POLITECNICO DI TORINO

MSc in Sustainable Nuclear Energy Engineering

Master's Thesis

Development of a multiphysics and multi-scale modeling Approach for Molten Salt Reactors with GeN-FOAM and OpenModelica

Supervisors:
Dr. N. Abrate
Prof. A. Froio
Prof. S. Dulla

Candidate:
Luigi Areddia

23rd July 2026

Abstract

The majority of nuclear power plants worldwide currently are pressurized water reactors (PWRs). In these, water is used as a coolant and is pumped under high pressure to the reactor core, where it is heated by the energy released by the fission of nuclei and released as steam, which drives turbines to produce electricity. Molten Salt Reactors (MSRs) are Generation-IV nuclear fission reactors in which either the fuel and/or the coolant is a molten salt. If salt is used as a primary coolant, it can absorb huge amounts of heat at atmospheric pressure, enabling these reactors to operate at very high temperatures. This could in turn enable the production of high-grade processing heat, thereby opening up the possibility of decarbonising industrial processes as well as working in cogeneration mode. International interest is increasing in MSRs, because they have the potential to provide large amounts of efficient and cost effective electricity with some liquid-fuel MSRs designs that allow for continuous on-site chemical reprocessing to remove waste products and add new fuel, which enables the reactor to consume a much higher percentage of its fuel and generate significantly less long-lived waste when using a fast neutron spectrum. Furthermore, their operation at atmospheric pressure renders them inherently safe by design. One of the most promising MSRs design is based on liquid fuel, in which fissile material is dissolved into a molten salt-based coolant. The liquid fuel circulates through the reactor core where it undergoes fission and generates heat. The large amount of heat generated is then transferred to a secondary loop, where it drives a turbine to produce electricity. The liquid fuel circulation introduces an additional layer of complexity to the reactor design, making these systems inherently multiphysics. MSR reactors offer a number of advantages in terms of intrinsic safety. Given the liquid nature of the fuel, an increase in temperature results in the salt expanding, thereby slowing down the fission reaction. Furthermore, MSRs function at temperatures higher then 500°C, thereby achieving enhanced thermodynamic efficiency and generating high-temperature heat, which is optimal for industrial processes and hydrogen production.

The main challenges in the design of MSRs are the total absence of experimental data and the development of a reliable and accurate multiphysics simulation tool that can predict the complex behavior of the reactor under different operating conditions. On top of this, the inherent multiphysics features of the MSR cores make the simulation of normal and accidental transient scenarios with legacy system-level codes (e.g., CATHARE) unsuitable, as they are mostly based on the point-kinetic model; hence, in order to guarantee an acceptable simulation time while retaining the required accuracy, the system-level code should be coupled to a full-core model. The aim of this thesis is thus to develop an integrated multi-scale

and multiphysics model of the MSR plant coupling a full-core model, developed with the GeN-Foam code, with a system-level model, developed in the OpenModelica language. This thesis work serves as a foundational step toward the development of a tool able to provide a high-fidelity description of the intrinsic 3D multiphysics core behaviour, still relying on a fast-running, lumped-parameter system model within OpenModelica for both primary and secondary loops.
At first, the fullcore model in GeN-Foam, which is rooted in the OpenFOAM multiphysics toolkit, is benchmarked against the so-called CNRS benchmark, which is a multi-step numerical benchmark developed, as part of the European EVOL MSR project, to assess the ability of a code to accurately capture the complex behavior of a MSR. After this verification, the fullcore multiphysics model is coupled with OpenModelica. This integration aims to facilitate the development of a more comprehensive and precise system-level model of the reactor, as demonstrated comparing the behaviour of system model with a point-kinetic description with respect to the same system but employing a 3D multiphysics model of the core.

Table of Contents

Chapter 1

Introduction

This introductory chapter presents the general framework in which molten salt reactor research is under development. In *Section 1.1* the climate change and the motivations for pursuing nuclear energy are introduced. To have a more thorough understanding, nuclear fission and reactor technology are briefly explained in *Sections 1.2* and lastly an overview of the reactor types and new technologies is provided in *Section 1.3*.

1.1 The dependence on fossil fuels

Over the past five decades, global energy consumption has been characterised by a persistent reliance on fossil fuels, despite significant technological and policy advancements in low-carbon alternatives. As is well documented, oil, coal and natural gas have remained the predominant sources of primary energy for some time. Collectively, these energy sources have accounted for the overwhelming majority of global demand for a considerable period, see *Figure 1.1*.
Since the 1970s, oil has been the most significant single energy source. Following an initial increase, oil consumption exhibited fluctuations related to geopolitical crises and economic downturns, but the long-term trajectory has been upward, reaching more than 55,000 TWh in 2024. Coal consumption witnessed a marked increase from the early 2000s onwards, primarily driven by substantial industrial growth in Asia, and stabilised at approximately 40,000 TWh after 2010. Natural gas has exhibited the most sustained growth trajectory, having increased fourfold since the mid-1970s. This is largely attributable to its promotion as a comparatively cleaner and more versatile fossil fuel.

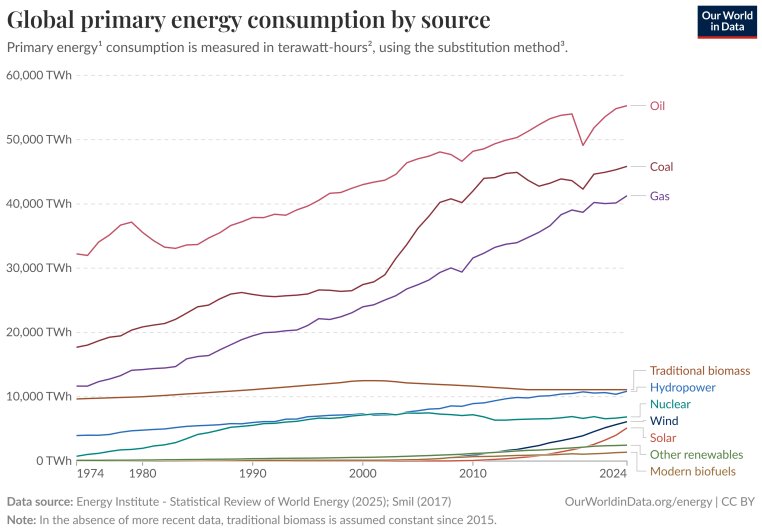


Figure 1.1: *Total primary energy consumption by source, 1974 and 2024 [1].*

Conversely, the augmentation of renewable and nuclear energy has been more constrained in absolute terms, although their relative growth rates have been substantial. Hydroelectric increased steadily to approximately 10,000 TWh, while nuclear energy underwent rapid expansion during the late twentieth century, subsequently plateauing after the 1990s. Since the early 2000s, wind and solar energy showed exponential growth, propelled by cost reductions, technological innovation, and policy incentives. However, the collective impact of these sources remains significantly smaller than that of fossil fuels.

The data presented herein underscores two significant dynamics. Firstly, the global demand for energy has increased substantially, meaning that even the rapid growth of renewable sources has not displaced fossil fuels on a large scale. Secondly, the pervasive role of fossil fuels in industrial production, electricity generation and transportation highlights the inherent challenge of achieving substantial decarbonisation. Whilst the accelerating deployment of wind and solar power provides evidence of transition, the persistence of fossil fuel dominance illustrates the complexity of aligning energy systems with climate mitigation targets.

In the future, nuclear energy is likely to play a pivotal role in bridging this gap. Contrarily to variable renewable sources, nuclear power provides a stable, large-scale, and low-carbon supply of electricity that can support both baseload demand and the increasing electrification of energy systems. Advancements in reactor technologies, including small modular reactors (SMRs) and Generation-IV reactors, have the potential to address some of the economic and safety challenges that have hindered nuclear expansion since the 1990s. Moreover, as an increasing number of governments adopt net-zero targets, nuclear energy has the potential to play

a pivotal role in complementing renewables. This is due to its ability to ensure grid stability, reduce reliance on natural gas, and accelerate the phase-out of coal. The trajectory of the energy transition will therefore depend not only on continued investment in renewables but also on the willingness of policymakers and societies to reintegrate nuclear energy as a central pillar of decarbonisation strategies.

1.2 Nuclear energy

1.2.1 Fission

As previously mentioned, nuclear energy is a suitable low-carbon energy supply able to provide base load allowing to approach the net zero objective.
Nuclear energy is a form of energy released from the nucleus, the core of atoms. This source of energy can be produced in two ways: fission, when nuclei of atoms split into 2/3 fragments, or fusion, when nucleii fuse together. The nuclear energy, which is currently employed worldwide to produce electricity, is through nuclear fission, while technology to generate electricity from fusion is at the R&D phase [2].

Discovered in 1939 by Otto Hahn, Lise Meitner, and Fritz Strassmann by bombarding elements with neutrons, nuclear fission is the process where the nucleus of an atom splits into two or more smaller nucleii and other particles. These particles can include neutrons, alpha particles (helium nuclei), beta particles (electrons), and gamma rays (which consist of particles of light, or photons).

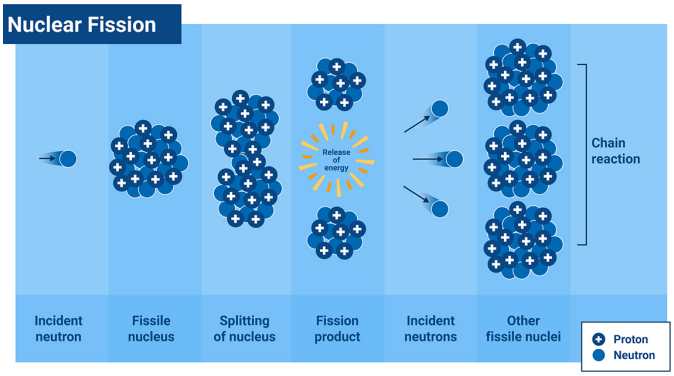


Figure 1.2: *Nuclear fission process of Uranium-235 [2].*

The fission reaction is initiated by the absorption of an incident neutron, which subsequently produces a neutron. This process induces a chain reaction, whereby the release of neutrons from fissile isotopes undergoing nuclear fission is followed

by the subsequent absorption of some of these neutrons in fissile isotopes. In the process, a small number of neutrons are ejected from the reaction (this number is aleatoric, usually between 2.5 and 3.0). These free neutrons then interact with the surrounding medium, and if fissile fuel is present, some may be absorbed and cause further fissions.
Consequently, the cycle repeats, thereby generating a self-sustaining reaction. In the instance of each reaction, energy is released in the form of heat and radiation.

1.2.2 Nuclear power plants

In a nuclear power plant, the process of converting heat into electricity is similar to the utilisation of heat from fossil fuels, such as coal, gas and oil, in conventional power generation. Different reactors share the same key components: the core, where nuclear fission occurs, the turbine, where high-pressure steam flows pass through the blade to convert thermal energy into mechanical energy used by a generator to produce electricity, and the condenser, where the steam is cooled down and converted back to water to be reused in the cycle. Main difference in the two typical design of LWR is the presence of steam generator, see *figure 1.3*. This enables steam to be produced from the heat absorbed by the high pressure water coming from the core.

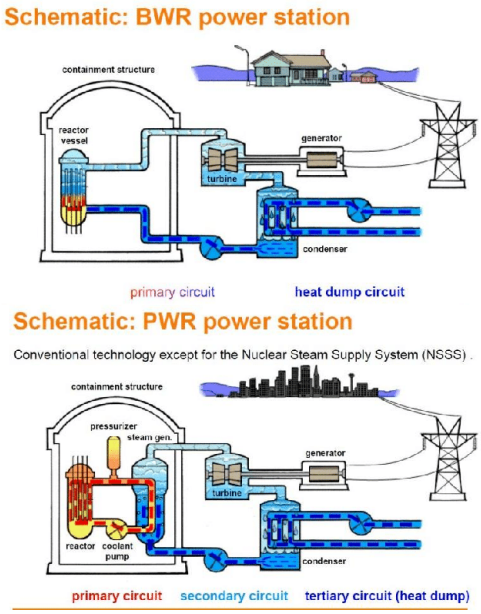


Figure 1.3: *Schematic of a PWR and a BWR [3].*

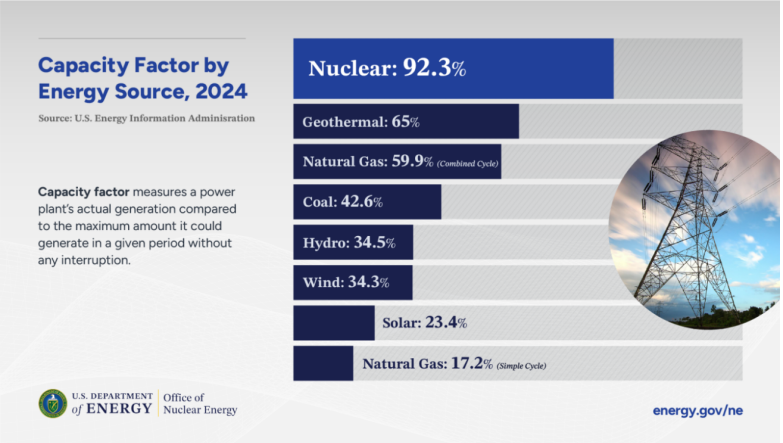


Figure 1.4: *Capacity factors by energy production technology in 2024 [4].*

Nuclear energy appears to be a highly suitable option, given its capacity factor, defined as the ratio of energy produced to the total energy potential of the plant (see *Figure 1.4*), which is one of the highest among power plants. This ensures that nuclear energy is not susceptible to intermittent issues, and the fuel requirements are relatively modest. Indeed, the utilisation of nuclear technologies for energy production relies on nuclear reactions, which exhibit a substantially higher energy density compared to other technologies by several orders of magnitude. Moreover, these technologies do not generate CO_2 during the operational phase.

1.3 Reactors and challenges

1.3.1 Reactors history and types

The issue of global warming necessitates a comprehensive approach, one of which includes the consideration of costs as a factor in the resolution of this issue. It is evident that further advancements in the field of nuclear technology are imperative to ensure the fulfilment of future energy demands. The development of nuclear reactors can be categorised into several generations and, as stated in [5], each generation is indicative of a distinct stage of technological advancement, with corresponding improvements in safety features.

1. **First Generation** nuclear reactors were the early experimental and commercial reactors built from the 1950s to the 1960s. They were primarily focused on proving the feasibility of nuclear energy for electricity generation. These reactors were based on natural uranium or slightly enriched uranium fuel and used graphite or heavy water as moderators.

2. **Second Generation** reactors were developed from the 1960s to the 1990s and saw more widespread commercial use. They featured improved safety and operational characteristics compared to the first generation. Light water reactors (LWRs) became the dominant type, including Pressurized Water Reactors (PWRs) and Boiling Water Reactors (BWRs). These reactors were designed to produce electricity efficiently, but their safety systems often required active human intervention.

3. **Third Generation/Third Generation+** reactors, developed from the 1990s to the present, represent significant advancements in safety, efficiency, and sustainability. They incorporate passive safety features that can operate without human intervention or external power sources. Examples include the Advanced Boiling Water Reactor (ABWR) and the European Pressurized Reactor (EPR). These reactors also focus on reducing waste and improving fuel utilization.

4. **Fourth Generation** reactors are currently in the research and development phase. They represent the next step in nuclear reactor technology and aim to further improve safety, sustainability and waste management.
   Fourth-generation reactors focus on innovative concepts such as fast neutron reactors, molten salt reactors, and gas-cooled fast reactors. These reactors have the potential to use alternative fuels and contribute to nuclear waste reduction

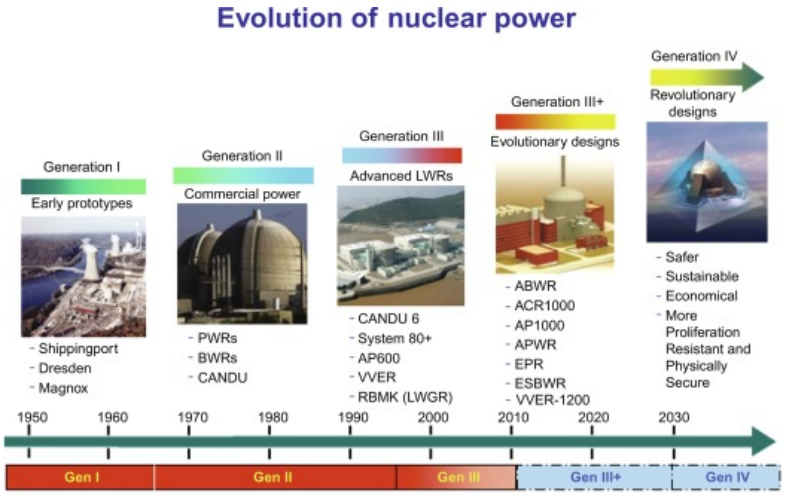


Figure 1.5: *Nuclear reactor generations [5].*

1.3.2 The role of Gen IV reactors

The Generation IV International Forum (GIF) was initiated, as reported in [6], by the US Department of Energy in 2000 and formally chartered in mid-2001. It is an international collective representing governments of 13 countries where nuclear energy is significant now and also seen as vital for the future. Most are committed to joint development of the next generation of nuclear technology. The original charter members of GIF are Argentina, Brazil, Canada, France, Japan, South Korea, South Africa, the UK and the USA. They have been joined by Switzerland, China, Russia, Australia and, through the Euratom research and training programme, the European Union. The purpose of GIF is to share R&D rather than build reactors. After two years' deliberation and review of about one hundred concepts, late in 2002 GIF (then representing ten countries) announced the selection of six reactor technologies which they believed representing the future of nuclear energy. These were selected on the basis of being clean, safe and cost-effective means of meeting increased energy demands on a sustainable basis, while being resistant to diversion of materials for weapons proliferation and secure from terrorist attacks [7]. The reactor technologies chosen are:

1. **Gas-cooled fast reactor (GFR)**. Like other helium-cooled reactors which have operated or are under development, GFRs will be high-temperature units, typically 800-850°C. They employ similar reactor technology to the VHTR, suitable for power generation, thermochemical hydrogen production or other process heat.

