. *Disassembly Phase Energetics: An Examination of the Impact of SIMMER Models and Assumptions*. U.S. Nuclear Regulatory Commission, 1979. URL <https://books.google.de/books?id=Q30MHAACAAJ>.
- A. F. Henry. The Application of Reactor Kinetics to the Analysis of Experiments. *Nuclear Science and Engineering*, 3:52–70, 1958.
- Gerhard Heusener, Günther Kessler, and Heinrich Lauber. *Analysis of hypothetical accidents for SNR-300*. Kernforschungszentrum Karlsruhe ; 1834. Ges. f. Kernforschung mbH, Karlsruhe, 1973.
- E. P. Hicks and D. C. Menzies. Theoretical Studies on the Fast Reactor Maximum Accident. ANL- 7120, Argonne National Laboratory, 1965.
- K. M. Horst. SOUTHWEST EXPERIMENTAL FAST OXIDE REACTOR DEVELOPMENT PROGRAM. GEAP- 4594, Atomic Power Equipment Department, General Electric, July 1964.
- IEA. World Energy Outlook 2014. <http://www.worldenergyoutlook.org/publications/weo-2014/>, November 2014.

- IEA. World Energy Outlook Special Report 2015: Energy and Climate Change. <http://www.worldenergyoutlook.org/energyclimate/>, June 2015.
- Fast Reactor Database 2006 Update*. International Atomic Energy Agency, Vienna, 2007. IAEA-TECDOC-1531.
- J. F. Jackson, M. G Stevenson, and J. F. Marchaterre. Trends in LMFBR Hypothetical-Accident Analysis. In *Proc. Fast Reactor Safety Mtg.*, Beverly Hills, California, April 1974.
- W. M. Jacobi and Y. S. Tang. Clinch River Breeder Reactor: a combined power and fuel source. In *national meeting on the American Institute of Chemical Engineers*. Westinghouse Electric Corp., Madison, Pa. (USA). Advanced Reactors Div., 1974.
- V. Z. Jankus. A THEORETICAL STUDY OF DESTRUCTIVE NUCLEAR BURSTS IN FAST POWER REACTORS. Report ANL-6512, Argonne National Laboratory, Reactor Engineering Division, 9700 South Cass Avenue, Argonne Illinois, February 1962.
- Alan V. Jones. Calculation of Fuel Penetration and Freezing in Channels with a Two-Phase Flow Code. *Nuclear Science and Engineering*, 105(2):105–122, June 1990.
- Kamiyama Kenji. Improvement of SIMMER-III Freezing Model. Research Document JNC TN9400 2003 - 039, Japan Nuclear Cycle Development Institute, June 2003.
- R. B. Kidman, R. E. Schenter, R. W. Hardie, and W. W. Little. The Shielding Factor Method of Generating Multigroup Cross Sections for Fast Reactor Analysis. *Nuclear Science and Engineering*, 48(2):189–201, June 1972.
- L. J. Koch. *Experimental Breeder Reactor-II: An Integrated Experimental Fast Reactor Nuclear Power Station*. American Nuclear Society, November 2008.
- L. J. Koch, H. O. Monson, D. Okrent, M. Levenson, and W. R. Simmons. HAZARD SUMMARY REPORT EXPERIMENTAL BREEDER REACTOR II (EBR-II). ANL-5719, Argonne National Laboratory, May 1957.
- S. Kondo, A. Furatani, and M. Ishikawa. SIMMER-II application and validation studies in Japan for energetics accomodation of sever LMFBR accidents. In *Proc. Fast Reactor Safety Mtg.*, Knoxville, Tennessee, April 1985.
- K. D. Lathrop. Ray Effects in Discrete Ordinates Equations. *Nuclear Science and Engineering*, 32:357–369, 1968.
- K. D. Lathrop. Remedies for Ray Effects. *Nuclear Science and Engineering*, 45:255–268, 1971.
- Kirsten F. Laurin-Kovitz and E.E. Lewis. Perturbation Theory Based on the Variational Nodal Transport Method in X-Y-Z Geometry. In *Proc. Int. Conf. on Mathematics and Computations, Reactor Phys*, Portland, Oregon, 1995.

