



HAL
open science

Coupling between Monte Carlo neutron transport and thermal-hydraulics for the simulation of transients due to reactivity insertions

Margaux Faucher

► **To cite this version:**

Margaux Faucher. Coupling between Monte Carlo neutron transport and thermal-hydraulics for the simulation of transients due to reactivity insertions. Physics [physics]. Université Paris Saclay (ComUE), 2019. English. ⟨NNT : 2019SACLS387⟩. ⟨tel-02406396⟩

HAL Id: tel-02406396

<https://theses.hal.science/tel-02406396v1>

Submitted on 12 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Coupling between Monte Carlo neutron transport and thermal-hydraulics for the simulation of transients due to reactivity insertions

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

Ecole doctorale n°576 Particules, Hadrons, Énergie, Noyau, Instrumentation, Image,
Cosmos et Simulation (PHENIICS)
Spécialité de doctorat : Énergie nucléaire

Thèse présentée et soutenue à Saclay, le 18 octobre 2019, par

MARGAUX FAUCHER

Composition du Jury :

Sandra Dulla Professeur, Politecnico di Torino (Italie)	Rapporteur
Elsa Merle Professeur (HDR), CNRS/LPSC	Rapporteur
Cheikh M'Backé Diop Directeur de recherche, CEA/DEN	Président
Jan Dufek Professeur, KTH Royal Institute of Technology (Suède)	Examineur
Adrien Gruel Ingénieur-chercheur, CEA/DEN	Examineur
Emmanuel Gobet Professeur (HDR), École Polytechnique	Directeur de thèse
Andrea Zoia Ingénieur-chercheur (HDR), CEA/DEN	Directeur de thèse
Davide Mancusi Ingénieur-chercheur, CEA/DEN	Encadrant

Acknowledgements

I am writing these lines while this Ph.D. is ending. This is the moment when I am thinking about all the people who were part of this adventure. Whether for scientific help or for friend support, I am thinking about a lot of people.

I would like to start by thanking Elsa Merle and Sandra Dulla for reviewing my thesis. I am deeply grateful for their careful reading and their comments on it. I am also grateful to Adrien Gruel and Jan Dufek for being part of the examining committee, and to Cheikh Diop for assuming the presidency of the jury.

It was an honour for me to do this thesis under the direction of Andrea Zoia. Thank you for giving me this opportunity and for all the interesting discussions we had and the ideas we shared. I would like to express my gratitude to Emmanuel Gobet for accepting the co-direction of my thesis. I have learnt a lot during our work sessions at École Polytechnique and I really want to thank you for the time you took when I had very little knowledge in the field of stochastic differential equations. To my supervisor Davide Mancusi, you asked me what I regret about this thesis, I can tell you that doing it under your supervision is the first thing I do not regret. Thank you for letting your door always open and to share your infinite knowledge with me.

I really enjoyed being involved in the McSAFE project, and doing joint works with KIT and VTT teams. I would like to thank them for that, with a special thank to Diego Ferraro and Manuel Garcia from KIT. Thank you for your kindness and all the discussions we had for the simulations of TMI and Spert. This joint work has taught me a lot. I wish you the best in your future dynamic works!

Je souhaite remercier ma famille qui m'a toujours beaucoup soutenue, en commençant par ma mère. Merci de m'avoir toujours encouragée. Venir faire cette thèse à Paris a été le fruit de beaucoup de réflexions, et je te remercie pour tes encouragements, mais aussi de m'avoir montré que tu me soutiendrais quel que soit le chemin que je choisirais. Mamie et Boumpa, vous m'accompagnez toujours.

J'aimerais remercier un autre homme sage, Ueva. Merci pour ta présence et ton soutien depuis le début, et plus particulièrement pendant ces trois années, où j'ai parfois eu beaucoup le nez dans mon travail. Maururu aussi de m'avoir intégrée dans ton univers aux saveurs chaudes et océanes.

Eva et Sibylle mes amies de toujours, quel bonheur de vous avoir auprès de moi, de partager toutes ces aventures avec vous. Que ce soit en auto-stop fou ou sous la tente, vous prenez décidément bien soin de la petite !

J'ai une grande pensée pour mes amis de prépa, Félix et Gus, sans qui ces deux ans auraient

été bien plus difficiles. Grande pensée aussi pour toutes les amitiés que j'ai construites dans ma ville de coeur, Grenoble. Un big up spécial pour la fine équipe qui s'est retrouvée au CEA : Adriana, Olivier, Benito, Benjamin et Barbara. A nos trêves méridiennes, et aux plus folles soirées vins et fromages !

Je continue mon tour de France avec Toulouse. J'ai passé moins d'un an là-bas, mais les amitiés que j'y ai construites n'en sont pas moins fortes. Je pense à de nombreuses personnes de Sogeti, et en particulier à Fabien, avec qui c'était presque un plaisir de travailler sur DWTTT3D, et à Colin dont l'appétit n'a d'égal que sa générosité, ou bien l'inverse je ne sais pas ! Merci également à la talentueuse Lamia de m'avoir fait rencontrer Bernard, entre autres :-).

Mon expérience au SERMA m'a beaucoup apportée aussi par les belles rencontres que j'y ai faites. Wesley and your unfailing enthusiasm, you have been a real support both for mental and magic fun. Thomas, nos discussions m'ont fait décompresser autant qu'elles ont dérangé nos voisins, c'est dire ! Laura, la vie semble être aventures et générosité avec toi ! Valentin, que ce soit par nos soirées à l'opéra ou au Tango, j'ai adoré la complicité que nous avons créée. Michel et Henri, merci pour votre mentoring et nos troupitudes ! Nicholas, j'ai adoré partager mon bureau avec toi. Merci à l'équipe complète des thésards et à tous les autres avec qui nous avons partagé beaucoup pendant ces trois années. Je m'étonne encore de la chance que nous avons eue de former ce groupe. J'aimerais finalement remercier Stéphane Bourganel qui m'a encadrée pendant mon stage au SERMA en 2013 et a beaucoup contribué à mon envie de revenir !

Merci à tous pour votre présence à ma soutenance, elle m'a profondément touchée. J'espère renouveler sans modération ces moments ensemble !



Contents

Introduction	9
1 Description of the physical mechanisms in a nuclear reactor	19
1.1 Nuclear interaction probability	19
1.1.1 Microscopic cross sections	19
1.1.2 Macroscopic cross sections	20
1.2 Equations for neutron and precursor evolution	20
1.2.1 Transport equation for neutrons coupled with equation for precursors	20
1.2.2 Eigenmode decomposition	21
1.2.3 Point-kinetics via the k -eigenmodes	23
1.3 Fission chains	24
1.3.1 Multiplication factor	25
1.3.2 Neutrons and precursors	25
1.3.3 Fission chain length	26
1.4 Deterministic methods for solving transport equation	27
1.5 Monte Carlo particle transport	28
1.5.1 Principle	28
1.5.2 Variance-reduction and population-control techniques	29
1.5.3 TRIPOLI-4	30
1.6 Coupling between neutron transport, thermal-hydraulics and thermomechanics	30
1.6.1 Description of the feedback effects	30
1.6.2 Thermal-hydraulics solvers	31
1.6.3 Conservation equations	32
1.6.4 Presentation of SUBCHANFLOW	34
I Kinetic Monte Carlo: time-dependent Monte Carlo neutron transport	35
2 Description of kinetic methods for TRIPOLI-4	36
2.1 Challenges	36
2.2 Time dependence	37
2.2.1 Extending the phase space	37
2.2.2 Scoring time grid	37
2.2.3 Simulation time grid	38
2.3 Critical source	38
2.3.1 Sampling the neutrons and precursors	38
2.3.2 Normalization between criticality and kinetic calculations	40
2.3.3 Optimizing the use of criticality cycles	41
2.3.4 Readjustment of the emitted number of neutrons	42
2.4 Precursors forced decay	43

2.5	Population control	45
2.5.1	Russian roulette and splitting	46
2.5.2	Combing	46
2.5.3	Evaluating the efficiency of the two population-control methods	48
2.6	Branchless collisions	54
2.6.1	Description of the algorithm	54
2.6.2	Evaluating the efficiency of the method	54
2.7	Development of a population importance method for fast kinetic configurations	56
2.7.1	Description of the algorithm	56
2.7.2	Evaluating the efficiency of the method	58
2.7.3	Choice of the optimal importance ratio	58
2.8	Development of the capability to handle time-dependent geometry	61
2.9	Conclusion	62
3	Extensive tests of the kinetic methods	63
3.1	Verification tests on SPERT III E-core	63
3.1.1	Presentation of SPERT III E-core	63
3.1.2	Preliminary criticality calculations	64
3.1.3	Steady state	64
3.1.4	Reactivity insertion	67
3.1.5	The role of precursors	68
3.1.6	Rod drop	69
3.2	Verification tests on the TMI 3x3 mini-core	72
3.2.1	The TMI 3x3 mini-core	72
3.2.2	Preliminary criticality calculations	73
3.2.3	Reactivity insertions	73
3.3	Investigating the correlations between time steps	80
3.3.1	Critical configuration	80
3.3.2	Subcritical configuration	81
3.3.3	Fission chains length and impact on correlations	81
3.3.4	Conclusion	82
3.4	Investigating the impact of the scoring mesh spatial discretization	83
3.4.1	Difference between kinetic and criticality calculations	83
3.4.2	Dependence on the time step	84
3.5	Conclusion	86
II	Dynamic Monte Carlo: time-dependent Monte Carlo neutron transport coupled with thermal-hydraulics	87
4	Development of a coupling between TRIPOLI-4 and thermal-hydraulics	88
4.1	Development of a multi-physics interface for TRIPOLI-4	88
4.1.1	Development of a supervisor	88
4.1.2	Exchanging data from the mesh to the geometry	89
4.1.3	Coupling between TRIPOLI-4 and SUBCHANFLOW	90
4.1.4	Presentation of SALOME tools	91
4.2	Criticality calculations with feedback	92
4.2.1	Description	92
4.2.2	Storing source capability	95
4.2.3	Discussion on the choice of the simulation parameters	95

4.3	Transient calculations	95
4.3.1	Description	95
4.3.2	Normalization between criticality and transient calculations	96
4.4	Parallel calculations	97
4.4.1	Thermal-hydraulics “rendez-vous” point	97
4.4.2	Role of the different parallel units	97
4.4.3	Memory footprint	99
4.5	Conclusion	99
5	Verification of the coupling for criticality simulations with feedback	100
5.1	Introduction on the benchmark work	100
5.2	Description of the models used by the different codes	100
5.2.1	TRIPOLI-4 and Serpent 2 models	100
5.2.2	SUBCHANFLOW model	101
5.3	Comparison between TRIPOLI-4 and Serpent 2 without feedback	102
5.4	Description of the coupling schemes	103
5.4.1	Architecture	103
5.4.2	Convergence criteria	103
5.5	Benchmark results with feedback	104
5.5.1	Convergence	104
5.5.2	Coolant temperatures and densities	104
5.5.3	Fuel temperatures	104
5.5.4	Comparison between calculations with and without feedback	107
5.6	Conclusion	110
6	Testing the coupling with dynamic simulations	111
6.1	Preliminary criticality calculations with feedback	111
6.1.1	Critical boron search	111
6.1.2	Source for the dynamic calculations	111
6.2	Steady state	115
6.3	Transients	116
6.3.1	Control-rod extraction	120
6.3.2	Control-rod extraction and reinsertion	128
6.4	Conclusion	132
III	Preliminary analysis of the coupled system stability	133
7	Description of a simple model representative of the coupled system	134
7.1	Description of the problem	134
7.2	The reference model	134
7.2.1	Presentation of the deterministic coupled system	134
7.3	From a deterministic to a stochastic system	136
7.3.1	Adding Brownian noise	136
7.3.2	Reparametrization	137
7.3.3	Limitations of the modeling choices	138
7.4	Preliminary results on Z_t	138
7.4.1	Linear equation	138
7.4.2	Commutation of A_t and A_s	139
7.4.3	Alternative representation	140
7.5	Conclusion	140

8	Analytical and numerical analysis of the model	141
8.1	Analysis of the Brownian term	141
8.1.1	Gaussian process	141
8.1.2	Properties of the matrix \mathcal{A}	141
8.1.3	Analysis of the covariance matrix	142
8.1.4	Numerical simulations of the sytem	145
8.1.5	Numerical simulations with TRIPOLI-4	146
8.1.6	Conclusion	148
8.2	Analysis of the thermal-hydraulics coupling term	149
8.2.1	Preliminary analysis	149
8.2.2	Numerical simulations of the full system	150
8.3	Simplified model	153
8.3.1	Hypothesis	153
8.3.2	Verification of the hypothesis	153
8.3.3	Discussion on the non linearity	153
8.4	Conclusion	155
8.4.1	Brownian term	155
8.4.2	Full system	155
8.4.3	Discussion	155
	Conclusions	156
	Appendix	162
A.1	Useful formulas for the point kinetics	163
A.2	Flattop-Pu	165
A.3	SPERT III E-core	166
A.4	The TMI-1 3x3 mini-core	168
A.5	Mathematical tools	170
A.5.1	Itô process	170
A.5.2	Useful properties on matrices	170
A.5.3	Nomenclature for the analysis of the March-Leuba system	171
A.5.4	Properties on the matrix \mathcal{A}	172
A.5.5	Correlation among time intervals	173
A.6	Extended analysis of the TMI-1 3x3 mini-core dynamic simulations	174
A.6.1	Time evolution of the fuel temperature	174
A.6.2	Comparison between T4/SCF and SSS2/SCF	174
	Résumé en français	177
	References	182

Introduction

The energy produced in nuclear reactors is released by interactions between neutrons and heavy nuclei contained in the fuel. One of the main issues for the study of a reactor behaviour is to model the propagation of the neutrons, described by the Boltzmann transport equation, in the presence of multi-physics phenomena, such as the coupling between neutron transport, thermal-hydraulics and thermomechanics. Pressurized water reactors (PWRs) are designed so as to ensure that the different feedback effects involved are negative: the feedbacks in the fuel and the moderator for example induce a decrease of the neutron flux in case of an initial increase in the reactivity of the neutron population. Operators are bound to prove safety authorities that any operation does not jeopardize the safety and stability of the reactor. For this purpose, design and safety analysis of nuclear reactors are performed with multi-physics simulation tools. Modeling the multi-physics behaviour is in fact highly challenging because of the vast number of unknowns, as well as the large size of the system, which implies simultaneously taking into account nuclear interactions as well as macroscopic fluid motions and mechanical deformations. Most often, simulation tools for multi-physics are built by coupling separate simulation tools for each subfield of physics. This has the advantages of separating the concerns of the development of the different simulation tools and making the simulation more modular.

Concerning the Boltzmann neutron transport equation, two types of strategies are commonly applied. Deterministic methods numerically solve the equation by discretizing the phase space, at the expense of introducing approximations. Stochastic methods, called “Monte Carlo” methods, are based on the random sampling of a large number of neutron trajectories. A mean value and an associated statistical uncertainty are determined for each observable of interest by taking the ensemble averages over the simulated histories. Monte Carlo methods allow for an exact resolution of the transport equation, at the expense of a slow convergence of the statistical uncertainty on the results, which goes as $1/\sqrt{N}$, N being the number of histories. In order to reduce the uncertainty, the most natural solution is to increase the statistics, i.e., the number of simulated trajectories. However, the slow convergence rate makes Monte Carlo a time-consuming method, even if it is well-suited for parallel calculations by nature. The computation time and memory footprint necessary for simulating real-size systems are very large and represent a serious limitation of stochastic methods.

Because of these considerations, Monte Carlo methods are today almost exclusively devoted to criticality and fixed-source calculations, where the system is supposed to be at equilibrium (so that there is no time dependence) and thermal-hydraulics and thermomechanics quantities are supposed to be constant throughout the whole simulation. For such stationary calculations, Monte Carlo methods serve as reference tools for the verification of deterministic methods. Non-stationary scenarios such as transient accidents have been handled by deterministic codes only (Downar et al., 2002; Dulla et al., 2008; D’Auria et al., 2008; Gomez-Torres et al., 2012; Laureau et al., 2015; Knebel et al., 2016), until recent years. In order to extend Monte Carlo methods to non-stationary configurations so as to provide reference results for deterministic

tools for time-dependent problems, two paths must be explored. First, the so-called “kinetic” Monte Carlo shall explicitly take into account the time dependence in neutron transport, including the delayed neutrons. Second, the so-called “dynamic” Monte Carlo shall combine the kinetic methods with the physical feedbacks, such as thermal-hydraulics and thermomechanics.

Thermal-hydraulics concerns hydraulic flows in thermal fluids. This problem is described by non-linear equations, and there are different approaches to its solution. In nuclear reactors, it is customary to consider simplified versions of the problem by introducing different “scales” in relation to the size of the system under analysis: a global scale with system codes, a component scale with sub-channel codes or a local scale with computational fluid dynamics (CFD) codes. CFD codes solve fluid flow problems through turbulence models. They finely model the physical exchanges, at the expense of strong requirements of computation time and memory. This contrasts with sub-channel codes, which solve the equations on a coarser mesh, and provide reliable (although approximate) and fast-running tools for the prediction of fluid flows in steady-state and transient configurations.

The development of reliable and fast numerical tools for the multi-physics simulation of reactor cores (coupling of neutron flux with thermal-hydraulics and thermomechanical feedbacks, in stationary and non-stationary regimes) has undergone intensive research efforts in recent years. This is witnessed by the innovation agendas SNETP, NUGENIA and H2020, and in particular the European projects NURESIM (2005-2008), NURISP (2009-2012), NURESAFE (2013-2015), HPMC (2011-2014) and McSAFE (2017-2020)¹. Similar initiatives have been undertaken in China and in the USA (for instance, the CESAR² project or the CASL³ consortium). The final goal of these efforts is to pave the way towards a full “digital reactor core”, allowing even extreme (i.e., inaccessible to experimental evidence) conditions to be probed and the associated uncertainties to be quantified.

In order to understand the context of the work done in this field, in the following we provide a non-exhaustive list of the coupling efforts that have been conducted between Monte Carlo neutron transport and thermal-hydraulics codes, as well as of the development of kinetic Monte Carlo methods.

One of the first attempts at coupling a Monte Carlo and a thermal-hydraulics code appears to have been performed with the Monte Carlo code MCNP (X-5 Monte Carlo Team, 2003). Several couplings were set up, all in stationary conditions, i.e., without taking into account the time-dependence, and with thermal-hydraulics at different scales. The first used CFD codes, with the very first attempt (Mori et al., 2003) between MCNP4C (Briesmeister, 2000) and SIMMER-III (Yamano et al., 2003). However, because of the computation time limitation, only a one-set coupling was implemented: SIMMER-III was run as a first step, and the resulting temperatures and densities were introduced in the MCNP4C model, without any other thermal-hydraulics update. The first real couplings were later performed (Seker et al., 2007; Cardoni, 2011) between MCNP5 and the CFD codes STAR-CD (CD-adapco, 2005) and STAR-CCM+ (CD-adapco, 2009). Test cases were limited to small systems: up to a 3x3 array of pin cells.

Sub-channel codes have been also considered: an internal coupling between MCNP5 and the thermal-hydraulics sub-channel code COBRA-TF (Avramova and Salko, 2016) was imple-

¹cordis.europa.eu/projects

²cesar.mcs.anl.gov

³www.casl.gov

mented by Sanchez and Al-Hamry (2009). This enabled for stationary coupled calculations for a fuel assembly. In order to further increase the size of the simulated systems, MCNP5 was then coupled to the thermal-hydraulics system code ATHLET (Lerchl and Austregesilo, 1998) at assembly level (Bernnat et al., 2012). The speed of the system code made it possible to perform stationary coupled calculations on a full PWR core, in the context of the PURDUE benchmark (Kozlowski and Downar, 2007). On-the-fly Doppler broadening was later introduced in MCNP to take into account the temperature dependence of the cross sections (Yesilyurt et al., 2012).

Intensive efforts were also made for multi-physics calculations with the Monte Carlo code Serpent 2 (Leppänen et al., 2015). A multi-physics interface was implemented (Leppänen et al., 2012), including internal solvers for the resolution of the heat transfer equation in the fuel and of thermal-hydraulics in the moderator. Serpent 2 can be also coupled with an external thermal-hydraulics solver. The temperature dependence of the cross sections was taken into account with the target motion sampling method (TMS) (Viitanen and Leppänen, 2012, 2014).

Serpent 2 was internally coupled to the sub-channel code SUBCHANFLOW (Imke and Sanchez, 2012), using the multi-physics interface (Daeubler et al., 2015). This coupling was verified by code-to-code comparison on two 3x3 mini cores: against the coupling between the Monte Carlo code TRIPOLI-4[®] (Brun et al., 2015) and SUBCHANFLOW (Sjenitzer et al., 2015), and against the coupling between MCNP5 and SUBCHANFLOW (Ivanov et al., 2013a). The coupling work between Serpent 2 and SUBCHANFLOW also made it possible to perform coupled calculations on the full PWR benchmark mentioned above (Kozlowski and Downar, 2007). All the simulations concerned the stationary state of the reactor.

An external coupling between Serpent 2 and the thermal-hydraulics CFD code OpenFOAM (OpenFOAM Foundation, 2017) was then implemented through the multi-physics interface by Tuominen et al. (2016) using external files. This work provided a new coupling between a Monte Carlo code and thermal-hydraulics at the CFD scale. Stationary calculations were performed on a fuel assembly with 4x4 pins (for this reduced test case, about 2 millions cells were used). However, extending these calculations to larger systems using this coupled tool seems hardly feasible, given the large number of cells required by CFD. Other couplings were recently performed with CFD codes, over small systems such as a pin cell (Wang et al., 2018), and even a TRIGA reactor (Henry et al., 2017).

All these works concerned stationary coupled calculations: up to full cores with sub-channel or system codes, or small systems with CFD codes. For the resolution of transient problems, the coupling infrastructure and the methods need to be adapted. For this purpose, the first step was the investigation of kinetic Monte Carlo methods: a summary of recent developments is given in the following.

Sjenitzer and Hoogenboom (2013) and Hoogenboom and Sjenitzer (2014) were the first to probe kinetic methods in a Monte Carlo code, for both TRIPOLI-4 and MCNP5, by taking into account the precursors, and using critical source sampling. Specific variance-reduction techniques for kinetic Monte Carlo were tested, such as forced precursor decay (Légrády and Hoogenboom, 2008) and branchless collisions. *Russian roulette and splitting* and *combing* (Booth, 1996) were also implemented as population control techniques. Their work enabled the first Monte Carlo kinetic calculations for production neutron transport codes. Due to the high computer cost, kinetic simulations have been performed so far at the scale of fuel assemblies.

Kinetic methods were also implemented in the Monte Carlo code Serpent 2 by Leppänen (2013). As a first step, delayed neutron emission was neglected and only prompt neutrons were considered. The new methods were verified by comparison to MCNP5 calculations on two small systems: Flattop-Pu and STACY-30 benchmarks (OECD Nuclear Energy Agency, 1995). This work was later extended to the treatment of delayed neutrons by Valtavirta et al. (2016).

Mylonakis et al. (2017) developed a transient module with kinetic methods for the Monte Carlo code OpenMC, including the generation of the critical source and the handling of one precursor group. The Russian roulette was also implemented as a population control method. The development of these methods made it possible to perform kinetic simulations of simplified systems.

In order to perform low computational cost kinetic calculations, (Laureau et al., 2015, 2017) has set up a time-dependent version of the fission matrix method. The matrices are computed once with a preliminary Monte Carlo criticality calculation, and are discretized in time. Then, the time evolution of the system is solved using the matrices.

GUARDYAN, a new Monte Carlo code for time-dependent calculations, was recently developed (Molnar et al., 2019), using Graphics Processing Units (GPUs) for accelerated calculations. Kinetic simulations of a whole core transient were performed on the Training Reactor at Budapest University of Technology and Economics, and validated against experimental data. A good agreement was obtained between simulation results and experimental data, showing an attractive application of GPUs for Monte Carlo simulations.

Finally, Sjenitzer et al. (2015) were the first to combine the kinetic methods with thermal-hydraulics. An external coupling was performed between TRIPOLI-4 and the sub-channel code SUBCHANFLOW. This work resulted in the first coupling between a Monte Carlo code and a thermal-hydraulics code in non-stationary conditions. Transient calculations were performed on a mini-core with 3x3 assemblies, and were compared to the results obtained by DYN SUB (Gomez-Torres et al., 2012), the coupling scheme between the deterministic code DYN3D (Grundmann et al., 2005) and the thermal-hydraulics sub-channel code FLICA (Toumi et al., 2000).

All these investigations reveal that, thanks to the growing computer power, it is now feasible to apply Monte Carlo methods to the calculation of non-stationary configurations. However, much progress must still be achieved for the kinetic Monte Carlo methods, which require specific variance-reduction techniques in order to reduce the computational time. Moreover, kinetic methods and the coupling with thermal-hydraulics are most often handled separately. Also, the feedbacks are mostly dealt with by using sub-channel codes. In order to improve the accuracy of thermal-hydraulics modeling, attempts at coupling Monte Carlo neutron transport with CFD codes have been considered; however, further investigations are necessary in order to meet the different challenges listed above: computer time and memory handling.

The work performed in the present Ph. D. thesis provides some advances in the context of the coupled simulations of non-stationary neutron transport with thermal-hydraulics feedbacks. In particular, the goal of this Ph. D. thesis is to develop, verify and test a non-stationary coupling scheme between the Monte Carlo code TRIPOLI-4 and the thermal-hydraulics sub-channel code SUBCHANFLOW, so as to provide a reference tool for the simulation of reactivity-induced

transients in PWRs. The coupling is intended to be generic in scope, in order to simplify future couplings with thermomechanics and thermal-hydraulics codes via integration in the SALOME platform (Bergeaud and Lefebvre, 2010; SALOME, 2019). The stability and the robustness of the proposed algorithms are extensively analysed.

Plan of the thesis

This work is divided in three parts: first, the development of kinetic methods in TRIPOLI-4 is investigated, i.e., solving the time dependent neutron-precursor coupled problem without thermal-hydraulics feedbacks, with special focus on variance-reduction techniques for the time variable. Second, “dynamic” methods are addressed with the combination of kinetic methods with thermal-hydraulics via the development of a coupling scheme between TRIPOLI-4 and SUBCHANFLOW and the investigation of the resulting algorithms. Finally, a preliminary work for the stability analysis of the coupling scheme is presented. The detailed plan of the thesis is the following.

The physical mechanisms involved in a nuclear reactor are briefly recalled in Chapter 1. The basis of nuclear interactions is presented, as well as the two different approaches to solve the neutron transport equation: deterministic and Monte Carlo methods. The intimate coupling between neutron transport, thermal-hydraulics and thermomechanics in PWRs is also introduced.

In Chapter 2, we illustrate the necessary methodology for kinetic simulations in TRIPOLI-4, without taking into account thermal-hydraulics feedbacks. The required algorithms are described, such as the time dependence, the sampling of the source, but also the population-control and variance-reduction techniques, which are necessary for kinetic calculations. A critique of the proposed methods is presented, which is essential to select the most suitable algorithms based on the characteristics of the system. We will focus in particular on the development of a new variance-reduction method carried out during the thesis. This new algorithm is well suited to fast kinetic configurations, where other methods fail to improve the statistics. The description of the method was published in Faucher et al. (2018), and its efficiency was assessed in Faucher et al. (2019a). Another key contribution is also presented: the capacity of TRIPOLI-4 to handle different input geometries within the same simulation, which is essential in order to update the system configuration during the time evolution, such as for the insertion or extraction of the control rods.

In Chapter 3, we will then verify the capacity of TRIPOLI-4 to perform kinetic simulations on realistic systems. Several simulations will be performed on the experimental reactor SPERT III E-core in different configurations: critical, control rod extraction and rod drop. The calculations have been published in (Faucher et al., 2018) mentioned above. Other calculations are presented, on a mini-core based on the TMI-1 (Three Mile Island) reactor, and were published in a verification work through a comparison with the Monte Carlo code Serpent 2 for different reactivity insertions (?). We will also present our preliminary investigation of the correlations between time steps. The dependency of the relative uncertainty on the discretization of the kinetic scoring mesh will be also examined.

The following step is to develop the coupling scheme between TRIPOLI-4 and SUBCHANFLOW: materials and methods are provided in Chapter 4. For this purpose, we have set up a multi-physics interface for TRIPOLI-4 through the development of an application programming interface (API) combined with an external supervisor that is able to control the TRIPOLI-4 simulation. Then, in order to couple TRIPOLI-4 and SUBCHANFLOW, we adhered to the specifications of the SALOME platform, such as the ICoCo API for the coupling interface and the MEDCoupling library for the data exchange between the two codes. The architecture of the coupling scheme was published in Faucher et al. (2019b).

The work described in Chapter 5 aims at verifying the capacity of the coupling scheme to

perform criticality calculations with thermal-hydraulics feedbacks, so as to provide the initial steady-state for coupled transients. As a verification test, we will perform a coupled calculation of an assembly based on the TMI-1 reactor, and we perform a code-to-code comparison with respect to the coupling scheme between Serpent 2 and SUBCHANFLOW. This work was described in the publication (Faucher et al., 2019b) mentioned above.

Then, in Chapter 6, we demonstrate the coupling scheme capability to simulate transients with thermal-hydraulics feedbacks. Calculations will be performed on the mini-core benchmark based on the TMI-1 reactor: first the system will be simulated at steady state so as to verify its stability; then we will introduce reactivity in the system so as to probe the effects of the thermal-hydraulics feedbacks.

Due to the stochastic nature of the outputs produced by TRIPOLI-4, uncertainties are inherent to our coupling scheme and propagate along the coupling iterations. Moreover, thermal-hydraulics equations are non linear, so the prediction of the propagation of the uncertainties is not straightforward. The stability analysis of the coupling scheme is investigated in Chapters 7 (description of a simplified model) and 8 (analysis of the simplified model), in order to assess its convergence. This is a preliminary study aimed at quantifying the uncertainties in dynamic calculations.

List of published material

- Faucher, M., Mancusi, D., Zoia, A., 2018. New kinetic simulation capabilities for TRIPOLI-4[®]: Methods and applications. *Ann. Nucl. Energy* 120, 74 - 88.
- Faucher, M., Mancusi, D., Zoia, A., Ferraro, D., Garcia, M., Imke, U., Leppänen, J., Valtavirta, V., 2019b. Proceedings of the ICAPP 2019 conference. Juan-les-Pins, France.
- Faucher, M., Mancusi, D., Zoia, A., 2019a. Proceedings of the M&C 2019 conference. Portland, Oregon, USA.
- Ferraro, D., Faucher, M., Mancusi, D., Zoia, A., Valtavirta, V., Leppänen, J., Sanchez Espinoza, V., 2019. Proceedings of the M&C 2019 conference. Portland, Oregon, USA.
- Faucher, M., Mancusi, D., Zoia, A., 2020. Proceedings of the PHYSOR 2020 conference. Cambridge, England.

Chapter 1

Description of the physical mechanisms in a nuclear reactor

In this chapter we recall the key features of neutron transport with special emphasis on the time-dependent aspects, which are a key issue for understanding the different challenges encountered in this work. We also detail how nuclear reactor physics, thermal-hydraulics and thermomechanics are intimately coupled.

1.1 Nuclear interaction probability

1.1.1 Microscopic cross sections

Neutron transport is characterized by the different types of nuclear interactions that can take place between an incident neutron and a target nucleus: for example, scattering, capture, or fission (Bell and Glasstone, 1970). All nuclear interactions are described by microscopic cross sections, which carry the dimensions of an area, usually expressed in units of barns ($1 \text{ barn} = 10^{-24} \text{ cm}^2$). This value is roughly proportional to the cross-sectional area of the nucleus (the typical radius of a nucleus is about 10^{-12} cm) and corresponds to the apparent area of the target particle as seen by the incident particle. Partial cross sections are also related to the probability of occurrence of a given interaction: the higher the cross section for a given reaction type, the more likely is the interaction to occur.

Cross sections depend on the nature of the interaction but also on the nature of the incident and target particles, as well as on their energies. Cross sections can be experimentally measured at particular energies. Experimental data, often completed with the help of models, are evaluated and compiled into nuclear data libraries in standard format that can be read by transport codes. Examples of nuclear data libraries are JEFF-3.1 (Santamarina et al., 2009) or ENDF/B-VII (Chadwick et al., 2006).

At each collision, the probability of interaction is determined based on the partial cross section for each type of reaction, evaluated at the energy of the incident particle. For example, when an incident neutron collides with a heavy fissile nucleus, the fission probability is characterized by the ratio between the fission and the total cross sections. Figure 1.1 shows the microscopic fission cross section for uranium-235 (blue line), plotted against the energy of the incident neutron, for a range between 10^{-5} eV and 30 MeV . The dependency on the neutron energy is clearly visible: the fission cross section is much larger at low (thermal) energies. The cross section for uranium-238 is also plotted (orange line): it is smaller than for uranium-235. In PWRs, water is used as moderator, and thermalizes neutrons through multiple collisions. In

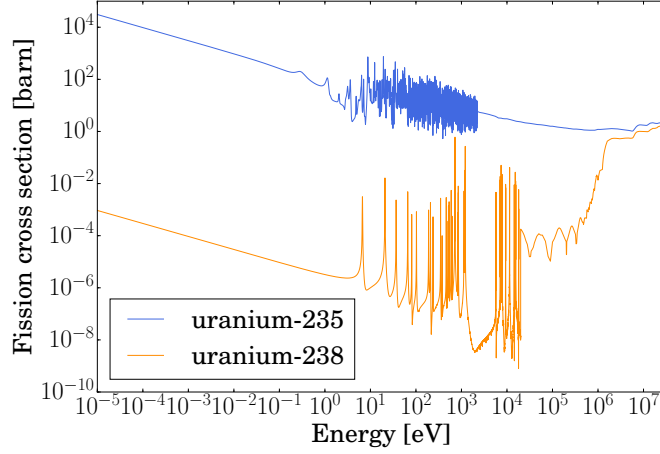


Figure 1.1 – Fission cross section for uranium-235 (blue line) and uranium-238 (orange line), as a function of the energy of the incident neutron. Data come from the JEFF-3.3 library (table MT=18).

order to increase the probability of thermal fissions in PWRs, isotopes with high thermal fission cross sections must be used: this is why natural uranium, mainly consisting of uranium-238, is enriched in uranium-235.

1.1.2 Macroscopic cross sections

Macroscopic cross sections Σ are related to microscopic cross sections σ through the target particle density N [cm^{-3}]:

$$\Sigma = N\sigma, \quad (1.1.1)$$

and are expressed in cm^{-1} . Macroscopic cross sections describe the probability for a neutron to interact with a material per unit length. For a material with k nuclei, the macroscopic cross section is given by

$$\Sigma = N_1\sigma_1 + N_2\sigma_2 + \dots + N_k\sigma_k. \quad (1.1.2)$$

1.2 Equations for neutron and precursor evolution

1.2.1 Transport equation for neutrons coupled with equation for precursors

The angular neutron flux $\varphi(\mathbf{r}, \mathbf{v}, t)$ fully characterizes the system behaviour. It is defined by

$$\varphi(\mathbf{r}, \mathbf{v}, t) = v n(\mathbf{r}, \mathbf{v}, t), \quad (1.2.1)$$

with \mathbf{r} the position vector, \mathbf{v} the velocity, t the time, $v = \mathbf{v} \cdot \boldsymbol{\Omega}$ the neutron speed, $\boldsymbol{\Omega}$ the angular direction vector and n the neutron density.

The evolution of the neutron flux is described by the time-dependent Boltzmann equation, coupled with the equations for the precursors concentrations $c_{i,j}$ (with i the isotope and j its precursor family), which read (Bell and Glasstone, 1970)

$$\frac{1}{v} \frac{\partial}{\partial t} \varphi(\mathbf{r}, \mathbf{v}, t) + L \varphi(\mathbf{r}, \mathbf{v}, t) = F_p \varphi(\mathbf{r}, \mathbf{v}, t) + \sum_{i,j} \chi_d^{i,j}(\mathbf{r}, v) \lambda_{i,j} c_{i,j}(\mathbf{r}, t) + \mathbf{S}(\mathbf{r}, \mathbf{v}, t) \quad (1.2.2)$$

and

$$\frac{\partial}{\partial t} c_{i,j}(\mathbf{r}, t) = \int v_d^{i,j}(v') \Sigma_f^i(\mathbf{r}, v') \varphi(\mathbf{r}, v', t) dv' - \lambda_{i,j} c_{i,j}(\mathbf{r}, t), \quad (1.2.3)$$

with the net disappearance operator

$$L f = \mathbf{\Omega} \cdot \nabla f + \Sigma_t(\mathbf{r}, v) f - \int \Sigma_s(\mathbf{r}, v' \rightarrow v) f(\mathbf{r}, v') dv', \quad (1.2.4)$$

and the prompt fission operator

$$F_p f = \sum_i \chi_p^i(\mathbf{r}, v) \int v_p^i(v') \Sigma_f^i(\mathbf{r}, v') f(\mathbf{r}, v') dv'. \quad (1.2.5)$$

Notation is as follows: Σ_t is the total macroscopic cross section, Σ_s is the differential scattering macroscopic cross section, χ_p^i is the normalized spectrum for prompt fission neutrons of isotope i , v_p^i is the average number of prompt fission neutrons of isotope i , Σ_f is the fission cross section, $\chi_d^{i,j}$ is the normalized spectrum of delayed neutrons emitted from precursor family j of isotope i , $\lambda_{i,j}$ is the decay constant of precursor family j of isotope i , $v_d^{i,j}$ is the average number of delayed fission neutrons of precursor family j of isotope i , and the double sum is extended over all fissile isotopes i and over all precursor families j for each fissile isotope.

The equations above are completed by assigning the proper initial and boundary conditions for φ and $c_{i,j}$. The quantity $\mathcal{S}(\mathbf{r}, v, t)$ represents the contribution due to an external source. We have assumed here that all physical parameters (such as cross sections, velocity spectra, and so on) are time-independent (Keepin, 1965; Akcasu et al., 1971). If N fissile isotopes are present, each associated to M precursors families, Eqs. (1.2.2) and (1.2.3) form a system of $1 + N \times M$ equations to be solved simultaneously. In order to keep notation simple in the following, we will only consider one isotope, so we drop the index i on the isotopes.

1.2.2 Eigenmode decomposition

k -eigenmodes

We consider the system without external source, i.e. $\mathcal{S}(\mathbf{r}, v, t) = 0$. By replacing the precursor Eq. (1.2.3) into Eq. (1.2.2) for the neutron flux, we get

$$\begin{aligned} \frac{1}{v} \frac{\partial}{\partial t} \varphi(\mathbf{r}, v, t) + L \varphi(\mathbf{r}, v, t) = F_p \varphi(\mathbf{r}, v, t) + \sum_j \chi_d^j(\mathbf{r}, v) \int v_d^j(v') \Sigma_f(\mathbf{r}, v') \varphi(\mathbf{r}, v', t) dv' \\ - \sum_j \chi_d^j(\mathbf{r}, v) \frac{\partial}{\partial t} c_j(\mathbf{r}, t). \end{aligned} \quad (1.2.6)$$

Now if we seek a stationary solution of Eq. (1.2.6), we get

$$L \varphi(\mathbf{r}, v) = F_p \varphi(\mathbf{r}, v) + \sum_j \chi_d^j(\mathbf{r}, v) \int v_d^j(v') \Sigma_f(\mathbf{r}, v') \varphi(\mathbf{r}, v') dv'. \quad (1.2.7)$$

By defining the total fission operator $F = F_p + F_d$, including the prompt fission operator, as defined by Eq. (1.2.5), and the delayed fission operator

$$F_d f = \sum_j \chi_d^j(\mathbf{r}, v) \int v_d^j(v') \Sigma_f(\mathbf{r}, v') f(\mathbf{r}, v') dv', \quad (1.2.8)$$

Eq. (1.2.7) can be rewritten as

$$L\varphi(\mathbf{r}, \mathbf{v}) = F\varphi. \quad (1.2.9)$$

The k -eigenmodes φ_k associated to the Boltzmann equation Eq. (1.2.2) emerge by imposing that the system should be exactly critical without external sources and asking by which factor k the fission terms should be rescaled in order to make the time derivative vanish (Bell and Glasstone, 1970; Cullen et al., 2003):

$$L\varphi_k(\mathbf{r}, \mathbf{v}) = \frac{1}{k} F\varphi_k(\mathbf{r}, \mathbf{v}). \quad (1.2.10)$$

α -eigenmodes

For bounded domains, using the separation of variables, an exponential relaxation of the kind

$$\varphi(\mathbf{r}, \mathbf{v}, t) = \varphi_\alpha(\mathbf{r}, \mathbf{v})e^{\alpha t} \quad (1.2.11)$$

and

$$c_j(\mathbf{r}, t) = c_\alpha^j(\mathbf{r})e^{\alpha t} \quad (1.2.12)$$

may be postulated for both the neutron flux and the precursors concentrations. Here the value α represents the relaxation frequency (Bell and Glasstone, 1970; Duderstadt and Martin, 1979). Eqs. (1.2.11) and (1.2.12) can be more rigorously justified by resorting to Laplace transform or equivalently to spectral analysis (Duderstadt and Martin, 1979). Yet, proving the feasibility of such a relaxation is highly non-trivial in general, and precise (although not very restrictive) conditions are required on the geometry of the domain and on the material cross sections (Bell and Glasstone, 1970; Larsen and Zweifel, 1974; Duderstadt and Martin, 1979). Here, for the sake of simplicity, we will assume that such conditions are met (which is typically the case for almost all systems of practical interest) and that separation of variables is allowed.

Then, substituting Eqs. (1.2.11) and (1.2.12) into Eqs. (1.2.2) and (1.2.3), respectively, yields the (coupled) natural eigenmode equations

$$\frac{\alpha}{v}\varphi_\alpha(\mathbf{r}, \mathbf{v}) + L\varphi_\alpha(\mathbf{r}, \mathbf{v}) = F_p\varphi_\alpha(\mathbf{r}, \mathbf{v}) + \sum_j \chi_d^j(\mathbf{r}, v)\lambda_j c_\alpha^j(\mathbf{r}) \quad (1.2.13)$$

and

$$\alpha c_\alpha^j(\mathbf{r}) = \int v_d^j(v')\Sigma_f(\mathbf{r}, v')\varphi_\alpha(\mathbf{r}, v')d\mathbf{v}' - \lambda_j c_\alpha^j(\mathbf{r}), \quad (1.2.14)$$

which formally represent a system of eigenvalue equations for the flux φ_α and the precursors c_α^j , the eigenvalues being α .

It is customary to formally solve Eq. (1.2.14) for the precursor concentration and to replace the resulting c_α^j into Eq. (1.2.13). This yields the eigenvalue problem for φ_α (Weinberg, 1952; Cohen, 1958; Henry, 1964; Bell and Glasstone, 1970)

$$\frac{\alpha}{v}\varphi_\alpha(\mathbf{r}, \mathbf{v}) + L\varphi_\alpha(\mathbf{r}, \mathbf{v}) = F_p\varphi_\alpha(\mathbf{r}, \mathbf{v}) + \sum_j \frac{\lambda_j}{\lambda_j + \alpha} F_d^j\varphi_\alpha(\mathbf{r}, \mathbf{v}). \quad (1.2.15)$$

The inverse of the largest eigenvalue α_0 for this equation is commonly referred to as the “reactor period” (Cohen, 1958; Henry, 1964; Kaper, 1967; Bell and Glasstone, 1970). It provides information on the asymptotic time behaviour of the system in any regime. For an example of application of the dominant natural eigenvalue in reactor physics, see, e.g., Zoia et al. (2014a, 2015).

Observe that, if $\alpha = 0$ is an admissible dominant eigenvalue of Eq. (1.2.15), then Eq. (1.2.10) admits a solution with $k = 1$ and $\varphi_\alpha = \varphi_k$. Contrary to the k -eigenmodes Eq. (1.2.10), the natural eigenmode Eqs. (1.2.13) and (1.2.14) are well defined also for non-multiplying systems, in which case we have

$$\frac{\alpha}{v} \varphi_\alpha(\mathbf{r}, \mathbf{v}) + L \varphi_\alpha(\mathbf{r}, \mathbf{v}) = 0. \quad (1.2.16)$$

1.2.3 Point-kinetics via the k -eigenmodes

The so-called point-kinetics equations are introduced in reactor physics so as to condense the behaviour of the neutron and precursor population to a model where only the time dependence is left (Bell and Glasstone, 1970). The equations are formally obtained from the exact time-dependent Boltzmann equation weighted by the adjoint flux. The equation adjoint to Eq. (1.2.10) is

$$L^\dagger \varphi_k^\dagger(\mathbf{r}, \mathbf{v}) = \frac{1}{k} F^\dagger \varphi_k^\dagger(\mathbf{r}, \mathbf{v}). \quad (1.2.17)$$

where φ_k^\dagger denotes the adjoint critical eigenmode for the neutron flux, and the adjoint of a linear operator \mathcal{L} is defined as customary via the relation

$$\langle \mathcal{L}f, g \rangle = \langle f, \mathcal{L}^\dagger g \rangle, \quad (1.2.18)$$

with $\langle f, g \rangle$ the scalar product between functions f and g .

The first step consists in multiplying the time-dependent Eq. (1.2.2) by the fundamental adjoint k eigenmode and by multiplying the equation for the adjoint k eigenmode by the time-dependent neutron flux $\varphi(\mathbf{r}, \mathbf{v}, t)$. The resulting equations are then integrated over space, energy and angle, which yields

$$\begin{aligned} \frac{\partial}{\partial t} \langle \varphi_k^\dagger(\mathbf{r}, \mathbf{v}), \frac{1}{v} \varphi(\mathbf{r}, \mathbf{v}, t) \rangle + \langle \varphi_k^\dagger(\mathbf{r}, \mathbf{v}), L \varphi(\mathbf{r}, \mathbf{v}, t) \rangle \\ = \langle \varphi_k^\dagger(\mathbf{r}, \mathbf{v}), F_p \varphi(\mathbf{r}, \mathbf{v}, t) \rangle + \sum_j \lambda_j \langle \varphi_k^\dagger(\mathbf{r}, \mathbf{v}), \chi_d^j(\mathbf{r}, \mathbf{v}) c_j(\mathbf{r}, t) \rangle \end{aligned} \quad (1.2.19)$$

and

$$\langle \varphi(\mathbf{r}, \mathbf{v}, t), L^\dagger \varphi_k^\dagger(\mathbf{r}, \mathbf{v}) \rangle = \frac{1}{k} \langle \varphi(\mathbf{r}, \mathbf{v}, t), F^\dagger \varphi_k^\dagger(\mathbf{r}, \mathbf{v}) \rangle. \quad (1.2.20)$$

By subtracting these equations from each other and using Eq. (1.2.18), we get

$$\frac{\partial}{\partial t} \langle \varphi_k^\dagger, \frac{1}{v} \varphi \rangle + \frac{1}{k} \langle \varphi_k^\dagger, F \varphi \rangle = \langle \varphi_k^\dagger, F_p \varphi \rangle + \sum_j \lambda_j \langle \varphi_k^\dagger, \chi_d^j c_j \rangle, \quad (1.2.21)$$

which we can rewrite as

$$\frac{\partial}{\partial t} \langle \varphi_k^\dagger, \frac{1}{v} \varphi \rangle = \frac{k-1}{k} \langle \varphi_k^\dagger, F \varphi \rangle - \langle \varphi_k^\dagger, F_d \varphi \rangle + \sum_j \lambda_j \langle \varphi_k^\dagger, \chi_d^j c_j \rangle. \quad (1.2.22)$$

Now, suppose that the time-dependent neutron flux can be factorized as $\varphi(\mathbf{r}, \mathbf{v}, t) \simeq \tilde{n}(t) \varphi_k(\mathbf{r}, \mathbf{v})$, i.e., the product of the fundamental k eigenmode (acting as a position- and velocity-dependent shape factor) times an amplitude factor $\tilde{n}(t)$ that depends only on time. This approximation is assumed to be valid for small departures from the critical point, i.e., when $k \simeq 1$. By replacing this flux decomposition into Eq. (1.2.22) we obtain an equation for the neutron population amplitude $\tilde{n}(t)$, namely,

$$\frac{\partial}{\partial t} \tilde{n}(t) = \frac{\rho - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} \tilde{n}(t) + \sum_j \lambda_j \tilde{c}_j(t), \quad (1.2.23)$$

where we have defined the so-called static reactivity

$$\rho = \frac{k - 1}{k}, \quad (1.2.24)$$

the effective delayed neutron fraction

$$\beta_{\text{eff}} = \frac{\langle \varphi_k^\dagger, F_d \varphi_k \rangle}{\langle \varphi_k^\dagger, F \varphi_k \rangle}, \quad (1.2.25)$$

the effective mean generation time

$$\Lambda_{\text{eff}} = \frac{\langle \varphi_k^\dagger, \frac{1}{v} \varphi_k \rangle}{\langle \varphi_k^\dagger, F \varphi_k \rangle}, \quad (1.2.26)$$

and the effective precursor concentrations

$$\tilde{c}_j(t) = \frac{\langle \varphi_k^\dagger, \mathcal{X}_d^j c_j \rangle}{\langle \varphi_k^\dagger, \frac{1}{v} \varphi_k \rangle}. \quad (1.2.27)$$

The kinetics parameters β_{eff} and Λ_{eff} appearing in Eq. (1.2.23) are called “effective” because they have been weighted by the adjoint eigenmode, which physically represents the asymptotic importance of the neutron population in a multiplying system (Keepin, 1965; Bell and Glasstone, 1970).

Eq. (1.2.23) yields the time evolution of the neutron population, under the approximations introduced above. The neutron evolution must be coupled to the equations for the effective precursor concentrations, which can be derived by following the same arguments as above, and read

$$\frac{\partial}{\partial t} \tilde{c}_j(t) = \frac{\beta_{j,\text{eff}}}{\Lambda_{\text{eff}}} \tilde{n}(t) - \lambda_j \tilde{c}_j(t), \quad (1.2.28)$$

where we have defined the effective delayed neutron fractions pertaining to each family j , i.e.,

$$\beta_{j,\text{eff}} = \frac{\langle \varphi_k^\dagger, F_d^j \varphi_k \rangle}{\langle \varphi_k^\dagger, F \varphi_k \rangle}. \quad (1.2.29)$$

Other useful formulas for point-kinetics analysis are provided in Sec. A.1.

1.3 Fission chains

After a fission reaction with the incident neutron, the fissile nucleus typically splits in two lighter nuclei, called fission products. Other particles are also emitted, including new neutrons, which may trigger new fissions. This branching process, where neutrons give birth to other neutrons, is called a “fission chain”, the control of which is essential for the operation of a reactor. A fission chain ends when there are no more neutrons to induce new fissions, i.e., when all neutrons from the fission chain got captured or leaked out of the system.

1.3.1 Multiplication factor

To define the sustainability of a fission chain, the effective neutron multiplication factor k_{eff} is defined as the average ratio between the number of neutrons in successive generations. When each fission causes one fission on average, the effective neutron multiplication factor k_{eff} is equal to 1 and the system is called “critical” (Keepin, 1965). In this case, the reaction is self-sustaining. Such stable chain is the basis for energy production in a nuclear reactor. If k_{eff} is larger than 1, the system is “supercritical”; in the absence of feedback effects, there is an exponential growth of the neutron population. Conversely, if it is smaller than 1, the system is “subcritical” and the population eventually dies out.

Another quantity commonly used to characterize the sustainability of the fission chains is called the “reactivity”, which is defined as

$$\rho = \frac{k_{\text{eff}} - 1}{k_{\text{eff}}}. \quad (1.3.1)$$

It is usually measured in pcm (1 pcm = 10^{-5}), or in dollars (1 \$ = β_{eff} , with β_{eff} the effective delayed neutron fraction defined by Eq. (1.2.25)). The reactivity expresses the deviation from the system with respect to criticality.

1.3.2 Neutrons and precursors

Two types of neutrons are released by fission, as illustrated in Fig. 1.2: prompt neutrons, emitted almost instantaneously after fission, and delayed neutrons, coming from the β^- decay of unstable fission products, called “precursors” (Keepin, 1965). Precursors are characterized by their decay constant λ [s^{-1}], i.e., the average decay time for a fission product to give rise to a delayed neutron by β^- decay. In nuclear data libraries, the different precursors are regrouped in families depending on their decay time (e.g., there are 8 families in the JEFF-3.1 library (Santamarina et al., 2009)).

Prompt and delayed neutrons have very different characteristics, as detailed below and summarized in Table 1.1. Here, λ is the typical decay constant of the precursors (each precursor family j has its own decay constant λ_j). All the parameters depend on the fuel composition and the configuration of the reactor core.

The prompt multiplication factor k_p and the delayed multiplication factor k_d can be defined as

$$k_p = (1 - \beta_{\text{eff}}) \times k_{\text{eff}}, \quad (1.3.2)$$

$$k_d = \beta_{\text{eff}} \times k_{\text{eff}}, \quad (1.3.3)$$

$$k_{\text{eff}} = k_p + k_d. \quad (1.3.4)$$

Energy

Prompt neutrons are emitted with an average kinetic energy around 2 MeV. Yet, as mentioned in Sec. 1.1, fission is more likely to happen when the energy of the incident neutron falls below 1 eV. Because of their large emission energy, prompt neutrons have to slow down before inducing fission. In contrast, delayed neutrons are created with a lower energy (around 400 keV),

	prompt neutrons	delayed neutrons
emission energy	2 MeV	400 keV
associated time scale	$\Lambda_{\text{eff}} \approx 20 \mu\text{s}$	$\lambda^{-1} \approx 1 \text{ s}$
average number per fission	$\nu_p \approx 2.4$	$\nu_d \approx 0.017$

Table 1.1 – Different characteristics of prompt and delayed neutrons (typical values for uranium-235 are given, with $\beta_{\text{eff}} \approx 700 \text{ pcm}$).

and therefore are less likely to disappear by leakage or capture, before inducing fission. Essentially, delayed neutrons have a higher probability to induce a thermal fission (Duderstadt and Hamilton, 1976).

The delay of delayed neutrons

Neutrons and precursors have very different typical time scales; the generation time for prompt neutrons is about $\Lambda_{\text{eff}} \approx 20 \mu\text{s}$, while the precursors decay time is about $\lambda^{-1} \approx 10 \text{ s}$ (these are typical values for a PWR). Because of this difference of scale, delayed neutrons slow down the time evolution of the system due to a change in reactivity. This is the reason why delayed neutrons play a major role in nuclear reactor control.

In a critical system ($k_{\text{eff}} = 1$), there are $(1 - \beta_{\text{eff}})$ prompt neutrons and β_{eff} delayed neutrons. Thus, the neutron effective lifetime l_{eff} , combining prompt and delayed neutrons lifetime, is defined as (Duderstadt and Hamilton, 1976)

$$l_{\text{eff}} = (1 - \beta_{\text{eff}}) \times \Lambda_{\text{eff}} + \sum_j \beta_{\text{eff},j} \left(\frac{1}{\lambda_j} + \Lambda_{\text{eff}} \right) \quad (1.3.5)$$

$$= \Lambda_{\text{eff}} + \sum_j \frac{\beta_{\text{eff},j}}{\lambda_j}. \quad (1.3.6)$$

Unbalanced ratio

The quantities $\beta_{\text{eff}}/\Lambda_{\text{eff}}$ and λ provide the rates at which a neutron is converted into a precursor, and conversely. The typical ratio $\beta_{\text{eff}}/(\lambda \times \Lambda_{\text{eff}})$ is of the order of 10^4 for water-moderated reactors. This implies that, when the neutron and precursor populations are in equilibrium, precursors are considerably more abundant than neutrons within the core. This suggests that the Monte Carlo simulation of such unbalanced populations (in terms of size and time scale) requires strategies and variance-reduction techniques that are distinct from those of stationary simulations.

1.3.3 Fission chain length

When $k_p < 1$, prompt fission chains are finite and eventually die out. Criticality is ensured by the presence of delayed neutrons. Indeed, each fission chain produces on average one precursor, which emits a delayed neutron after a decay time larger than the chain lifetime, thus after the prompt fission chain has died. Therefore, the delayed neutron does not participate in the fission chain that created it but starts a new one instead. In such cases ($k_p < 1$), the average length \bar{n} of

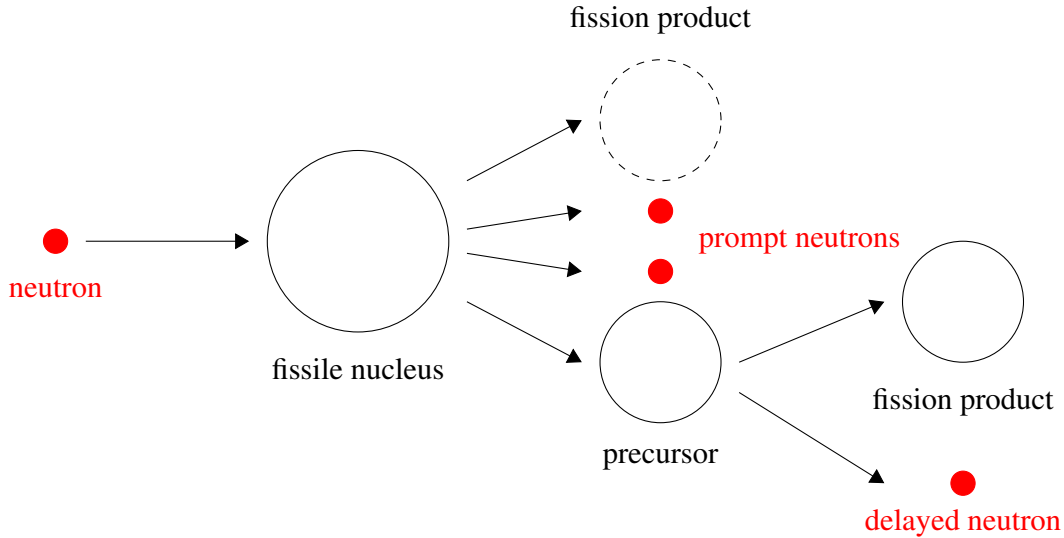


Figure 1.2 – Schematic depiction of neutron-induced fission with the different particles produced. After the fission, prompt neutrons and precursors are emitted almost instantaneously while delayed neutrons are emitted after precursors decay.

the fission chain is given by (Sjenitzer and Hoogenboom, 2011a):

$$\bar{n} = \frac{1}{1 - k_p}. \quad (1.3.7)$$

For instance, if the system is exactly critical, $k_p = 1 - \beta_{\text{eff}}$, and the average fission chain length is $\bar{n} = 1/\beta_{\text{eff}}$.

When the system is prompt critical or supercritical (i.e., $k_p \geq 1$), however, prompt neutrons alone are sufficient to maintain the chains, and some chains grow indefinitely. Some neutrons might get prematurely captured or leak, thus terminating the chain, but on average the fission chains are infinite.

1.4 Deterministic methods for solving transport equation

Deterministic methods solve the Boltzmann transport equation by discretizing the phase space: space, energy, time and angle (for instance, S_N for the decomposition on discrete directions, P_N for an expansion on the spherical harmonics and SP_N , simplified P_N). The advantage of such methods is a reasonable computation time, obtained at the expense of discretization errors.

Until very recently, the simulation of neutron transport in non-stationary conditions was entirely based on deterministic methods (which are usually fast for stationary conditions). For transient regimes, due to the very large number of unknowns ($\sim 10^{14}$) resulting from a fine discretization of phase space variables (space, angle, velocity and time), current state-of-the-art industrial codes employ a two-step approach: a detailed transport calculation at the lattice scale in stationary conditions in two dimensions is followed by a time evolution calculation for the neutron flux at the core scale, based on the cross sections determined in the course of the first step. The time-dependent step is typically carried out in simplified transport models (diffusion or SP_N , for instance) with a coarse energy discretization (Smith, 1979; D’Auria et al., 2004; Dulla et al., 2008; Larsen, 2011; Zhu, 2015).

Since the approximations introduced in the deterministic approach are problem-dependent (i.e., specific to each reactor type), the validity of the results thus obtained, as well as the assessment of the associated uncertainties, depend on the configuration under analysis (D’Auria et al., 2004; Dulla et al., 2008; Larsen, 2011). Thus, in order to relax these constraints and to consolidate the validation of deterministic codes, it is mandatory to develop a *best estimate* method (IAEA, 2003). This is especially true in view of the small number of experimental measurements available for transient reactor operation or accidents (IAEA, 2015).

1.5 Monte Carlo particle transport

1.5.1 Principle

Monte Carlo simulation is based on the realization of a large number of stochastic neutron trajectories, whose probability laws are determined in agreement with the underlying physical properties (the probability of particle-matter interaction, energy and angle distributions after collision, and so on). The transport simulation follows a random walk from one interaction to the next. The distance to the next collision is sampled according to the total macroscopic cross section; the particle is transported to this point and finally the interaction is sampled depending on the nucleus at the collision site. Monte Carlo methods allow for an exact treatment of the reactor geometry (Lux and Koblinger, 1991). Accordingly, Monte Carlo simulation is considered as the “golden standard” for neutron transport calculations (Bell and Glasstone, 1970; Lux and Koblinger, 1991).

Consider the time-independent version of Eq. (1.2.2)

$$L\varphi(\mathbf{r}, \mathbf{v}) = F_p \varphi(\mathbf{r}, \mathbf{v}) + \sum_j \chi_d^j(\mathbf{r}, \mathbf{v}) \lambda_j c_j(\mathbf{r}) + \mathcal{S}(\mathbf{r}, \mathbf{v}), \quad (1.5.1)$$

where

$$c_j(\mathbf{r}) = \frac{1}{\lambda_j} \int v_d^j(v') \Sigma_f^i(\mathbf{r}, v') \varphi(\mathbf{r}, v') dv'. \quad (1.5.2)$$

This equation describes a so-called “fixed-source” problem; in the following we will show how the problem is addressed with a Monte Carlo simulation. A source and a detector must be defined in the phase space. The purpose of the simulation is to estimate the response in the detector, meaning collecting the contributions of neutrons reaching a given phase-space region. In practice, N neutrons are emitted from the source and are transported through the phase space. Assuming that the simulated system is made of a homogeneous medium characterized by a total macroscopic cross section Σ_t , the exponentially distributed distance x between two interactions, travelled by a neutron with an energy E , is sampled from the equation

$$x = -\frac{1}{\Sigma_t(E)} \ln(1 - \xi), \quad (1.5.3)$$

with ξ a random number, uniformly distributed in the interval $[0,1]$. Note that, if the space is not homogeneous, the flight has to be split in different volumes, and, if x is larger than the distance to the boundary of the particle initial volume, the particle is first stopped at the volume boundary and the next flight length is then sampled.

At the new sampled position, the interacting nucleus k is chosen with probability

$$p_k = \frac{\Sigma_{k,t}(E)}{\Sigma_t(E)}, \quad (1.5.4)$$

with $\Sigma_{k,t}(E)$ the total macroscopic cross section for nucleus k .

Finally, the interaction l on nucleus k is sampled using the probability

$$p_{k,l} = \frac{\sigma_{k,l}(E)}{\sigma_{k,t}(E)}, \quad (1.5.5)$$

with $\sigma_{k,l}(E)$ the microscopic cross section for nucleus k associated to the interaction l .

After the interaction, if the neutron is still alive (it can be absorbed, or killed by the Russian roulette presented in Sec. 1.5.2), the distance to the next collision is sampled, and so on, until the particle is killed or the simulation stops. Neutron contributions are collected if they reach the detector. The whole process of transporting N particles is repeated M times (i.e., the statistical ensemble) with a different random seed each time.

The contributions being averaged over M independent replicas, the result comes with a statistical uncertainty on the ensemble average, which is inversely proportional to square root of the number of histories. Thus, the only reason for fluctuations in the results is the limited number of simulated trajectories. In order to accumulate significant statistics, Monte Carlo codes must simulate a large amount of particles. The order of magnitude is problem-dependent but it is usually very large. Hence, Monte Carlo simulations are very time-consuming. Fortunately, Monte Carlo particle transport codes generally have excellent parallel scalability, and are even sometimes “embarrassingly” parallel (Rosenthal, 1999). Indeed, since neutron histories are independent, each processor can follow its own set of particles. Parallel simulation allows for a considerable speed-up, in principle of the order of the number of available processors. When all processors have completed the simulation of particle histories, the final results are collected.

1.5.2 Variance-reduction and population-control techniques

The convergence of Monte Carlo simulations directly depends on the number of simulated particles, which also governs the computation time. In order to improve the convergence without slowing down the computation time, physical laws are not systematically enforced; it is possible to alter the physical processes as long as a compensation is applied elsewhere to keep the results unbiased. For this purpose, a so-called “statistical weight” is assigned to each particle at the beginning of the simulation, and evolves along the simulation so as to compensate the introduced changes in the sampling rules. However, the “variance-reduction” techniques require to control the statistical weights and size of the population in order to ensure that they do not vary too much. Such Monte Carlo simulations are called “non-analog”, as opposed to those preserving the regular sampling laws, which are called “analog”. The most representative examples of a variance-reduction technique, *implicit capture*, and of a population-control technique, *Russian roulette and splitting*, are presented below.

Implicit capture

In the absence of fission, a neutron having a collision can be either scattered or captured. In the second case, it is “killed”, meaning it is removed from the simulation. The probability p_a for a neutron having a collision to be absorbed is

$$p_a = \frac{\sigma_a}{\sigma_t}, \quad (1.5.6)$$

and the complementary probability p_s not to be absorbed (i.e., to be scattered) is

$$p_s = 1 - \frac{\sigma_a}{\sigma_t}, \quad (1.5.7)$$

with σ_a the microscopic absorption cross section and σ_t the microscopic total cross section. Implicit capture allows the particles to explore the phase space rather than being absorbed. In other words, particles are forced to scatter. In order to preserve a fair Monte Carlo game, the particle weights are multiplied by a factor $p_s < 1$. This way, the neutron weight decreases at each collision.

Russian roulette and splitting

At some point in the simulation, particle weights may become very low (for instance because of implicit capture). In that case, particles slow down the calculation without contributing much to the statistics. With implicit capture preventing particles to be absorbed, only leakage can kill particles. Thus, implicit capture has to be combined with other methods. The Russian roulette method helps to terminate particle histories. If the particle weight drops below a predefined threshold (usually 0.8), then a random number is generated. If it is below the initial weight, the particle is killed. Otherwise, its weight is set to a predefined value (usually 1). On the contrary, if the particle weight is above another predefined threshold (usually 2), it is split into several particles, each carrying a fraction of the original weight.

1.5.3 TRIPOLI-4

TRIPOLI-4 is a 3D continuous-energy Monte Carlo particle transport code devoted to shielding, reactor physics, criticality-safety and nuclear instrumentation. It can solve both fixed-source transport and eigenvalue problems. The code has been developed at CEA Saclay since the mid 90s in C++, with a few parts in C and Fortran. It uses nuclear data evaluation files written in ENDF format. For the temperature dependence of the cross sections, stochastic interpolation was implemented. TRIPOLI-4 supports execution in parallel mode. More details on this code can be found in Brun et al. (2015). The work presented in this thesis was performed within a development version of TRIPOLI-4.

1.6 Coupling between neutron transport, thermal-hydraulics and thermomechanics

A nuclear reactor is a complex system whose behaviour depends on the strong coupling between neutron transport, thermal-hydraulics and thermomechanics. The feedback effects involved are essential to the reactor stability. Indeed, a PWR is designed so as to ensure that the feedback effects are negative: any increase or decrease in the neutron power causes thermal-hydraulics and thermomechanics feedbacks that counter-react these variations and thus keep the reactor power stable. The main feedback effects are listed below.

1.6.1 Description of the feedback effects

As discussed above, microscopic cross sections strongly depend on the energy of the incident neutron. As a consequence, microscopic cross sections are also temperature-dependent, because of thermal motion of the collided nuclei, mainly via the Doppler effect. Moreover, macroscopic cross sections depend on the temperature also via the density effect, which affects the concentrations of the nuclei per unit volume.

Fuel temperature

When the fission power increases, the fuel temperature also increases, inducing modifications of cross sections. The most visible effect is the so-called “Doppler broadening” of the resonances in

the radiative capture cross section of uranium-238. The thermal motion of target nuclei changes the shape of the resonances: they become wider and flatter. With Doppler effect, neutrons are more likely to be captured on uranium-238 and less likely to induce fission on uranium-235, which leads to a decrease in fission power, thereby stabilizing the system. In a typical PWR, the Doppler coefficient α_T^F , is defined as

$$\alpha_T^F = \frac{1}{k_{\text{eff}}} \frac{dk}{dT_F}, \quad (1.6.1)$$

with T_F the fuel temperature, and ranges from -1 pcm/K to -4 pcm/K (Duderstadt and Hamilton, 1976).

Moderator temperature and density

When the moderator temperature increases, neutrons are less slowed down, and the neutron spectrum is hardened. This results in a reduction of absorption in uranium-235 as compared to absorption in uranium-238. Therefore, the power decreases.

The main moderator effect occurs from changes in the density. When the moderator density decreases, the macroscopic cross section decreases and the moderator becomes less efficient to slow down the neutrons. Therefore, neutrons are less efficient at inducing fission, the power decreases and the reactivity decreases. The two feedback effects in the moderator improve reactor stability. The moderator coefficient α_T^M , is defined as

$$\alpha_T^M = \frac{1}{k_{\text{eff}}} \frac{dk}{dT_M}, \quad (1.6.2)$$

with T_M the moderator temperature, and ranges from -8 pcm/K to -50 pcm/K (Duderstadt and Hamilton, 1976).

Thermomechanics

The heating in the reactor core induces thermal expansion and deformation of the different elements. For example, the fuel pellets expand at the expense of the gas gap, whose width decreases. As a result, the heat transfer between the fuel pellet and the cladding increases.

1.6.2 Thermal-hydraulics solvers

Fluid dynamics is described by non-linear conservation equations, and different approaches exist for their solution. For example, CFD codes finely model the physical exchanges, at the expense of strong requirements in terms of computation time and memory. Sub-channel codes are faster but make some approximations and solve the equations on a coarser mesh. Even if they provide approximate solutions, they are reliable tools and are often used to study thermal-hydraulics phenomena in nuclear reactor cores. System codes rely on an even less detailed description of the physics and provide the response of the components under consideration in the form of global averages. In the following, we focus on the sub-channel approach, which is used in this work.