2. **Lead-cooled fast reactor (LFR)**. The LFR is a flexible fast neutron reactor which can use depleted uranium or thorium fuel matrices, and burn actinides from LWR fuel. Liquid metal (Pb or Pb-Bi eutectic) cooling is at atmospheric pressure by natural convection (at least for decay heat removal).

3. **Molten salt reactor (MSR)** (now two variants): one a fast reactor with fissile material dissolved in the circulation fuel salt; the other with solid particle fuel in graphite and the salt functioning only as coolant. In what is considered to be a normal MSR, the uranium fuel is dissolved in the fluoride salt coolant which circulates through graphite core channels to achieve some moderation and an epithermal neutron spectrum.

4. **Sodium-cooled fast reactor (SFR)**. The SFR uses liquid sodium as the reactor coolant, allowing high power density with low coolant volume, at low pressure. It builds on some 390 reactor-years experienced with sodium-cooled fast neutron reactors over five decades and in eight countries, and was initially the main technology of interest in GIF, and remains at the forefront despite needing a sealed coolant system. A variety of fuels is possible. The SFR

utilises depleted uranium as the fuel matrix and has a coolant temperature of 500-550°C enabling electricity generation via a secondary sodium circuit, the primary one being at near atmospheric pressure.

5. **Supercritical water-cooled reactor (SCWR)**. This is a very high-pressure water-cooled reactor which operates above the thermodynamic critical point of water (374ºC, 22 MPa) to give a thermal efficiency about one-third higher than today's light water reactors from which the design evolves. The supercritical water (25 MPa and 510-550°C) directly drives the turbine, without any secondary steam system, simplifying the plant. Two design options are considered: pressure vessel and pressure tube. Passive safety features are similar to those of simplified boiling water reactors. Fuel is uranium oxide, enriched in the case of the open fuel cycle option.

6. **Very high-temperature gas reactor (VHTR)**. These are graphite-moderated, helium-cooled reactors, based on substantial experience. Outlet temperature of over 900°C and aiming for 1000ºC enables thermochemical hydrogen production via an intermediate heat exchanger, with electricity cogeneration, or direct high-efficiency driving of a gas turbine (Brayton cycle). At lower outlet temperatures, the Rankine steam cycle may be used for electricity generation, and this is the focus for demonstration projects.

A summary table is reported here below.

Generation IV reactor designs under development by GIF

	Neutron spectrum (fast/thermal)	Coolant	Temperature (°C)	Pressure*	Fuel	Fuel cycle	Size (MWe)	Use
Gas-cooled fast reactors	fast	helium	850	high	U-238 +	closed, on site	1200	electricity & hydrogen
Lead-cooled fast reactors	fast	lead or Pb-Bi	480-570	low	U-238 +	closed, regional	20-180** 300-1200 600-1000	electricity & hydrogen
Molten salt fast reactors	fast	fluoride salts	700-800	low	UF in salt	closed	1000	electricity & hydrogen
Molten salt reactor - **advanced high-temperature reactors**	thermal	fluoride salts	750-1000		UO_2 particles in prism	open	1000-1500	hydrogen
Sodium-cooled fast reactors	fast	sodium	500-550	low	U-238 & MOX	closed	50-150 600-1500	electricity
Supercritical water-cooled reactors	thermal or fast	water	510-625	very high	UO_2	open (thermal) closed (fast)	300-700 1000-1500	electricity
Very high temperature gas reactors	thermal	helium	900-1000	high	UO_2 prism or pebbles	open	250-300	hydrogen & electricity

Figure 1.6: *Summary of the main characteristics of Gen IV reactors [7].*

1.3.3 Molten Salt Reactors

Molten Salt Reactors (MSRs) utilise liquid salt as both fuel and coolant. The molten salt is composed of fissile material and circulates through a core, where the fission chain reaction occurs, and heat exchangers, where it is cooled. The term 'fuel circuit' is employed to denote the primary circuit equipped with the heat exchangers and the pumps. In conventional systems, the fuel salt exchanges heat with another salt, circulating within the 'intermediate circuit'. This process ultimately results in the transfer of heat to a power conversion system.

The potential advantages of using MSRs are a significant motivation for research and development projects in the nuclear industry, including quick reactivity feedbacks and the capacity to operate at elevated temperatures without pressurisation. Research is being conducted into both fluoride and chloride salts with uranium or plutonium and, on occasion, thorium for breeding purposes.
Fission products are removed continuously and the actinides are fully recycled, while plutonium and other actinides can be added along with U-238, without the need for fuel fabrication. Coolant temperature is 900°C at very low pressure. A secondary coolant system is used for electricity generation, and thermochemical hydrogen production is also feasible.

The european consortium dedicated to the R&D of molten salt reactors is the **EU kNowleDge hUb foR enAbling MolteN Salt ReaCtor safety development and dEployment(ENDURANCE)** project, [8]. The aim of the project is to support the safe operation and the technological development of Molten Salt Reactor (MSR) technology in Europe, through the knowledge advancement in different fields of MSR research and safety assessment, connecting the needs of reactor designers and industry with the university and research centre capabilities and the regulator requirements. The partners of the consortium include: Politecnico di Milano, FPM, CNRS, IMT, INPG, CV Rez, CEA, Framatome, HZDR, ASNR, KIT, NRG, Politecnico di Torino, TU Delft, JRC, ETHZ, Naarea, PSI, Seaborg, Stellaria, Thorizon, UCB.
The project has identified five critical areas in which the effort on R&D and safety assessment is more required to close the gap for demonstrating that MSR technology could be deployed in compliance with the highest safety standards, excelling in waste management and proliferation resistance, thus providing a relevant contribution to the integrated European energy vision [9]. As reported in [10] France's current nuclear fleet is based on light water reactors (LWRs) that use a combination of uranium oxide (UOX) and mixed oxide (MOX) fuels, implementing a mono-recycling strategy that recovers plutonium from spent UOX for reuse in MOX fuel. This approach has the advantage of saving uranium and limiting waste, but successive plutonium recycling in LWRs leads to degraded plutonium isotopic quality and increased minor actinide (MA) production, especially americium, which

dominates long-term radiotoxicity and waste heat load. Fast-spectrum reactors offer superior transmutation capabilities, as high-energy neutrons favor fission over capture reactions, thereby stabilising plutonium inventories and concomitantly reducing MA production. However, it should be noted that complete MA transmutation necessitates multi-recycling over extended periods of time. Molten salt fast reactors (MSFRs) have the potential to enhance solid-fueled systems by utilising liquid fuel, thereby eliminating limitations imposed by cladding, and facilitating continuous recycling through pyrochemical processes. France's ARAMIS (CEA/Orano) and ISAC (CEA/CNRS/EDF/Framatome/Orano) projects explore these concepts. As reported in [10] France's current nuclear fleet is based on light water

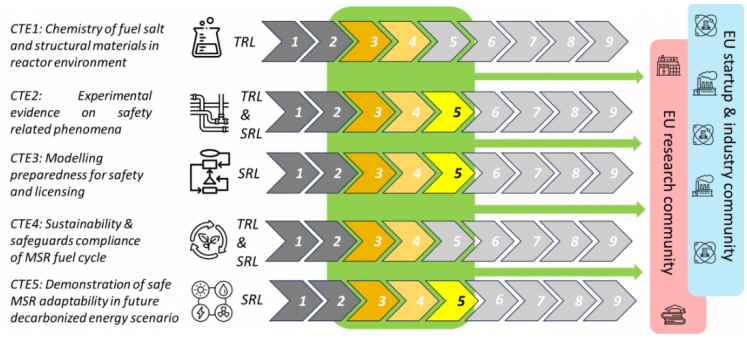


Figure 1.7: *ENDURANCE R&D activities [9].*

reactors (LWRs) that use a combination of uranium oxide (UOX) and mixed oxide (MOX) fuels, implementing a mono-recycling strategy that recovers plutonium from spent UOX for reuse in MOX fuel. This approach has the advantage of saving uranium and limiting waste, but successive plutonium recycling in LWRs leads to degraded plutonium isotopic quality and increased minor actinide (MA) production, especially americium, which dominates long-term radiotoxicity and waste heat load. Fast-spectrum reactors offer superior transmutation capabilities, as high-energy neutrons favor fission over capture reactions, thereby stabilising plutonium inventories and concomitantly reducing MA production. However, it should be noted that complete MA transmutation necessitates multi-recycling over extended periods of time. Molten salt fast reactors (MSFRs) have the potential to enhance solid-fueled systems by utilising liquid fuel, thereby eliminating limitations imposed by cladding, and facilitating continuous recycling through pyrochemical processes. France's ARAMIS (CEA/Orano) and ISAC (CEA/CNRS/EDF/Framatome/Orano) projects explore these concepts.

1.4 Objectives and structure of the work

The main objective of this thesis is to develop and assess a multiphysics and multi-scale simulation framework for liquid fuel molten salt reactors by coupling the spatially resolved GeN-Foam solver with a system-level plant model implemented in OpenModelica. The framework is intended to retain a detailed description of the coupled neutronic and thermal hydraulic behaviour of the reactor core while enabling the flexible representation of external systems, actuators and control strategies.

The work is organised around two specific objectives. First, the GeN-Foam model is implemented and numerically verified against the CNRS MSR benchmark, considering single physics, steady state coupled and time dependent cases. Second, the coupled framework established between GeN-Foam and OpenModelica is applied to an idealised degradation of the heat removal capability, and the influence of the parameters of a PI-controlled emergency cooling action is investigated.

Chapter 2 describes the CNRS benchmark, the physical models and the numerical methodology. Chapter 3 presents the benchmark results and the comparison with the reference code-to-code solutions. Chapter 4 introduces the GeN-Foam–OpenModelica co-simulation architecture and verifies the coupling interface. Chapter 5 develops the idealised emergency cooling model and discusses the associated transient and parametric analyses. The final chapter summarises the main conclusions, limitations and possible future developments.

Chapter 2

CNRS Benchmark description and calculus methodology

2.1 Introduction to the benchmark

The increasing interest in MSR clashes with a significant problem: the verification & validation of the models and tools for their simulation is not currently possible, as experimental data are available only from the Molten Salt Reactor Experiment [11]. Constructed in 1964 at the Oak Ridge National Laboratory (ORNL), this was a thermal reactor with graphite moderator and salt channels, characteristics which make it vastly different from the current fast MSR designs. Verification of single physics codes can be achieved for many applications, for examples in the field of CFD and neutronics, but verification of multiphysics codes remains a difficult task. The use of a molten salt both as fuel and coolant leads to unique physics phenomena. A distinctive feature of molten salt reactors is the transport of delayed neutron precursors (DNP) by the circulating fuel. Unlike conventional solid fuel reactors, where precursors remain essentially stationary within the fuel matrix, in MSRs they are advected by the flow field and may decay far from their production location. As a consequence, the delayed neutron source distribution becomes strongly dependent on the thermal hydraulic field, introducing an additional coupling between fluid dynamics and reactor kinetics. Correct modelling of DNPs transport is therefore essential for accurate prediction of reactor reactivity and transient behaviour. Another distinctive feature of MSRs is the thermal feedback effect induced by fuel expansion. These features are absent in traditional solid fuel reactors, therefore classical scientific codes used in the nuclear community are

unsuitable for simulating MSRs behavior. For this reason, following the research activities conducted at the LPSC/CNRS-Grenoble in 2015 in the framework of the European EVOL MSR project, a benchmark was proposed in 2019. The benchmark was named after the centre, and consequently, it is now referred to as the 'CNRS Benchmark' [12] in order to provide a common reference for the validation of multiphysics codes for MSR applications.

2.1.1 Description of the benchmark

Code verification is particularly important for MSR multiphysics codes for the peculiarities mentioned in previous subsection. The complex interaction between neutronics and thermal hydraulics makes it difficult to isolate the source of discrepancies in full core simulations. For this reason, the CNRS benchmark was developed using a step-by-step approach in which physical phenomena are progressively introduced. This methodology allows the verification of individual physics models and their coupling strategies before moving to fully coupled transient calculations. Therefore, the objective of this work is not only to reproduce the benchmark solutions, but also to assess the capability of GeN-Foam to accurately model the coupled neutronic and thermal hydraulic phenomena that characterize molten salt reactor systems.
The benchmark is approached using a simplified 2D geometry and boundary conditions to simulate a bare reactor core. The salt flow is considered to be incompressible and laminar, and buoyancy is modeled adopting the Boussinesq approximation, decay heat is neglected as well. No prescription is given regarding the neutronics model.
The benchmark consists of three phases designed to assess the coupling capabilities of multiphysics tools are subdivided into steps. In phase 0 single physics verification is performed to validate the single modules, in phase 1 steady state coupling is tested and in phase 2 time dependent coupling is performed.

2.1.2 Geometry and boundary conditions

The geometry of the benchmark is a 2D, 2m by 2m cavity filled with molten salt, as shown in Figure 2.1. The cavity centerlines AA' and BB' shown in figure are used for pointwise comparisons. The thermal hydraulic boundary conditions are similar to another well known benchmark in the CFD field, which is the lid driven cavity. The walls are thus considered to be all adiabatic, and the velocity is set to zero at the walls (no slip condition), except for the top wall where a constant velocity of 0.5 m/s is imposed.

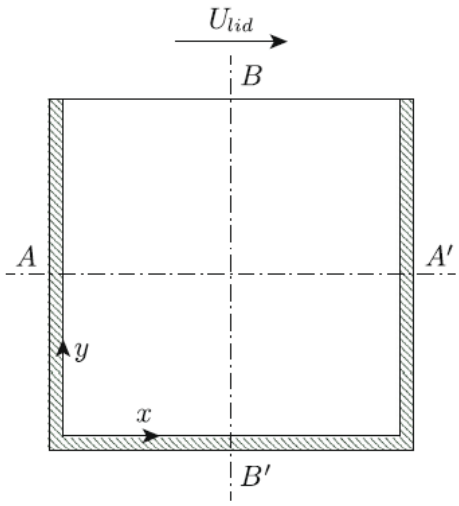


Figure 2.1: *Geometry of the CNRS benchmark [12].*

The initial temperature is equal to 900 K. In order to simulate the salt cooling effect, a volumetric heat sink is applied as follows:

$$q'''(r) = \gamma(T_{ref} - T(r)) \tag{2.1}$$

where $T(r)$ is the salt temperature at point r, T_{ref} is 900 K and γ is the volumetric heat transfer coefficient, uniform throughout the entire domain. For convenience, the same reference temperature is chosen for the Boussinesq approximation.
The domain is treated as a homogeneous, bare reactor. Therefore, the standard vacuum conditions are applied for the neutron flux to each boundary, together with a homogeneous Neumann condition for the delayed neutron precursors.

2.1.3 Fuel salt and nuclear data

The fuel salt is $LiF - BeF_2 - UF_4$, whose molar composition is reported in table 2.1

	^{6}Li	^{7}Li	^{9}Be	^{19}F	^{235}U
atomic fraction (%)	2.11488	26.0836	14.0992	56.3969	1.30545

Table 2.1: *Molar composition of the fuel salt.*

Its fluid properties are considered constant with temperature and uniform in space. They are reported in Table 2.2.

Property	Units	Value
Density	kg m^{-3}	2.0×10^3
Kinematic viscosity	m^2 s^{-1}	2.5×10^{-2}
Volumetric heat capacity	J m^{-3} K^{-1}	6.15×10^6
Thermal expansion coefficient	K^{-1}	2.0×10^{-4}
Prandtl number	–	3.075×10^5
Schmidt number	–	2.0×10^8

Table 2.2: *Thermophysical properties*

The input neutronic data are taken from the JEFF-3.1 library (Koning et al., 2006) at T_{ref}= 900 K. In this benchmark, the Doppler effect on microscopic cross sections is negligible; this avoids biases arising from the differences in treatment of this complex effect by each code. Consequently, cross sections and diffusion coefficients are affected only by the salt expansion feedback and scale with the temperature according to

$$\Sigma_r(T) = \Sigma_r(T_{ref}) \frac{\rho_{fuel}(T)}{\rho_{fuel}(T_{ref})} \tag{2.2}$$

$$D(T) = D(T_{ref}) \frac{\rho_{fuel}(T_{ref})}{\rho_{fuel}(T)} \tag{2.3}$$

where ρ_{fuel} is the fuel density, D is the diffusion coeffient and Σ_i is the generic macroscopic cross section.
Group constants were generated with Serpent and condensed into six groups, using the spatial homogenization and group collapsing models. Finally, delayed neutron precursors are grouped into eight families. The whole set of data and the energy group division are reported in Appendix A.