- 
- R. D. Lawrence. Progress in Nodal Methods for the Solution of the Neutron Transport Equations. *Progress in Nuclear Energy*, 17:271–301, 1986.
- W. B. Loewenstein. The Physics Design of EBR-II. ANL- 6383, Argonne National Laboratory, July 1961.
- L. B. Luck, C.R. Bell, M. W. Asprey, and G. P. DeVault. A TRANSITION-PHASE CALCULATION OF A LARGE, HETEROGENEOUS-CORE LMFBR. In *American Nuclear Society Winter Meeting*, November 1981. San Francisco, California.
- Marco Marchetti and Andrei Rineiski. Variational Nodal Method with Heterogeneous Nodes: Application to Reactor Analyses. *Transaction of the American Nuclear Society*, 107:535–536, November 11-15 2012. San Diego, California.
- Marco Marchetti, Fabrizio Gabrielli, Andrei Rineiski, and Werner Maschek. The SIMMER/PARTISN Capability for Severe Accident Analyses. In *The Role of Reactor Physics toward a Sustainable Future - PHYSOR 2014*, September 28th - October 3rd 2014. Kyoto, Japan.
- Werner Maschek. A brief review of transition phase technology. Technical report, Kernforschungszentrum, Karlsruhe, 1982.
- Werner Maschek and Margaret W. Asprey. SIMMER-II Recriticality Analyses for a Homogeneous Core of the 300-MW(electric) Class. *Nuclear Technology*, 63(2):330–336, November 1983.
- Werner Maschek, Claus Dieter Munz, and Leonhard Meyer. Investigations of Sloshing Fluid Motions in Pools Related to Recriticalities in Liquid-Metal Fast Breeder Reactor Core Meltdown Accidents. *Nuclear Technology*, 98(1):27–43, April 1992.
- K. Morita and E.A. Fischer. Thermodynamic properties and equations of state for fast reactor safety analysis: Part I: Analytic equation-of-state model. *Nuclear Engineering and Design*, 183(3):177 – 191, 1998. ISSN 0029-5493. doi: [http://dx.doi.org/10.1016/S0029-5493\(98\)00175-7](http://dx.doi.org/10.1016/S0029-5493(98)00175-7). URL <http://www.sciencedirect.com/science/article/pii/S0029549398001757>.
- K. Morita, Y. Tobita, S. Kondo, E A Fischer, and K. Thurnay. SIMMER-III Analytic Equation-of-State Model. Research Document JNC TN9400 2000 - 005, Japan Nuclear Cycle Development Institute, May 1999.
- K. Morita, Y. Tobita, S. Kondo, and E A Fischer. SIMMER-III Analytic Thermophysical Property Model. Research Document JNC TN9400 2000-04, Japan Nuclear Cycle Development Institute, 2000.
- Nobuyuki Nonaka and Ikken Sato. Improvement of Evaluation Method for Initiating-Phase Energetics Based on CABRI-1 In-Pile Experiments. *Nuclear Technology*, 98(1):54–69, April 1992.
- A. Jr. Padilla. Transient Analysis of Fuel-Sodium Interaction. *Transaction of the American Nuclear Society*, 13, 1970.