A sub-channel is the flow area delimited by adjacent fuel rods. Sub-channel codes only consider two directions of the flow through sub-channels: axial and lateral (lateral covers all directions orthogonal to axial direction). The axial length of each sub-channel is divided in slices, and the flow is transmitted through these axial volumes. The axial flow is usually treated

as the dominant one-dimensional flow, and the lateral flow is simplified. Lateral flow is assumed to enter a sub-channel volume through “gaps”, formed by adjacent fuel rods. Examples of sub-channel codes are COBRA-TF (Avramova and Salko, 2016), FLICA (Toumi et al., 2000) and SUBCHANFLOW (Imke and Sanchez, 2012).

1.6.3 Conservation equations

Fluid dynamics is based on three fundamental equations, which are conservation equations for mass

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{V}) = 0, \quad (1.6.3)$$

momentum

$$\frac{\partial(\rho \vec{V})}{\partial t} + \nabla \cdot (\vec{V} \otimes \rho \vec{V}) - \nabla \cdot \tau + \nabla P = \rho \vec{g}, \quad (1.6.4)$$

and energy

$$\frac{\partial \rho}{\partial t} \left(\rho \left(e + \frac{|\vec{V}|^2}{2} \right) \right) + \nabla \cdot \left(\rho \left(e + \frac{\vec{V}^2}{2} \right) \vec{V} \right) - \nabla \cdot \vec{\varphi} = \nabla \cdot (\tau \cdot \vec{V}) + \rho \vec{g} \cdot \vec{V} - \nabla \cdot (P \vec{V}), \quad (1.6.5)$$

with ρ the density, \vec{V} the field velocity, \otimes the outer product, τ the stress vector, P the pressure, \vec{g} the gravity acceleration vector, e the internal energy and $\vec{\varphi}$ the conductive heat flux vector.

Because of the limitation of the flow to two directions for sub-channel solvers, approximations are introduced in the conservation equations. For instance, for mass conservation, Eq. (1.6.3) becomes

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho V_x)}{\partial x} + \frac{1}{A} w_k = 0, \quad (1.6.6)$$

with V_x the axial speed, A the sub-channel flow area and w_k the linear mass flow rate ($\text{kg m}^{-1} \text{s}^{-1}$) through the k -th gap.

Then, mass, momentum and energy conservation equations are discretized over the cells of the mesh with a finite difference scheme. The resulting equations simultaneously solved by SUBCHANFLOW in each cell are presented below (Imke and Sanchez, 2012).

Mass conservation

$$A_{i,j} \frac{\Delta X_j}{\Delta t} (\rho_{i,j} - \rho_{i,j}^{\text{old}}) + (m_{i,j} - m_{i,j-1}) + \Delta X_j \sum_k w_{k,j} = 0. \quad (1.6.7)$$

Momentum conservation

The momentum conservation equation is decomposed in two equations: axial momentum

$$\begin{aligned} \frac{\Delta X_j}{\Delta t} (m_{i,j} - m_{i,j}^{\text{old}}) + m_{i,j} U'_{i,j} + \Delta X_j \sum_k w_{k,j} U'_{k,j} &= -A_{i,j} (p_{i,j} - p_{i,j-1}) - g A_{i,j} \Delta X_j \rho_{i,j} \\ &- \frac{1}{2} \left(\frac{\Delta X f \phi^2}{D_h \rho l} + K v' \right)_{i,j} m_{i,j} \frac{m_{i,j}}{A_{i,j}} - \Delta X_j \sum_k w'_{k,j} (U'_{i,j} - U'_{n(k),j}), \end{aligned} \quad (1.6.8)$$

and lateral momentum (where convective transport is neglected)

$$\frac{\Delta X_j}{\Delta t} (w_{k,j} - w_{k,j}^{\text{old}}) + (\bar{U}'_{k,j} w_{k,j} - \bar{U}'_{k,j-1} w_{k,j-1}) = \frac{s_k}{l_k} \Delta X_j \Delta p_{k,j-1} - \left(K_G \frac{\Delta X v'_k}{s_k l_k} \right)_j |w_{k,j}| w_{k,j}. \quad (1.6.9)$$

Energy conservation

$$\frac{A_{i,j}}{\Delta t} [\rho''_{i,j}(h_{i,j} - h_{i,j}^{\text{old}}) + h_{i,j}(\rho_{i,j} - \rho_{i,j}^{\text{old}})] + \frac{1}{\Delta X_j} (m_{i,j}h_{i,j} - m_{i,j-1}h_{i,j-1}) \quad (1.6.10)$$

$$+ \sum_k w_{k,j}h_{k,j} = Q_{i,j} - \sum_k w'_{k,j}(h_{i,j} - h_{n(k),j}). \quad (1.6.11)$$

Notation in the previous expressions is as follows:

- i : channel index,
- j : slice index,
- k : gap,
- $n(k)$: channel neighbor belonging to gap k ,
- A : cross-sectional area of the sub-channel,
- ΔX : axial cell length (m),
- Δt : time step (s),
- m : axial mass flow rate (kg s^{-1}),
- h : specific mixture enthalpy (J K^{-1}),
- h_{fg} : evaporation enthalpy (J K^{-1}),
- Q : linear power released to the sub-channel (W m^{-1}),
- w' : turbulent cross-flow ($\text{kg m}^{-1} \text{s}^{-1}$),
- p : pressure at axial boundary (Pa),
- f : single-phase friction coefficient,
- g : gravity (m/s^2),
- ϕ^2 : two-phase friction multiplier,
- D_h : hydraulic diameter (m),
- K : axial pressure loss coefficient,
- K_G : lateral gap pressure loss coefficient,
- s : gap width between two neighboring rods (m),
- l : distance of neighboring sub-channels midpoints (m),
- $\rho'' = \rho^{\text{old}} - h_{\text{fg}} \frac{\partial \psi}{\partial h}$,
- $\psi = \rho_{\text{liq}}x(1 - \alpha) - \rho_{\text{vap}}(1 - x)$,
- x : steam quality,
- α : void fraction,
- $v' = \frac{x^2}{\alpha \rho_{\text{vap}}} + \frac{(1 - x)^2}{(1 - \alpha) \rho_{\text{liq}}}$,
- $U' = \frac{m}{A} v'$,
- liq : liquid,
- vap : vapor,
- old : value at previous time step.

1.6.4 Presentation of SUBCHANFLOW

SUBCHANFLOW is a sub-channel code developed at KIT (Karlsruhe Institute of Technology, Germany) and based on the COBRA family of solvers (D. Basile and Brega, 1999; Rowe, 1973; Wheeler et al., 1976). It solves the 3 conservation equations detailed in Sec. 1.6.3, as well as the constitutive equations (e.g., to compute axial and lateral flow rates, pressure, enthalpy and void fraction), with the fluid being modeled as a homogeneous two-phase liquid-vapor mixture. The system is solved axially, layer by layer, with an implicit scheme. The code can solve transients and steady-state problems.

SUBCHANFLOW also solves heat transfer in the fuel. In order to compute the fuel rod temperature, each axial slice of the fuel is divided into radial rings and a finite-volume method is applied. The fraction of the power that is directly put in the coolant must be defined. The fuel rod temperature is then computed depending on the power deposited in the rod and on the cladding-to-coolant heat transfer properties.

Different empirical correlations for pressure drop, heat transfer and void generation are available. The physical models were validated with experimental data from the NUPEC PWR Subchannel and Bundle Tests (Imke and Sanchez, 2012). For the boundary conditions, the temperature at the inlet and the pressure at the outlet must be assigned.

One two-dimensional and two three-dimensional input meshes (one for the coolant and one for the fuel) are required for the calculation. For the coolant, the mesh can be either coolant- or fuel-centered. Figure 1.3 shows the two different options to model a channel. A preprocessor allows generating the meshes for complex geometries (squared or hexagonal) at different levels: pin, assembly or core level.

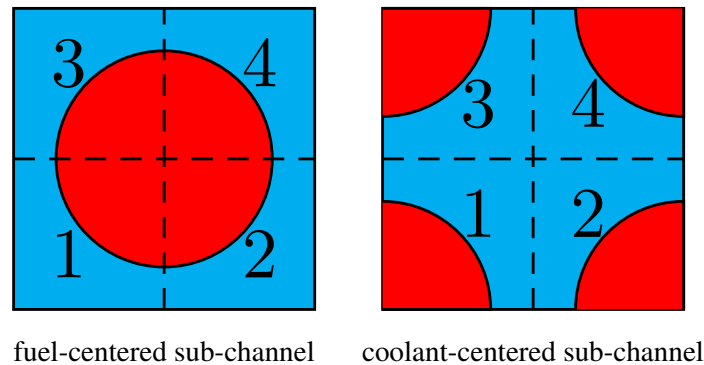


Figure 1.3 – Two different ways to model a sub-channel: either the cell is centered on the fuel rod (in red), or it is centered on the coolant (in blue).

Part I

Kinetic Monte Carlo: time-dependent Monte Carlo neutron transport

Chapter 2

Description of kinetic methods for TRIPOLI-4

In this chapter, we detail the strategy adopted in TRIPOLI-4 to implement the kinetic Monte Carlo methods. In this thesis, we have implemented a new variance-reduction technique for the specific needs of kinetic simulations. The efficiency of the different techniques is investigated in this chapter.

2.1 Challenges

Monte Carlo stationary calculations, such as those that solve criticality or fixed-source problems, do not take time into account. Precursors are not explicitly simulated, and their contributions are integrated into the total fission operator, cumulating the prompt and the delayed fission operators defined in Eqs. (1.2.5) and (1.2.8). The Boltzmann equation (Eq. 1.2.2) is solved under the assumption that the time derivative of the flux vanishes (e.g., solving Eq. (1.2.10)).

To this day, Monte Carlo methods have been almost exclusively applied to the solution of stationary problems, due to the very high computational cost (in terms of both memory and CPU time) required to generate the particle trajectories (Lux and Koblinger, 1991). However, thanks to the growing available computer power and efficient variance-reduction techniques, Monte Carlo methods can now be also applied to the solution of non-stationary transport problems, as witnessed by the increasing number of scientific publications on this subject (Légrády and Hoogenboom, 2008; Sjenitzer and Hoogenboom, 2011a, 2013; Hoogenboom and Sjenitzer, 2014; Leppänen, 2013; Valtavirta et al., 2016; Mylonakis et al., 2017; Trahan, 2018; Molnar et al., 2019).

The goal of “kinetic” (i.e., time-dependent) Monte Carlo methods is to address the solution of the full time-dependent Boltzmann equation, coupled to the precursor equation (Eq. 1.2.3). The first attempts at solving these equations were proposed by Kaplan (1958) in the 1950s, soon somewhat neglected because of limitations in computer power, and later reconsidered by Légrády and Hoogenboom (2008) and further improved by Sjenitzer and Hoogenboom (2011a, 2013).

A few key issues have to be taken into account when dealing with time-dependent Monte Carlo transport:

1. the phase space must be extended so as to accommodate the time variable;

2. if the initial conditions correspond to a critical (i.e., equilibrium) regime, the particle source must be prepared in the equilibrium state before running the kinetic Monte Carlo simulation for the time-dependent transport;
3. population control should be enforced to prevent the neutron and precursor populations to die out or to grow unbounded;
4. the quantities $\beta_{\text{eff}}/\Lambda_{\text{eff}}$ and λ provide the average rates at which a neutron is converted into a precursor, and conversely. The typical ratio $\beta_{\text{eff}}/(\lambda \times \Lambda_{\text{eff}})$ is of the order of 10^4 for light-water-moderated reactors. This implies that at the critical regime, when the neutron and precursor populations are in equilibrium, precursors are considerably more abundant than neutrons within the core;
5. the code must be able to simulate real transient scenarios, hence the geometry should also be time-dependent.

The purpose of this chapter is to address each issue and show which strategies have been chosen for TRIPOLI-4.

2.2 Time dependence

2.2.1 Extending the phase space

In kinetic simulations, the precursor population is considered in addition to the neutron population. A new precursor particle type has been thus defined for TRIPOLI-4. Precursors are regrouped in families depending on their decay time, as defined by the nuclear data library. Precursors are not transported, but serve as a “buffer” for delayed neutron emission.

In stationary Monte Carlo simulations, the phase space explored by the neutrons is described by space, energy and angle coordinates: time is not explicitly considered. In kinetic simulations, the time-dependent behaviour of the particles (both neutrons and precursors) is simply taken into account by assigning each particle a “time label” t that is set to $t = 0$ s at the beginning of the simulation and progressively updated in the course of the simulated history on the basis of the particle position and speed.

2.2.2 Scoring time grid

Statistical events in stationary Monte Carlo simulations are recorded independently of their time of occurrence, since time is not explicitly considered. In order to collect events in time-dependent transport, a scoring time grid must be defined: when collisions or track lengths are to be recorded so as to estimate the particle flux or other tallies, events are partitioned into the time bins. By construction, the contents of each bin $[t_q, t_{q+1}]$ at the end of the simulation represent the ensemble-averaged time integral of the corresponding Monte Carlo score, between times t_q and t_{q+1} .

The scoring time mesh is arbitrary, and the bins $[t_q, t_{q+1}]$ can be irregularly spaced and have any size: contrary to deterministic methods, which require the time mesh to be sufficiently fine in order for the calculation to converge, the Monte Carlo bins provide the exact integral over the phase space elements.

2.2.3 Simulation time grid

Specific variance-reduction and population-control techniques implemented in TRIPOLI-4 for kinetic capabilities share a common requirement: they must be applied on the basis of a time mesh. For convenience, in TRIPOLI-4, one single simulation time grid is defined by the user for the different techniques. The available population-control techniques in TRIPOLI-4 are *Russian roulette and splitting* and *combing* (see Sec. 2.5); for variance reduction, we can use forced decay (see Sec. 2.4), branchless collisions (see Sec. 2.6.1) and population importance sampling (see Sec. 2.7).

It is important to note that the simulation time grid does not incur any discretization error. The simulation time steps are in principle arbitrary. Later we will detail why the system reactivity in practice imposes some constraints on their size.

2.3 Critical source

2.3.1 Sampling the neutrons and precursors

Sampling equations

The typical configuration at the initial time for a kinetic Monte Carlo simulation is either a system containing only a neutron source (e.g., for reactor start-up calculations), or a system in a critical state (e.g., for the departure from a stationary regime). The former case does not require extra calculations before the kinetic simulation: the kinetic simulation starts directly with the specified neutron source. The latter deserves instead a special treatment on specific points, which will be detailed here.

The simplest way to prepare a reactor configuration on the critical state is to apply the power iteration algorithm starting from an arbitrary source. This formally corresponds to solving the critical k eigenvalue Eq. (1.2.10) for the dominant eigenpair $\{k_{\text{eff}}, \varphi_{k_{\text{eff}}}\}$, and thus inferring the initial neutron and precursor population at equilibrium on the basis of the fundamental eigenmode $\varphi_{k_{\text{eff}}}(\mathbf{r}, \mathbf{v})$. More precisely, the number of starting neutrons corresponding to the critical condition is given by (Sjenitzer and Hoogenboom, 2013)

$$N^{\text{eq}}(\mathbf{r}, \mathbf{v}) = \frac{\psi_{k_{\text{eff}}}(\mathbf{r}, \mathbf{v})}{v\Sigma_t(\mathbf{r}, v)}, \quad (2.3.1)$$

where $\psi_{k_{\text{eff}}} = \Sigma_t \varphi_{k_{\text{eff}}}$ is the critical collision density that can be estimated at each collision during the power iteration, provided that the fission sources have converged to their asymptotic shape. Observe that the distribution of the neutron population $N^{\text{eq}}(\mathbf{r}, \mathbf{v})$ is not given by the fission sources computed by the power iteration, namely, $N^{\text{eq}}(\mathbf{r}, \mathbf{v}) \neq F_p \varphi_k(\mathbf{r}, \mathbf{v}) + F_d \varphi_k(\mathbf{r}, \mathbf{v})$.

Practically, the neutron source is sampled over all collisions occurring during the last power iteration, and the weight w_n^{eq} of each source neutron n is

$$w_n^{\text{eq}} = \frac{w_n}{v\Sigma_t(\mathbf{r}, v)}, \quad (2.3.2)$$

with w_n the weight of the neutron at collision.

Concerning the precursor population at the initial time, by imposing the steady state condition in Eq. (1.2.3) we get

$$c_j^{\text{eq}}(\mathbf{r}) = \frac{1}{\lambda_j} \int v_d^j(v') \frac{\Sigma_f}{\Sigma_t}(\mathbf{r}, v') \psi_{k_{\text{eff}}}(\mathbf{r}, v') dv'. \quad (2.3.3)$$

Following Sjenitzer and Hoogenboom (2013), all possible precursors associated to a given fission event are combined into a single representative precursor particle carrying the total statistical weight. For this combined precursor particle for a single nuclide we get

$$c^{\text{eq}}(\mathbf{r}) = \sum_j c_j^{\text{eq}}(\mathbf{r}), \quad (2.3.4)$$

thus

$$c^{\text{eq}}(\mathbf{r}) = \frac{1}{\bar{\lambda}} \int v_d(v') \frac{\Sigma_f(\mathbf{r}, v')}{\Sigma_t(\mathbf{r}, v')} \psi_{k_{\text{eff}}}(\mathbf{r}, v') \mathbf{d}v', \quad (2.3.5)$$

where

$$v_d(v') = \sum_j v_d^j(v'), \quad (2.3.6)$$

and

$$\bar{\lambda} = \frac{\beta}{\sum_j \frac{\beta_j}{\lambda_j}} \quad (2.3.7)$$

is the family-averaged decay rate of the combined precursor (Sjenitzer and Hoogenboom, 2013), with β_j the fraction of fission neutrons which are emitted by the j -th family, and $\beta = \sum_j \beta_j$. Note that β and β_j are not the effective parameters (i.e., adjoint-weighted parameters), but the unweighted kinetics parameters.

Similarly as Eq. (2.3.2), the weight of each source precursor w_c^{eq} is sampled over all collisions according to

$$w_c^{\text{eq}} = \frac{1}{\bar{\lambda}} w_n v_d(v') \frac{\Sigma_f(\mathbf{r}, v)}{\Sigma_t(\mathbf{r}, v)}. \quad (2.3.8)$$

In order to select the proper family for the decay of the combined precursor, care must be taken since the family repartition in the critical regime is different from that at the fission events. Indeed, the probability of choosing the j -th family at time t , for precursors created at time t_0 , is given by

$$p(j, t|t_0) = \frac{\lambda_j f_d^j e^{-\lambda_j(t-t_0)}}{\sum_k \lambda_k f_d^k e^{-\lambda_k(t-t_0)}}, \quad (2.3.9)$$

with f_d^j the fraction of delayed neutrons of the j -th family. At equilibrium it is given by

$$f_d^j = \frac{\bar{\lambda} \beta_j}{\lambda_j \beta}, \quad (2.3.10)$$

whereas, at fission events, it is given by

$$f_d^j = \frac{\beta_j}{\beta}. \quad (2.3.11)$$

At equilibrium, the decay time probability density of a precursor particle generated at t_0 is

$$P^{\text{eq}}(t|t_0) = \sum_j \bar{\lambda} \frac{\beta_j}{\beta} e^{-\lambda_j(t-t_0)}, \quad (2.3.12)$$

and the statistical weight of a delayed neutron emitted upon decay by a precursor particle originating from the equilibrium condition is given by

$$\mathcal{W}_{\text{delayed}}^{\text{eq}}(t|t_0) = P^{\text{eq}}(t|t_0) w_C, \quad (2.3.13)$$

with w_C the initial precursor weight. The energy of the delayed neutron must be chosen by sampling the delayed neutron spectra corresponding to the sampled family. Finally, the statistical weight of a precursor coming from the equilibrium source is

$$\mathcal{W}_c^{\text{eq}}(t|t_0) = w_C \sum_j \frac{\bar{\lambda} \beta_j}{\lambda_j \beta} e^{-\lambda_j(t-t_0)}. \quad (2.3.14)$$

At fission events, the decay time probability density at a time t for precursors generated at time t_0 is given by

$$P(t|t_0) = \sum_j \lambda_j \frac{\beta_j}{\beta} e^{-\lambda_j(t-t_0)}, \quad (2.3.15)$$

and the physical weight of the precursor at a time t is given by

$$\mathcal{W}(t|t_0) = w_C \sum_j \frac{\beta_j}{\beta} e^{-\lambda_j(t-t_0)}. \quad (2.3.16)$$

The different precursor types described by Eqs. (2.3.14) and (2.3.16) are practically implemented by storing the distribution over families in the precursor particle.

Fission source and critical source

For an illustration of the difference between the fission source transferred from one power iteration to another and the critical source sampled for the kinetic simulation, we have performed a criticality calculation followed by the sampling of the critical source on a simple test case. The configuration is a 2x2 array of pin cells and we have simulated 200 batches with 4×10^4 neutrons per batch. The histograms of the particle weights for both fission and critical sources are presented in Fig. 2.1. The weight of fission neutrons sampled during the last power iteration are represented on the left: neutrons are located in the fuel only. Neutrons (resp. precursors) sampled for the critical source are represented in the middle (resp. right). Particles are sampled over all collisions occurring in the last power iteration according to Eqs. (2.3.2) and (2.3.8). Neutrons can be found anywhere in the cells, including in the moderator; actually, there are even more neutrons in the moderator than in the fuel. On the contrary, precursors are created at the fission source locations. This is why precursors are located in the fuel only. Moreover, precursors are not transported during the kinetic simulation, so they will not move from their initial location in the fuel.

Also, the unbalanced ratio between neutrons and precursors in the critical source is clearly visible: precursor weights are much higher than neutron weights. Quantitatively, the ratio between the total weight of precursors and the total weight of neutrons is about 10^4 . In fact, as mentioned above, precursors are considerably more abundant than neutrons at equilibrium.

2.3.2 Normalization between criticality and kinetic calculations

It is interesting to notice that the neutron and precursor weights sampled for the kinetic calculation according to Eqs. (2.3.2) and (2.3.8) have the dimensions of a time. This is unusual, since *Russian roulette and splitting* need dimensionless weights to act upon. Therefore, the weights must be normalized for the kinetic calculation in such a way that they become dimensionless. The most natural choice is to normalize the weights by dividing them by the total neutron and precursor weight. The normalization factor q that must be applied is

$$q = \frac{\sum_n w_n^{\text{eq}} + \sum_c w_c^{\text{eq}}}{N}, \quad (2.3.17)$$

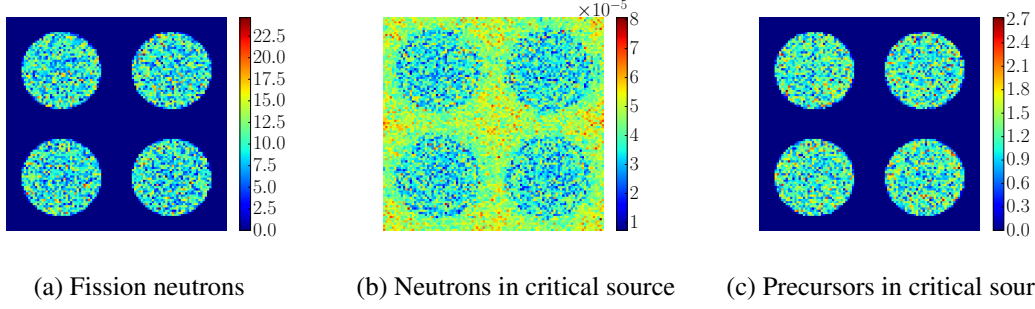


Figure 2.1 – Histogram of the particle weights in a 2x2 array of pin cells. Left: fission neutrons sampled during the last power iteration are located in the fuel only. Middle: neutrons sampled for the critical source can be found anywhere in the cells. Right: precursors for the critical source are created at the fission source locations.

with w_n^{eq} defined by Eq. (2.3.2), w_c^{eq} defined by Eq. (2.3.8) and N being the intensity of the fission source. Actually, q is a close estimator of the neutron effective lifetime. Indeed,

$$\frac{\sum_n w_n^{\text{eq}}}{N} \approx \frac{\iint \frac{1}{v} \varphi_{k_{\text{eff}}}(\mathbf{r}, \mathbf{v}) \, d\mathbf{r} d\mathbf{v}}{\frac{1}{k_{\text{eff}}} \iint F \varphi_{k_{\text{eff}}}(\mathbf{r}, \mathbf{v}) \, d\mathbf{r} d\mathbf{v}}, \quad (2.3.18)$$

which is the unweighted analog of $k_{\text{eff}} \times \Lambda_{\text{eff}}$, according to the definition of the generation time (the formula was given by Eq. (1.2.26) for the “effective” parameter, which is weighted by the adjoint eigenmode). Similarly,

$$\frac{\sum_c w_c^{\text{eq}}}{N} \approx \frac{\sum_j \frac{1}{\lambda_j} \int v_d^j(v) \Sigma_f(\mathbf{r}, v) \varphi_{k_{\text{eff}}}(\mathbf{r}, v) \, d\mathbf{r} d\mathbf{v}}{\frac{1}{k} \iint F \varphi_{k_{\text{eff}}}(\mathbf{r}, v) \, d\mathbf{r} d\mathbf{v}} \quad (2.3.19)$$

$$\approx \frac{\sum_j \frac{1}{\lambda_j} \iint F_d^j \varphi_{k_{\text{eff}}}(\mathbf{r}, v) \, d\mathbf{r} d\mathbf{v}}{\frac{1}{k_{\text{eff}}} \iint F \varphi_{k_{\text{eff}}}(\mathbf{r}, v) \, d\mathbf{r} d\mathbf{v}}, \quad (2.3.20)$$

which is the unweighted analog of $k_{\text{eff}} \times \sum_j \beta_{\text{eff},j} / \lambda_j$ (one can refer to Eq. (1.2.25) for a definition of the effective parameter). Finally, we get

$$q \approx k_{\text{eff}} \times l_0, \quad (2.3.21)$$

with l_0 being an unweighted analog of the neutron effective lifetime defined by Eq. (1.3.6).

2.3.3 Optimizing the use of criticality cycles

The TRIPOLI-4 implementation of the sampling of the neutron and precursor population for a critical system proceeds by first running a regular power iteration over a large number of *inactive cycles*, with neutrons alone. During this convergence phase, the (arbitrary) neutron source relaxes to the dominant k eigenmode of the system (Sec. 1.2.2). Fission neutrons are transferred from one cycle to the next one. During the last inactive cycle, neutron and precursor sources for the kinetic phase are sampled at each collision, according to Eqs. (2.3.1) and (2.3.5), in order to convert the critical flux $\varphi_{k_{\text{eff}}}(\mathbf{r}, \mathbf{v})$ into neutron and precursor particle densities. The kinetic sources are then injected at $t = 0$ s and are followed until the stopping time is reached.

We must now consider that TRIPOLI-4 averages the calculation results over consecutive “batches” (or replicas). In the implementation proposed by Sjenitzer (2013), each batch represented a genuine independent replica of the full calculation, namely a source-convergence phase followed by a kinetic phase. This solution however is time-consuming. We have set a new scheme in order to reduce the number of inactive cycles: the fission neutrons generated in the last inactive cycle of the first batch are injected in the inactive cycles of the following one. This reduces the overhead of inactive cycles, at the cost of introducing correlations among batches (the estimation of the statistical uncertainty over the calculation results can be thus affected). Note, however, that this is essentially the same difficulty that one faces in conventional criticality calculations. A scheme of the process is illustrated in Fig. 2.2. The user can assess the impact of such correlations by increasing the number of inactive cycles for each replica.

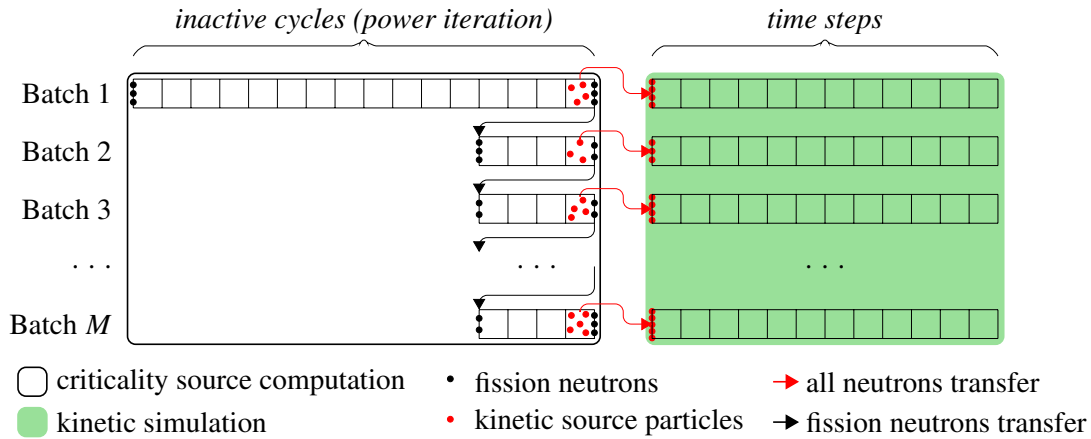


Figure 2.2 – Kinetic TRIPOLI-4 simulation process with M batches. A complete power iteration is performed once by the first batch. During the final inactive cycle, precursors are sampled so that the kinetic simulation can begin while the generated fission neutrons are transferred to the second batch, which runs a few additional power iterations in order to ensure reasonable decorrelation. The third batch gets the fission neutrons generated by the second one, and so on. In a parallel run, each processor would apply this scheme.

2.3.4 Readjustment of the emitted number of neutrons

The simulation of a nuclear reactor is necessarily affected by uncertainties at different levels: geometry, technological details such as the compositions, and nuclear data. For this reason, the effective multiplication factor for nominally critical conditions may fail to be equal to 1. The associated stationary flux $\varphi_{k_{\text{eff}}}$ will also in general be distorted with respect to the actual stationary flux. Even small deviations from 1 may be important in kinetic Monte Carlo simulations, because the system will drift away from the stationary state during the kinetic simulation. Therefore, some adjustments have to be made in order to minimize the deviations from the critical conditions.

For example, uncertainties on the geometry can be mitigated by adjusting control-rods positions. It is also possible to compensate the lack of precision on the compositions by adjusting the boron concentration in the moderator. Finally, regarding uncertainties on nuclear data, in TRIPOLI-4, it is possible to rescale the number of neutrons ν produced by fission by the multiplication factor k_{eff} (obtained by a preliminary criticality calculation), similarly to what is done for kinetic simulations in deterministic solvers. By modifying the production term, the deviation from criticality can thus be mitigated. The residual deviation is typically of the order of the

statistical uncertainty over the k_{eff} value that was used for the correction. The other corrections (adjusting control-rod positions or boron concentration) have a more direct physical meaning. However, the search for the critical parameter values is iterative and requires a large number of calculations (Mancusi and Zoia, 2018), whereas rescaling the number of fission neutrons only requires one extra calculation. At any rate, the physical impact of this choice deserves further investigation.

2.4 Precursors forced decay

In principle, it would be possible to explicitly simulate prompt and delayed neutrons at fission events by closely following the laws of physics: prompt neutrons (on average ν_p) would be instantaneously emitted and would inherit the time label t_f of the parent particle at birth (i.e., at fission); precursors would not be directly taken into account: instead, their decay time τ would be sampled from the corresponding exponential probability $P_j(\tau) = \lambda_j e^{-\lambda_j \tau}$ (j being the index of the precursor family). Delayed neutrons (on average ν_d) would then be injected into the simulation at a time $t_f + \tau$, precisely at the fission site associated to the parent particle (we neglect precursor migration), and with energy and angle determined by the delayed fission distribution.

However, in an almost critical system, each neutron chain will on average induce a single precursor, which will in turn induce a neutron chain by decaying to a delayed neutron. The lifetime of the neutron chain is of the order of 10 ms, while the decay takes a few seconds (Keepin, 1965). Due to this large separation between the neutron and precursor time scales, in an analog Monte Carlo simulation no particle would be produced during the time between the creation of a precursor and its decay to a delayed neutron (Légrády and Hoogenboom, 2008). In a real system at full power the number of fission events per unit time is so large ($\sim 10^{19} \text{ s}^{-1}$) that fission chains will thoroughly superpose by the mere effect of statistics (Sjenitzer, 2013). In Monte Carlo simulations the number of simulated chains is however much smaller than in real reactors because of limitations on CPU time and memory. Hence, ingenious algorithms must be conceived so as to handle the presence of the two time scales.

A possible strategy to overcome this issue has been proposed by Légrády and Hoogenboom (2008) and consists in considering two separate populations, namely neutrons and precursors, and applying variance-reduction techniques to the precursor population. The precursor particles do not directly contribute to Monte Carlo scores during the simulation: their role is to provide a “buffer” for delayed neutrons and regulate the statistical weight of the neutron population that is transported and thus directly contributes to Monte Carlo scores. Precursors can be created at fission events and can possibly also be present at the beginning of the simulation in the case of a critical source, as detailed in Sec. 2.3.

However, the introduction of the precursor particles does not solve the problem of the under-sampling of neutrons between the creation of a precursor and its decay into a delayed neutron, as illustrated in Fig. 2.3. Moreover, because of the very unbalanced ratio between neutrons and precursors, it is not possible to indefinitely increase the number of neutrons: the permanent storage of 10^4 precursors per neutron would induce memory issues.

One solution consists in *forcing the decay* of the precursor population into delayed neutrons, which ensures a larger fraction of neutrons in the fission chains and thus a reduced variance in the total population per unit time, as illustrated in Fig. 2.4. The idea is to modify the decay probability of the precursor particles, and yet preserve an unbiased Monte Carlo game by correspondingly modifying the statistical weight of the delayed neutrons emitted upon decay. Several

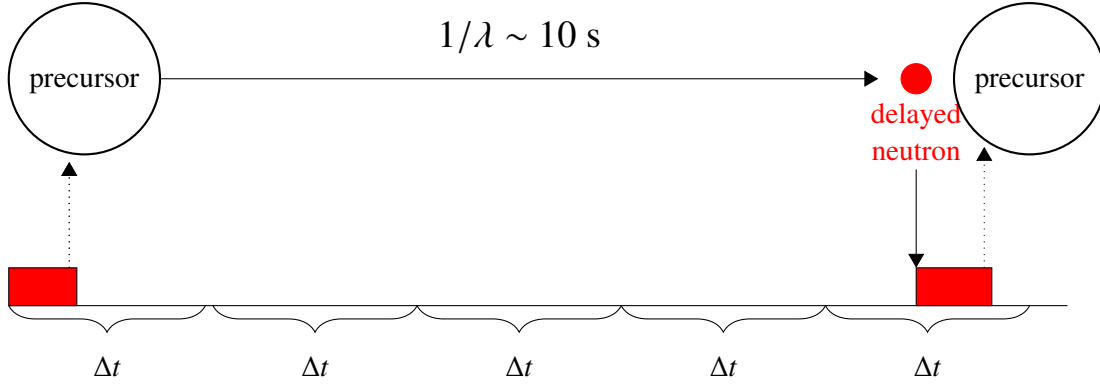


Figure 2.3 – Scheme of the analog process of precursor decay with one precursor. When the precursor finally decays, only one time bin is filled with a delayed neutron: this leads to variance jumps in the different time bins.

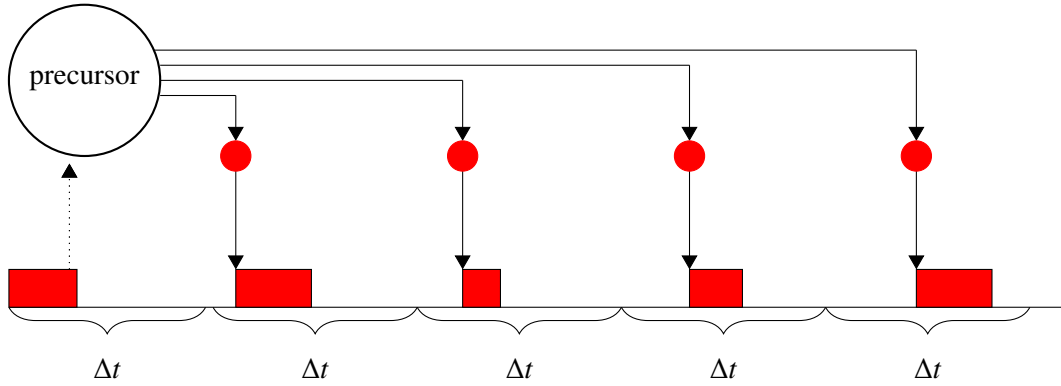


Figure 2.4 – Scheme of the forced decay with one precursor. The precursor is forced to decay in each time bin Δt , so that each time bin is filled with neutrons.

choices are possible, the simplest being to introduce an arbitrary time mesh $t_0 < t_1 < \dots < t_N$ and force decay in each bin of the mesh (Légrády and Hoogenboom, 2008): this is the simulation time mesh introduced in Sec. 2.2.3. Stochastic neutron transport formally corresponds to branching exponential flights, which ensures the Markovian nature of the process: as such, the particle trajectories can be stopped and restarted at each time bin crossing without altering the statistical features of the underlying transport process (Sjenitzer and Hoogenboom, 2013). Thus, TRIPOLI-4 was modified so as to stop particles whenever they reach the end of a bin of the time mesh. The time of the forced decay can be taken, e.g., to be uniformly distributed in each bin $[t_q, t_{q+1}]$, i.e.,

$$p_{decay}(t) = \frac{1}{t_{q+1} - t_q} \chi(t_q, t_{q+1}), \quad (2.4.1)$$

where $\chi(a, b)$ is the marker function of the interval $[a, b]$. In order for the emitted delayed neutron to undergo a fair Monte Carlo game, its statistical weight must be taken as

$$\mathcal{W}_{delayed}(t|t_0) = (t_{q+1} - t_q) \sum_j \lambda_j \frac{\beta_j}{\beta} e^{-\lambda_j(t-t_0)} w_C, \quad (2.4.2)$$

where w_C is the initial weight of the emitting precursor.

Once the delayed neutron has been created during the time step $[t_q, t_{q+1}]$, the precursor is not killed, but just added to the buffer of particles that are to begin the following time step

$[t_{q+1}, t_{q+2}]$, with a weight given by Eq. (2.3.16)

$$\mathcal{W}(t_{q+1}|t_0) = w_C \sum_j \frac{\beta_j}{\beta} e^{-\lambda_j(t_{q+1}-t_0)}.$$

The delayed neutrons, once emitted, are transported during the current time step, and may possibly initiate new fission chains. The process is illustrated in Fig. 2.5.

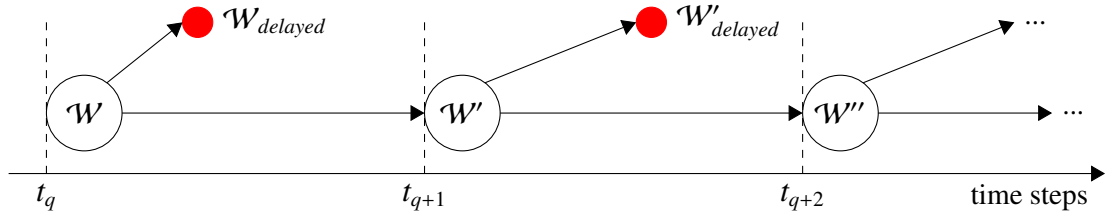


Figure 2.5 – Scheme of delayed neutron emission with forced decay algorithm. Precursors (black circles) are forced to decay in each bin of the simulation time grid, in order to produce a delayed neutron (red disks), and then survive for the next time step.

2.5 Population control

During kinetic simulations, the standard *Russian roulette and splitting* is applied at each collision but this is most often not enough to control the population size. Depending on the reactivity of the system, the population size (number of neutrons and precursors) may shrink to zero (sub-critical case) or grow without bounds (supercritical case) during the time steps. Both scenarios are problematic from the point of view of Monte Carlo simulation; for this reason, TRIPOLI-4 applies some control mechanisms to the population size during the calculation: *Russian roulette and splitting* or *combing*, detailed below. In principle these mechanisms could be applied at every change of the population size, but this would incur a large CPU overhead. A weaker control can be applied by defining in advance at which times population control will be applied. This is one of the purposes of the simulation time grid defined in Sec. 2.2.3. When the time labels of the particles reach the end of a time step of this grid, population control is applied.

While the simulation time steps are in principle arbitrary, in practice population control restricts the choice of the interval boundaries. If the system is super-critical, the population might still diverge within a step if it is not small enough; conversely, if the system is sub-critical the population might disappear before the regularizing mechanisms of population control get the opportunity to act. Moreover, since precursors are forced to decay in each time step and then survive for the following time step, the weight of the delayed neutrons that are emitted decreases as a function of time. At the same time, the number of precursors typically increases over time, and population control should be effectively applied.

As explained in Sec. 2.2.3, for the sake of simplicity, we choose to apply population control at the end of the time intervals selected for forced decay. This has two advantages: *i*) the user only needs to input one time grid, and *ii*) it does not introduce additional stopping times for the precursors. However, in some cases it may be advantageous to apply population control independently of forced decay.

2.5.1 Russian roulette and splitting

The *Russian roulette and splitting*, as introduced in Sec. 1.5.2, can be separately applied to the neutron and precursor populations banked at the end of each time step. The roulette would be straightforward for neutrons, but it would have a negligible effect since neutrons are already submitted to *Russian roulette and splitting* at every collision, so that their weight has in principle already been regulated. Therefore, in TRIPOLI-4, it is not applied to neutrons. Concerning precursors, the roulette is applied on the basis of the expected delayed neutron weight that can be emitted during the following time step (Sjenitzer and Hoogenboom, 2011a). The expected delayed neutron weight during time step $[t_q, t_{q+1}]$ is given using Eq. (2.4.2) by

$$\bar{W}_{delayed}(t_q, t_{q+1}|t_0) = \frac{1}{t_{q+1} - t_q} \times \int_{t_q}^{t_{q+1}} \mathcal{W}_{delayed}(t|t_0) dt \quad (2.5.1)$$

$$= w_C \sum_j \frac{\beta_j}{\beta} (e^{-\lambda_j(t_q - t_0)} - e^{-\lambda_j(t_{q+1} - t_0)}), \quad (2.5.2)$$

where t_{q+1} is the starting time of the following time interval. For precursors generated by the equilibrium source, Eq. (2.3.13) gives

$$\bar{W}_{delayed}^{eq}(t_q, t_{q+1}|t_0) = \frac{1}{t_{q+1} - t_q} \times \int_{t_q}^{t_{q+1}} \mathcal{W}_{delayed}^{eq}(t|t_0) dt \quad (2.5.3)$$

$$= w_C \sum_j \frac{\bar{\lambda} \beta_j}{\lambda_j \beta} (e^{-\lambda_j(t_q - t_0)} - e^{-\lambda_j(t_{q+1} - t_0)}). \quad (2.5.4)$$

2.5.2 Combing

The combing method (Booth, 1996) is a population-control technique that can be used at the end of each time step in kinetic Monte Carlo, as an alternative to *roulette and splitting*. The aim of combing, as initially proposed by T. E. Booth for stationary Monte Carlo transport problems, is to normalize the number of transported particles while preserving their total weight and possibly the proportion of respective weights in the case of multiple particle species. For kinetic simulations, combing can achieve variance reduction and save computer time by keeping the population size approximately constant over time steps. Indeed, combing prevents the population from growing without bounds for a super-critical system, and from dying for a sub-critical system. It should be stressed that combing might also prove useful for critical conditions, in that it prevents the occurrence of the so-called ‘‘critical catastrophe’’, i.e., the divergence of the variance of the population due to fluctuations of fission chains (De Mulatier et al., 2015).

Suppose that at the end of a time step the population is composed of K individuals, which are to be combed to M individuals. We define W to be the length of the comb, i.e., the total weight of the K individuals to be combed, namely,

$$W = \sum_{i=1}^K w_i. \quad (2.5.5)$$

The idea behind the comb is to sample with replacement M particles among the K by using an equally spaced weight interval over the total available weight. It has been shown that this algorithm is unbiased provided that a random offset is imposed at the beginning of the comb, when selecting the first particle. More precisely, the positions of the comb teeth are defined as

$$T_m = \xi \frac{W}{M} + (m - 1) \frac{W}{M}; \quad m = 1, \dots, M, \quad (2.5.6)$$

where ξ is a uniform random number $\xi \sim \mathcal{U}(0, 1]$ needed for determining the starting position of the comb (see Fig. 2.6).

Depending on ξ , either j or $j + 1$ teeth hit an interval of length w_i , with

$$j \leq w_i \frac{M}{W} \leq j + 1. \quad (2.5.7)$$

This means that

$$p_{i,j} = j + 1 - w_i \frac{M}{W} \quad (2.5.8)$$

yields the probability that j teeth fall in the interval i , and

$$p_{i,j+1} = w_i \frac{M}{W} - j \quad (2.5.9)$$

yields the probability that $j + 1$ teeth fall in the interval i . The expected weight for a single particle after combing is

$$\bar{w}'_i = p_{i,j} j \frac{W}{M} + p_{i,j+1} (j + 1) \frac{W}{M} = w_i \frac{M}{W} \frac{W}{M} = w_i. \quad (2.5.10)$$

Each new particle (out of M) is then assigned a weight

$$w'_i = \frac{W}{M}. \quad (2.5.11)$$

This way, the total expected weight after combing is preserved:

$$W' = \sum_{i=1}^M w'_i = W. \quad (2.5.12)$$

In the more general case of several populations, as for the neutrons and precursors, the comb is independently applied to each population.

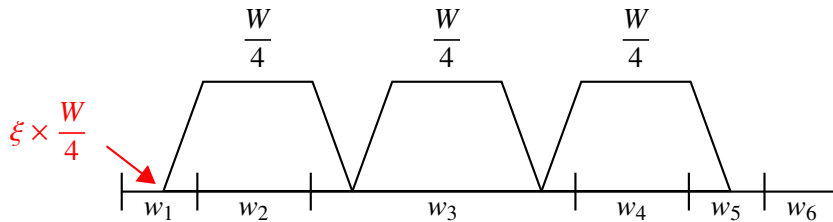


Figure 2.6 – An application of the combing method (Booth, 1996) with 6 particles with total weight W being combed into $M = 4$ particles. Particle 1 is copied once with weight W/M . Particle 3 is copied twice with weight W/M . Particle 5 is copied once with weight W/M . Particles 2, 4 and 6 are not copied. One should note that the total weight after combing remains equal to W .

In addition to the population-control method (either *Russian roulette and splitting* or *combing*) that is applied at the end of each time step of the simulation, combing is applied at the very beginning of the kinetic simulation, just after the sampling of the critical neutron and precursor populations. This has two goals. First, as explained in Sec. 2.3.2, after the critical sampling, weights carry an additional factor having the dimension of a time and they must be normalized so that they represent a proper statistical weight. Second, the total number of particles at $t = 0$ will be equal to the number of particles N chosen by the user.

configuration	nominal	critical	subcritical	supercritical
k_{eff}	$1.255690 \pm 2.9 \times 10^{-5}$	$0.999995 \pm 2.2 \times 10^{-5}$	$0.965595 \pm 2.0 \times 10^{-5}$	$1.007246 \pm 2.1 \times 10^{-5}$
Λ_{eff} [μs]	13.2655 ± 0.0016	16.6591 ± 0.0019	16.4866 ± 0.0018	16.6964 ± 0.0018
β_{eff} [pcm]	701.40 ± 0.88	699.53 ± 0.85	699.37 ± 0.81	700.70 ± 0.80
$\beta_{\text{eff},1}$ [pcm]	20.48 ± 0.15	20.42 ± 0.15	20.31 ± 0.14	20.65 ± 0.14
$\beta_{\text{eff},2}$ [pcm]	102.38 ± 0.34	102.60 ± 0.33	101.81 ± 0.31	102.29 ± 0.31
$\beta_{\text{eff},3}$ [pcm]	59.56 ± 0.26	59.58 ± 0.25	59.29 ± 0.24	56.56 ± 0.23
$\beta_{\text{eff},4}$ [pcm]	131.93 ± 0.38	132.05 ± 0.37	131.96 ± 0.35	132.24 ± 0.35
$\beta_{\text{eff},5}$ [pcm]	226.48 ± 0.50	225.60 ± 0.48	225.64 ± 0.46	226.17 ± 0.45
$\beta_{\text{eff},6}$ [pcm]	74.62 ± 0.29	74.04 ± 0.28	74.74 ± 0.27	74.20 ± 0.26
$\beta_{\text{eff},7}$ [pcm]	62.67 ± 0.26	62.08 ± 0.25	62.21 ± 0.24	62.24 ± 0.24
$\beta_{\text{eff},8}$ [pcm]	23.10 ± 0.16	23.17 ± 0.16	23.41 ± 0.15	23.37 ± 0.15
ρ [β]	~ 29	~ 0	~ -5	~ 1
D [cm^2]	30.6673 ± 0.0038	30.6681 ± 0.0040	30.6084 ± 0.038	30.6801 ± 0.037

Table 2.1 – Multiplication factor, adjoint-weighted kinetics parameters and migration area D for the TMI assembly in different configurations, as computed with the JEFF-3.1.1 nuclear data library. Precursors are regrouped into 8 families.

2.5.3 Evaluating the efficiency of the two population-control methods

We have tested the two population-control methods in different kinetic simulations. For their comparison, we have considered the figure of merit, defined as

$$FOM = \frac{1}{\sigma^2 T}, \quad (2.5.13)$$

where σ is the estimate of the standard error on the score of interest and T is the simulation time. For this analysis, the score of interest for the comparison of the FOM is the total neutron flux in each time bin.

As a benchmark configuration, we have selected an un-rodged assembly from the TMI-1 reactor core, whose specifications can be found in Ivanov et al. (2013b). A short description is also provided in Sec. A.4. The assembly is a realistic system, yet allowing for a reasonable computational cost. We have studied different scenarios (critical, subcritical and supercritical configurations) in order to evaluate the impact of the population-control methods depending on the state of the system.

The multiplication factor and the adjoint-weighted kinetics parameters for the benchmark have been obtained by running a regular power iteration calculation with 50 inactive cycles and 21000 active cycles, each with 8×10^4 neutrons. The configuration is adjusted to be critical by rescaling the mean number of neutrons produced by fission, ν , by the multiplication factor k_{eff} obtained by the previous criticality calculation, as described in Sec. 2.3.4. Results for the nominal (before rescaling ν) and critical configurations (after rescaling ν) are displayed in the second and third columns of Table 2.1. The decay constants are displayed in Table 2.2. Nuclear data for this configuration are taken from the JEFF-3.1.1 library (Santamarina et al., 2009).

For the kinetic simulations, results were averaged over 2000 batches of 8000 particles.

Critical configuration

For a first comparison of the two population-control methods, we have prepared the assembly on the critical state described in Table 2.1 (critical configuration) with the power iteration al-

λ_1	0.01246670
λ_2	0.02829170
λ_3	0.04252440
λ_4	0.1330420
λ_5	0.2924672
λ_6	0.6664877
λ_7	1.634781
λ_8	3.554600
$\bar{\lambda}$	0.08165179

Table 2.2 – Decay constants, in s^{-1} , as given by the JEFF-3.1.1 nuclear data library. Precursors are regrouped into 8 families. The average β -weighted decay constant $\bar{\lambda}$ [s^{-1}] is also provided. Statistical fluctuations on $\bar{\lambda}$ are negligible.

gorithm. Then we have followed the kinetic evolution of the neutron flux over 10 s with 100 regularly spaced intervals by increments of $\Delta t = 0.1$ s. Population control is applied at the end of each time interval, using either *Russian roulette and splitting* or *combing*. We have also performed the simulation without any population control. Since the interval between population control events is longer than the average fission chain length, essentially all prompt neutrons chains die within a time step. Therefore, the simulation mostly evaluates the efficiency of population-control methods on the precursor population.

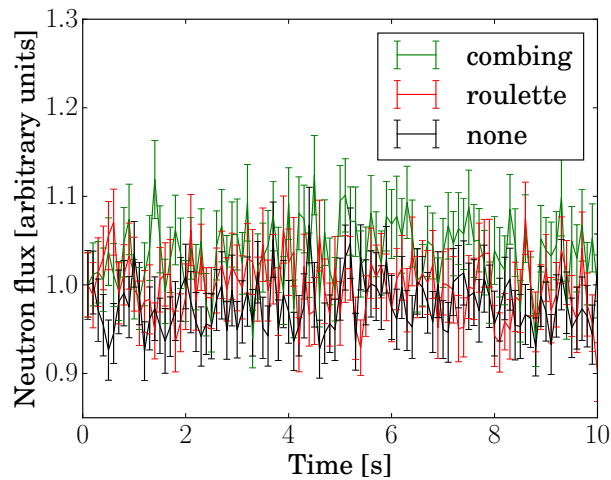
The ratios between the FOM with and without population control are shown in Fig. 2.7 and are close to 1 (note that the computational time of each simulation was about 800 CPU hours). For this configuration, population control is not necessary because the variations on the precursor population size are small. In principle, population control is necessary even for critical configurations, because of the “critical catastrophe” (De Mulatier et al., 2015). However, in our example, on a short time scale, the simulation does not suffer from fluctuations on the population size.

When looking at the standard error on the total precursor weight, *Russian roulette and splitting* leads to an increased variance as compared to the simulation without population control. It is interesting to notice that the large variance on the precursors weight does not affect the FOM. This can be explained by the fact that the roulette, as a population-control method, is actually applied on precursors and not on neutrons. Thus, the roulette induces direct fluctuations on the precursor population, but not on the neutron population.

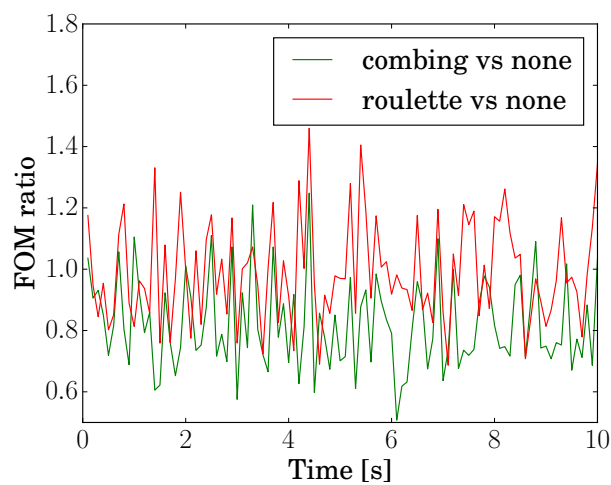
The same Figure shows steps on precursor weights, which are probably due to the decay time of the different precursor families. Indeed, precursors are rouletted on the basis of the expected delayed neutron weight, given by Eq. (2.4.2), which has an exponential decay ruled by the decay times λ_j . When the neutron weight drops below the roulette threshold, the precursor get rouletted. Hence, the frequency at which precursors get rouletted is directly related to the families decay time.

Subcritical configuration

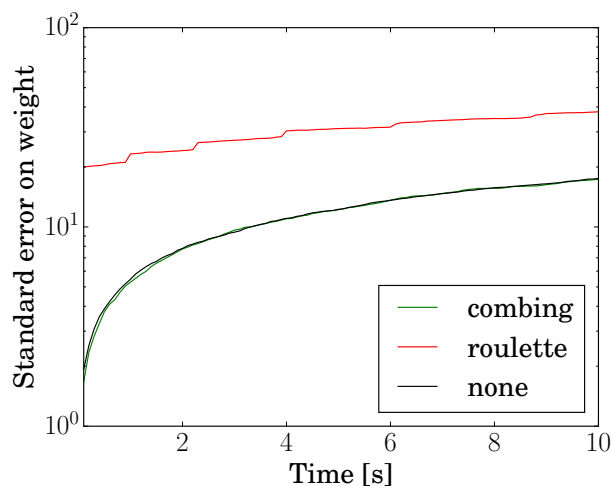
In order to emphasize the differences between the two methods, we have then studied a subcritical configuration so as to capture larger variations of the precursor population size. As with the critical configuration, the problem has been prepared in a critical state with the power iteration



Neutron flux



FOM ratio



Precursors weight uncertainty

Figure 2.7 – Time evolution of the total neutron flux in the TMI assembly. The source is prepared in a critical state. Population control is performed every $\Delta t = 0.1$ s. Green: combing. Red: Russian roulette and splitting. Black: no population control. Top: total neutron flux. Middle: FOM ratios with respect to the calculation without population control. Bottom: absolute uncertainty on the total precursor weight.

algorithm, and then, at the beginning of the kinetic simulation (i.e., at $t = 0$ s), we have increased the boron concentration in the moderator. The extra 500 ppm of boron introduce about -5 \$ in the system: see Table 2.1 (subcritical configuration). As above, the flux behaviour is monitored over 10 s partitioned into 100 regularly spaced intervals by increments of $\Delta t = 0.1$ s. At the end of each time interval, *Russian roulette and splitting* or combing is used as the population-control method. Similarly as in the previous configuration, the simulation mostly evaluates the efficiency of population-control methods on the precursor population.

Results are shown in Fig. 2.8. The FOM ratio between Russian roulette and no population control is close to 1. However, in this configuration, combing improves the FOM by about a factor of 5. Regarding the uncertainty on the total precursor weight, the steps for the Russian roulette appear at the same times as for the previous configuration.

Supercritical configuration

We have finally considered a supercritical configuration, for which we have prepared the critical state and at the beginning of the kinetic simulation (i.e., at $t=0$ s), we have reduced the boron concentration. The extracted 100 ppm of boron introduced about 1 \$ in the system: see Table 2.1 (supercritical configuration). Since the variations of the systems are fast, we apply population control more often than for the previous configuration. The flux behaviour is monitored over 0.1 s being partitioned into 100 regularly spaced intervals by increments of $\Delta t=1$ ms.

Results are shown in Fig. 2.9. For this supercritical configuration observed over small time steps, the FOM ratio between *Russian roulette and splitting* and no population control is about 1, even if combing seems to become more and more efficient as the population size increases.

Conclusion

In conclusion, we have found that combing is more efficient than the Russian roulette and splitting as a population-control method, especially when the population size varies dramatically.

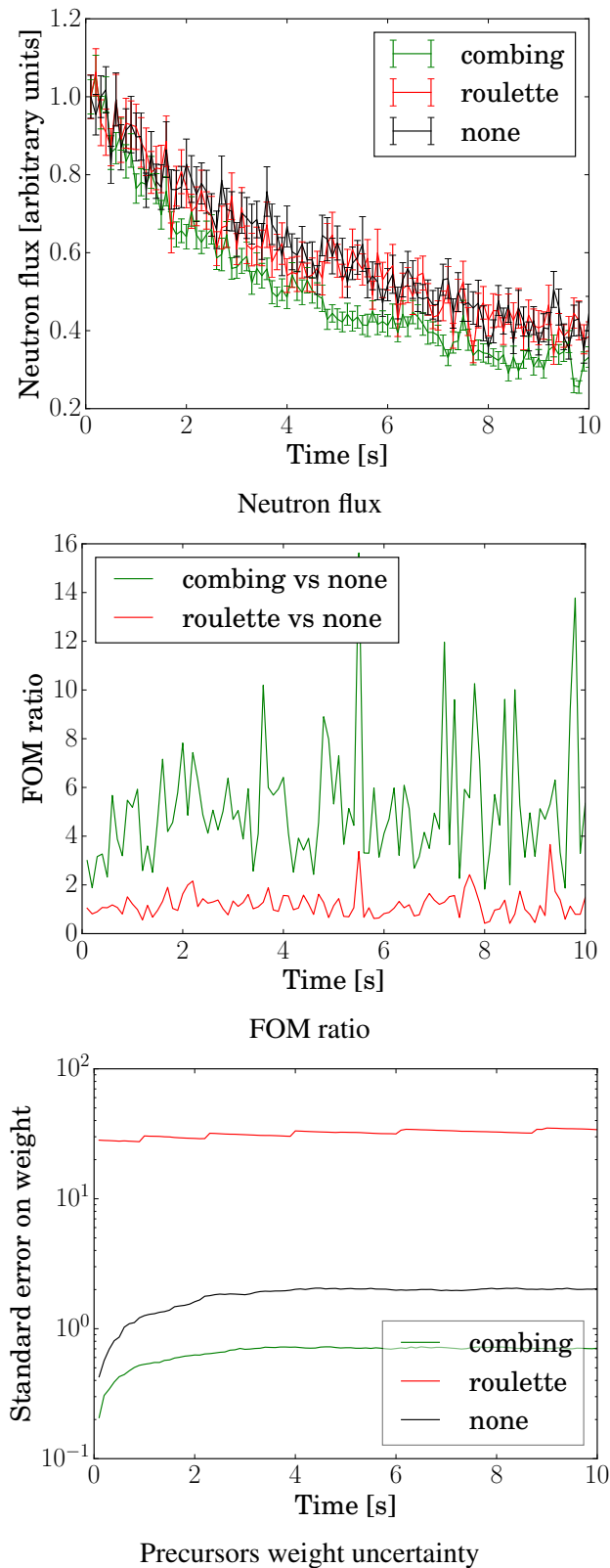


Figure 2.8 – Time evolution of the total neutron flux in the TMI assembly. The source is prepared in a critical state. At the beginning of the kinetic simulation, a reactivity of $-5\ \$$ is introduced. Population control is performed every $\Delta t = 0.1$ s. Green: combing. Red: Russian roulette and splitting. Black: no population control. Top: total neutron flux. Middle: FOM ratios with respect to the calculation without population control. Bottom: absolute uncertainty on the total precursor weight.

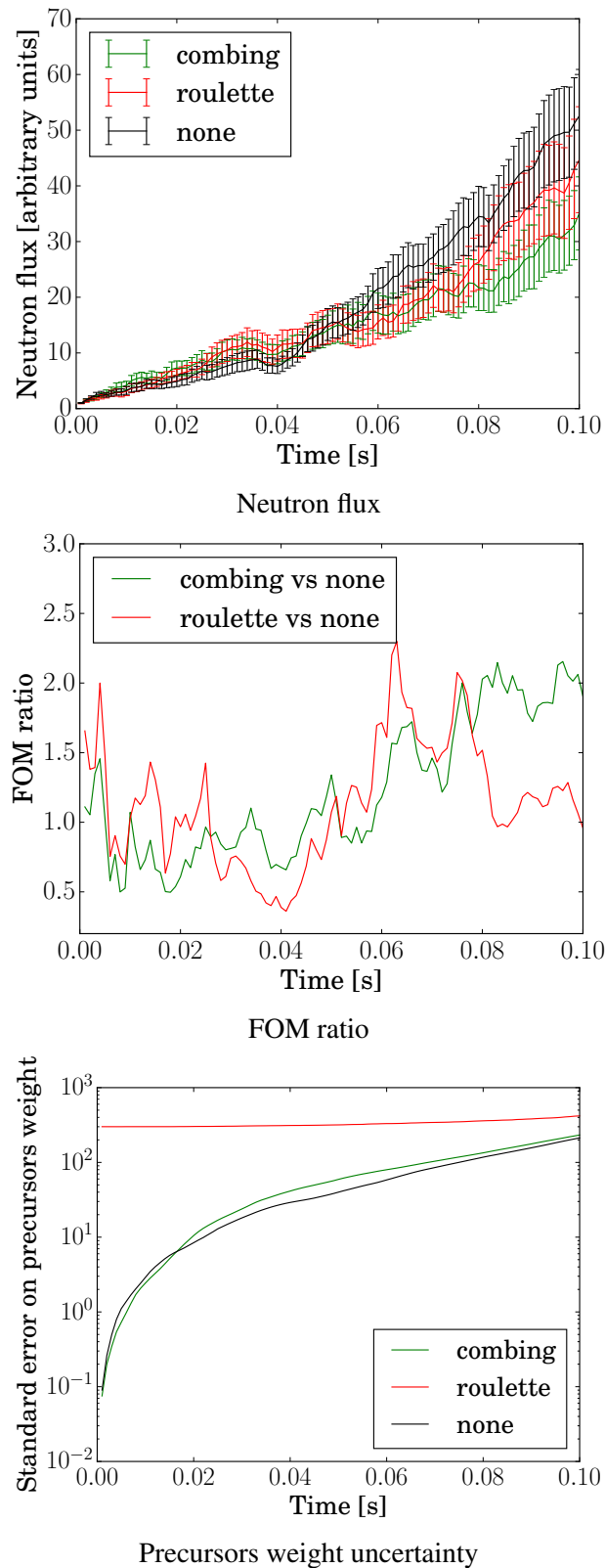


Figure 2.9 – Time evolution of the total neutron flux in the TMI assembly. The source is prepared in a critical state. At the beginning of the kinetic simulation, we reduce the boron concentration, introducing 1 \$ reactivity. Population control is performed every $\Delta t=1$ ms. Green: combing. Red: Russian roulette and splitting. Black: no population control. Top: total neutron flux. Middle: FOM ratios with respect to the calculation without population control. Bottom: absolute uncertainty on the total precursor weight.

2.6 Branchless collisions

2.6.1 Description of the algorithm

The fission chains in Monte Carlo transport are responsible for increased variance and spatial correlations in multiplying media with respect to the case of purely diffusing and absorbing configurations (Zoja et al., 2014b; Dumonteil et al., 2014; De Mulatier et al., 2015). These fluctuations, whose origin lies in the fission-induced spatial correlations between successive generations, might globally hinder the convergence of the kinetic Monte Carlo calculations. In TRIPOLI-4 we have chosen to cope with the variance associated to fission chains by implementing the *branchless* collisions method (Lux and Koblinger, 1991). The idea behind this algorithm is to suppress the variability due to the simultaneous propagation of the several branches associated to a fission event (i.e., the histories of all the neutrons descending from a common ancestor) and to collapse all the contributions into a single history carrying the average weight of all the descendants. In principle, the branchless transport has been shown to be quite effective in reducing the variance in multiplying systems (Sjenitzer and Hoogenboom, 2013).

The branchless method uses analog scattering combined with forced fission so that after each collision, the neutron is either a scattering neutron or a fission neutron. In order to impose that exactly one particle emerge from each collision, even after fission, the following statistical weight correction is applied:

$$w' = w \frac{\nu_p \Sigma_f + \Sigma_s}{\Sigma_t}. \quad (2.6.1)$$

Note that in a non-multiplying medium, the branchless method behaves as implicit capture.

The probability that the collision corresponds to scattering is

$$P_{\text{scattering}} = \frac{\Sigma_s}{\nu_p \Sigma_f + \Sigma_s}, \quad (2.6.2)$$

and therefore we choose the final state of the neutron from the scattering kernel. The complementary probability that the collision corresponds to fission is

$$P_{\text{fission}} = \frac{\nu_p \Sigma_f}{\nu_p \Sigma_f + \Sigma_s}, \quad (2.6.3)$$

and therefore we choose the final state of the neutron from the fission kernel.

2.6.2 Evaluating the efficiency of the method

The branchless method produces exactly one particle after each collision, by applying a correction weight defined by Eq. (2.6.1). If the correction weight is larger than 1, the exiting particle might get split. In that case, the benefit of the branchless method would be lost because of the creation of correlated particles. For a critical system, however, the weight correction is on average close to 1. In this case, branchless should be efficient and produce only one particle after a collision. We have selected a critical and a supercritical configuration to study the efficiency of the method.

Critical configuration

We have prepared again the TMI assembly on the critical state with the power iteration algorithm. Then we have followed the kinetic evolution of the neutron flux over 10 s. The use of the branchless method improves the FOM by about a factor of 6, as shown in Fig. 2.10.

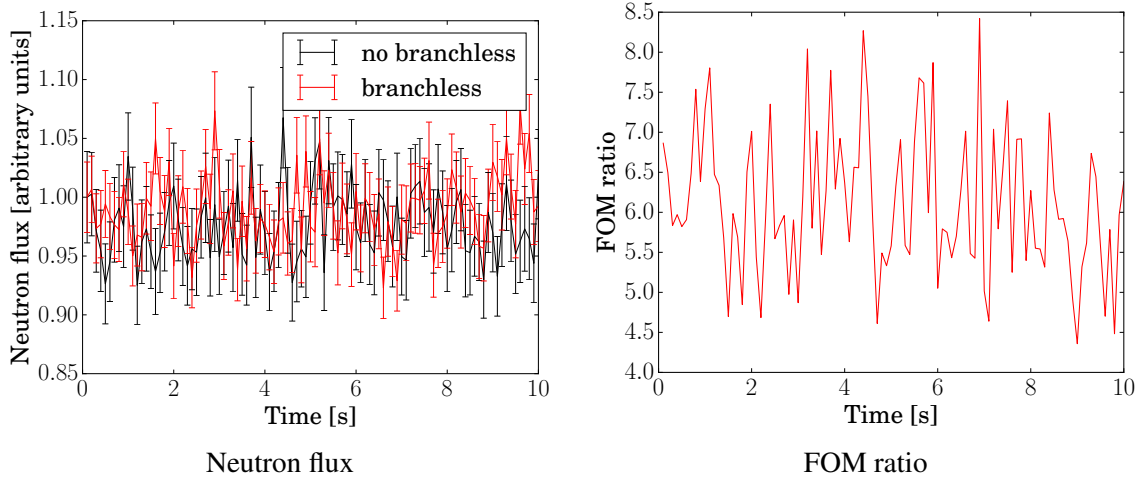


Figure 2.10 – Time evolution of the total neutron flux in TMI assembly with and without branchless method. The source is prepared in a critical state. Left: Total neutron flux. Right: FOM ratio between simulations with/without branchless collisions.

Supercritical configuration

We have then considered the supercritical configuration introduced in Sec. 2.5.3, with the extracted 100 ppm of boron introducing about 1 \$ in the system. The flux behaviour is scored over 0.1 s being partitioned into 100 regularly spaced intervals by increments of $\Delta t = 1$ ms. Results are shown in Fig. 2.11. The difference between the kinetic evolution resulting from the two simulations might look surprising, but there is actually no bias in either simulation. The time steps are strongly correlated (particles are transferred from one step to the next one) and large fluctuations in one time step directly impact the following time steps: the interpretation of kinetic results is much more challenging than for stationary calculations. One solution would be to increase the statistics to reduce fluctuations. In the time interval between $t = 0$ s and $t = 0.06$ s, where the flux from the two simulations are in reasonable agreement, the use of branchless collisions improves the FOM by about a factor of 4. Beyond this time limit, it is not possible to draw any conclusions on the efficiency of the branchless method. Correlations between time steps will be investigated in Sec. 3.3.