2.2 Gen-Foam implementation

2.2.1 Physical models

Thermal hydraulics

The thermal hydraulic behaviour of the molten salt is described by the incompressible Navier-Stokes equations coupled with the energy conservation equation. Since the density variations are relatively small and mainly affect the buoyancy force, the Boussinesq approximation is adopted. Assuming an incompressible Newtonian fluid, the conservation of mass is expressed by the continuity equation

$$\nabla \cdot \mathbf{u} = 0, \tag{2.4}$$

where $\mathbf{u}$ denotes the fluid velocity vector. The conservation of momentum is governed by the incompressible Navier-Stokes equations

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}\right)=-\nabla p+\mu\nabla^2\mathbf{u}+\rho\mathbf{g}, \tag{2.5}$$

where ρ is the fluid density, p is the pressure, μ is the dynamic viscosity, and $\mathbf{g}$ is the gravitational acceleration vector. The left hand side represents the inertial effects, while the right hand side accounts for pressure, viscous diffusion, and body forces. Natural convection induced by temperature differences is modelled using the Boussinesq approximation. Under this assumption, density is considered constant throughout the governing equations except in the buoyancy term, where it is expressed as

$$\rho=\rho_0\left[1-\beta(T-T_0)\right], \tag{2.6}$$

where ρ_0 is the reference density at the reference temperature T_0, and β is the volumetric thermal expansion coefficient. The resulting buoyancy force appearing in the momentum equation becomes

$$\mathbf{F}_b=-\rho_0\beta(T-T_0)\mathbf{g}, \tag{2.7}$$

allowing density variations to drive the flow while preserving the incompressibility assumption. In GeN-Foam, the governing equations are discretized using the finite volume method within the OpenFOAM framework. The pressure-velocity coupling is handled through the PIMPLE algorithm, a combination of PISO (pressure-implicit split-operator) and SIMPLE, [13]. Within one time step, the algorithms solve a pressure equation, to enforce mass conservation, with an explicit correction to velocity to satisfy momentum conservation. The energy equation is solved in a segregated manner, the **onePhase** subsolver of GeN-Foam is used to solve the coupled system of equations for the fluid flow and heat transfer in terms of specific enthalpy, sum of internal energy and kinematic pressure, i.e. $h=e+\frac{p}{\rho}$

$$\frac{\partial(\rho h)}{\partial t}+\nabla\cdot(\rho\mathbf{u}h)+\frac{\partial(\rho K)}{\partial t}+\nabla\cdot(\rho\mathbf{u}K)-\frac{\partial p}{\partial t}=-\nabla\cdot\mathbf{q}+\nabla\cdot(\boldsymbol{\tau}\cdot\mathbf{u})+\rho r+\rho\mathbf{g}\cdot\mathbf{u} \tag{2.8}$$

Where K is the specific kinetic energy, $\mathbf{q}$ is the heat flux, $\boldsymbol{\tau}$ is the stress tensor, and r is the volumetric heat source. Being the thermophysical properties of the fuel salt constant the temperature field can be retrived by simply dividing the specific enthalpy by the specific heat capacity, i.e. $T=h/c_p$. For both steady state and transient calculation the salt temperature, pressure and veocity fields are obtained.

Neutronics

The neutronics model is based on the multi-group diffusion approximation, which is suitable for the simplified geometry and the vacuum boundary conditions of the benchmark. Due to the diffusion approximation just the P_0 scattering matrix is required, as the diffusion coefficients is generated with the P_1 correction to recover the first order anisotropy effect.
Handling liquid fuel, the multi-group diffusion system must include the advection of DNPs. The resulting system of equations is the following:

$$\begin{cases} \dfrac{1}{v_g}\dfrac{\partial \phi_g}{\partial t} = \nabla \cdot (D_g \nabla \phi_g) - \Sigma_{r,g}\phi_g + \dfrac{\chi_{p,g}}{k_{\text{eff}}}(1-\beta_{\text{tot}})S_{n,g} \\ \qquad\qquad + \dfrac{\chi_{p,g}}{k_{\text{eff}}}(1-\beta_{\text{tot}})S_{n,g} + \chi_{d,g}S_d + S_{s,g} + S_g, \quad (g = 1, \ldots, G) \\ \dfrac{\partial C_k}{\partial t} + \nabla \cdot (\mathbf{u}C_k) = \dfrac{\beta_k}{k_{\text{eff}}} \sum_{h=1}^{G} \nu\Sigma_{f,h}\phi_h - \lambda_k C_k, \qquad (k = 1, \ldots, N) \end{cases} \tag{2.9}$$

The explicit multi-group source terms are defined as:

$$S_{n,g} = \sum_{h \neq g} \nu\Sigma_{f,h}\phi_h, \quad S_d = \sum_{k=1}^{N} \lambda_k C_k, \quad S_{s,g} = \sum_{h \neq g} \Sigma_{h \to g}\phi_h$$

Where ϕ_g is the neutron flux in group g, C_k is the concentration of delayed neutron precursors in family k, v_g is the neutron speed in group g, D_g is the diffusion coefficient in group g, $\Sigma_{r,g}$ is the removal cross section in group g, $\nu\Sigma_{f,g}$ is the production cross section in group g, $\chi_{p,g}$ and $\chi_{d,g}$ are the prompt and delayed neutron spectra respectively, β_{tot} is the total delayed neutron fraction, β_k is the delayed neutron fraction for family k, λ_k is the decay constant for family k, $S_{n,g}$ represents Fission source from other energy groups ($h \neq g$), S_d is the total delayed neutron source from all precursor groups and $S_{s,g}$ is the scattering source from other energy groups. For the calculation **6 energy groups** and **8 delayed neutron precursors family** were employed.
In steady state, source free, calculations, i.e. when time derivatives are set to zero, an eigenvalue problem is solved. The resolution yields the effective multiplication factor k_{eff}, the precoursors distribution and the neutron fluxes distribution. Consequently, the volumetric power density is determined. Introducing the time dependent terms the unknown is no longer k_{eff} but instead, the evolution in time of neutron flux, precursors concentration and reactor power.

2.2.2 Coupling strategy

Unlike conventional solid fuel reactors, where precursors remain essentially stationary within the fuel matrix, in MSRs they are advected by the flow field and may decay far from their production location. As a consequence, the delayed neutron source distribution becomes strongly dependent on the thermal hydraulic solution, introducing an additional coupling between fluid dynamics and reactor kinetics. Correct modelling of precursor transport is therefore essential for accurate prediction of reactor reactivity and transient behaviour, and constitutes one of the primary verification targets of the CNRS benchmark. The solution of the coupled problem is performed through a partitioned approach, obtained through a fixed-point (Picard) iterative procedure. The neutronics and thermal hydraulics sub-solvers are executed sequentially while exchanging the relevant coupling fields. At each coupling iteration, the neutronics solver computes the volumetric fission power density, which is passed to the thermal hydraulic solver. The latter, in this specific case, is subsequently tasked with calculating the velocity, pressure, and temperature fields. These updated quantities are then transferred back to the neutronics solver, where they are used to updated the fuel volumetric power density based on the coupling fields, incorporating feedbacks from coolant temperature. The procedure is repeated until the prescribed convergence criteria are satisfied.
The coupling between physics is achieved by projecting coupling variables from the mesh they are calculated, to the mesh they need to be used, the coupling mechanism chosen is volumetric coupling. This method is is best suited for geometrically overlapping domains and is achieved by volumetrically mapping fields from one mesh to another, making use of OpenFOAM mapping algorithms. In this approach, coupling variables are projected from the mesh on which they are computed to the mesh on which they are required.
To account for density feedback, the fuel density is evaluated at four reference temperatures between 950 and 1400 K, and the corresponding macroscopic cross sections and diffusion coefficients are scaled according to the salt expansion. During the simulation, the neutronics subsolver receives the temperature field mapped from the thermal hydraulic mesh and interpolates the group constants to the local temperature using the built in Radial Basis Function (RBF) interpolation scheme. In order to preserve the physics, the interpolation between reference states is linear. Indeed, a material's macroscopic cross section and its mass density exhibit a linear relationship. It is important to note that the temperature interpolation is linear too, due to the fact that the fuel density is linearly related to the temperature, as a consequence of the salt expansion phenomenon.

2.3 Numerical implementation

2.3.1 Mesh and discretization

GeN-Foam is built on the OpenFOAM framework, which employs a finite volume method (FVM) for the spatial discretization of both the thermohydraulic and neutronics equations. The primary advantages of FVM are associated with its remarkable flexibility and scalability, in conjunction with its general capacity to yield sparse diagonally dominant matrices. These properties, in turn, provide fast and efficient matrix solutions. In the field of thermal hydraulics, FVM is highly advantageous because its mathematically conservative formulation strictly enforces the conservation of mass, momentum, and energy across discrete cell volumes. From a neutronics perspective, FVM is also adequate for diffusion problems. Reliable solutions are achieved by employing centered schemes, thereby ensuring the effective resolution of neutron diffusion.
Notwithstanding its considerable suitability for convection-driven problems and CFD analysis, the method exhibits certain limitations when resolving specific neutronics phenomena. Resolving the bare reactor's vacuum boundaries is highly sensitive to cell size. If the FVM mesh is too coarse, problems related to "artificial diffusion" arise, as reported in [14], causing the numerical smearing of steep flux gradients, overestimating neutron leakage, and consequently yielding an artificially unerestimated k_{eff}.
Given the extremely simple geometry of the problem, the type of mesh adopted for both thermal hydraulic and neutronic calculations is a structured Cartesian mesh with a uniform cell size. This type of mesh is easily generated using OpenFOAM's built-in utility, **blockMesh**, which allows for the creation of hexahedral meshes based on user defined blocks and vertices. Moreover, this method provides an excellent mesh in terms of non-orthogonality, skewness, and aspect ratio, which are crucial mesh quality parameters strictly required by the finite volume method.The mesh resolution is ultimately chosen to ensure an adequate representation of the flow and temperature fields, as well as the neutron flux distribution.
A mesh independence study is performed in order to determine the optimal cell size that balances accuracy and computational cost for both meshes. The approach involves the execution of simulations with progressively refined meshes for both physics, isolating them and systematically ensuring grid independence across the entire benchmark. The results of grid independence study are presented in the following sections. The procedure of the Richardson extrapolation is described in [15] and is the same for both physics. The exact, in other words grid independent, value can be evaluated as :

$$\varphi_{exact} = \frac{r_{21}^{p}\varphi_1 - \varphi_2}{r_{21}^{p} - 1} \tag{2.10}$$

The 3 grid used to evaluate this value are indexed from 1 to 3, starting respectively from the finest to the coarsest, φ_x are respectively the value of the variable in the three meshes. r is the grid refinement factor, defined as $r_{12} = h_2/h_1$, where h is the mesh size defined for structured grid as

$$h = [(\Delta x_{max})(\Delta y_{max})]^{1/2} .$$

The three meshes used for both physics are respectively characterized by a number of cells equal to 4×10^4 16×10^4 and 64×10^4 .
p is the apparent order of convergence and is evaluated as:

$$p = \left[\frac{1}{\ln(r_{21})}\right]\left[\ln\left|\frac{\varepsilon_{32}}{\varepsilon_{21}}\right| + q(p)\right]$$
$$q(p) = \ln\left(\frac{r_2^p - s}{r_{32}^p - s}\right)$$
$$s = 1 \cdot \text{sign}\left(\frac{\varepsilon_{32}}{\varepsilon_{21}}\right)$$

Where $\epsilon_{32} = \varphi_3 - \varphi_2$. Once p is calculated using a non-linear solver we can evaluate ϕ_{exact} using 2.10.The error estimate

$$e_a^{21} = \left|\frac{\varphi_1 - \varphi_2}{\varphi_1}\right|$$

is used to evaluate the Grid convergence index (GCI) as:

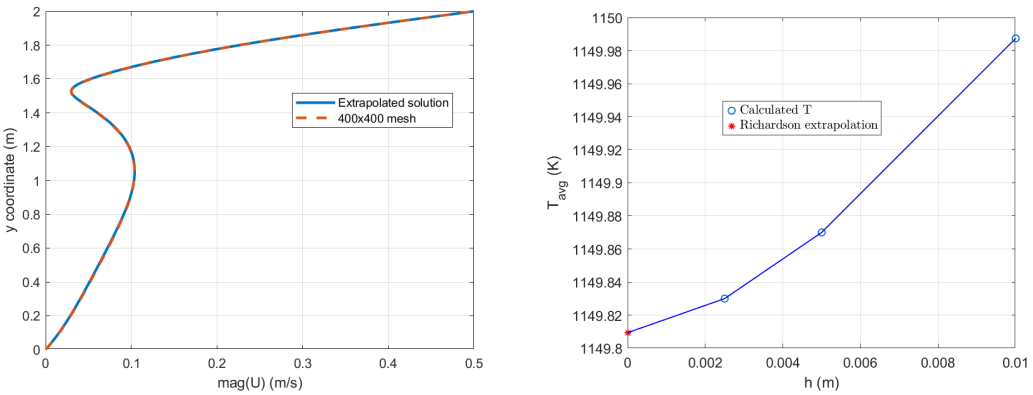
$$\text{GCI}_{\text{fine}}^{21} = \frac{F_s \cdot e_a^{21}}{r_{21}^p - 1}$$

where F_s is a safety factor that for three meshes study is set equal to 1.25. This value of F_s is demonstrated to provide results with a 95% confidence interval [15].

Thermal hydraulic

The grid independence study is performed by running 3 simulations with the different mesh resolutions, for the lid driven cavity flow and passive scalar temperature transport, (2.4.1), without neutronics. The variables to monitor for each of the two cases are the velocity profile along BB' centerlines and the average temperature in the domain. The results are shown below.
For velocity the estimated grid convergence index for the 400x400 mesh was 1.5×10^{-3} %. The extrapolated and numerical profiles are virtually indistinguishable, as can be seen in figure 2.2a, demonstrating that the spatial discretization error for this benchmark quantity is negligible.The asymptotic convergence condition was verified by evaluating the Grid Convergence Index consistency criterion,

$GCI_{23}/(r^p GCI_{21})$, which was found to be equal to 1. This confirms that the three meshes lie in the asymptotic range of convergence and that the Richardson extrapolation provides a reliable estimate of the discretization error.



(a) *Comparison of the velocity profile with Richardson extrapolation*

(b) *Extrapolated and computed average temperature in the domain*

For what regards the passive scalar transport, since neither temperature nor fluxes are imposed as boundary condition the choice of the parameter for grid convergence study is the average temperature in the domain. The estimated grid convergence index for the intermediate mesh is 3.2×10^{-3} %. Results are reported in figure 2.2b.
The low values of GCI are attributed to the structured uniform nature of the mesh and the laminar flow. Consequently, the 400x400 mesh was considered sufficiently fine for all subsequent simulations.

Neutronics

The case accounted for this grid independence study is the single physics neutronics, 2.4.1. The quantity considered is the k_{eff} as it is highly sensitive to spatial discretization. The neutron balance is heavily dictated by the fraction of neutrons leaking out of the vacuum boundaries. A coarse mesh uses centered schemes that struggle to capture the steep, non-linear drop-off of the neutron flux near these vacuum walls. This numerical error artificially smears the flux outward, forcing the solver to overestimate the number of neutrons leaking out of the system.

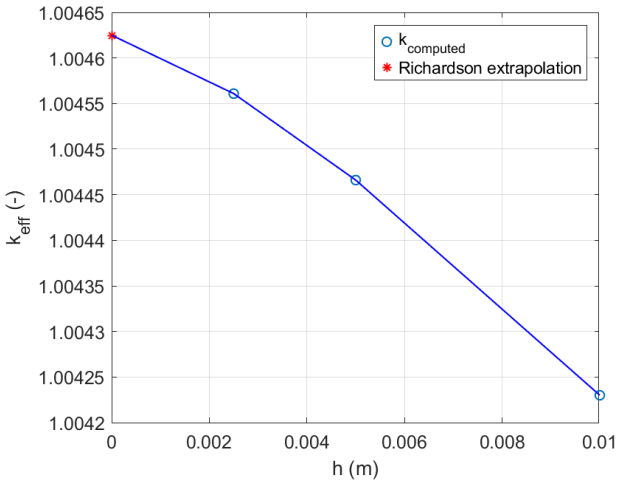


Figure 2.3: *Extrapolated and computed* k_{eff}

The estimated Grid Convergence Index for the 400x400 mesh was $\sim$ 0.4 % meaning that the mesh is sufficiently refined to avoid excessive neutron leakage wrong computation. Further refinement is possible, however the gain is limited compared to the huge computational cost required to run the simulation.

Multiphysics

For the multiphysics case the deleyed neutron source on AA' centerline has been chosen. This is particularly significant since involves both physics. In fact the however the concentration of precoursor is a neutronic quantity, they are advected in the domain by the flow field. The case taken in exam is the steady state full coupling, 2.4.2.

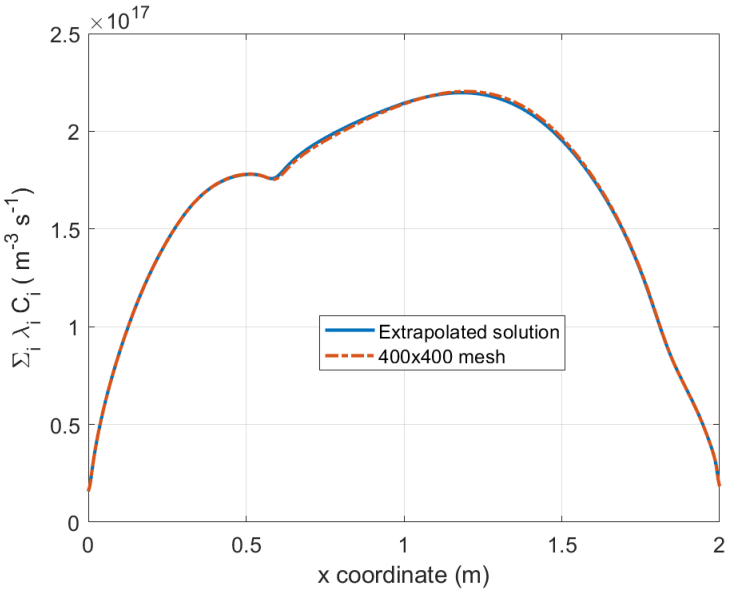


Figure 2.4: *Comparison of delayed neutron precoursors along AA' centerline*

The calculated Grid Convergence Index was 1.14%, showing that further mesh refinement is expected to produce only a minor change in the quantity of interest. Together with the observed convergence of the results, this indicates that the adopted grid provides an adequate balance between numerical accuracy and computational cost.