- G. Palmiotti, E. E. Lewis, and C. B. Carrico. VARIANT: VARIational Anisotropic Nodal Transport for Multidimensional Cartesian and Hexagonal Geometry Calculation. Report ANL-95/40, Argonne National Laboratory, 9700 South Cass Avenue, Argonne Illinois 60439, October 1995.
- H. P. Planchon, G. H. Golden, J. I. Sackett, D. Mohr, L. K. Chang, E. E. Feldman, and P. R. Betten. Results and implications of the experimental breeder reactor ii inherent safety demonstration tests. *Nuclear Science and Engineering*, 100(4):549 – 557, December 1988.
- G. Rimpault. Physics Documentation of ERANOS - The ECCO Cell Code ERANOS : Manuel des Methodes - Le Code de Cellule ECCO. Report SPRC/LEPh 97-001, CEA, 1997.
- A. Rineiski. Intra-Nodal Reactivity Calculations Based on the Variational Nodal Method. In *Proc. Int. Conf. M&C, Salt Lake City Utah*, September 2001.
- Andrei Rineiski. KIN3D: Module de Cinetique Spatiale et de Perturbations pour TGV2. Report SPRC/LEPh 97-203, CEA, 1997.
- Andrei Rineiski, Barbara Vezzoni, Daling Zhang, Xue Nong Chen, Fabrizio Gabrielli, and Marco Marchetti. Sodium void effect reduction and minor actinide incinerator in esfr. *Transaction of the American Nuclear Society*, 104:720–721, June 26-30 2011. Hollywood, Florida.
- P. Royl and B. Kuczera. Analysis of hypothetical overpower accidents in the SNR-300 Mark 1A core with modeling of failure incoherence. *Nuclear Engineering and Design*, 43(2): 239 – 248, 1977. ISSN 0029-5493. doi: [http://dx.doi.org/10.1016/0029-5493\(77\)90002-4](http://dx.doi.org/10.1016/0029-5493(77)90002-4). URL <http://www.sciencedirect.com/science/article/pii/0029549377900024>.
- M. Segev. A Theory of Resonance-Group Self-Shielding. *Nuclear Science and Engineering*, 56(1):72–82, January 1975.
- W. T. Sha and T. H. Huges. VENUS: A Two-Dimensional Coupled Neutronics-Hydrodynamics Computer Program for Fast-Reactor Power Excursions. Report ANL-7701, Argonne National Laboratory, 9700 South Cass Avenue, Argonne Illinois 60439, October 1970.
- D. E. Simpson, J. W. Hagan, and A. E. Waltar. Preliminary Analysis of Postulated Maximum Accidents for the FFTF. BNWL - 760, BATTELLE NORTHWEST LABORATORY, November 1968.
- D. E. Simpson, A. E. Walter, and Padilla A. Assessment of Magnitude and Uncertainties of Hypothetical Accidents for the FFTF. Technical report, Handford Engineering Development Laboratory, March 1971.
- Tohru Suzuki, Yoshiharu Tobita, and Ryodai Nakai. Evaluation of recriticality behavior in the material-relocation phase for japan sodium-cooled fast reactor. *Journal of Nuclear Science and Technology*, 52(11):1448–1459, 2015.