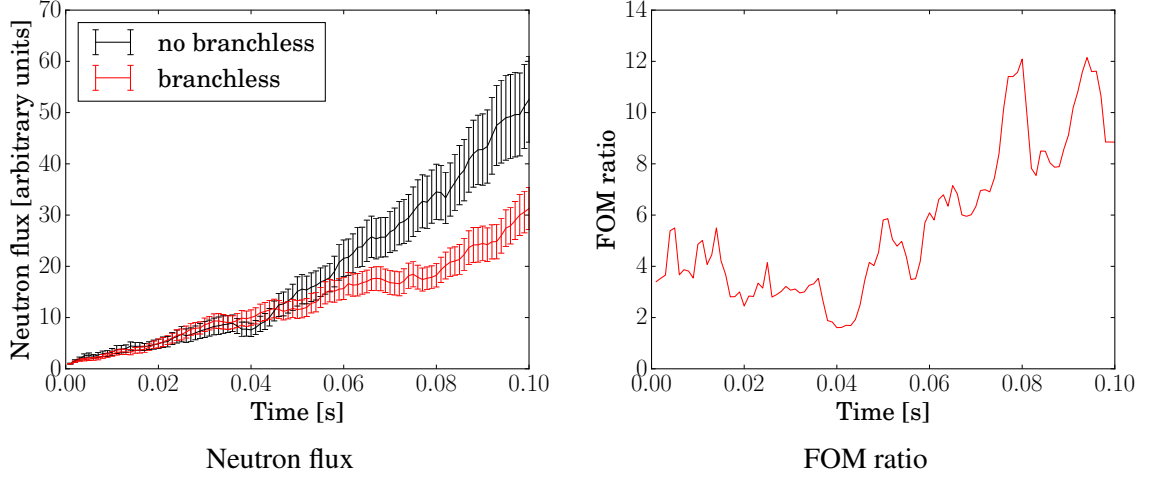


Figure 2.11 – Time evolution of the total neutron flux in TMI assembly with and without branchless method. The source is prepared in a critical state. At the beginning of the kinetic simulation, a reactivity of 1 \$ is introduced. Left: Total neutron flux. Right: FOM ratio between simulations with/without branchless collisions.

2.7 Development of a population importance method for fast kinetic configurations

2.7.1 Description of the algorithm

The kinetic simulation of fast transients requires a finely discretized time mesh in order to capture the rapid time variations of the population. However, if the simulation time steps are too small, ordinary roulette will kill most of the delayed neutrons, even when forced decay is applied. Indeed, the statistical weight of delayed neutrons is proportional to the time interval $t_{q+1} - t_q$, as given by Eq. (2.4.2), which is rewritten here for the convenience of the reader:

$$\mathcal{W}_{delayed}(t|t_0) = (t_{q+1} - t_q) \sum_j \lambda_j \frac{\beta_j}{\beta} e^{-\lambda_j(t-t_0)} w_C.$$

Thus, for small time intervals, $\mathcal{W}_{delayed}$ is also small and particles will not survive the roulette. With no delayed neutrons to initiate new fission chains, the simulation will then suffer from severe under-sampling issues.

In order to increase the number of simulated neutrons and therefore improve the efficiency of simulations over short time scales, we have thus implemented a new time-dependent importance sampling scheme in the development version of TRIPOLI-4. The key principle of importance sampling methods is to modify the particle weights, by preserving the physical weights, in such a way that

$$W = W' \times I, \quad (2.7.1)$$

with W the initial weight, W' the new weight and I the (positive) importance factor. For our method, one single importance value is associated to each species (i.e., population), meaning that all neutrons share the same importance value, and the same applies for precursors.

In order to observe both fast and slow transients, one needs a non-uniform simulation time grid and importance factors. On one hand, fast transients can exclusively be observed over very short times (time steps might be of the order of the mean generation time); on the other hand,

delayed neutrons need a few seconds to be emitted. Hence, short time grids should be associated with the application of an importance factor to favour the forced decay of delayed neutrons. On the contrary, importance is not necessary for widely spaced time grids, since delayed neutrons will be typically emitted with a large statistical weight. Therefore, we have implemented our method in such a way that the importance values can be modified along the simulation time steps, depending on the size of each step.

For each time step i , the user may provide an importance ratio $R_i = I_{N,i}/I_{C,i}$. At the beginning of a time step, including the first one, TRIPOLI-4 uses the current particle weights to define a global neutron importance $I_{N,i}$ and a global precursor importance $I_{C,i}$, in such a way that the physical weights are preserved:

$$\begin{aligned} W_{N,i}I_{N,i} &= W_{N,i+1}I_{N,i+1}, \\ W_{C,i}I_{C,i} &= W_{C,i+1}I_{C,i+1}, \\ W_{N,i} + W_{C,i} &= W_{N,i+1} + W_{C,i+1}, \end{aligned}$$

with $W_{N,i}$ the total neutron population weight and $W_{C,i}$ the total precursor population weight at the end of time step i . The meaning of the importance factor $I_{N,i}$ ($I_{C,i}$) is that a simulated neutron (precursor) actually represents $I_{N,i}$ ($I_{C,i}$) physical neutrons (precursors). Note that the population weights are only adjusted if the importance ratio changes (nothing needs to be done if $R_{i+1} = R_i$).

When neutrons produce precursors or precursors decay into neutrons, particle weights need to be adjusted to take into account the population importance factors: the weight of a delayed neutron created by a precursor decay is divided by R_i ; similarly, when a neutron creates a precursor through fission, the precursor weight is multiplied by R_i . Finally, in order to keep the result unbiased, neutron (precursor) scores are multiplied by the importance factor $I_{N,i}$ ($I_{C,i}$).

Figure 2.12 illustrates how the importance is applied along the simulation with realistic simulation data. For simplicity, only one time step is considered, so we drop index i . At the end of the critical source calculation, the sampled total neutron weight is much lower than the total precursor weight ($W_N = 10^{-1}$ and $W_C = 10^3$), leading to a very unbalanced ratio: $W_C/W_N = 10^4$. The importance method is applied with a factor $R = 10^{-2}$. With this factor, the global neutron and precursor importance values can be computed:

$$I_N = X_N + R \times X_C, \tag{2.7.2}$$

$$I_C = X_C + \frac{1}{R} \times X_N, \tag{2.7.3}$$

with

$$X_N = \frac{W_N}{W_N + W_C}, \tag{2.7.4}$$

$$X_C = \frac{W_C}{W_N + W_C}. \tag{2.7.5}$$

Therefore, we obtain $I_N \approx 10^{-2}$ and $I_C \approx 1$. All neutrons weights are divided by I_N , making the population ratio more balanced: $W'_C/W'_N = 10^2$. Then, during the kinetic simulation, when a neutron produces a precursor, the precursor weight is multiplied by R ; conversely, when a precursor decays into a delayed neutron, the neutron weight is divided by R . Thus, delayed neutron weights are increased by a factor 10^2 and are more likely to survive the Russian roulette. Finally, during the scoring phase, all neutron scores are multiplied by I_N .

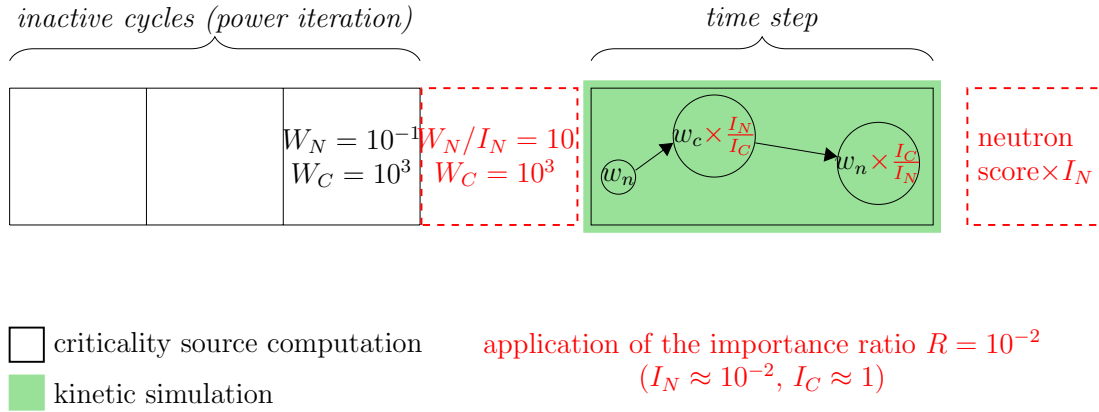


Figure 2.12 – Schematic representation of the importance method during a kinetic simulation with one time step. The method operates at three different steps, represented in red: at the end of the critical source calculation, within the time steps and during the scoring phase.

2.7.2 Evaluating the efficiency of the method

As a first verification test for this method, we have performed a kinetic simulation of Flattop-Pu (PU-MET-FAST-006) (OECD Nuclear Energy Agency, 1995), which is a benchmark configuration consisting of a plutonium sphere with a natural uranium reflector. For a description of the system, we refer the reader to Sec. A.2. Nuclear data for this configuration are taken from the ENDF/B-VII library (Chadwick et al., 2006). Flattop-Pu is a fast system, characterized by a small delayed neutron fraction, as compared to PWRs ($\beta_{\text{eff}} \approx 279$ pcm) and also a very short generation time ($\Lambda_{\text{eff}} \approx 13$ ns). Because of the specificity of the Flattop-Pu kinetics parameters, the delayed neutron undersampling issue is amplified, and Flattop-Pu is an interesting configuration to assess our importance sampling method.

After the system has been prepared on the critical state, its kinetic evolution is followed up to 1 ms with an observation time partitioned into 50 regularly spaced intervals of $\Delta t = 20 \mu\text{s}$. Population control is enforced by performing combing at the end of each time step. Since the time step is short compared to precursors decay times (and very long as compared to the average fission chain length), the use of the importance algorithm is necessary to favour neutron production from precursor decay. The total neutron flux displayed in Fig. 2.13 was computed using the track length estimator with (red curve) and without (blue curve) the importance algorithm. CPU times were similar for the two calculations and of the order of 1300 CPU hours. The use of a constant population importance ratio $R = 10^{-4}$ produces a gain of figure of merit of about 63. The flux does not exhibit any drift, as expected since $k_{\text{eff}} \approx 1$, while the behaviour of the flux computed without importance is difficult to interpret because of the large statistical uncertainty on the results. An example of a kinetic simulation with varying time steps and importance factors will be shown in Chapter 3.

2.7.3 Choice of the optimal importance ratio

Intuitively, it is easy to understand that at some point decreasing the neutron importance can not indefinitely improve the FOM. Indeed, neutrons will end up originating mostly from splitting and thus will be strongly correlated. Hence, there must exist an optimal value for the importance ratio.

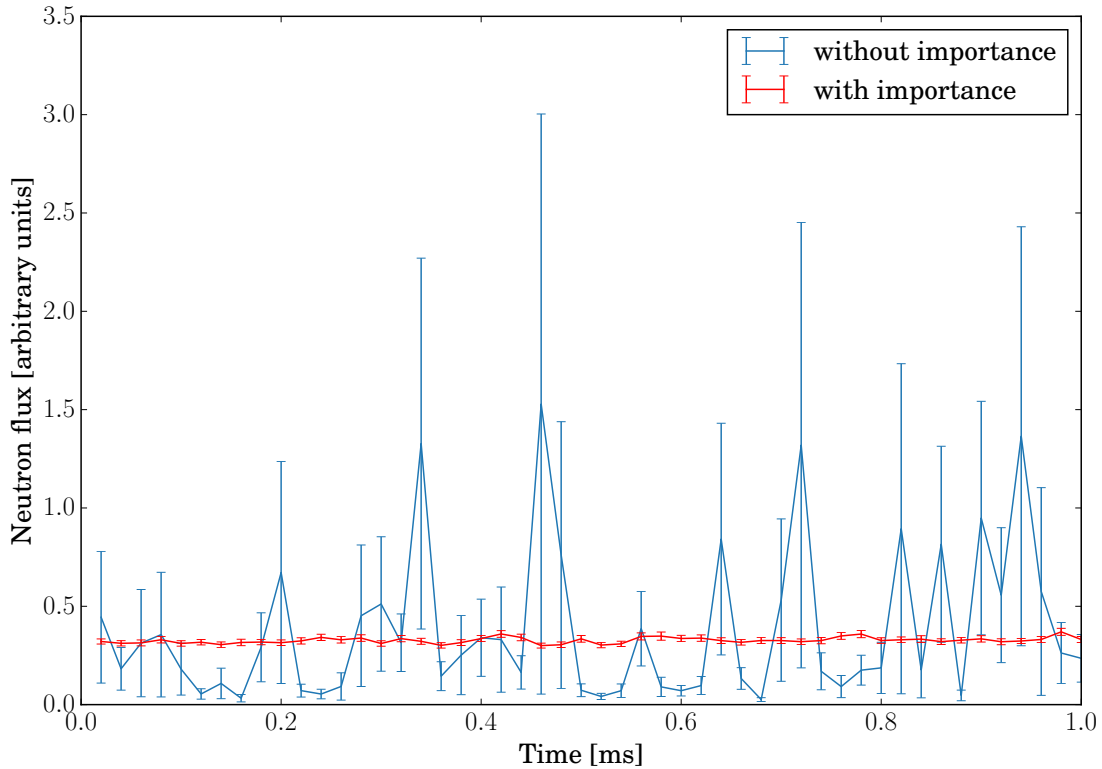


Figure 2.13 – Time evolution of the total neutron flux in Flattop-Pu at its critical state computed with TRIPOLI-4. The system is prepared on the critical state and its kinetic evolution is followed up to 1 ms. Blue curve: simulation without importance factor. Orange curve: simulation with a population importance ratio of $R = 10^{-4}$. The two calculations were performed with similar CPU times. The use of an importance enables to improve the figure of merit by a factor 63.

The choice of the importance ratio depends on a set of parameters, including the simulation time grid, the inserted reactivity and the kinetics parameters. We attempt here to characterize the optimal value, or at least its dependence on the kinetics parameters of the system and the time step length. To this end, we have performed several simulations on the TMI-1 assembly with different values of the importance ratio. We have chosen ratios starting from 1 (i.e., no importance sampling applied) to 10^{-7} . For each ratio, the number of particles and batches is adapted so that the estimated standard error on the total neutron flux is below 10%. The flux is collected over three time intervals having different sizes (0.001 s, 0.01 s and 0.1 s), so as to assess the impact of the time step size on the efficiency of the importance method.

Figure 2.14 shows the integral of the total neutron flux over each scoring time bin, with the associated standard error, for the different ratios. In all cases, scores are well converged on a common value. Hence, it makes sense to compare the associated FOM.

The FOM are presented in Fig. 2.15 for the different simulations, after normalization to the FOM obtained when no importance sampling is applied. For the three time steps, an optimal plateau is reached, which suggests that there exists a wide optimal range for the choice of the importance ratio. At smaller values of the importance ratio, the FOM decreases: if the neutron importance is too small, the time spent in the simulation of extra neutrons worsens the FOM because the extra sampled neutrons will mostly originate from splitting and thus will be strongly correlated. Another observation is that the importance method becomes more efficient as the

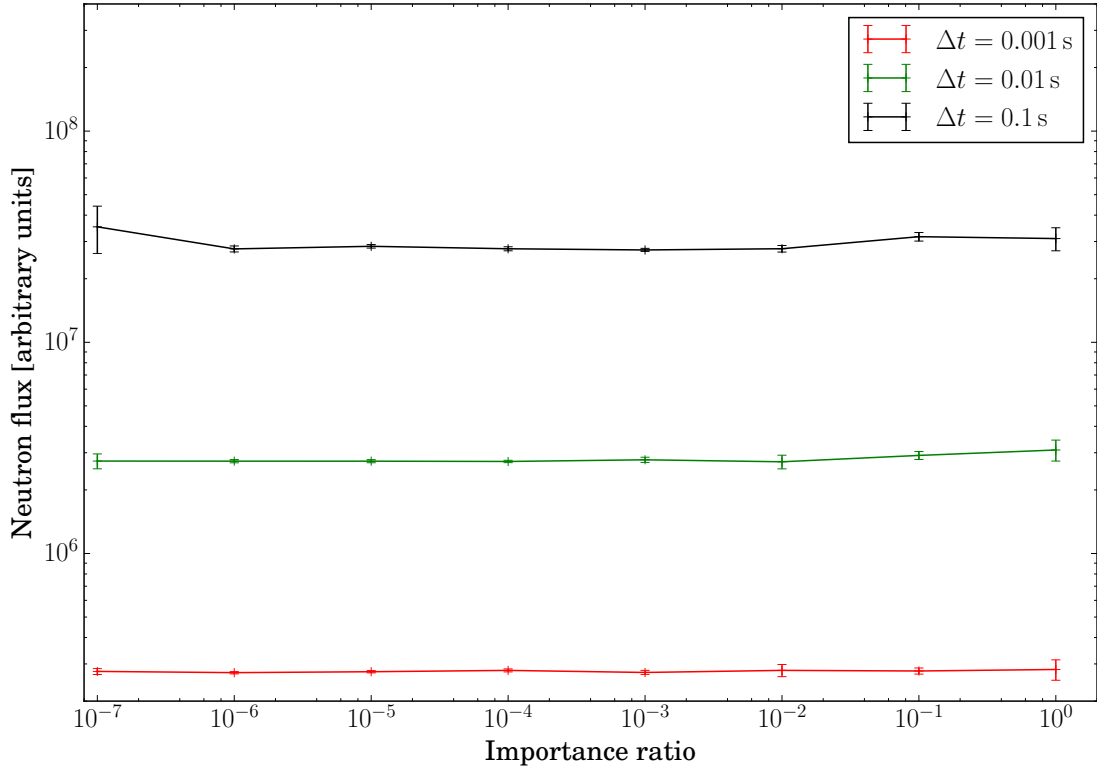


Figure 2.14 – Integral of the total neutron flux over the scoring time bin, with their standard error for different importance ratios and sizes of the scoring time bin.

time step gets smaller. Indeed, for $\Delta t = 0.001$ s, the optimal FOM gain is about 110, while it is only 19 for $\Delta t = 0.01$ s and 5 for $\Delta t = 0.1$ s. In fact, the smaller the time step, the less efficient forced decay becomes at creating delayed neutrons with weights large enough to survive the Russian roulette. The importance method compensates the lack of delayed neutrons with more neutrons at the beginning of the kinetic calculation, and more neutrons created within the time bins. The upper limit of the plateau decreases when the time step decreases because stronger importance sampling is needed to cope with the lack of delayed neutrons.

In conclusion, the gain of figure of merit strongly depends on the size of the time bin: the smaller the time step, the higher the gain. On the contrary, the optimal population importance ratio does not depend much on the size of the time bin. Indeed, for the different sizes that we have studied, there exists a wide optimal range for the choice of the importance ratio, and 10^{-3} seems to be a good generic choice for a system with kinetics parameters similar to that of TMI-1 (which are typical of a PWR).

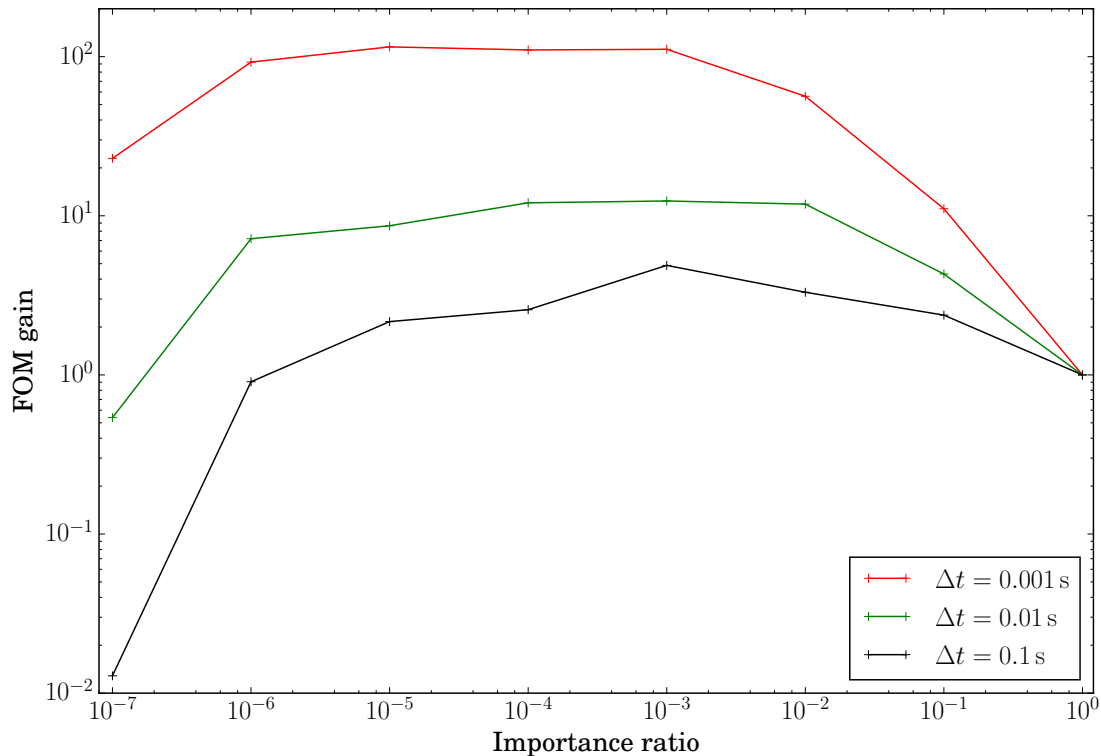


Figure 2.15 – Gains of figures of merit with respect to the calculations without importance sampling, as a function of the importance ratios.

2.8 Development of the capability to handle time-dependent geometry

In order to simulate transients with moving reactor parts, in the frame of this thesis we have implemented a new option in the development version of TRIPOLI-4 that makes it possible to modify the geometry and material compositions at each time step. This new option, which for the moment is only available for ROOT geometries, paves the way for all types of time-dependent simulations: different geometries can be defined for the criticality calculation and for each time step of the kinetic simulation. At the beginning of each step of the simulation time grid, the geometry of the previous step is replaced by the new one. As long as the geometries have the same bounding box, there is no constraint on their internal structure (number of volumes, composition of volumes, etc.). Note that the change on the geometry is only discrete: for now, TRIPOLI-4 does not handle continuous changes. Various examples will be presented through this work.

It should be noted that this development contains a limitation regarding moving volumes containing fissile material. Indeed, precursors are created in the fissile materials. If the geometry change concerns one of these materials, and since precursors do not move, they might be later found in non-fissile materials. To avoid that, strategies could be developed, such as shifting the precursor position by the same distance as the fuel shift. Anyway, this problem seems to deserve a specific strategy for each configuration depending on the location of the change in the geometry.

With this implementation of the time-dependent geometry, every change in the geometry

requires to define a new geometry as input. Therefore, the memory use grows as the number of the geometries. In order to reduce the memory footprint, we have done some cleaning in the TRIPOLI-4 classes in charge of the geometry. Notwithstanding this optimization work, the memory requirements for handling each geometry in TRIPOLI-4 are still large. Moreover, most of the time, the changes in the geometry will be slight: e.g., the control rod insertion depth varies but the number of volumes and the composition do not change. Considering such slight changes, the implementation could be improved in order to reduce the memory occupation: for example, one single geometry could be given as input, and modified along the simulation depending on a varying parameter (e.g., the rod insertion depth).

2.9 Conclusion

We have detailed the algorithms that are necessary to pave the way for Monte Carlo kinetic simulations with TRIPOLI-4, including the algorithms that were recently proposed in the literature and our original contributions.

In order to assess the efficiency of the different methods, we have considered an assembly based on the TMI-1 reactor, in different configurations. We have found that combing is more efficient than the *Russian roulette and splitting* as a population-control method, especially when the population size varies dramatically; the branchless collisions method has proved to be effective in reducing the variance.

Our first new contribution to kinetic algorithms in TRIPOLI-4 was also detailed: a time-dependent population importance sampling scheme for variance reduction. For the assessment of its efficiency, we have again considered the assembly based on the TMI-1 reactor, in different configurations. The importance method has been tested and proved to be very efficient for simulations with small time steps, where forced decay fails to produce delayed neutrons with weights large enough to survive the Russian roulette. The choice of an optimal importance ratio has been also examined: as the time step decreases, the optimal importance ratio also decreases. The figure of merit reaches an optimal plateau, and is thus weakly dependent on the choice of the ratio.

We have finally detailed another contribution to extending TRIPOLI-4 capabilities: time-dependent geometries are now handled, which makes it possible to simulate all types of transients.

The description of the different kinetic methods was published in Faucher et al. (2018), and the evaluation of the variance-reduction and population-control techniques was published in Faucher et al. (2019a).

Chapter 3

Extensive tests of the kinetic methods

Kinetic capabilities have been implemented in TRIPOLI-4 as a first step towards the simulation of reactor transients with physical feedbacks due to thermal-hydraulics and thermomechanics couplings. The algorithms that have been chosen are those described in Chapter 2. A few relevant verification tests on two complex systems are presented in this chapter in order to illustrate the capabilities of the extended code. Correlations between time steps, as well as the impact of the spatial discretization of the kinetic scores, are also investigated.

3.1 Verification tests on SPERT III E-core

3.1.1 Presentation of SPERT III E-core

The Special Power Excursion Reactor Test III (SPERT-III) is a small pressurized-water research reactor. It was built in the United States in the 1960s (Heffner and Wilson, 1961; Houghtaling et al., 1965; McCardell et al., 1969) in order to investigate transient behaviour of nuclear reactors. The E-core type consists of a pressurized light-water-moderated core with 4.8%-enriched UO_2 fuel pellets arranged in a regular lattice of cylindrical pins. It contains 60 assemblies, including 48 fuel assemblies with 25 (5x5) pin-cells, 4 assemblies with 16 (4x4) pin-cells, and 8 control rods moving pairwise. The control rods contain fuel in the lower section and a neutron absorber (18-8 stainless steel with 1.35 weight percent of ^{10}B) in the upper section. In our model, as described in the following, we will move the control rods in order to increase or decrease the core reactivity. The full insertion of the control rods into the core corresponds to the shutdown configuration. At the center of the core, a transient cruciform rod is made of the same neutron absorber as in the control rods in the lower part and of 18-8 stainless steel in the upper part. A detailed description of the system is provided in Sec. A.3. In the experimental campaign that was carried out in the 1960s, the transient rod was rapidly ejected from the core to initiate power excursions: the lower absorber part is located outside the core during the transient (Heffner and Wilson, 1961; Houghtaling et al., 1965; McCardell et al., 1969). The results of the reactivity insertion experiments carried out in the SPERT III E-core are extensively documented in McCardell et al. (1969).

In recent years, several calculations have been performed on the SPERT III E-core configurations in order to validate dynamic reactor simulations in steady and transient states. Most were carried out with deterministic codes, coupled with thermal-hydraulics codes, which however suffer from various approximations (Kosaka et al., 1988; Ikeda and Takeda, 2001; Aoki et al., 2009; Grandi and Moberg, 2012; Wang et al., 2013; Yamaji et al., 2014; Grandi, 2014). More recently, Monte Carlo models for MCNP (Olson, 2013a,b), KENO (Cao et al., 2015), TRIPOLI-4 (Zoia and Brun, 2016) and Serpent 2 (Knebel et al., 2016; Levinsky et al., 2019)

have been also proposed, and specifications for an international benchmark have been collected at IAEA (Olson, 2013a,b,c; IAEA, 2015).

The geometry of SPERT III E-core has been modelled for TRIPOLI-4 with ROOT according to the technical description found in Heffner and Wilson (1961); Dugone (1965); Houghtaling et al. (1965). A radial and two axial views of the model are illustrated in Fig. 3.1. The first axial view shows the control rods at their critical height, while the second one shows the shutdown configuration with the control rods fully inserted into the core. The validation of the model for static conditions was detailed by Zoia and Brun (2016). The kinetic methods implemented in TRIPOLI-4 were tested on this model (Faucher et al., 2018).

3.1.2 Preliminary criticality calculations

For the purpose of the investigations carried out in this thesis, the multiplication factor and the adjoint-weighted kinetics parameters for the cold zero power configuration have been obtained by running a regular power iteration calculation with 500 inactive cycles and 2500 active cycles, each with 4×10^5 neutrons. Results for the nominally critical configuration are displayed in Table 3.1 (nominal configuration). Nuclear data for this configuration are taken from the JEFF-3.1.1 library (Santamarina et al., 2009), for which the decay constants were already displayed in Table 2.2. The average β -weighted decay constant $\bar{\lambda}$ (Eq. (2.3.7)) computed by TRIPOLI-4 was also provided.

Knowledge of these parameters allows estimating the asymptotic inverse reactor period via the approximated formula

$$\omega = \frac{\bar{\lambda}\rho}{\beta_{\text{eff}} - \rho}, \quad (3.1.1)$$

obtained by resorting to the point-kinetics approximation (Keepin, 1965). Because of the slight bias in the reactivity due to uncertainties in the nuclear data and in the geometrical model ($\rho \sim 0.2\%$), we get $\omega \sim 0.015 \text{ s}^{-1}$, which would lead to a deviation of approximately 15% after 10 seconds. In order to avoid such spurious drift, we have rescaled the number of fission neutrons ν by the obtained value of k_{eff} as explained in Sec. 2.3.4. The new multiplication factor and adjoint-weighted kinetics parameters are displayed in Table 3.1 (critical configuration). As expected, the rescaled TRIPOLI-4 calculation is roughly within 10 pcm of an exactly critical state. The kinetics parameters are stable between the two configurations. The value for the rescaled asymptotic period is $\omega \sim 6.3 \text{ ms}^{-1}$, leading to a negligible deviation from criticality of about 0.63% after 10 seconds.

3.1.3 Steady state

Prior to scoring the time evolution of the reactor in stationary conditions, the system has been prepared on the critical configuration according to the algorithm described in Sec. 2.3. Then, its kinetic evolution is monitored over 10 s, partitioned into 100 regularly spaced intervals by increments of $\Delta t = 0.1 \text{ s}$. At the end of each time interval, combing is used as the population control method (as for all the following simulations). The aim of the simulation is to present a first example of a kinetic Monte Carlo simulation of a realistic core at steady state observed over a few seconds with control of the population size.

The neutron flux displayed in Fig. 3.2 was computed using the track length estimator with a computational cost of about 3000 CPU hours. The total neutron flux does not exhibit any drift (orange line), as expected for a system close to the critical state. As a comparison, we also

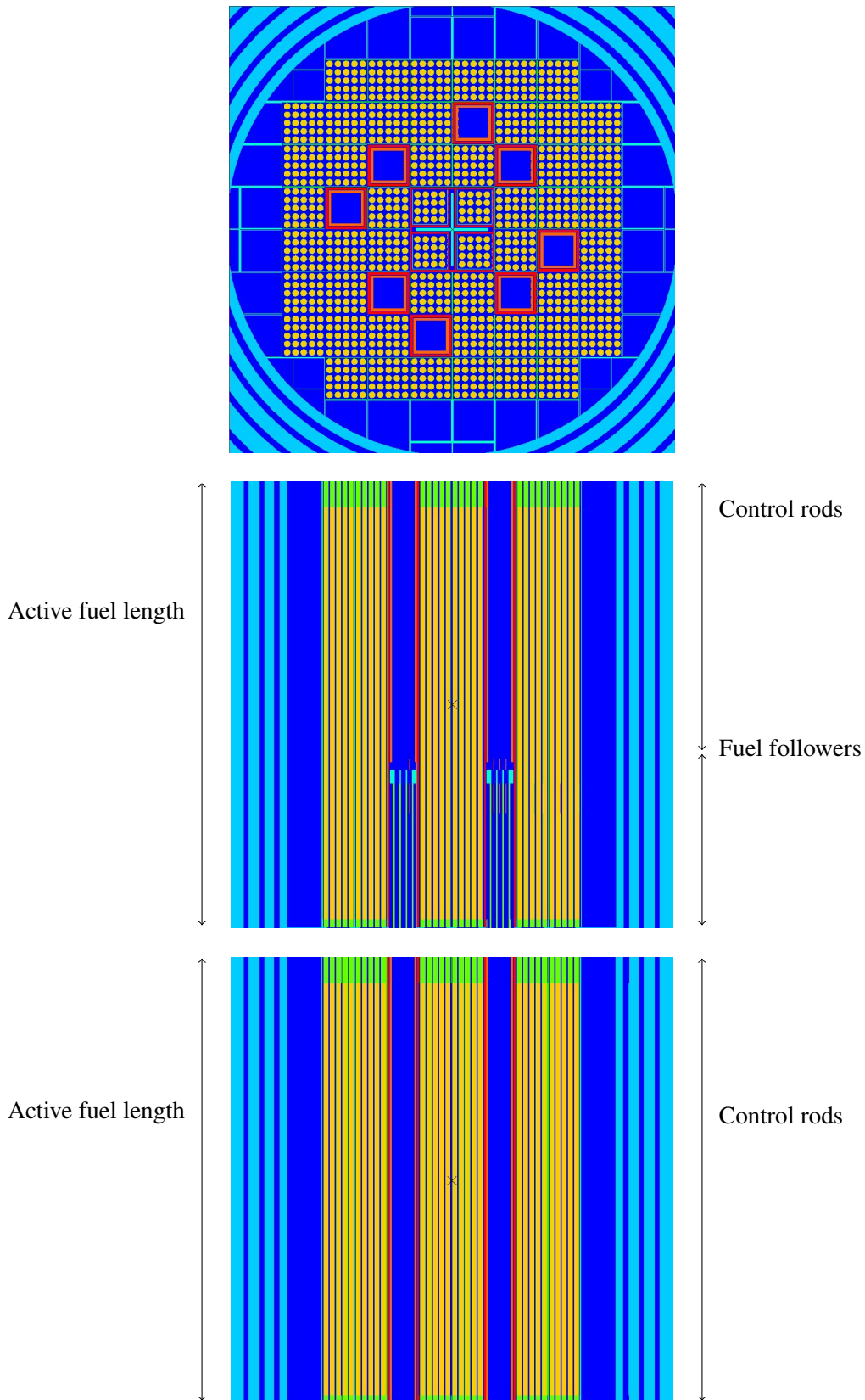


Figure 3.1 – Top: radial cut of the TRIPOLI-4 model for the operational loading of the SPERT III E-core at mid-plane. Middle: axial cut of the TRIPOLI-4 model, passing through two control rods. The lower portion of the control rod is a fuel follower, and the upper portion (orange) is the absorbing section. In between lies a flux suppressor. Bottom: axial cut of the TRIPOLI-4 model, passing through two fully inserted control rods, corresponding to the shutdown configuration.

configuration	nominal	critical	supercritical	prompt supercritical	rod drop
k_{eff}	$1.00116 \pm 1.1 \times 10^{-4}$	$0.99994 \pm 1.1 \times 10^{-4}$	$1.00388 \pm 1.1 \times 10^{-4}$	$1.01501 \pm 1.1 \times 10^{-4}$	$0.87340 \pm 1.0 \times 10^{-4}$
Λ_{eff} [μs]	17.349 ± 0.013	17.365 ± 0.012	17.2963 ± 0.0084	17.086 ± 0.013	19.897 ± 0.018
β_{eff} [pcm]	763.0 ± 4.6	765.93 ± 4.5	786.5 ± 4.5	764.9 ± 4.3	775.7 ± 4.9
$\beta_{\text{eff},1}$ [pcm]	23.73 ± 0.82	24.72 ± 0.82	22.80 ± 0.78	24.22 ± 0.82	22.70 ± 0.83
$\beta_{\text{eff},2}$ [pcm]	111.0 ± 1.8	106.7 ± 1.7	113.0 ± 1.8	108.8 ± 1.7	106.4 ± 1.8
$\beta_{\text{eff},3}$ [pcm]	67.9 ± 1.3	67.3 ± 1.3	67.7 ± 1.4	65.3 ± 1.3	67.8 ± 1.4
$\beta_{\text{eff},4}$ [pcm]	148.5 ± 2.0	145.7 ± 2.0	149.6 ± 2.1	146.8 ± 2.0	148.7 ± 2.2
$\beta_{\text{eff},5}$ [pcm]	237.0 ± 2.5	245.9 ± 2.5	251.1 ± 2.5	243.4 ± 2.6	250.6 ± 2.8
$\beta_{\text{eff},6}$ [pcm]	83.3 ± 1.5	82.7 ± 1.5	84.4 ± 1.6	80.8 ± 1.4	85.2 ± 1.6
$\beta_{\text{eff},7}$ [pcm]	66.1 ± 1.3	66.8 ± 1.3	72.0 ± 1.4	69.5 ± 1.4	68.6 ± 1.4
$\beta_{\text{eff},8}$ [pcm]	25.53 ± 0.81	26.07 ± 0.86	25.97 ± 0.84	26.06 ± 0.82	25.80 ± 0.87
ρ [pcm]	116 ± 11	-6 ± 11	386 ± 11	1479 ± 11	-14495 ± 13
ρ [$\$$]	~ 0.2	~ 0	~ 0.5	~ 2	~ -19

Table 3.1 – Multiplication factor and adjoint-weighted kinetics parameters for SPERT III E-core in different configurations, as computed with the JEFF-3.1.1 nuclear data library. Precursors are regrouped into 8 families. Error bars are not given for the reactivity expressed in dollars because we did not measure correlations between k_{eff} and β_{eff} . See the text for the precise meaning of the configurations.

present the flux computed without rescaling the number of fission neutrons (Table 3.1, nominal configuration). Without this adjustment, the flux reveals a clear drift (blue line). The impact of the extra 116 pcm on the kinetic behaviour of the system proves that the rescaling of the number of fission neutrons is essential in order to reproduce the steady state.

At the time of the simulation, we had little experience on the use of the population importance sampling method described in Sec. 2.7, and we had supposed that the time intervals were long enough to allow for the emission of delayed neutrons with sufficiently large statistical weights. Now, regarding the study conducted in Sec. 2.7.3, it appears that the variance-reduction method would be efficient in this configuration, with $\Delta t = 0.1$ s.

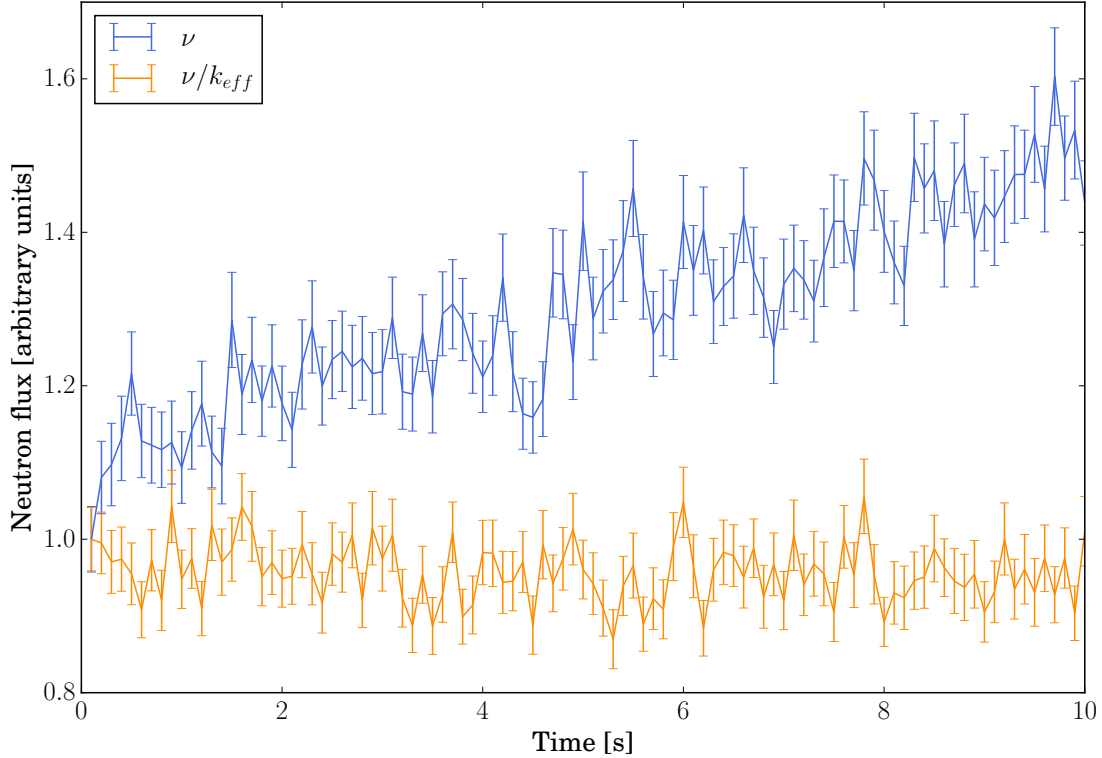


Figure 3.2 – Time evolution of the total neutron flux in SPERT III E-core computed with TRIPOLI-4 at nominal conditions, with (orange line) and without (blue line) rescaling the number of fission neutrons ν by k_{eff} .

3.1.4 Reactivity insertion

Starting from the critical configuration considered in previous Sec. 3.1.3, we have then analysed a departure from the stationary state initiated by the simultaneous movement of the 8 control rods. For this purpose, we have used the new capability of TRIPOLI-4 to handle time-dependent geometries, introduced in Sec. 2.8. At $t = 1$ s, the rods are extracted from the core by 0.8 cm, introducing a reactivity worth of approximately 0.5 \$. At $t = 6$ s, the control rods are brought back to their initial position and the system returns to a steady state (different from that of the initial stationary condition). In practice, TRIPOLI-4 takes two different geometries as inputs. The first one, corresponding to control rods at nominal height, is used for sampling the critical source, then between $t = 0$ s and $t = 1$ s and finally between $t = 6$ s and $t = 10$ s. The second one, corresponding to control rods extracted by 0.8 cm, is used between $t = 1$ s and $t = 6$ s. The kinetics parameters for both critical and supercritical configurations are reported in Table 3.1 (critical and supercritical configurations); they do not vary significantly. The simulation time grid and the scoring time grid cover the range from $t = 0$ s to $t = 10$ s by time steps of $\Delta t = 0.1$ s. The importance sampling method is not used, similarly to Sec. 3.1.3.

The total neutron flux computed during the TRIPOLI-4 simulation is displayed in Fig. 3.3. The flux remains stationary between $t = 0$ s and $t = 1$ s, until the control rods are extracted. At $t = 1$ s, a prompt jump is visible before the delayed-neutron excursion sets in. The excursion follows an exponential behaviour with a best-fit time constant of 0.25 s^{-1} . This value should be compared to the value of the delayed asymptotic reactor period, $\alpha_d = (0.426 \pm 0.019) \text{ s}^{-1}$, which we have computed by a reactor period calculation, as presented in Sec.1.2.2, and described in (Zoia et al., 2014a, 2015), with 5000 inactive cycles and 20000 active cycles of 10^4 neutrons. The time constant of kinetics is smaller than the asymptotic period since the system does not have enough time to relax onto the asymptotic α -eigenstate. Then, at $t = 6$ s, when the control rods revert to their initial position, there is a prompt drop and the system relaxes back to steady-state conditions.

For comparison, we have performed the numerical integration of the adjoint-weighted point-kinetics equations, presented in Sec. 1.2.3 (Eqs. (1.2.23) and (1.2.28)), considering the 8 precursor families from JEFF-3.1.1 nuclear data library, with the kinetics parameters reported in Tables 3.1 (critical and supercritical configurations) and 2.2. The point-kinetics solution is shown in Fig. 3.3 and is in good agreement with the TRIPOLI-4 simulation result. The delayed constant of the point-kinetics excursion is in better agreement with TRIPOLI-4 best-fit constant than the delayed asymptotic reactor period, because TRIPOLI-4 and point-kinetics take the full time evolution into account, while the reactor period calculation gives the value associated to the long-time dominant precursor family, which is the largest one.

3.1.5 The role of precursors

Precursors drive the behaviour of slow kinetics because of their long decay constants. It is perhaps less obvious that their influence needs to be taken into account whenever the precursor population is initially at equilibrium with the neutrons, even if the time evolution of the system is expected to be chiefly prompt-neutron-driven. Here, we want to show that they are also important for fast kinetics simulations, such as reactivity insertion accidents, and thus need to be accounted for. As an illustration, we present the simulation results of a transient computed on the SPERT III E-core with and without precursors. The system was initially prepared on the critical configuration. Then, at $t = 0$ ms, the 8 control rods are extracted from the core by 3.2 cm, introducing a positive reactivity worth of about 2\$. The kinetic evolution of the core is monitored over 1 ms by time steps of $\Delta t = 200 \mu\text{s}$. Similarly to Sec. 3.1.4, TRIPOLI-4 takes two geometries as inputs: one with the control rods at critical height, and one with the control rods extracted by 3.2 cm. The kinetics parameters for the second configuration are reported in Table 3.1 (prompt supercritical configuration).

When precursors are taken into account, each fission results on average in $\nu = \nu_p + \nu_d$ neutrons, among which ν_d delayed neutrons appear after the decay time. If precursors are neglected, there is no decay and therefore all neutrons must be instantaneously emitted at fission. This begs the question of how many neutrons should then be created. There are two reasonable options: either ν or ν_p neutrons. The simulation results with both options are presented in Fig. 3.4. A third calculation, taking precursors into account, serves as a reference.

Clearly, neither simplified simulation scheme agrees with the reference flux, even though the system is prompt supercritical. The population grows too quickly when fissions produce ν neutrons on average, and too slowly when they produce only ν_p neutrons. The latter choice should asymptotically provide a very good approximation of the reactor period in prompt supercritical conditions. However, the simulation with precursors is in a transient state considering

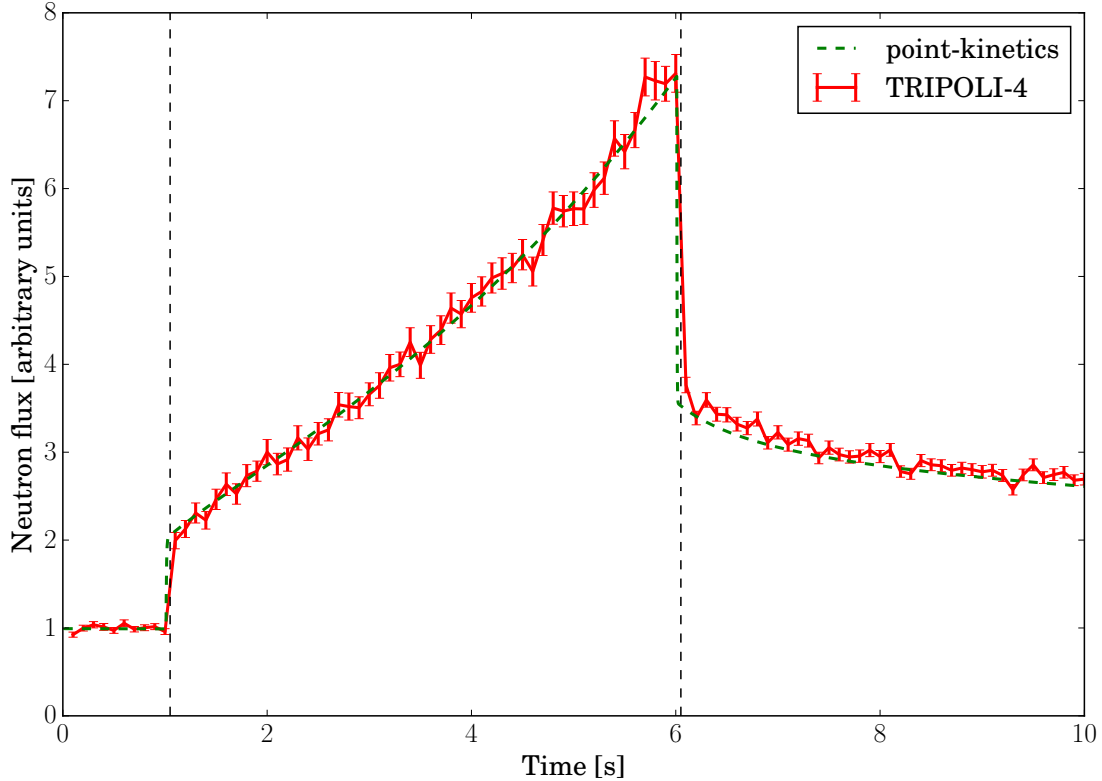


Figure 3.3 – Time evolution of the total neutron flux in SPERT III E-core. The critical configuration receives a reactivity insertion at $t = 1$ s with a 0.8 cm withdrawal of the control rods. The system returns to critical state at $t = 6$ s when the control rods go back to their initial position. The red line is the solution of the TRIPOLI-4 simulation while the green lines are the solutions of the point-kinetics models for the critical and the reactivity insertion configurations. The Best-fit exponential time constants for the excursion between $t = 1$ s and $t = 6$ s and for $t > 6$ s are 0.25 s^{-1} and -0.65 s^{-1} respectively.

the simulation time. To observe the asymptotic state, we should follow the kinetic behaviour over a longer time; only then would the red and green fluxes evolve with the same period. For the present choice of time interval, precursors are essential to the simulation in order to reveal the transient part. Treating the delayed neutrons as prompt, as it is done in the blue calculation, results in the right behaviour over very short time scales, but asymptotically diverges away from the reference calculation.

Therefore, neglecting the precursors in kinetic simulations, even in prompt supercritical conditions, seems inappropriate if precursors are very abundant in the initial state, which is precisely the case of excursions from criticality.

3.1.6 Rod drop

We now illustrate the feasibility of kinetic Monte Carlo simulations involving the observation of both prompt and delayed neutron evolution. For this purpose, we have prepared the critical state with the power iteration algorithm on the configuration presented in Table 3.1 (nominal configuration), and we have assumed that the 8 control rods are moved to the shutdown position at the beginning of the kinetic simulation. Hence, the rods are fully inserted into the core, introducing about $-19 \$$ static reactivity worth in the system: see Table 3.1 (rod-drop configuration)

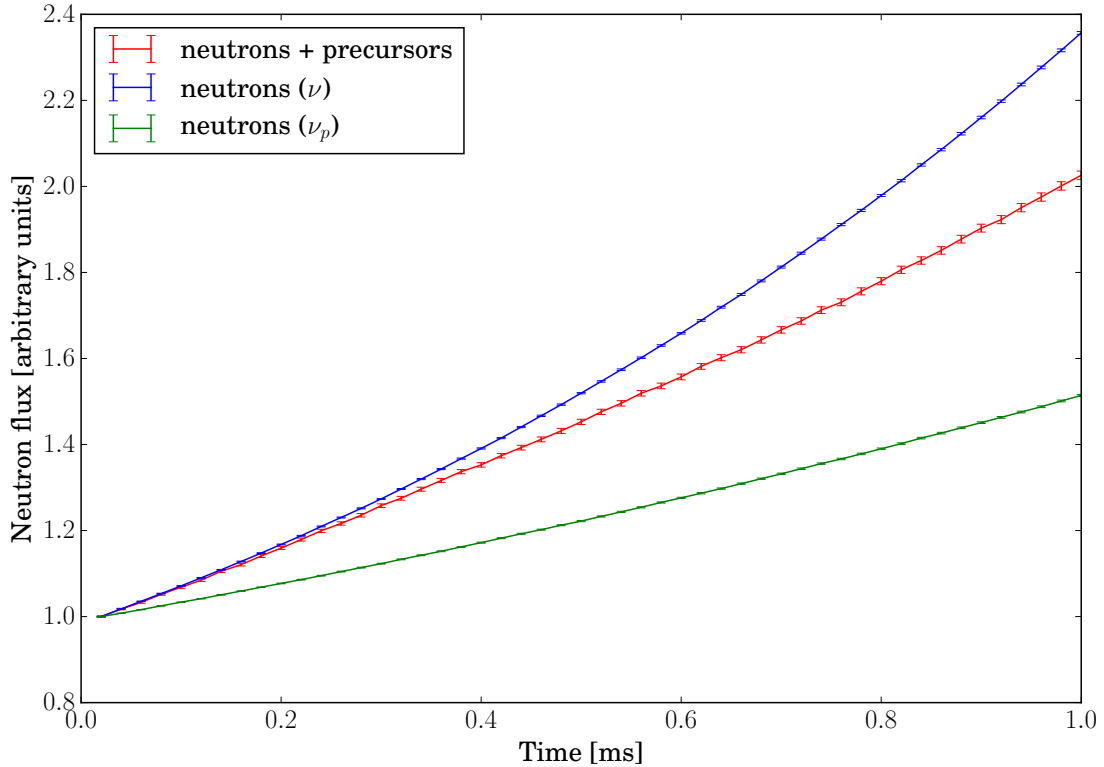


Figure 3.4 – Time evolution of the total neutron flux in SPERT III E-core. The system is prepared on the critical condition and is perturbed at $t = 0$ s with a reactivity insertion of approximately 2% . Its kinetic evolution is followed up to 1 ms. Red: TRIPOLI-4 simulation with precursors. Blue: TRIPOLI-4 simulation without precursors and ν neutrons created at each fission on average. Green: TRIPOLI-4 simulation without precursors and ν_p neutrons created at each fission on average.

for the kinetics parameters. We have also computed the prompt and delayed alpha eigenvalues for this configuration by two reactor period calculations with 5000 inactive cycles and 20000 active cycles of 10^4 neutrons.

For the kinetic simulation, we have used a time-dependent population importance ratio so as to observe the two time scales of the system due to both prompt and delayed neutrons. Indeed, in order to properly observe the prompt drop, combing must be frequently performed to prevent the population from dying away. Hence, the simulation begins with a time grid from $t = 0$ ms to $t = 1$ ms with $\Delta t = 200 \mu\text{s}$ time intervals for the combing. A population importance ratio of $R = 10^{-3}$ has been used in order to increase the number of neutrons since the time intervals are too small to allow for delayed neutron emission. The delayed behaviour was captured over a larger time interval, $0.1 \text{ s} \leq t \leq 500 \text{ s}$, in steps of $\Delta t = 0.1 \text{ s}$. No importance factor was applied over this interval.

The TRIPOLI-4 simulation results are displayed in Fig. 3.5. Two distinct behaviours can be clearly identified: *i*) a prompt drop and *ii*) a slow decay due to delayed neutrons. To get a better view of the prompt drop, Fig. 3.6 zooms on the time interval $0 \text{ ms} \leq t \leq 1 \text{ ms}$. Here, the flux tends to decrease exponentially with a best-fit time constant of -7300 s^{-1} , which is different from the value of prompt alpha $\alpha_p = (-5074 \pm 79) \text{ s}^{-1}$ since the system has not reached its asymptotic state. After the prompt drop, the flux approaches an exponential decay with a best-fit

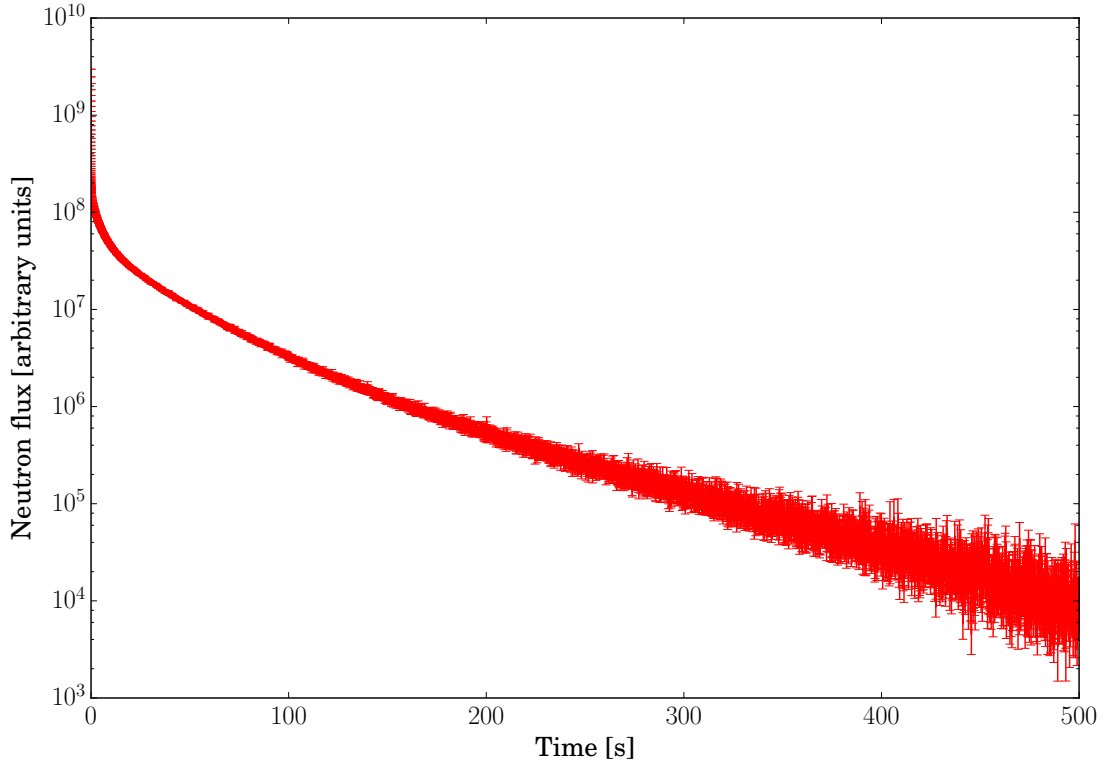


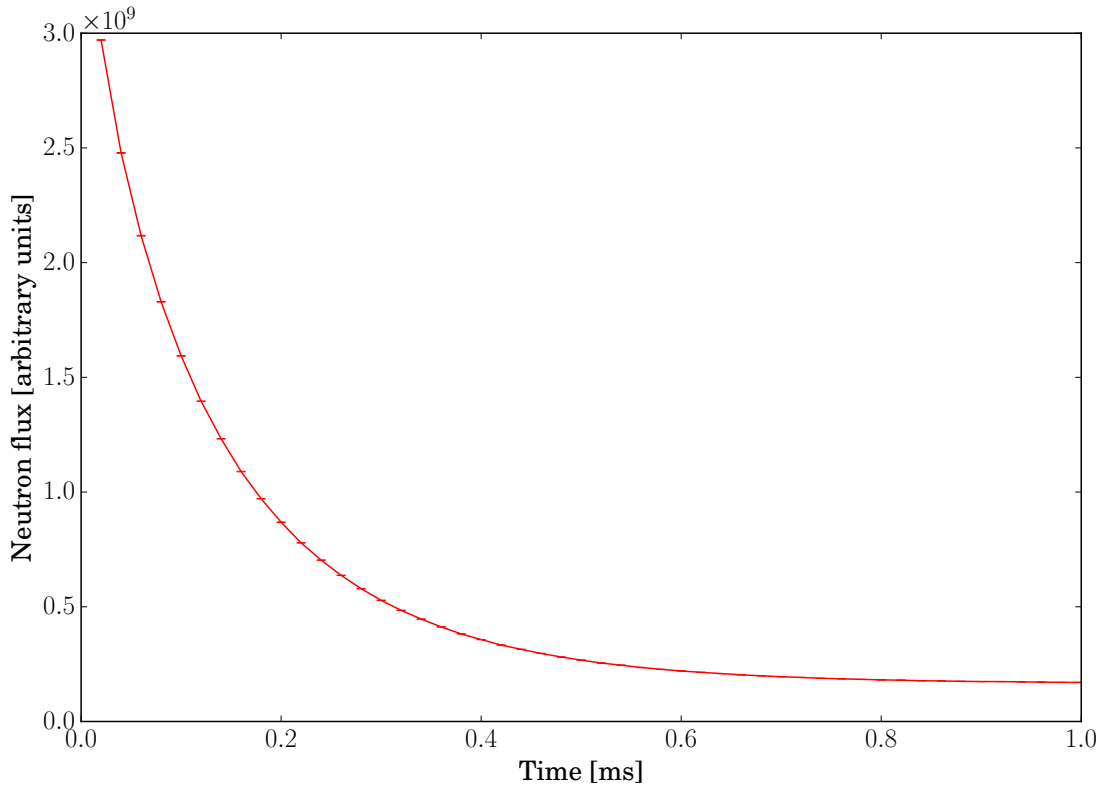
Figure 3.5 – Time evolution of the total neutron flux in SPERT III E-core. The source is prepared in a critical state. At the beginning of the kinetic simulation ($t = 0$ s), the control rods are fully lowered into the core, introducing 19 \$ anti-reactivity. The figure shows the prompt drop but also the slow decay due to delayed neutrons. An exponential fit for this decay part yields a time constant of 14 ms^{-1} . It was possible to simulate the two time scales through a change in the importance factor along the kinetic simulation.

time constant of -14 ms^{-1} , which is in good agreement with the calculated value of the asymptotic reaction period, $\alpha_d = (-12.20 \pm 0.51) \text{ ms}^{-1}$. Hence, the TRIPOLI-4 simulation result is in agreement with the alpha eigenvalue calculations for the delayed regime.

This kinetic simulation can also be used to assess the anti-reactivity insertion, expressed in dollars, by resorting to an integral-count technique (Keepin, 1965). When applying the Laplace transform to the point-kinetics equation, the reactivity can be expressed as

$$\rho [\text{\$}] \sim \frac{n_0}{\beta_{\text{eff}}} \times \frac{\sum_i \frac{\beta_{\text{eff},i}}{\lambda_i}}{\int_0^\infty n(t) dt}, \quad (3.1.2)$$

with n_0 the integral of neutron count before the rod drop, and $\int_0^\infty n(t) dt$ the total neutron count after the drop. Using the effective delayed neutron fraction presented in Table 3.1 (rod-drop configuration), the decay constants and the results of the kinetic simulation, we obtain $\rho \sim -17 \text{\$}$, which is in reasonable agreement with the $-19 \text{\$}$ obtained by the static k_{eff} calculation.

Figure 3.6 – Same as Fig. 3.5, for $0 \text{ ms} < t < 1 \text{ ms}$.

3.2 Verification tests on the TMI 3x3 mini-core

In this section, we present a second set of kinetic simulations performed on the transient benchmark described by the NEA (Ivanov et al., 2013b), from which the assembly studied in the previous chapter was extracted. The system is a 3x3-assembly mini-core based on the full core geometry of the TMI-1 reactor. This work was performed within the framework of the European project McSAFE, as a joint work with KIT (Karlsruher Institut für Technologie) and VTT (Technical Research Centre of Finland). Four different transient scenarios have been defined and simulations were performed with TRIPOLI-4 and Serpent 2 for code-to-code comparisons. KIT and VTT teams were in charge of Serpent 2 calculations, while we were in charge of TRIPOLI-4 calculations. The long-term goal of this benchmark problem is to compare transients simulations, taking into account thermal-hydraulics feedbacks; the kinetic simulations presented here represent a first step towards this goal. The benchmark results were published in ?.

3.2.1 The TMI 3x3 mini-core

The 3x3 mini-core is based on the TMI-1 reactor. Each of the 9 fuel assemblies consists of 15x15 rods, made of 4.12% enriched UOX. Each fuel assembly also contains four ($\text{Gd}_2\text{O}_3 + \text{UO}_2$) burnable poison pins. The active fuel length is 353.06 cm, and the width of each assembly is 21.64 cm. An instrumentation tube is located in the center of each assembly. Assemblies are surrounded by a reflector (whose width is 21.64 cm), made of borated water and stainless steel. The central assembly contains 16 extra control rods composed of a Ag-In-Cd core and Inconel cladding. For the critical hot zero power configuration, control rods are fully inserted, meaning that the insertion depth is 353.06 cm. The core active height is divided in 10 axial slices, and there is no radial discretization. A more detailed description of the system is given in Sec. A.4.

For our code-to-code comparisons, we have defined an axial gradient for the temperatures and densities: coolant temperatures range from 563 K to 572 K, and fuel temperatures range from 900 K to 979 K, as described in Table 3.2. Material compositions were taken from Ivanov et al. (2013b) and are presented in Table A.6. We have implemented the geometry with ROOT. A radial view and an axial view of the geometry are illustrated in Fig. 3.7.

axial slice	fuel temperature [K]	coolant temperature [K]	coolant density [g/cm ³]
1 (bottom)	900	563	0.745
2	909	564	0.743
3	918	565	0.740
4	926	566	0.738
5	935	567	0.736
6	944	568	0.733
7	953	569	0.731
8	962	570	0.729
9	971	571	0.727
10 (top)	979	572	0.725

Table 3.2 – Fuel temperatures, coolant temperatures and coolant densities in the ten axial slices in the mini-core benchmark.

3.2.2 Preliminary criticality calculations

Kinetics parameters for the nominal configuration were computed with a criticality calculation and are displayed in Table 3.3 (nominal configuration). The multiplication factor $k_{\text{eff}} = 1.00124 \pm 1.7 \times 10^{-4}$ is close to 1. However, as mentioned earlier, even such small deviation will cause the system to drift away from the stationary state during a kinetic simulation. For this reason, in order to minimize these deviations, we have determined the boron concentration leading to $k_{\text{eff}} \approx 1$ by using the critical boron search functionality available in TRIPOLI-4. The critical concentration was found to be 1493 ppm, meaning that the 13 ppm of boron account for 128 pcm, resulting in a differential boron worth of about 9 pcm/ppm, which is a realistic value. We have performed our simulations with this adjusted value. Kinetics parameters for this critical configuration are displayed in Table 3.3 (critical configuration).

In order to characterize the control rods worth, we have made several static criticality calculations, for different extraction heights, ranging from 0 cm (rods are fully inserted and the system is critical at critical depth) to 353.06 cm (full extraction of the rods). The reactivity values (see Fig. 3.8) obtained with TRIPOLI-4 were compared to the results of Serpent 2 for the same configurations, and are in good agreement.

3.2.3 Reactivity insertions

The four reactivity excursion transients presented in the following have been computed on the Cobalt CEA supercomputer from the TGCC (Très Grand Centre de Calcul, Bruyères-le-Châtel, France), during about 16000 CPU hours, in order to attain a small standard error on the global integrated neutron flux (0.5% at most in each time step).

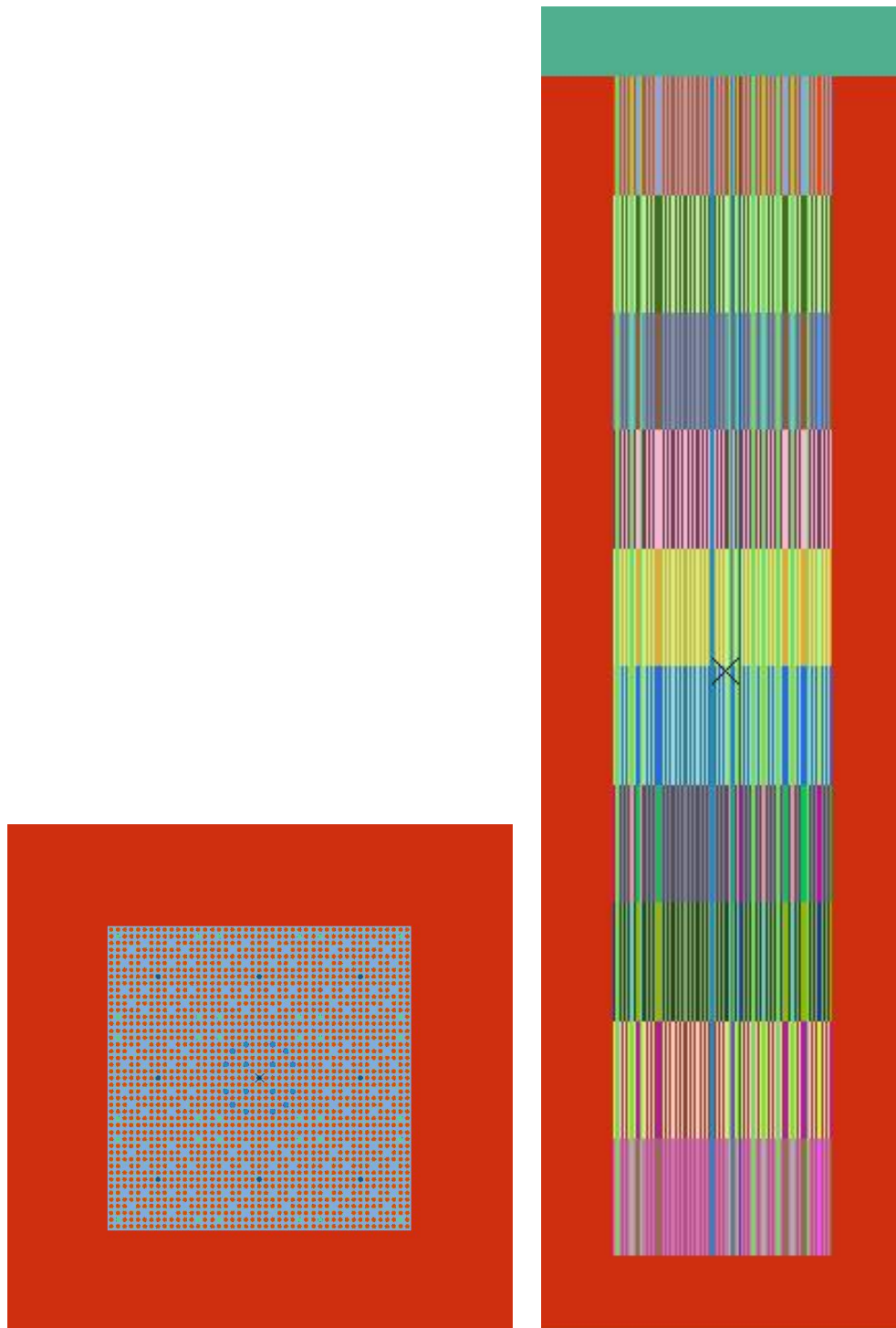


Figure 3.7 – Views of the ROOT implementation of the TMI-1 mini-core geometry. Each color defines one composition. Left: radial cut with the central assembly with control rods (in blue). The burnable poison pins are represented in green. Right: axial cut.

Scenario A

The system has been initially prepared on the critical state with the power iteration algorithm. Then, the 8 control rods located in the central assembly have been extracted by 40 cm between $t = 0.3$ s and $t = 1.3$ s: the insertion depth varies from $h_c = 353.06$ cm to $h_{10} = 313.06$ cm.