2.3.2 Boundary Conditions

GeN-Foam include the built in thermal hydraulic boundary condition present in OpenFOAM, thus in the model for velocity **movingWall** boundary condition with a prescribed velocity of 0.5 m/s has been implemented on top wall, while **noSlip** is prescribed on the other walls. The adiabatic condition of the reactor is ensured by enforcing zero gradient of temperature at the boundary. For pressure the homogenous Neumann boundary condition cannot be used due to the fact that ignores any pressure gradient required to balance body forces. Instead the **fixedFluxPressure** enforces consistency between the pressure gradient and the prescribed velocity flux, automatically appling the pressure gradient necessary to cancel the normal component of gravity.
To satisfy the vacuum boundary conditions for the flux, the **albedo** boundary condition, available in GeN-Foam, has been applied. This is a well known boundary condition in reactor physics problems. It depends on γ, which is computed as $\gamma = \frac{1-\alpha}{1+\alpha} \cdot \frac{1}{2}$, where α is the albedo coefficient. This value ranges from zero, when the surface is perfect absorber, to 1, where the surface is a reflector. To simulate

vacuum a value of γ equal to 0.5 is prescribed (α). For precoursors, homogenous Neumann boundary conditions are prescribed. The internal fields are inizialized as uniform with the values reported in table 2.3 .
For steady state the implementation of the heat sink has been performed using the buit in OpenFOAM function **scalarSemiImplicitSource** that allows to add a source to the energy equation using an explicit and an implicit term. For transient calculation, where gamma varies in time, a script was written and fed to OpenFOAM using **scalarCodedSource**. The latter is reported in appendix B.1.

Table 2.3: *Summary of GeN-Foam Boundary Conditions and Internal Fields.*

Thermal hydraulics			
Variable (Object)	**Patch**	**Type**	**Value**
P	*internalField*	*uniform*	*0*
	movingWall	fixedFluxPressure	uniform 0
	fixedWalls	fixedFluxPressure	uniform 0
	frontAndBack	empty	–
T	*internalField*	*uniform*	*900*
	movingWall	zeroGradient	–
	fixedWalls	zeroGradient	–
	frontAndBack	empty	–
U	*internalField*	*uniform*	*(0 0 0)*
	movingWall	fixedValue	uniform (0.5 0 0)
	fixedWalls	noSlip	–
	frontAndBack	empty	–
Neutronics			
defaultFlux	*internalField*	*uniform*	*1*
	movingWall	albedoSP3 ($\gamma = 0.5$)	uniform 1
	fixedWalls	albedoSP3 ($\gamma = 0.5$)	uniform 1
	frontAndBack	empty	–
defaultPrec	*internalField*	*uniform*	*1*
	movingWall	zeroGradient	–
	fixedWalls	zeroGradient	–
	frontAndBack	empty	–

2.3.3 Convergence criteria

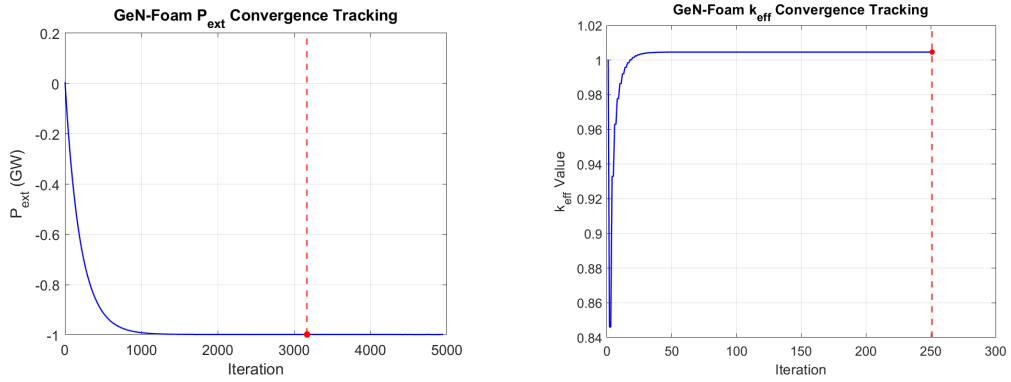
For steady state state calculations, stopping criteria are employed. The stopping criteria are based on monitoring a quantity for each physics: for thermal hydraulics the variance on the extracted power is monitored, while for neutronics the variance on the effective multiplication factor is monitored. These operations were performed during postprocessing, letting the simulation run for a fixed number of iterations, and then checking the convergence criteria.
Being the energy equation solved in terms of specific enthalpy the heat sink has been implemented as additional negative source term, as follows:

$$q''' = \gamma(T_{ref} - \frac{h(r)}{c_p}) \tag{2.11}$$

Utilizing the PIMPLE algorithm, the solution of the steady state is obtained as pseudo-transient solution. Total extracted power from the domain can be thus evaluated, at each time step, as follows:

$$P(t) = \int_V q'''(r)dV = V\gamma T_{ref} - \frac{\gamma}{c_p}\int_V h(r)dV \tag{2.12}$$

The integral is evaluated using the OpenFOAM built-in function, which performs a volume integral over the entire domain, the other quantities are all constant and uniform in the domain. The variance of the extracted power is evaluated with a script that reads the output of the integral, evaluates the extracted power and calculates the variance over the last 100 iterations. The simulation is considered converged when the variance is below 10^{-6}.
The same procedure is applied to the neutronics solver, monitoring the effective



(a) *Variance of the extracted power during the steady state simulation.*

(b) *Variance of the effective multiplication factor during the steady state simulation.*

Figure 2.5: *Convergence criteria plots.*

multiplication factor. The variance is evaluated over the last 100 iterations and the simulation is considered converged when the variance is below 10^{-10}.
For the time dependent calculations the simulation was run for at least 5 periods of the sinusoidal perturbation.

2.3.4 Numerical schemes and time stepping

Thermal hydraulic gradients are reconstructed through the default OpenFOAM Gauss linear gradient scheme, which provides second order accuracy. Convective terms in the momentum equations are discretized using the second order linear upwind scheme, reducing numerical diffusion while maintaining solution stability. The transport equation for the energy variable is discretized using the van Leer Total Variation Diminishing scheme, which offers second order accuracy in smooth regions while preventing spurious oscillations in the presence of steep gradients. Diffusive terms are approximated using a second order central differencing (Gauss linear) scheme, while linear interpolation is adopted to evaluate variables at cell faces. Since the computational mesh is orthogonal, orthogonal corrections are employed for both the Laplacian and surface-normal gradient operators.
For what regard the neutron resolution, spatial gradients are evaluated using the Gauss theorem with linear interpolation, while diffusion terms are approximated by a second order central differencing scheme (Gauss linear). Cell-face values are obtained through linear interpolation, providing an overall second order accurate spatial discretization on the orthogonal computational mesh. The balance equation for the delayed neutron precursors includes a convective term, which is discretized using a first order upwind scheme (Gauss upwind) to ensure numerical stability and preserve boundedness of the precursor concentration field. Conversely, the neutron flux equation is discretized using the second order linearUpwind scheme, which reduces numerical diffusion while maintaining a stable solution. Orthogonal surface-normal gradient evaluation is employed consistently throughout the discretization, as the computational mesh is orthogonal.
Both the steady state initialization and the subsequent transient multiphysics simulations in this study are driven by the PIMPLE algorithm, the standard solution architecture within the OpenFOAM framework for time dependent fluid dynamics. Its role is crucial as it acts as the master numerical driver that tightly couples the thermal hydraulics and the neutron kinetics. The MSFR design, ideed, exhibits extreme non-linear physical coupling, where temperature dictates fluid density (thermal expansion), which subsequently drives fluid momentum (buoyancy) and alters macroscopic cross sections (density feedback) modifying the power generation, hence a robust iterative scheme is required to prevent numerical divergence. In steady state, pseudo-transient approach is utilized: the time dependent equations are integrated over a sufficiently long physical time domain until the time derivatives of all key fields asymptotically approach zero.
To manage these highly coupled non-linearities during both the steady state convergence and the actual transients, the algorithm employs a dual loop iterative structure within each time step. The **inner Correctors** which focus specifically on resolving the pressure-velocity coupling, by solving the pressure Poisson equation

multiple times. The algorithm ensures that the continuity equation is strictly satisfied. And the **outer correctors**, which, specifies the number of times the entire system of equations is solved within one time step. This ensures that the thermal hydraulic fields and neutronic power distribution reach a converged, consistent state before advancing the simulation time.
The advancement of the temporal domain is governed by an adaptive time stepping scheme for steady state calculations and fixed time step for transient calculation in order to prevent scenarios where a rapidly shrinking adaptive time step artificially dampens a physical oscillation, or a suddenly expanding time step causes the multiphysics iterations to diverge entirely due to the extreme non-linearities. The selected time steps for transient calculation are sufficiently small to accurately resolve the characteristic time scales of the imposed perturbation while maintaining numerical stability, either by fixing the Δt or the maximum Courant number.

2.4 Benchmark Cases Considered

As previously stated, the benchmark is divided into three phases. This section provides a detailed description of the physics involved, the input data, and the observables quantities for each phase. The latter are reported as either profile along the centrelines AA' and BB' or, with regard to neutronics, as the scalar value of the reactivity derived from the effective multiplication factor k_{eff}.

2.4.1 Phase 0: Single physics verification

In this phase, as preliminary benchmark of the codes, steady state simulations are carried out without any coupling between the different physics. For this phase three steps are involed.

Step 0.1: Velocity field

The first step of the benchmark is a single physics verification of the velocity components along centerlines for a lid driven cavity flow. The objective of this step is to assess the capability of the codes to accurately reproduce the velocity field in a simple benchmark case. The only input data are the hydraulic boundary conditions.

Step 0.2: Neutronics

The second step studies the neutronics criticality eigenvalue problem with static and isothermal fuel. It aims to verfy the fission source distribution and the estimation of the effective multiplication factor. The input data are the fuel temperature

and the power used to normalize the fluxes. The output data are the effective multiplication factor and the fission rate distribution's profiles, i.e. $\int \Sigma_f \Phi \mathrm{dE}$.

Step 0.3: Neutronics

The last step of the first phase aims to verify the capability of the code to solve a passive scalar transport problem to retrive the correct temperature field in the cavity. The input data required for this step came from the previous step, in fact the fixed flow field and fixed power distribution from respectively step 0.1 and 0.2 are used. Moreover, the volumetric heat sink is implemented to simulate the salt cooling effect. The temperature distribution along the centerlines are the output data to be compared with the benchmark results.

2.4.2 Phase 1: Steady state coupling

In this phase, steady state solutions are found gradually coupling the several physics phenomena characterizing the molten salt fast system. Four steps are involved in this phase, introducing progressively coupling between the two physics, testing the flexibility of the code.

Step 1.1: Circulating fuel

The first step is aimed to assess a correct evaluation of the effects of the fluid flow on neutronics. The input of the simulation are the same of step 0.2 with the difference that the critical eigenvalue problem is solved in presence of the fixed flow field coming from step 0.1. The observable quantities are the deleyed neutron source, i.e. $\Sigma_i \lambda_i C_i$, along centerlines and the reactivity derived from the effective multiplication factor, in particular the consistency of reactivity loss due to fuel motion is verified $(\rho - \rho_{0.2})$.

Step 1.2: Power coupling

The coupling between neutronics and thermal hydraulics is investigated in the simple case of a fixed velocity field. The input of the simulation are the the fixed flow field from step 0.1 and the 1 GW power in presence of the heat sink. The goal is to analyze the flux shape deformation caused by the non-uniform fuel temperature field and the effects of this deformation, in turn, on the temperature distribution. The capability of reproducing correctly this coupling, via the fuel density feedback, is assessed in this step. The observable quantities for thermalhydraulics are just the temperature profiles along the centerlines, while for neutronics they are the reactivity change from step 1.1 and the variation of the fission rate density along AA' and BB' with respect to the solution obtained at Step 0.2, i.e. $\int \Sigma_f \Phi \mathrm{dE} - \int \Sigma_{f_{0.2}} \Phi_{0.2} \mathrm{dE}$.

Step 1.3: Buoyancy

The full multiphysics problem is analyzed in the simplest conditions, without an external source of momentum, i.e. lid velocity equal to zero. The salt flow is driven by buoyancy effects caused by the temperature gradients. The main objective is to test the codes' capability to predict the correct velocity field induced by the fission heat source and the correct reactivity introduction due to the movement of precursors. The input data are the same of previous steps unless for the fact that the flow field is not given but is calculated by the code. The output to be compared are for thermal hydraulics the velocity and temperature profiles along the centerlines, while for neutronics the reactivity change from step 0.2 and the delayed neutron source along the centerlines.

Step 1.4: Full coupling

This step is thought as representative of realistic reactor simulations in terms of physical phenomena, because it involves the full solution of the multiphysics problem. All the phenomena analyzed separately in the previous steps are introduced here simultaneously. To assess the capability of the code to correctly reproduce the coupling between the two physiscs a parametric study is performed, varying the lid velocity from 0 to 0.5, stepping 0.1 m/s, and varying the power from 0.2 to 1 GW, stepping 0.2 GW. The results are collected in terms of reactivity change from step 0.2.

2.4.3 Phase 2: Time dependent coupling

In this phase, the time dependent behaviour of the coupled system is investigated. The objective is to assess the response of the system in terms of gain and phase shift, which are the observables variables, to a perturbation in the frequency domain. This type of analysis allows characterizing the outcomes of the multiphysics codes as a function of the excitation frequency, observing in a synthetic, yet quantitative way the different physics at work and their associated dynamics. As a perturbation, is was studied how an oscillation on the heat transfer coefficient affects the power production.Starting from the initial condition, the volumetric heat transfer coefficient γ is uniformly perturbed according to a sine wave of amplitude 10% and frequency $f \in [0.0125; 0.025; 0.05; 0.1; 0.2; 0.4; 0.8]$. The variation in the cooling of the salt leads to a sinusoidal power trend induced by the negative density feedback.

Chapter 3

CNRS Benchmark results

In this chapter, the results obtained with the GeN-Foam implementation of the CNRS benchmark are presented and compared with the reference code-to-code results reported in the benchmark paper, see [12]. The comparison is performed using the same form of normalized discrepancy adopted in the reference work, although with an important limitation: only the nine tabulated points available along the benchmark centerlines are used here. Therefore, the discrepancies reported in this chapter should be interpreted as nine point estimates of the benchmark discrepancies, rather than as exact reproductions of the 201 point metrics used by the original authors. Formula for the error evaluation are reported below. At each of the 9 points, spanning from 0 to 2 m with 0.25 m step, N_c values (one per code) of a quantity Q are collected. The average value of this quantities is defined as $Q_{avg}(r_i) = \frac{1}{N_c}\Sigma_{c=1}^{N_c} Q_c(r_i)$. The discrepancy of each quantity Q_c is then calculated as

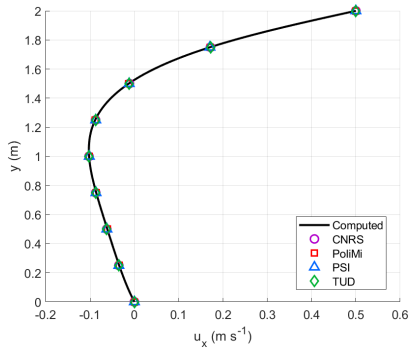
$$\epsilon_c = \sqrt{\frac{\sum_{i=1}^{9} \left(Q_{\mathrm{GeN},i} - Q_{\mathrm{avg},i}\right)^2}{\sum_{i=1}^{9} Q_{\mathrm{avg},i}^2}}. \tag{3.1}$$

The average discrepancy is then calculated as $\epsilon = \frac{1}{N_c}\Sigma_{c=1}^{N_c}\epsilon_c$.
For each benchmark step, the relevant quantities are compared either along the AA' and BB' centerlines, see fig. 2.1, or in terms of integral reactor parameters, such as the reactivity variation with respect to a reference case. Field distributions are also reported in order to support the interpretation of the pointwise comparisons. Since the original benchmark is a code-to-code verification exercise rather than a validation against experimental data, the objective of this chapter is not to demonstrate absolute accuracy, but to assess whether the GeN-Foam model reproduces the expected coupled neutronic and thermal hydraulic behaviour within the range of discrepancies observed among the reference codes, thus verifying its consistency.

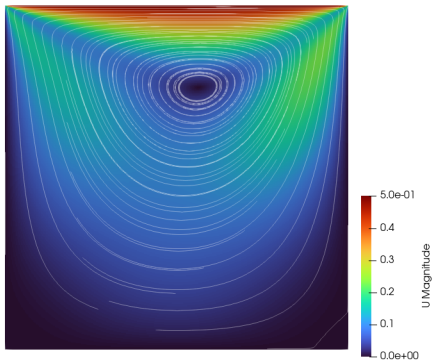
3.1 Phase 0 results

3.1.1 Step 0.1

The average relative discrepancy is 0.65% for velocity profiles along AA' and decreases to 0.53% for the BB' profiles. Excellent agreement is also observed qualitatively, as shown by the velocity component profiles and the velocity field presented in figures 3.1.