- 
- Toshikazu Takeda and Hideaki Ikeda. 3-D Neutron Transport Benchmarks. Technical Report NEARCP-L-330, OECD/NEA, March 1991.
- A. M. Tentner. The SAS4A LMFBR Whole Core Accident Analyses. In *Int. Topl. Mtg. Fast Reactor Safety*. American Nuclear Society, April 21-25 1985.
- T. G. Theofanous and C. R. Bell. An Assessment of CRBR Core Disruptive Accident Energetics. In *Int. Topl. Mtg. Fast Reactor Safety*, volume 1, Knoxville, Tennessee, April 21-25 1985. American Nuclear Society.
- Y. Tobita, K. Morita, Ken-ichi Kawada, Niwa Hajime, and N. Nonaka. Evaluation of CDA Energetics in the Prototype LMFBR with Latest Knowledge and Tools. In *7th International Conference on Nuclear Engineering*, Tokyo, Japan, April 19-23 1999.
- Yoshiharu Tobita, Hidemasa Yamano, and Ikken Sato. Analytical study on elimination of severe recriticalities in large scale LMFBRs with enhancement of fuel discharge. *Nuclear Engineering and Design*, 238(1):57 – 65, 2008.
- A. E. Waltar, Padilla A., and R. J. Shields. MELT-II A Two-Dimensional Neutronics-Heat Transfer Computer Program for Fast Reactor Safety Analysis. , WADCO Corp., 1971.
- Alan E. Waltar and Albert B. Reynolds. *Fast Breeder Reactors*. Pergamon international library of science, technology, engineering, and social studies. Pergamon Press, New York, 1981. ISBN 0-08-025982-0; 0-08-025983-9.
- T. R. Wehner and C. R. Bell. SIMMER-II ANALYSIS OF TRANSITION-PHASE EXPERIMENTS. In *International Topical Meeting On Fast Reactor Safety*, April 21-25 1985. Holiday Inn World's Fair, Knoxville, Tennessee.
- Jr. Wise, W.R., J.F. Proctor, and L.P. Walker. Enrico Fermi shield-plug response to a 1000-lb TNT accident. *Experimental Mechanics*, 3(10):245–252, 1963. ISSN 0014-4851. doi: 10.1007/BF02325871. URL <http://dx.doi.org/10.1007/BF02325871>.
- Steven A. Wright, Erhard A. Fischer, Peter K. Mast, and Gustav Schumacher. In-Pile Determination of Fuel Disruption Mechanisms Under LMFBR Loss-of- Flow Accident Conditions. *Nuclear Technology*, 71(1):326–340, October 1985a.
- Steven A. Wright, Gustav Schumacher, and Peter R. Henkel. In-Pile Observations of Fuel and Clad Relocation During LMFBR Core Disruptive Accidents in the STAR Experiments. *Nuclear Technology*, 71(1):187–216, October 1985b.
- H. Yamano, S. Fujita, Y. Tobita, K. Kamiyama, Sa. Kondo, K. Morita, E.A. Fischer, D. J. Brear, N. Shirakawa, X. Cao, and M. Sugaya. SIMMER-III: A Computer Program for LMFR Core Disruptive Accident Analysis. Research Document JNC TN9400 2003-071, Japan Nuclear Cycle Development Institute, August 2003a.
- H. Yamano, S. Fujita, Y. Tobita, Sa. Kondo, K. Morita, M. Sugaya, M. Mizuno, S. Hosono, and T. Kondo. SIMMER-IV: A Three-Dimensional Computer Program for LMFR Core Disruptive Accident Analysis. Research Document JNC TN9400 2003-070, Japan Nuclear Cycle Development Institute, August 2003b.

Hidemasa Yamano, Yoshiharu Tobita, Satoshi Fujita, and Werner Maschek. First 3-D calculation of core disruptive accident in a large-scale sodium-cooled fast reactor. *Annals of Nuclear Energy*, 36(3):337 – 343, 2009. ISSN 0306-4549. PHYSOR 2008.

# Nomenclature

## General

$\delta_{ij}$	Kronecker delta function
$\langle f, g \rangle$	Scalar product between function $f$ and $g$
$\mathbf{r}$	Neutron position
$\Omega$	Neutron direction of travel
$E$	Neutron energy
$f_{npar}$	Fraction of the computation that can not be parallelized
$g$	Energy group
$k_0$	Multiplication factor at steady state
$p$	Pressure
$P_i(x)$	Legendre polynomial of order $i$
$PE$	Number of processors
$S$	Parallel speedup
$S_{max}$	Maximum achievable parallel speedup
$t$	Time
$T_{parallel}$	Parallel computational time
$T_{serial}$	Serial computational time
$V$	Volume
$v$	Neutron velocity
$W$	Mechanical work
$\mathbf{A}$	A matrix
$\mathbf{A}^t$	Transpose of matrix $\mathbf{A}$

$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$
<b>Delayed Neutrons</b>	
$\beta$	Total delayed neutron fraction
$\beta_k$	Delayed neutron factor for delayed neutron family $k$
$\chi_{i,g}^d$	Delayed fission spectrum for family $i$ into energy group $g$
$\chi_k^d(E)$	Delayed fission spectrum for delayed family
$\lambda_k$	Decay constant for delayed neutron family $k$
$C_k(\mathbf{r}, t)$	Spatial delayed neutron precursors concentration for family $k$
ID	Number of delayed neutron precursors families