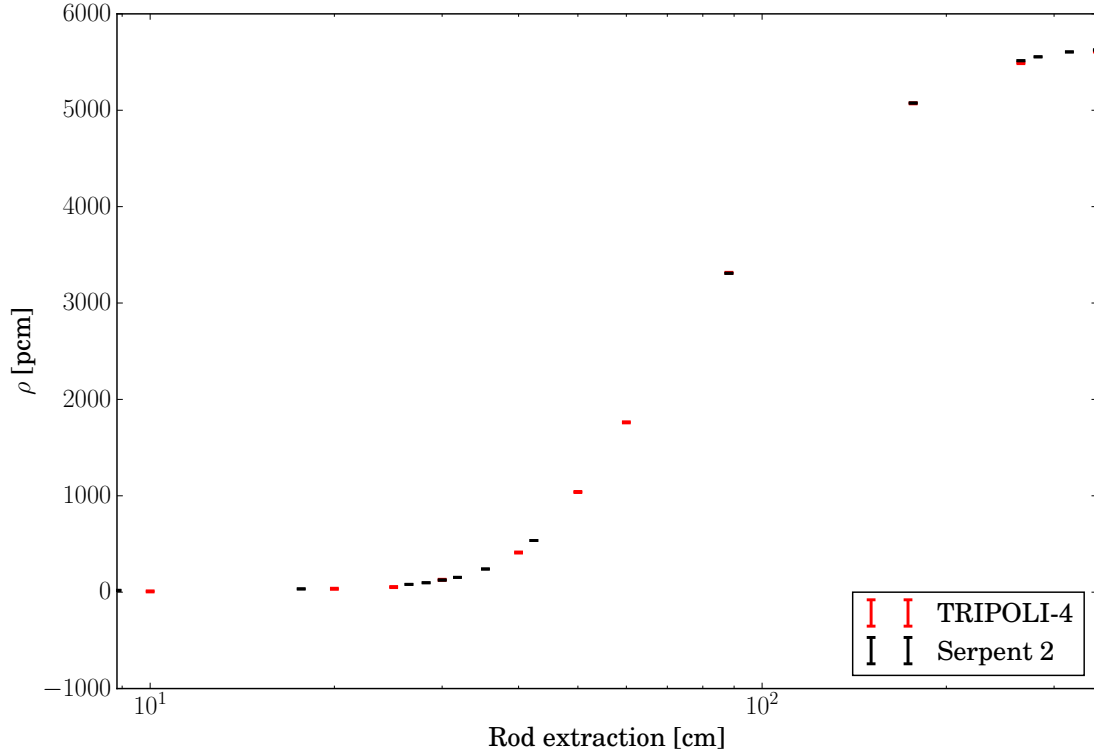


Figure 3.8 – Static reactivity as a function of control rod extraction for the TMI-1 mini-core. Criticality calculations were performed for several configurations: starting from critical configuration with complete insertion of the control rods in the core, up to complete extraction. Red: TRIPOLI-4 simulation results. Black: Serpent 2 simulation results. Error bars are plotted but they are not visible because of the graph scale (standard errors are below 10 pcm).

configuration	nominal	critical	supercritical
boron [ppm]	1480	1493	1493
k_{eff}	$1.00124 \pm 1.7 \times 10^{-4}$	$0.99995 \pm 5.3 \times 10^{-5}$	$1.00413 \pm 5.3 \times 10^{-5}$
Λ_{eff} [μs]	16.568 ± 0.015	16.5529 ± 0.0050	16.5519 ± 0.0049
β_{eff} [pcm]	756.5 ± 6.8	755.5 ± 2.2	757.8 ± 2.2
$\beta_{\text{eff},1}$ [pcm]	22.09 ± 1.1	24.72 ± 0.38	22.38 ± 0.78
$\beta_{\text{eff},2}$ [pcm]	108.0 ± 2.6	106.68 ± 0.82	107.9 ± 1.8
$\beta_{\text{eff},3}$ [pcm]	63.7 ± 2.0	67.31 ± 0.64	66.3 ± 1.4
$\beta_{\text{eff},4}$ [pcm]	142.5 ± 2.9	145.75 ± 0.94	141.4 ± 2.1
$\beta_{\text{eff},5}$ [pcm]	245.8 ± 3.8	245.91 ± 1.23	243.4 ± 2.5
$\beta_{\text{eff},6}$ [pcm]	83.0 ± 2.2	82.69 ± 0.71	83.4 ± 1.6
$\beta_{\text{eff},7}$ [pcm]	65.8 ± 2.0	66.80 ± 0.64	67.9 ± 1.4
$\beta_{\text{eff},8}$ [pcm]	25.62 ± 1.3	26.07 ± 0.39	25.16 ± 0.84
ρ [pcm]	124 ± 17	-5 ± 11	411.4 ± 5.3
ρ [\$]	~ 0.2	~ 0	~ 0.5

Table 3.3 – Multiplication factor and adjoint-weighted kinetics parameters for the mini-core in different configurations, as computed with the JEFF-3.1.1 nuclear data library. Precursors are regrouped into 8 families. Error bars are not given for the reactivity expressed in dollars because we did not measure correlations between k_{eff} and β_{eff} . See the text for the precise meaning of the configurations.

Thus, the velocity for rod extraction is 40 cm s^{-1} . At the end of the movement, this perturbation makes the system supercritical with $\Delta\rho \sim 0.5\%$. The control rods depth progressively goes back to the critical value h_c , at the same speed of 40 cm s^{-1} , between $t = 1.3 \text{ s}$ and $t = 2.2 \text{ s}$. The kinetic evolution of the system is monitored over $t = 5 \text{ s}$ by increments of $\Delta t = 0.1 \text{ s}$. In order to describe this scenario consistently with Serpent 2, we have used 11 different geometries as inputs: that is, we have discretised the control rod ramp in 10 steps, with the control rod position at each step n being given by

$$h_n = h_c - n \times 4 \text{ cm.} \quad (3.2.1)$$

The first geometry with $h_c = 353.06 \text{ cm}$ is used for the preparation of the critical source, and for the kinetic simulation, up to $t = 0.3 \text{ s}$. The second one with $h_1 = 349.06 \text{ cm}$ is used between $t = 0.3 \text{ s}$ and $t = 0.4 \text{ s}$, the third one with $h_2 = 345.06 \text{ cm}$ is used between $t = 0.4 \text{ s}$ and $t = 0.5 \text{ s}$, and so on. The multiplication factor and adjoint-weighted kinetics parameters for the supercritical configuration with control rods insertion depth h_{10} are displayed in Table 3.3 (supercritical configuration). Results of the kinetic simulation are displayed in Fig. 3.9 (scenario A). TRIPOLI-4 and Serpent 2 are in good agreement, and point-kinetics yields a good approximation of the flux evolution.

Simulation results for the score at the fuel pin-cell level are also presented in Fig. 3.10, for three different time bins. The neutron flux is presented with the associated absolute and relative uncertainty. It is interesting to notice that the relative uncertainty in a cell for a given time step is roughly constant (between 0.5% and 0.7% depending on the time step), and is roughly equivalent to the relative uncertainty on the global integrated score (about 0.5%). This outcome is at first surprising since one would have naively expected the relative uncertainty to be proportional to the volume, and therefore the relative uncertainty to be larger for the score at pin-cell level than for the integrated score. This issue will be further investigated in Sec. 3.4.

Effect of the geometry discretization We have made a code-to-code comparison on a transient scenario with a non-continuous modeling of the rod extraction movement: as explained, to be consistent with Serpent 2 simulation, we have used a discretized extraction of the control rods with 11 different geometries. We have verified that TRIPOLI-4 and Serpent 2 were in good agreement, and we want now to that assess the capability of the 10-step discretization to represent a continuous ramp movement. In order to analyse the impact of the discretization, we have performed three additional TRIPOLI-4 simulations with finer discretizations. The first one uses 32 steps for the 40 cm extraction: rods are extracted from the core by 1.25 cm every 0.03125 s; the second one uses 100 geometries: rods are extracted from the core by 0.4 cm every 0.01 s; the third one uses 200 geometries: rods are extracted from the core by 0.2 cm every 0.01 s.

Figure 3.11 shows the relative difference on the total neutron flux, with respect to the simulation with 200 extraction steps. Results show a clear deviation at the peak: the flux peaks at higher values when the discretization is coarser, showing that a 10-step discretization is not fine enough to represent a continuous ramp movement. 32-step discretization results are in better agreement with the 200-step, and 100-step discretization results are even slightly better. Finally, we can conclude that 100 steps in 100 s is a fine enough discretization to capture the flux variations during the 40 cm rod extraction.

Scenario B

Transient scenario B begins in the same way as scenario A: rods are extracted by 40 cm and are then reinserted between $t = 1.3 \text{ s}$ and $t = 2.2 \text{ s}$. The same pattern is then repeated between $t = 2.5 \text{ s}$ and $t = 4.4 \text{ s}$. The time evolution of the total neutron flux is presented in Fig. 3.9

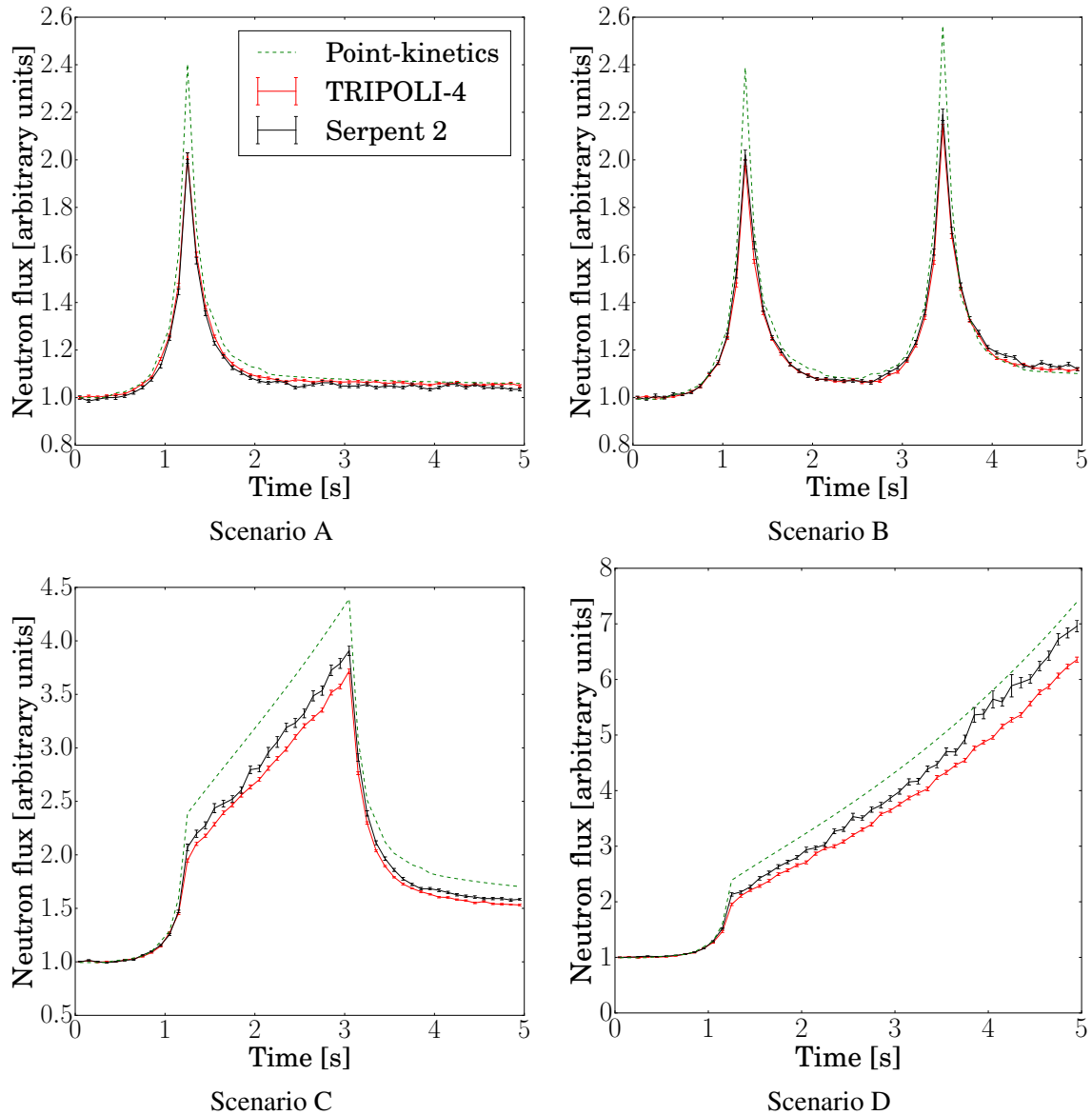


Figure 3.9 – Time evolution of the total neutron flux in the mini-core during the 4 different transient scenarios A, B, C and D. The system is first prepared on the critical state, then its kinetic evolution is monitored over 5 s. The system receives about a 0.5\$ reactivity insertion with a progressive extraction of the control rods, up to 40 cm and returns to critical state when the control rods return to their critical height (except for scenario D, where rods are not reinserted). The red line shows the result of TRIPOLI-4 simulation, the black line shows the result obtained with Serpent 2 and the dashed green line is the approximation from point-kinetics.

(scenario B). TRIPOLI-4 and Serpent 2 are in good agreement, and point-kinetics gives a fairly good approximation of the flux behaviour.

Scenario C

For scenario C, control rods are again extracted between $t = 0.3$ s and $t = 1.3$ s, and the super-critical configuration is maintained up to $t = 3.1$ s in order to observe the flux excursion. Rods are later reinserted to the initial height between $t = 3.1$ s and $t = 4$ s. The time evolution of the

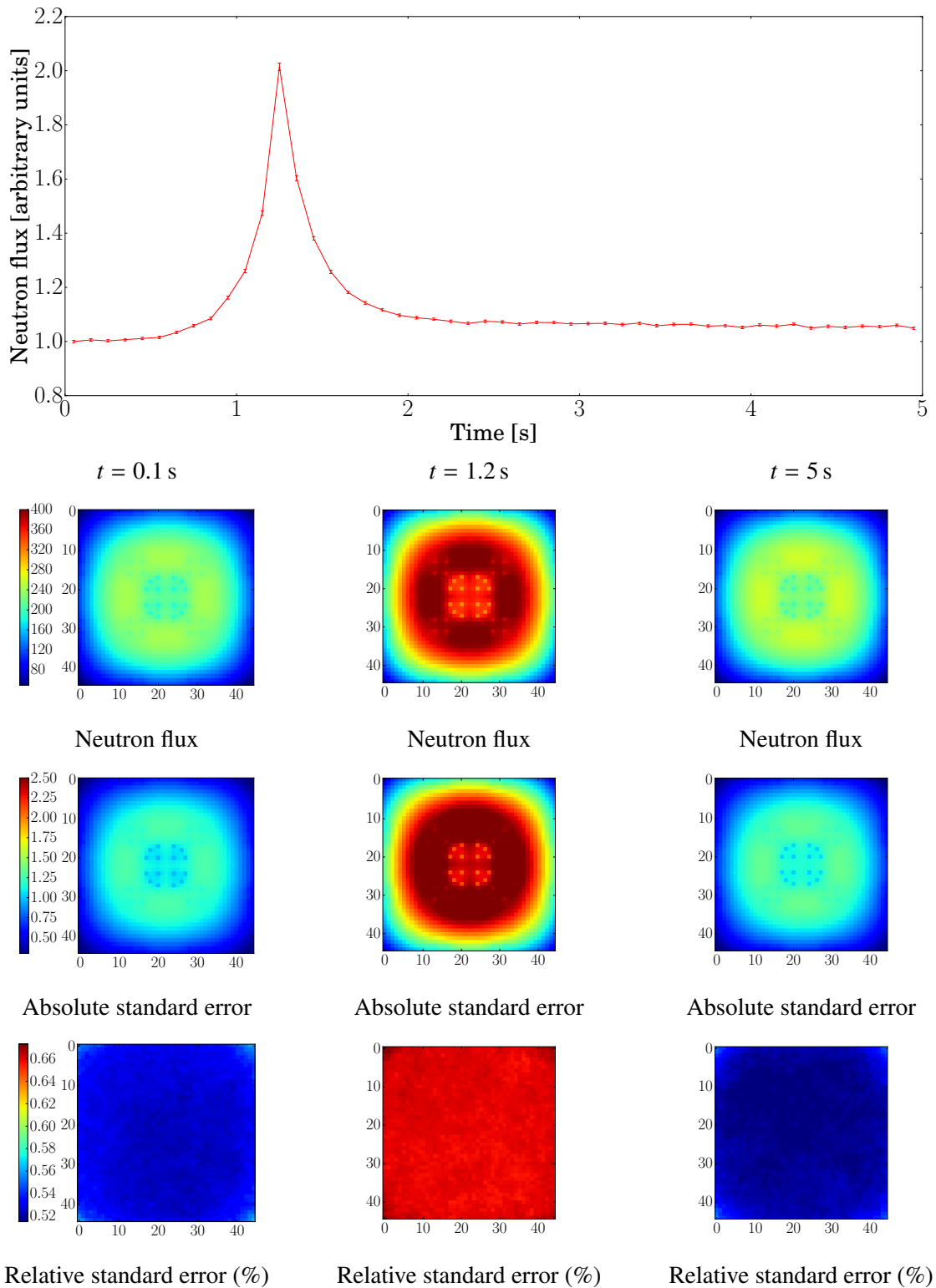


Figure 3.10 – Neutron flux at pin cell resolution, with the associated absolute and relative uncertainty, during scenario A transient, for the third slice, and at three different time steps: $t = 0$ s (steady state), $t = 1.2$ s (power peak) and $t = 5$ s (back to steady state).

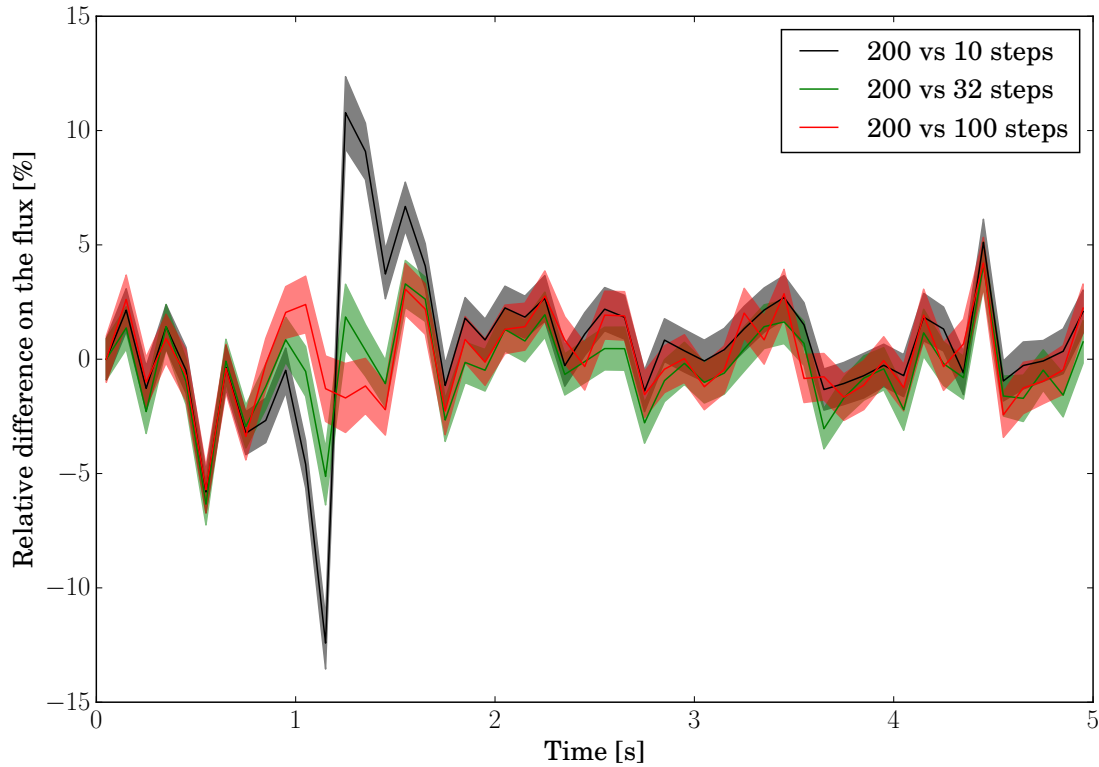


Figure 3.11 – TRIPOLI-4 simulation of transient A, with 4 different discretizations of the rods extraction: 10, 32, 100 and 200 steps. Black: relative difference between 200 and 10 steps. Green: relative difference between 200 and 32 steps. Red: relative difference between 200 and 100 steps.

total neutron flux is presented in Fig. 3.9 (scenario C). TRIPOLI-4 and Serpent 2 are in good agreement, up to the flux excursion, where Serpent 2 results seem to attain higher flux values than TRIPOLI-4 results. In fact, the strong correlation between time steps is the reason for this systematic deviation. Correlations between time steps in a kinetic calculation make the analysis much more challenging than for a criticality calculation, and will be investigated in Sec. 3.3.

Point-kinetics overestimates the flux excursion for this scenario. In fact, the agreement between point-kinetics and TRIPOLI-4 is significantly worse than for the 0.5\$ reactivity insertion scenario simulated with SPERT III E-core (Fig. 3.3). There are actually differences between the two transient simulations that might explain this deviation. First, the TMI-1 mini-core has a preferential direction (the active height is 353.06 cm for a width of about 60 cm). Point-kinetics might be more suited for SPERT III E-core, whose dimensions are more homogeneous (the active height is about 1 m for a width of about 50 cm). Second, the movement of the rod extraction is discretized for the mini-core, while it is instantaneous for SPERT III E-core; this may induce differences in the flux excursion. The two differences are for the moment only hypothesis to explain the difference of behaviour of the point-kinetics and deserve further investigation.

Scenario D

Transient scenario D depicts rod extraction without further reinsertion of the rods. Because the simulation does not take into account the thermal-hydraulics feedbacks, the power keeps increasing until the end of the simulation. Results are presented in Fig. 3.9 (scenario D). As

for scenario C, Serpent 2 and TRIPOLI-4 are in agreement even if correlations between time steps induce a deviation during the flux excursion. Also, point-kinetics overestimates again the flux profile. This last scenario is of interest for further investigations with thermal-hydraulics coupling, for the observation of the flux decrease due to feedbacks.

3.3 Investigating the correlations between time steps

In a kinetic TRIPOLI-4 calculation, particles are transferred from one time step to the next one. Therefore, scores collected in two different time steps are not independent. As it was observed in Secs. 2.6.2 and 3.2.3, scores are actually strongly correlated. This makes it quite difficult to assess for example the statistical significance of fluctuations.

The aim of the kinetic simulations presented below on SPERT III E-core is to characterize the correlations, due to fission chains, between the scores in different time steps. Such correlations have a short time range; thus, we have monitored the flux over small time intervals. Also, since correlations may depend on the inserted reactivity, we have performed the study on two different configurations: critical ($\rho \approx 0$ \$) and subcritical ($\rho \approx -19$ \$). Both configurations were already presented in Sec. 3.1.

3.3.1 Critical configuration

The system is prepared on the critical configuration. The kinetics parameters were reported in Tables 3.1 and 2.2 (critical configuration). The kinetic evolution is monitored over a total observation time of 10 ms, partitioned into 500 intervals of $\Delta t = 20 \mu\text{s}$. The time step is therefore approximately equal to the mean generation time (the mean generation time Λ_{eff} is about $17 \mu\text{s}$): $\Delta t \approx \Lambda_{\text{eff}}$.

We have simulated 3000 batches with 12000 particles per batch, and we do not use any population-control method. The total neutron flux was collected over the time intervals using the track length estimator. We have then computed the two-time correlation function for the total neutron flux $\xi(t_0, t)$.

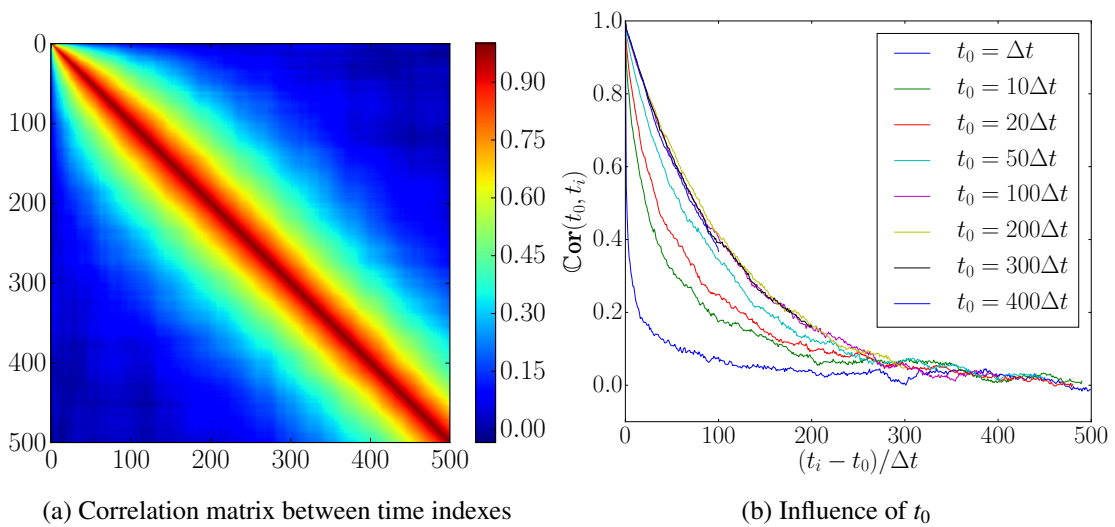


Figure 3.12 – Two-time correlation function for a critical configuration. The total neutron flux was collected over 500 time steps with TRIPOLI-4.

Figure 3.12a presents the correlation matrix, and shows the progressive decorrelation of the score at a time step from the previous time steps. Figure 3.12b presents the correlation function as a function of the final time t , for different initial times t_0 . The plot reveals two parts. For small values of $|t - t_0|$, correlations quickly drop off with time. Then, after about 200 time steps, correlations stop decreasing and remain stable around a low value. Also, the decay constant seems to depend on the initial time t_0 . Indeed, the decay goes faster for small values of t_0 . Then, new fission chains begin and bring additional dependency among particles sharing a common parent. As of $t_0 = 100 \times \Delta t$, the decay seems to be stable for different values of t_0 . In fact, for $t_0 < 100 \times \Delta t$, correlations are not yet representative of the total correlations of the system.

3.3.2 Subcritical configuration

As mentioned above, the variance on the neutron flux may depend on the kinetic of the system. Therefore, after examining the critical configuration, we have also considered the rod drop configuration: control rods are fully inserted in the core, introducing about $-19\ \$$ static reactivity worth in the system. The kinetics parameters were reported in Table 3.1 (rod drop configuration). As for the previous simulation, the kinetic evolution is monitored over a total observation time of 10 ms, partitioned into 500 intervals of $\Delta t = 20\ \mu\text{s}$. Again, the time step is approximately equal to Λ_{eff} , and 3000 batches have been simulated with 12000 particles per batch. A population importance ratio of $R = 10^{-3}$ has been used in order to increase the number of neutrons.

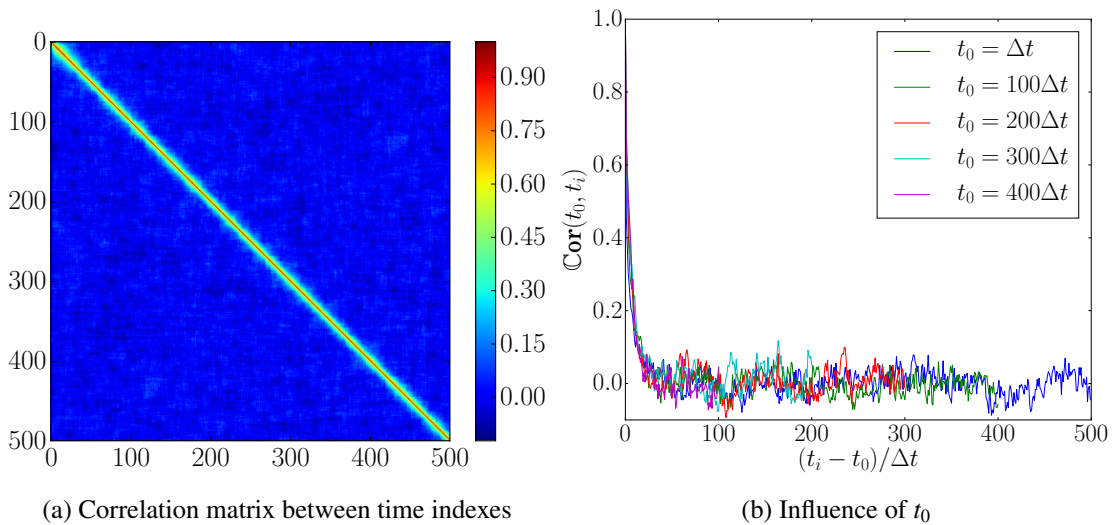


Figure 3.13 – Two-time correlation function for the rod drop subcritical configuration ($\rho \approx -19\ \$$). The total neutron flux was collected over 500 time steps with TRIPOLI-4.

Simulation results are presented in Fig. 3.13. Correlations drop drastically, reaching a low asymptotic value after a few steps only. The decay is much faster than for the critical configuration presented in Sec. 3.3.1. Fission chain length can explain this difference, as discussed in Sec. 3.3.3.

3.3.3 Fission chains length and impact on correlations

In order to explain the difference of behaviour between the correlations decay in the critical and the subcritical configurations, we have analyzed the properties of the fission chains. As explained in Sec. 1.3.3, for prompt-subcritical configurations (i.e., $k_p < 1$), the mean number \bar{n}

of fissions per fission chain is given by

$$\bar{n} = \frac{1}{1 - (1 - \beta_{\text{eff}}) \times k_{\text{eff}}}, \quad (3.3.1)$$

We investigate the consequences of this formula for the two configurations.

Critical configuration

For the critical configuration illustrated in Fig. 3.12b, Eq. (3.3.1) evaluated with parameters from Table 3.1 yields $\bar{n} = 130$ and the fission chain lifetime is $\bar{n} \times \Lambda_{\text{eff}} \approx 2$ ms. When considering the initial time $t_0 = 100 \times \Delta t$, correlations follow an exponential decay with a best-fit time constant of $\alpha = 0.009/\Delta t$ (cf. Fig. 3.14a for the fit). Thus, the decay time is $1/\alpha \approx 2$ ms, which is about the fission chain lifetime. In fact, within a fission chain, particles are correlated because they share a common parent. When the fission chain ends, the creation of correlated particles stops. Hence, as expected, correlations decay time is closely connected to fission chain lifetime.

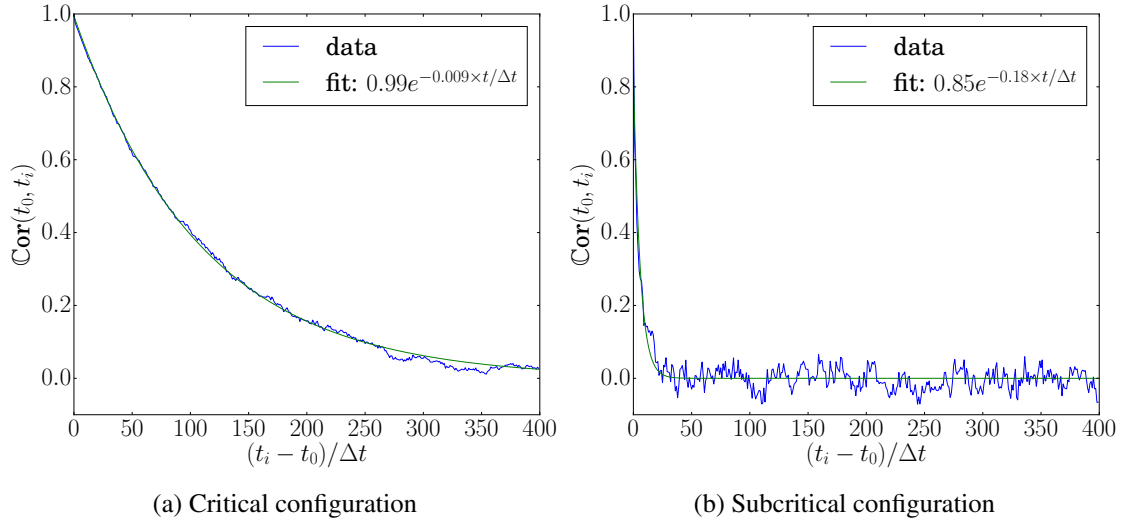


Figure 3.14 – Fitting of the correlations decay for critical and subcritical configurations presented in Figs 3.12b and 3.13b with initial time $t_0 = 100 \times \Delta t$.

Subcritical configuration

For the subcritical configuration illustrated in Fig. 3.13b, Eq. 3.3.1 evaluated with kinetics parameters from Table 3.1 yields $\bar{n} = 7$, so the fission chain lifetime is $\bar{n} \times \Lambda_{\text{eff}} \approx 0.1$ ms. Fission chains are on average much shorter than in the critical configuration. When considering the initial time $t_0 = 100 \times \Delta t$, the best-fit time constant is $\alpha = 0.18/\Delta t$ (cf. Fig. 3.14b for the fit). Here again, the decay time $1/\alpha \approx 0.1$ ms is in good agreement with the fission chain lifetime.

3.3.4 Conclusion

We have studied short-time correlations for the total neutron flux on two configurations, and we have found that correlation decay is driven by the fission chains lifetime. This conclusion was expected since the end of a fission chain kills the short-time correlations. Correlations drop below 10% after approximately $100 \times \Lambda_{\text{eff}}$ for the critical configuration, and after $20 \times \Lambda_{\text{eff}}$ for the subcritical configuration. More precisely, the decay constant is about 2 ms for the critical configuration and 0.1 ms for the subcritical rod drop configuration. Correlations decay more quickly

in the critical configuration than in the subcritical configuration because the fission chains are much shorter in the subcritical configuration.

This investigation shows the quick decay of correlations due to fission chains. But fission chains are not the only mechanism that can generate correlations. Indeed, precursors survive for a long time, compared to typical simulation time intervals ($\Delta t \leq 0.1$ s). Forced decay creates correlated delayed neutrons. The correlations due to precursors are left for future investigation.

Finally, population-control methods reduce correlations. Thus, the impact of the simulation time mesh should be further investigated as well.

3.4 Investigating the impact of the scoring mesh spatial discretization

The simulation scores (e.g., neutron flux and power) are often computed on a spatial mesh superimposed to the geometry for the purpose of coupling neutron transport and thermal-hydraulics. The statistics collected in each mesh cell improves with the number of neutrons reaching it, which is roughly proportional to the volume of the cell (whose size is adjusted depending on the level of spatial detail needed for the simulation). Therefore, one would naively expect that the relative uncertainty in a cell is proportional to its volume. In Sec. 3.2.3 however, for the kinetic simulation of transient A scenario, results at pin-cell level were presented in Fig. 3.10 and the associated relative uncertainties were about the same as for the global integrated score. We investigate here the scaling of the variance for kinetic simulations as a function of the spatial mesh size and compare our results to the case of static eigenvalue calculations, for the same configuration. For this study, we do not consider the full TMI-1 mini-core but the same assembly as in Chapter 2 to achieve a reasonable computational cost of the simulations. This investigation work was part of a publication (Faucher et al., 2019a).

3.4.1 Difference between kinetic and criticality calculations

We have performed a criticality and a kinetic calculation with two different spatial scoring meshes: in the first one, the flux is integrated over the whole assembly; in the second one, the flux is radially discretized at pin-cell level and axially discretized in 10 slices (the mesh consists of $N = 15 \times 15 \times 10$ cells). For the spatially resolved results, we have furthermore computed the mean score and the mean relative uncertainty σ_{rel}^m , which we have defined as

$$\sigma_{\text{rel}}^m = \frac{\sqrt{\frac{1}{N} \sum_{i,j,k} \sigma_{i,j,k}^2}}{\frac{1}{N} \sum_{i,j,k} \varphi_{i,j,k}}, \quad (3.4.1)$$

with $\sigma_{i,j,k}$ the absolute standard error in cell (i, j, k) and $\varphi_{i,j,k}$ the flux in cell (i, j, k) . For the kinetic calculation, we have simulated a single time step of $\Delta t = 10^{-1}$ s. Results are presented in Table 3.4. As expected, for both criticality and kinetic simulations, the mean score on a cell is a factor of N smaller than the global score. In the criticality calculation, as one would naively expect, the relative uncertainty is roughly multiplied by a factor \sqrt{N} ; on the other hand, in the kinetic calculation, the obtained result is less intuitive: the relative uncertainty on a cell turns out to be roughly the same as the global relative uncertainty.

configuration	criticality		kinetic ($\Delta t = 10^{-1}$ s)	
	flux [a.u.]	relative uncertainty (%)	flux [a.u.]	relative uncertainty (%)
assembly	5.20×10^2	1.83×10^{-2}	5.85×10^2	7.03×10^{-1}
1/10 pin cell	5.20×10^2	0.47×10^0	5.85×10^2	7.13×10^{-1}
ratio	1	0.04	1	0.99

Table 3.4 – Neutron flux and relative mean squared standard error for a criticality and a kinetic simulation of the TMI assembly in a critical configuration.

3.4.2 Dependence on the time step

The difference between the criticality and the kinetic simulations lies in the different ways the lifetime of the neutrons is taken into account. For the criticality calculation, scores are collected over power iteration cycles, which simulate a single generation (the generation time Λ_{eff} being about 17 μs). The migration area computed by TRIPOLI-4 is $D \approx 31 \text{ cm}^2$, which covers a few cells; thus a neutron can explore only a few cells on average within a cycle. For the kinetic calculation, however, scores are collected over $\Delta t = 10^{-1}$ s, which is larger than the lifetime of a fission chain (on average $\Lambda_{\text{eff}}/\beta_{\text{eff}} \approx 2 \times 10^{-3}$ s, as explained in Sec. 1.3.3). Consequently, in the time bin, the different neutrons from a fission chain make strongly correlated contributions to several cells of the assembly. Correlations make the variance on the global (integrated) score increase because of the additive covariance terms.

We can assume that the distance travelled by neutrons from a same fission chain during a time Δt increases as the square root of the time. Thus, assuming that $\Delta t < \Lambda_{\text{eff}}/\beta_{\text{eff}}$, the distance can be estimated by

$$l_{\Delta t} = \sqrt{6 \times D \times \Delta t / \Lambda_{\text{eff}}}, \quad (3.4.2)$$

and the travelled length during $\Delta t = 10^{-3}$ s is found to be about 1 m, which is of the order of the size of the system. Now, for shorter values of Δt , the explored length also decreases. Below $\Delta t = 10^{-3}$ s, we can guess that the fission chains will not have time to explore all the cells of the assembly, and that the relative uncertainty on the cells will be larger than on the global score.

In order to verify this conjecture, we have scored the neutron flux in time intervals of several sizes. The ratio between the relative uncertainty on the score at cell-level and integrated over the whole assembly are displayed in Fig. 3.15. For $\Delta t = 10^{-2}$ s and $\Delta t = 10^{-3}$ s, the relative uncertainty is found to be roughly the same between the two scoring meshes. However, below $\Delta t = 10^{-4}$ s, the relative uncertainty at pin-cell level becomes sensibly larger than the integrated one, although the ratio observed in the criticality calculation is not attained. As a matter of fact, the ratio between the two relative uncertainties reaches a plateau at $\Delta t = 10^{-5}$ s, which is about the generation time. The reason for this plateau requires further investigation.

For larger systems (whose typical size is larger than 1 m), the exploration of all the cells requires more time. For the TMI-1 3x3 mini-core for example (as a reminder, it consists of 9 fuel assemblies and the active height is 353.06 cm), the time required to explore the whole system, as given by Eq. 3.4.2, is about $\Delta t = 10^{-2}$ s, which is larger than the average fission chain lifetime. Therefore, cells should be less correlated than for the assembly: regarding the size of the system, neutrons from a same fission chain can not explore the whole mini-core. In order to verify this assumption, we have computed the ratio between the relative uncertainty on the global score and on the score at cell-level (the mesh consists of $N = 45 \times 45 \times 10$ cells), over

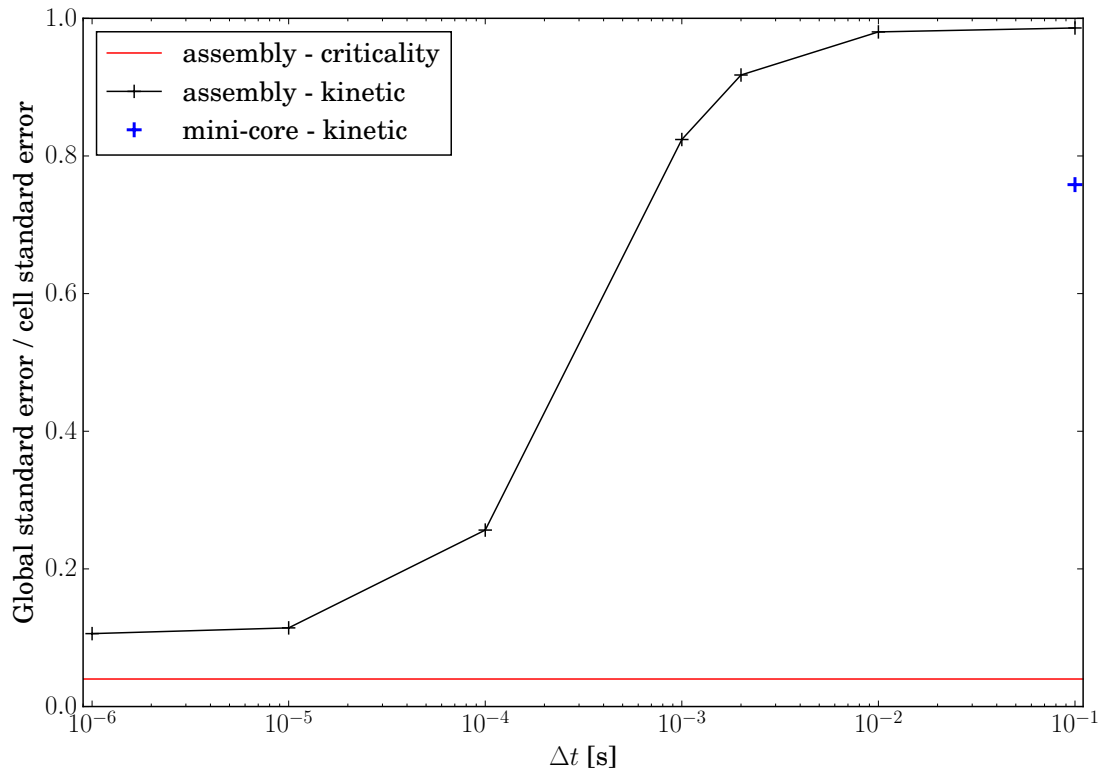


Figure 3.15 – Ratio between the relative uncertainty on the score integrated over the whole assembly and at cell-level. The red line denotes the criticality calculation reference value for the assembly. The black line denotes the results of the kinetic calculations on the assembly for the different sizes of time step. The ratio decreases as the time step decreases, but it reaches a plateau before reaching the value of the criticality calculation. The blue cross presents the result of the kinetic calculation on the 3x3 mini-core for $\Delta t = 0.1$ s.

$\Delta t = 10^{-1}$ s (see Fig. 3.15). As expected, the ratio is smaller than 1.

3.5 Conclusion

We have numerically tested the kinetic Monte Carlo methods implemented in TRIPOLI-4, and detailed in Chapter 2, on two realistic configurations by examining the response of both prompt and delayed neutrons to different types of reactivity insertions. The first configuration was SPERT III E-core, a research reactor composed of 60 assemblies. The kinetic simulation of the reactor at steady state was presented. We have stressed the importance of taking into account precursors, even within a prompt supercritical excursion, and finally we have demonstrated the value of a time-dependent importance sampling scheme for observing both prompt and delayed regimes when simulating a rod drop. This work was published in Faucher et al. (2018).

The second configuration was a mini-core based on the TMI-1 reactor with 3x3 assemblies. Different transients were simulated and benchmarked against the Monte Carlo code Serpent 2. Results were satisfactory and encourage the application of the new kinetic capabilities that are now available in TRIPOLI-4 to the simulation of reactor transients with thermal-hydraulics feedbacks, which is the purpose of Part II. I co-authored a publication on the code-to-code comparison (?).

In the different kinetic simulations that we have performed, we have noticed that the scores are strongly correlated between time steps. Therefore, we have performed a numerical investigation of the correlations. Correlations due to fission chains have an exponential decay, and the decay time depends on the fission chain lifetime.

Finally, we have highlighted an interesting behaviour regarding the dependency of the relative uncertainty on the discretization of the scoring grid of a kinetic calculation. The contribution of the different neutrons from the same fission chain within a time step increases the variance on the total neutron flux due to the presence of correlations.

Part II

Dynamic Monte Carlo: time-dependent Monte Carlo neutron transport coupled with thermal-hydraulics

Chapter 4

Development of a coupling between TRIPOLI-4 and thermal-hydraulics

The purpose of this chapter is to present the new multi-physics interface for TRIPOLI-4, part of which has been developed in the frame of this thesis. More specifically, we present the coupling scheme between TRIPOLI-4 and the thermal-hydraulics sub-channel code SUBCHANFLOW.

4.1 Development of a multi-physics interface for TRIPOLI-4

In order to be able to perform multi-physics calculations with TRIPOLI-4 and thermal-hydraulics, we have developed a supervisor, i.e., an external program, whose aim is to orchestrate data exchange between the coupled codes. The implementation and the role of the supervisor are detailed below.

4.1.1 Development of a supervisor

In the latest version of TRIPOLI-4 (v11.0, released in December 2018), there is no multi-physics interface. The calculation parameters (geometry, compositions, scores, simulation options, etc.) are defined using the traditional TRIPOLI-4 input file. TRIPOLI-4 parses the file and creates an in-memory representation of the input file, which is subsequently processed to start the actual calculation. The input file-based approach is however unsuitable for a multi-physics calculation, which requires frequent updates of the material compositions. Therefore, we have chosen to split the parsing and the initialization of the different classes: this led to the development of an Application Programming Interface (API) for the TRIPOLI-4 library. The aim of the API objects is to permit the programmatic initialization of TRIPOLI-4 classes, bypassing input-file parsing.

Nonetheless, even with the API objects, all of the TRIPOLI-4 classes are exposed to the user, and setting up a calculation requires to be familiar with the code architecture. In order to simplify the use of the API, we have implemented an *in-between* class: T4_Facade, which can be called by the supervisor. T4_Facade provides a single entry point for the TRIPOLI-4 API, reduces the contact surface between TRIPOLI-4 and the supervisor and, in a nutshell, acts as an in-between layer between TRIPOLI-4 and the supervisor.

The development of the supervisor was performed within the frame of this thesis: we have developed the supervisor as a C++ program and we have then performed a verification work on simple test cases in order to make sure that a supervisor-driven calculation gives the same

results as a standalone TRIPOLI-4 calculation.

The supervisor can drive a TRIPOLI-4 calculation step by step, but the main benefit of the supervisor is that it can also drive other codes. The coupling between TRIPOLI-4 and thermal-hydraulics is intended to be generic in scope, in order to simplify future coupling schemes with other codes: for instance, thermomechanics feedbacks could be taken into account, or thermal-hydraulics could be solved at different scales with CFD or system codes. In this regard, the SALOME platform (Bergeaud and Lefebvre, 2010; SALOME, 2019) has been chosen as the tool for the development of coupling schemes within the McSAFE project. We have used two SALOME components in the supervisor: the MEDCoupling library and the ICoCo API. ICoCo serves as interface to facilitate the coupling, and MEDCoupling is used for the data transfer, as illustrated in Fig. 4.1, which gives a schematic representation of the coupling between TRIPOLI-4 and another code. The SALOME platform, as well as its tools, will be presented in detail in Sec. 4.1.4.

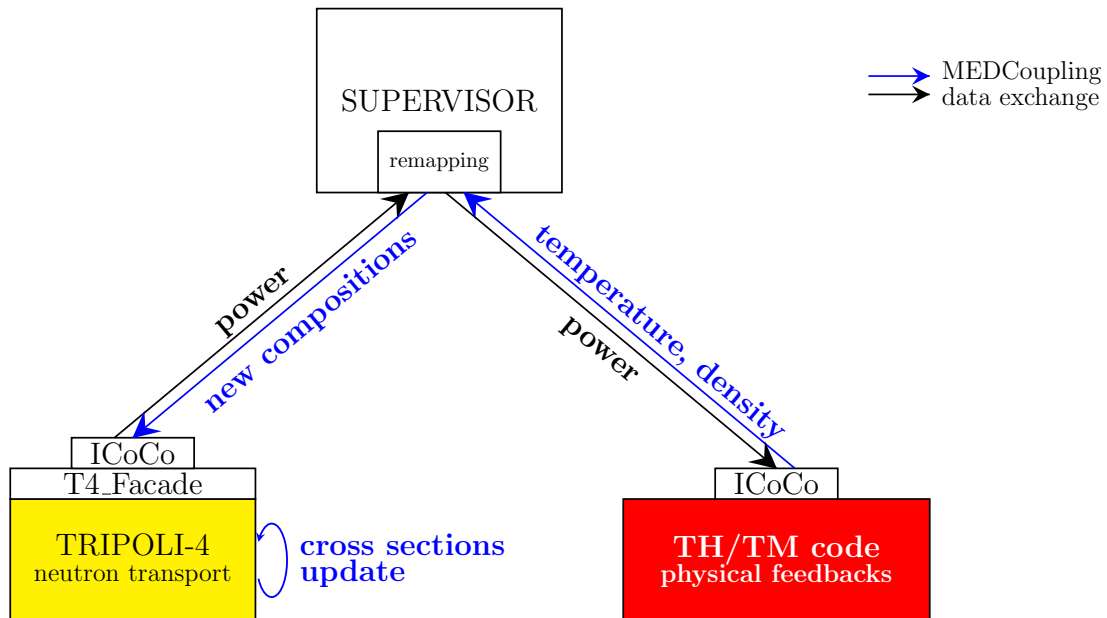


Figure 4.1 – Architecture of the supervisor-driven calculations performed using T4_Facade class, the ICoCo API and the MEDCoupling library.

In conclusion, the role of the supervisor is to initialize the solvers, launch the calculations and manipulate the fields, such as for performing data exchange, or remapping the fields from one mesh to another.

4.1.2 Exchanging data from the mesh to the geometry

A TRIPOLI-4 calculation takes as input a geometry composed of volumes, and material compositions are assigned to the volumes. During the coupled calculation, the supervisor receives temperature and density fields from the thermal-hydraulics code on a mesh, but TRIPOLI-4 only allows one temperature and density value per volume, and compositions must be updated in the volumes and not in the cells of the mesh. In general, the supervisor has the non-trivial task of projecting the temperature and density fields obtained from thermal-hydraulics onto the volumes of the TRIPOLI-4 geometry. One can distinguish two scenarios, depending on the size of the mesh cells relative to the volumes: when the cells are larger than the volumes, it makes

sense to assign the temperature and density values in the cell to all the volumes it encompasses. On the other hand, when several cells fit within a single volume, a suitable averaging strategy needs to be devised. For the moment, we have worked around this difficulty by constructing the TRIPOLI-4 geometry in such a way that such averaging is never necessary: cells and volumes coincide. In the long run, it is essential to extend the capabilities of the TRIPOLI-4 tracking algorithms to support continuously changing density or temperature fields. This could be done, for instance, using a variant of Woodcock's delta-tracking algorithm (Woodcock et al., 1965) as done in Serpent 2 (Leppänen, 2010).

The scores in TRIPOLI-4 can be computed either on volumes or on a spatial mesh superimposed on the geometry. We have chosen to score the power on a mesh, because the required tallies can be directly generated in MED format. The discretization of the scoring mesh is in principle independent of the one used for receiving temperature and density fields. In practice, the thermal-hydraulics fields should not be computed on a mesh finer than the power mesh; on the contrary, there is no need for a fine spatial discretization of TRIPOLI-4 results if the thermal-hydraulics is solved on a coarser mesh. Therefore, we have chosen to use the same mesh for scoring the power and receiving temperature and density fields.

4.1.3 Coupling between TRIPOLI-4 and SUBCHANFLOW

Two types of calculations can be performed with the new multi-physics interface of TRIPOLI-4: criticality calculations with feedback and transient (i.e., dynamic) calculations. Their architecture is described in Secs. 4.2 and 4.3, for execution in sequential mode first. The specificities of parallel calculations will be provided in Sec. 4.4.

In the following, we focus on the specific case of the coupling between TRIPOLI-4 and SUBCHANFLOW.

Interpolation between the meshes

A fuel-centered mesh is defined for TRIPOLI-4 to score the power. For SUBCHANFLOW, two meshes are defined: a fuel-centered mesh is defined to receive the power distribution from TRIPOLI-4, and a coolant-centered mesh is defined for the resolution of the conservation equations presented in Sec. 1.6.3. Note that the TRIPOLI-4 and SUBCHANFLOW meshes must have the same bounding box, but they do not need to be exactly superimposed or to be numbered the same way, as long as a spatial interpolation is performed between them.

TRIPOLI-4 computes the power distribution over the cells of the mesh. Then, the distribution is transferred to SUBCHANFLOW as a MEDCoupling input field. For the projection between the TRIPOLI-4 mesh and the SUBCHANFLOW mesh, the MEDCouplingRemapper tool is used to assign the power coming from TRIPOLI-4 to the correct cell in the fuel-centered SUBCHANFLOW mesh. The sub-channel code then computes the updated properties of the fuel (temperatures) and the moderator (temperatures and densities), which are then transferred to TRIPOLI-4. Again, the MEDCouplingRemapper tool is used to transfer the SUBCHANFLOW results (temperatures and densities) to the corresponding cells in TRIPOLI-4 mesh.

Temperature dependence of the cross sections

There are two distinct methods to take into account the temperature dependence of the cross sections. The first uses pre-broadened cross sections: such methods are fast, but they have a

large memory footprint due to the storing of cross sections for each nuclide on a tight temperature grid. The second concerns the so-called “on-the-fly” methods. Such methods only require the cross sections at one temperature, but they involve additional calculations for the Doppler broadening at each required temperature.

For the tabulated methods, the least expensive and also least accurate option to compute the cross section at a temperature T that is not tabulated is to use the closest available cross section as a replacement. Better accuracy is provided by stochastic interpolation. As a compromise between accuracy, computational cost and memory footprint, stochastic interpolation was introduced in TRIPOLI-4. The algorithm is the following. Assume that the temperature T is bracketed by T_{inf} and T_{sup} . For each flight, we randomly choose T_{inf} or T_{sup} , depending on how close T is to the end of the interval. The probability to choose the cross section associated to T_{inf} for T is

$$p(T) = \frac{T_{\text{sup}}^{\mu} - T^{\mu}}{T_{\text{sup}}^{\mu} - T_{\text{inf}}^{\mu}}, \quad (4.1.1)$$

and the probability to choose the cross section associated to T_{sup} is $1 - p(T)$. The coefficient μ depends on the selected option: either linear ($\mu = 1$) or square root ($\mu = 1/2$) interpolation are available in TRIPOLI-4. As an example for the fuel, temperatures typically range from 600 K to 2000 K, and are tabulated over 15 values in the CEAV512 nuclear data library delivered with TRIPOLI-4, with smaller intervals for lower temperatures, where variations of nuclear data are larger.

The best-known Doppler broadening method is the SIGMA1 algorithm developed by Cullen (1979). However, the on-the-fly broadening of the cross sections with this method is a prohibitive time-intensive process. The target motion sampling method (TMS) (Viitanen and Leppänen, 2012, 2014), which has been developed in Serpent 2, lends itself to on-the-fly calculations of the broadened sections: the target velocity is first sampled, and then collisions are handled in the target-at-rest frame.

4.1.4 Presentation of SALOME tools

SALOME is an open-source software that provides a generic platform integrating numerical solvers from various fields of physics. The platform integrates computer-aided-design modules, meshing algorithms, and advanced 3D visualization functionalities (with the use of ParaView). SALOME is developed by CEA, EDF and OpenCascade (Bergeaud and Lefebvre, 2010; SALOME, 2019).

MEDCoupling library for data exchange

MEDCoupling (Bergeaud and Lefebvre, 2010; SALOME, 2019) is a C++ library provided by the SALOME platform. It defines a format (the “MED” format) of data structures, describing spatially discretized scalar, vector or tensor fields. The data structures are easy to manipulate, to serialize and to exchange through parallel units. A MED field is characterized by two elements: the mesh, which is the spatial discretization of the geometric domain, and the field, which represents the physical data on the mesh. Fields can be carried by the mesh nodes, cells, etc. The library also provides powerful tools for algebraic manipulation. For instance, the MEDCouplingRemapper class allows to interpolate a field from one mesh to another.

In this work we have used the MEDCoupling library in order to perform in-memory exchange of the fields between TRIPOLI-4 and SUBCHANFLOW. Version 7.8.0 of the library

was used, with version 3.2.0 of MEDFile library, which is a prerequisite for MEDCoupling. Note that TRIPOLI-4 has the capability to provide outputs in MED format.

ICoCo for the coupling interface

ICoCo (Deville and Perdu, 2012) is a C++ API defining a standard interface for code integration and coupling. It was developed at CEA (DEN/DANS/DM2S/STMF) in the framework of the European NURISP project. The main “Problem” class defines a common interface for all solvers (neutron transport, thermal-hydraulics, thermomechanics, etc.) and abstracts away the solver-specific details. The goal of Problem is to simplify the implementation of solver-agnostic coupling schemes, and facilitate the substitution of one solver for another in an existing scheme. The class provides several methods, for initialization, solving, or field exchange for example. Figure 4.2 gives the common call order to the main methods and shows that the search for steady states or the computation of transients is easily achieved thanks to the generic interface. Moreover, MEDCoupling fields are compliant with ICoCo, and the combination of the two tools allow performing in-memory exchanges between codes in a simple way.

```
// Instantiate pb object from the SCFProblem class
SCFProblem *pb = new SCFProblem();

// Set the input file
pb->setDataFile("filename");

// Initialize pb with parameters from the input file
pb->initialize();

// Calculate the steady state
bool converged;
pb->solveSteadyState(converged);

// Calculate the transient over 0.1 s
double dt = 0.1;
pb->initTimeStep(dt);
pb->solveTimeStep();
pb->validateTimeStep();

// Terminate the calculation
pb->terminate();
```

Figure 4.2 – Example of call order of the main methods from the SCFProblem class implemented in ICoCo. First, an instance of the Problem class is initialized. Then, the steady state is computed, followed by a transient computed over 0.1 s. Finally, the computation is terminated to free the memory.

4.2 Criticality calculations with feedback

4.2.1 Description

The coupling scheme for criticality calculations with feedback is divided into iterations; for each iteration n , TRIPOLI-4 computes a power distribution \hat{P}_{fuel}^n over a few power iteration cycles.

The TRIPOLI-4 cycles are called “inner” iterations. In contrast, we call the coupling iterations “outer” iterations. Figure 4.3 gives a simple representation of one outer iteration.

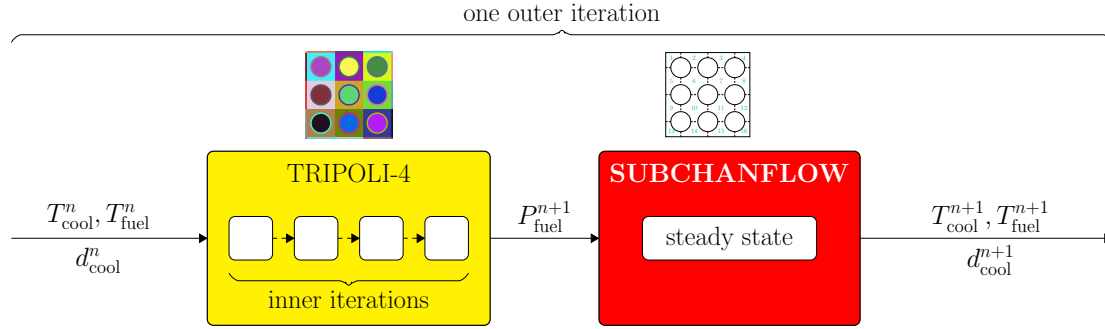


Figure 4.3 – Scheme of one outer iteration for the criticality coupling scheme. TRIPOLI-4 computes a power distribution \hat{P}_{fuel}^n over a few power iteration cycles. A relaxation scheme is imposed on the power distribution. The relaxed distribution P_{fuel}^{n+1} is then transferred to SUBCHANFLOW, which computes the corresponding steady-state configuration. The updated temperatures in the fuel rods and the coolant as well as the new densities in the coolant are transferred to the next iteration and will be used by TRIPOLI-4.

The criticality problem solved by TRIPOLI-4 is Eq. (1.2.10), which is linear in φ_k . Thus, the normalization of φ_k can not be determined by TRIPOLI-4 and must be imposed from the outside. One way to fix the normalization is to request that the total power deposited in the core is equal to some assigned value P_0 in watts. To this end, any score value must be multiplied by

$$r_s = \frac{P_0}{\sum_{i,j,k} \hat{P}_{i,j,k}^n}, \quad (4.2.1)$$

with $\hat{P}_{i,j,k}^n$ the power integrated over the (i, j, k) cell at iteration n . The factor r_s physically corresponds to the source intensity. The normalized distribution is then transferred to SUBCHANFLOW, which computes the corresponding steady-state temperatures and densities. The updated temperatures in the fuel rods and the coolant, as well as the new densities in the coolant, are transferred to TRIPOLI-4 and applied to the corresponding material composition.

In order to achieve a smooth convergence, a relaxation is imposed on the power distribution, as defined by the following equation

$$P_{i,j,k}^{n+1} = (1 - \alpha)P_{i,j,k}^n + \alpha \hat{P}_{i,j,k}^{n+1}, \quad (4.2.2)$$

with α the relaxation coefficient. We choose $\alpha = 1/2$, which has been found to be an optimal value in other coupling works (Daeubler et al., 2014; Gill et al., 2017). More sophisticated algorithms exist, such as the one proposed by Dufek and Gudowski (2006), where the number of neutron histories can grow along the iterations. Investigations of such techniques is left for future work.

The progress of the whole steady-state calculation from the first to the last outer iteration is illustrated in Fig. 4.4. We consider that convergence has been reached if the changes in the fuel temperature, coolant temperature, coolant density and multiplication factor from an iteration to the next one lie below some user-defined threshold. Quantitatively, we define the ℓ_2 -norm

residual as the convergence criterion for the coupling scheme:

$$\Delta X^n = \frac{\sqrt{\sum_{i,j,k} (X_{i,j,k}^n - X_{i,j,k}^{n-1})^2}}{\sqrt{C}} < \varepsilon_X, \quad (4.2.3)$$

for the different physical properties X , with C the total number of cells ($C = 1$ for the multiplication factor).

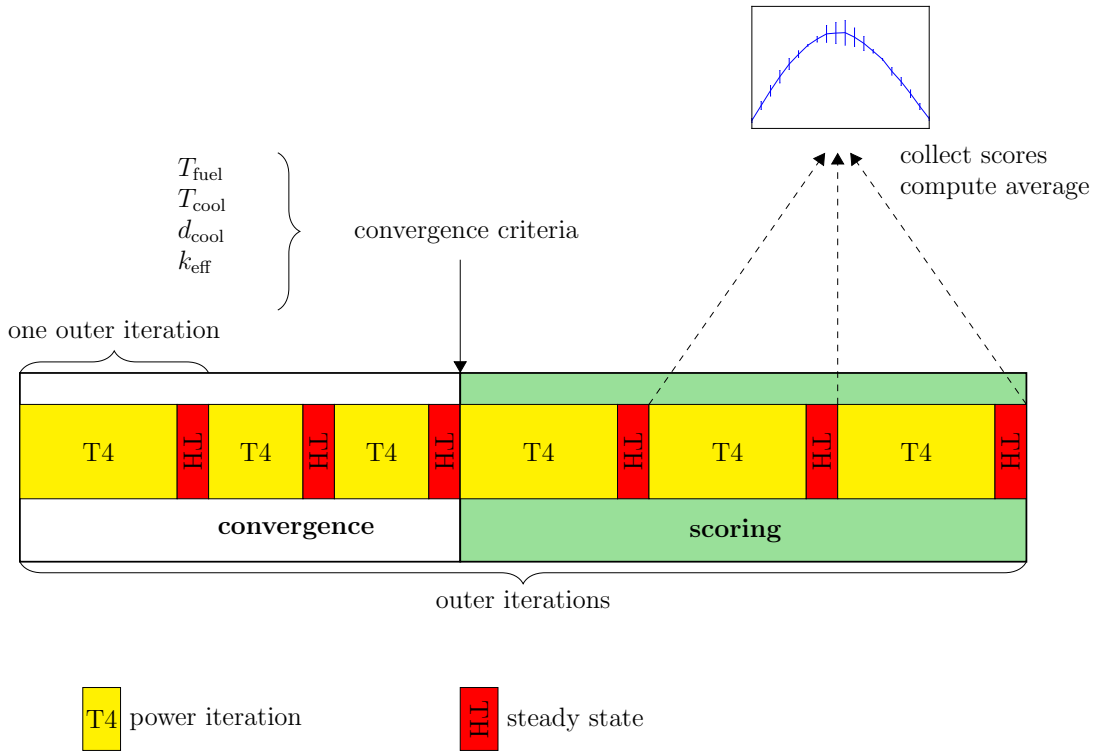


Figure 4.4 – Architecture of the outer iterations of the criticality coupling scheme. We iterate between TRIPOLI-4 and thermal-hydraulics (TH) until we satisfy some convergence criteria on the fuel temperature, coolant temperature, coolant density and multiplication factor. After convergence, additional iterations are performed, over which scores are collected.

Once convergence has been reached, we perform additional outer iterations, over which we score the power, temperatures, densities and the multiplication factor. At the end of the iterations, the different results (power, temperature and density fields) are averaged over the outer iterations. The calculation of the statistical uncertainties is however not straightforward, since the outer iterations are not independent. Indeed, correlations between iterations come from two sources. First, fission neutrons are transferred from one iteration to another. This is the same kind of correlations that exists in criticality calculations. Second, iterations are correlated through the thermal-hydraulics fields. Because of these correlations, the statistical uncertainty of the results is difficult to estimate. In fact, the only rigorous way to determine the real uncertainty would be to perform independent replicas of the coupling scheme, but this would induce very large computational times. Alternatively, we could preserve this scheme and combine it with the “blocking” (or “block-averaging”) method (Flyvbjerg and Petersen, 1989). With the blocking method, consecutive values are replaced by their average, so that correlations between samples are reduced, at the expense of reducing the sample size.

4.2.2 Storing source capability

In TRIPOLI-4, it is possible to store the fission sources in a file after the power iteration cycles: the position, energy, direction and weight of each particle are stored and can be used as the initial state for another calculation. This capability is especially valuable for dynamic calculations. Indeed, the source might be common to several transient scenarios and the convergence calculation need to be performed only once. Moreover, as we will explain in Sec. 4.4, criticality coupled calculations have poor parallel scalability.

4.2.3 Discussion on the choice of the simulation parameters

In the coupling scheme that we have developed for criticality calculations with feedback, the user must set the values of several simulation parameters, mainly the number of inner and outer iterations. The choice of the two parameters is discussed below.

Number of inner iterations

The data exchange process between TRIPOLI-4 and SUBCHANFLOW has a substantial computational cost, and it is better to ensure that the neutron source has converged on the new fundamental mode before performing the next outer iteration. Indeed, in order to minimize the number of rendez-vous points, we recommend to perform enough inner iterations to allow the fission source to converge to the fundamental mode corresponding to the updated temperature and density distributions.

The fission-source convergence actually depends on the convergence of the outer iterations (convergence of the thermal-hydraulics fields). Therefore, the number of inner iterations could be adapted along the outer iterations. However, in our work, for the sake of simplicity, we have chosen a constant number of inner cycles of 10, except for the first outer iteration, where 50 inner cycles are performed.

Number of outer iterations

The number of outer iterations for the convergence phase must be large enough to ensure convergence of the thermal-hydraulics fields. Typically, for most of our calculations, we have used 100 outer iterations, but we observed a posteriori that a smaller number would have been sufficient. For instance, as it will be shown later in Chapter 5 (see Fig. 5.2), convergence for the criticality calculation with feedback of the TMI-1 assembly was reached after about 10 outer iterations, which is consistent with the observations for deterministic coupled calculations on systems of similar size (Gomez-Torres et al., 2012).

4.3 Transient calculations

4.3.1 Description

Once the source has converged by means of the criticality calculation described in Sec. 4.2, the dynamic calculation can start. The scheme is similar to the one used for kinetic calculations, illustrated in Fig. 2.2, with SUBCHANFLOW calculations intercalated. The time-dependent coupling is illustrated in Fig. 4.5: we are formally applying the explicit Euler discretized scheme for the system of coupled neutron transport and thermal-hydraulics equations. This scheme is the simplest implementation of a time integration with external updates, and we could improve on it in order to speed up the convergence and enhance the stability of the scheme. However, we

must also keep in mind that the stochastic nature of TRIPOLI-4 makes it challenging to adapt higher-order deterministic schemes (for instance, the application of fixed-point algorithms becomes complex due to the presence of random fluctuations in the Monte Carlo calculations).

It should be emphasized that the dynamic scheme is performed successively to the criticality calculation with feedback. If TRIPOLI-4 fission sources are stored during a previous criticality calculation, it is possible to use them to start the dynamic calculations. For this purpose, an additional power iteration is performed before the first time step, during which neutrons and precursors are sampled according to Eqs. (2.3.2) and (2.3.8).

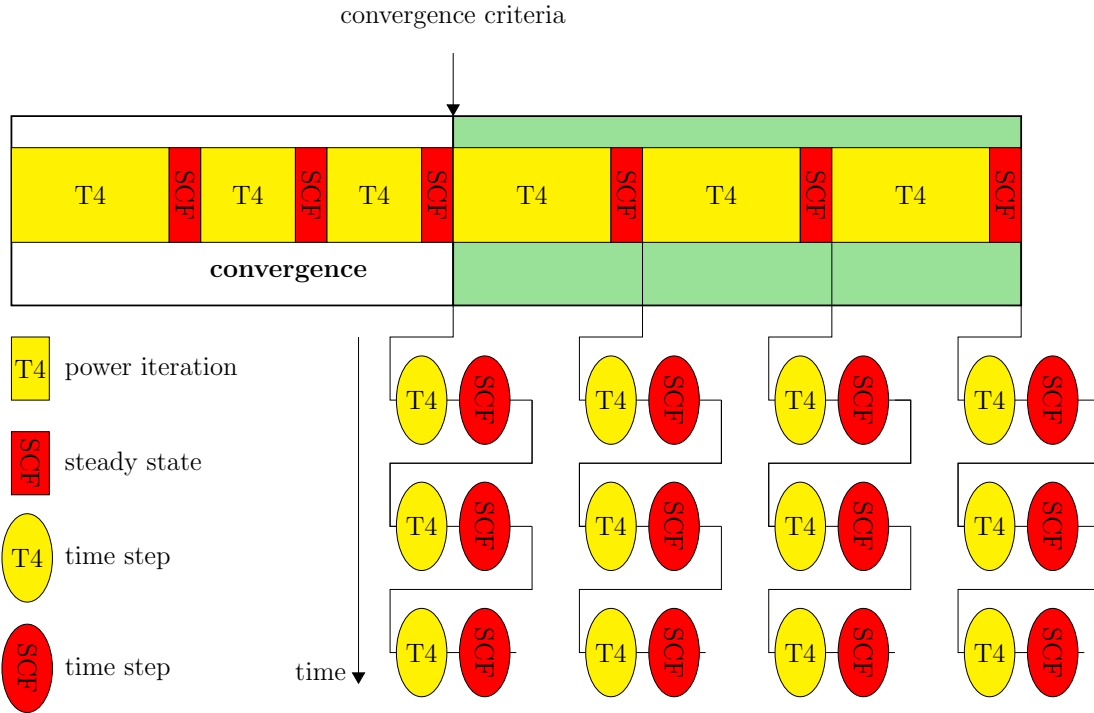


Figure 4.5 – Architecture of the dynamic coupling scheme. After the source has reached convergence during the criticality calculation illustrated in Fig. 4.4, TRIPOLI-4 computes the power distribution P_{fuel}^t for each time step t . The distribution is transferred to SUBCHANFLOW, which computes the corresponding transient state after the same time interval. The updated temperatures in the fuel rods and the coolant, as well as the new densities in the coolant, are transferred to TRIPOLI-4, which uses them as initial conditions for the following time step. The convergence phase can be skipped if source files have been previously generated.