(a) *Horizontal velocity component along BB'.*



(b) *Velocity field with streamlines*

Figure 3.1: *Results from step 0.1*

3.1.2 Step 0.2

Even in this case, despite the different model used to solve the eigenvalue problem, excellent agreement is obtained, with a predicted reactivity equal to 444.6 pcm. The average discrepancy is 0.42%. Good agreement is also observed in the spatial

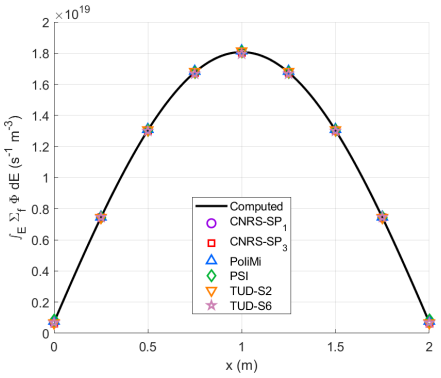
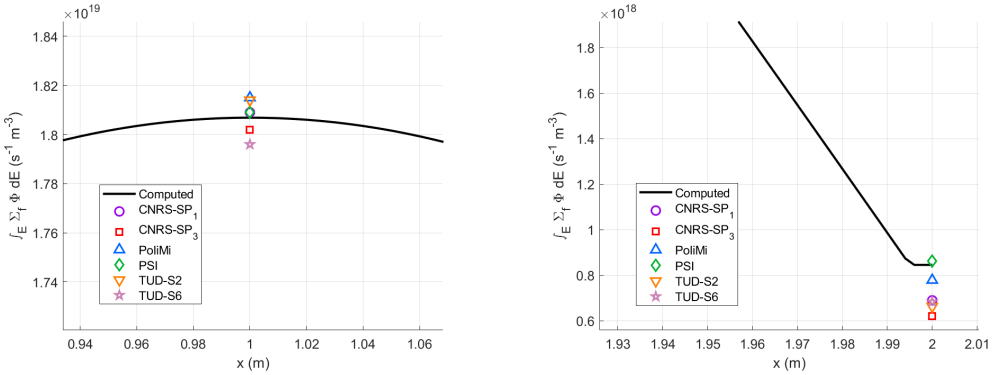


Figure 3.2: *Fission rate density along AA'.*

distributions. However, the enlarged views in figures 3.3a and 3.3bshow that the largest differences occur close to the vacuum boundaries. This behaviour is expected, since diffusion based models and transport based models treat boundary leakage differently. In the present GeN-Foam model, the vacuum condition is represented through an Albedo type boundary condition, which affects the near boundary flux shape and, through normalization, also slightly modifies the peak value. It has to be noted that peak and boundary values are closer to the PSI solution, which is consistent with the fact that PSI also used a previous version of GeN-Foam, although with a different mesh strategy.



(a) *Zoom on peak of figure 3.4.* **(b)** *Zoom on the boundary of figure 3.4*

Figure 3.3: *Results of step 0.2.*

3.1.3 Step 0.3

Excellent agreement is observed in this step. Average discrepancies along the centerlines are limited to 0.13% and 0.17% respectively for AA' and BB'. Small differences can be seen on the moving lid wall due to different mesh discretization.

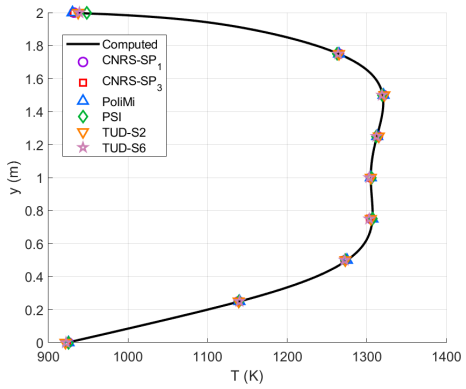
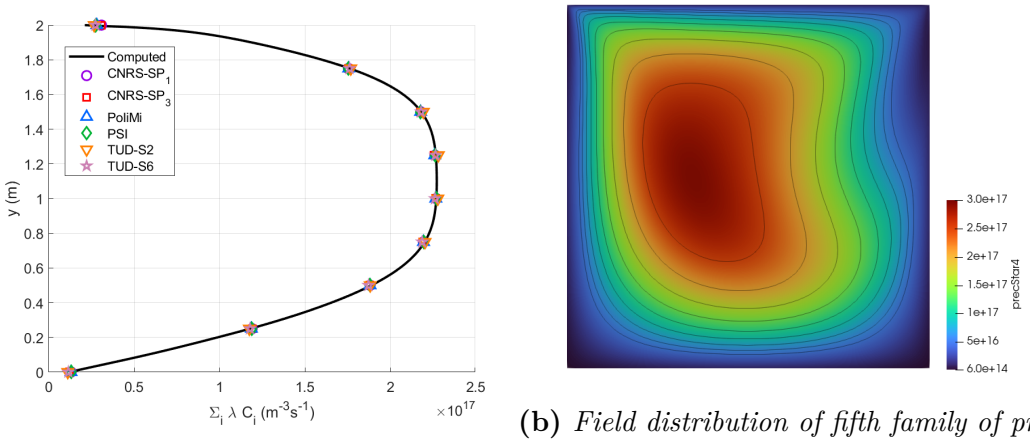


Figure 3.4: *Temperature profile along AA'.*

3.2 Phase 1

3.2.1 Step 1.1

Even for this case good agreement between results is ensured, the average discrepancy is along AA' centerline aligns with the results of other codes with a value of 0.37%. Conversely, the discrepancy along BB' is higher, ∼1%. This difference can be partly attributed to the higher sensitivity of the vertical centerline to precursor advection. The BB' line crosses regions affected by the lid driven recirculation and includes points close to the top and bottom boundaries, where the velocity field and precursor gradients are more sensitive to spatial discretization and boundary condition treatment. Since Step 1.1 uses the velocity field obtained in Step 0.1, small differences in the hydrodynamic solution may propagate into the precursor transport equation. The good agreement obtained for the reactivity defect, equal to (-61.5) pcm, indicates that the discrepancy is mainly local and does not significantly affect the integral neutronic response.



(a) *Deleyed neutron source along BB'.*

(b) *Field distribution of fifth family of precoursors.*

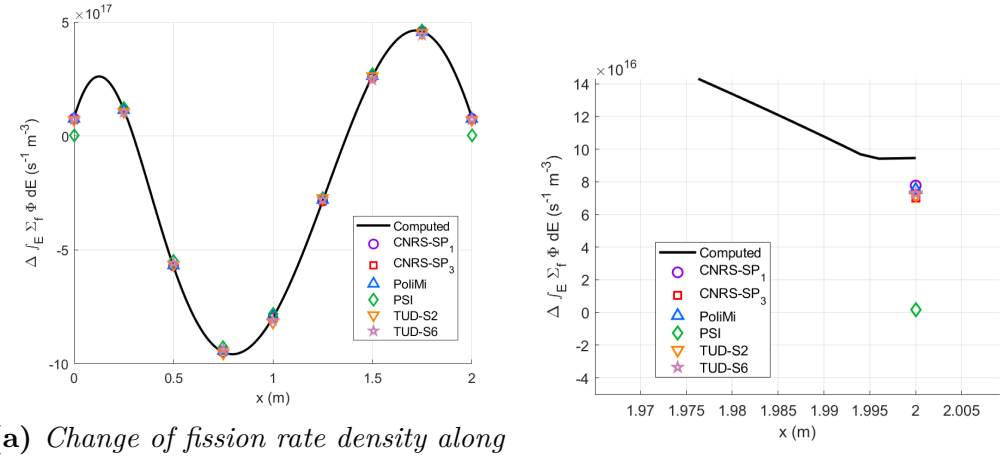
Figure 3.5: *Results of step 1.1.*

The effect of the fixed flow field can be appreciated from figure 3.5b. GeN-Foam index precoursor groups from zero.

3.2.2 Step 1.2

For the power coupling case, the average discrepancy for both centerlines does not exceeds 1.5%: the largest contribution to the discrepancy appears in the fission-rate-density variation. The larger discrepancy observed close to the boundary can be partly attributed to the propagation of the local fission rate density differences

already present in Step 0.2. Since the reported observable is defined as the difference between the coupled fission rate density and the Step 0.2 reference distribution, any boundary discrepancy in the static neutronic solution directly affects the resulting variation. This effect is amplified near the vacuum boundary, figure 3.6b, as already analyzed. In addition, Step 1.2 introduces density feedback through the temperature dependent cross sections and diffusion coefficients, which can further increase the local sensitivity of the fission rate redistribution near the edge. Nevertheless, good agreement is observed for fission rate density on both centerlines, as well as reactivity change with respect to step 1.1, with a predicted value of -1158.7 pcm. Excellent agreement in temperature profiles on both centerlines is obtained.



(a) *Change of fission rate density along AA'.*

(b) *Zoom on 2m boundary.*

Figure 3.6: *Graphs for change of fission rate density along AA'.*

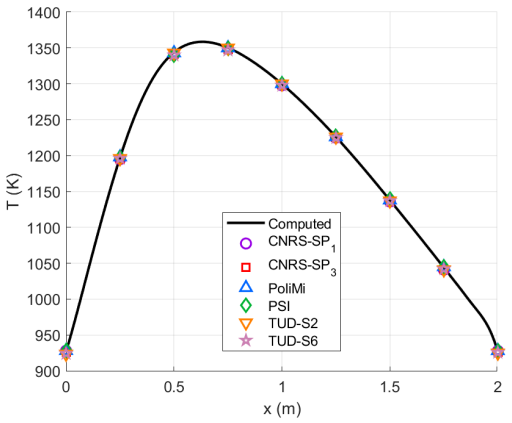
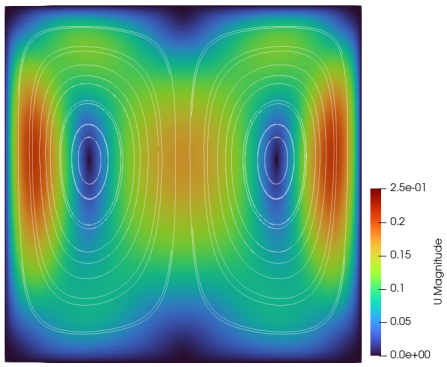


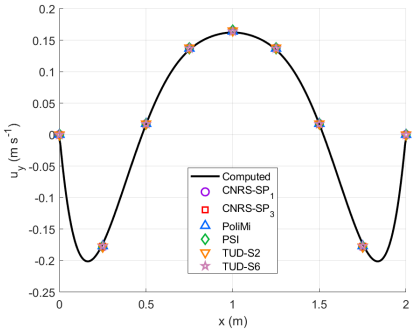
Figure 3.7: *Temperature profile along AA'.*

3.2.3 Step 1.3

Very good agreement is obtained for the buoyancy case with an average discrepancies of 0.71% and 0.46% respectively for AA' and BB'. Graphs and field distributions are reported below.



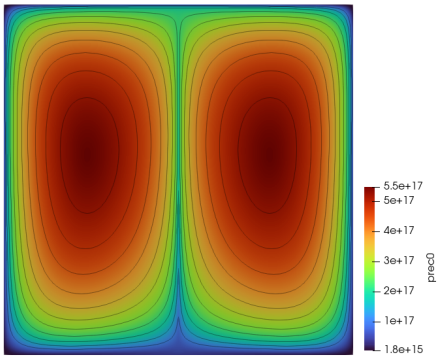
(a) *Velocity field with streamlines.*



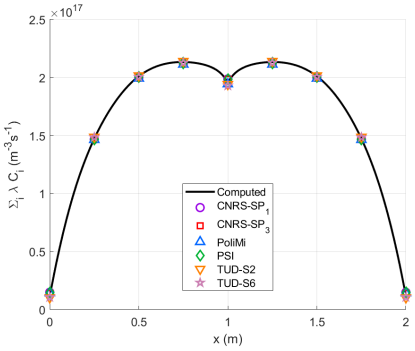
(b) *Vertical component of velocity profile along AA'*

Figure 3.8: *Velocity graphs.*

Even in this case the discrepancies along centerlines of delayed neutron source result minimal with values of 0.25% and 0.5% respectively for BB' and AA'. Further confirmations of the solution's accuracy were provided by the reactivity change computed, which yielded a result of -1222.8 pcm.



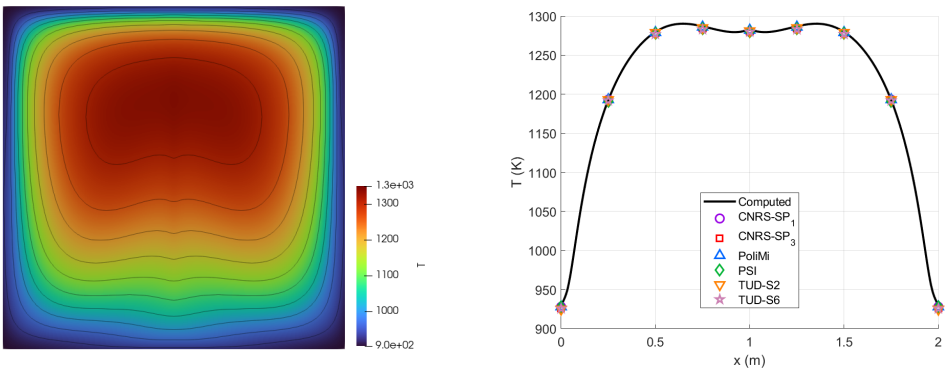
(a) *First family of deleyed neutron precursors field with isolines.*



(b) *Deleyed neutron precursors source profile along AA'.*

Figure 3.9: *Delayed neutron precursors graphs.*

Discrepancies for temperature drops below 0.2% for both profiles.



(a) *Temperature field with isolines.* **(b)** *Temperature profile along AA'.*

Figure 3.10: *Temperature graphs.*

3.2.4 Step 1.4

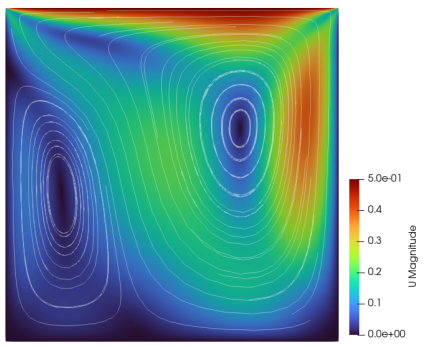
The reactivity change with respect to Step 0.2 is reported in Table 3.1, for each reactor power and lid velocity considered. As expected, higher power levels cor-

Table 3.1: *Reactivity change from Step 0.2 as a function of P and U_{lid}.*

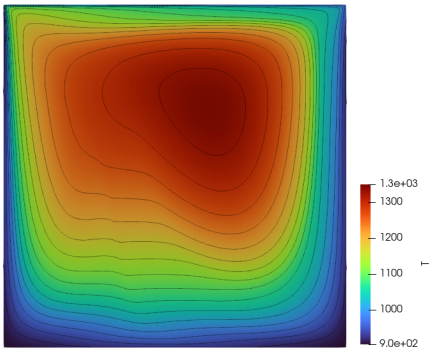
U_{lid} (m s^{-1})	$\rho - \rho_{s0.2}$ (pcm)				
	P (GW)				
	0.2	0.4	0.6	0.8	1.0
0.0	-270.22	-505.95	-737.43	-980.57	-1224.23
0.1	-272.00	-506.23	-741.90	-979.86	-1223.25
0.2	-273.51	-506.47	-740.70	-970.80	-1220.01
0.3	-273.93	-506.56	-739.31	-968.54	-1215.71
0.4	-278.22	-506.53	-735.72	-965.43	-1210.43
0.5	-280.10	-507.02	-734.02	-969.58	-1208.20

respond to more significant insertions of negative reactivity, due to the increased average temperature in the system. The lid velocity has a dual effect on reactivity, depending on the power level. At low power, higher lid velocities correspond to higher negative reactivity differences, whereas the effect is reversed from P= 0.6 GW. In fact, the more intense is forced convection, the more precursors and energy get redistributed in the cavity, and this influences reactivity via two competing mechanisms. Shifting the peak of the precursors concentration further from the cavity center introduces more negative reactivity, with respect to the case *Ulid* = 0, whereas shifting the temperature pick introduces positive reactivity as the temperature gets lower in the region of higher importance. According to the power

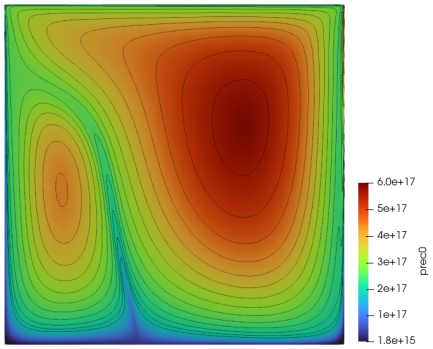
level, one effect prevails on the other, because at lower power the effect of the temperature redistribution is less significant. Results are satisfactory, the average discrepancy with other codes is 0.97%. Significant field figures are reported below.