### Heterogeneous Variational Nodal Method

$\xi_i$	Expansion moment on nodal basis function $f_i(\mathbf{r}, \boldsymbol{\Omega})$
$\xi_{k,i}$	Expansion moment on sub-node $k$ onto sub-node basis function $f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})$
$C_{k,m,i}$	Moment of basis function $f_i$ onto sub-node basis function $f_{k,m}$ for sub-node $k$
$f_i(\mathbf{r}, \boldsymbol{\Omega})$	Basis function over a generic sub-node $k$
$f_{k,i}(\mathbf{r}, \boldsymbol{\Omega})$	Basis function over a generic sub-node $k$
$W_i(x, y)$	$x$ and $y$ directed basis function
$Z_i(z)$	Axial directed basis function
$Z_{k,i}(x)$	Axial directed basis function on sub-node $k$

### Quasistatic and Point Kinetics

$\beta^{eff}(t)$	Total effective delayed neutron fraction
$\beta_k^{eff}(t)$	Effective delayed neutron fractions for family $k$
$\gamma$	Quasistatic constraint
$\Lambda(t)$	Neutron generation time
$\mathbf{F}_{dk}[\cdot]$	Quasi-stationary delayed neutron source operator, for family $k$
$\mathbf{F}_d[\cdot]$	Total quasi-stationary delayed neutron source operator
$\mathbf{F}_p[\cdot]$	Prompt neutron fission operator
$\mathbf{M}[\cdot]$	Transport operator

$\mathbf{S}_d[\cdot]$	Delayed neutron source operator
$\phi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$	Angular flux
$\Phi(\mathbf{r}, E, t)$	Scalar flux
$\rho(t)$	Reactivity
$\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t)$	Flux shape
$\varphi^*(\mathbf{r}, \boldsymbol{\Omega}, E)$	Stationary angular adjoint flux
$\varphi^*(\mathbf{r}, E)$	Stationary scalar adjoint flux
$c_k(t)$	Effective delayed neutron precursors concentrations
$F(t)$	Weighted total fission source production
$f_v$	Velocities adjustment-factor
$g_k$	Precursor concentration adjustment-factor
$h_k$	Delayed neutron fraction adjustment-factor
$N(t)$	Point kinetics amplitude
$Q_g(\mathbf{r}, t)$	External group neutron source at time $t$
<b>Cross Sections</b>	
$\chi^p(E)$	Prompt fission spectrum
$\chi_g^p$	Prompt fission spectrum into energy group $g$
$\chi_g^t$	Total (prompt and delayed) fission spectrum on energy group $g$
$\eta_i(E)$	Number of fission neutrons emitted per neutron absorbed for specific isotope $i$
$\nu$	Number of neutrons emitted per fission event
$\nu_g$	Average number of neutron emitted by fission caused by neutrons in energy group $g$
$\sigma_a^i(E)$	Microscopic absorption cross section for isotope $i$
$\Sigma_f(\mathbf{r}, E, t)$	Macroscopic fission cross section
$\sigma_f^i(E)$	Microscopic fission cross section for isotope $i$
$\Sigma_s(\mathbf{r}, E' \rightarrow E, \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', t)$	Double-differential macroscopic scattering cross section
$\Sigma_t(\mathbf{r}, E, t)$	Macroscopic total cross section
$\Sigma_{\text{power}}$	Power cross section

$\Sigma_{f,g}(\mathbf{r})$	Macroscopic fission cross section for energy group $g$
$\Sigma_{s,gg'}(\mathbf{r})$	Macroscopic scattering cross section from energy group $g'$ to group $g$
$\Sigma_{s,g}(\mathbf{r})$	Macroscopic scattering cross section, isotropic scattering
$\Sigma_{t,g}(\mathbf{r})$	Macroscopic total cross section for energy group $g$
$D_g(\mathbf{r})$	Diffusion coefficient for energy group $g$

### Diffusion-Synthetic Acceleration

$\mathbf{J}_g^l(\mathbf{r})$	Energy group neutron current at inner iteration $l$
$\Phi_g^k(\mathbf{r})$	Energy group scalar flux during outer iteration $k$
$\Psi_g^l$	Energy group angular flux at inner iteration $l$
$F^k(\mathbf{r})$	Spatial fission source distribution at outer iteration $k$
$k$	Outer iteration index
$l$	Inner iteration index
$R_g^l(\mathbf{r})$	Source correction scheme correction term