4.3.2 Normalization between criticality and transient calculations

For the normalization factor between the criticality and the transient calculations, we use the source intensity normalization coefficient that was computed in the criticality calculation, according to Eq. (4.2.1). We also take into account the relative norm q , defined by Eq. (2.3.17), between the criticality and the kinetic calculations, which is computed during the sampling of neutrons and precursors. Also, the thermal-hydraulics solver needs a power as input, but TRIPOLI-4 kinetic scores are integrated over a time interval Δt . Hence, scores must be divided by the length of the time interval to obtain rates. Thus, the conversion factor r_t applied to the

scores at the end of each time step of the transient calculation is

$$r_t = \frac{r_s \times q}{\Delta t}. \quad (4.3.1)$$

4.4 Parallel calculations

The size of the systems that we want to simulate is such that coupled calculations are very time-consuming. Therefore, it was necessary to implement the capability for parallel execution in the supervisor. We describe here the architecture of the calculations in parallel mode.

4.4.1 Thermal-hydraulics “rendez-vous” point

As discussed in deeper detail in Chapter 7, thermal-hydraulics equations are non-linear and we do not have much knowledge about the way the statistical fluctuations stemming from TRIPOLI-4 will propagate along the outer iterations. Moreover, the result of the coupling scheme might be biased, as witnessed by the Jensen’s inequality:

$$f(\mathbb{E}(X)) \leq \mathbb{E}[f(X)], \quad (4.4.1)$$

with f a convex function representing the solution of the thermal-hydraulics problem and X a random variable representing the power field calculated by TRIPOLI-4¹. The symbol \mathbb{E} denotes the ensemble average. For these reasons, we have decided to minimize the statistical fluctuations on the input to the thermal-hydraulics solver. To this end, we have chosen to run SUBCHANFLOW once for all the processors, so that the stochastic results of the different parallel units are averaged. The statistics of the power distribution given to SUBCHANFLOW is increased and thus the possible bias minimized.

The results obtained by the different parallel units are correlated via the thermal-hydraulics fields, which are shared by all the simulators. Again, this complicates the assessment of the real statistical uncertainty. This situation contrasts with the typical “embarrassingly” parallel nature of the Monte Carlo calculations, in which the parallel units are truly independent, and the assessment of the uncertainty is straightforward. As discussed above, we prefer to increase the statistics given to the non-linear solver even if it comes at the expense of extra correlations. One way of improving our scheme would be to average the Monte Carlo results over packets of simulators. This way, we would get truly independent results, while averaging at the same time TRIPOLI-4 results over several simulators.

4.4.2 Role of the different parallel units

In the native parallel scheme of TRIPOLI-4, there are three types of parallel units. The “monitor” directs the parallel units; the “simulators” are in charge of the particle transport; finally, the “scorer” collects the scores sampled by the simulators.

We have adapted this scheme to the supervisor. One parallel unit, the monitor, is in charge of orchestrating the other TRIPOLI-4 parallel units, of running the supervisor and SUBCHANFLOW. Another parallel unit (the scorer) is in charge of collecting the scores. The simulators are in charge of the TRIPOLI-4 simulation. When all simulators have completed the inner cycles (in the case of a criticality calculation with feedback) or a time step (in the case of a transient calculation), the scorer collects the results and sends the averages to the monitor. The monitor casts

¹The sign of the inequality is reversed for concave f .

the simulation results in the form of MED fields, manipulates them (e.g., applies the appropriate normalization factor, performs the remapping between TRIPOLI-4 and SUBCHANFLOW meshes) and sends the power distribution to SUBCHANFLOW, before finally launching the SUBCHANFLOW calculation. We emphasize the fact that the SUBCHANFLOW run can only start when all the simulators have completed their task.

For parallel calculations with a large number of processors, fluctuations on the population size induce large fluctuations on the computational cost of the simulators. This is problematic, since the monitor must wait for all the simulators to finish; hence, the calculation is hampered by the slowest simulator.

When the SUBCHANFLOW calculation is over, the monitor receives the resulting temperature and density distributions in MED format. Again, the monitor manipulates the data (e.g., remaps them from SUBCHANFLOW to TRIPOLI-4 meshes and projects them to the TRIPOLI-4 volumes). With the new temperatures, the monitor loads the new cross sections, if needed. New compositions (with updated temperatures and densities) are serialized to the scorer and to the simulators, which in turn load the new cross sections. A new cycle (or time step) can then begin. The architecture of a parallel criticality calculation with feedback is illustrated in Fig. 4.6 for four simulators. In this example, simulator 3 is slower than the others and slows down the simulation. On the contrary, simulator 2 finishes first and stays inactive until the other simulators have finished their task.

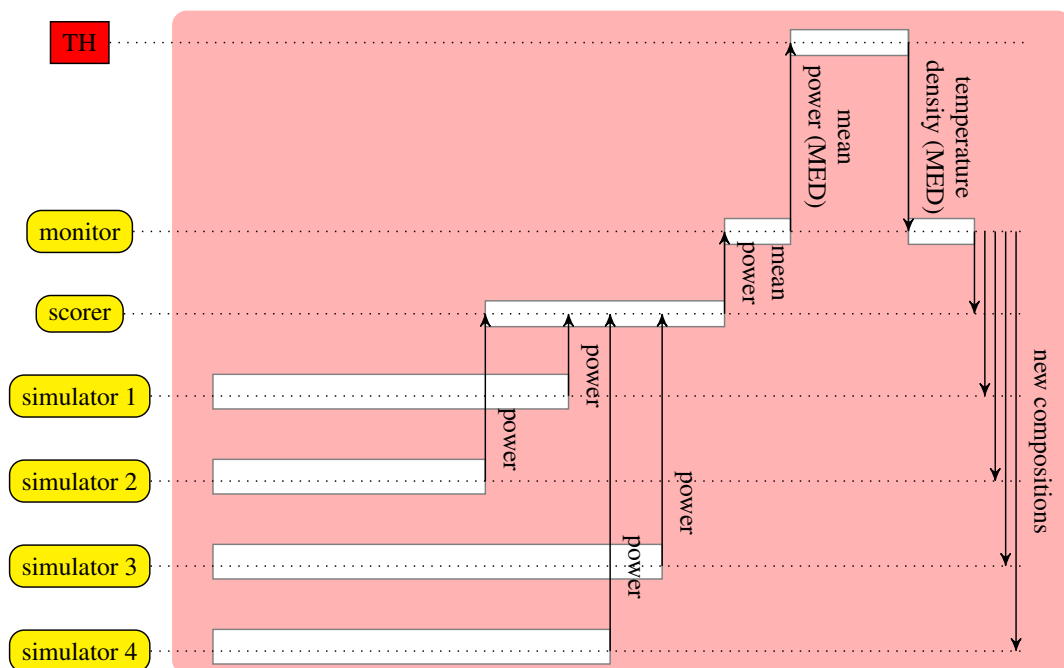


Figure 4.6 – Architecture of the coupled iteration, in parallel mode with four simulators. When all the simulators have completed their task, the scorer collects the results and transfers the average to the monitor. After manipulating the fields (normalization and remapping), the monitor launches the SUBCHANFLOW calculation. Hence, SUBCHANFLOW is run once for all simulators so as to reduce the statistical fluctuations on its input data. With SUBCHANFLOW results, the monitor updates the compositions and sends them to the scorer and the simulators. All the parallel units then load new cross sections, if necessary.

Note that, as in usual criticality calculations, each simulator must independently achieve convergence of the fission sources. The required work scales as the number of simulators, and therefore the convergence phase does not benefit from parallelization. In contrast, during the scoring phase, the calculation results are averaged and the statistics collected in a given wall-clock time scales with the number of processors, provided that the simulation is efficient.

Also, in a criticality calculation run in parallel mode, each simulator stores its own fission sources. Therefore, for an optimal use, the dynamic calculation should be launched with the same number of simulators, each using a different source file.

4.4.3 Memory footprint

Dynamic calculations require a large amount of memory, mainly for nuclear data and compositions. Since compositions are associated to volumes in TRIPOLI-4, the geometries generally consist of a large number of volumes and are also memory-consuming (about 10^5 volumes for a single geometry of the TMI-1 3x3 mini-core described in Sec. 3.2). Moreover, as explained in Sec. 2.8, the capability of TRIPOLI-4 to handle time-dependent geometries is not optimal: TRIPOLI-4 requires as many geometries as discretization steps.

Most of the memory is allocated during the initialization phase (mainly TRIPOLI-4 and SUBCHANFLOW initialization). After that, the memory use is roughly stable. Indeed, the different geometries and nuclear data corresponding to the initial state are loaded at the beginning. Some new nuclear data may be loaded in the following, but the calculation generally covers a large range of temperatures as of the initialization phase. Nonetheless, the initialization is performed by each parallel unit, which results in a large total memory occupation; no memory is shared among parallel units within the same node.

To conclude, the memory footprint represents a limitation for massively parallel dynamic calculations with TRIPOLI-4, although future improvements on the management of the geometry could allow to mitigate this issue.

4.5 Conclusion

The external coupling between TRIPOLI-4 and SUBCHANFLOW is now operational, thanks to the development of a multi-physics interface for TRIPOLI-4, which allows a supervisor to control TRIPOLI-4 via an ICoCo API. Data exchange between the two codes is performed in memory using the MEDCoupling library. The development of the supervisor was performed within the frame of this thesis.

For criticality calculations with feedback, care must be taken regarding the simulation parameters (for instance, the number of outer iterations and the number of inner iterations). For transient calculations, care must be taken regarding the memory footprint, especially in the case of a massively parallel execution.

Verification tests of the TRIPOLI-4 multi-physics capabilities will be presented in Chapter 5, with a criticality calculation. Examples of dynamic calculations will be presented in Chapter 6.

Chapter 5

Verification of the coupling for criticality simulations with feedback

We have set up a benchmark configuration within the McSAFE project, in a joint work with the Serpent and the SUBCHANFLOW development teams, in order to assess the recently developed multi-physics capabilities of Monte Carlo codes. In this chapter we probe the multi-physics capabilities of TRIPOLI-4 for criticality calculations with thermal-hydraulics feedbacks. Calculation results on the TMI-1 assembly were published in Faucher et al. (2019b). The present work, combined with the kinetic method described in the first part of this thesis, is a stepping stone towards the analysis of operational and accidental transients, discussed in Chapter 6.

5.1 Introduction on the benchmark work

The idea behind the McSAFE benchmark is to develop a comparison for global and detailed parameters using independent calculations stemming from different approaches, calculation schemes and neutron transport codes, with a given thermal-hydraulics code. The objective is to perform a criticality calculation with feedback on the fuel assembly based on the TMI-1 reactor, with a pin-by-pin description. For this purpose, we have considered the couplings between two Monte Carlo neutron-transport codes, TRIPOLI-4 and Serpent 2, and the sub-channel code SUBCHANFLOW.

In the following, the coupling scheme between TRIPOLI-4 and SUBCHANFLOW is referred to as “T4/SCF” and the coupling scheme between Serpent 2 and SUBCHANFLOW is referred to as “SSS2/SCF”.

5.2 Description of the models used by the different codes

5.2.1 TRIPOLI-4 and Serpent 2 models

Geometry

The TMI-1 fuel assembly models used by T4 and SSS2 are very similar. The axial length of 353.06 cm is divided into 20 slices. Radial reflective boundary conditions are applied. All compositions have the same temperature at the beginning of the simulation: 900 K for the 204 enriched UOX pins and the 4 burnable poison pins, and 563 K for the coolant channels, the guide tubes and the instrumentation tubes. All the fuel and coolant compositions are individualized in the T4 geometry, so as to be able to independently update their temperatures and densities.

The critical boron concentration for the TMI-1 3x3 mini-core, detailed in Sec. 3.2.2, has been used. Since we consider in this chapter the reflected central assembly only, which does not contain control rods, the 1480 ppm of boron do not lead to a critical configuration and the multiplication factor is larger than 1.

Scores

T4 computes fission rates, while SSS2 computes a deposited fission energy. Both are converted to a power and normalized. There is a slight difference between the SSS2 score and the fission rate computed by T4. Indeed, in SSS2 for each fission the deposited energy is scaled with the fission Q-value for the fissioning isotope (Tuominen et al., 2019). Since our system only contains uranium-235 and uranium-238, and since the T4 and SSS2 power maps are normalized to the same integral value, the relative difference in power due to the different scoring strategy is thus of the order of the relative difference between the fission Q-values for uranium 235 and uranium-238 (about 3%) multiplied by the fraction of fissions on uranium-238 (also about 3%), which is equal in total to about 1‰.

5.2.2 SUBCHANFLOW model

Geometry

The SCF preprocessor generates complex geometries for SCF at different levels: pin, assembly or core level. We have used the preprocessor to generate the geometry of the assembly with a pin-by-pin description. Two meshes are generated: a fuel-centered mesh for receiving the power, and a coolant-centered mesh for the resolution of the equations. Both meshes are divided into 20 axial slices, thus the total number of cells for the fuel-centered mesh is $15 \times 15 \times 20$ and $16 \times 16 \times 20$ for the coolant-centered mesh (for illustration, see Fig. 5.1). The preprocessor produces the meshes in MED format to be used by the T4/SCF coupling scheme.

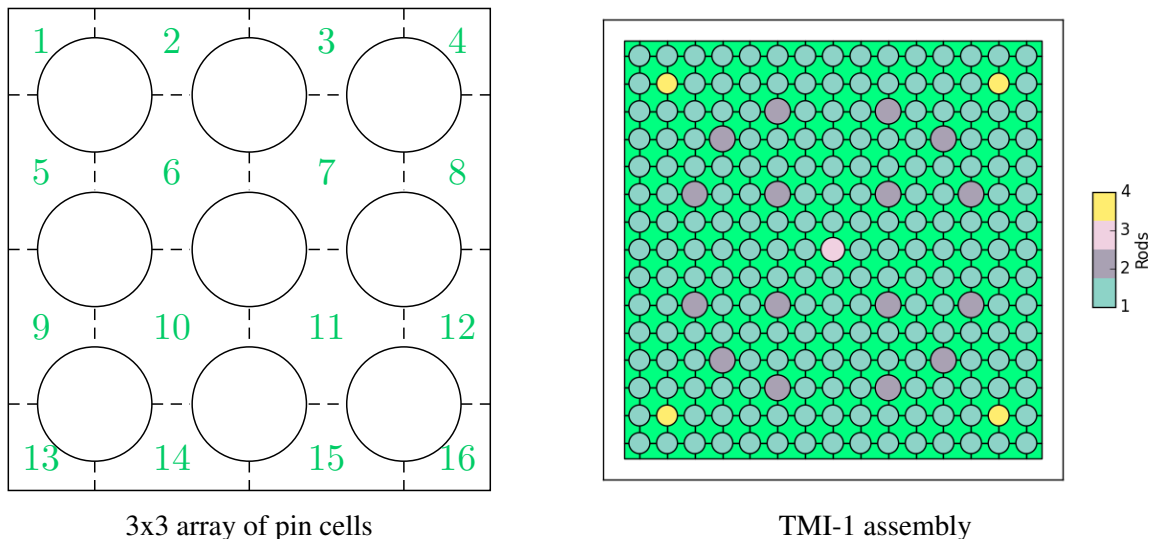


Figure 5.1 – Coolant-centered channels for the SCF model of the TMI-1 assembly. Left: example of numbering of a coolant-centered mesh for a 3x3 array of pin cells (the total number of cells per slice is 4×4). Right: the TMI-1 assembly (the total number of cells per slice is 16×16). Rods 1 are fuel rods, rods 2 are guide tubes, rod 3 is the instrumentation tube and rods 4 are burnable poison pins.

coupling scheme	T4/SCF	SSS2/SCF
coolant inlet temperature [K]	565	565
outlet pressure [Pa]	15.5132	15.5132
mass flow rate [kg s ⁻¹]	85.96	85.96
number of axial slices	20	20
fuel temperature model	volume averaged	volume averaged
total power [MW]	15.66	15.66
temperature dependence of the cross sections	stochastic interpolation	target motion sampling
number of particles per cycle	7×10^4	10^5
number of inactive cycles	50	50
number of inner iterations before convergence (M)	10	3000
number of inner iterations after convergence (M')	100	N.A.
number of outer iterations before convergence (n)	100	11
number of outer iterations after convergence (n')	100	0
data exchange	in memory (MED)	external files
simulation time (CPU.h)	210	820

Table 5.1 – Operating conditions of the TMI-1 assembly, and simulation parameters for the two coupling schemes, T4/SCF and SSS2/SCF.

Simulation parameters

The coolant inlet temperature is set to 565 K, the outlet pressure is 15.51 MPa and the inlet mass flow rate is 85.96 kg s⁻¹. The power is normalized to 15.66 MW. The operating conditions are summarized in Table 5.1.

We assume that all the power is deposited in the fuel. In order to compute the fuel rod temperature, each axial slice of the fuel is divided into ten radial rings and the heat diffusion equation is solved with a finite-volume method. The fuel temperature returned by SCF is the the volume average over the radial nodes.

5.3 Comparison between TRIPOLI-4 and Serpent 2 without feedback

As a preliminary step, we have performed criticality calculations with the two Monte Carlo codes without feedback, in order to verify the equivalence of the neutronics models before independently coupling them to SCF. Results for the multiplication factor are in good agreement: we have obtained $k_{\text{eff}} = 1.24140 \pm 2 \times 10^{-5}$ for T4 and $k_{\text{eff}} = 1.24102 \pm 3 \times 10^{-5}$ for SSS2, respectively (see Table 5.2). The discrepancy is less than 40 pcm and is statistically significant (about 12σ).

We have singled out three main differences in the models used by T4 and SSS2. First, the temperature dependence of the cross sections is not accounted for in the same way: stochastic square-root interpolation is used in T4, while SSS2 uses target motion sampling (see Sec. 4.1.3). However, this probably does not entirely explain the 40 pcm discrepancy, considering the 50 K intervals used by T4 for the tabulated data. Indeed, if we had taken the closest temperature (which is at 25 K at maximum), the deviation would be less than 50 pcm, assuming an average Doppler effect of 2 pcm/K. Since we have used stochastic interpolation, the deviation due

	T4	SSS2	Δk_{eff} Monte Carlo
without SCF	$1.24140 \pm 2 \times 10^{-5}$	$1.24102 \pm 3 \times 10^{-5}$	+38 pcm
with SCF	$1.23754 \pm 5 \times 10^{-5}$	$1.23664 \pm 3 \times 10^{-5}$	+90 pcm
Δk_{eff} SCF	-386 pcm	-438 pcm	

Table 5.2 – Multiplication factor for the calculations without and with feedback for the two schemes, for the TMI-1 assembly.

to the temperature dependence of the cross sections should be in principle smaller. Second, the geometries for T4 and SSS2 were built independently, and there might be slightly different modeling choices between them. Finally, it should be noted that the standard error on the multiplication factor computed by the codes is underestimated because of correlations between batches, so the true error bar might be considerably larger than 2 pcm for T4 and 3 pcm for SSS2.

The axial power profile is displayed in Fig. 5.9 (dashed lines) for one rod (the rod is identified in Fig. 5.6), showing that T4 and SSS2 calculations are in very good agreement.

After this verification, we can now examine the coupling of the two Monte Carlo codes with the same thermal-hydraulics code, SCF.

5.4 Description of the coupling schemes

5.4.1 Architecture

For a precise description of the coupling scheme between T4 and SCF, we refer the reader to Sec. 4.2. At the time of this work, the T4 supervisor capability for parallel running was not fully implemented yet: in the following, we will thus present results from a sequential run of T4/SCF, while SSS2/SCF calculation was run in parallel.

The initial guess for the source has an axial cosine distribution. The T4 calculation begins with 50 extra inactive cycles. Then, coupling iterations begin and SCF is run every $M = 10$ neutron transport cycles (the “inner” iterations) in order to reach the fundamental mode. The SSS2/SCF scheme is similar, except that no extra cycles are discarded at the beginning of the simulation and $M = 3000$ cycles are run per outer iteration.

In SSS2/SCF scheme, unlike in T4/SCF, relaxation is performed on the temperatures and densities, with the same coefficient $\alpha = 1/2$ (see Eq. (4.2.2)).

5.4.2 Convergence criteria

The outer iterations stop when convergence criteria, presented in Eq. (4.2.3), are met. For the residual value introduced in Eq. (4.2.3), we have chosen a convergence criterion $\varepsilon_{T_{\text{fuel}}} = 5$ K for the rod temperature, $\varepsilon_{T_{\text{coolant}}} = 1$ K for the coolant temperature, $\varepsilon_{d_{\text{coolant}}} = 1$ g/m³ for the coolant density and $\varepsilon_{k_{\text{eff}}} = 10$ pcm for the multiplication factor. Once the convergence criteria were met, we have averaged the simulation over $n' = 100$ extra outer iterations with T4/SCF, while the SSS2/SCF calculation was terminated. Thus, the reported SSS2/SCF uncertainties are not

representative of the fluctuations in the thermal-hydraulics fields, and surely underestimate the actual uncertainty. The different coupling parameters are specified in Table 5.1.

5.5 Benchmark results with feedback

5.5.1 Convergence

The convergence criteria, as defined by Sec. 5.4.2, were met for the fuel rod temperatures, the coolant temperatures and densities and the multiplication factor. Figure 5.2 shows the evolution of the residual for the fuel temperature, which has the slowest convergence rate as compared to coolant temperature, coolant density and multiplication factor. The residual value quickly converges: it gets below 5 K within about 10 iterations for both schemes, which is consistent with the values used by deterministic coupled schemes (Gomez-Torres et al., 2012).

5.5.2 Coolant temperatures and densities

The map for the radial (i.e., axially-averaged) temperatures obtained for the coolant with T4/SCF is presented in Fig. 5.3. On this map, six channels are labelled A to F. Figure 5.4 presents the axial coolant temperature and density profiles for the different channels. Coolant temperatures range from 565 to 600 K and densities range from 660 to 743 kg/m³. We have also computed the standard error over the active iterations for all the channels. The standard error obtained is about 0.02 K on average for the coolant temperatures, and about 0.1 kg/m³ for the coolant densities, suggesting a good convergence of the coolant fields.

The results obtained with SSS2/SCF are also presented. The two coupling schemes are in very good agreement, as confirmed also by Fig. 5.5, which presents the difference between the coolant temperatures coming from the two coupling schemes divided by the standard error obtained with T4/SCF calculation. The difference is larger at the top of the assembly because the inlet temperature is imposed (cf. Table 5.1). The standard error computed for T4/SCF calculation might be underestimated, as expected because of the correlations (see Sec. 4.2). Still, the difference between the two coupling schemes is small.

5.5.3 Fuel temperatures

The map for the axially-averaged fuel temperature of the 225 rods obtained with T4/SCF is presented in Fig. 5.6. The minimum value is reached in the guide tubes and the instrumentation tube, where there is no fission. The 4 burnable poison pins also have low temperatures because of the neutron captures in gadolinium, leading to a lower fission rate. The standard error is about 2 K on average for the fuel temperature, which converges more slowly than the coolant temperature. Four rods are identified on the map: rods number 1, 3 and 4 are fuel pins, with temperatures ranging from 630 to 1150 K while rod number 2 is a burnable poison pin, with temperatures ranging from 600 to 730 K. For this calculation, cross sections were loaded for 15 different fuel temperatures.

The axial temperature profiles obtained with the two coupling schemes in these rods is presented in Fig. 5.7, and their difference divided by the standard error from T4/SCF calculation is plotted in Fig. 5.8. The two coupling schemes are in good agreement. As for the coolant fields, the standard error might be underestimated because of correlations between the iterations.

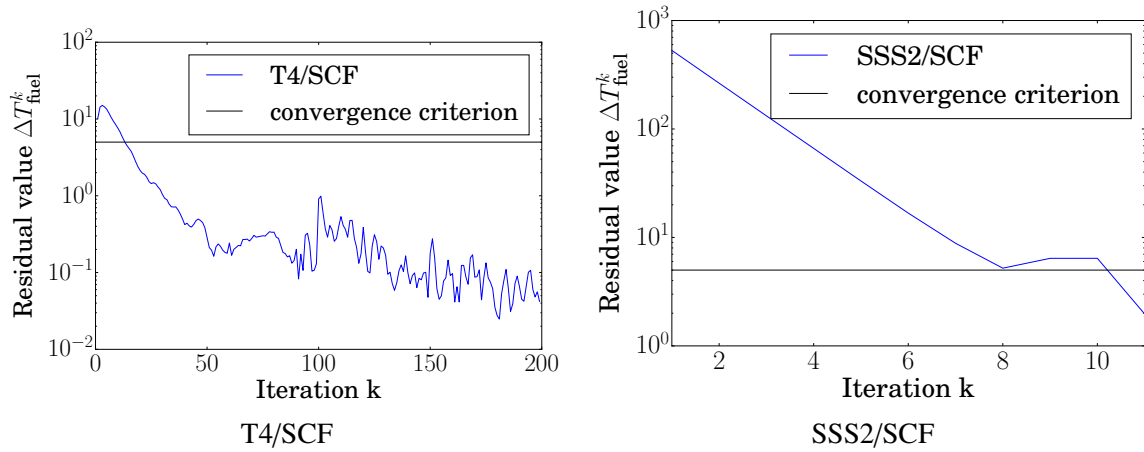


Figure 5.2 – Evolution of the fuel temperature convergence for the TMI-1 assembly, for the two coupled calculations. SSS2/SCF performs fewer outer iterations than T4/SCF, with more inner iterations (3000 instead of 10 for T4/SCF).

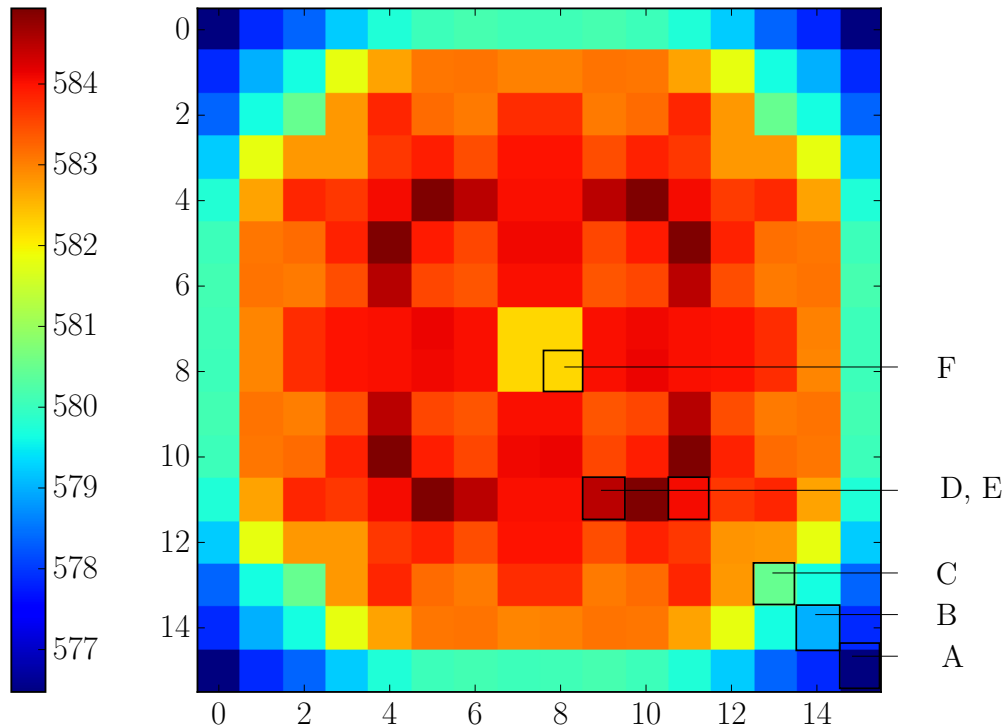


Figure 5.3 – Coolant temperature map (in kelvins) obtained with T4/SCF for the TMI-1 assembly, averaged across all axial slices.

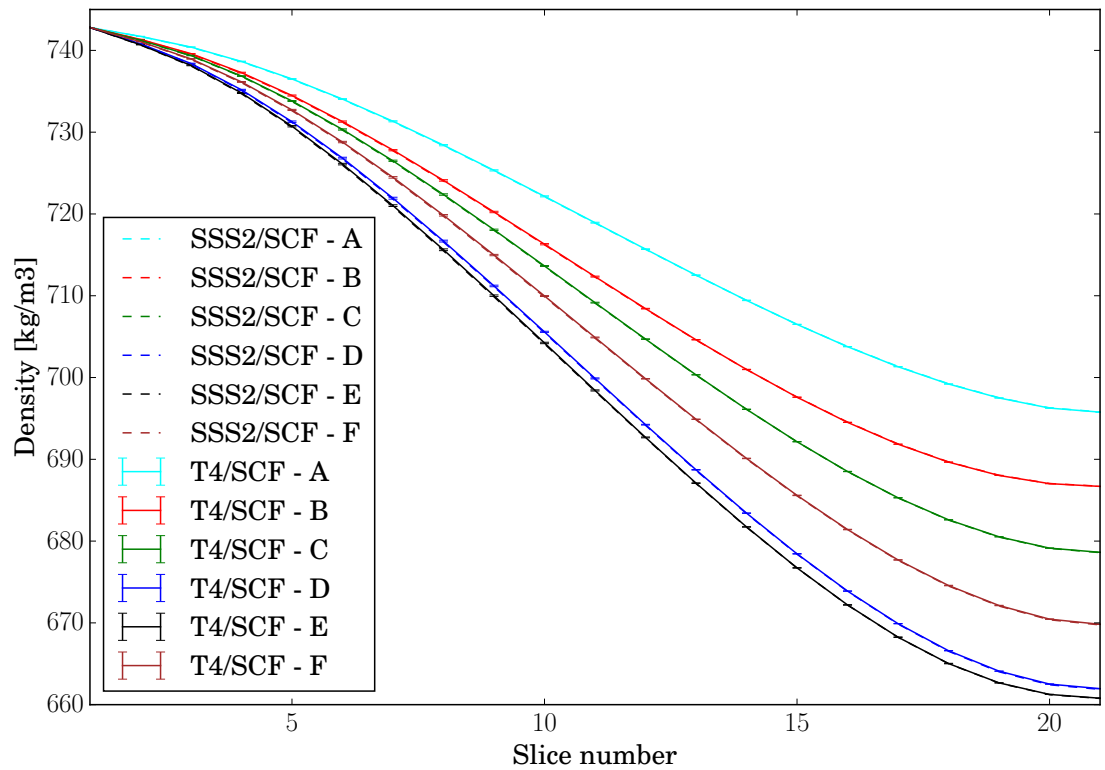
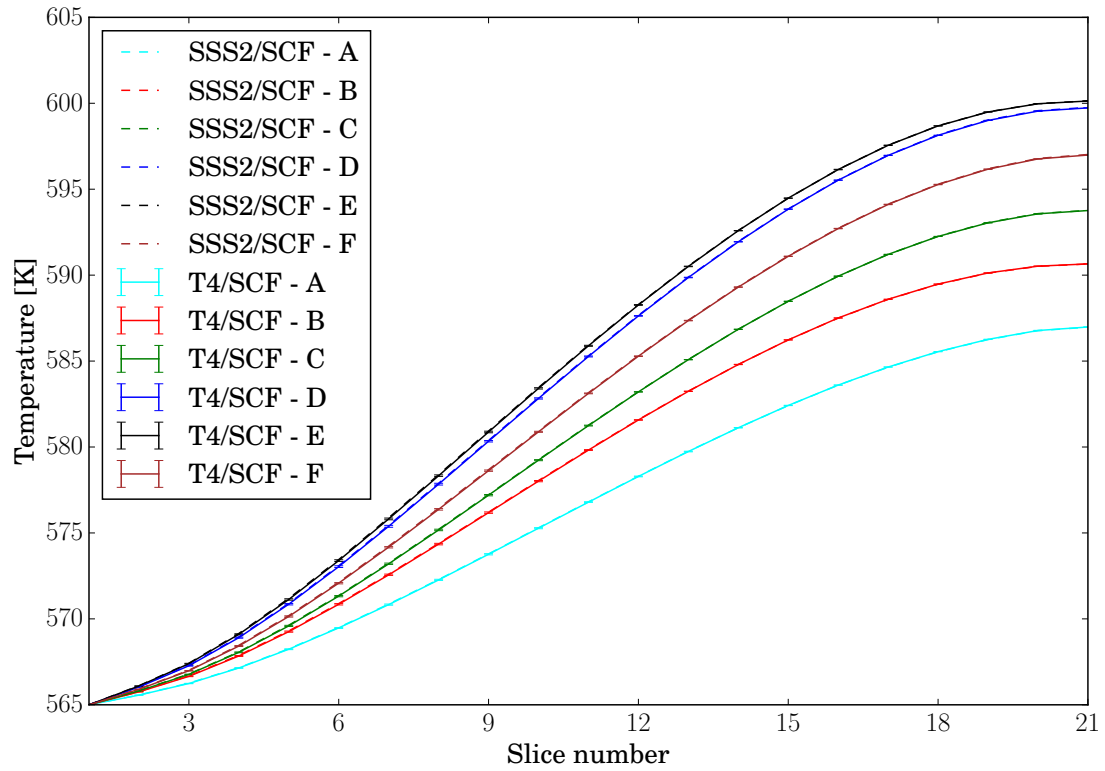


Figure 5.4 – Coolant temperatures and densities obtained with T4/SCF (solid lines) and SSS2/SCF calculations (dashed lines), for the six channels of the TMI-1 assembly identified in Fig. 5.3, at the inlet of each slice. Slice number 1 is located at the bottom of the assembly. Top: coolant temperature profiles in 6 channels. Bottom: coolant density profiles in 6 channels.

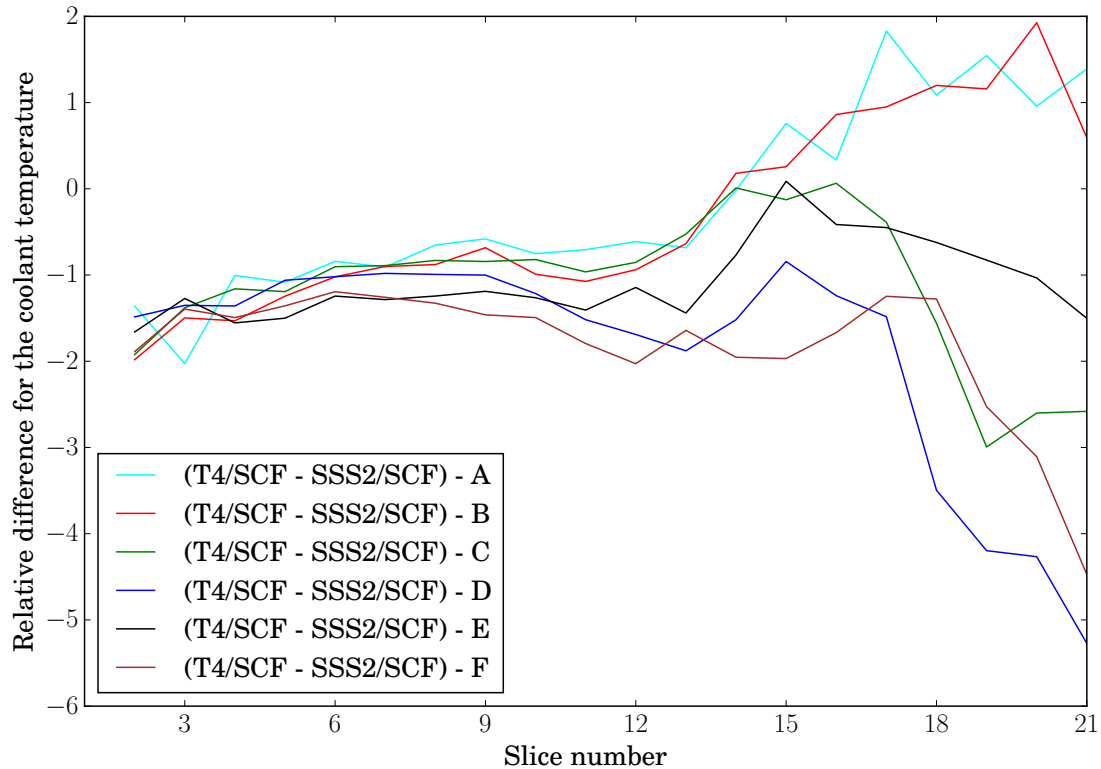


Figure 5.5 – Difference between axial coolant temperature profiles obtained with T4/SCF and SSS2/SCF divided by the estimated standard error for T4/SCF calculation.

5.5.4 Comparison between calculations with and without feedback

In order to illustrate the effects of thermal-hydraulics feedbacks on neutron transport, we have compared the coupled simulations with the Monte Carlo codes and SCF to the simulations with the Monte Carlo codes only. Table 5.2 details the values of the multiplication factor obtained with the two types of calculations; it is lower for the coupled calculations: 252 pcm for T4/SCF and 285 pcm lower for SSS2/SCF. The discrepancy between T4/SCF and SSS2/SCF is about 90 pcm, which is larger than the 38 pcm obtained without feedback, but still in the same order of magnitude.

Figure 5.9 shows the axial power profile for the rod number 1 (as defined in Fig. 5.6), with and without feedback. The T4 and SSS2 calculations, with and without coupling are in very good agreement. It is interesting to see that the shape of the power profile is flatter in the calculation with feedback, which was expected on physical grounds.

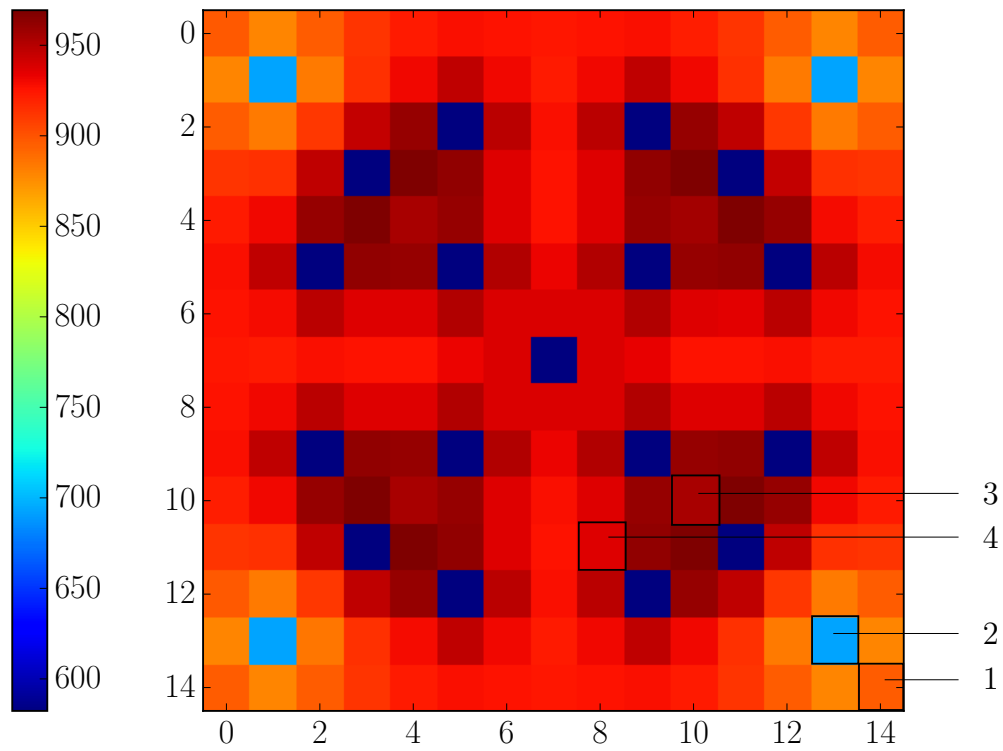


Figure 5.6 – Fuel temperature map (in kelvins) obtained with T4/SCF for the TMI-1 assembly, averaged across all axial slices.

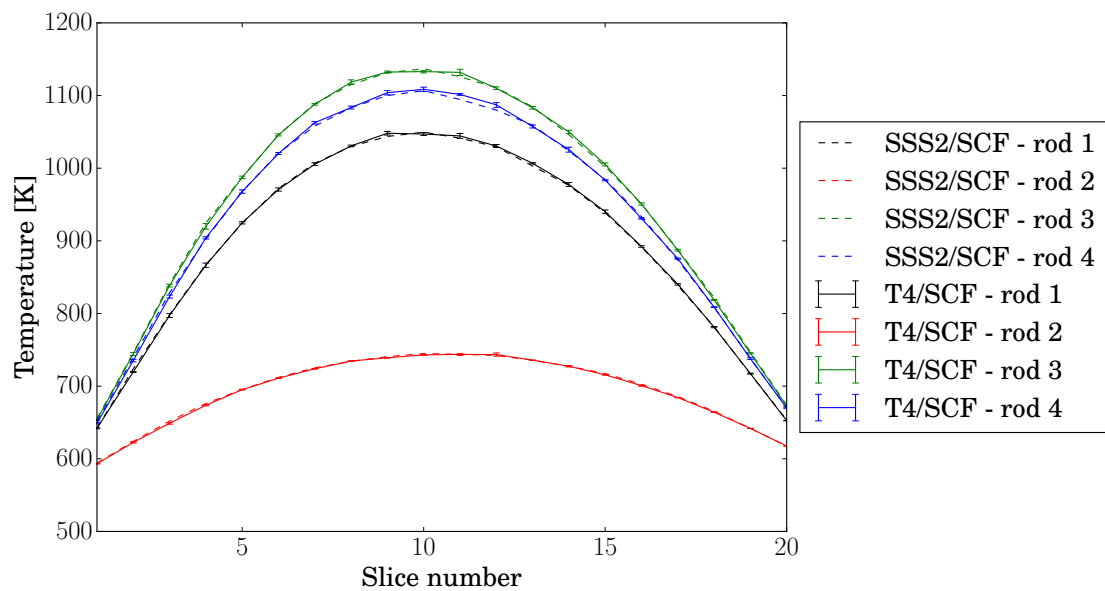


Figure 5.7 – Axial fuel temperature profiles obtained with T4/SCF (solid lines) and SSS2/SCF (dashed lines).

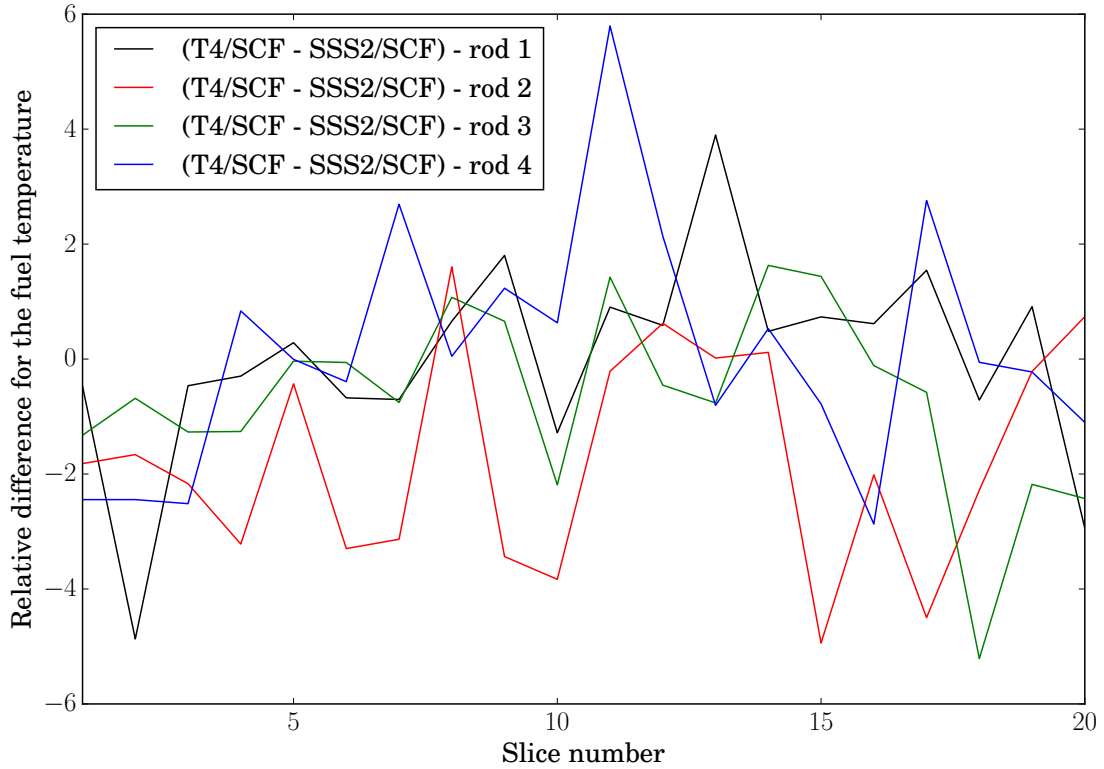


Figure 5.8 – Difference between axial fuel temperature profiles obtained with T4/SCF and SSS2/SCF divided by the estimated standard error for T4/SCF calculation.

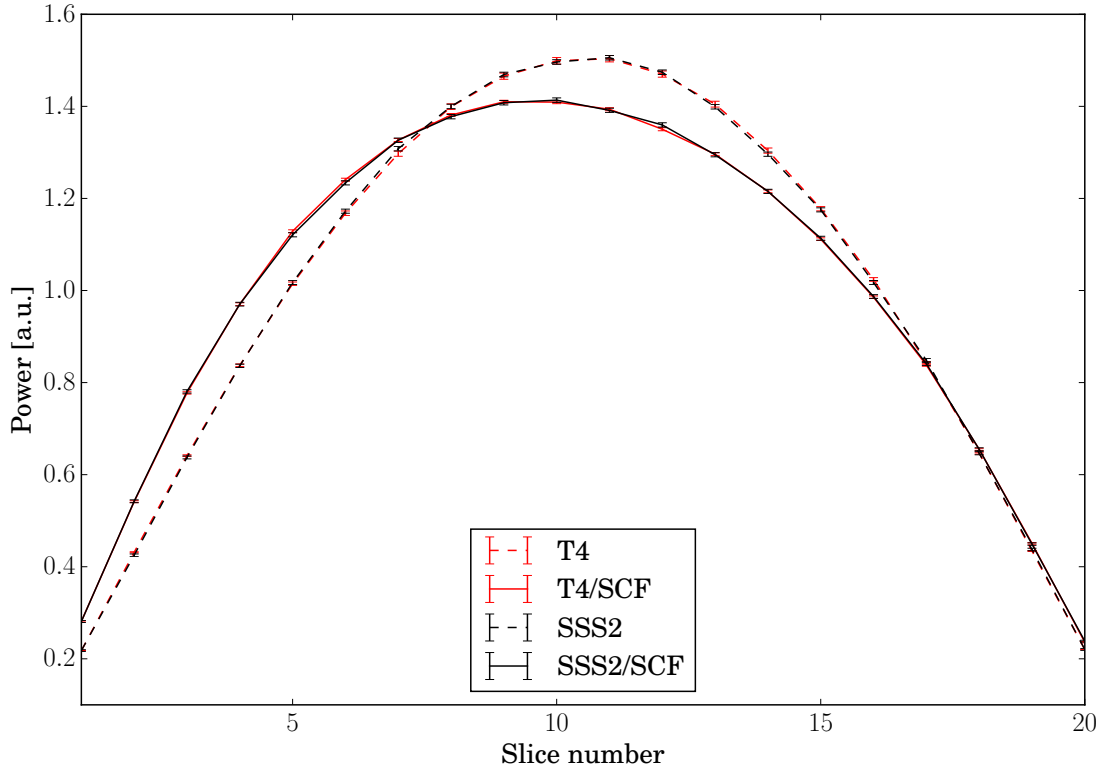


Figure 5.9 – Axial power profile for rod 1 of the TMI-1 assembly computed by T4 with an isothermal calculation without feedback (dashed line) and with SCF feedback (solid lines).

5.6 Conclusion

In this chapter we have verified the capabilities of TRIPOLI-4 for the multi-physics calculation of an assembly based on the TMI-1 core: this was achieved by proposing a benchmark work in collaboration with the Serpent and SUBCHANFLOW development teams. The power, temperature and density profiles are in very good agreement between the two schemes, and encourage further simulations with the new coupling scheme between TRIPOLI-4 and SUBCHANFLOW. The results of the benchmark were published in Faucher et al. (2019b).

The aim of this work was to perform a comparison between two coupling schemes with given simulation parameters. For further physical analysis of the system, these parameters should be subject to investigation, such as the number of axial slices, or the radial discretization of the fuel rods. Also, power deposition in the fuel and the coolant could be distinguished. In general, the spatial discretization choices should be carefully examined.

The next step will consists in combining the multi-physics capabilities with the kinetic Monte Carlo methods in order to perform time-dependent coupled calculations and simulate transients with thermal-hydraulics feedbacks.

Chapter 6

Testing the coupling with dynamic simulations

Now that we have verified the multi-physics capabilities of TRIPOLI-4 in Chapter 5, with a criticality calculation with thermal-hydraulics feedback, we present dynamic calculations, performed on the TMI-1 3x3 mini-core for different scenarios.

6.1 Preliminary criticality calculations with feedback

The preliminary step prior to any dynamic calculation on the TMI-1 3x3 mini-core introduced in Sec. 3.2 is to compute the initial state. For this purpose, the same scheme as the one used for the TMI-1 assembly in Chapter 5 is used. For the mini-core consisting of 9 fuel assemblies considered here, the power is normalized to 140.94 MW. Also, we have increased the number of axial slices from 20 to 30, in light of the long active height of 353.06 cm. The aim of the criticality calculation with feedback is to provide the source for dynamic calculations. Thus, the system must be prepared on a critical state. Therefore, we have performed an additional preliminary calculation, in order to compute the critical boron concentration.

6.1.1 Critical boron search

To compute the critical boron concentration, we have performed a criticality calculation with feedback, starting from the boron concentration of the isothermal calculation, 1480 ppm. In this calculation, we have used the critical boron search option implemented in T4 which is exposed to the supervisor via the T4.Facade class. The resulting critical boron concentration is (1305.5 ± 5.1) ppm (the standard error was estimated over 10 independent simulators), which is smaller than the “isothermal” concentration, as expected. Indeed, as already observed in Table 5.2 for the assembly, the thermal-hydraulics feedback absorb some reactivity, thus less boron is needed to ensure a critical configuration. Here, the difference accounts for about (1400 ± 40) pcm if we consider a differential boron worth of about 8 pcm/ppm.

6.1.2 Source for the dynamic calculations

Now that the critical boron concentration has been found, we have performed another criticality calculation with feedback, with the new concentration of 1305.5 ppm in order to compute the equilibrium thermal-hydraulics fields and the fission sources for the dynamic calculations. Support for parallelism in the supervisor was developed at the time of this work and the calculation was performed using 1000 processors on the Cobalt cluster at the TGCC (Très Grand Centre de

Calcul, Bruyères-le-Châtel, France). The multiplication factor is $k_{\text{eff}} = 1.00018 \pm 8 \times 10^{-5}$, showing that the system is close to a critical state. We have stored the resulting thermal-hydraulics fields: fuel temperatures, coolant temperatures and coolant densities. The fields are presented below. The fission sources (position, energy, direction and weight of the fission neutrons) were also stored at the end of the calculation.

Coolant temperatures and densities

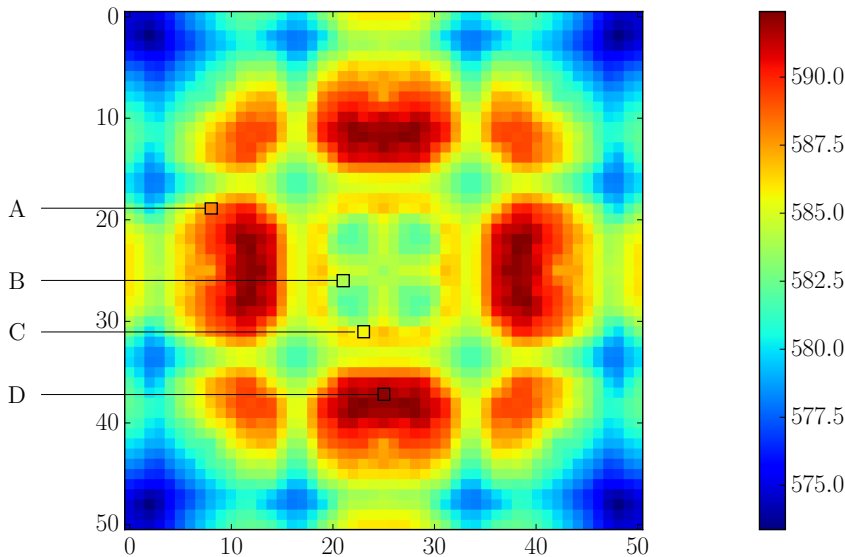


Figure 6.1 – Coolant temperature map (in kelvins) obtained with T4/SCF for the critical state of the TMI-1 3x3 mini-core, averaged across all axial slices. The temperatures are used as initial values for the dynamic calculations. Four cells are identified on the map for later analysis of simulation results.

The radial map for the axially-averaged temperatures obtained for the coolant is shown in Fig. 6.1. Temperatures range from 565 K to 608 K and are higher in the unrodded assemblies, as expected. Four cells (A to D) are identified on the map for later analysis of simulation results. The radial map for the axially-averaged coolant densities is shown in Fig. 6.2. Coolant densities range from 635 kg/m³ to 743 kg/m³.

Fuel temperatures

The map for the axially-averaged fuel temperatures is shown in Fig. 6.3. Temperatures range from 565 K to 1590 K. Again, temperatures are higher in the unrodded assemblies. The control rods, guide tubes and burnable poison pins are easy to recognize because of their low temperature. Four rods (1 to 4) are identified on the map for later analysis of simulation results.

Neutron power

Figure 6.4 presents the axial power profile for the rod 1 identified in Fig. 6.3 obtained without (dashed line) and with feedback (solid line). The profile obtained with the calculation with feedback is very asymmetric: the power peak is shifted towards the bottom of the mini-core.

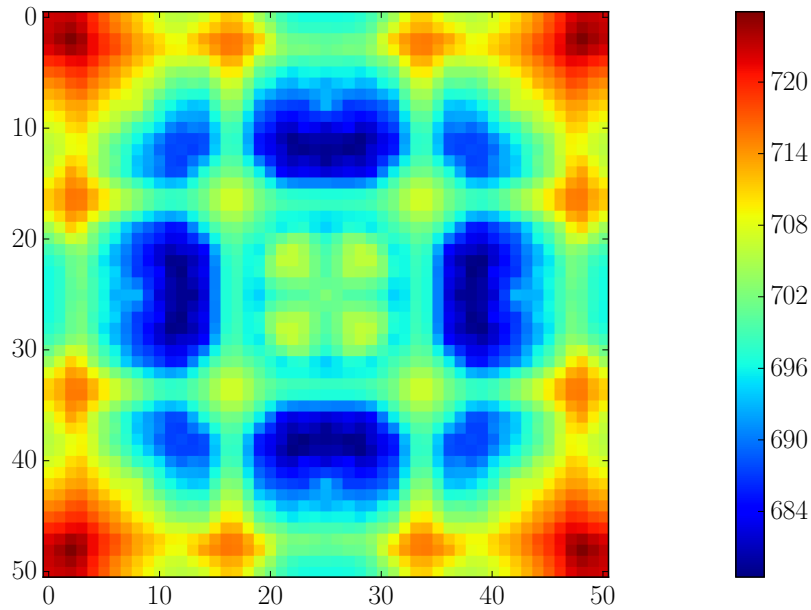


Figure 6.2 – Coolant density map (in kg/m^3) obtained with T4/SCF for the TMI-1 3x3 mini-core, averaged across all axial slices. The densities are used as initial values for the dynamic calculations.

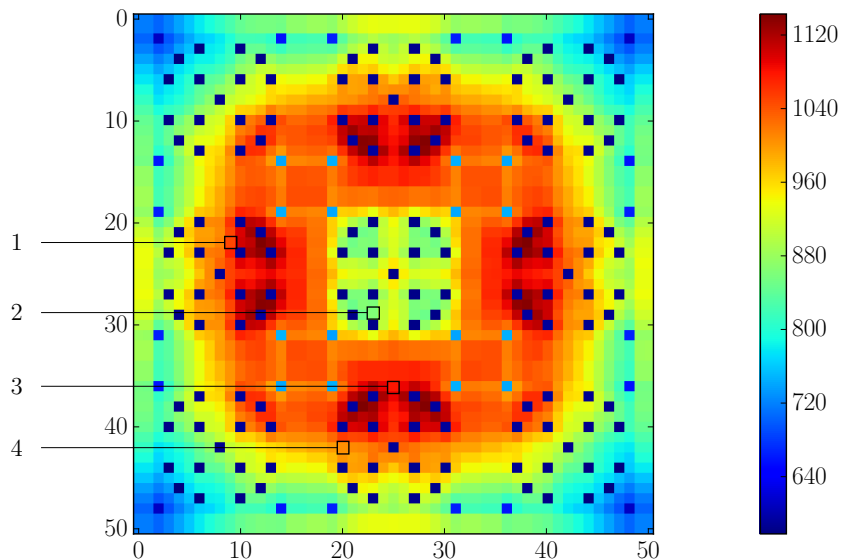


Figure 6.3 – Fuel temperature map (in kelvins) obtained with T4/SCF for the TMI-1 3x3 mini-core, averaged across all axial slices. The temperatures are used as initial values for the dynamic calculations. Four rods are identified on the map for later analysis of simulation results.

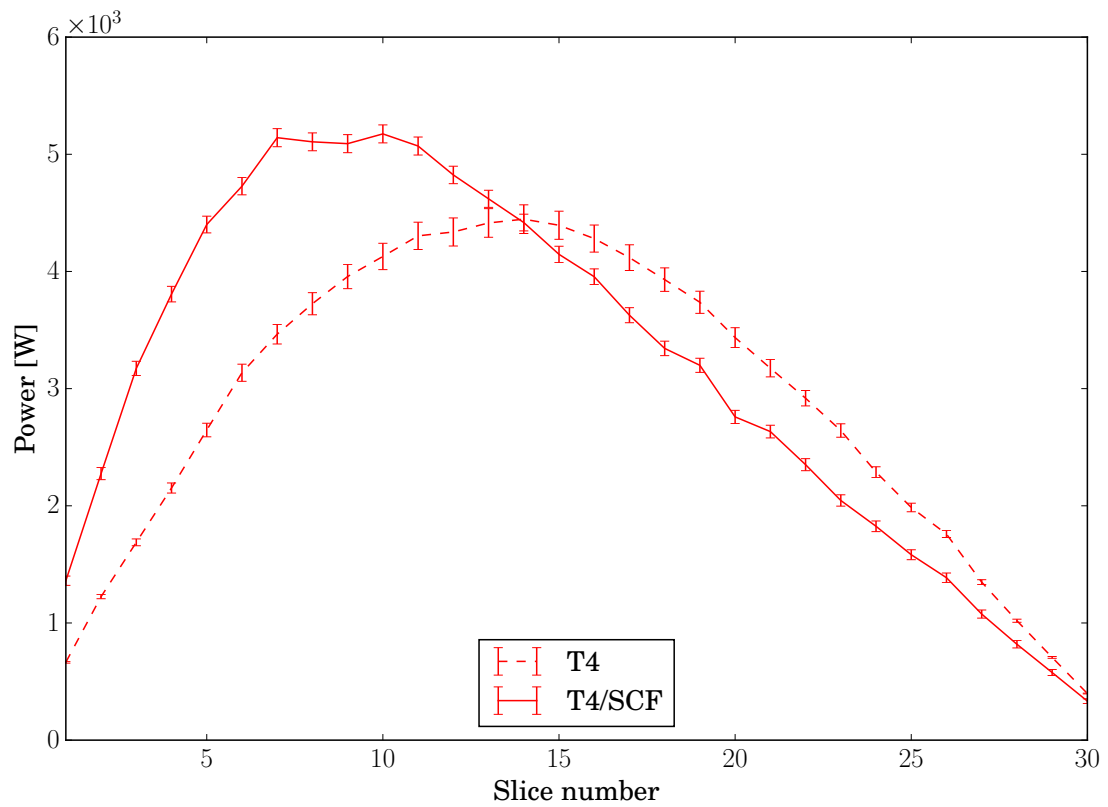


Figure 6.4 – Axial power profile (in watts) for the rod 1 obtained without (dashed line) and with feedback (solid line) for the TMI-1 3x3 mini-core.

6.2 Steady state

As a first verification test of the dynamic capabilities of TRIPOLI-4, we have performed a steady-state calculation. Starting from the stored source (fission sources and thermal-hydraulics fields) presented in Sec. 6.1, we have followed the time evolution of the power during 5 s with 50 regularly spaced intervals by increments of $\Delta t = 0.1$ s. At the end of each time interval, the neutron power is averaged over all simulators and transferred to SUBCHANFLOW, which then runs and solves the thermal-hydraulics equations for the time step. The next time step begins with the updated temperatures and densities. The purpose of the simulation is to present a first example of a dynamic Monte Carlo simulation of a realistic system at steady state observed over a few seconds, with thermal-hydraulics feedback. We have used 1000 parallel units during 24 h, each simulator has performed six complete batches. Therefore, the power was averaged over 6000 batches, and the computational cost of the simulation was 24000 h.

In order to reduce correlations between batches, we have chosen to perform ten additional power iterations at the beginning of each batch, starting from the fission sources and thermal-hydraulics fields of the previous batch. SUBCHANFLOW is called with the new power distribution and the compositions are updated. This way, for each batch, the dynamic simulation begins with different fission sources, temperature and density fields: the batches being weakly correlated, they should provide a reliable estimation of the standard deviation. This process was used for all the different dynamic calculations presented below (Secs. 6.2 to 6.3.2).

As expected, the total fission power integrated over the mini-core is roughly constant with time, as shown in Fig. 6.5. The total power fluctuates around a value close to 140.94 MW, which is the normalization value used of for the stationary calculation (Sec. 6.1). We can conclude that the dynamic normalization factor defined in Eq. (4.3.1) correctly rescales the power scored in the dynamic phase to the power level of the criticality calculation with feedback; if this were not the case, the thermal-hydraulics fields would be out of equilibrium with the power field and a drift would appear.

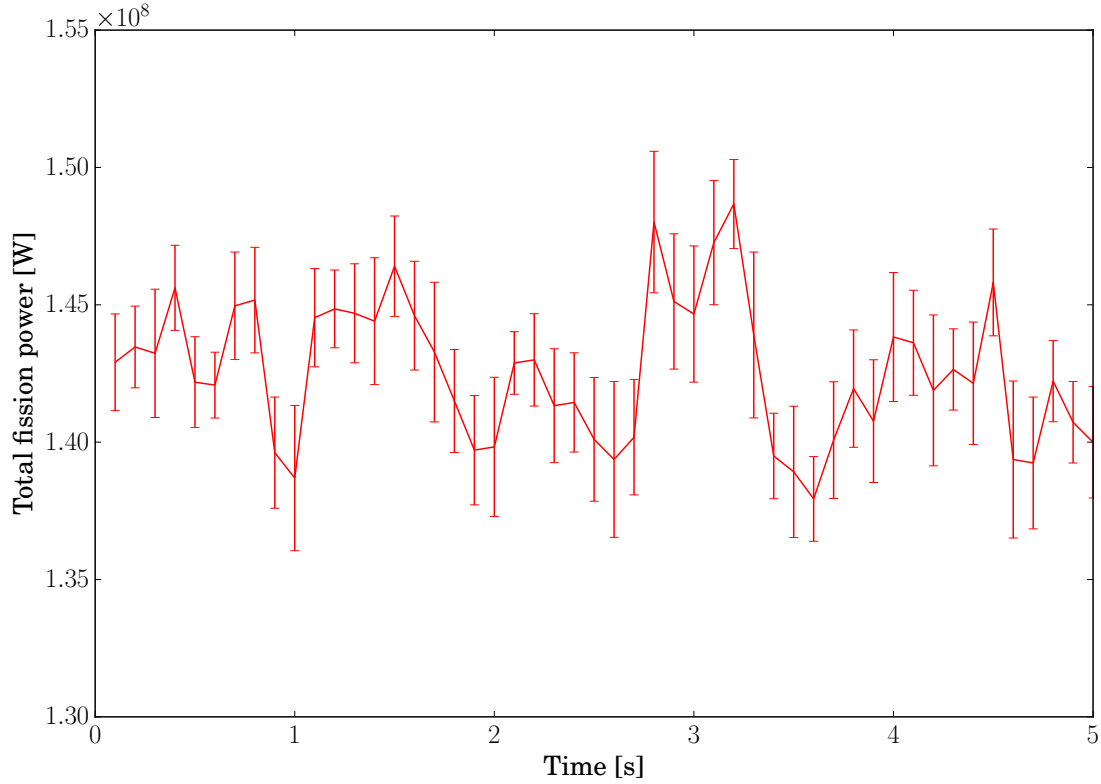


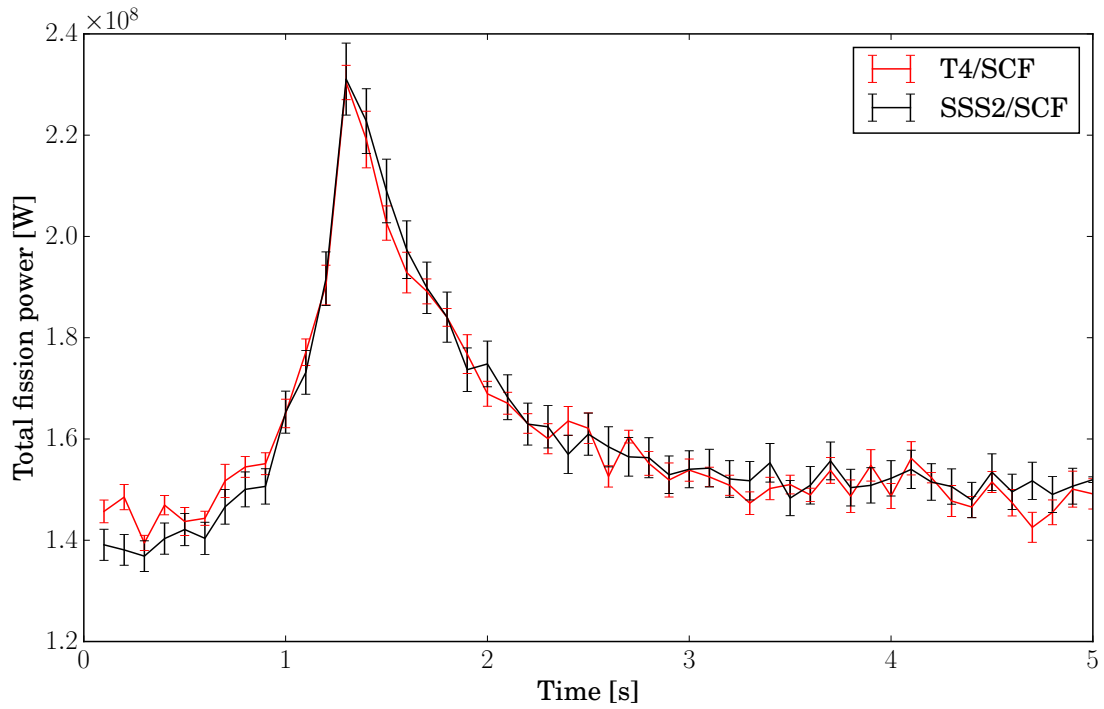
Figure 6.5 – Time evolution of the total power (in watts) in the TMI-1 3x3 mini-core computed with T4/SCF in steady-state conditions. The total power is stable and fluctuates around 140.94 MW.

6.3 Transients

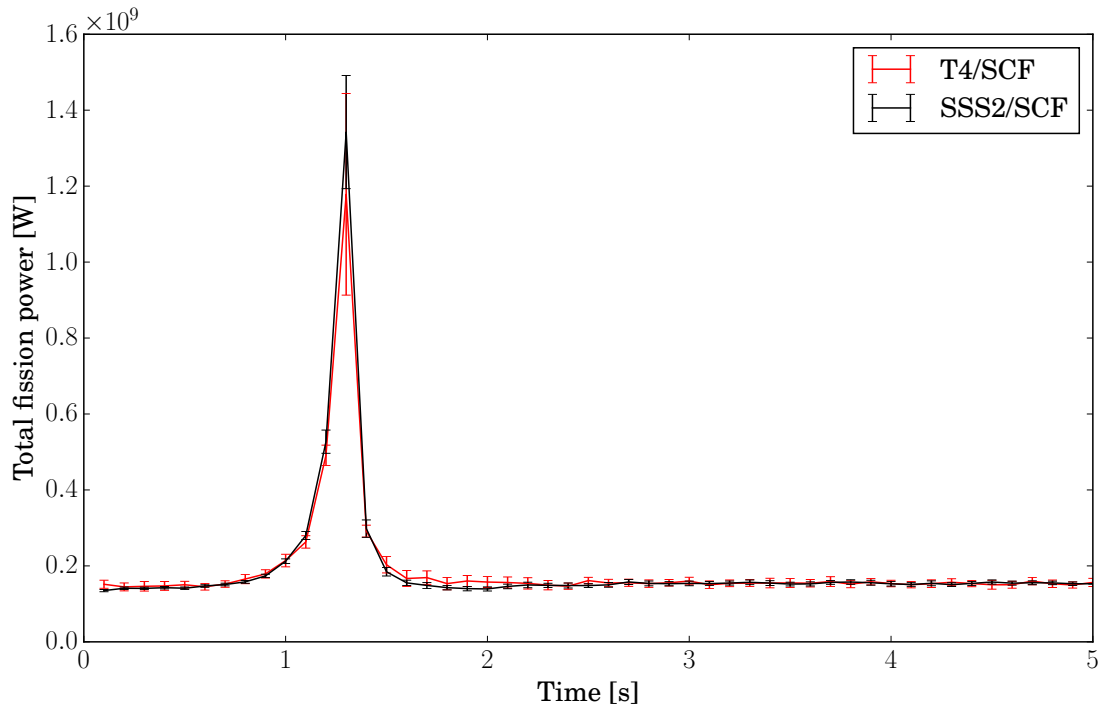
Now that we have verified the capabilities of the T4/SCF coupling scheme to compute the dynamic evolution of a system in steady-state conditions, we turn our attention to some of the scenarios that were presented in Sec. 3.2.3 and solved without thermal-hydraulics feedback. We focus in particular on scenarios C and D. The purpose of the following simulations is to analyse the impact of thermal-hydraulics feedback on power excursions. The different calculations have been run on the CEA Cobalt supercomputer using 1000 parallel units for 24 hours. We also present the comparison between our results and the SSS2/SCF results published in Ferraro et al. (2019b).