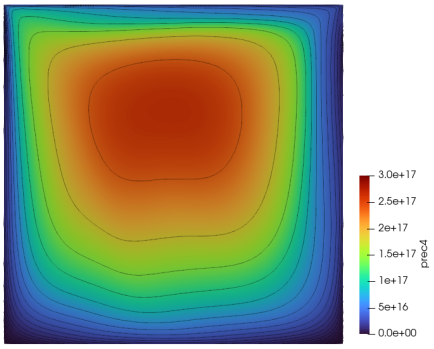
(a) *Velocity field with streamlines.*



(b) *Temeprature field with isolines.*



(c) *First family of delayed neutron precursor field with isolines.*



(d) *Fifth family of delayed neutron precursor field with isolines.*

Figure 3.11: *Results of steady state coupling.*

3.3 Phase 2

Gain and phase shift were evaluated after the system response reached an asymptotic behavior at all frequencies.

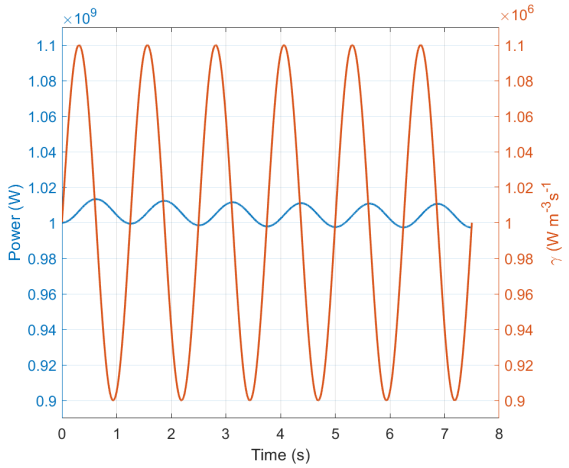
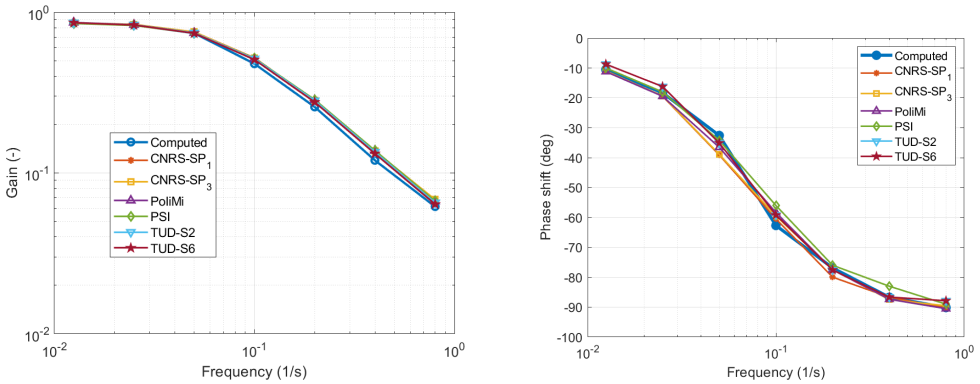


Figure 3.12: *Power and γ oscillations for f=0.8 Hz.*

Good agreement is found among the codes, as the average discrepancy predicted is below 2.8%. The time step adopted is the same of PoliMi and as can be seen from figure 3.13 the main discrepancies are related to frequencies 0.1, 0.2 and 0.4 Hz where the time step is the same. Improvements may be obtained by decreasing the Δt values for these simulations, nevertheless the results are satisfactory.



(a) *Power gain bode graphs.* **(b)** *Phase shift bode graphs.*

Figure 3.13: *Bode diagrams of power gain and phase shift as a function of the frequency of the γ wave.*

From a physics point of view, the system dynamic response is as expected. At low frequencies, the power follows the variation of the extracted one (i.e., gain ≈ 1 and minimal phase shift), as precursors approach an equilibrium during the slow transient. At high frequencies, on the contrary, the contribution of the delayed neutrons to the entire neutron population is filtered, as some precursors families do not 'perceive' the temperature reactivity feedback due to their long half life; so, the system response amplitude decreases (i.e., gain less than one). Moreover, the phase shift approaches -90°; in fact, when γ reaches a maximum, the temperature time derivative is at its maximum (at high frequencies, the time derivative term in the energy equation is dominating over the convective and diffusive ones), and so is the reactivity change and the derivative of the neutrons population, which shifts to a cosine wave.

3.4 Conclusions

Summary of results is reported in table 3.2.

Table 3.2: *Summary of the main discrepancies obtained in the CNRS benchmark.*

Case	Main observable	Average discrepancy (%)
Step 0.1	Velocity profiles	0.59
Step 0.2	Fission rate density and reactivity	0.42
Step 0.3	Temperature profiles	0.15
Step 1.1	Delayed neutron source and reactivity defect	0.68
Step 1.2	Temperature and fission rate variation	1.43
Step 1.3	Velocity, temperature and delayed neutron source	0.59
Step 1.4	Reactivity defect	0.97
Phase 2	Gain and phase shift	2.8

Overall, the results obtained in Chapter 3 show that the GeN-Foam model is able to reproduce the main single physics and coupled behaviours of the CNRS benchmark. The discrepancies remain within the range expected from the reference code-to-code comparison and are mainly associated with boundary treatment, mesh discretization effects, and transient time step resolution. The agreement obtained for the reactivity variations is particularly important, since these quantities represent integral indicators of the coupled neutronic and thermal hydraulic response. Therefore, the benchmark provides sufficient confidence in the GeN-Foam model to use it as the high fidelity core solver in the co-simulation framework developed in the following chapter.

Chapter 4

GeN-Foam-OpenModelica co-simulation

The CNRS benchmark verifies the capability of GeN-Foam to reproduce the coupled neutronic and thermal hydraulic behaviour of the molten-salt cavity. The following step is to extend the model toward a multi-scale framework, where the core is solved by GeN-Foam and the external/system-level response is represented in OpenModelica.
In this chapter the motivations for the co-simulation are explained in section 4.1. Section 4.2 aims to explain the coupling architecture of the numerical model defining the algorithm behind the co-simulation. The limitations of the model and the discussion about the coupling accuracy are presented in Section 4.3. Once the coupling setup has been defined, the co-simulation interface is verified, in section 4.4, by reproducing one transient case from Section 3.3 using a γ signal generated in OpenModelica.
The final aim of the co-simulation framework is to analyse accident scenarios and emergency cooling strategies in Chapter 5. In the present chapter, the objective is limited to the verification of the data exchange between GeN-Foam and OpenModelica.

4.1 Motivation for Co-simulation

The previous chapters have focused on the verification of the GeN-Foam model against the CNRS benchmark. However, while the high fidelity code provides a detailed spatial description of the reactor core, the analysis of plant level transients and control strategies requires the representation of additional system dynamics, such as actuation laws, feedback controllers, simplified external circuits, and safety-related systems. For this reason, a co-simulation approach is introduced in this

chapter, coupling the high-fidelity multiphysics core model developed in GeN-Foam with a system-level model implemented in OpenModelica.
OpenModelica (OM) is an open source modelling and simulation environment based on the Modelica language, mainly intended for the analysis of complex physical and engineering systems. Modelica is an equation based, object oriented language that allows different components to be described through their physical relationships rather than through an explicit sequence of computational instructions. This makes this tool particularly suitable for system-level dynamic simulations, where thermal hydraulic components, control systems, sensors, actuators, and feedback logic can be represented in a modular way.
This approach allows the intrinsic multiphysics response of the core to be retained, while enabling the flexible modelling of control and emergency cooling strategies outside the CFD/neutronic solver environment. Therefore, the purpose of the present set up is not to replace the GeN-Foam model, but to extend its applicability toward transient system-level analyses in which the interaction between core dynamics and external control actions plays a central role.
At the present stage, the OpenModelica model does not represent a detailed physical secondary circuit. Instead, the action of the external cooling system is represented through the effective volumetric heat-transfer coefficient. The resulting model should therefore be interpreted as a reduced-order representation of the cooling action rather than as a complete plant model. This is consistent with the objective of the thesis, which aims to test the coupling state of the two codes.

4.2 Co-simulation architecture

As reported in the user guide, [16], GeN-Foam provides several interface points to communicate with **Functional Mock-up Units** (FMUs). FMUs are containers of software and data that are based on a widely employed communication standard called **Functional Mockup Interface** (**FMI**). The FMI is a free standard that defines a container and an interface to exchange dynamic simulation models using a combination of XML files, binaries and C code.
Using FMI, GeN-Foam and OpenMdelica are run as independent solvers and exchange a limited number of scalar quantities at prescribed communication times. GeN-Foam solves the coupled neutronic and thermal hydraulic problem inside the reactor domain, including neutron diffusion, delayed neutron precursor transport, energy conservation and thermal feedback. From this solution, the total reactor power $P(t)$ is evaluated and transferred to OpenModelica. The OpenModelica model receives this signal and computes the value of the volumetric heat transfer coefficient γ(t), which represents the action of the external cooling or control system.

The file to be added to the standard GeN-Foam architecture are:

- a Python coupling driver, called through the **controlDict**, which coordinates the advancement and data exchange between the two solvers;
- a JSON configuration file defining the correspondence between the GeN-Foam and FMU variables;
- the .fmu generated from the OM model.

The **fvOptions** file is modified in order to get the γ from Modelica and is reported in Appendix B.2.

The coupling between GeN-Foam and OpenModelica is performed through a staggered explicit exchange of information. At the beginning of each communication interval, GeN-Foam advances the multiphysics solution using the current value of the volumetric heat transfer coefficient γ. Once the GeN-Foam time step, or group of internal time steps, has been completed, the total reactor power is evaluated and passed to OpenModelica. The OpenModelica model then updates the system-level equations and computes the new value of γ, according either to a prescribed perturbation law or to the control strategy implemented for the emergency cooling system. This updated value is finally transferred back to GeN-Foam and used during the following communication interval.

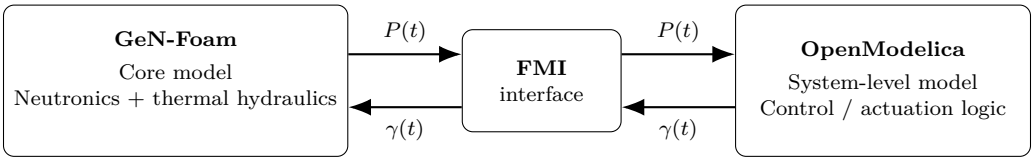


Figure 4.2: *General co-simulation architecture adopted for the coupling between GeN-Foam and OpenModelica through the FMU interface.*

In the verification case presented in this chapter, the coupling is used in a simplified one-way form, in which OpenModelica prescribes γ(t), while $P(t)$ is used only for post-processing and comparison. The full two-way architecture is employed in Chapter 5. For both cases the mesh utilized in GeN-Foam is the same of the one used to report the results in previous calculation.

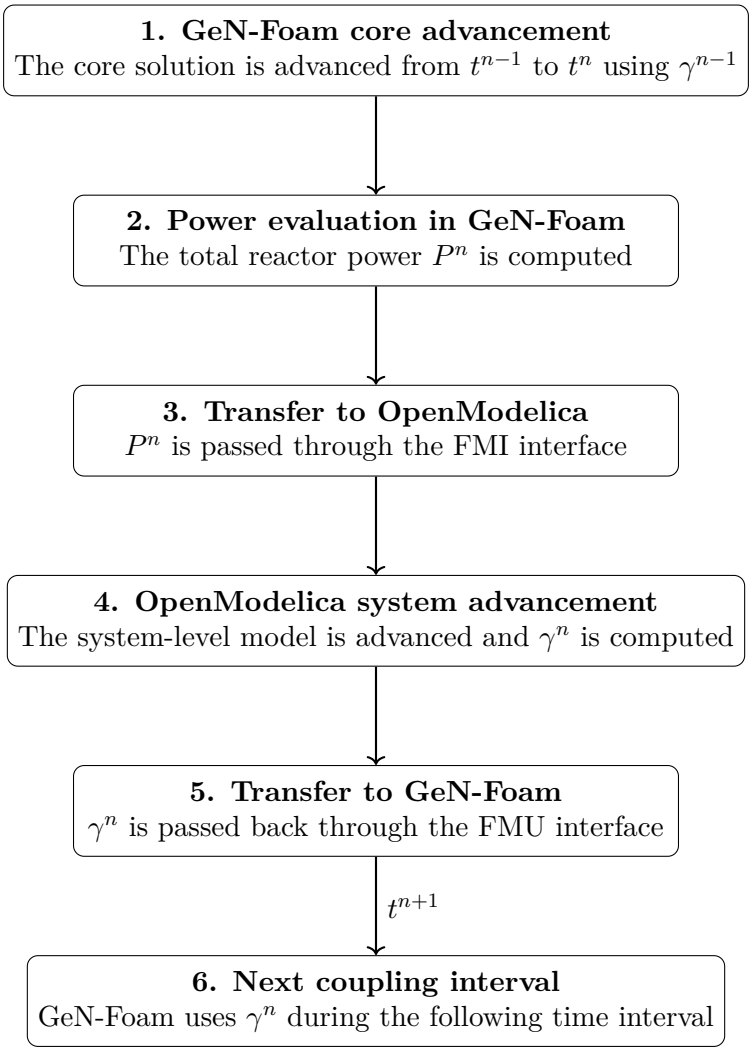


Figure 4.1: *Explicit staggered coupling algorithm adopted for the GeN-Foam/OpenModelica co-simulation.*

4.3 Discussion of coupling accuracy and limitations

In the present implementation, no internal iterations are performed between the two codes within the same communication step. Therefore, the coupling, at co-simulation level, is explicit and the value of γ used by GeN-Foam over a given interval is based on the information available from the previous exchange, as described in [**Schweizer2015Cosimulation**]. This approach is simple and computationally efficient, but it introduces a coupling delay that must be kept small with respect to

the characteristic time scale of the transient under investigation. For this reason, the communication time step has to be selected carefully, especially when fast variations of the cooling coefficient are imposed.
For the closed-loop simulations presented in Chapter 5, the GeN-Foam solution was advanced using adaptive time stepping. The time-step size was automatically adjusted to limit the maximum Courant number to $Co_{\max} = 1$. The latter is a non-dimensional parameter that defines the stability of a numerical scheme, as described in [17]. This approach allows the solver to reduce the time step when the velocity field or its spatial gradients become more demanding, while avoiding unnecessarily small time steps during the slower portions of the transient.
The quantities exchanged through the FMU interface are scalar quantities, such as local probe values, volume-averaged quantities, or domain-integrated variables. Consequently, the OpenModelica model does not have access to the full spatial fields computed by GeN-Foam. Spatially resolved control cannot be performed directly unless suitable scalar indicators are explicitly defined and exchanged, for example temperatures at selected points, regional averages, or maximum values over the domain. The exchanged variables for verification case are then reported in table 4.1

Table 4.1: *Variables exchanged through the FMU interface in the co-simulation verification case.*

Variable	Direction	Unit	Meaning
$\gamma(t)$	OpenModelica → GeN-Foam	W m^{-3} K^{-1}	Volumetric heat-transfer coefficient
$P(t)$	GeN-Foam output	W	Total reactor power
t	Coupling driver	s	Communication time

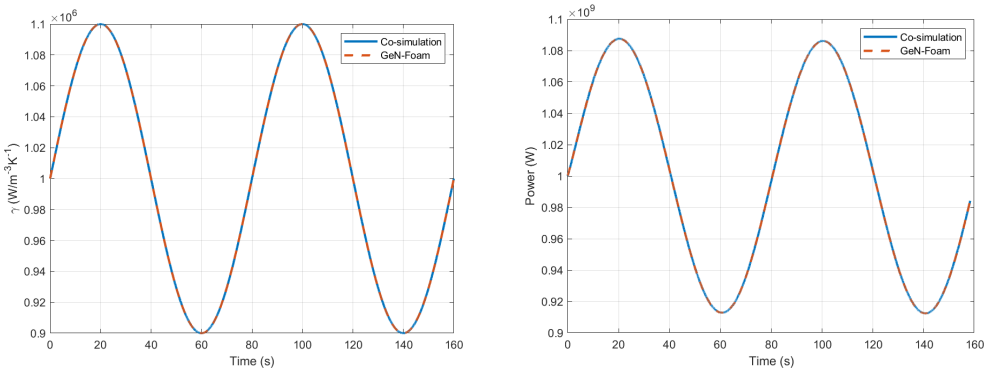
4.4 Verification case

In the verification case considered in this chapter, the co-simulation interface is tested in a simplified one way configuration. OpenModelica is used to generate the prescribed sinusoidal evolution of the volumetric heat transfer coefficient, i.e.

$$\gamma(t) = \gamma_0 \cdot (1 + 0.1 sin(2\pi f t)) \quad \text{with } \gamma_0 = 10^6 \ Wm^{-3}K^{-1} \tag{4.1}$$

which is passed to GeN-Foam through the FMU interface. The reactor power computed by GeN-Foam is recorded and compared with the previous computed reference solution, but it is not fed back to OpenModelica. Therefore, this test verifies the correct transmission and application of the $\gamma(t)$ signal, while the closed loop exchange between reactor power and cooling action is introduced only in the emergency cooling analysis of Chapter 5. In this first test, only the frequency

f=0.0125 Hz is considered. This frequency corresponds to the slowest transient of the benchmark and is therefore suitable as an initial verification of the coupling procedure. Comparison of results is presented in figure 4.3.



(a) *Comparison of γ time profile.* **(b)** *Comparison of power time profile.*

Figure 4.3: *Comparison between solutions using co-simulation and GeN-foam.*

Both γ and power profile overlap on the time period considered almost perfectly, a local difference between the two signals can be observed, however is negligible compared with the perturbation period considered as the effect produced on the power.