### Cross Section Shielding

$\bar{\sigma}_i$	Shielded microscopic cross section for isotope $i$
$\bar{\sigma}_{t,i}$	Shielded total cross section for isotope $i$
$\sigma_i^\infty$	Infinitely diluted microscopic cross section for isotope $i$
$\sigma_{0,i}$	Background cross section for isotope $i$
$f_s^i(\sigma_{0,i}, T_i)$	Shielding factor for isotope $i$
$N_i$	Atomic density of isotope $i$
$T_i$	Temperature of isotope $i$

### Variational Nodal Method

$\psi_\gamma^+$	Even-parity flux moment vector onto nodal surface $\gamma$
$\psi^+$	Global vector of even-parity flux moments on nodal volume
$\psi_\gamma^-$	Global vector of odd-parity flux moments on nodal surface $\gamma$
$\mathbf{j}_\gamma^+$	Outgoing partial currents moments on nodal interface $\gamma$
$\mathbf{j}_\gamma^-$	Incoming partial currents moments on nodal interface $\gamma$

$\mathbf{n}_\gamma$	Outward directed normal vector for nodal surface $\gamma$
$\mathbf{s}$	Global vector of neutron flux moments
$\Psi_g^+(\mathbf{r}, \boldsymbol{\Omega})$	Even-parity flux
$\Gamma_\gamma$	Nodal surface
$\mathbf{J}_g(\mathbf{r})$	Group neutron current
$\Psi_g^-(\mathbf{r}, \boldsymbol{\Omega})$	Odd-parity flux
$\Phi_g(\mathbf{r})$	Energy group scalar flux
$\psi_{im}^+$	Even-parity flux moments
$\psi_{jn_\gamma}^-$	Odd-parity flux moments
$\Psi_g(\mathbf{r}, \boldsymbol{\Omega})$	Energy group $g$ angular flux
$d_i(\mathbf{r})$	Spatial basis function over node volume
$g_m^+(\boldsymbol{\Omega})$	Angular basis function for even-parity flux
$h(\mathbf{r})_{j_\gamma}$	Spatial basis function over node surface $j_\gamma$
$k_{m_\gamma}(\boldsymbol{\Omega})$	Angular basis function for odd-parity flux
$S_g(\mathbf{r})$	Neutron source into group $g$
$s_i$	Neutron flux source moments
$\omega(\mathbf{r})$	Exponential term of the fitting function $P(\mathbf{r}, t)$
$\tilde{\chi}_g(\mathbf{r}, t_1)$	Artificial prompt fission spectrum into energy group $g$ at time $t_1$
$\tilde{\Sigma}_{t,g}(\mathbf{r}, t_1)$	Artificial total macroscopic cross section at $t_1$ for group $g$
$\tilde{Q}_g(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$	Artificial external neutron source between times $t_0$ and $t_1$ for energy group $g$
$\tilde{Q}_g^+(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$	Even-parity artificial external neutron source between $t_0$ and $t_1$
$\tilde{Q}_g^-(\mathbf{r}, \boldsymbol{\Omega}, t_0, t_1)$	Odd-parity artificial external neutron source between $t_0$ and $t_1$
$F(\mathbf{r}, t)$	Neutron fission source at time $t$ and point $\mathbf{r}$
$P(\mathbf{r}, t)$	Flux fitting function