	40 cm rod extraction		30 cm rod extraction	
	T4/SCF	SSS2/SCF	T4/SCF	SSS2/SCF
k_{eff}	$1.00966 \pm 11 \times 10^{-5}$	$1.00967 \pm 7 \times 10^{-5}$	$1.00382 \pm 11 \times 10^{-5}$	$1.00355 \pm 7 \times 10^{-5}$
ρ [\$]	~ 1.3	~ 1.3	~ 0.5	~ 0.5

Table 6.1 – Multiplication factor and reactivity for the TMI-1 3x3 mini-core with the control rods extracted by 40 cm and 30 cm, obtained with T4/SCF and SSS2/SCF.

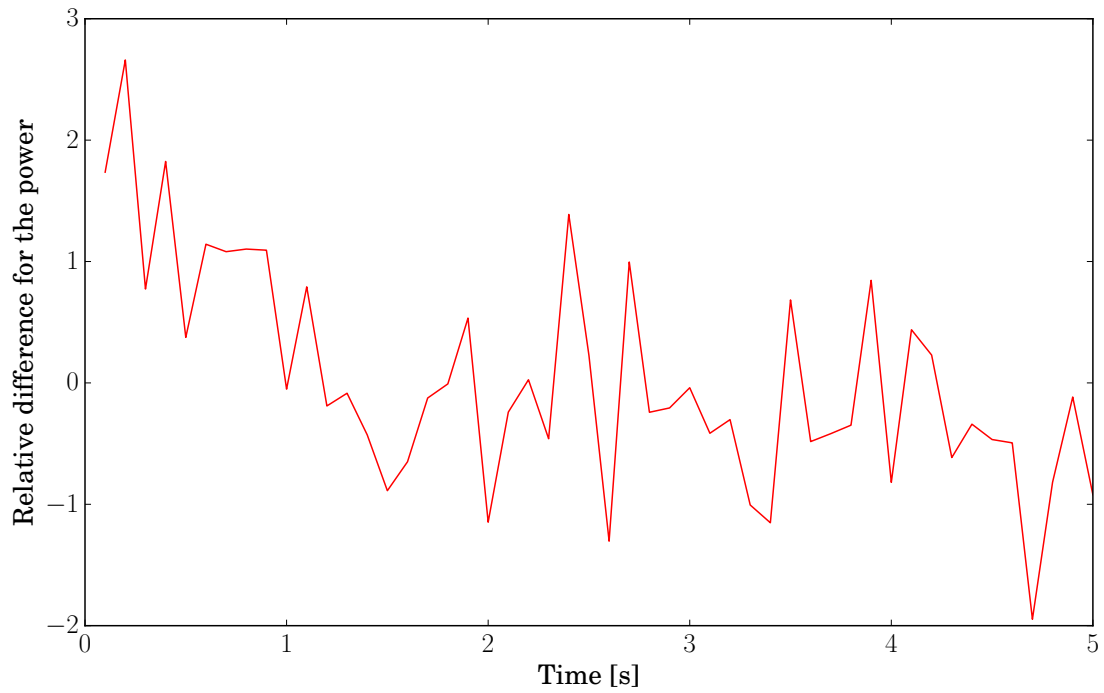


(a) 30 cm rod extraction

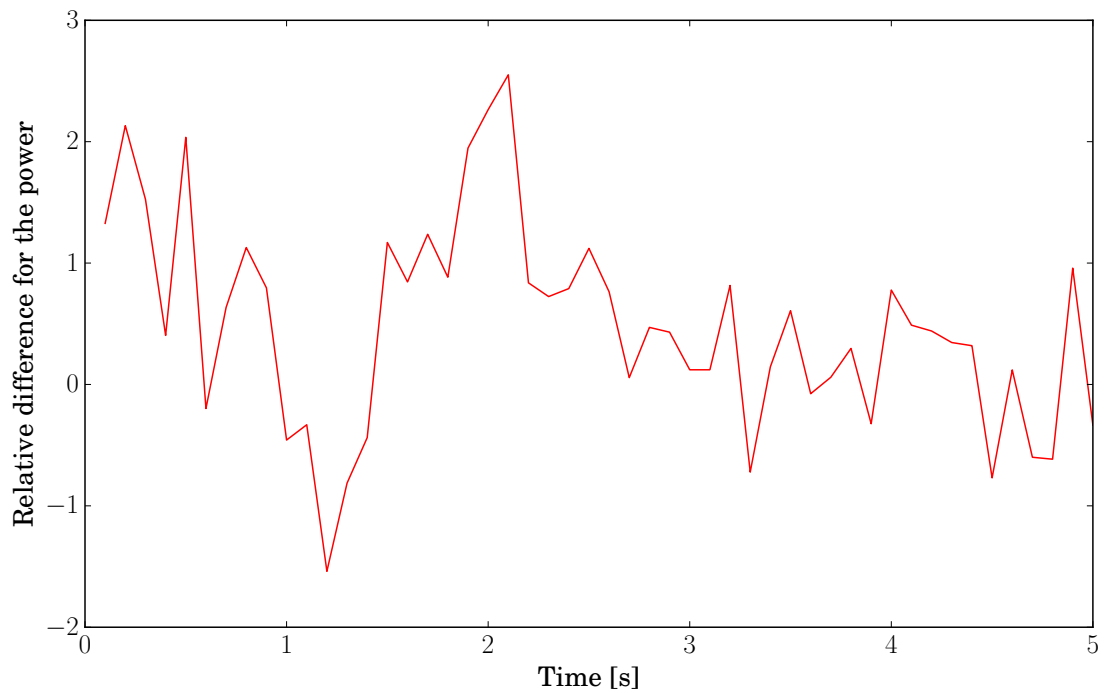


(b) 40 cm rod extraction

Figure 6.6 – Time evolution of the total power (in watts) in the TMI-1 3x3 mini-core computed with T4/SCF (red line) and SSS2/SCF (black line) with the control rods progressively extracted between $t = 0.3$ s and $t = 1.3$ s. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.

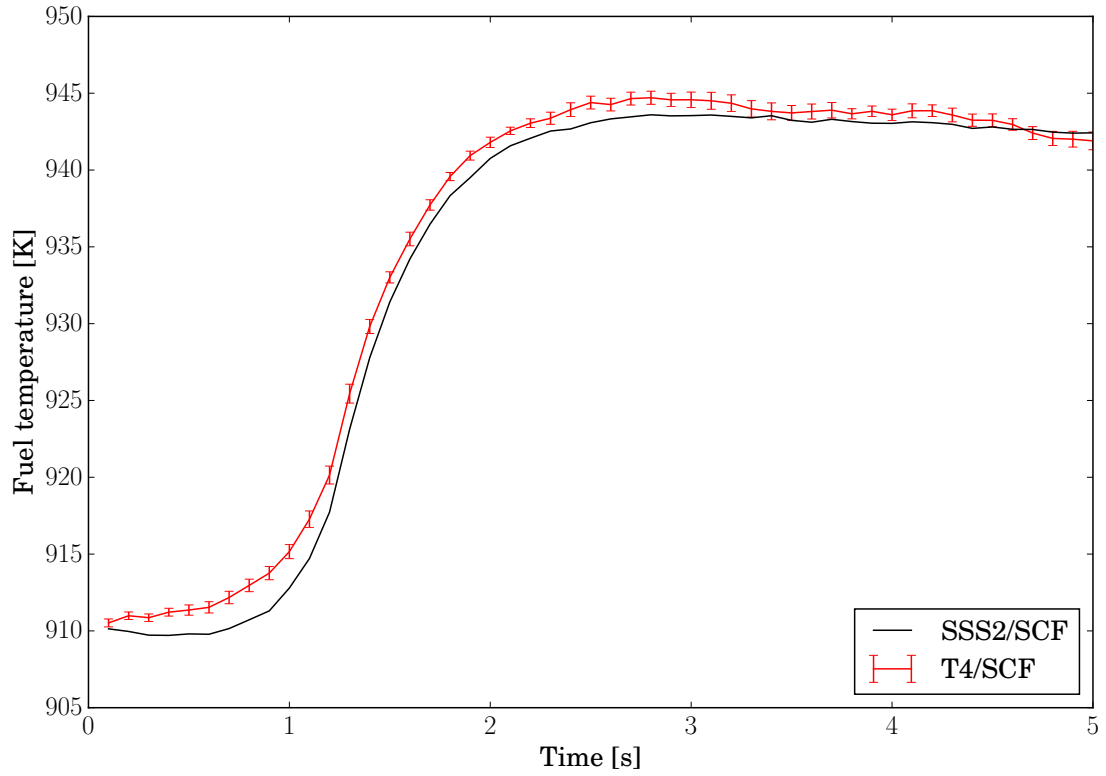


(a) 30 cm rod extraction

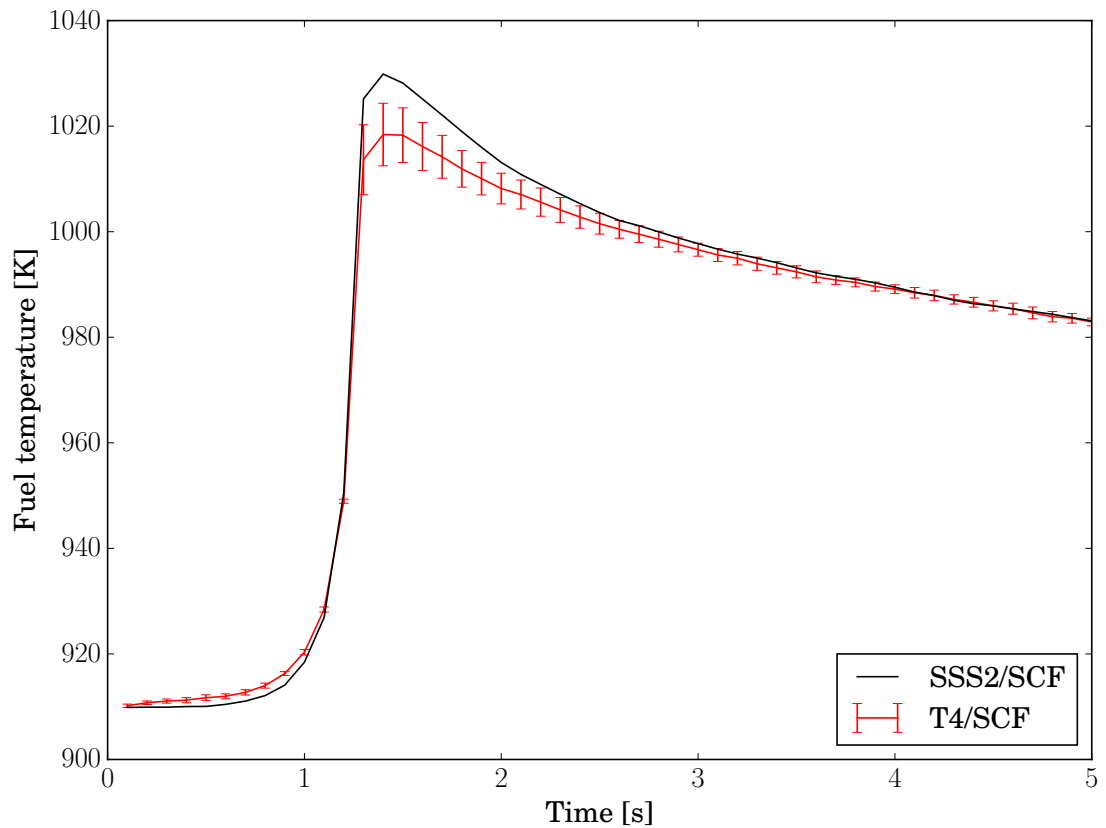


(b) 40 cm rod extraction

Figure 6.7 – Difference between total power obtained with T4/SCF and SSS2/SCF divided by the estimated standard error. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.



(a) 30 cm rod extraction



(b) 40 cm rod extraction

Figure 6.8 – Time evolution of the average fuel rod temperature (in kelvins) in the TMI-1 3x3 mini-core computed with T4/SCF (red line) and SSS2/SCF (black line) with the control rods progressively extracted between $t = 0.3$ s and $t = 1.3$ s. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.

6.3.1 Control-rod extraction

As a reminder of scenario D, the control rods are progressively extracted by 40 cm between $t = 0.3$ s and $t = 1.3$ s. It should be noted that the critical configuration is very different from the configuration of Sec. 3.2; this is due to the temperature and density fields, which are now determined by thermal-hydraulics feedback, but also the difference in boron concentration (1305 ppm vs. 1493 ppm). We have computed the induced reactivity with a criticality calculation with thermal-hydraulics feedback. For this new configuration, extracting the rods by 40 cm makes the system prompt supercritical with $\Delta\rho \approx 1.3$ \$: the multiplication factor is $k_{\text{eff}} = 1.00966 \pm 11 \times 10^{-5}$. We have also performed another simulation with a less severe extraction, extracting the rods by 30 cm. The multiplication factor for this configuration is $k_{\text{eff}} = 1.00382 \pm 11 \times 10^{-5}$, which corresponds to a reactivity insertion of about $\Delta\rho \approx 0.5$ \$. The multiplication factors are summarized in Table 6.1, which also presents the multiplication factors coming from SSS2/SCF calculations. The two coupling schemes are in good agreement.

As in Sec. 3.2, the rod extraction is discretized in 10 steps. The memory allocation for the initialization of the dynamic calculation, using 11 geometries for TRIPOLI-4 part, is about 7 GB per processor, which exceeds the memory per core limitation of the cluster. Thus, we had to use two cores per parallel unit. Each simulator completed six full batches in 24 h for the 30 cm scenario and three full batches for the 40 cm scenario. The time evolution of the total power for both scenarios is shown in Fig. 6.6, and the difference between T4/SCF and SSS2/SCF divided by the estimated standard error is plotted in Fig. 6.7.

We analyse the 30 cm scenario first. The impact of the thermal-hydraulics feedback is clearly visible. First, the power is stable around 140.94 MW. Then, the power increases up to about 220 MW, before decreasing because of the thermal-hydraulics feedback, and reaches a new equilibrium with an asymptotic value of about 150 MW, which is higher than initial power. The results obtained with SSS2/SCF are also presented (Ferraro et al., 2019b). There is a very good agreement between T4/SCF and SSS2/SCF.

For the 40 cm extraction, the power is also initially centered around 140.94 MW, as expected. Then, the power increases up to 1100 MW: the power increases by a factor 8. During the last time step of the rod extraction (between $t = 1.2$ s and $t = 1.3$ s), the error bars for both calculations are very large due to large fluctuations on the population size. Then, the power decreases because of the thermal-hydraulics feedback, and reaches an asymptotic value that is similar to the one that is reached in the 30 cm extraction. Here, however, the power rise and drop are much sharper. Again, T4/SCF and SSS2/SCF are in very good agreement. The time evolution of power presented here can be compared to the one in Fig. 3.9 for the same scenario, but without thermal-hydraulics feedback. It is interesting to notice that the agreement between T4/SCF and SSS2/SCF is actually better than the agreement between T4 and SSS2 for the kinetic calculations presented in Sec. 3.2.3, where there was a deviation between the Monte Carlo codes during the flux excursion (see Fig. 3.9). The thermal-hydraulics feedback absorbs most of the reactivity difference between T4 and SSS2 and drives the system towards the same equilibrium configuration around 150 MW.

The time evolution of the fuel temperature averaged over all the rods of the mini-core is presented in Fig. 6.8. Error bars from SSS2/SCF were not available, which limits our comparison analysis. As the reactivity increases, the average fuel temperature increases, then it remains stable after about 2 s. The equilibrium state is defined by higher temperatures for the 40 cm scenario than for the 30 cm. At the same time, the coolant temperature increases and the coolant

density decreases as shown by Figs. 6.9 and 6.10.

The axial profile in the four rods identified in Fig. 6.1 is also provided in Fig. 6.11 for the time step between $t = 1.1$ s and $t = 1.2$ s (during the rod extraction). For the coolant, the temperature and density axial profiles in the four cells identified in Fig. 6.1 are provided in Figs. 6.12 and 6.13 for the same time step. Note that for technical reasons, the coolant temperature and density values for T4/SCF are actually taken from one of the pin cells around the channels, as shown in Fig. 6.1, while SSS2/SCF values are taken from the channels. This may explain the small differences between T4/SCF and SSS2/SCF, which is anyway of the order of 1 K. For this reason, we will focus on the fuel properties to avoid any ambiguity. Overall, there is a good agreement for all the observables presented.

The increase in the fuel temperature with time for the 40 cm scenario is shown in Sec. A.6.1 for three slices (Fig. A.2). The slice 3 is located at the bottom of the mini-core, where the reactivity is actually inserted through the rod extraction; the slices 10 and 15 are located above the rod extraction. For this reason, the increase is much larger for the slice 3: the temperature increases up to 900 K at the center of the mini-core. For the slices 10 and 15, the temperature mainly increases in the surrounding assemblies, since at these heights, the center of the mini-core still contains control rods. For an extensive comparison between T4/SCF and SSS2/SCF results for the 40 cm scenario, color maps of the difference between the two coupling scheme are provided in Sec. A.6.2 for three time steps (Fig. A.3). In some rods, the difference can reach large values (up to 60 K); this might be due to statistical fluctuations in the neutron population. Overall, the difference is small.

Discussion on the efficiency of the simulation

In order to evaluate the efficiency of the parallelism scheme, we study the distribution of the calculation time, defined as the time interval between the beginning of the first history and the end of the last history in a given time step, for a given simulator. Our sample consists of the calculation times of all the simulators, collected over all the complete batches. Figure 6.14 presents histograms of the simulation time for the 30 cm scenario (red bars) and the 40 cm scenario (green bars), for two different time steps. The first time step ranges from $t = 0.2$ s to $t = 0.3$ s, where the two scenarios are identical. As expected, the computation times are similar for the two scenarios. The fastest simulators have completed the calculation in about 100 s and had to wait for the slowest ones, up to 380 s (for the 40 cm scenario) or 250 s (for the 30 cm scenario). The difference between the fastest and the slowest simulators may seem large: it implies that 50% of the simulators were inactive for about 170 s (120 s for the 30 cm scenario). The situation, however, is much more serious for the last time step of the rod extraction (between $t = 1.2$ s and $t = 1.3$ s), especially for the 40 cm extraction scenario. Actually, there are large variations on the population size and therefore also large fluctuations among the simulators. Consequently, the calculation was slowed by one simulator who suffered from severe positive fluctuations during the rod extraction. The median time step is 130 s, the fastest simulators completed the calculation in 110 s and had to wait for the slowest simulator, which completed the calculation in 6700 s. Half of the simulators were thus inactive for almost two hours because of one single simulator. The large fluctuations in calculation time probably reflect fluctuations in neutron population, which grow with the size of the neutron population itself and are amplified by the branching nature of the fission process. In order to reduce the total waiting time and increase the simulation efficiency, we have tried to enforce population control on a tight grid during this time step, but no improvement was observed. It may be worth it to use branchless collisions, as shown by the test case performed on a supercritical configuration in Sec. 2.6.2.

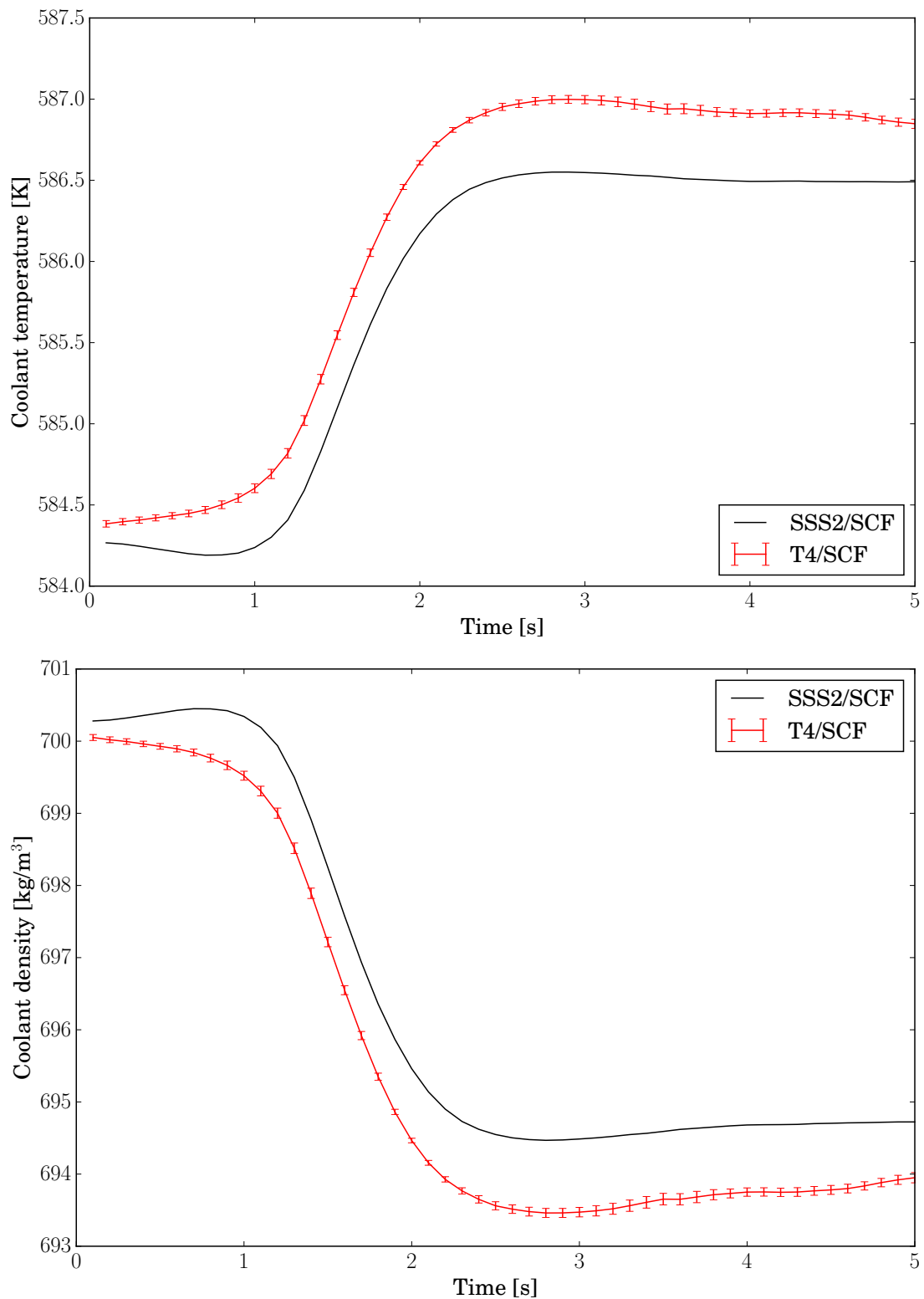


Figure 6.9 – Time evolution of the coolant fields in the TMI-1 3x3 mini-core computed with T4/SCF (red line) and SSS2/SCF (black line) with the control rods progressively extracted by 30 cm between $t = 0.3$ s and $t = 1.3$ s. Top: average coolant temperature (in kelvins). Bottom: average coolant density (in kg/m^3).

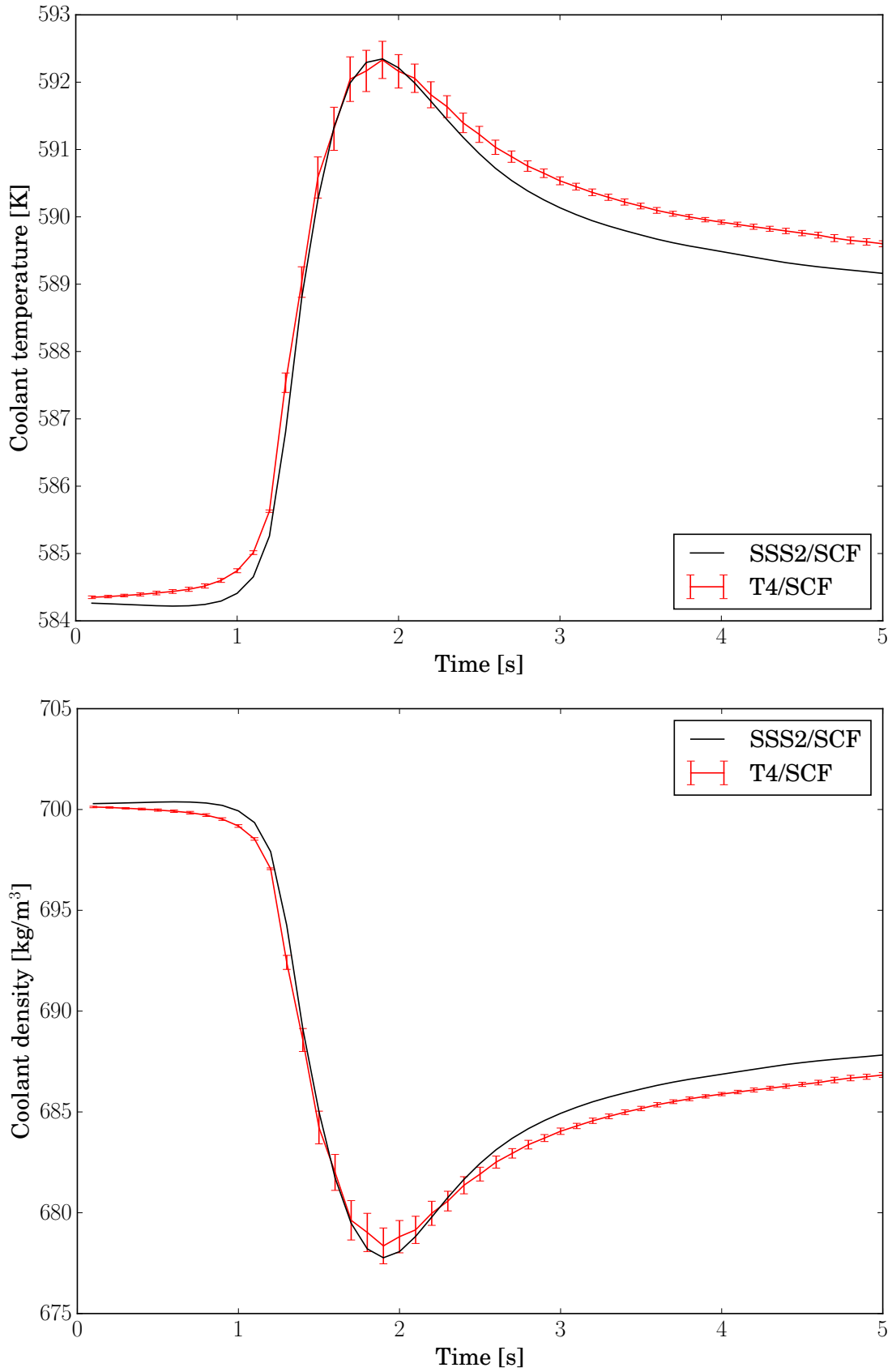
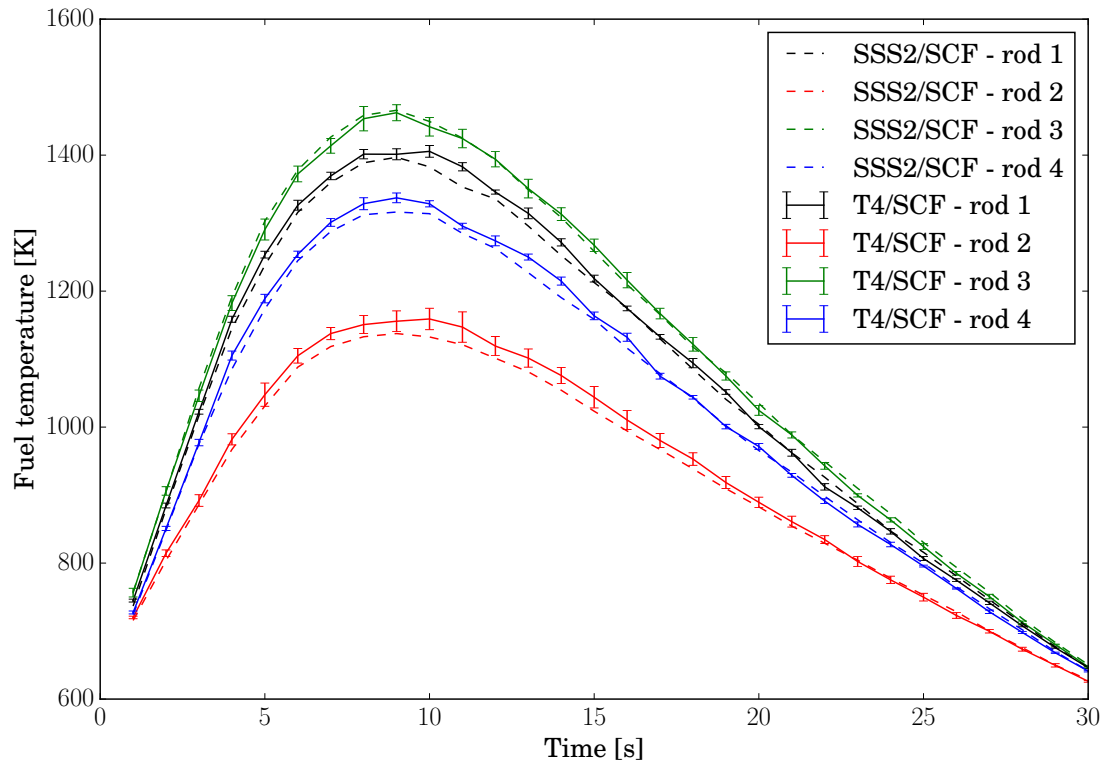
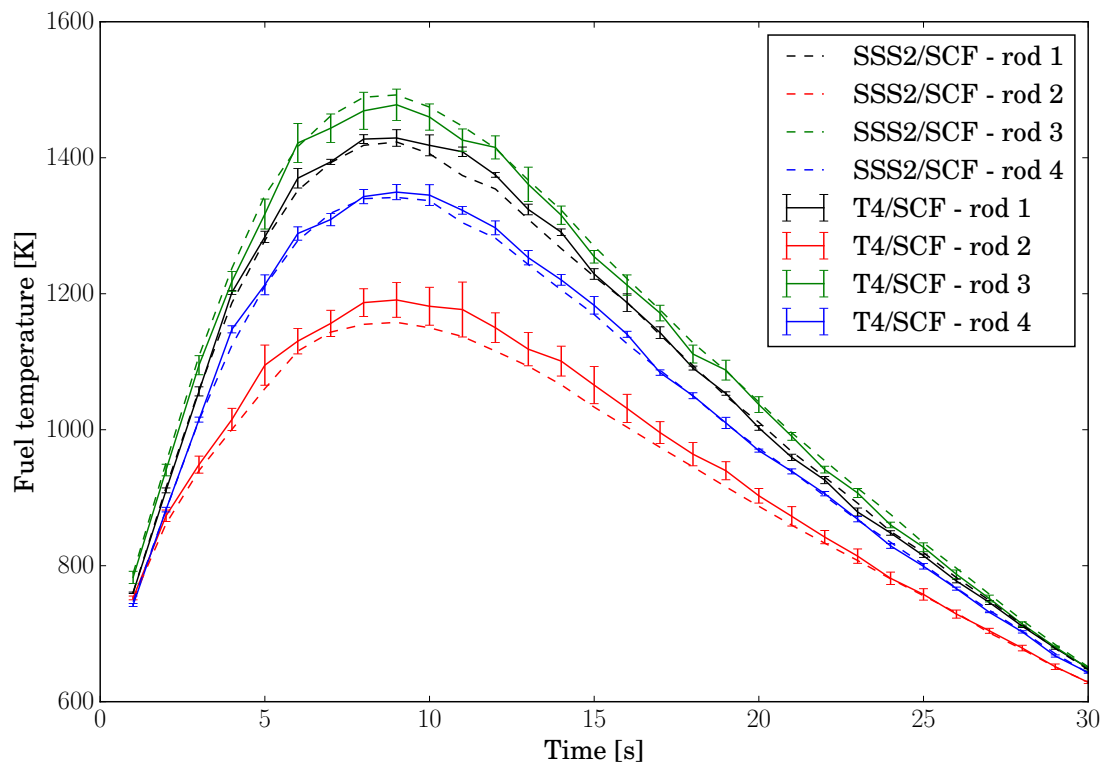


Figure 6.10 – Time evolution of the coolant fields in the TMI-1 3x3 mini-core computed with T4/SCF (red line) and SSS2/SCF (black line) with the control rods progressively extracted by 40 cm between $t = 0.3$ s and $t = 1.3$ s. Top: average coolant temperature (in kelvins). Bottom: average coolant density (in kg/m^3).



(a) 30 cm rod extraction



(b) 40 cm rod extraction

Figure 6.11 – Axial fuel temperature profiles (in kelvins) for rods 1 to 4 in the TMI-1 3x3 mini-core computed with T4/SCF (solid line) and SSS2/SCF (dashed line) between $t = 1.1$ s and $t = 1.2$ s. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.

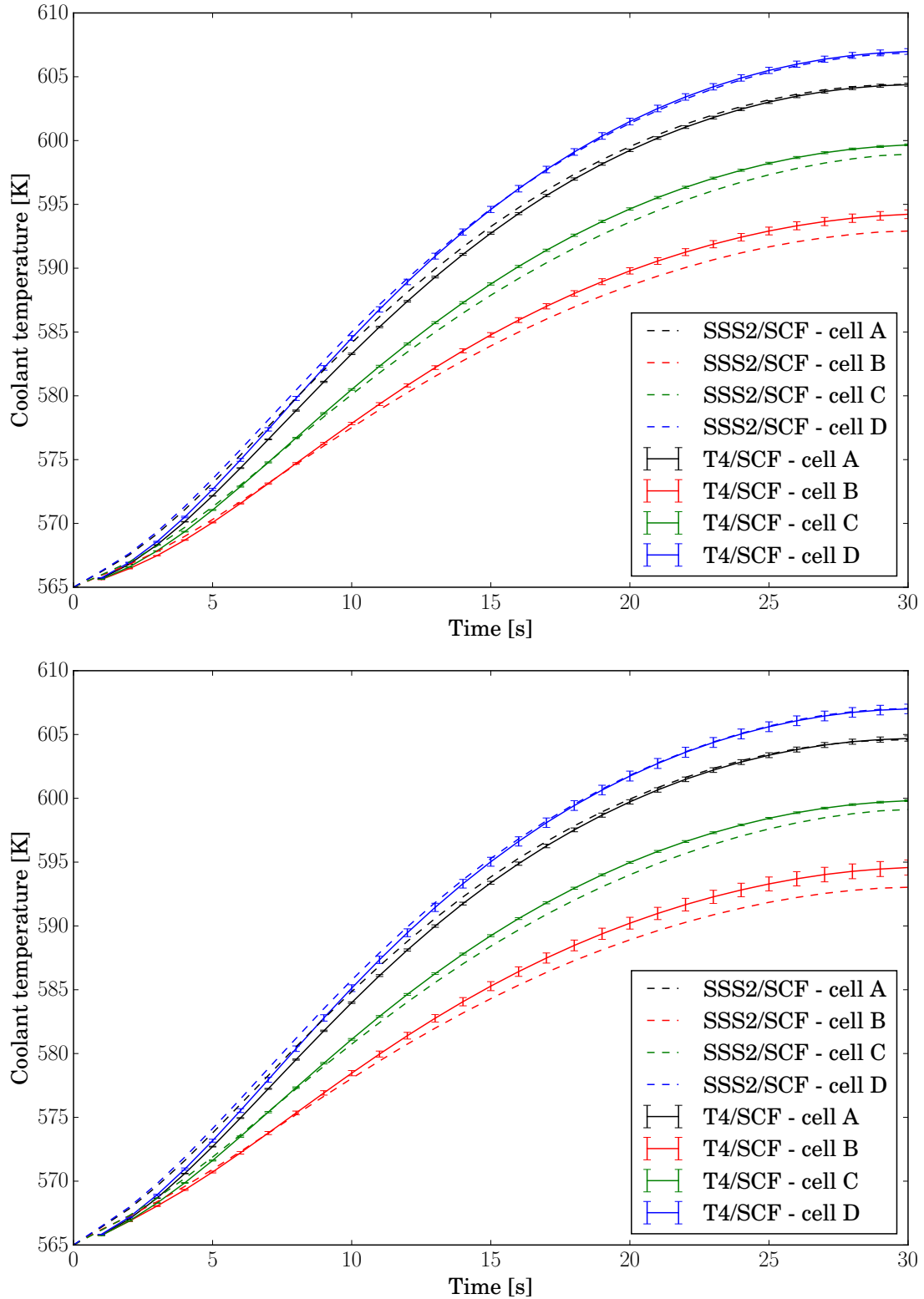


Figure 6.12 – Axial coolant temperature profiles (in kelvins) for cells A to D in the TMI-1 3x3 mini-core computed with T4/SCF (solid line) and SSS2/SCF (dashed line) between $t = 1.1$ s and $t = 1.2$ s. Note that the coolant temperature and density values for SSS2/SCF values are taken from one of the neighbouring channels. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.

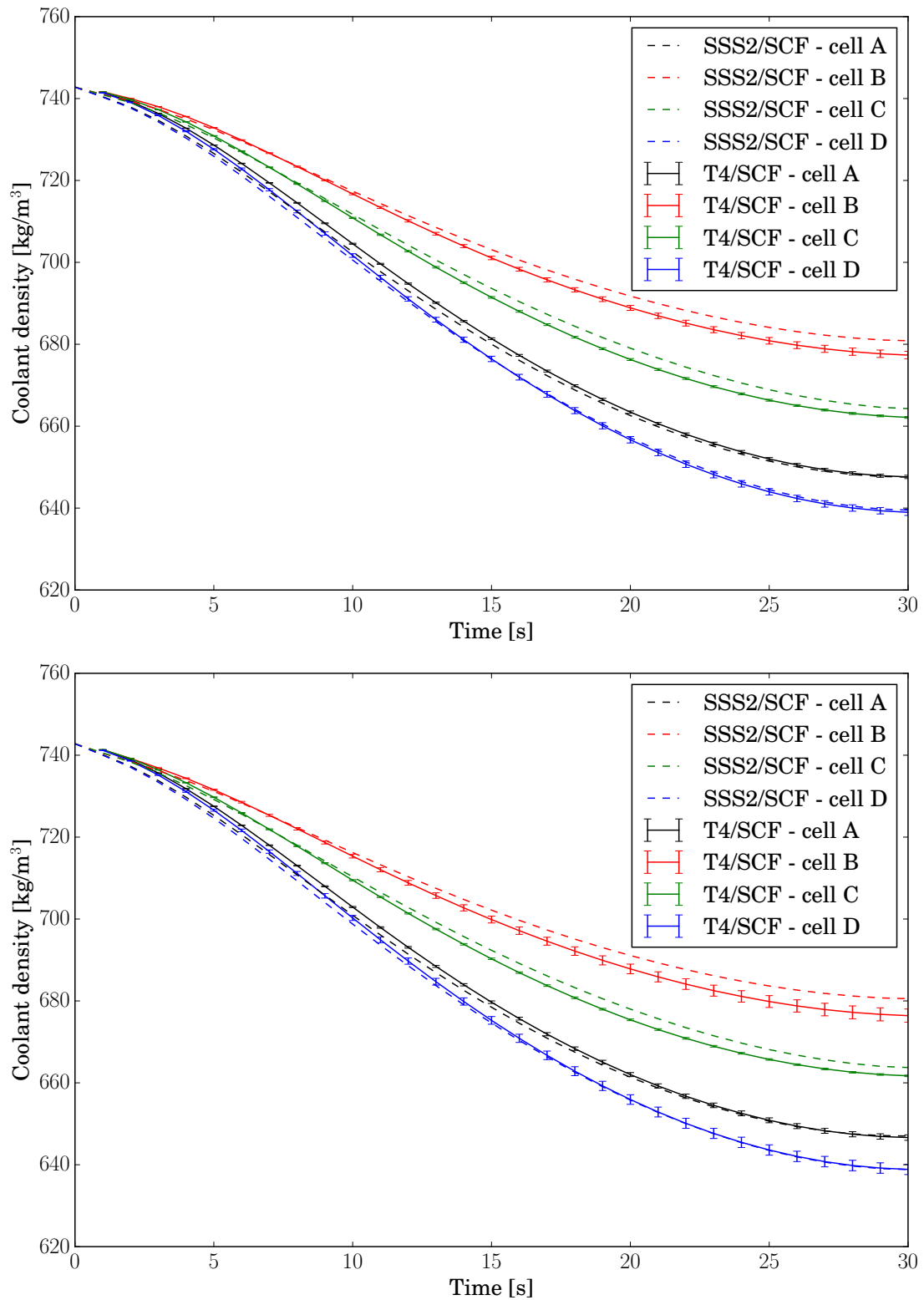


Figure 6.13 – Axial coolant density profiles (in kg/m^3) for cells A to D in the TMI-1 3x3 mini-core computed with T4/SCF (solid line) and SSS2/SCF (dashed line) between $t = 1.1$ s and $t = 1.2$ s. Note that the coolant temperature and density values for SSS2/SCF values are taken from one of the neighbouring channels. Top: 30 cm rod extraction. Bottom: 40 cm rod extraction.

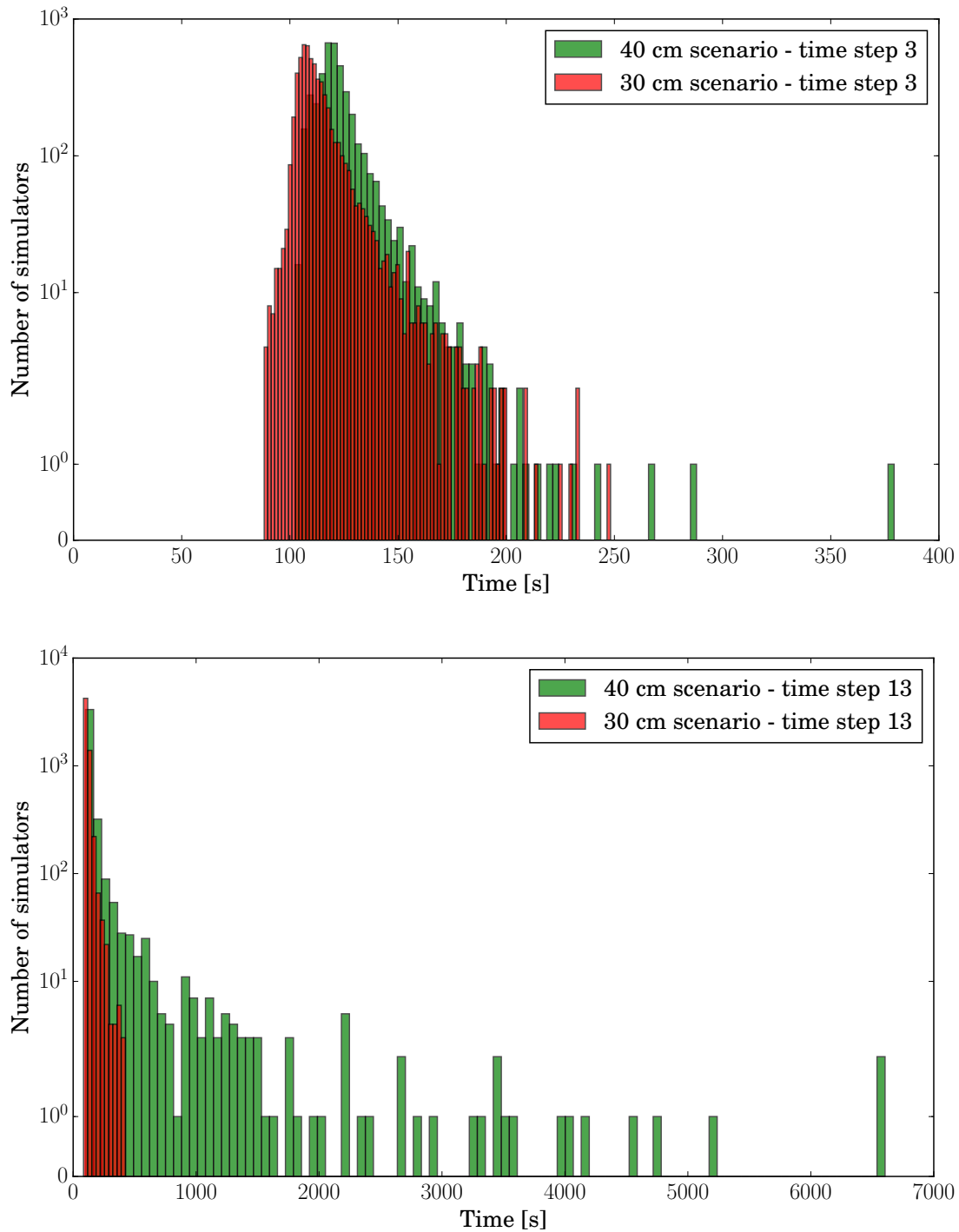


Figure 6.14 – Histograms of the simulation computation times for the 30 cm (red bars) and 40 cm (green bars) extraction scenarios. Top: third time step (between $t = 0.2$ s and $t = 0.3$ s). The simulation times are similar between the two scenarios, and relatively homogeneous among the simulators. Bottom: last time step of the rod extraction (between $t = 1.2$ s and $t = 1.3$ s). The 40 cm scenario is hampered by large fluctuations on the population size: half of the simulators completed the simulation in 130 s and had to wait until 6700 s for the slowest one. The discrepancy is much smaller and does not represent a severe issue for the 30 cm scenario.

Our coupling scheme exacerbates the negative impact of the fluctuations, because the supervisor needs to wait for *all* the simulators to complete their histories before calling the thermal-hydraulics solver. One way to improve the efficiency of the simulation would be to split the calculation into smaller, truly independent replicas, with independent couplings to thermal-hydraulics. When all simulators of the same packet have completed the time step, SUBCHANNELFLOW could be run without waiting for the other packets, and at least these simulators could start the next time step. Therefore, we would still be waiting for some packets, but fewer simulators would stay inactive during this time.

6.3.2 Control-rod extraction and reinsertion

In addition to the comparisons with SSS2/SCF for scenario D, we have also simulated scenario C with the T4/SCF coupling scheme. In this scenario, control rods are extracted between $t = 0.3$ s and $t = 1.3$ s, and later reinserted to the initial height between $t = 3.1$ s and $t = 4$ s. The total power as a function of time is shown in Fig. 6.15 and can be compared to the one presented in Fig. 3.9, for the same scenario without thermal-hydraulics feedback. The power is stable at first, then it increases by a factor 8 before starting to decrease, similarly to scenario D. Then, at $t = 3.1$ s, when the control rods are reinserted, the power decreases to a new equilibrium value around 90 MW. Each simulator completed four full batches in 24 h.

The time evolution of the fuel temperature is presented in Fig. 6.16. The time evolution of the coolant temperature and coolant density is presented in Fig. 6.17. It appears that the thermal-hydraulics fields have not reached yet the asymptotic state at the end of the 5 s.

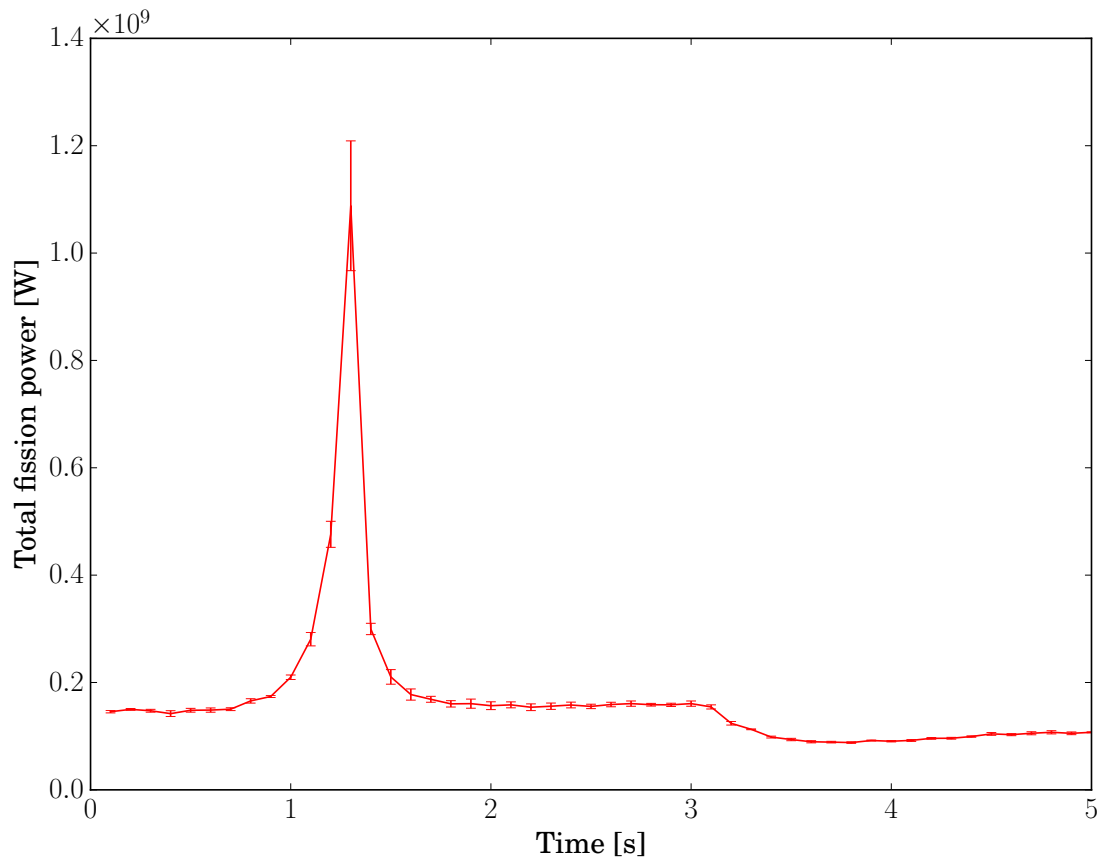


Figure 6.15 – Time evolution of the total power (in watts) in the TMI-1 3x3 mini-core computed with T4/SCF during scenario C transient, where control rods are progressively extracted by 40 cm between $t = 0.3$ s and $t = 1.3$ s and reinserted back to their initial position between $t = 3.1$ s and $t = 4.1$ s.

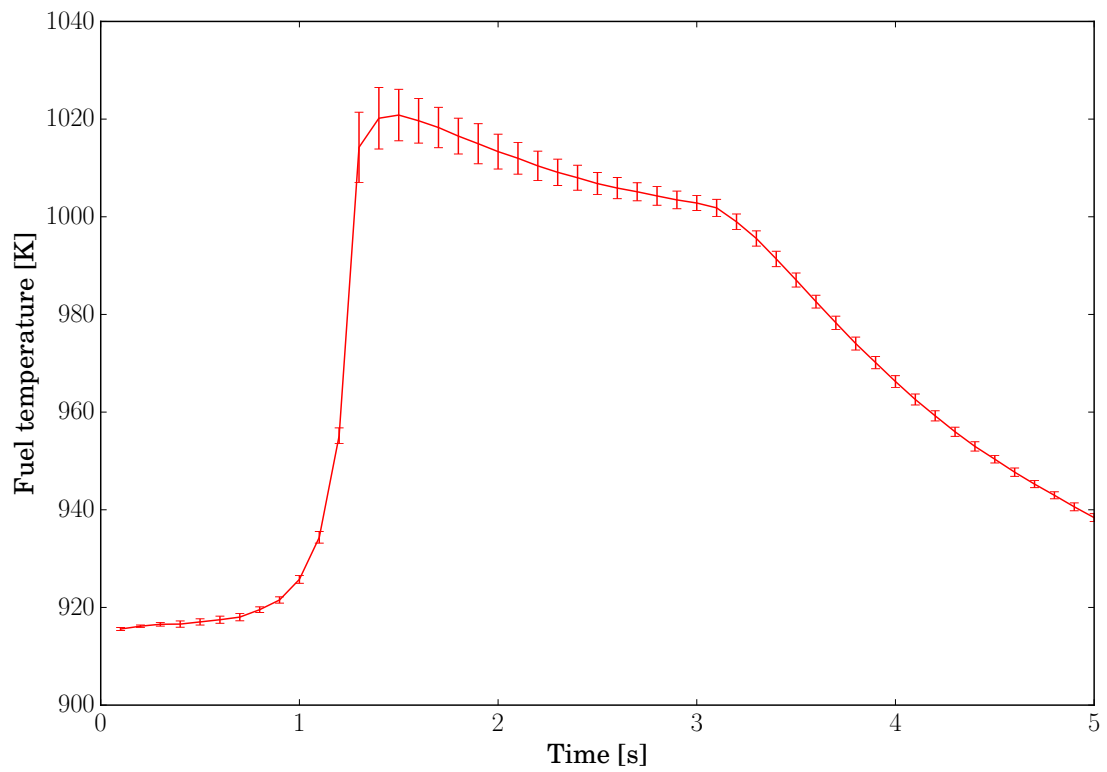


Figure 6.16 – Time evolution of the average fuel rod temperature (in kelvins) in the TMI-1 3x3 mini-core computed with T4/SCF during transient scenario C.

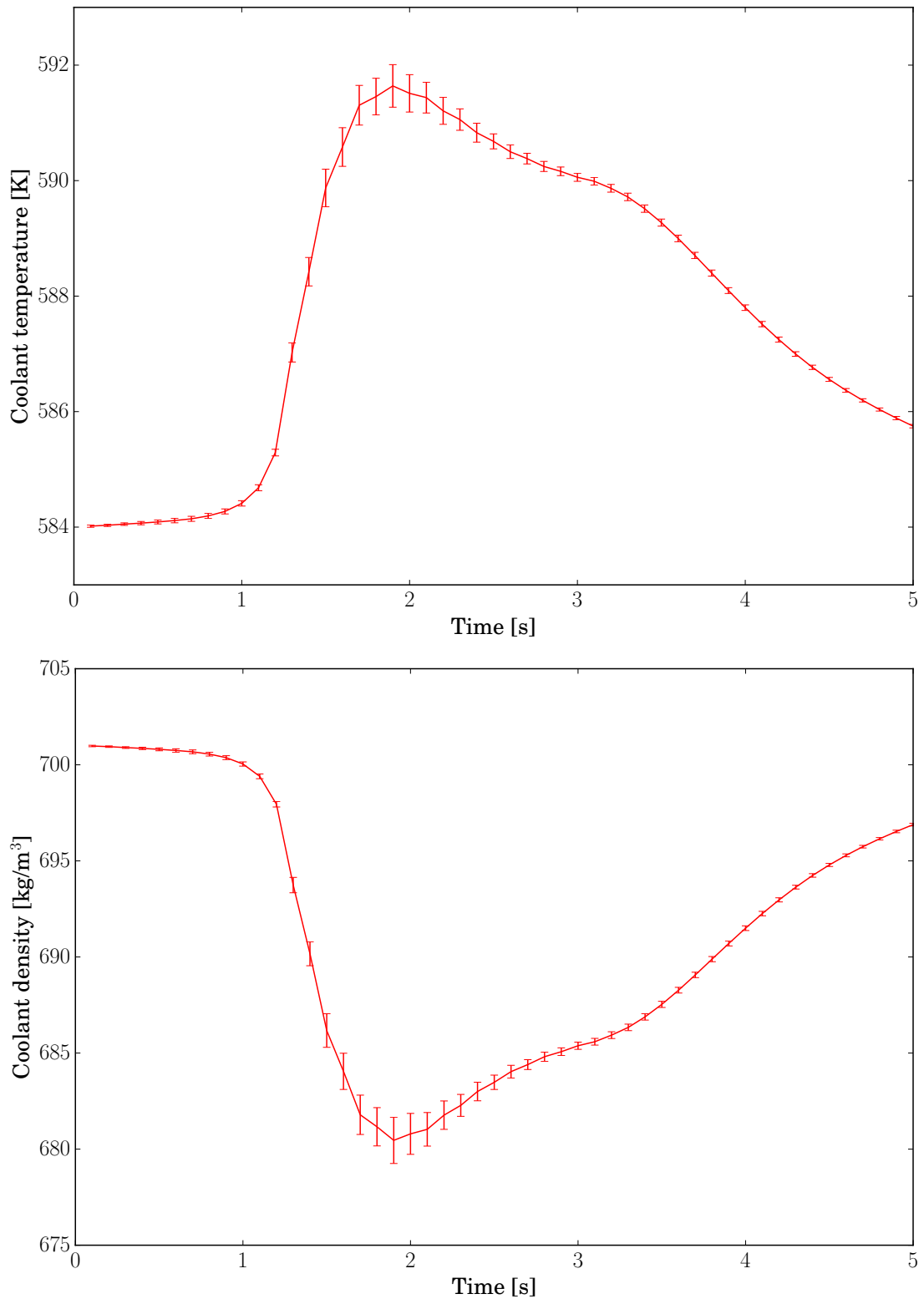


Figure 6.17 – Time evolution of the coolant fields in the TMI-1 3x3 mini-core computed with T4/SCF during scenario C. Top: average coolant temperature (in kelvins). Bottom: average coolant density (in kg/m³).

6.4 Conclusion

We have performed the first dynamic simulations of the TMI-1 3x3 mini-core with TRIPOLI-4, starting from an initial state that we calculated once with a criticality calculation with feedback. We reused it for the different dynamic calculations, which saved significant computational time. In order to reduce correlations between the batches, we performed additional power iterations for each batch on the fission sources of the previous batch, and ran SUBCHANFLOW to update the thermal-hydraulics fields.

We have first verified the capability for dynamic simulations with a steady-state calculation. The power remains stable with time. We have then studied the same reactivity-insertion scenario (scenario D) as in Sec. 3.2.3, with the rods being extracted by 30 cm and 40 cm. The simulation of the peak power is very challenging because of the large variations on the population size, which cause large variations among the simulators. The 40 cm rod extraction especially induces a very large reactivity insertion (the system becomes prompt supercritical), and fluctuations on the population size lead to fluctuations in CPU time among the simulators. In particular, one simulator was about 50 times slower than the average. Because of our implementation of the coupling scheme, and more specifically because of the thermal-hydraulics rendez-vous points, the simulation was hampered by few slow simulators that sampled rare events. Actually, the 40 cm rod extraction is much more challenging than the 30 cm extraction. Indeed, for the 30 cm scenario, the discrepancy among simulators was not such an issue because the reactivity insertion was much lower (0.5 \$ against 1.3 \$). For the two scenarios, the feedback mechanisms make the power decrease and reach a new equilibrium state. We generally found good agreement between the T4/SCF and SSS2/SCF results. We have also simulated scenario C, where the rods are later reinserted. Similarly, the power reaches a new equilibrium state. Within the 5 s of the simulation, the thermal-hydraulics fields do not reach a new equilibrium.

There are different perspectives for the improvement of our dynamic calculations. In the 40 cm rod extraction scenario, it seems necessary to reduce the waiting time due to the thermal-hydraulics rendez-vous points. In order to mitigate the impact of fluctuations in CPU time among simulators, the simulation could be split up in independent replicas. However, such solution is in tension with the need to minimize the statistical fluctuations resulting from the Monte Carlo simulation, as explained in Sec. 4.4.1. Substantial profits would be also obtained by improving the handling of the geometry by TRIPOLI-4. Indeed, the definition of a single geometry for transient calculations would greatly reduce the memory use, and each parallel unit would not end up having to use the memory of two cores. Finally, the convergence of the selected explicit Euler scheme deserves further investigation.

Part III

Preliminary analysis of the coupled system stability

Chapter 7

Description of a simple model representative of the coupled system

7.1 Description of the problem

The coupling between Monte Carlo neutron transport and thermal-hydraulics is intended as a reference for coupled deterministic calculations. For this purpose, it is essential to be able to assess the uncertainty stemming from the reference Monte Carlo coupled calculations. In order to study the convergence over the coupling iterations, two aspects must be examined. First, the output of a Monte Carlo code is characterized by statistical fluctuations. Second, thermal-hydraulics equations are non-linear. The combination of these two aspects raises the following questions: how do the statistical fluctuations coming from Monte Carlo transport propagate along the coupling iterations? Thermal-hydraulics feedback effects have a stabilizing effect in a PWR. Notwithstanding, is it possible that, despite the feedbacks, and because of amplification of the noise through the iterations, the system does not return to equilibrium?

The work presented in this chapter aims at providing a preliminary basis for further analysis of noise propagation in the neutron-transport-thermal-hydraulics coupled scheme. For this purpose, we propose a simplified model to conduct both numerical and analytical investigation.

7.2 The reference model

In order to model the coupling between a stochastic neutron transport code and thermal-hydraulics without having to deal with the complex models present in production codes such as TRIPOLI-4 and SUBCHANFLOW, we will consider a simplified model originally proposed by March-Leuba (1986). The model consists of five deterministic coupled equations, describing the multi-physics behaviour of a boiling water reactor (BWR), including neutron transport and thermal-hydraulics. In order to mimic the stochastic behaviour of Monte Carlo transport in the model, we add noise on the neutron concentration equation, and we study the propagation of such noise through the system and over the coupling iterations. In the following sections, we will describe the five-equation system. All the notations will be recalled in Sec. A.5.3.

7.2.1 Presentation of the deterministic coupled system

March-Leuba (1986) proposed a system of five coupled equations to model the coupling between neutron transport and thermal-hydraulics in a BWR. The model describes the evolution

of the following deterministic variables:

- N_t is the excess neutron concentration normalized to the steady-state neutron concentration,
- C_t is the excess precursors concentration normalized to the steady-state neutron concentration,
- T_t is the excess fuel temperature, expressed in kelvin,
- P_t^α is the excess void reactivity,
- P_t is the total excess reactivity over zero.

The model variables can be expressed as a function of the steady state quantities $N'_0, C'_0, T'_0, P_0^\alpha$ as follows

$$\begin{aligned} N_t &= \frac{N'_t - N'_0}{N'_0}, \\ C_t &= \frac{C'_t - C'_0}{N'_0}, \\ T_t &= T'_t - T'_0, \\ P_t^\alpha &= P_t^{\alpha'} - P_0^{\alpha'}, \end{aligned}$$

where $N'_t, C'_t, T'_t, P_t^{\alpha'}$ are the absolute quantities.

For our analysis, we make the assumption that we start from the equilibrium, meaning that

$$N_0 = C_0 = T_0 = P_0^\alpha = (dP_t^\alpha/dt)_0 = P_0 = 0. \quad (7.2.1)$$

The deterministic system reads then:

$$dN_t = \left(\frac{P_t - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t + \lambda C_t + \frac{P_t}{\Lambda_{\text{eff}}} \right) dt, \quad (7.2.2)$$

$$dC_t = \left(\frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t - \lambda C_t \right) dt, \quad (7.2.3)$$

$$dT_t = (a_1 N_t - a_2 T_t) dt, \quad (7.2.4)$$

$$dP_t^\alpha = \left(-a_3 P_t^\alpha - a_4 \int_0^t P_s^\alpha ds - k \int_0^t T_s ds \right) dt, \quad (7.2.5)$$

$$P_t = P_t^\alpha - D T_t, \quad (7.2.6)$$

with a_1, a_2, a_3, a_4, k being model parameters and D the Doppler coefficient. Their numerical values are given in Table 7.1. As a reminder of the parameters defined in Chapter 1, λ is the typical precursor decay constant, Λ_{eff} is the neutron effective mean generation time and β_{eff} is the effective delayed neutron fraction. The order of magnitude of the kinetics parameters is given in Table 7.2.

The first two equations of the system, namely Eqs. (7.2.2) and (7.2.3), describe the time-evolution of the neutron concentration N_t and the precursor concentration C_t , as given by the point-kinetics model (see Sec. 1.2.3) for excess and normalized quantities. They are related to the thermal-hydraulics Eqs. (7.2.4) and (7.2.5) via the reactivity P_t , defined by Eq. (7.2.6). More precisely, the Doppler effect on the fuel temperature T_t is modeled by Eq. (7.2.6), with a Doppler coefficient $D = 2.52 \text{ pcm K}^{-1}$. The void excess reactivity is defined by the second order differential Eq. (7.2.5), controlled by the fuel temperature. Finally, there is a non-linearity in Eq. (7.2.2) because of the term $P_t \times N_t$, where P_t depends on N_t . The difficulty in solving

reactor type	a_1 [$\text{K}^{-1} \text{s}^{-1}$]	a_2 [s^{-1}]	a_3 [s^{-1}]	a_4 [s^{-2}]	k [$\text{K}^{-1} \text{s}^{-2}$]	D [K^{-1}]
BWR	25.04	0.23	2.25	6.82	3.7×10^{-3}	2.52×10^{-5}
PWR	25.04	0.23	6	6.82	3.7×10^{-5}	2.52×10^{-5}

Table 7.1 – Values for the parameters of the coupled system, as given by March-Leuba (1986) for a BWR ($a_3^2 - 4 \times a_4 < 0$), and adapted for a PWR ($a_3^2 - 4 \times a_4 > 0$ for stability, and k is reduced to take into account the high pressure in the reactor).

analytically the system stems from this non-linearity.

It is interesting to notice that the system models the behaviour of a BWR, but it may actually also be used to describe a PWR, provided that we make slight modifications in the parameters a_3 and k . Indeed, the moderator always has a stabilizing effect on the reactivity in a PWR, while this is not always true in a BWR. To ensure the stability of the system, there must be no oscillations in the void excess reactivity Eq. (7.2.5), meaning that the discriminant must be positive, i.e., $a_3^2 - 4 \times a_4 > 0$. Moreover, the effect of the fuel temperature on the void fraction in a PWR should be smaller than in a BWR, because of the very high pressure in the reactor. For example, we can choose $a_3 = 6 \text{ s}^{-1}$ and $k = 3.7 \times 10^{-5} \text{ K}^{-1} \text{ s}^{-2}$ to model the feedback effects in a PWR. In the following we keep the modified values for a_3 and k , since our primary goal is the investigation of PWRs.

λ [s^{-1}]	Λ_{eff} [s]	β_{eff}
0.08	1.73×10^{-5}	763×10^{-5}

Table 7.2 – Typical values of the kinetics parameters for a PWR (values are given for the SPERT III E-core).

The solution of Eq. (7.2.3) gives an explicit representation of the precursor concentration

$$\begin{aligned}
 C_t &= C_0 e^{-\lambda t} + \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} e^{-\lambda t} \int_0^t N_s e^{\lambda s} ds \\
 &= \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} e^{-\lambda t} \int_0^t N_s e^{\lambda s} ds \quad \text{since } C_0 = 0.
 \end{aligned} \tag{7.2.7}$$

We get a similar expression for the fuel temperature using Eq. (7.2.4):

$$\begin{aligned}
 T_t &= T_0 e^{-a_2 t} + a_1 e^{-a_2 t} \int_0^t N_s e^{a_2 s} ds \\
 &= a_1 e^{-a_2 t} \int_0^t N_s e^{a_2 s} ds \quad \text{since } T_0 = 0.
 \end{aligned} \tag{7.2.8}$$

7.3 From a deterministic to a stochastic system

7.3.1 Adding Brownian noise

We begin by considering the two deterministic neutron transport Eqs. (7.2.2) and (7.2.3), for N_t and C_t . In order to “mimic” the stochastic behaviour of the Monte Carlo transport, we have

added a noise term to the equation for N_t . The noise should have the same characteristics as the statistical fluctuations coming from a Monte Carlo simulation, which can be reasonably approximated by a Gaussian noise (Lapeyre et al., 1997). For time-dependent Monte Carlo simulations, the Gaussian random variables are correlated between the time steps: Sec. 3.3 showed that correlations have an exponential decay. These considerations suggest that in our model we should add a Gaussian noise with correlations between the time steps having an exponential decay.

In order to simplify the problem, we have decided at first to add a Brownian noise term W_t to the equation for N_t , scaling the mean generation time Λ_{eff} . A Brownian motion is a continuous-time stochastic process $(W_t)_{t \geq 0}$ such that

- $W_0 = 0$,
- for $0 \leq s \leq t$, $W_t - W_s \sim \mathcal{N}(0, t - s)$.
- W_t has independent increments.

This is clearly not a faithful representation of Monte Carlo fluctuations, since a Brownian noise is uncorrelated with respect to the previous time steps; this choice can be interpreted as if the Gaussian random fluctuations in a Monte Carlo simulation were introduced at the beginning of each step without any dependency to the previous steps. This is not true in the Monte Carlo simulations discussed in the previous chapters (samples are recycled from one iteration to another), but represents a preliminary step to simplify the mathematical analysis. With this modeling choice, we get the following stochastic system for N_t and C_t

$$dN_t = \left(\frac{P_t - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t + \lambda C_t + \frac{P_t}{\Lambda_{\text{eff}}} \right) dt + c_{W,N}(\Lambda_{\text{eff}}) dW_t, \quad (7.3.1)$$

$$dC_t = \left(\frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t - \lambda C_t \right) dt, \quad (7.3.2)$$

with $c_{W,N}(\Lambda_{\text{eff}})$ the amplitude of the Brownian noise, expressed in $s^{-1/2}$. The term $-\beta_{\text{eff}}/\Lambda_{\text{eff}} \times N_t$ in Eq. (7.3.1) represents a strong mean reversion process and should induce a decorrelation between the time intervals. Hence, the neutron concentration N_t should not behave as a simple Brownian motion, but should present a mean reversion behaviour. It should be noted that we should also add a Brownian noise term to the equation for C_t being correlated to the noise on N_t , but for simplicity we add it only to the equation for N_t .