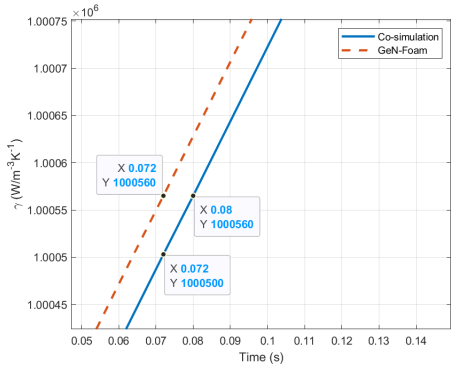


Figure 4.4: *Enlargement of figure 4.3a*

The enlargment in figure 4.4 show exactly the limitation preseted in Section 4.3. An offset can be observed between the γ signal imposed directly in GeN-Foam and the one exchanged through the FMU interface. The difference at each time instant with respect to total γ value is 0.001%, which is judged negligible. The γ value prescribed by the FMU interface is the same of the one prescribed by GeN-Foam but shifted of exactly one time step, 8×10^{-3} s. The evidence presented herein

indicates that the observed lag is not related to the physical reactor response, but rather to a numerical effect introduced by the explicit staggered exchange of data.

To quantify the agreement between the result of Section 3.3 and co-simulation case, the relative error with respect to GeN-Foam only simulation are computed for both gain and phase shift. The two relative errors are 0.13% and 0.34% respectively for gain and phase-shift.
The agreement obtained in this one way verification confirms that the OpenModelica generated cooling coefficient is correctly received and applied by GeN-Foam. The next chapter extends the same interface to a two way coupling, in which the power calculated by GeN-Foam is used by OpenModelica to determine the emergency cooling response.

Chapter 5

Emergency cooling system model and transient analysis

This chapter investigates an idealised control strategy for compensating for a partial degradation of the heat-removal capability. Starting from the fully coupled steady state introduced in Section 3.2.4 a transient co-simulation is run in order to define the parameters to be adopted for a closed feedback loop. The sections are divided in the following manner: in Section 5.1 the scenario simulated is explained with the hypothesis and physical reference to the phenomena involved. In section 5.2, the numerical model constructed in OpenModelica is presented in conjunction with the modelling of the source of disturbance. Section 5.3 analyses the influence of the PI-controller parameters and identifies a suitable tuning for the selected reference transient. Finally the discussion of the results obtained with a brief discussion of the effects of the transient on the reactor variables.

5.1 Physical interpretation of the cooling system

The scenario taken into account considers that during the normal operations, i.e. steady state of Section 3.2.4, the reactor equilibrium is perturbed by a generic failure in the cooling system. The latter produces a decrease of 30% of the cooling efficiency, i.e. the volumetric heat transfer coefficient. The transient behaviour of the reactor is studied and a backup cooling strategy is proposed.
This approach proposes to treat γ, instead of a constant value, as done in the CNRS Benchmark, as the actuation variable of an idealized external cooling system. As already described in Section 3.3, a loss of cooling capability causes the salt temperature to increase. In a liquid fuel MSR, this temperature increase affects the neutronic behaviour through density feedback: as the fuel salt expands, the fuel density decreases, modifying the macroscopic cross sections and introducing

negative reactivity. As a consequence, the reactor power tends to decrease.
Result of this transient without no controls applied are reported in figure 5.1. The emergency cooling system is therefore introduced to counteract the thermal

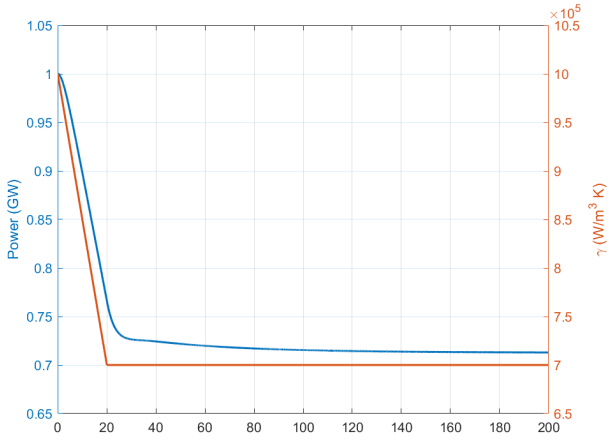


Figure 5.1: *Transient behaviuor in absence of controls.*

excursion by increasing the heat removed from the domain. However, the interaction between cooling and power is not purely thermal: by reducing the fuel temperature, an increase in γ may also partially recover reactivity and affect the power evolution. For this reason, the performance of the emergency cooling action must be assessed by considering both the thermal response and the neutronic power response.
The model proposed in section 5.2 is not constructed as a detailed heat-exchanger or secondary-loop geometry. Instead, it is represented at system level in OpenModelica through a prescribed or controlled evolution of $\gamma(t)$, which is then transferred to GeN-Foam through the FMU interface. This modelling choice is consistent with the simplified nature of the CNRS benchmark and allows the effect of different cooling actuation strategies to be studied without introducing additional geometrical and thermal hydraulic complexity. Therefore, the model should be interpreted as an idealized emergency heat-removal action rather than as a detailed design of a real safety system. Two main hypothesis are introduced with the model:

- **continuous acting control**;
- **no control on temperature**

The first assumption is that the backup system does not need to be triggered, whenever power deviates from 1 GW reference value the system "adjust" the γ to restore the steady state. The other states that the variables exchanged through the FMI are power and heat transfer coefficient, then no control on temperature is operated, the change in the field are consequences of power and γ, i.e. no

information on temperaturea are exchanged through the FMI.
The objective of the following analysis is to evaluate how the system response depends on the characteristics of this emergency cooling action. In particular, the influence of the controller parameters is assessed in terms of reactor power evolution, maximum temperature, and settling behaviour. These quantities provide an indication of the effectiveness and stability of the idealized emergency cooling strategy.

5.2 OpenModelica control model

The model implemented represents the control logic of an auxiliary heat-removal system. The model receives as input the total reactor power calculated by GeN-Foam and returns the effective volumetric heat-transfer coefficient.

The value of gamma produced at each time step by this architecture can be

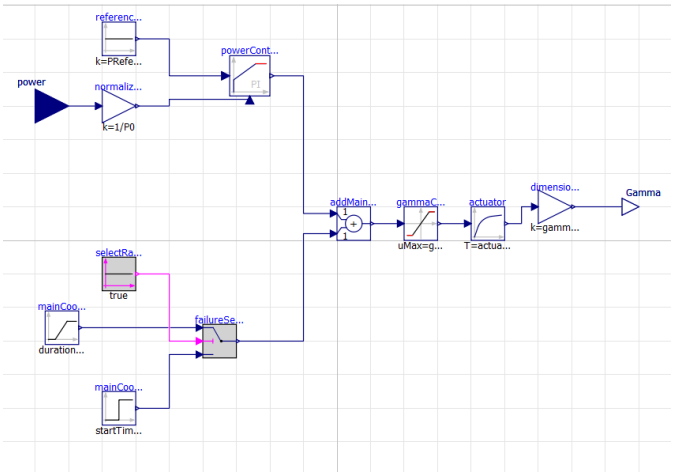


Figure 5.2: *Block diagram of emergecy cooling model.*

summurized as:

$$\gamma(t) = \gamma_0 + \Delta\gamma_{disturbance}(t) + \Delta\gamma_{backup}(t)$$

The contribution of disturbance, i.e. the degradation of the main cooling system, is imposed using a Modelica source block as cen be seen in the lower part of figure 5.2.
Part of the backup cooling contribution come instead from upper part of the block diagram where the input reactor power is normalized with respect to the nominal power P_0 and compared with a prescribed reference value. The resulting error is processed by a limited PI controller, which commands the contribution. The anti-windup mechanism helps the backup command decrease correctly after

saturation or after the main system recovers. The total cooling command is obtained by summing the degraded main cooling availability and the controller requested backup contribution. This signal is then limited between prescribed minimum and maximum values and passed through a first-order actuator to account for the finite response time of the physical cooling system. Finally, the relative command is multiplied by the nominal coefficient γ_0, producing the dimensional value of γ sent back to GeN-Foam. In this way, the model simulates the response of a controlled heat removal system that attempts to compensate for a partial loss of cooling and recover the nominal power level.

5.2.1 Disturbance modeling

For what regards the disturbance modeling two types of actuation are considered: step and ramp variation. A step variation is used as a limiting case to evaluate the immediate response of the coupled reactor model to an abrupt change in cooling capability. Although useful for sensitivity analysis, this approach is not fully representative of a physical system, since real actuators such as pumps, valves, or auxiliary heat exchangers cannot modify the heat-removal rate instantaneously. For this reason, ramp variation is introduced to represent a finite activation time of the emergency cooling system. The ramp approach provides a more realistic description of a failure in the secondary that leads to a progressive degradation of the cooling capacity. For this reason is therefore adopted for the parametric study. For sake of completness, the core response in terms of power is reported in figure 5.3 for both step and ramp variation. Both cases are simulated with the same parameters of backup system. For what regards the parameters of the failure contribution, the failure fraction, i.e. the final height of the blocks, is set to -30% of reference value. For ramp, the duration was set equal to 20 s.

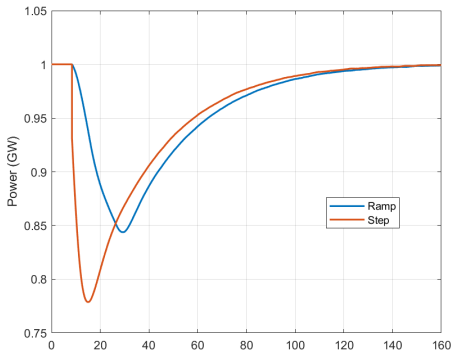


Figure 5.3: *Power response comparison between the step and ramp source of disturbance.*

It is important to highlight that neither the failure fraction nor the ramp duration are intended to represent a specific designed component, since no detailed heat-exchanger or emergency-cooling loop is modelled in the present work. Instead, they are introduced as fixed parameters to allow the parametric study of the parameters of the backup contribution.
The full set of parameters used for the following studies can be retrived from code in appendix B.3.

5.3 Parametric study

The parametric study is focused on the PI controller parameters rather than on the full set of accident-scenario variables. This choice is motivated by the objective of this part of the work, which is to assess the influence of the control law on the coupled GeN-Foam/OpenModelica response once a representative cooling degradation scenario has been defined. The failure fraction and the ramp duration determine the severity and time scale of the imposed disturbance, whereas the PI parameters determine how the emergency cooling system reacts to that disturbance. Varying all these quantities simultaneously would make the interpretation of the results less clear, since changes in the reactor response could be caused either by a different accident severity or by a different controller tuning. Therefore, the failure scenario is kept fixed, using the same values of section 5.2.1, and the parametric study is restricted to the controller parameters, allowing the effects of PI actions on stability, overshoot, settling time and cooling effort to be isolated.
This approach does not provide a complete sensitivity analysis of all possible accident conditions. Rather, it represents a first controller-tuning study for a selected reference transient. The extension of the analysis to different failure fractions, ramp durations, and cooling-system capacities is left as a possible future development.
The requrirements to be satisfied by the closed loop feedback control are:

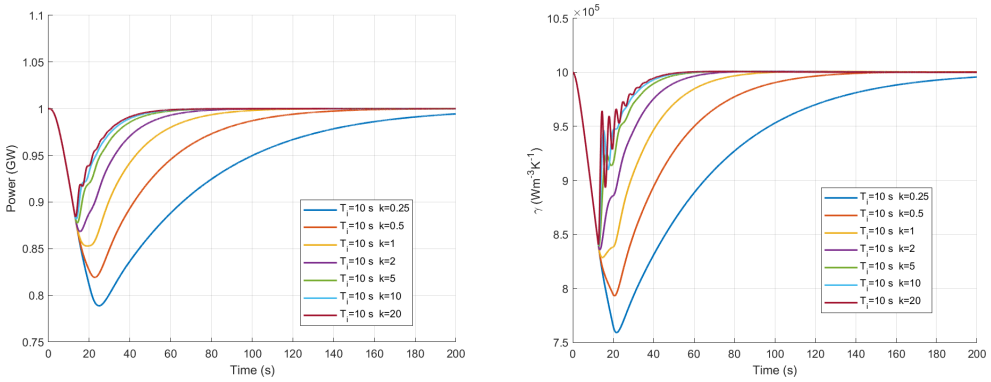
- limited deviation from the nominal reactor power;
- short settling time
- null or minimal overshoot
- absence of sustained oscillations;
- limited variation and rate of change of γ

The PI variables on which parametric study is performed are the **gain** (k) and the **integration time** (T_i). One parameter is fixed and the other is varied. Once the separate influence is analysed, the influnce of the total integral gain of the PI,

defined as $K_i = k/T_i$, is investigated to define the final parameters for the closed loop.

5.3.1 Study on the gain

The integration time selected for this part of the parametric study is $T_i = 10s$. The values of the gain tested span from 0.25 to 20. The corresponding reactor-power and heat-transfer-coefficient histories are shown in Figures 5.4



(a) *Power response of the reactor for different values of k.*

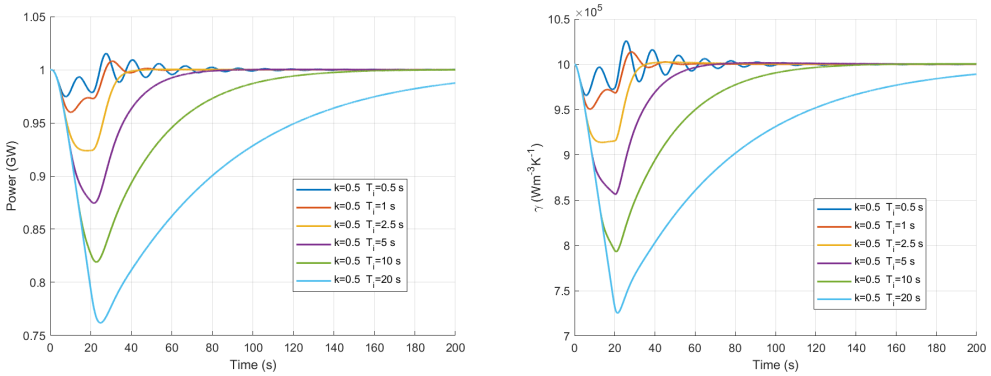
(b) *Volumetric heat transfer coefficient for different values of k.*

Figure 5.4: *Power and γ beahviour for different values of k.*

As expected increasing the gain strengthens the immediate response of the controller and reduces the settling time. However, for $k \gtrsim 5$, the cooling command becomes increasingly oscillatory for the selected integral time. These oscillations are also transmitted to the reactor power, indicating that the controller is becoming excessively aggressive for the selected integration time.
No appreciable positive power overshoot is observed for the configurations considered, while the lowest power reached is approximately 789 MW.

5.3.2 Study on the integration time

The gain selected for this part of the parametric study is $k = 0.5$. The values of the integration time tested span from 0.5 to 20 s. Results are reported in figure 5.5



(a) *Power response of the reactor for different values of T_i.*

(b) *Volumetric heat transfer coefficient for different values of T_i.*

Figure 5.5: *Power and γ beahviour for different values of T_i.*

Even in this case the behaviour follows the expections. increases the integral gain and causes the accumulated power error to be corrected more rapidly. The settling time is consequently reduced. However, for $T_i < 2.5$ s, the γ response becomes oscillatory and the reactor-power response becomes underdamped. Conversely, large values of T_i produce a smooth but slow recovery and allow a larger power deviation. The lowest power reached among the tested cases is approximately 762 MW.
An intermediate integral time is therefore preferable because it provides sufficiently rapid error correction without producing the oscillations associated with excessive integral action.