# Acronyms

<b>ADS</b> Accelerator Driven System .....	45
<b>ANL</b> Argonne National Laboratory.....	25, 74
<b>BR</b> Breeding Ratio .....	2, 3
<b>CDA</b> Core Disruptive Accident.....	10, 21, 24, 49, 69
<b>CR</b> Control Rod .....	45, 60, 91, 96–102, 105, 109
<b>CRBR</b> Clinch River Breeder Reactor.....	21
<b>DBA</b> Design Basis Accident .....	5, 6
<b>DSA</b> Diffusion Synthetic Acceleration .....	37, 38
<b>EBR-I</b> Experimental Breeder Reactor-I .....	3
<b>EBR-II</b> Experimental Breeder Reactor-II.....	3, 11, 19, 20, 74, 127
<b>ECCS</b> Emergency Core Cooling System .....	5
<b>ESFR</b> European Sodium Fast Reactor.....	23, 66, 71, 98, 99, 103, 128
<b>FBR</b> Fast Breeder Reactor .....	3, 20
<b>FCI</b> Fuel-Coolant Interaction.....	15
<b>FFTF</b> Fast Flux Test Facility .....	20, 21
<b>FR</b> Fast Reactor .....	1–6, 9, 11, 17, 19–21, 26, 49, 74, 116, 121, 124, 127
<b>GFR</b> Gas Fast Reactor .....	1
<b>GIF</b> Generation-IV International Forum .....	1, 3
<b>HCDA</b> Hypothetical Core Disruptive Accident .....	6, 7, 13, 18, 19, 71, 127–129

<b>IAEA</b> International Atomic Energy Agency.....	3, 74
<b>IEA</b> International Energy Agency.....	1
<b>IHX</b> Intermediate Heat Exchanger.....	4, 5, 74
<b>IQS</b> Improved Quasistatic Method.....	30, 31
<b>LANL</b> Los-Alamos National Laboratory.....	37
<b>LFR</b> Lead Fast Reactor.....	1
<b>LMFBR</b> Liquid Metal Fast Breeder Reactor.....	4, 21–23, 56, 59
<b>LOF</b> Loss of Flow accident.....	7, 15, 20–23
<b>LOF-d-TOP</b> Loss of Flow driven Transient Overpower accident.....	21, 22
<b>LOHS</b> Loss of Heat Sink.....	7
<b>LWR</b> Light Water Reactor.....	5, 6
<b>MA</b> Minor Actinide.....	3
<b>MFR</b> Molten Fuel Fast Reactor.....	1
<b>PPS</b> Plant Protection System.....	6, 7
<b>QS</b> Quasistatic Method.....	30, 32, 35, 36, 48, 51, 52, 54, 55
<b>SCWR</b> Supercritical Water Reactor.....	1
<b>SEFOR</b> Southwest Experimental Fast Oxide Reactor.....	19, 20
<b>SFR</b> Sodium Fast Reactor.....	1, 20, 24
<b>SHRT</b> EBR-II Shutdown Heat Removal Testing.....	76
<b>TOP</b> Transient Overpower accident.....	7, 8, 20–22
<b>TP</b> Transition Phase.....	70, 71
<b>TUC</b> Transient Under Cooling accident.....	7, 9
<b>ULOF</b> Unprotected Loss of Flow accident.....	7, 11, 15, 17, 20, 24, 59, 66, 71, 76, 82, 116, 127, 128
<b>VHTR</b> Very High Temperature Reactor.....	1
<b>VNM</b> Variational Nodal Method.....	39, 40, 44, 45, 89–91, 96





Modeling of Fast Reactor accidents is a complicated task, yet it is one of the most important steps in order to build safe reactors. The outcome of an accident is determined by different physical phenomena, all acting at almost the same time. Safety analysts must track all these different phenomena. Multi-physics codes have been developed for this task. They must contain accurate models for fluid-dynamics, neutronics, and structures.

Past and recent analyses have been limited to the approximate description of the neutronic field, for example by using a rough description of the energy and/or of the angular dependence of the neutron flux. In this work, different neutronic solvers are selected and coupled into a general multi-physics code for fast reactor accident analysis. Performances of each of them is then assessed. Some emphasis has been put also in assessing the speed of these solvers for determining the neutron flux. This is a particular important aspect, as there has been a tendency to abandon simplified two-dimensional reactor models in favor of full three-dimensional ones, increasing therefore the computational burden. Here it is shown that using efficient neutronic solvers, three-dimensional analyses of such accidental sequences might become practical.

ISSN 1869-9669  
ISBN 978-3-7315-0611-9