7.3.2 Reparametrization

Since the neutron transport variables N_t and C_t are related to each other, we introduce the vector Z_t defined by

$$Z_t = \begin{pmatrix} N_t \\ C_t \end{pmatrix}, \quad (7.3.3)$$

and we keep the three thermal-hydraulics Eqs. (7.2.4), (7.2.5) and (7.2.6) in their initial form.

Based on Eqs. (7.3.1) and (7.3.2), Z_t satisfies the following vector-valued stochastic differential equation

$$dZ_t = \left(A_t Z_t + \frac{P_t}{\Lambda_{\text{eff}}} e^{(1)} \right) dt + \vec{c}_W(\Lambda_{\text{eff}}) dW_t, \quad (7.3.4)$$

with the matrix

$$A_t = \begin{pmatrix} \frac{P_t - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} & \lambda \\ \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} & -\lambda \end{pmatrix}, \quad (7.3.5)$$

the vector

$$\vec{c}_W(\Lambda_{\text{eff}}) = \begin{pmatrix} c_{W,N}(\Lambda_{\text{eff}}) \\ 0 \end{pmatrix}, \quad (7.3.6)$$

and $e^{(i)}$ the vector of the canonical basis, so that

$$e^{(1)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

In order to simplify the calculations, we choose the following notations

$$\tau = \frac{\Lambda_{\text{eff}}}{\beta_{\text{eff}}},$$

$$r = \frac{\lambda \Lambda_{\text{eff}}}{\beta_{\text{eff}}}.$$

According to Table 7.2, the parameter τ has dimension of a time, and is very small: $\tau \approx 10^{-3}$ s. The parameter r is dimensionless and represents the ratio between neutron and precursor concentrations at equilibrium. This ratio is also very small: of the order of $r \approx 10^{-4}$. To summarize, we have

$$\tau \ll 1 \text{ s},$$

$$r \ll 1.$$

7.3.3 Limitations of the modeling choices

We have made several approximations in order to simplify the study of the system. The different assumptions are discussed below.

- For simplicity, we have chosen a Brownian noise to mimic the noise coming from Monte Carlo simulations. However, Brownian increments are independent, which is not the case for the random fluctuations coming from Monte Carlo transport. Indeed, particles are transferred from one step to another, so that steps are not independent. For a more faithful representation of the fluctuations, the Brownian noise should be replaced by a Gaussian noise with correlations between the time steps.
- A Brownian noise should also be added to the equation of the precursor concentration. Furthermore, this noise should be correlated to the noise on N_t .
- We have assumed that the system is starting from the equilibrium state, i.e., with the initial conditions $N_0 = C_0 = T_0 = P_0^\alpha = (dP_t^\alpha/dt)_0 = 0$. This is a restrictive hypothesis which limits the following analysis to systems close to the critical state. For more general results, the system should be studied with different initial conditions.

We will carry out our analysis with these hypotheses at first.

7.4 Preliminary results on Z_t

7.4.1 Linear equation

One may be tempted to solve the stochastic differential Eq. (7.3.4) for Z_t , which is a linear equation with a perturbative term, in the following way. Assume $(M_t)_{0 \leq t \leq T}$ is a 2×2 matrix-valued function of time, continuously differentiable, satisfying the following system

$$\begin{cases} dM_t = A_t M_t dt, \\ M_0 = I_2, \end{cases}$$

with I_2 the 2×2 identity matrix and A_t defined by Eq. (7.3.5). The existence of such a system is ensured because this is a linear ordinary differential equation, with bounded and continuous coefficient A_t . We can easily demonstrate that the matrix M_t^{-1} exists and satisfies

$$\begin{cases} dM_t^{-1} &= -A_t M_t^{-1} dt, \\ M_0^{-1} &= I_2. \end{cases}$$

We can use the Itô product rule on the product $M_t^{-1} Z_t$ (see Revuz and Yor (2004), Chapter IV, Exercise 3.9):

$$\begin{aligned} d(M_t^{-1} Z_t) &= M_t^{-1} dZ_t + dM_t^{-1} Z_t \\ &= M_t^{-1} \left(A_t Z_t + \frac{P_t}{\Lambda_{\text{eff}}} e^{(1)} \right) dt + M_t^{-1} \overrightarrow{c_W}(\Lambda_{\text{eff}}) dW_t - M_t^{-1} A_t Z_t dt \\ &= M_t^{-1} \left(\frac{P_t}{\Lambda_{\text{eff}}} e^{(1)} \right) dt + M_t^{-1} \overrightarrow{c_W}(\Lambda_{\text{eff}}) dW_t. \end{aligned}$$

Finally, we get the following expression for Z_t

$$Z_t = M_t \left[\int_0^t M_s^{-1} \frac{P_s}{\Lambda_{\text{eff}}} (UZ_s + e^{(1)}) ds + \int_0^t M_s^{-1} \overrightarrow{c_W}(\Lambda_{\text{eff}}) dW_s \right], \quad (7.4.1)$$

with the matrix

$$U = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

For further analysis of the vector Z_t , we need to have an explicit representation of the matrix M_t . As a first step, we verify whether the matrices A_t and A_s commute. Indeed, if they commute, Eq. (A.5.6) would give the following expression for M_t : $M_t = \exp(\int_0^t A_s ds)$.

7.4.2 Commutation of A_t and A_s

The matrix A_t defined by Eq. (7.3.5) is stochastic and depends on time. For our analysis, we decompose A_t into a constant matrix and a time-dependent matrix:

$$A_t = \frac{\mathcal{A}}{\tau} + P_t U, \quad (7.4.2)$$

with the matrix

$$\mathcal{A} = \begin{pmatrix} -1 & r \\ 1 & -r \end{pmatrix} \quad (7.4.3)$$

The matrices A_t and A_s commute if and only if $A_t \times A_s = A_s \times A_t$. However we have

$$A_t \times A_s - A_s \times A_t = \left(\frac{\mathcal{A}}{\tau} \times U - U \times \frac{\mathcal{A}}{\tau} \right) (P_s - P_t).$$

Thus, A_t and A_s commute if and only if $\mathcal{A} \times U = U \times \mathcal{A}$. However,

$$\mathcal{A} \times U = \begin{pmatrix} -1 & 0 \\ 1 & 0 \end{pmatrix} \text{ and } U \times \mathcal{A} = \begin{pmatrix} -1 & r \\ 0 & 0 \end{pmatrix}.$$

Therefore, the matrices A_t and A_s do not commute and we cannot use the result of Eq. (A.5.6). Since we do not have an explicit representation of M_t , Eq. (7.4.1) might not be the good starting point for the study of Z_t . Instead, we derive an alternative representation using the Itô lemma.

7.4.3 Alternative representation

Using the notations introduced above, we have the following stochastic differential equation for the vector Z_t , which is an Itô process

$$dZ_t = \left(\frac{\mathcal{A}}{\tau} Z_t + \frac{P_t}{\Lambda_{\text{eff}}} (UZ_t + e^{(1)}) \right) dt + \vec{c}_W(\Lambda_{\text{eff}}) dW_t. \quad (7.4.4)$$

Note that:

$$UZ_s = \begin{pmatrix} N_s \\ 0 \end{pmatrix}.$$

When applying the Itô lemma, recalled by Eq. (A.5.3), to Z_t , with the function f defined by $f(t, x) = e^{-\mathcal{A}t/\tau} x$, we obtain the following proposition.

Proposition 7.4.1

$$Z_t = Z_t^W + Z_t^P, \quad (7.4.5)$$

with

$$Z_t^W = \int_0^t e^{\frac{\mathcal{A}}{\tau}(t-s)} \vec{c}_W(\Lambda_{\text{eff}}) dW_s, \quad (7.4.6)$$

$$Z_t^P = \int_0^t e^{\frac{\mathcal{A}}{\tau}(t-s)} \frac{P_s}{\Lambda_{\text{eff}}} (UZ_s + e^{(1)}) ds. \quad (7.4.7)$$

The first term, Z_t^W , defined in Prop. (7.4.1) is due to the noise coming from the Monte Carlo transport. The second term, Z_t^P , comes from the coupling between thermal-hydraulics (through the perturbative reactivity P_s) and neutron transport (through the neutron concentration in UZ_s). At equilibrium, the total excess reactivity is null: $P_t = 0$. Therefore, we consider P_t as a perturbation and Z_t^W represents the unperturbed stochastic system.

7.5 Conclusion

In order to better understand the propagation of the noise along the coupling iterations of Monte Carlo transport/thermal-hydraulics calculations, we have selected a simple model of five equations representing the interactions between neutron transport and thermal-hydraulics. In order to mimic the stochastic behaviour of a Monte Carlo code, we have added a Brownian noise term on the equation for the neutron concentration.

For further analysis, a noise term should also be added on the equation on the precursor concentration. Moreover, correlations between time steps should be taken into account. Hence, Brownian noise is not a faithful representation of the noise due to Monte Carlo transport. We have also made the restrictive hypothesis that the system starts from the equilibrium.

The key result of our preliminary analysis is that we have decomposed the contributions related to neutron transport in two parts. The first part comes from the Brownian noise; the second part comes from the coupling with the thermal-hydraulics. Both contributions will be investigated in Chapter 8. The objective will be to study the variance-covariance matrix of the full system.

Chapter 8

Analytical and numerical analysis of the model

In this chapter, we consider the system of five equations presented in Chapter 7, and we provide a preliminary basis for further analytical investigation of the system. More specifically, we will analyse the probability distribution of Z_t^W in Sec. 8.1 and the probability distribution of Z_t^P in Sec. 8.2. Numerical simulations will be also presented in order to assess the validity of the different hypotheses that we make. As mentioned in Chapter 7, we start from the equilibrium, i.e., from the critical state.

8.1 Analysis of the Brownian term

8.1.1 Gaussian process

In this section, we examine Z_t^W , the part of Z_t due to the Brownian noise, as if the neutron transport system were not coupled to the thermal-hydraulics (i.e., when the excess reactivity is null: $P_t = 0$). The objective is to investigate the variance-covariance matrix Σ_t^W of Z_t^W (the element (i, j) of the matrix Σ_t^W is the covariance between the i -th and j -th elements of Z_t^W). As a reminder of Eq. (7.4.6), Z_t^W is defined by

$$Z_t^W = e^{\frac{\mathcal{A}t}{\tau}} \times \int_0^t e^{-\frac{\mathcal{A}s}{\tau}} \vec{c}_W(\Lambda_{\text{eff}}) dW_s.$$

Therefore, Z_t^W is the product of $e^{\frac{\mathcal{A}t}{\tau}}$ and a Wiener integral (see Revuz and Yor (2004), Chapter IV, Exercise 2.16), thus Z^W is a Gaussian process and its characteristics are known.

Proposition 8.1.1 Z^W is a Gaussian process, centered, and its marginal distribution at time t is $\mathcal{N}\left(0, \tau \int_0^t e^{\mathcal{A}s} \vec{c}_W(\Lambda_{\text{eff}}) \vec{c}_W(\Lambda_{\text{eff}})^\top (e^{-\mathcal{A}s})^\top ds\right)$. Its covariance function (between two marginals) is

$$\forall t \geq s, \quad \text{Cov}(Z_t^W, Z_s^W) = e^{\frac{\mathcal{A}}{\tau}(t-s)} \text{Var}(Z_s^W). \quad (8.1.1)$$

For a proof of Prop. (8.1.1), see Sec. A.5.5.

8.1.2 Properties of the matrix \mathcal{A}

The matrix \mathcal{A} defined by Eq. (7.4.3) satisfies the following properties. The proof of the different statements are provided in Sec. A.5.4.

Proposition 8.1.2 1. $\forall k \geq 1, \mathcal{A}^k = [-(1+r)]^{k-1} \times \mathcal{A}$.

2. The spectrum of \mathcal{A} is $\text{Sp}(\mathcal{A}) = \{0, -(1+r)\}$.

3. $\forall s, e^{\mathcal{A}s} = \text{Id} + \mathcal{A} \times J(s)$, with

$$J(s) = \frac{1 - e^{-(1+r)s}}{1+r}.$$

Thus, $\forall s$,

$$e^{\mathcal{A}s} = \begin{pmatrix} \frac{r + e^{-(1+r)s}}{1+r} & r \times \frac{1 - e^{-(1+r)s}}{1+r} \\ \frac{1 - e^{-(1+r)s}}{1+r} & \frac{1+r}{1+r} \end{pmatrix}. \quad (8.1.2)$$

4. The matrix $e^{\mathcal{A}s}$ is invertible and $e^{-\mathcal{A}s} = \text{Id} + \mathcal{A} \times J(-s)$.

8.1.3 Analysis of the covariance matrix

Using Props. (8.1.2) and (8.1.1), we can compute the variance-covariance matrix Σ_t^W of Z_t^W

$$\Sigma_t^W = c_{W,N}^2 \times \tau \times \begin{pmatrix} \int_0^{\frac{t}{\tau}} [1 - J(s)]^2 ds & \int_0^{\frac{t}{\tau}} [1 - J(s)] \times J(s) ds \\ \int_0^{\frac{t}{\tau}} [1 - J(s)] \times J(s) ds & \int_0^{\frac{t}{\tau}} J(s)^2 ds \end{pmatrix}.$$

Proposition 8.1.3

$$\Sigma_t^W = c_{W,N}^2 \times \tau \times M_r \left(\frac{t}{\tau} \right), \quad (8.1.3)$$

when defining the matrix function

$$M_r(t) = \begin{pmatrix} \int_0^t \left(\frac{r + e^{-(1+r)s}}{1+r} \right)^2 ds & \int_0^t \left(\frac{r + e^{-(1+r)s}}{1+r} \right) \times \left(\frac{1 - e^{-(1+r)s}}{1+r} \right) ds \\ \int_0^t \left(\frac{r + e^{-(1+r)s}}{1+r} \right) \times \left(\frac{1 - e^{-(1+r)s}}{1+r} \right) ds & \int_0^t \left(\frac{1 - e^{-(1+r)s}}{1+r} \right)^2 ds \end{pmatrix}.$$

We first detail the matrix $M_r \left(\frac{t}{\tau} \right)$:

$$M_r \left(\frac{t}{\tau} \right) = \frac{1}{(1+r)^2} \times \begin{pmatrix} r^2 \times \frac{t}{\tau} + \frac{1 - e^{-2(1+r)\frac{t}{\tau}}}{2(1+r)} + 2r \times \frac{1 - e^{-(1+r)\frac{t}{\tau}}}{1+r} & r \times \frac{t}{\tau} + \frac{(1-r) \times (1 - e^{-(1+r)\frac{t}{\tau}})}{1+r} + \frac{e^{-2(1+r)\frac{t}{\tau}} - 1}{2(1+r)} \\ r \times \frac{t}{\tau} + \frac{(1-r) \times (1 - e^{-(1+r)\frac{t}{\tau}})}{1+r} + \frac{e^{-2(1+r)\frac{t}{\tau}} - 1}{2(1+r)} & \frac{t}{\tau} + \frac{1 - e^{-2(1+r)\frac{t}{\tau}}}{2(1+r)} + 2 \times \frac{e^{-(1+r)\frac{t}{\tau}} - 1}{1+r} \end{pmatrix}. \quad (8.1.4)$$

The matrix function M_r only depends on the time t . We can identify three regimes: short time ($t \ll \tau$), long-time ($\tau \ll t \ll \tau/r$) and very long-time ($t \gg \tau/r$). Note that τ is the duration of the fission chain and $\tau/r = 1/\lambda$ is the typical precursor decay constant.

	short-time approximation	reference
$\text{Var}[N_t^W]$	1.00×10^{-4}	1.00×10^{-4}
$\text{Var}[C_t^W]$	6.42×10^{-12}	6.48×10^{-12}
$\text{Cor}[N_t^W, C_t^W]$	8.66×10^{-1}	8.66×10^{-1}

Table 8.1 – Verification of the short-time analytical approximation for the variance of N_t^W , C_t^W and their correlation, at short-time range ($t = 10^{-6}$ s).

Short-time approximation ($t \ll \tau$)

In the short-time approximation ($t \ll \tau$, i.e., $t \ll 2$ ms for the chosen model parameters), the time is much shorter than the duration of the fission chain. Thus, we can expect that the behaviour is driven by neutrons and not precursors. When we look at the short-time dynamics of the matrix M_r , using the fact that $r \ll 1$, we get

$$\Sigma_t^W \underset{t \ll \tau}{\approx} c_{W,N}^2 \times \tau \times \begin{pmatrix} \frac{t}{\tau} & \frac{1}{2} \times \left(\frac{t}{\tau}\right)^2 \\ \frac{1}{2} \times \left(\frac{t}{\tau}\right)^2 & \frac{1}{3} \times \left(\frac{t}{\tau}\right)^3 \end{pmatrix}. \quad (8.1.5)$$

Similarly as we have defined Z_t^W , we can define N_t^W and C_t^W :

$$Z_t^W = \begin{pmatrix} N_t^W \\ C_t^W \end{pmatrix}, \quad (8.1.6)$$

for which we have, according to Eq. (8.1.5):

$$\text{Var}[N_t^W] \underset{t \ll \tau}{\approx} c_{W,N}^2 \times t, \quad (8.1.7)$$

$$\text{Var}[C_t^W] \underset{t \ll \tau}{\approx} \frac{c_{W,N}^2}{3} \times \frac{t^3}{\tau^2}, \quad (8.1.8)$$

$$\text{Cov}[N_t^W, C_t^W] \underset{t \ll \tau}{\approx} \frac{c_{W,N}^2}{2} \times \frac{t^2}{\tau}. \quad (8.1.9)$$

In this regime, the variance of N_t^W increases linearly with time. On the contrary, the correlation between N_t^W and C_t^W does not depend much on time:

$$\text{Cor}[N_t^W, C_t^W] \underset{t \ll \tau}{\approx} \frac{\sqrt{3}}{2}.$$

The evaluation of the coefficients of the matrix M_r detailed in Eq. (8.1.4) substantiates the short-time approximation introduced here. For example, with $t = 10^{-6}$ s and $c_{W,N} = 10 \text{ s}^{-1/2}$ (the choice of $c_{W,N}$ will be motivated in the long-time approximation), we have obtained the numerical results presented in Table 8.1, which support our theoretical findings.

Long-time approximation ($\tau \ll t \ll \tau/r$)

Now we look at the long-time dynamics of Σ_t^W ($\tau \ll t \ll \tau/r$, i.e., 2 ms $\ll t \ll 12$ s), where neutrons are locally in equilibrium with the precursors. Using the fact that $e^{-(1+r)\frac{t}{\tau}} \rightarrow 0$ and $r \times e^{-(1+r)\frac{t}{\tau}} \rightarrow 0$, we get

$$M_r \left(\frac{t}{\tau} \right) \approx \begin{pmatrix} r^2 \times \frac{t}{\tau} + \frac{1}{2(1+r)} + 2r \times \frac{1}{1+r} & r \times \frac{t}{\tau} + \frac{1-r}{1+r} - \frac{1}{2(1+r)} \\ r \times \frac{t}{\tau} + \frac{1-r}{1+r} - \frac{1}{2(1+r)} & \frac{t}{\tau} + \frac{1}{2(1+r)} - 2 \times \frac{1}{1+r} \end{pmatrix}. \quad (8.1.10)$$

	long-time approximation	reference
$\text{Var}[N_t^W]$	1.13×10^{-1}	1.13×10^{-1}
$\text{Var}[C_t^W]$	5.00×10^2	5.00×10^2
$\text{Cor}[N_t^W, C_t^W]$	3.01×10^{-2}	3.01×10^{-2}

Table 8.2 – Verification of the long-time analytical approximation for the variance of N_t^W , C_t^W and their correlation, at long-time range ($t = 5$ s).

We also have

$$\begin{aligned} r &\ll 1, \\ \frac{t}{\tau} &\gg 1, \\ r \times \frac{t}{\tau} &\ll 1, \end{aligned}$$

so we get the following approximation

$$M_r\left(\frac{t}{\tau}\right)_{\tau \ll t \ll \tau/r} \approx \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{\tau} & \frac{1}{\tau} \end{pmatrix}. \quad (8.1.11)$$

From this we can deduce that

$$\text{Var}[N_t^W]_{\tau \ll t \ll \tau/r} \approx \frac{c_{W,N}^2 \times \tau}{2}, \quad (8.1.12)$$

$$\text{Var}[C_t^W]_{\tau \ll t \ll \tau/r} \approx c_{W,N}^2 \times t, \quad (8.1.13)$$

$$\text{Cov}[N_t^W, C_t^W]_{\tau \ll t \ll \tau/r} \approx \frac{c_{W,N}^2 \times \tau}{2}. \quad (8.1.14)$$

Within the long-time approximation, it is interesting to notice that the variance of N_t^W does not depend much on time. The variance is proportional to the square of $c_{W,N}$, which is a free parameter in our model. In order to model a standard deviation of about 0.1 (which corresponds to a standard deviation of 10% on N_t^W), we set $c_{W,N} = 10 \text{ s}^{-1/2}$.

In this regime, however, the correlation between N_t^W and C_t^W is time-dependent:

$$\begin{aligned} \text{Cor}[N_t^W, C_t^W]_{\tau \ll t \ll \tau/r} &\approx \frac{\frac{\tau}{2}}{\sqrt{\frac{\tau}{2} \times t}} \\ &\approx \sqrt{\frac{\tau}{2t}}. \end{aligned}$$

The exact numerical calculation of the coefficients of the matrix M_r confirms the long-time approximation. For example, the results are compared in Table 8.2, with $t = 5$ s and $c_{W,N} = 10 \text{ s}^{-1/2}$.

Since the variance of N_t^W does not depend on time, for two time steps t and s we have

$$\text{Var}(N_t^W)_{\tau \ll t \ll \tau/r} \approx \text{Var}(N_s^W), \quad (8.1.15)$$

and using Eq. (A.5.12), Prop. (8.1.1) gives the following result.

Proposition 8.1.4 $\forall t, s : \tau \ll t - s \ll \tau/r,$

$$\text{Cor}(N_t^W, N_s^W) \approx e^{-\frac{(1+r)(t-s)}{\tau}}.$$

Proposition (8.1.4) shows an interesting result: the correlations between the neutron concentration at two times s and t decrease exponentially with the difference $t - s$, with a decay constant α^W defined by

$$\alpha^W = \frac{1+r}{\tau}, \quad (8.1.16)$$

which is equivalent to

$$\alpha^W = \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} + \lambda \quad (8.1.17)$$

$$\approx \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}}. \quad (8.1.18)$$

This analytical value is very close to the inverse of the fission chain average length, for critical configurations ($k_{\text{eff}} \approx 1$). Indeed, in the case of a critical configuration, the average fission chain length is precisely $\Lambda_{\text{eff}}/\beta_{\text{eff}}$ (cf. Eq. (3.3.1)).

In the simulations discussed in the previous chapters (see e.g., Sec. 3.1), we have noticed that, for a given system, the values of β_{eff} and Λ_{eff} do not depend much on the reactivity insertion. Thus, the correlation decay time that we have found in Eq. (8.1.17) will be approximately constant for different configurations of a same system. However, one should keep in mind that all the analysis developed in this work is valid only for configurations close to the critical state.

It is interesting to compare the analytical value in Eq. (8.1.17) with the decay time that we have numerically found in Sec. 3.3 with TRIPOLI-4 simulations. For the critical configuration, $1/\alpha^W \approx 2$ ms is in good agreement with the best-fit time constant found in Fig. 3.14a. For the subcritical configuration however, the decay time given by $1/\alpha^W$ is much longer than the 0.1 ms obtained with the best-fit time constant in Fig. 3.14b: using the kinetics parameters of the rod-drop configuration for the calculation of $1/\alpha^W$ is not sufficient to get a good estimation of the decay constant. In fact, the rod-drop configuration is very far from this equilibrium state, thus the analytical analysis developed in this section can not be used.

Very long-time approximation ($t \gg \tau/r$)

Since we usually do not simulate more than a few seconds for the typical reactor transients considered in the previous chapters, we will not consider this case, where t would be longer than τ/r , which is about 12 s. It should be noted however that in this case the variance of N_t^W is no longer constant. Indeed, when looking at Eq. (8.1.10), the term $r^2 \times t/\tau$ can not be neglected anymore. Thus, the variance increases linearly with time. This is coherent with the findings in Williams (1971); Houchmandzadeh et al. (2015).

8.1.4 Numerical simulations of the system

We have performed numerical simulations of N_t^W and C_t^W (cf. Fig. 8.1) in order to monitor the evolution of $\text{Var}[N_t^W]$, $\text{Var}[C_t^W]$ and $\text{Cor}[N_t^W, C_t^W]$ over time, as presented in Fig. 8.1. We have performed 10^4 independent replicas using fourth-order Runge-Kutta integration, according to the following algorithm:

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4),$$

with

$$\begin{aligned}
 k_1 &= f(t_n, y_n), \\
 k_2 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right), \\
 k_3 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right), \\
 k_4 &= f(t_n + h, y_n + hk_3), \\
 t_n &= t_n + h,
 \end{aligned}$$

where h is the integration time step. At the end of each integration time step, a Brownian motion is added to the deterministic solution for N_t^W . In practice, we have generated a normal distribution with standard deviation \sqrt{h} . We have observed that the integration time step necessary to achieve convergence of the solution with the Brownian term is much smaller than the time step necessary to achieve convergence of the deterministic system (i.e., with $c_{W,N} = 0$). The retained simulation parameters are the following: the step-size is $h = 10^{-4}$ s, the initial time is $t_0 = 0$ s, and the total simulation time is 5 s (the number of steps is $n = 5 \times 10^4$). For more clarity on the plot, only one realization is presented for N_t^W and C_t^W (red lines). The sample variances over all the realizations are plotted (blue lines). For comparison, the analytical expression that we obtained for the variances are also plotted (green lines) and are in good agreement with the simulations.

In Figs. 8.1a and 8.1c, it is clearly visible that the neutron time scale is much shorter than the precursor time scale. The variances of both N_t^W and C_t^W (Figs 8.1b and 8.1d) behave as expected: the one of N_t^W varies rapidly, and after a very short time, the variance fluctuates around 0.11, as expected. The variance of C_t^W increases linearly with time, as it would be the case for a Brownian motion, and we find $\text{Var}[C_t^W] \approx 500$ at the final time $t = 5$ s, as found in Table 8.2. We have also plotted the time evolution of the correlation between N_t and C_t in Fig. 8.1f: the correlation decreases with time towards a constant value around 0.03, which is coherent with the analytical result obtained using the long-time approximation (cf. Table 8.2).

8.1.5 Numerical simulations with TRIPOLI-4

We want now to verify the findings that we have obtained with our analytical study with respect to the TRIPOLI-4 kinetic simulations presented in Sec. 3.3.1. Figure 8.2 shows the time evolution of the variance of the total statistical weight of the neutron population as simulated by TRIPOLI-4. It is interesting to see that the plot is divided in two parts. First, the variance increases linearly with time up to $t = 1$ ms. Then, the variance remains stable. This observation is in good agreement with the analytical results that we have derived in this section: in particular, at short-time range the variance seems to obey Eq. (8.1.7) and at long-time range Eq. (8.1.12), which is consistent with our theoretical prediction.

Moreover, we can deduce an order of magnitude for $c_{W,N}$: the linear fit as well as the constant value provide a value of about 0.1 s^{-1} for $c_{W,N}^2$. However, the coefficient depends on the convergence of the TRIPOLI-4 simulation. Here, the simulation was well-converged: the relative standard error is about 1% in each time step. For consistency with dynamic simulations, which are much more time-consuming, we can target an uncertainty of about 10%, which would lead to $c_{W,N} = 1 \text{ s}^{-1/2}$, which is relatively close to the value of $c_{W,N} = 10 \text{ s}^{-1/2}$ that we have set up for our simplified model.

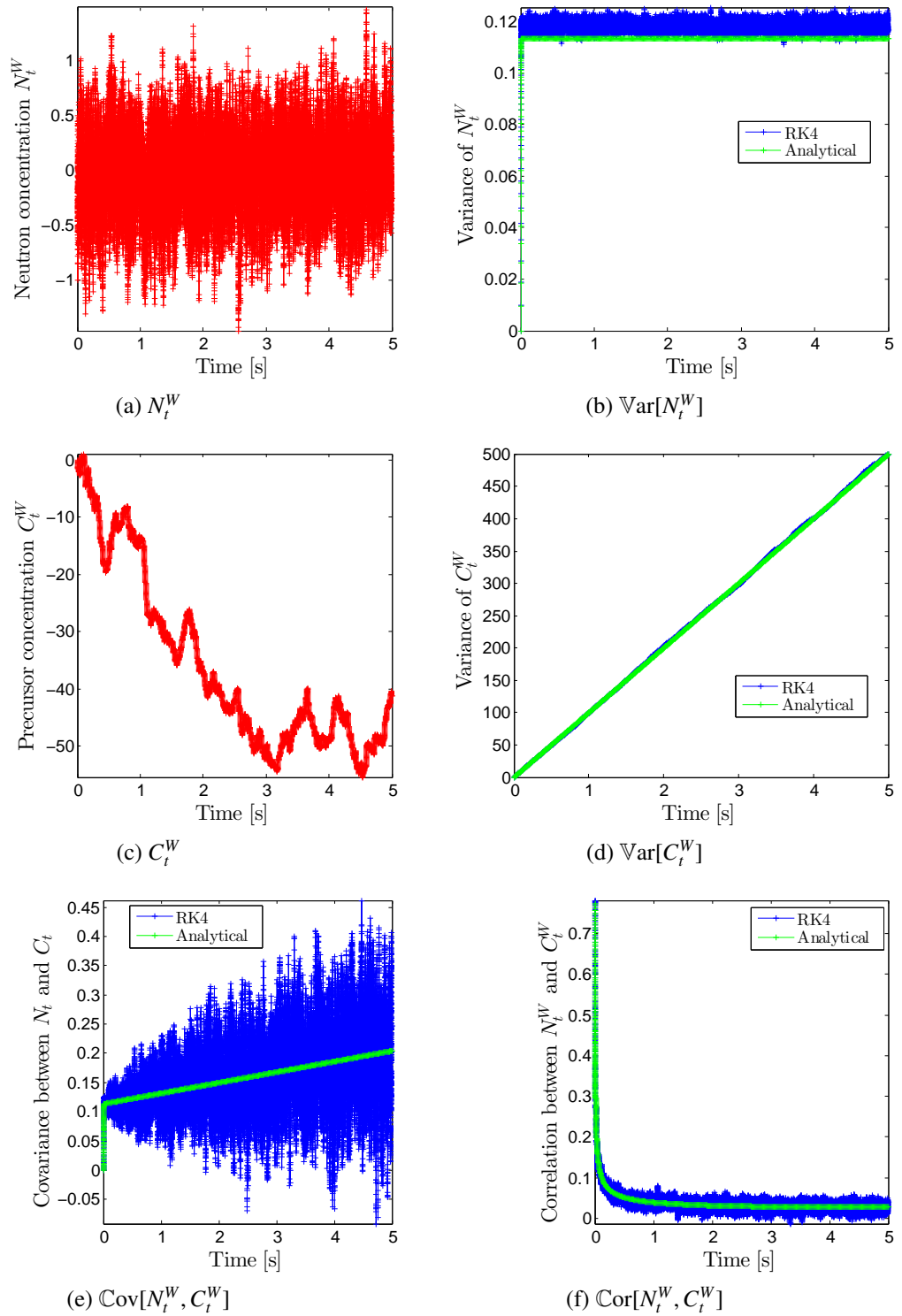


Figure 8.1 – 10^4 independent simulations of the neutron transport equations with $P_t = 0$ (without coupling with the thermal-hydraulics) were performed using the fourth-order Runge-Kutta method. The step for numerical integration is 10^{-4} s. The red lines show the time evolution of the neutron population N_t^W and the precursor population C_t^W (one realization only is presented for more clarity). The blue lines show sample estimates of $\text{Var}[N_t^W]$, $\text{Var}[C_t^W]$, $\text{Cov}[N_t^W, C_t^W]$ and $\text{Cor}[N_t^W, C_t^W]$ and the green lines show the analytical results, which are in good agreement with the simulations.

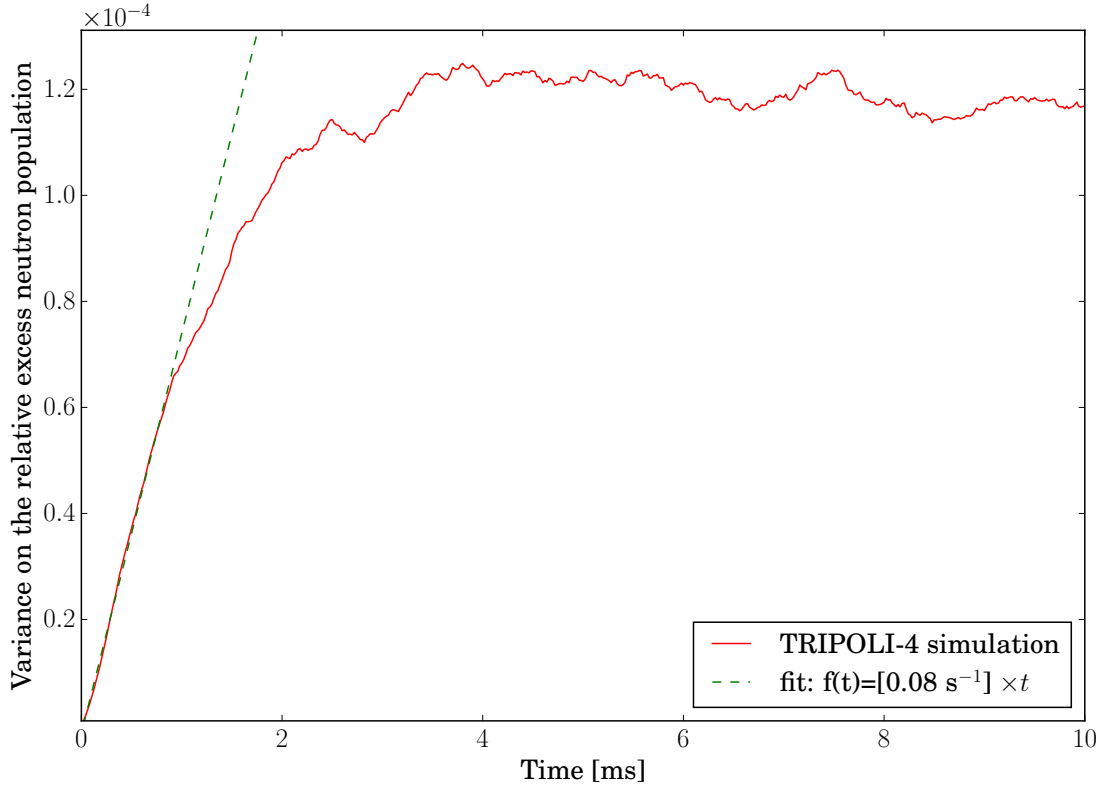


Figure 8.2 – Time evolution of the variance on the neutron population, for the TRIPOLI-4 simulation presented in Sec. 3.3.1. The plot is divided in two parts: the variance of N_t depends linearly on time up to $t \approx 1$ ms, then it reaches a plateau.

8.1.6 Conclusion

In this section, we have examined the propagation of the Brownian noise that we added on N_t for a critical system, without the coupling with the thermal-hydraulics (i.e., $P_t = 0$). More specifically, we have investigated the covariance matrix Σ_t^W of Z_t^W . Three regimes were identified. At short-time range, ($t \ll \tau$), $\text{Var}[N_t^W]$ linearly depends on time, while there is a cubic dependency for $\text{Var}[C_t^W]$. At long-time range ($\tau \ll t \ll \tau/r$), $\text{Var}[N_t^W]$ does not depend on time, while $\text{Var}[C_t^W]$ depends linearly on it. The very long-time regime ($t \gg \tau/r$) is not of interest for the Monte Carlo simulations that we target, and was not considered.

We have found that correlations between the neutron concentration at two times, s and t , decrease exponentially with time, with a decay constant $\alpha^W = \beta_{\text{eff}}/\Lambda_{\text{eff}} + \lambda$, which is about the inverse of the average duration of the fission chain.

Numerical calculations of the variance of N_t^W and C_t^W , as well as their correlation were performed in order to confirm the approximations and to plot the evolution of the correlation along time. The variance of N_t^W is characteristic of the mean reversion process. On the contrary, the variance of C_t^W increases linearly with time, as expected for a Brownian motion. The correlation between N_t^W and C_t^W exponentially decreases with time. It should be recalled that the analysis that was developed here and the conclusions that were drawn only apply to configurations close to the critical state.

8.2 Analysis of the thermal-hydraulics coupling term

We now estimate the covariance matrix of Z_t^P , Σ_t^P , which represents the contribution to the variance due to the coupling with thermal-hydraulics.

8.2.1 Preliminary analysis

After investigating Z_t^W , we study Z_t^P , the other component of Z_t . As a reminder, Eq. (7.4.7) stipulates that

$$Z_t^P = \int_0^t e^{\frac{a}{\tau}(t-s)} \frac{P_s}{\Lambda_{\text{eff}}} (UZ_s + e^{(1)}) ds,$$

which can be expanded to yield

$$Z_t^P = \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times \frac{1 + N_s}{1 + r} \left(r + e^{-\frac{(1+r)(t-s)}{\tau}} \right) ds. \quad (8.2.1)$$

This term is non-linear in N_s , since P_s depends on N_s . We can define N_t^P and C_t^P :

$$Z_t^P = \begin{pmatrix} N_t^P \\ C_t^P \end{pmatrix} \quad (8.2.2)$$

and rewrite the first component of Eq. (8.2.1) as

$$N_t^P = \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times \frac{1 + N_s}{1 + r} \times \left(r + e^{-\frac{(1+r)(t-s)}{\tau}} \right) ds, \quad (8.2.3)$$

which is not exactly integrable because of the non linearity, even if we consider the deterministic system only (i.e., $c_{W,N} = 0$). We can simplify the expression of N_t^P when looking at short-time and long-time scales, namely

$$\begin{aligned} N_t^P &\xrightarrow{t \ll \tau} \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times (1 + N_s) ds, \\ N_t^P &\xrightarrow{t \gg \tau} \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times \frac{r}{1 + r} \times (1 + N_s) ds. \end{aligned}$$

Similarly, the second component of the vector Z_t^P is C_t^P and reads

$$C_t^P = \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times \frac{1 + N_s}{1 + r} \times \left(1 - e^{-\frac{(1+r)(t-s)}{\tau}} \right) ds, \quad (8.2.4)$$

and looking at short-time and long-time regimes we get

$$\begin{aligned} C_t^P &\xrightarrow{t \ll \tau} 0, \\ C_t^P &\xrightarrow{t \gg \tau} \int_0^t \frac{P_s}{\Lambda_{\text{eff}}} \times \frac{1 + N_s}{1 + r} ds. \end{aligned}$$

When looking at N_t^P and C_t^P in the long-time range, we find the equilibrium ratio:

$$\frac{N_t^P}{C_t^P} \xrightarrow{t \gg \tau} r. \quad (8.2.5)$$

8.2.2 Numerical simulations of the full system

Given the difficulty of solving analytically the nonlinear coupled system of five equations, we have solved it numerically. We have chosen to follow the time evolution over 5 s, with time intervals of 10^{-4} s. We have performed 10^4 independent simulations using the fourth-order Runge-Kutta method introduced above and the same Brownian noise as for the uncoupled simulations. By setting the total reactivity P_t to zero at each coupling iteration, we have first checked that we obtained the same solutions as in Sec. 8.1.4. Then, we have taken into account the variations of P_t .

The time evolution of N_t , C_t (red lines) and their variance (blue lines) are presented in Fig. 8.3. The neutron population still reveals a mean reversion process. Surprisingly, the variance reaches the same plateau as for the uncoupled system, presented in Fig. 8.1a: the variance fluctuates around 0.11. Actually, it seems that with our simulation parameters the contribution to the variance due to the coupling is small as compared to the one due to the Brownian motion. On the contrary, the behaviour of the variance of the precursor population differs from the uncoupled behaviour: the variance does not increase linearly with time anymore: the feedbacks “bend” the linear trend (for comparison the linear trend is plotted in green). The time evolution of one realization of T_t , P_t^α and P_t is presented in Fig. 8.4. As expected, the behaviour of the fuel temperature is very similar to the behaviour of the precursor concentration, since their equations, Eqs. (7.2.7) and (7.2.8), have the same shape.

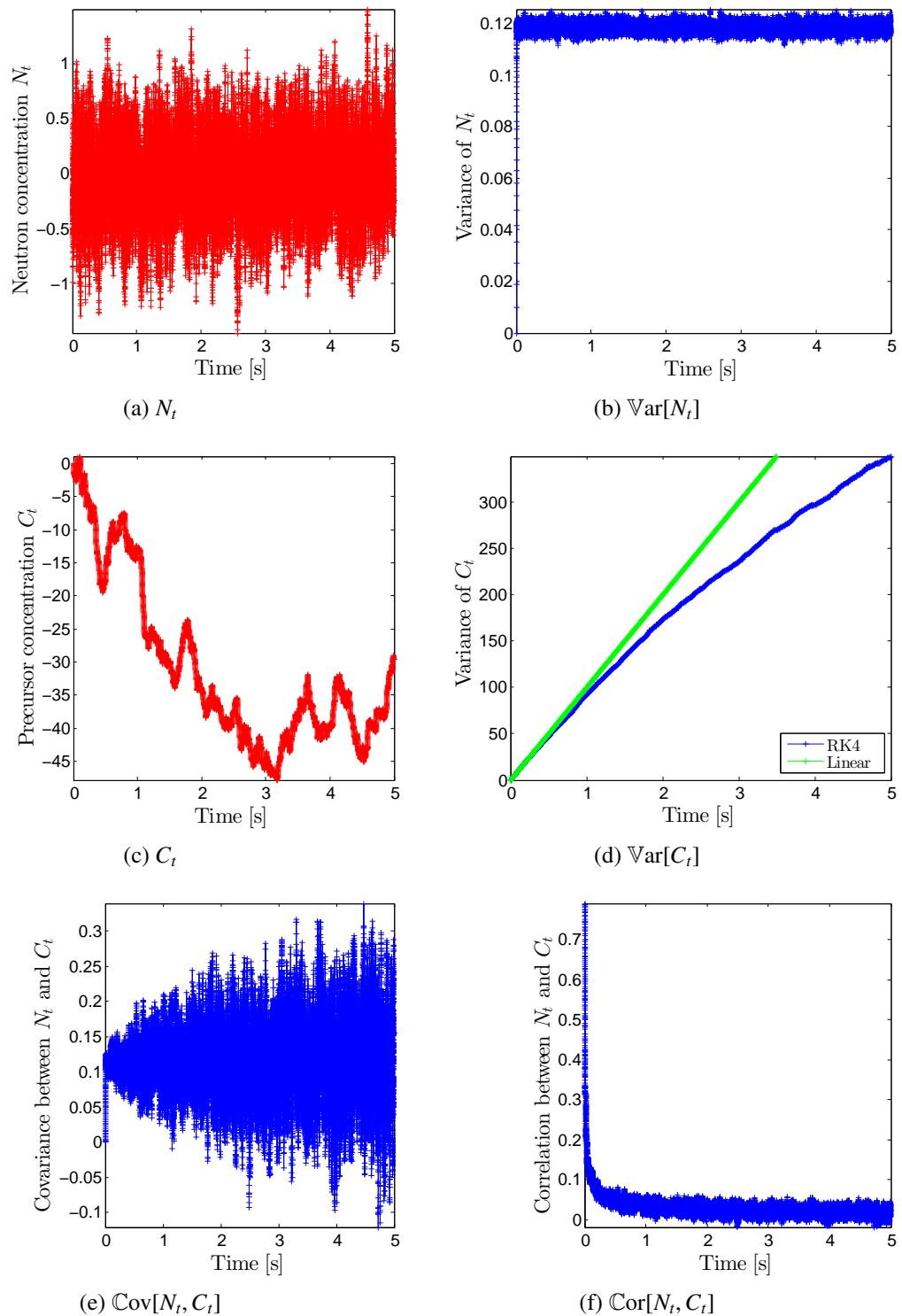


Figure 8.3 – Time evolution of the neutron population N_t and the precursor population C_t for the complete system of five equations Eqs. (7.2.2) - (7.2.6). 10^4 independent simulations using the fourth-order Runge-Kutta method were performed, but one realization only is presented for more clarity (red lines). Sample estimates of $\text{Var}[N_t]$, $\text{Var}[C_t]$, $\text{Cov}[N_t, C_t]$ and $\text{Cor}[N_t, C_t]$ are also plotted (blue lines). The time step for integration is 10^{-4} s. A linear trend is plotted in green for comparison with $\text{Var}[C_t]$.

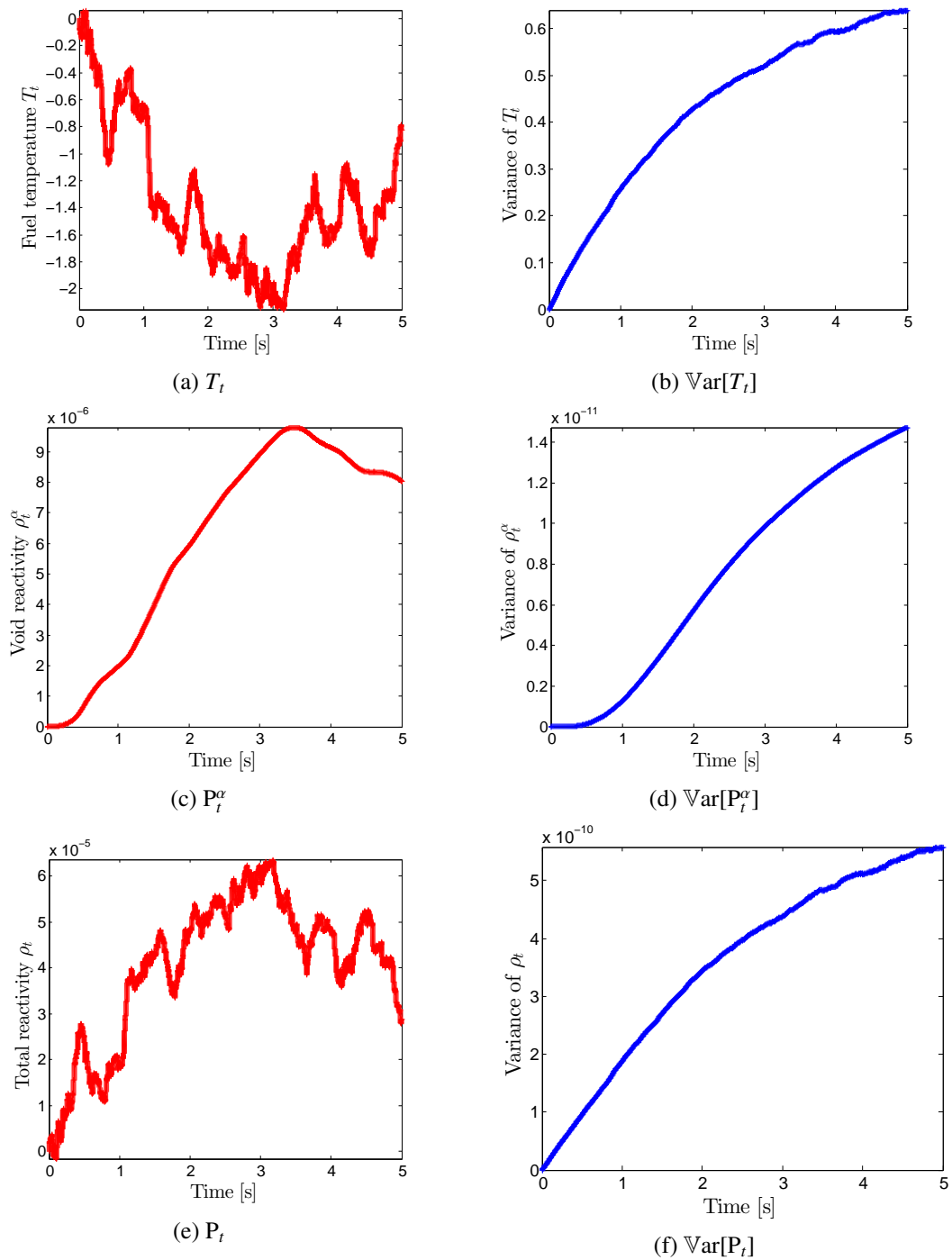


Figure 8.4 – Time evolution of the fuel temperature T_t , the void reactivity P_t^α and the total reactivity P_t (red lines). Sample estimates of the variance of the three physical quantities are also plotted (blue lines). Results come from the same simulations as in Fig. 8.3.

8.3 Simplified model

8.3.1 Hypothesis

In this section, we make two assumptions in order to be able to advance in the analytical study of the coupled term. First, we notice that the coefficients a_3 , a_4 and k in Eq. (7.2.5) for P_t^α are small. Thus, for $t \ll \tau/r$, we make the assumption that $P_t^\alpha = 0$; that is, we neglect the void reactivity which is a reasonable approximation for a PWR system. Thus, the system of five equations reduces to a system of three equations:

$$dN_t = \left(\frac{-DT_t - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t + \lambda C_t - \frac{DT_t}{\Lambda} \right) dt + c_{W,N}(\Lambda_{\text{eff}}) dW_t, \quad (8.3.1)$$

$$dC_t = \left(\frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}} N_t - \lambda C_t \right) dt, \quad (8.3.2)$$

$$dT_t = (a_1 N_t - a_2 T_t) dt. \quad (8.3.3)$$

8.3.2 Verification of the hypothesis

In order to assess the validity of the hypothesis that we have introduced, we have performed some numerical simulations of the simplified system Eqs. (8.3.1) - (8.3.3) in order to verify that they are in good agreement with the full system Eqs. (7.2.2) - (7.2.6). The simulation time is 5 s, with integration time steps of 10^{-4} s. We have performed 10^4 independent simulations using the fourth-order Runge-Kutta method and we have used the same sample of Brownian increments as for the resolution of the full model presented in Figs. 8.3 and 8.4. Results are presented in Fig. 8.5. The behaviour of N_t and C_t , as well as their variance, is very similar to the one obtained with the full model. We can conclude that the full model can be reasonably well represented by the simplified model, Eqs. (8.3.1) - (8.3.3).

8.3.3 Discussion on the non linearity

The analytical study of the variance of N_t^P seems to be very challenging because of the non linearity. It could be tempting to consider that the variations on the neutron population are small and to assume that $N_s \ll 1$, in order to suppress the non linearity in Eq. (8.2.3). However, we are precisely interested in transient configurations where the neutron population varies drastically. Therefore, we can not afford such hypothesis, and we have no choice but to deal with the non-linearity. However, the assumptions that we have made have allow for a simplification of the system: the system of five equations has been reduced to a system of three equations, which is much more convenient.

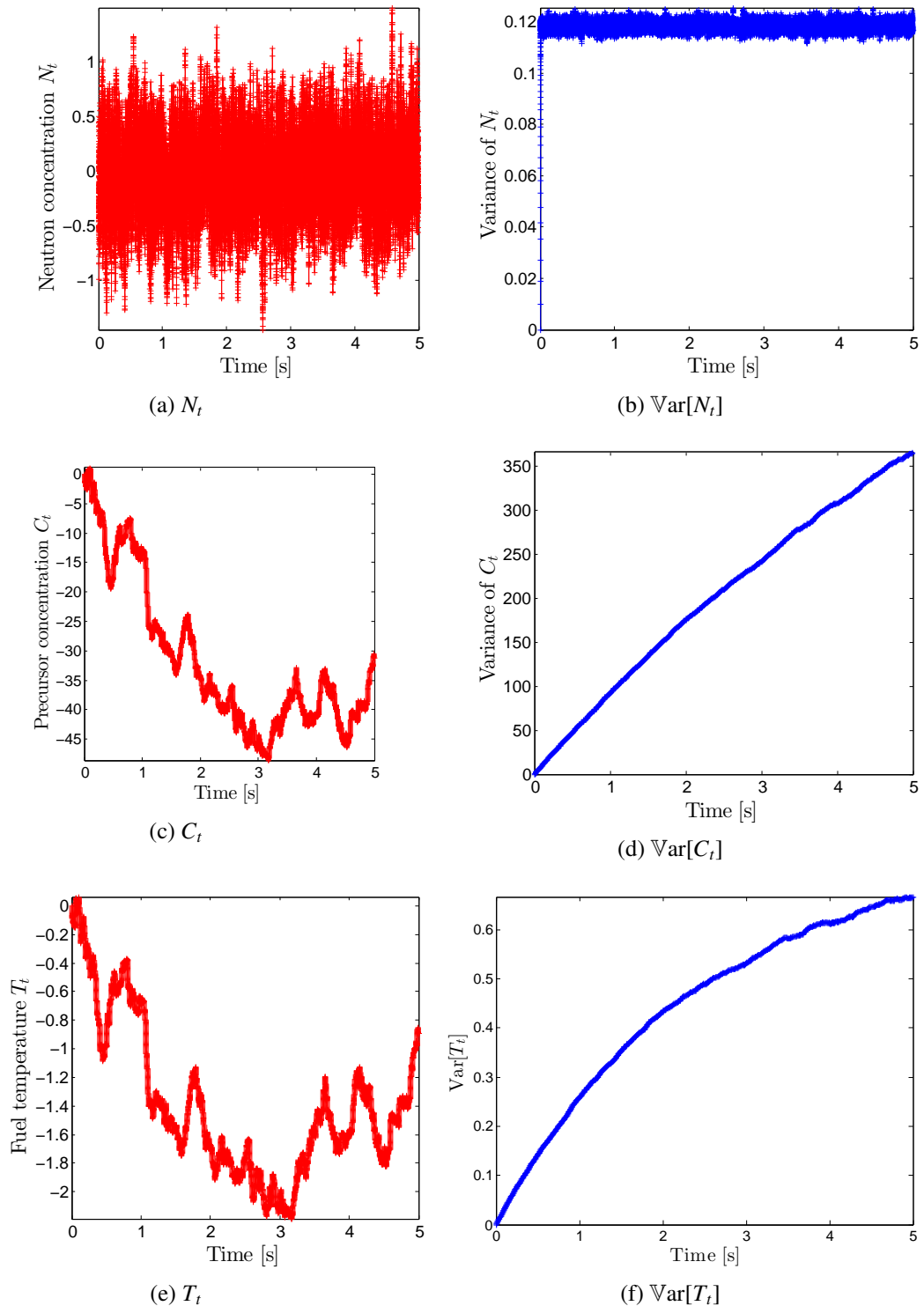


Figure 8.5 – Same as Fig.8.3, but for the simplified system ($P_t^\alpha = 0$). The results are very similar to the ones obtained with the complete system.

8.4 Conclusion

8.4.1 Brownian term

As a first step towards the analysis of the noise induced by the Monte Carlo transport, we have studied the neutron and precursor equations at fixed reactivity. More specifically, we have studied the covariance matrix Σ_t^W . The neutron concentration reveals a strong mean reversion process, while the precursor concentration behaves as a Brownian motion. Two characteristic times have been found: $\Lambda_{\text{eff}}/\beta_{\text{eff}}$, which is the duration of the fission chain for a critical system, and $1/\lambda$, which is the typical precursor decay constant. In the short-time regime ($t \ll \Lambda_{\text{eff}}/\beta_{\text{eff}}$), the variance of N_t^W increases linearly with time. In the long-time regime ($\Lambda_{\text{eff}}/\beta_{\text{eff}} \ll t \ll 1/\lambda$), the variance of N_t^W is constant. The short-time and long-time approximations of the variance of the neutron and the precursor populations were verified against numerical simulations.

8.4.2 Full system

We have then performed numerical simulations of the full system, including the coupling with thermal-hydraulics, and we have noticed that the contribution to the variance on the neutron population due to the coupling is small in comparison to the variance due to the Brownian term. This finding deserves further investigation, although an analytical analysis of the full system does not seem to be feasible. Indeed, the analysis of the covariance matrix Σ_t^P , coming from the coupling, is much more challenging than the analysis of Σ_t^W , because of the non-linearity in the neutron concentration equation. In order to simplify the system, we have neglected the void excess reactivity P_t^α , which is a reasonable approximation for PWRs. This way, we have reduced the five-equations model into a three-equations model. Even if the non-linearity can still not be neglected, the analysis of the reduced model is simpler.

8.4.3 Discussion

The analytical study of the stability of the 5-equations simplified model is a highly non-trivial task. We have made several assumptions in order to simplify the system and numerical simulations gave us some intuition on the behaviour of the different physical quantities involved in the propagation of the noise induced by the Monte Carlo transport in coupled simulations. It appears that the non-linearity, which is the major component of the coupling model, seriously hinders the full analytical analysis of the system. For further analysis, the linearization of the neutron equation seems necessary. An alternative approach would be to consider the moment equations Akcasu and Williams (2004); Sjenitzer and Hoogenboom (2011b).

Our investigation has been carried out by using a Brownian motion, which is not representative of the full complexity branching processes occurring in the Monte Carlo simulation. For further analysis, other types of noises must be considered. Also, the analysis that we have developed is valid only for small departures from the critical state. The system should be also studied with different initial conditions.

Conclusions

The purpose of this Ph. D. thesis was to develop a non-stationary coupling scheme between the Monte Carlo code TRIPOLI-4 and thermal-hydraulics, so as to provide a reference tool for the simulation of reactivity-induced transients in PWRs. This purpose includes the extensive verification of the coupling scheme, as well as the characterization of its stability. In short, this work aims at identifying the roadblocks to the simulation of a realistic reactor core in non-stationary conditions with Monte Carlo neutron transport and thermal-hydraulics feedback.

The first part of the work is devoted to testing the kinetic capabilities of TRIPOLI-4 (i.e., time dependent without thermal-hydraulics feedback), evaluating the different existing methods and implementing a new variance-reduction technique.

In Chapter 2, we have described the algorithms implemented in TRIPOLI-4 for kinetic calculations, namely those related to the explicit handling of the precursors. We have also presented the existing population-control methods (*Russian roulette and splitting* and *combing*) and variance-reduction techniques (forced decay and branchless collisions), which are necessary for kinetic calculations. We assessed the efficiency of the different methods on an assembly based on the TMI-1 reactor, in different configurations. Forced decay was systematically applied, while other methods were separately applied; future work may concern the impact of the combination of different methods. We have found that *combing* is more efficient than the *Russian roulette and splitting* as a population-control method, especially when the population size varies dramatically; the branchless collisions method has also proved to be effective in reducing the variance.

We also described a new time-dependent population importance sampling scheme for variance reduction. For the assessment of its efficiency, the method has been tested on the TMI-1 assembly and proved to be very efficient for simulations with small time steps, where forced decay fails to produce delayed neutrons with weights large enough to survive the Russian roulette. The choice of an optimal importance ratio was also examined: as the simulation time step decreases, the optimal importance ratio also decreases. The figure of merit reaches an optimal plateau, and is thus weakly dependent on the choice of the ratio. We have finally detailed another contribution to extending the TRIPOLI-4 capabilities: time-dependent geometries are now handled, which makes it possible to simulate all types of transients, such as rod extraction scenarios. Our implementation relies on the definition of duplicated geometries and needs to be optimized in order to reduce the memory occupation, which currently impedes massively parallel calculations.

In Chapter 3, we have numerically tested the kinetic Monte Carlo methods implemented in TRIPOLI-4 on two realistic configurations by examining the response of both prompt and delayed neutrons to different types of reactivity insertions. Several simulations have been performed on the experimental reactor SPERT III E-core in different configurations: critical, con-

trol rod extraction and rod drop. We have stressed the importance of taking into account precursors, even within a prompt supercritical excursion. We have also demonstrated the value of our time-dependent importance sampling scheme for observing both prompt and delayed regimes when simulating a rod drop. Simulations were also performed on a 3x3 mini-core based on the TMI-1 reactor, and benchmarked against the Monte Carlo code Serpent 2 with four different reactivity insertions. Results were satisfactory and encourage further application of the new kinetic capabilities of TRIPOLI-4. Simulations of two of these scenarios with thermal-hydraulics feedbacks are also presented in Chapter 6.

We have also presented our preliminary characterization of the correlations between the simulation time steps. Correlations due to fission chains have an exponential decay, and the decay time was empirically found to be close to the fission chain average lifetime. We have verified this hypothesis with a simple analytical model in Chapter 8. The dependency of the relative uncertainty on the spatial discretization of the kinetic scoring mesh has been also examined. This study stems from the observation that, within a time step, the relative uncertainty in a cell is not proportional to its volume, as one would naively expect. In particular, we have observed that for time steps larger than the average fission chain lifetime, the relative uncertainty on the global integrated score is roughly the same as the relative uncertainty in a cell at pin level. In fact, neutrons from the same fission chain are correlated; when they explore the system, they make strongly correlated contributions to several cells of the scoring mesh. Therefore, the variance on the global integrated score increases because of these correlations.

The second part of this work concerns the development of a multi-physics interface for TRIPOLI-4, and more specifically the development of a coupling scheme between TRIPOLI-4 and the thermal-hydraulics sub-channel code SUBCHANFLOW.

We have detailed our implementation choices in Chapter 4, namely the multi-physics interface for TRIPOLI-4 based on the development of an API combined with an external supervisor that drives TRIPOLI-4 and an external code. We have used tools from the SALOME platform: the ICoCo API for the coupling interface and the MEDCoupling library for data exchange between the two codes. TRIPOLI-4 computes the power distribution, which is transferred to SUBCHANFLOW. The thermal-hydraulics code computes the updated properties of the fuel (temperatures) and of the moderator (temperatures and densities). The temperature dependence of the cross sections is taken into account in TRIPOLI-4 with stochastic interpolation. We have side-stepped the problem of projecting the thermal-hydraulics fields onto the volumes of the TRIPOLI-4 geometry by making cells of the mesh coincide with the volumes.

Two types of calculations can be performed with the supervisor: criticality calculations with feedback and transient (i.e., dynamic) calculations. Criticality calculations with feedback are divided into outer iterations between calls to thermal-hydraulics (for the convergence of temperature and density fields) and inner iterations between calls to TRIPOLI-4 (for the convergence of fission sources). Relaxation of the power distribution is used in order to suppress oscillations between successive iterations. Transient calculations start from an initial source obtained with a preliminary criticality calculation and are integrated in time steps with an Euler explicit scheme. The criticality source (thermal-hydraulics fields and fission sources) can be stored at the end of the criticality calculation and reused for different transient simulations.

The large computing requirements of dynamic calculations made it necessary to implement the capability of the supervisor to run in parallel mode. In order to reduce the statistical fluctu-

ations of the input given to the non-linear thermal-hydraulics solver, the neutron power distribution is averaged over all the available parallel units and SUBCHANFLOW is run once. The interdependency of the simulators through the thermal-hydraulics fields makes it difficult to assess the uncertainty on the coupling results. Splitting up the whole simulation into independent replicas would produce truly independent results. It would have the additional benefit of reducing the waiting times between fast and slow simulators (see Chapter 6). However, at the same time, it would amplify the noise transferred to the thermal-hydraulics solver.

In Chapter 5, we have verified the capabilities of TRIPOLI-4 for criticality calculations with feedback. This was achieved in the framework of a benchmark established in collaboration with the Serpent and SUBCHANFLOW development teams, within the context of the McSAFE European project. The selected configuration was the TMI-1 assembly, with a pin-by-pin description. The resulting power, temperature and density distributions are in very good agreement between the TRIPOLI-4/SUBCHANFLOW and the Serpent 2/SUBCHANFLOW coupling schemes, and encourage further simulations with the new coupling scheme between TRIPOLI-4 and SUBCHANFLOW. There are several avenues to explore in order to improve the coupling scheme. For example, one could define the thermal-hydraulics simulation parameters in view of the comparison with experimental data; such parameters include the number of axial slices and the number of radial rings in the discretization of fuel pins (the latter point is especially important for future burnup calculations). Also, we could refine the coupling scheme by introducing a more realistic evaluation of the energy deposition (energy from fission, but also from scattering and capture), possibly distinguishing between power deposition in the fuel and the coolant. In any case, refining the thermal-hydraulics model requires the development of new algorithms to handle continuous space-dependent temperatures and densities.

In Chapter 6 we have presented transient simulations with thermal-hydraulics feedbacks. Calculations were performed on the TMI-1 mini-core benchmark: first the system was simulated at steady state to verify its stability; then we have introduced reactivity in the system to probe the effects of the thermal-hydraulics feedbacks. Two different rod extraction scenarios were simulated: 30 cm and 40 cm. We have benchmarked our results against the coupling between Serpent 2 and SUBCHANFLOW, and a good agreement was found. The 40 cm scenario is very challenging because of the large fluctuations on the population size, which induce large fluctuations in CPU times between the parallel units. Therefore, the calculation efficiency seriously suffers from few rogue parallel units. This simulation pointed out the necessity to harden the coupling scheme against population fluctuations. Also, averaging the power distribution over packets of parallel units would allow to reduce the waiting times between simulators. In addition to the spatial discretization choices discussed in Chapter 5, dynamic calculations raise the question of the choice of the time discretization, and in particular of the convergence of our Euler explicit scheme. Additionally, when the transient is initiated by a moving reactor part (such as a control rod), the motion must be sufficiently finely discretized.

The third part of this work consists of a preliminary study for quantifying the statistical uncertainties stemming from a dynamic calculation.

In Chapter 7, we have selected a simplified deterministic model representing the coupling between neutron transport and thermal-hydraulics in a BWR and we have extended it to accommodate a noise term, so as to conduct both numerical and analytical analysis of the noise propagation along the coupling iterations. The deterministic model consists of five ordinary differential equations for the neutron concentration, the precursor concentration, the fuel tem-

perature, the void reactivity and the total reactivity. Several hypothesis have been made in order to simplify the study. First, we have modified two input parameters in order to adapt the model to the description of a PWR. Also, the stochastic behaviour of TRIPOLI-4 has been modeled to a first approximation as a Brownian noise term on the neutron concentration equation. This is not a faithful representation of Monte Carlo fluctuations, since a Brownian noise is uncorrelated with respect to the previous time steps. For further investigation, the Brownian noise should be replaced by a Gaussian noise with correlations between the time steps.

We have presented a preliminary analysis of the system of five stochastic differential equations in Chapter 8. As a first step, we have studied the impact of the Brownian term on the neutron and precursor concentrations in the absence of feedback. The neutron concentration reveals a strong mean reversion process. We have found three different regimes: times shorter than the average duration of the fission chains, times longer than the typical precursor lifetime, and intermediate times. The variance on the neutron concentration varies rapidly in the short-time regime, and then varies linearly with time with a large time constant. The conclusions that we have obtained with our analytical study were verified against numerical simulations.

The analysis of the full system is much more challenging, because of the non-linearity in the neutron concentration equation. In order to simplify the system, we have neglected the void excess reactivity, which is a reasonable assumption for a PWR. We have thus reduced the five-equation model to a three-equation model. Even if the analysis of this model is simpler, an analytical study seems hardly feasible because of the remaining non-linear term. Thus, we have performed numerical simulations of the full system in order to characterize its behaviour. To conclude, the analytical study of the stability of the five-equation model is difficult and requires either a linearization of the feedback mechanism. An approach based on the moment equations may also prove fruitful.

Perspectives

The coupling scheme between TRIPOLI-4 and thermal-hydraulics is operational and allows to perform both stationary and non-stationary calculations with feedback, for different types of scenarios. We have tested it on short transients on a mini-core consisting of nine fuel assemblies with a rather roughly discretized thermal-hydraulics feedback. These calculations are close to the limit of what is feasible in the current state of affairs, both in terms of time and memory requirements. As far as calculation time is concerned, increasing the number of simulators is not yet an option because it would also increase the waiting times due to the thermal-hydraulics rendez-vous points, which are already a hindrance with 1000 processors. As a short-term workaround, a reorganization of the parallel execution scheme could be envisaged. In the long-term, it seems necessary to perform tight population control in order to avoid large fluctuations on the population size. New variance-reduction techniques could also help to achieve this goal. Concerning the memory limitations, this issue could be dealt with by modifying the representation of the temperature and density fields in TRIPOLI-4. Up to now, the fields are represented in a discrete way using the volumes of the geometry as a support. Therefore, a fine representation requires geometries consisting of a very large number of volumes (already more than 100 000 for the mini-core). If the temperatures and densities were continuously representable, independently of the volumes, the memory requirements would be much lower. These modifications, both in terms of time and memory, should achieve valuable improvements and unlock the simulation of systems of larger size.

It should be emphasized that this thesis led to a benchmark work with another existing stochastic scheme, with a given thermal-hydraulics solver. A simple model and coupling scheme were chosen, the only purpose being the comparison of the two multi-physics tools. For validation purposes however, improvements to the models and the coupling scheme should be considered. Convergence studies should be performed for both TRIPOLI-4 and thermal-hydraulics models (e.g., the spatial discretization of the resolution mesh, such as the number of axial slices). The coarseness of the time discretization is also of paramount importance for convergence. The time step should be carefully chosen, with an adaptive method if necessary, in order to refine the calculation where the system is changing rapidly.

Appendix

A.1 Useful formulas for the point kinetics

The derivation of the point-kinetics equations was detailed in Sec. 1.2.3. The system of equations, for neutrons and M groups of precursors, is the following

$$\begin{aligned}\frac{\partial}{\partial t} \tilde{n}(t) &= \frac{\rho - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} \tilde{n}(t) + \sum_{j=1}^M \lambda_j \tilde{c}_j(t), \\ \frac{\partial}{\partial t} \tilde{c}_j(t) &= \frac{\beta_{j,\text{eff}}}{\Lambda_{\text{eff}}} \tilde{n}(t) - \lambda_j \tilde{c}_j(t).\end{aligned}$$

The system can be written in matrix form as

$$\dot{p}(t) = Ap(t), \tag{A.1.1}$$

with

$$p(t) = \begin{pmatrix} \tilde{n}(t) \\ \tilde{c}_1(t) \\ \dots \\ \tilde{c}_M(t) \end{pmatrix},$$

$$A = \begin{pmatrix} (\rho - \beta_{\text{eff}})/\Lambda_{\text{eff}} & \lambda_1 & \dots & \lambda_M \\ \beta_{1,\text{eff}}/\Lambda_{\text{eff}} & -\lambda_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \beta_{j,\text{eff}}/\Lambda_{\text{eff}} & 0 & -\lambda_j & 0 \\ \dots & \dots & \dots & \dots \\ \beta_{M,\text{eff}}/\Lambda_{\text{eff}} & 0 & \dots & -\lambda_M \end{pmatrix}.$$

For simplicity, we consider the system for one family of precursors with time constant λ_{eff} , and we solve it by diagonalizing A . The characteristic equation is

$$(\rho - \beta_{\text{eff}} - \Lambda_{\text{eff}}\omega)(\lambda_{\text{eff}} + \omega) + \beta_{\text{eff}}\lambda = 0, \tag{A.1.2}$$

which can be rewritten as

$$\Lambda_{\text{eff}}\omega^2 + (\beta_{\text{eff}} - \rho + \lambda\Lambda_{\text{eff}})\omega - \rho\lambda = 0. \tag{A.1.3}$$

Equation (A.1.3) admits real solutions if

$$\Delta = (\beta_{\text{eff}} - \rho + \lambda\Lambda_{\text{eff}})^2 + 4\rho\lambda\Lambda_{\text{eff}} > 0. \tag{A.1.4}$$

Such condition is trivially satisfied if $\rho \geq 0$ (i.e., critical or supercritical configurations). In a subcritical configuration, the condition reads

$$\beta_{\text{eff}} > -\left(\sqrt{|\rho|} - \sqrt{\lambda\Lambda_{\text{eff}}}\right)^2, \quad (\text{A.1.5})$$

which is also trivially true. Therefore, we shall assume that Eq. (A.1.3) always admit real solutions. Its eigenvalues are

$$\omega_{\pm} = \frac{\rho - \beta_{\text{eff}} - \lambda\Lambda_{\text{eff}} \pm \sqrt{\Delta}}{2\Lambda_{\text{eff}}}. \quad (\text{A.1.6})$$

It is instructive to consider the condition

$$\frac{\lambda\Lambda_{\text{eff}}}{\beta_{\text{eff}}} \ll 1 - \frac{\rho}{\beta_{\text{eff}}}, \quad (\text{A.1.7})$$

which is typically satisfied in PWRs. Equation (A.1.4) can be rewritten to make the ratio $\lambda\Lambda_{\text{eff}}/\beta_{\text{eff}}$ explicit:

$$\Delta = \beta_{\text{eff}}^2 \left[\left(1 - \frac{\rho}{\beta_{\text{eff}}} + \frac{\lambda\Lambda_{\text{eff}}}{\beta_{\text{eff}}} \right)^2 + 4 \frac{\rho}{\beta_{\text{eff}}} \frac{\lambda\Lambda_{\text{eff}}}{\beta_{\text{eff}}} \right].$$

Therefore, to first order in $\lambda\Lambda_{\text{eff}}/\beta_{\text{eff}}$, one has

$$\Delta = (\beta_{\text{eff}} - \rho)^2 + 2(\beta_{\text{eff}} + \rho)\lambda\Lambda_{\text{eff}} + o\left[\left(\frac{\lambda\Lambda_{\text{eff}}}{\beta_{\text{eff}}}\right)^2\right],$$

and

$$\sqrt{\Delta} = |\beta_{\text{eff}} - \rho| + \frac{(\beta_{\text{eff}} + \rho)}{|\beta_{\text{eff}} - \rho|} \lambda\Lambda_{\text{eff}} + o\left[\left(\frac{\lambda\Lambda_{\text{eff}}}{\beta_{\text{eff}}}\right)^2\right].$$

There are two distinct regimes: prompt subcritical ($\rho < \beta$) and prompt supercritical ($\rho > \beta$).

For prompt subcritical ($\rho < \beta_{\text{eff}}$), to first order in $\lambda\Lambda_{\text{eff}}/\beta_{\text{eff}}$, the eigenvalues are

$$\begin{aligned} \omega_+ &= \lambda \frac{\rho}{\beta_{\text{eff}} - \rho} \\ \omega_- &= \frac{\rho - \beta_{\text{eff}}}{\Lambda_{\text{eff}}}. \end{aligned}$$

For prompt supercritical ($\rho > \beta_{\text{eff}}$), to first order in $\lambda\Lambda_{\text{eff}}/\beta_{\text{eff}}$, the eigenvalues are

$$\begin{aligned} \omega_+ &= \frac{\rho - \beta_{\text{eff}}}{\Lambda_{\text{eff}}} \\ \omega_- &= \lambda \frac{\rho}{\beta_{\text{eff}} - \rho}. \end{aligned}$$

We can cast these results in a unified form by defining the prompt and delayed eigenvalues

$$\omega_p = \frac{\rho - \beta_{\text{eff}}}{\Lambda_{\text{eff}}}, \quad (\text{A.1.8})$$

$$\omega_d = \lambda \frac{\rho}{\beta_{\text{eff}} - \rho}. \quad (\text{A.1.9})$$

A.2 Flattop-Pu

The Flattop-Pu benchmark (OECD Nuclear Energy Agency, 1995) has a simple spherical geometry. It is made of an inner plutonium sphere with an outer natural uranium reflector, as shown by Fig. A.1. The inner radius is $R_1 = 4.53$ cm and the outer radius is $R_2 = 24.14$ cm. The description of the compositions is detailed in Table A.1.

The multiplication factor and the adjoint-weighted kinetics parameters have been obtained with TRIPOLI-4 by running a regular power iteration calculation with 100 inactive cycles and 1000 active cycles, each with 2×10^5 neutrons. Nuclear data for this configuration are taken from the ENDF/B-VII library. Results are displayed in Table A.2. The decay constants have also been extracted during the power iteration and are displayed in Table A.3; the average β -weighted decay constant $\bar{\lambda}$ computed by TRIPOLI-4 is also provided.

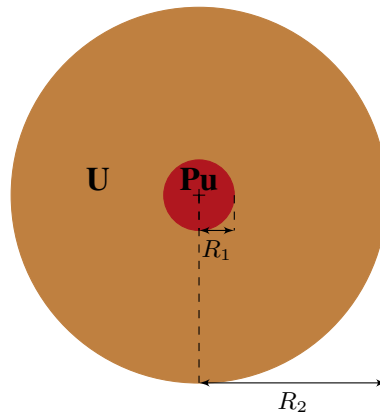


Figure A.1 – A 2-D cross-section view of Flattop-Pu, a plutonium sphere with a natural uranium reflector.

isotop	density [10^{24} atoms.cm $^{-3}$]
Pu239	3.6697×10^{-2}
Pu240	1.8700×10^{-3}
Pu241	1.1639×10^{-4}
Ga69	8.8692×10^{-5}
Ga71	5.8858×10^{-5}
U234	2.6438×10^{-6}
U235	3.4610×10^{-4}
U238	4.7721×10^{-2}

Table A.1 – Material compositions for the Flattop-Pu benchmark, expressed as atomic densities.

configuration	critical
k_{eff}	$1.000081 \pm 7.9 \times 10^{-5}$
Λ_{eff} [ns]	13.330 ± 0.017
β_{eff} [pcm]	279.3 ± 2.1
$\beta_{\text{eff},1}$ [pcm]	6.73 ± 0.32
$\beta_{\text{eff},2}$ [pcm]	58.58 ± 0.97
$\beta_{\text{eff},3}$ [pcm]	46.27 ± 0.88
$\beta_{\text{eff},4}$ [pcm]	115.0 ± 1.4
$\beta_{\text{eff},5}$ [pcm]	43.09 ± 0.82
$\beta_{\text{eff},6}$ [pcm]	9.54 ± 0.38
ρ [pcm]	8.1 ± 7.9
ρ [\$]	~ 0

Table A.2 – Multiplication factor and adjoint-weighted kinetics parameters for Flattop-Pu in two different configurations, as computed by TRIPOLI-4 with the ENDF/B-VII nuclear data library. Precursors are regrouped into 6 families. Error bars are not given for the reactivity expressed in dollars because we did not measure correlations between k_{eff} and β_{eff} .

λ_1	0.01250190
λ_2	0.03027633
λ_3	0.1131895
λ_4	0.3330504
λ_5	1.323749
λ_6	9.989597
$\bar{\lambda}$	0.105716

Table A.3 – Decay constants, in s^{-1} , for Flattop-Pu at nominal conditions, as given by the ENDF/B-VII nuclear data library. Precursors are regrouped into 6 families. The average β -weighted decay constant $\bar{\lambda}$ is also provided. Error bars are negligible (about 0.01%).

A.3 SPERT III E-core

The Special Power Excursion Reactor Test III (SPERT-III) is a small pressurized-water research reactor. It was built in the United States in the 1960s (Heffner and Wilson, 1961; Houghtaling et al., 1965; McCardell et al., 1969) in order to investigate transient behaviour of nuclear reactors. The E-core type consists of a pressurized light-water-moderated core with 4.8%-enriched UO_2 fuel pellets arranged in a regular lattice of cylindrical pins. It contains 60 assemblies, including 48 fuel assemblies with 25 (5x5) pin-cells, 4 assemblies with 16 (4x4) pin-cells, and 8 control rods moving pairwise. The control rods contain fuel in the lower section and a neutron absorber (18-8 stainless steel with 1.35 weight percent of ^{10}B) in the upper section. The full insertion of the control rods into the core constitutes the shutdown configuration. At the center of the core, a transient cruciform rod is made of the same neutron absorber as in the control rods in the lower part and of 18-8 stainless steel in the upper part. In the experimental campaign that was carried out in the 1960s, the transient rod was rapidly ejected from the core to initiate power excursion: the lower absorber part is located outside the core during the transient (Heffner and Wilson, 1961; Houghtaling et al., 1965; McCardell et al., 1969). The description of the compositions is provided in Table A.4.

A.4 The TMI-1 3x3 mini-core

The TMI-1 3x3 mini-core benchmark, defined by Ivanov et al. (2013b), is made of nine 15x15 assemblies. Each fuel assembly is made of 204 4.12%-enriched UOX with Zircaloy-4 cladding. There are also 4 burnable poison pins with $Gd_2O_3+UO_2$ fuel and Zircaloy-4 cladding, and an instrumentation tube located in the center. Additionally, there are 16 guide tubes which are filled with borated water, except for the central assembly, where they contain control rods made of Ag-In-Cd and Inconel cladding. The moderator consists of borated light water. Assemblies are surrounded by a reflector (width is 21.64 cm), made of stainless steel and borated water. The critical boron concentration is 1493 ppm for TRIPOLI-4 model. The active fuel length is 353.06 cm, and the width of each assembly is 21.64 cm. For a critical Hot Zero Power configuration, the control rods are fully inserted, meaning that the insertion depth is 353.06 cm. The imensions of the system are provided in Table A.5 and the compositions are listed in Table A.6.

element	parameter	dimension [cm]
fuel pins	pellet radius	0.4695
	gas radius	0.4788
	cladding radius	0.5461
instrumentation tube	inner radius	0.56005
	outer radius	0.6261
guide tubes	inner radius	0.63245
	outer radius	0.6731
control rods	radius	0.56007
	cladding radius	0.61722
general	pin pitch	1.4427
	assembly width	21.81
	water blade thickness	0.1695
	active height	353.06

Table A.5 – Dimensions for the TMI-1 3x3 mini-core benchmark (Ivanov et al., 2013b).

A.5 Mathematical tools

A.5.1 Itô process

Definition

An Itô process $(X_t)_{t \geq 0}$ is a stochastic process which can be expressed as

$$X_t = x_0 + \int_0^t b(s, X_s) ds + \int_0^t \sigma(s, X_s) dW_s, \quad (\text{A.5.1})$$

with x_0 the deterministic initial condition, b called the drift coefficient and σ the diffusion coefficient. W is a Wiener process (also called a Brownian motion).

Itô's lemma

Assume X_t is an Itô process with a drift coefficient b and a diffusion coefficient σ and let $f \in C^{1,2}(\mathbb{R}^+ \times \mathbb{R}^q, \mathbb{R})$. Itô's lemma states that the process $(f(t, X_t))_{0 \leq t \leq T}$ is also an Itô process and that it satisfies (see (Revuz and Yor, 2004), Chapter IV, Theorem 3.3)

$$f(t, X_t) = f(0, x_0) + \int_0^t \partial_s f(s, X_s) ds + \int_0^t \nabla_x f(s, X_s) b_s ds + \int_0^t \nabla_x f(s, X_s) \sigma_s dW_s \quad (\text{A.5.2})$$

$$+ \frac{1}{2} \int_0^t \sum_{k,l=1}^q f''_{x_k, x_l}(s, X_s) [\sigma_s \sigma_s^\top]_{k,l} ds. \quad (\text{A.5.3})$$

Itô integral

Let W_t be a Wiener process and f a stochastic process. The following integral is also a stochastic process and is called “Itô integral”:

$$I(f) = \int_0^\infty f(t) dW_t, \quad (\text{A.5.4})$$

for which we have the following property:

$$\mathbb{E}(I^2(f)) = \int_0^t \mathbb{E}(f^2(t)) dt. \quad (\text{A.5.5})$$

Note: if f is a deterministic function, $I(f)$ is called “Wiener integral”.

A.5.2 Useful properties on matrices

Commuting matrices

Assume $(A_t)_{0 \leq t \leq T}$ is a matrix-valued function of time and $(Z_t)_{0 \leq t \leq T}$ is vector-valued function of time, continuously differentiable such that they satisfy the linear ordinary differential equation $\dot{Z}_t = A_t Z_t$. If A_t and A_s commute for any s and t , then the solution Z_t can be represented explicitly as

$$Z_t = e^{\int_0^t A_s ds} Z_0, \quad (\text{A.5.6})$$

where $e^{\mathcal{A}}$ is the exponential of the matrix \mathcal{A} defined by

$$e^{\mathcal{A}} = \sum_{k=0}^{\infty} \frac{\mathcal{A}^k}{k!}. \quad (\text{A.5.7})$$

Exponential matrix

A square matrix A is always diagonalizable in \mathbb{C} . Plus, we know that $\text{Sp}(e^A) = e^{\text{Sp}(A)} \neq 0$, hence the exponential matrix e^A has only non-zeros eigenvalues, and is therefore always invertible. Moreover, e^{-A} is its inverse since $e^A e^{-A} = e^{A-A} = e^0 = \text{Id}$.

Covariance matrix of a Gaussian random vector

Assume M is a matrix and \vec{G} is a Gaussian random vector of i.i.d. centered reduced variables, i.e., $\forall i, G_i \sim \mathcal{N}(0, 1)$. Then the covariance matrix Σ of the random vector $M\vec{G}$ is

$$\begin{aligned}\Sigma &= \mathbb{E} \left(((M\vec{G})_i (M\vec{G})_j)_{ij} \right) \\ &= \mathbb{E} \left((M\vec{G})(M\vec{G})^\top \right) \\ &= \mathbb{E} \left(M\vec{G}\vec{G}^\top M^\top \right) \\ &= M\mathbb{E} \left(\vec{G}\vec{G}^\top \right) M^\top.\end{aligned}$$

It gives

$$\Sigma = MM^\top. \quad (\text{A.5.8})$$

A.5.3 Nomenclature for the analysis of the March-Leuba system

In the analysis performed in Chapters 7 and 8 and the sections below, we use the following notations:

- \dot{Z} : time derivative of Z ,
- $^\top$: transposed matrix,
- N'_t : neutron concentration,
- C'_t : precursor concentration,
- T'_t : fuel temperature [K],
- P_t^{α} : void reactivity,
- P'_t : total reactivity,
- λ : typical decay constant of precursors [s^{-1}],
- Λ_{eff} : effective mean generation time [s],
- β_{eff} : effective delayed neutron fraction,
- W : Wiener process,
- $\vec{c}_W(\Lambda_{\text{eff}})$: vector with two components proportional to a power of Λ_{eff} ,
- $c_{W,N}(\Lambda_{\text{eff}})$: amplitude of the Brownian noise on N_t [$s^{-1/2}$],
- I_2 : 2x2 identity matrix,
- Z_t : vector with N_t and C_t as components,
- Z_t^W : part of Z_t due to the Brownian motion,
- Z_t^P : part of Z_t due to the coupling with the reactivity P_t ,
- N_t^W : part of N_t due to the Brownian motion,
- N_t^P : part of N_t due to the coupling with the reactivity P_t ,
- C_t^W : part of C_t due to the Brownian motion,

- C_t^P : part of C_t due to the coupling with the reactivity P_t ,
 $\tau = \frac{\Lambda_{\text{eff}}}{\beta_{\text{eff}}}$: average time scale for a neutron to be converted into a precursor [s],
 $r = \frac{\lambda \Lambda_{\text{eff}}}{\beta_{\text{eff}}}$: ratio between the average time scales for a neutron to be converted into a precursor,
 and conversely,
 Σ_t : covariance matrix of Z_t ,
 Σ_t^P : covariance matrix of Z_t^P ,
 Σ_t^W : covariance matrix of Z_t^W .

A.5.4 Properties on the matrix \mathcal{A}

Recurrence relation

It is interesting to notice that

$$\begin{aligned}\mathcal{A}^2 &= \begin{pmatrix} 1+r & -r-r^2 \\ -1-r & r+r^2 \end{pmatrix} \\ &= -(1+r) \times \mathcal{A}.\end{aligned}$$

Thus, the characteristic polynomial of \mathcal{A} is $P_{\mathcal{A}}(\lambda) = \lambda^2 + (1+r) \times \lambda$, and the matrix \mathcal{A} has two eigenvalues:

$$\text{Sp}(\mathcal{A}) = \{0, -(1+r)\}.$$

Furthermore, by induction, we deduce the following relation:

$$\forall k \geq 1, \quad \mathcal{A}^k = [-(1+r)]^{k-1} \times \mathcal{A}. \quad (\text{A.5.9})$$

Exponential matrix

Using the recurrence relation Eq. (A.5.9), we develop the exponential matrix of \mathcal{A} :

$$\begin{aligned}e^{\mathcal{A}s} &= \sum_{k=0}^{\infty} \frac{\mathcal{A}^k s^k}{k!} \\ &= \text{Id} + \sum_{k=1}^{\infty} \mathcal{A} \times \frac{[-(1+r) \times s]^k}{-(1+r) \times k!} \\ &= \text{Id} + \mathcal{A} \times \left[\frac{e^{-(1+r) \times s} - 1}{-(1+r)} \right] \\ &= \text{Id} + \mathcal{A} \times I(s),\end{aligned}$$

with

$$I(t) = \frac{1 - e^{-(1+r)t}}{1+r}.$$

Thus, the exponential matrix can be written

$$e^{\mathcal{A}s} = \begin{pmatrix} 1 - I(s) & r \times I(s) \\ I(s) & 1 - r \times I(s) \end{pmatrix} = \begin{pmatrix} \frac{r + e^{-(1+r)s}}{1+r} & r \times \frac{1 - e^{-(1+r)s}}{1+r} \\ \frac{1 - e^{-(1+r)s}}{1+r} & \frac{1 + r e^{-(1+r)s}}{1+r} \end{pmatrix}. \quad (\text{A.5.10})$$

A.5.5 Correlation among time intervals

Z_t^W satisfies the following stochastic differential equation

$$Z_t^W = e^{\frac{\alpha}{\tau}(t-s)} Z_s^W + \int_s^t e^{\frac{\alpha}{\tau}(t-u)} \overrightarrow{c_W}(\Lambda_{\text{eff}}) dW_u. \quad (\text{A.5.11})$$

Therefore, we can compute the covariance matrix between two time steps s and t

$$\text{Cov}(Z_t^W, Z_s^W) = \text{Cov}(e^{\frac{\alpha}{\tau}(t-s)} Z_s^W, Z_s^W) + \text{Cov}\left(\int_s^t e^{\frac{\alpha}{\tau}(t-u)} \overrightarrow{c_W}(\Lambda_{\text{eff}}) dW_u, Z_s^W\right).$$

Using the fact that increments of a Brownian motion between two steps s and t are independent, we get

$$\text{Cov}(Z_t^W, Z_s^W) = \text{Cov}(e^{\frac{\alpha}{\tau}(t-s)} Z_s^W, Z_s^W).$$

Finally, we have

$$\text{Cov}(Z_t^W, Z_s^W) = e^{\frac{\alpha}{\tau}(t-s)} \times \text{Var}(Z_s^W). \quad (\text{A.5.12})$$

A.6 Extended analysis of the TMI-1 3x3 mini-core dynamic simulations

In the following, we provide an extended analysis of the results obtained for the fuel temperature with T4/SCF for the transient scenario D presented in Sec. 6.3.1, with the rods being extracted by 40 cm between $t = 0.3$ s and $t = 1.3$ s. A comparison with SSS2/SCF results is also provided.

A.6.1 Time evolution of the fuel temperature

Figure A.2 shows the radial map of the increase in the fuel temperature relatively to the first time step. Three different times are considered: $t = 1.3$ s (end of the rod extraction), $t = 2.1$ s and $t = 5$ s, as well as three slices: the slice 3 is located at the bottom of the mini-core, where the reactivity is inserted through the rod extraction, while the slices 10 and 15 are located above the rod extraction. The time evolution of the fuel temperature is different for the three slices. For the slice 3, the increase in the temperature is higher at the center of the mini-core, where the rods are extracted, and reach up to 900 K. The increase is much smaller for the slice 10 (and even smaller for the slice 15), and can be observed in the surrounding assemblies.

A.6.2 Comparison between T4/SCF and SSS2/SCF

Figure A.3 shows the difference between T4/SCF and SSS2/SCF results for the fuel temperature, averaged over all axial slices, for the slices 3 and 10. Values are given for three different times: $t = 0.1$ s, $t = 1.3$ s and $t = 5$ s. The difference is larger for $t = 1.3$ s, because of the large fluctuations on the neutron population size during the power excursion. In some rods, the difference can be quite high (up to 60 K), here again this might be due to statistical fluctuations. Overall, we find a reasonable agreement between T4/SCF and SSS2/SCF.

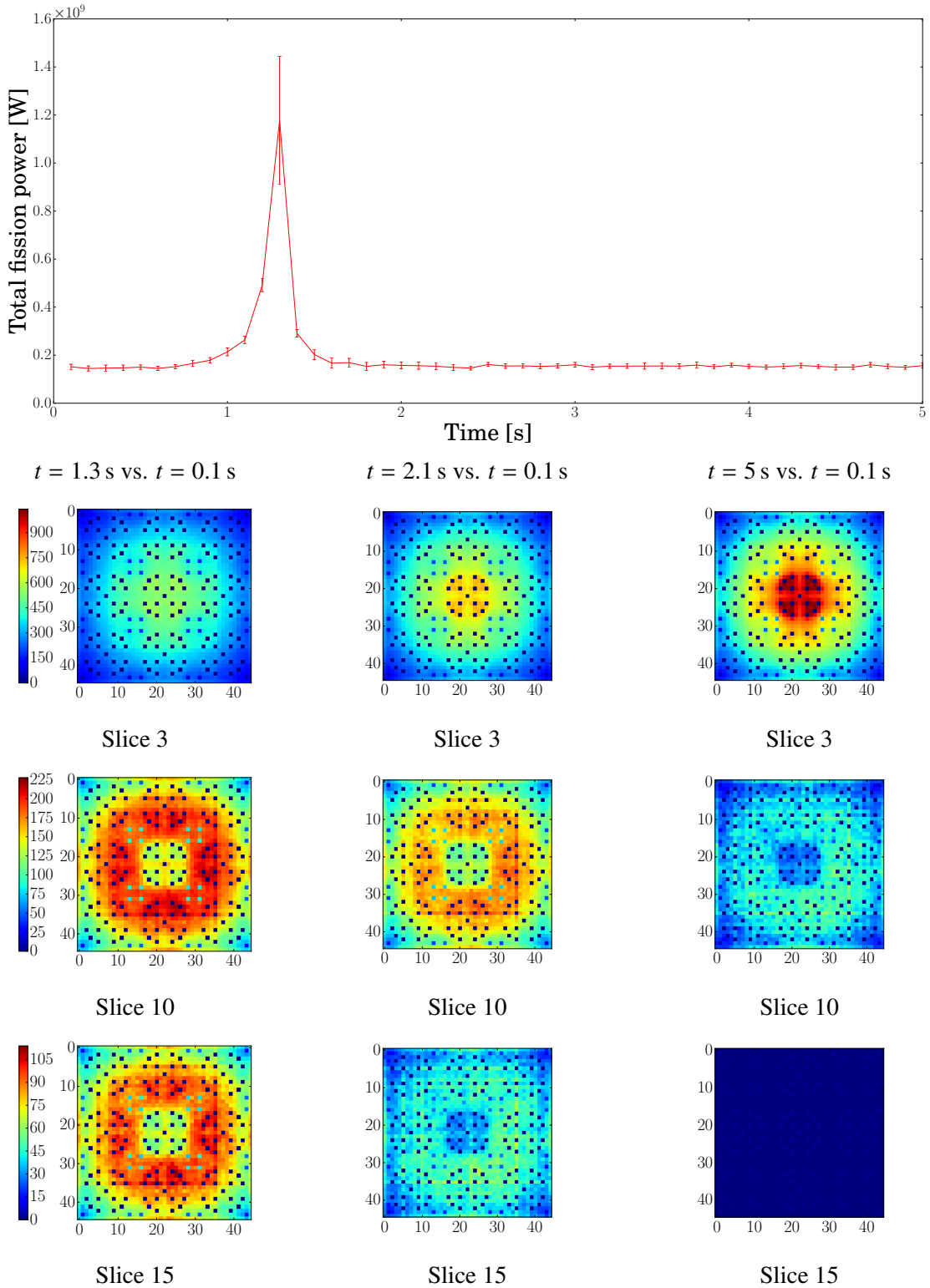


Figure A.2 – Radial map of the increase in the fuel temperature (in kelvins) obtained with T4/SCF for the TMI-1 3x3 mini-core during transient scenario D, at three different times: $t = 1.3$ s, $t = 2.1$ s and $t = 5$ s, relatively to $t = 0.1$ s. Three slices are represented: the slice 3 is located at the bottom of the mini-core, where the reactivity is inserted through the rod extraction, while the slices 10 and 15 are located above the rod extraction.

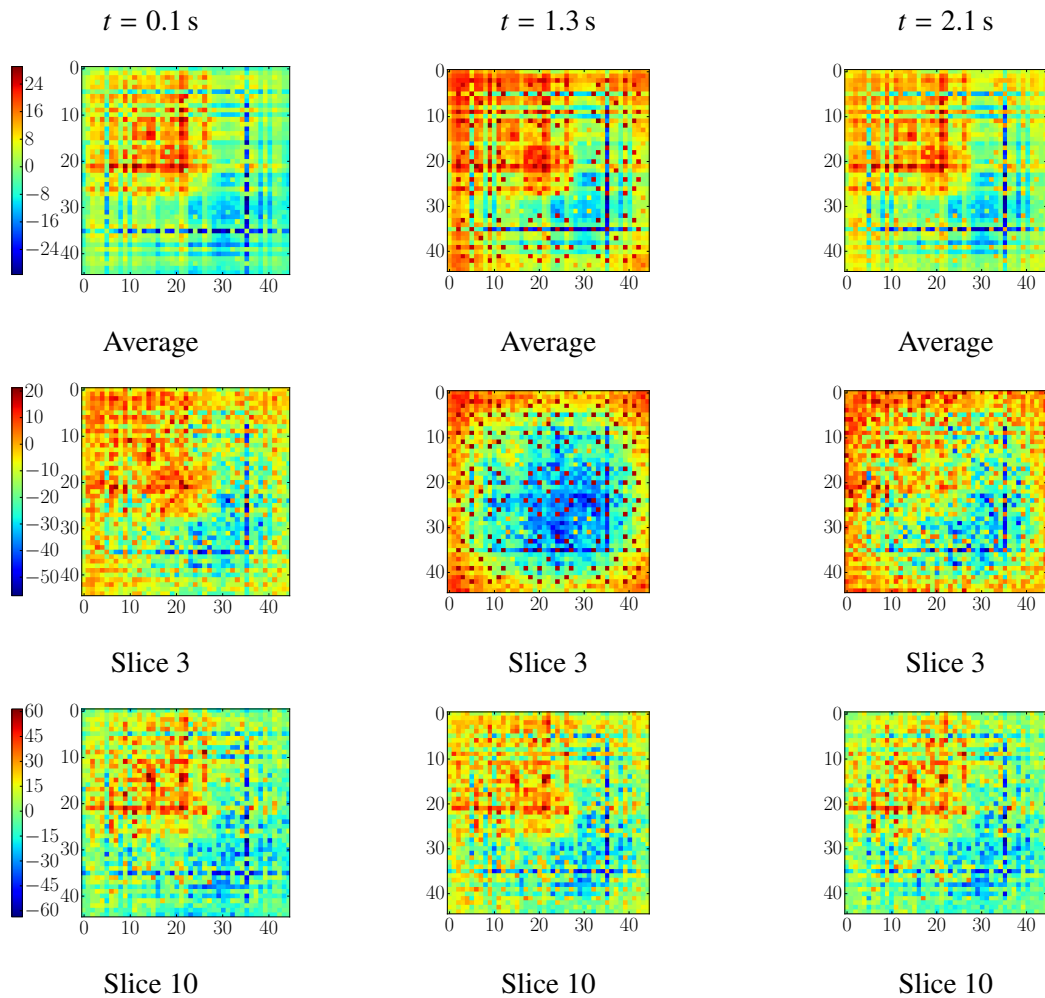


Figure A.3 – Difference map (in kelvins) for the fuel temperature obtained with T4/SCF and SSS2/SCF for the TMI-1 3x3 mini-core during transient scenario D, at three different time steps: $t = 0.1$ s, $t = 1.3$ s and $t = 2.1$ s. Difference values are given for the average across all axial slices, and for the slices 3 and 10. The highest values can be observed for $t = 1.3$ s, where there are large fluctuations on the neutron population size.

Résumé en français

Introduction

L'énergie produite dans un réacteur nucléaire provient des interactions entre les neutrons et les noyaux lourds présents dans le combustible. L'un des principaux enjeux pour l'étude de l'évolution d'un réacteur consiste à modéliser la propagation de ces neutrons, décrite par l'équation de Boltzmann, tout en prenant en compte les phénomènes multiphysiques, c'est-à-dire les interactions entre la neutronique, la thermohydraulique et la thermomécanique. En effet, il existe un fort couplage entre ces différentes physiques : l'effet modérateur et l'effet Doppler dans le combustible notamment ont un effet stabilisant sur la puissance neutronique. L'étude de ce système multiphysique est particulièrement complexe en raison du grand nombre de variables, ainsi que des dimensions du système.

Pour résoudre l'équation du transport, il existe deux types de méthodes. Le premier regroupe les méthodes déterministes, qui résolvent l'équation numériquement en discrétisant l'espace des phases. Le second regroupe les méthodes stochastiques, aussi appelées « méthodes Monte-Carlo », qui reposent sur le tirage aléatoire d'un grand nombre de trajectoires neutroniques, sur lesquelles une moyenne et une incertitude sont déterminées. Les méthodes Monte-Carlo offrent une résolution exacte du système, au prix de cette incertitude statistique sur le résultat. Pour la réduire, la solution la plus naturelle consiste à augmenter la statistique, c'est-à-dire le nombre de trajectoires simulées. Ce n'est pas toujours une solution viable car le temps de calcul augmente plus vite que la convergence ne s'améliore.

C'est pourquoi en général les méthodes déterministes sont utilisées pour la conception et l'opération des réacteurs commerciaux. Le Monte-Carlo est utilisé ponctuellement comme outil de référence pour les calculs stationnaires, c'est-à-dire sans dépendance temporelle. En revanche, les calculs non stationnaires, comme les transitoires accidentels par exemple, étaient traités jusqu'à récemment par les codes déterministes uniquement. Pour les configurations non stationnaires, le développement de méthodes Monte-Carlo qui prennent en compte la dépendance en temps du système neutronique, mais aussi le couplage avec les autres physiques (méthodes « dynamiques »), a pour but de servir de référence aux calculs déterministes. Avec l'augmentation des moyens de calcul, de telles simulations deviennent de plus en plus faisables ; cependant, les calculs restent très longs et les couplages spécifiques.

C'est dans ce contexte que j'ai effectué mon travail de thèse. Il a consisté à mettre en place une chaîne de calcul pour la simulation couplée neutronique Monte-Carlo, avec le code TRIPOLI-4, en conditions non stationnaires et avec prise en compte des contre-réactions thermohydrauliques dans le but de servir de référence aux calculs de transitoires liés à des insertions de réactivité sur des réacteurs de type REP (Réacteurs à Eau Pressurisée).

Ce travail est organisé en trois parties. Dans un premier temps, nous avons considéré les

méthodes « cinétiques » dans TRIPOLI-4, c'est-à-dire avec prise en compte du temps mais sans prise en compte des contre-réactions, en incluant une évaluation des méthodes existantes ainsi que le développement de nouvelles méthodes. Ensuite, nous avons développé un schéma de couplage entre TRIPOLI-4 et le code de thermohydraulique sous-canal SUBCHANFLOW. Enfin, nous avons réalisé une analyse préliminaire de la propagation des incertitudes au sein du calcul couplé sur un modèle simplifié. En effet, les fluctuations statistiques sont inhérentes à notre schéma de par la nature stochastique de TRIPOLI-4. De plus, les équations de la thermohydraulique étant non-linéaires, la propagation des incertitudes au long du calcul doit être étudiée afin de caractériser la convergence du résultat.

Méthodes cinétiques pour la simulation neutronique Monte-Carlo avec TRIPOLI-4

Le but des méthodes Monte-Carlo cinétiques est de résoudre l'équation de Boltzmann dépendante du temps, couplée avec l'équation d'évolution des précurseurs. Le traitement explicite des précurseurs nécessite d'utiliser des méthodes de contrôle de population et de réduction de variance. Nous avons détaillé et évalué les méthodes existantes dans TRIPOLI-4 : la *roulette russe* et le *splitting* et le *combing* pour le contrôle de la population, la décroissance forcée et les collisions non branchantes pour la réduction de variance. Pour leur évaluation, nous avons considéré un assemblage basé sur les spécifications du réacteur TMI-1 (Three Mile Island).

Nous avons également détaillé une nouvelle méthode de réduction de variance, basée sur un schéma d'échantillonnage préférentiel appliqué sur les populations des neutrons et des précurseurs. L'efficacité de cette méthode a été démontrée sur l'assemblage de TMI-1, et se montre particulièrement intéressante lorsque les pas de temps de la simulation sont petits : dans ce cas, les neutrons retardés sont tués par la roulette russe malgré l'application de l'algorithme de décroissance forcée des précurseurs. Nous avons examiné le choix optimal du rapport d'importance entre neutrons et précurseurs selon la taille du pas de temps de la grille de simulation : plus ce dernier est petit, plus le rapport d'importance doit l'être également. Le facteur de mérite est peu dépendant du facteur d'importance autour de la valeur optimale. Nous avons enfin détaillé une nouvelle contribution à l'extension de TRIPOLI-4 pour les simulations cinétiques : la géométrie peut désormais varier au cours du temps, et il est ainsi possible de simuler tous types de transitoires, comme par exemple des extractions de barres de contrôle.

Nous avons testé les méthodes cinétiques sur deux systèmes réalistes. D'abord, le réacteur expérimental SPERT III E-core a été simulé dans différentes configurations : critique, extraction des barres de contrôle et chute des barres. Nous avons montré l'importance de la prise en compte des précurseurs, même lors d'une excursion sur-critique prompte. Nous avons aussi montré l'intérêt d'adapter le facteur d'importance au cours du temps, afin d'observer l'évolution prompte et retardée lors d'une chute des barres de contrôle. Quatre scénarios ont été simulés dans le cadre d'un travail collaboratif de vérification sur un mini-cœur basé sur le réacteur TMI-1. Les résultats issus de TRIPOLI-4 ont été comparés avec ceux issus du code Monte-Carlo Serpent 2, et un bon accord a été trouvé entre les deux codes. Deux de ces quatre scénarios ont été simulés plus tard avec prise en compte des contre-réactions thermohydrauliques.

Dans une simulation cinétique, le temps d'observation est généralement plus long qu'un temps de génération, ce qui induit des phénomènes de corrélations entre les différents pas de temps de la grille d'encaissement. D'abord, nous avons étudié la décroissance des corrélations avec le temps. Nous avons observé que la constante de décroissance est environ égale à la durée

de vie moyenne des chaînes de fission. Ensuite, nous avons étudié la dépendance de l'incertitude relative à la discrétisation en temps du maillage des scores, suite à l'observation que cette incertitude ne semblait pas être proportionnelle au volume de la cellule du maillage. Plus précisément, nous avons remarqué que pour des temps de simulation longs devant la durée de vie de la chaîne de fission, l'incertitude sur le score global était du même ordre que l'incertitude sur le score à l'échelle d'une cellule de combustible. En fait, les neutrons issus d'une même chaîne de fission sont corrélés, et lorsqu'ils explorent le système, ils déposent des contributions corrélées dans différentes cellules. Ainsi, la variance sur le score global augmente à cause de ces corrélations.

Développement d'un couplage entre la TRIPOLI-4 et la thermohydraulique

Une fois les méthodes cinétiques de TRIPOLI-4 extensivement étudiées et vérifiées, nous avons développé une interface multiphysique pour TRIPOLI-4, et plus spécifiquement un schéma de couplage entre TRIPOLI-4 et le code de thermohydraulique sous-canal SUBCHANFLOW. Tout d'abord, nous avons séparé la lecture du jeu de données TRIPOLI-4 de l'initialisation des différentes classes, grâce au développement d'une interface de programmation (Application Programming interface, API). Ensuite, un programme externe, appelé superviseur, a été mis en place pour piloter le calcul TRIPOLI-4 en même temps qu'un autre calcul (thermohydraulique). Le superviseur fait appel aux outils de la plateforme SALOME : l'API ICoCo pour l'interfaçage entre les codes et la librairie MEDCoupling pour l'échange de données entre les codes. TRIPOLI-4 calcule la distribution de puissance, qui est transférée à SUBCHANFLOW. Le solveur thermohydraulique calcule les nouvelles propriétés du combustible (températures) et du modérateur (températures et densités), qui sont transférées à TRIPOLI-4.

Le schéma de couplage mis en place permet de réaliser deux types de calcul : des calculs critiques couplés et des transitoires couplés. Les calculs critiques couplés sont divisés en itérations externes entre les appels à la thermohydraulique (pour que les champs de température et de densités convergent) et itérations internes entre les appels à TRIPOLI-4 (pour que les sources de fission convergent). Un schéma de relaxation est utilisé dans les calculs critiques couplés afin de réduire les oscillations entre itérations successives. Les transitoires commencent avec une source préalablement calculée avec un calcul critique couplé, et sont intégrés en temps avec un schéma d'Euler explicite. La possibilité de sauvegarder la source (les champs thermohydrauliques et les sources de fission) à la fin d'un calcul de criticité permet de la réutiliser pour différents transitoires.

Le superviseur peut être lancé en exécution parallèle, avec une organisation proche de celle du schéma parallèle de TRIPOLI-4. L'un des processeurs, le « moniteur », est chargé de diriger les autres processeurs, qui réalisent le calcul TRIPOLI-4 ; il exécute aussi le superviseur ainsi que SUBCHANFLOW. Un autre processeur, le « scoreur » est chargé de collecter les scores. Les autres, appelés « simulateurs » sont en charge de la simulation TRIPOLI-4. Afin de réduire les fluctuations statistiques données en entrée au solveur thermohydraulique qui est non linéaire, la puissance neutronique est moyennée sur les différents processeurs et SUBCHANFLOW est exécuté une fois. L'interdépendance des simulateurs via les champs de température et de densité complique l'estimation de l'incertitude sur le résultat du couplage. Diviser la simulation en répliques indépendantes permettrait d'obtenir des résultats indépendants, mais les fluctuations statistiques transférées au solveur thermohydraulique serait alors amplifiées.

Vérification du couplage entre TRIPOLI-4 et SUBCHANFLOW

Une fois le couplage mis en place, nous avons vérifié les capacités de TRIPOLI-4 à faire des calculs critiques avec contre-réactions thermohydrauliques, au travers d'un travail de collaboration entre les équipes de développement de Serpent 2 et de SUBCHANFLOW, dans le cadre du projet européen McSAFE. La configuration retenue est l'assemblage de TMI-1, avec une description crayon par crayon. Les distributions de puissance, température et densité obtenues sont en très bon accord entre les schémas de couplage TRIPOLI-4/SUBCHANFLOW et Serpent 2/SUBCHANFLOW.

Des simulations de transitoires avec contre-réactions thermohydrauliques ont été ensuite présentées. Le mini-cœur basé sur le réacteur TMI-1 a d'abord été simulé dans son état stationnaire afin de vérifier sa stabilité. Ensuite, des scénarios d'insertion de réactivité ont été présentés. Deux scénarios d'extraction de barres ont été simulés : 30 cm and 40 cm. Les résultats du couplage entre TRIPOLI-4 et SUBCHANFLOW ont été comparés avec ceux issus du schéma de couplage entre Serpent 2 et SUBCHANFLOW, et un bon accord a été trouvé.

L'extraction des barres de 40 cm rend le système sur-critique prompt, et les fortes fluctuations sur la taille de la population induisent des fluctuations sur le temps de calculs des différents simulateurs. Le calcul est très pénalisé par quelques processeurs qui échantillonnent des événements rares. Cette simulation montre qu'il faut renforcer le schéma de couplage afin de le rendre moins sensible aux fluctuations sur la population. Une autre solution pourrait être de moyenner la puissance neutronique sur moins de simulateurs afin de réduire les temps d'attente moyen entre les appels à SUBCHANFLOW, mais alors les fluctuations statistiques transmises au solveur thermo-hydraulique seraient amplifiées.

Analyse préliminaire de la stabilité du schéma de couplage

Afin de réaliser une première caractérisation de la stabilité de notre schéma de couplage, nous avons sélectionné un modèle simplifié décrivant le couplage entre la neutronique et la thermohydraulique dans un REB (Réacteur à Eau Bouillante). Le modèle est constitué de cinq équations différentielles déterministes pour l'évolution temporelle de la concentration neutronique (cette équation est non linéaire), la concentration des précurseurs, la température du combustible, la réactivité de vide et la réactivité totale.

Nous avons fait plusieurs hypothèses pour simplifier l'étude du modèle. Tout d'abord, nous avons modifié deux paramètres afin d'adapter le modèle à la description d'un REP. Afin d'imiter le comportement de TRIPOLI-4, nous avons ajouté un terme de bruit sur l'équation des neutrons, sous la forme d'un terme brownien, même si cette modélisation ne constitue pas une représentation fidèle des fluctuations issues du code Monte Carlo, puisque les incréments d'un mouvement brownien sont décorrélés des incréments précédents. Nous avons présenté une analyse du système stochastique, en commençant par l'étude de l'impact du brownien sur les concentrations des neutrons et des précurseurs, en l'absence de contre-réactions. Nous avons identifié trois régimes : celui des temps petits devant la durée de vie d'une chaîne de fission, celui des temps longs devant le temps de vie des précurseurs, et celui des temps intermédiaires. Les conclusions issues de notre étude analytique ont été vérifiées à l'aide de simulation numériques.

L'analyse du système complet est compliquée par le terme non linéaire dans l'équation décrivant l'évolution des neutrons. Afin de simplifier le système, nous avons négligé la réactivité de vide. Ainsi, nous avons réduit le système initial à un système à trois équations. Même si l'ana-

lyse du système résultant est plus simple, une étude analytique ne semble pas réalisable en l'état, à cause du terme non linéaire. C'est pourquoi nous avons réalisé des simulations numériques du système pour caractériser son comportement.

Conclusions

Finalement, le schéma de couplage entre TRIPOLI-4 et la thermohydraulique est opérationnel et permet de réaliser des simulations stationnaires et non stationnaires avec prise en compte des contre-réactions. Les deux types de simulations ont été vérifiés par comparaison avec un schéma de calcul existant. Une analyse préliminaire de la stabilité du couplage a été réalisée.

Bibliography

- Akcasu, A., Williams, M. M. R., 2004. An analytical study of particle transport in spatially random media in one dimension: Mean and variance calculations. *Nuclear Science and Engineering* 148.
- Akcasu, Z., Lellouche, G., Shotkin, L., 1971. *Mathematical methods in nuclear reactor dynamics*. Academic Press, USA.
- Aoki, S., Suemura, T., Ogawa, J., Takeda, T., 2009. Analysis of the SPERT III core using ANCK code with the chord weighting method. *Journal of Nuclear Science and Technology* 46, 239 – 251.
- Avramova, M. N., Salko, R. K., 2016. CTF theory manual. Tech. Rep. ORNL/TM-2016/430, Oak Ridge National Laboratory, USA.
- Bell, G. I., Glasstone, S., 1970. *Nuclear reactor theory*. Van Nostrand Reinhold, New York, USA.
- Bergeaud, V., Lefebvre, V., 2010. Proceedings of the SNA + MC 2010 conference. Tokyo, Japan.
- Bernnat, W., Buck, M., Mattes, M., Zwermann, W., Pasichnyk, I., Velkov, K., 2012. Proceedings of the PHYSOR 2012 conference. Knoxville, Tennessee, USA.
- Booth, T., 1996. A weight (charge) conserving importance-weighted comb for Monte Carlo. In: *Proceedings of Radiation Protection and Shielding Division Topical Meeting*. Falmouth, Massachusetts, USA.
- Briesmeister, J. F., 2000. MCNP - A General Monte Carlo N-Particle Transport Code, Version 4C. Tech. Rep. LA-13709, Los Alamos National Laboratory, USA.
- Brun, E., Damian, F., Diop, C., Dumonteil, E., Hugot, F., Jouanne, C., Lee, Y., Malvagi, F., Mazzolo, A., Petit, O., Trama, J. C., Visonneau, T., Zoia, A., 2015. TRIPOLI-4[®], CEA, EDF and AREVA reference Monte Carlo code. *Annals of Nuclear Energy* 82, 151 – 160.
- Cao, L., Gerlach, A., Xu, Y., Downar, T., Lee, J. C., 2015. Neutronics modeling of the SPERT III E-Core critical experiments with MPACT and KENO. *Annals of Nuclear Energy* 80, 207 – 218.
- Cardoni, J. N., 2011. Proceedings of the M&C 2011 conference. Rio de Janeiro, Bresil.
- CD-adapco, 2005. STAR-CD Version 3.26. Melville, New York, USA.
- CD-adapco, 2009. User Guide: STAR-CCM+ Version 2.10.017.
- Chadwick, M. B., et al., 2006. ENDF/B-VII.0: Next generation evaluated nuclear data library for nuclear science and technology. *Nuclear Data Sheets* 107, 2931 – 3060.

- Cohen, E. R., 1958. Proceedings of the Second United Nations Conference on the Peaceful Uses of Atomic Energy. Geneva, Switzerland.
- Cullen, D. E., 1979. Program SIGMA1 (version 79-1): Doppler broaden evaluated cross sections in the evaluated nuclear data file/version B (ENDF/B) format. Tech. Rep. UCRL-50400, Lawrence Livermore National Laboratory, USA.
- Cullen, D. E., Clouse, C. J., Richard, P., Little, R. C., 2003. Static and dynamic criticality: are they different? Tech. Rep. UCRL-TR-201506, Lawrence Livermore National Laboratory, USA.
- D. Basile, R. Chierici, M. B. E. S., Brega, E., 1999. COBRA-EN, an updated version of the COBRA-3C/MIT code for thermal-hydraulic transient analysis of light water reactor fuel assemblies and cores. Tech. Rep. 1010/1, ENEL-CRTN Compartimento di Milano, Italy.
- Daeubler, M., Ivanov, A., Sjenitzer, B. L., Sanchez, V., Stieglitz, R., Macian-Juan, R., 2015. High-fidelity coupled Monte Carlo neutron transport and thermal-hydraulic simulations using Serpent 2/SUBCHANFLOW. *Annals of Nuclear Energy* 83, 352 – 375.
- Daeubler, M., Jiménez, J., Sancheza, V., 2014. Proceedings of the PHYSOR 2014 conference. Kyoto, Japan.
- D’Auria, F., Soloviev, S., Malofeev, V., Ivanov, K., Parisi, C., 2008. The three-dimensional neutron kinetics coupled with thermal-hydraulics in RBMK accident analysis. *Nuclear Engineering and Design* 238 (4), 1002 – 1025.
- D’Auria, F., et al., 2004. Neutronics and thermal-hydraulic coupling in LWR technology. Tech. Rep. 5434, OECD Nuclear Energy Agency, Paris, France.
- De Mulatier, C., Dumonteil, E., Rosso, A., Zoia, A., 2015. The critical catastrophe revisited. *Journal of Statistical Mechanics* 2015 (8), P08021.
- Deville, E., Perdu, F., 2012. Documentation of the Interface for Code Coupling : ICoCo. Tech. Rep. DEN/DANS/DM2S/STMF/LMES/RT/12-029/A, CEA, France.
- Downar, T. J., Barber, D., Matthew Miller, R., Lee, C.-H., Kozlowski, T., Lee, D., Xu, Y., Gan, J., Joo, H., Jin, J., Cho, Y., Lee, K., P Ulses, A., 2002. Proceedings of the PHYSOR 2002 conference. Seoul, Korea.
- Duderstadt, J., Hamilton, L., 1976. *Nuclear Reactor Analysis*. John Wiley and sons, New York, USA.
- Duderstadt, J., Martin, W., 1979. *Transport theory*. John Wiley and sons, New York, USA.
- Dufek, J., Gudowski, W., 2006. Stochastic approximation for Monte Carlo calculation of steady-state conditions in thermal reactors. *Nuclear Science and Engineering* 152, 274 – 283.
- Dugone, J., 1965. SPERT-III reactor facility: E-core revision. Tech. Rep. IDO-17036, U.S. Atomic Energy Commission, Idaho Operations Office and Phillips Petroleum Company, USA.
- Dulla, S., Mund, E. H., P. Ravetto, P., 2008. The quasi-static method revisited. *Progress in Nuclear Energy* 50, 908 – 920.
- Dumonteil, E., Malvagi, F., Zoia, A., Mazzolo, A., Artusio, D., Dieudonné, C., Mulatier, C. D., 2014. Particle clustering in Monte Carlo criticality simulations. *Annals of Nuclear Energy* 63, 612 – 618.

- Faucher, M., Mancusi, D., Zoia, A., 2018. New kinetic simulation capabilities for TRIPOLI-4[®]: Methods and applications. *Annals of Nuclear Energy* 120, 74 – 88.
- Faucher, M., Mancusi, D., Zoia, A., 2019a. Proceedings of the M&C 2019 conference. Portland, Oregon, USA, submitted.
- Faucher, M., Mancusi, D., Zoia, A., Ferraro, D., Garcia, M., Imke, U., Leppänen, J., Valtavirta, V., 2019b. Proceedings of the ICAPP 2019 conference. Juan-les-Pins, France.
- Ferraro, D., Faucher, M., Mancusi, D., Zoia, A., Valtavirta, V., Leppänen, J., Sanchez, V., 2019a. Proceedings of the M&C 2019 conference. Portland, Oregon, USA, submitted.
- Ferraro, D., Garcia, M., Valtavirta, V., Imke, U., Tuominen, R., Leppänen, J., Sanchez, V., 2019b. Serpent/SUBCHANFLOW pin-by-pin coupled transient calculations for a PWR mini-core. *Annals of Nuclear Energy* In press.
- Flyvbjerg, H., Petersen, H. G., 1989. Error estimates on averages of correlated data. *The Journal of Chemical Physics* 91 (1), 461–466.
- Gill, D. F., Griesheimer, D. P., Aumiller, D. L., 2017. Numerical methods in coupled Monte Carlo and thermal-hydraulic calculations. *Nuclear Science and Engineering* 185 (1), 194–205.
- Gomez-Torres, A. M., Sanchez, V., Ivanov, K., Macian-Juan, R., 2012. DYN SUB: A high fidelity coupled code system for the evaluation of local safety parameters – Part I: Development, implementation and verification. *Annals of Nuclear Energy* 48, 108 – 122.
- Grandi, G., 2014. Proceedings of the PHYSOR 2014 conference. Kyoto, Japan.
- Grandi, G., Moberg, L., 2012. Proceedings of the PHYSOR 2012 conference. Vol. 4. Knoxville, Tennessee, USA.
- Grundmann, U., Rohde, U., Mittag, S., Kliem, S., 2005. DYN3D version 3.2 - code for calculation of transients in Light Water Reactors (LWR) with hexagonal or quadratic fuel elements - description of models and methods. Tech. Rep. FZR-434, Wissenschaftlich-Technische Berichte / Forschungszentrum Rossendorf, Germany.
- Heffner, R., Wilson, T., 1961. SPERT III reactor facility. Tech. Rep. IDO-16721, U.S. Atomic Energy Commission, Idaho Operations Office and Phillips Petroleum Company, USA.
- Henry, A., 1964. The application of inhour modes to the description of nonseparable reactor transients. *Nuclear Science and Engineering* 20, 338 – 351.
- Henry, R., Tiselj, I., Snoj, L., 2017. CFD/Monte-Carlo neutron transport coupling scheme, application to TRIGA reactor. *Annals of Nuclear Energy* 110, 36 – 47.
- Hoogenboom, J., Sjenitzer, B., 2014. Proceedings of the SNA + MC 2013 conference.
- Houchmandzadeh, B., Dumonteil, E., Mazzolo, A., Zoia, A., 2015. Neutron fluctuations: The importance of being delayed. *Physical Review E* 92.
- Houghtaling, J., Norbert, J., Haire, J., 1965. Addendum to the SPERT III hazards summary report: Low-enrichment oxide core. Tech. Rep. IDO-17003, U.S. Atomic Energy Commission, Idaho Operations Office and Phillips Petroleum Company, USA.
- IAEA, 2003. Safety margins of operating reactors - analysis of uncertainties and implications for decision making. Tech. Rep. IAEA-TECDOC-1332, IAEA, Vienna, Austria.

- IAEA, 2015. Research reactor benchmarking database: Facility specification and experimental data. Tech. Rep. 480, IAEA, Vienna, Austria.
- Ikeda, H., Takeda, T., 2001. Development and verification of an efficient spatial neutron kinetics method for reactivity-initiated event analyses. *Journal of Nuclear Science and Technology* 38, 492 – 502.
- Imke, U., Sanchez, V., 2012. Validation of the subchannel code SUBCHANFLOW using the NUPEC PWR tests (PSBT). *Science and Technology of Nuclear Installations* 2012.
- Ivanov, A., Sanchez, V., Stieglitz, R., Ivanov, K., 2013a. High fidelity simulation of conventional and innovative LWR with the coupled Monte-Carlo thermal-hydraulic system MCNP-SUBCHANFLOW. *Nuclear Engineering and Design* 262, 264 – 275.
- Ivanov, K., Avramova, M., Kamerow, S., Kodeli, I., Sartori, E., Ivanov, E., Cabellos, O., 2013b. Benchmark for uncertainty analysis in modeling (UAM) for design, operation and safety analysis of LWRs. volume i: Specification and support data for the neutronics cases (phase i), version 2.1 (final specifications). Tech. Rep. NEA/NSC/DOC(2013)7, OECD Nuclear Energy Agency, Paris, France.
- Kaper, H. G., 1967. The initial-value transport problem for monoenergetic neutrons in an infinite slab with delayed neutron production. *Journal of Mathematical Analysis and Applications* 19, 207 – 230.
- Kaplan, E., 1958. Monte Carlo methods for equilibrium solutions in neutron multiplication. University of California Radiation Laboratory, USA.
- Keepin, G., 1965. *Physics of Nuclear Kinetics*. Addison-Wesley, Reading, UK.
- Knebel, M., Mercatali, L., Sanchez, V., Stieglitz, R., Macian-Juan, R., 2016. Validation of the Serpent 2-DYNSUB code sequence using the Special Power Excursion Reactor Test III (SPERT III). *Annals of Nuclear Energy* 91, 79 – 91.
- Kosaka, I., Kawata, N., Wakabayashi, T., 1988. Verification of the EUREKA-ATR code analysis of the SPERT-III E-core experiment. Tech. Rep. PNC-TN9410 88-057, Oarai Engineering Center, Japan.
- Kozłowski, T., Downar, T. J., 2007. OECD/NEA and U.S. NRC PWR MOX/UO₂ Core Transient Benchmark. Tech. Rep. NEA/NSC/DOC2006(20), OECD Nuclear Energy Agency, Paris, France.
- Lapeyre, B., Pardoux, E., Sentis, R., 1997. *Méthodes de Monte-Carlo pour les équations de transport et de diffusion*. Mathématiques et Applications. Springer Berlin Heidelberg.
- Larsen, E. W., 2011. Proceedings of the Frédéric Joliot / Otto Hahn summer school. Karlsruhe, Germany.
- Larsen, E. W., Zweifel, P. F., 1974. On the spectrum of the linear transport operator. *Journal of Mathematical Physics* 15, 1987 – 1997.
- Laureau, A., Aufiero, M., Rubiolo, P., Merle, E., Heuer, D., 2015. Transient Fission Matrix: Kinetic calculation and kinetic parameters β_{eff} and Λ_{eff} calculation. *Annals of Nuclear Energy* 85, 1035 – 1044.

- Laureau, A., Buiron, L., Fontaine, B., 2017. Local correlated sampling Monte Carlo calculations in the TFM neutronics approach for spatial and point kinetics applications. *EPJ Nuclear Sciences & Technologies* 3, 16.
- Légrády, D., Hoogenboom, J. E., 2008. Proceedings of the PHYSOR 2008 conference. Inter-laken, Switzerland.
- Leppänen, J., 2010. Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code. *Annals of Nuclear Energy* 37, 715 – 722.
- Leppänen, J., 2013. Proceedings of the M&C 2013 conference. Sun Valley, Idaho, USA.
- Leppänen, J., Pusa, M., Viitanen, T., Valtavirta, V., Kaltiaisenaho, T., 2015. The Serpent Monte Carlo code: Status, development and applications in 2013. *Annals of Nuclear Energy* 82, 142 – 150.
- Leppänen, J., Viitanen, T., Valtavirta, V., 2012. Multi-physics coupling scheme in the Serpent 2 Monte Carlo code. *Transactions of the American Nuclear Society* 107, 1165 – 1168.
- Lerchl, G., Austregesilo, H., 1998. ATHLET Mod 1.2 Cycle A, User's Manual, Rev. 1.
- Levinsky, A., Valtavirta, V., Adams, F., Anghel, V., 2019. Modeling of the SPERT transients using Serpent 2 with time-dependent capabilities. *Annals of Nuclear Energy* 125, 80 – 98.
- Lux, I., Koblinger, L., 1991. Monte Carlo particle transport methods: neutron and photon calculations. CRC Press, Boston, USA.
- Mancusi, D., Zoia, A., 2018. Chaos in eigenvalue search methods. *Annals of Nuclear Energy* 112, 354 – 363.
- March-Leuba, J., 1986. A reduced-order model of boiling water reactor linear dynamics. *Nuclear Technology* 75, 15 – 22.
- McCardell, R. K., Herborn, D. I., Houghtaling, J. E., 1969. Reactivity accident test results and analyses for the SPERT III E-core - A small oxide-fueled, pressurized-water reactor. Tech. Rep. IDO-17281, U.S. Atomic Energy Commission, Idaho Operations Office and Phillips Petroleum Company, USA.
- Molnar, B., Tolnai, G., Legrady, D., 2019. A GPU-based direct Monte Carlo simulation of time dependence in nuclear reactors. *Annals of Nuclear Energy* 132, 46 – 63.
- Mori, M., Maschek, W., Laurien, E., Morita, K., 2003. Proceedings of the GLOBAL 2003 conference. New Orleans, Louisiana, USA.
- Mylonakis, A. G., Varvayanni, M., Grigoriadis, D., Catsaros, N., 2017. Developing and investigating a pure Monte-Carlo module for transient neutron transport analysis. *Annals of Nuclear Energy* 104, 103 – 112.
- OECD Nuclear Energy Agency, Paris, F., 1995. International Handbook of Evaluated Criticality Safety Benchmark Experiments. NEA/NSC/DOC(95)03.
- Olson, A. P., 2013a. Consolidator's Report for the SPERT-III Benchmark. Tech. Rep. ANL/GTRI/TM-13/9, Argonne National Laboratory, USA.
- Olson, A. P., 2013b. Neutronics calculations for SPERT-III, E-core. Tech. Rep. ANL/GTRI/TM-13/10, Argonne National Laboratory, USA.

- Olson, A. P., 2013c. SPERT III E-CORE: Facility specifications. Tech. Rep. IDO-17036, Argonne National Laboratory, USA.
- OpenFOAM Foundation, 2017.
URL <https://openfoam.org/>
- Revuz, D., Yor, M., 2004. Continuous Martingales and Brownian Motion. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, Germany.
- Rosenthal, J. S., 1999. Parallel computing and Monte Carlo algorithms. *Far East Journal of Theoretical Statistics* 4, 207 – 236.
- Rowe, D. S., 1973. COBRA IIIC: digital computer program for steady state and transient thermal-hydraulic analysis of rod bundle nuclear fuel elements. Tech. Rep. BNWL-1695, Pacific Northwest National Laboratory, USA.
- SALOME, 2019.
URL <https://www.salome-platform.org/>
- Sanchez, V., Al-Hamry, A., 2009. Proceedings of the M&C 2009 conference. New York, USA.
- Santamarina, A., Bernard, D., Rugama, Y., 2009. The JEFF-3.1.1 Nuclear Data Library: JEFF Report 22, Validation Results from JEF-2.2 to JEFF-3.1.1. Tech. Rep. ISBN 978-92-64-99074-6, OECD Nuclear Energy Agency, Paris, France.
- Seker, V., Thomas, J. W., Downar, T. J., 2007. Reactor physics simulations with coupled Monte Carlo calculation and computational fluid dynamics. Vol. 1. pp. 345 – 349.
- Sjenitzer, B. L., 2013. The dynamic Monte Carlo method for transient analysis of nuclear reactors. Ph.D. thesis, Delft University of Technology, Netherlands.
- Sjenitzer, B. L., Hoogenboom, J. E., 2011a. A Monte Carlo method for calculation of the dynamic behaviour of nuclear reactors. *Progress in Nuclear Science and Technology* 2, 716 – 721.
- Sjenitzer, B. L., Hoogenboom, J. E., 2011b. Variance reduction for fixed-source Monte Carlo calculations in multiplying systems by improving chain-length statistics. *Annals of Nuclear Energy* 38, 2195 – 2203.
- Sjenitzer, B. L., Hoogenboom, J. E., 2013. Dynamic Monte Carlo method for nuclear reactor kinetics calculations. *Nuclear Science and Engineering* 175, 94 – 107.
- Sjenitzer, B. L., Hoogenboom, J. E., Escalante, J. J., Sanchez, V., 2015. Coupling of dynamic Monte Carlo with thermal-hydraulic feedback. *Annals of Nuclear Energy* 76, 27 – 29.
- Smith, K. S., 1979. An analytic nodal method for solving the two-group, multidimensional, static and transient neutron diffusion equations. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.
- Toumi, I., Bergeron, A., Gallo, D., Royer, E., Caruge, D., 2000. FLICA-4: a three-dimensional two-phase flow computer code with advanced numerical methods for nuclear applications. *Nuclear Engineering and Design* 200, 139 – 155.
- Trahan, T. J., 2018. A quasi-static Monte Carlo algorithm for the simulation of sub-prompt critical transients. *Annals of Nuclear Energy* 127, 257 – 267.

- Tuominen, R., Valtavirta, V., Leppänen, J., 2019. New energy deposition treatment in the Serpent 2 Monte Carlo transport code. *Annals of Nuclear Energy* 129, 224 – 232.
- Tuominen, R., Valtavirta, V., Peltola, J., Leppänen, J., 2016. Proceedings of the PHYSOR 2016 conference. Sun Valley, Idaho, USA.
- Valtavirta, V., Hessian, M., Leppänen, J., 2016. Proceedings of the PHYSOR 2016 conference. Sun Valley, Idaho, USA.
- Viitanen, T., Leppänen, J., 2012. Explicit treatment of thermal motion in continuous-energy Monte Carlo tracking routines. *Nuclear science and engineering: the journal of the American Nuclear Society* 171, 165 – 173.
- Viitanen, T., Leppänen, J., 2014. Target motion sampling temperature treatment technique with elevated basis cross-section temperatures. *Nuclear Science and Engineering* 177, 77 – 89.
- Wang, L., He, P., Sun, G., Hao, L., Song, J., 2018. Proceedings of the PHYSOR 2018 conference. Cancún, Mexico.
- Wang, R. C., Xu, Y., Downar, T., Hudson, N., 2013. Validation of the U.S. NRC coupled code system TRITON/TRACE/PARCS with the Special Power Excursion Reactor Test III (SPERT III). *Nuclear Technology* 183, 504 – 514.
- Weinberg, A., 1952. Current status of nuclear reactor theory. *American Journal of Physics* 20, 401 – 412.
- Wheeler, C., Stewart, C., Cena, R., Rowe, D., Sutey, A., 1976. COBRA-IV-I: an interim version of COBRA for thermal-hydraulic analysis of rod bundle nuclear fuel elements and cores. Tech. Rep. BNWL-196, Pacific Northwest National Laboratory, USA.
- Williams, M. M. R., 1971. *Mathematical methods in particle transport theory*. Wiley-Interscience, New York, USA.
- Woodcock, E. R., Murphy, T., Hemmings, P. J., Longworth, T. C., 1965. Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry. Tech. Rep. ANL-7050, Argonne National Laboratory, USA.
- X-5 Monte Carlo Team, 2003. MCNP - a general Monte Carlo N-particle transport code, Version 5, Volume I: Overview and theory. Tech. Rep. LA-UR-03-1987, Los Alamos National Laboratory, USA.
- Yamaji, H., Takemoto, Y., Kirimura, K., et al., 2014. Proceedings of the PHYSOR 2014 conference. Kyoto, Japan.
- Yamano, H., Fujita, S., Tobita, Y., Kamiyama, K., Kondo, S., Morita, K., Fischer, E. A., Brear, D. J., Shirakawa, N., Cao, X., Sugaya, M., Mizuno, M., S. Hosono, T. K., Maschek, W., Kiefhaber, E., Buckel, G., Rineiski, A., Flad, M., Suzuki, T., Coste, P., Pigny, S., Louvet, J., Cadiou, T., 2003. SIMMER-III: A computer program for LMFR core disruptive accident analysis. Tech. Rep. JNC-TN-9400-2001-002, Japan Nuclear Cycle Development Institute, Japan.
- Yesilyurt, G., Martin, W., B Brown, F., 2012. On-the-Fly Doppler Broadening for Monte Carlo Codes. *Nuclear Science and Engineering* 171, 239 – 257.
- Zhu, A., 2015. Proceedings of the M&C 2015 conference. Nashville, Tennessee, USA.

- Zoia, A., Brun, E., 2016. Reactor physics analysis of the SPERT III E-Core with TRIPOLI-4[®]. *Annals of Nuclear Energy* 90, 71 – 82.
- Zoia, A., Brun, E., Damian, F., Malvagi, F., 2015. Monte Carlo methods for reactor period calculations. *Annals of Nuclear Energy* 75, 627 – 634.
- Zoia, A., Brun, E., Malvagi, F., 2014a. Alpha eigenvalue calculations with TRIPOLI-4[®]. *Annals of Nuclear Energy* 63, 276 – 284.
- Zoia, A., Dumonteil, E., Mazzolo, A., De Mulatier, C., Rosso, A., 2014b. Clustering of branching Brownian motions in confined geometries. *Physical Review E* 90.

Titre : Couplage entre la simulation neutronique Monte-Carlo et la thermohydraulique pour les transitoires liés à des insertions de réactivité

Mots clés : Neutronique, simulation Monte-Carlo, thermohydraulique, multiphysique, transitoires, dynamique

Résumé : Dans le contexte de la physique des réacteurs, l'analyse du comportement non stationnaire de la population neutronique avec contre-réactions dans le combustible et dans le modérateur se rend indispensable afin de caractériser les transitoires opérationnels et accidentels dans les systèmes nucléaires et d'en améliorer par conséquent la sûreté. Pour ces configurations non stationnaires, le développement de méthodes Monte-Carlo qui prennent en compte la dépendance en temps du système neutronique, mais aussi le couplage avec les autres physiques, comme la thermohydraulique et la thermomécanique, a pour but de servir de référence aux calculs déterministes.

Ce travail de thèse a consisté à mettre en place une chaîne de calcul pour la simulation couplée neutronique Monte-Carlo, avec le code TRIPOLI-4, en conditions non stationnaires et avec prise en compte

des contre-réactions thermohydrauliques.

Nous avons d'abord considéré les méthodes cinétiques dans TRIPOLI-4, c'est-à-dire avec prise en compte du temps mais sans prise en compte des contre-réactions, en incluant une évaluation des méthodes existantes ainsi que le développement de nouvelles méthodes. Ensuite, nous avons développé un schéma de couplage entre TRIPOLI-4 et le code de thermohydraulique sous-canal SUBCHANFLOW. Enfin, nous avons réalisé une analyse préliminaire de la propagation des incertitudes au sein du calcul couplé sur un modèle simplifié. En effet, les fluctuations statistiques sont inhérentes à notre schéma de par la nature stochastique de TRIPOLI-4. De plus, les équations de la thermohydraulique étant non-linéaires, la propagation des incertitudes au long du calcul doit être étudiée afin de caractériser la convergence du résultat.

Title : Coupling between Monte Carlo neutron transport and thermal-hydraulics for the simulation of transients due to reactivity insertions

Keywords : Neutron transport, Monte Carlo simulation, thermal-hydraulics, multi-physics, transients, dynamic

Abstract : One of the main issues for the study of a reactor behaviour is to model the propagation of the neutrons, described by the Boltzmann transport equation, in the presence of multi-physics phenomena, such as the coupling between neutron transport, thermal-hydraulics and thermomechanics. Thanks to the growing computer power, it is now feasible to apply Monte Carlo methods to the solution of non-stationary transport problems in reactor physics, which play an instrumental role in producing reference numerical solutions for the analysis of transients occurring during normal and accidental behaviour.

The goal of this Ph. D. thesis is to develop, verify and test a coupling scheme between the Monte Carlo code TRIPOLI-4 and thermal-hydraulics, so as to provide a reference tool for the simulation of reactivity-

induced transients in PWRs.

We have first tested the kinetic capabilities of TRIPOLI-4 (i.e., time dependent without thermal-hydraulics feedback), evaluating the different existing methods and implementing new techniques. Then, we have developed a multi-physics interface for TRIPOLI-4, and more specifically a coupling scheme between TRIPOLI-4 and the thermal-hydraulics sub-channel code SUBCHANFLOW. Finally, we have performed a preliminary analysis of the stability of the coupling scheme. Indeed, due to the stochastic nature of the outputs produced by TRIPOLI-4, uncertainties are inherent to our coupling scheme and propagate along the coupling iterations. Moreover, thermal-hydraulics equations are non linear, so the prediction of the propagation of the uncertainties is not straightforward.