5.3.3 Combined influence of k and T_i

The separate studies identify $k = 0.5$ and $T_i = 2.5$ s as a suitable reference configuration. For this case,

$$K_i = \frac{k}{T_i} = 0.20 \text{ s}^{-1}.$$

As shown in Figure 5.6, the power deviation remains below approximately 8%, no significant power overshoot is observed, and the system returns close to its nominal condition in approximately 50 s. A small overshoot is present in the heat-transfer coefficient, but it is rapidly damped.
Additional combinations of k and T_i are then investigated, as reported in Table 5.1. The overall comparison shows that for high values of K_i, which means either

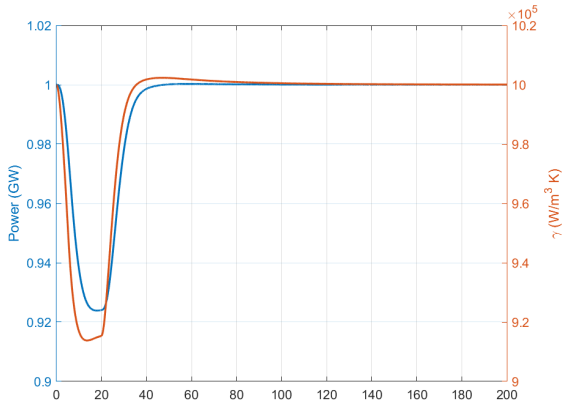
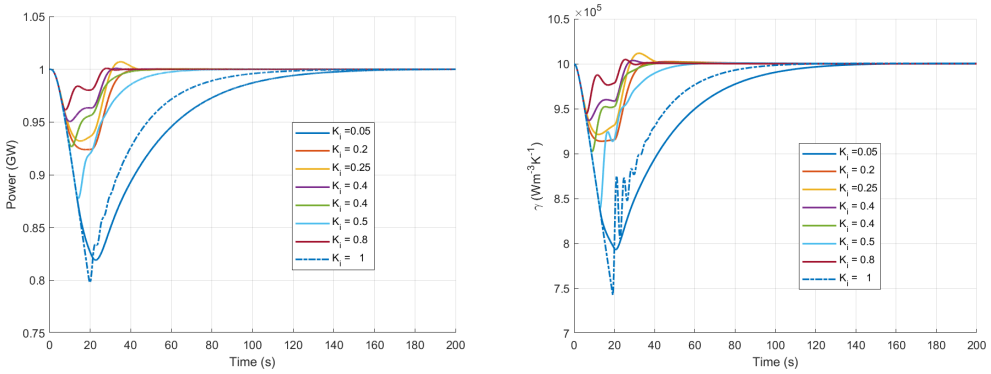


Figure 5.6: *Reference solution for K_i parametric study*

Table 5.1: *PI-controller parameter combinations considered in the parametric study.*

k	T_i (s)	$K_i = k/T_i$(s^{-1})
0.50	10.0	0.05
0.50	2.50	0.20
0.25	1.00	0.25
1.00	2.50	0.40
2.00	5.00	0.40
5.00	10.00	0.50
2.00	2.50	0.80
20.0	20.0	1.00

high value of k or low value of T_i, even if the power deviation might be minimal, γ



(a) *Power response of the reactor for different values of K_i.*

(b) *Volumetric heat transfer coefficient for different values of K_i.*

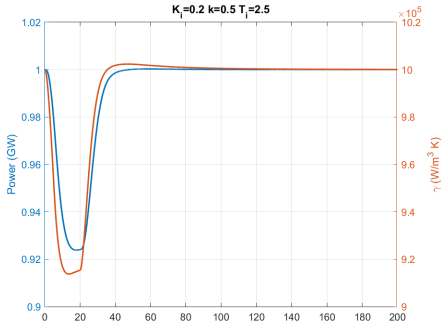
Figure 5.7: *Power and γ beahviour for different values of K_i.*

present a really high rate of change that migth also produce oscillations in power. Conversely for too low values of K_i, the low gain value or the high integration time produce a stable γ output, and in turn power response, however the power deviation is much higher and the settling time is considerably longer with respect to other combinations. Configuration characterized by $K_i = 5\ \mathrm{s}^{-1}$ is discarded due to the higher power deviation compared to other combinations.
The remaining combinations are then compared in figures 5.8 This comparison demonstrates that K_i alone is not sufficient to characterise the controller response. The two configurations with $K_i = 0.40$ produce different transients because the proportional gain and the integral time affect different parts of the control action. The proportional term determines the immediate response to the power error, whereas T_i determines the rate at which the accumulated error contributes to the command.

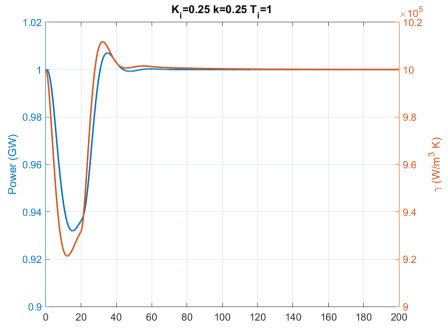
Among the tested configurations, the combination

$$\boxed{k = 1.0, \qquad T_i = 2.5\ \mathrm{s}, \qquad K_i = 0.40\ \mathrm{s}^{-1}}$$

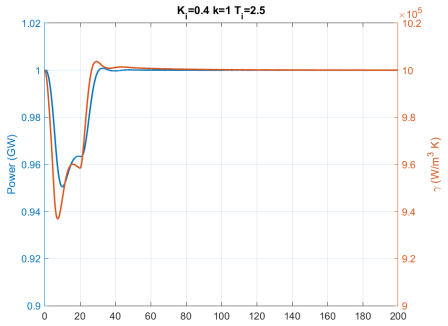
provides the most favourable compromise according to the adopted qualitative criteria. It produces the smallest power deviation among the four retained cases, recovers the nominal condition rapidly, and does not exhibit sustained oscillations or a significant power overshoot. The corresponding γ variation remains bounded, although its two stage evolution reflects the end of the imposed 20 s degradation ramp. The selected tuning should not be regarded as a universal optimum. It is conditional on the 30% cooling degradation, the 20 s ramp duration, the actuator time constant, the imposed saturation limits, and the explicit co-simulation communication step. A rigorous optimisation would require quantitative performance indices, such as the integral of the absolute error, maximum power and temperature



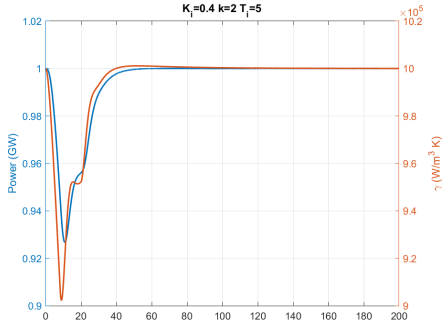
(a) *Power and* γ *beahviour for* $K_i = 0.2$ s^{-1}.



(b) *Power and* γ *beahviour for* $K_i = 0.25$ s^{-1}.



(c) *Power and* γ *beahviour for* $K_i = 0.4$ s^{-1}.



(d) *Power and* γ *beahviour for* $K_i = 0.4$ s^{-1}.

Figure 5.8: *Power and* γ *beahviour for different values of* T_i.

deviations, settling time, overshoot, and the maximum rate of change of γ, together with a sensitivity analysis over a wider set of transients.

5.4 Effects on reactor variables

The assumption of excluding temperature from the controller input is assessed a posteriori using the thermal-hydraulic results obtained from GeN-Foam. Figure 5.9 shows the evolution of the reactor power and of the maximum fuel-salt temperature for the controller configuration $k = 0.5$ and $T_i = 20$ s, which produces the largest power deviation among the investigated tunings.

Despite the significant reduction in reactor power, the maximum temperature increases gradually and reaches a peak variation of approximately $\Delta T_{\max} = 14.4$ K. This corresponds to about 1.1% of the initial maximum temperature. The limited thermal excursion results from the large thermal inertia of the fuel salt, together with the reduction in fission power caused by the negative temperature-reactivity feedback. Therefore, for the reference transient considered in this work, the use

of reactor power as the only feedback variable provides an acceptable simplified representation of the system response.

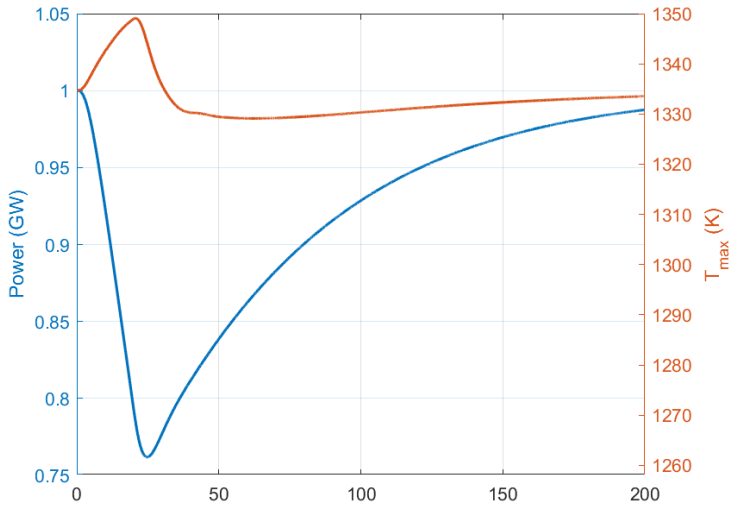


Figure 5.9: *Reactor power and maximum fuel-salt temperature during the transient for* $k = 0.5$ *and* $T_i = 20$ s.

Nevertheless, this result does not demonstrate that temperature feedback can be neglected under more severe cooling degradations or for different transient time scales. A more realistic safety-oriented controller should include a temperature-based signal or an explicit temperature protection limit.

The temperature variation, compared with power variation, is reported in figure 5.10 for the PI-controller parameters $k = 1$ and $T_i = 2.5$ s.

To assess the spatial effects of the selected control strategy, the main thermal hydraulic and neutronic fields are evaluated at the instant $t = t_{T_{\max}}$, at which the maximum fuel-salt temperature is reached, figure 5.11. The resulting distributions are compared with the initial steady-state fields reported in Figure 3.11. The comparison shows that the transient generated by the partial loss of cooling and the subsequent controller action does not produce a qualitative rearrangement of the reactor fields. The velocity distribution retains the main recirculation structures observed at steady state, indicating that the flow topology remains mainly governed by the imposed forced convection. The buoyancy force is too weak with the small temperature increase produced. The temperature field barely increases in magnitude, its overall spatial distribution remains almost identical to the initial one. The delayed-neutron precursor fields also preserve their characteristic asymmetric distributions, although their local magnitude is modified by the temporary reduction in reactor power and by the transient evolution of the flow field.

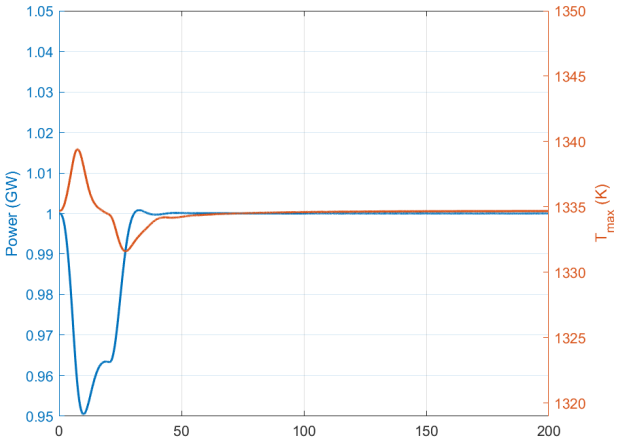
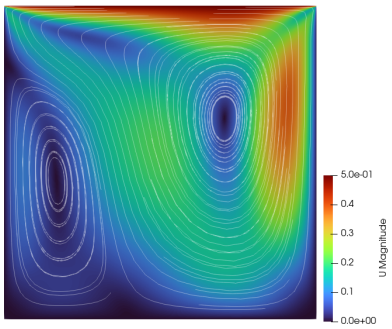
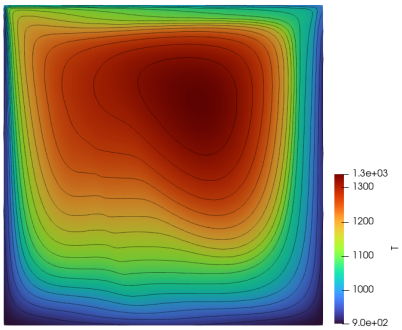


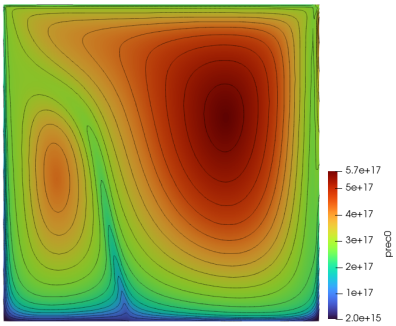
Figure 5.10: *Reactor power and maximum fuel-salt temperature during the transient for $k = 1$ and $T_i = 2.5$ s.*



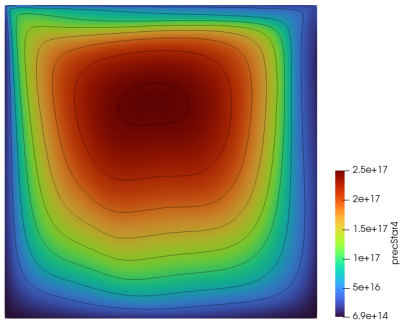
(a) *Velocity field with streamlines.*



(b) *Temeprature field with isolines.*



(c) *First family of delayed neutron precursor field with isolines.*



(d) *Fifth family of delayed neutron precursor field with isolines.*

Figure 5.11: *Fields at t=10 s.*

Therefore, for the selected PI tuning, the cooling transient primarily produces moderate variations in the magnitude of the coupled variables, rather than a substantial change in their spatial structure, confirming the effectiveness of the PI controller.

Chapter 6

Conclusions and future developments

This thesis investigated the development of a multiphysics and multi-scale simulation framework for liquid fuel molten salt reactors by coupling the spatially resolved GeN-Foam solver with a system-level model implemented in OpenModelica. The work pursued three main objectives: verifying the capability of GeN-Foam to reproduce the characteristic coupled behaviour of a molten salt fast reactor, establishing and verifying the communication between GeN-Foam and OpenModelica, and applying the resulting co-simulation framework to the analysis of an idealised emergency heat removal strategy.

The first part of the work focused on the implementation of the CNRS numerical benchmark in GeN-Foam. The benchmark provided a systematic verification procedure in which thermal hydraulics, neutronics, delayed neutron precursor transport, fuel density feedback and buoyancy were introduced progressively. This step-by-step approach made it possible to distinguish errors associated with the individual physical models from those introduced by their coupling.

The single physics calculations showed good agreement with the reference code-to-code results. The average discrepancies were approximately 0.59% for the velocity profiles, 0.42% for the static neutronic solution and 0.15% for the passive scalar temperature distributions. Similarly, satisfactory agreement was obtained when the different physical phenomena were coupled. The discrepancies were approximately 0.68% for delayed neutron precursor transport, 1.43% for the coupling between power and temperature, 0.59% for the buoyancy driven case and 0.97% for the fully coupled steady-state configuration.

The transient frequency domain analysis also reproduced the expected dynamic response of the system. At low perturbation frequencies, reactor power followed the variation in the heat removal coefficient with a gain close to unity and a limited

phase shift. At higher frequencies, the amplitude of the power response decreased and the phase shift approached $-90°$, consistently with the thermal inertia of the salt and the filtering effect associated with the different delayed neutron precursor families. The maximum average discrepancy obtained for the transient observables was 2.8%. The remaining differences with the reference solutions were mainly attributed to the spatial discretisation, the treatment of vacuum boundaries and the temporal resolution of the intermediate frequency cases.
These results do not constitute validation against experimental measurements, since the CNRS benchmark is itself a code-to-code comparison. Nevertheless, they provide evidence that the implemented GeN-Foam model correctly reproduces the principal neutronic and thermal hydraulic interactions represented in the benchmark. The model was therefore considered sufficiently verified for use as the spatially resolved reactor solver in the subsequent co-simulation analysis.

The second part of the thesis concerned the coupling of GeN-Foam with OpenModelica through the Functional Mock-up Interface. In the adopted architecture, GeN-Foam calculates the coupled reactor behaviour and transfers the total power to OpenModelica. The system-level model processes this information and returns the effective volumetric heat transfer coefficient γ, which represents the action of the external heat removal system.
The communication interface was first assessed through a one way verification case reproducing the slowest sinusoidal perturbation of the CNRS benchmark. The heat transfer coefficient generated by OpenModelica and the resulting power calculated by GeN-Foam showed an almost complete overlap with the corresponding GeN-Foam only solution. Relative differences of 0.13% and 0.34% were obtained for the power gain and phase shift, respectively. A delay equal to one communication step, $8 \times 10^{-3}\,\mathrm{s}$, was identified in the transferred value of γ. This delay is a numerical consequence of the explicit staggered coupling algorithm rather than a physical reactor response, and its influence was negligible for the transient considered.
The verification case therefore demonstrated that signals produced by OpenModelica are correctly transferred to and applied by GeN-Foam. More generally, it showed that the proposed architecture can retain the spatially resolved multiphysics description of the reactor while introducing control laws, actuator dynamics and external system behaviour in a modular system-level environment.

The framework was subsequently applied to an idealised degradation of the reactor heat removal capability. A 30% reduction in the main cooling contribution, occurring over a 20 s ramp, was imposed. The disturbance caused an increase in fuel salt temperature, which introduced negative reactivity through fuel expansion and consequently reduced reactor power. An auxiliary cooling contribution, governed by a limited PI controller, was introduced in OpenModelica to compensate for this perturbation.

The parametric investigation highlighted the compromise involved in selecting the proportional gain k and the integral time T_i. Increasing the gain or reducing the integral time accelerated the recovery of the nominal power, but also increased the rate of change of γ and could produce an oscillatory closed loop response. Conversely, weak proportional or integral action resulted in smoother behaviour but larger power deviations and longer settling times. The study also showed that the total integral gain

$$K_i = \frac{k}{T_i}$$

is not sufficient by itself to characterise the controller, since configurations with the same value of K_i may produce different responses depending on the relative contributions of the proportional and integral terms.
Among the investigated configurations, the parameters

$$k = 1.0, \qquad T_i = 2.5\,\mathrm{s}, \qquad K_i = 0.40\,\mathrm{s}^{-1}$$

provided the most favourable compromise according to the adopted qualitative criteria. This configuration limited the power deviation, restored the nominal condition rapidly and avoided significant power overshoot or sustained oscillations, while maintaining a bounded cooling command. This result should be interpreted as the best configuration among the tested cases rather than as a generally optimal controller. Its performance depends on the selected failure amplitude and duration, actuator time constant, saturation limits and co-simulation communication strategy.
The thermal hydraulic results were also used to assess the assumption of using reactor power as the only feedback variable. For the least favourable controller configuration considered, the maximum fuel salt temperature increased by approximately 14.4 K, corresponding to about 1.1% of its initial maximum value. The limited temperature excursion was caused by the large thermal inertia of the fuel salt and by the reduction in fission power produced by the negative temperature reactivity feedback.
For the selected controller configuration, the spatial fields evaluated at the instant of maximum temperature retained the principal characteristics of the initial steady state solution. The main recirculation structures remained unchanged, the temperature distribution increased only moderately, and the delayed neutron precursor fields preserved their characteristic asymmetric shapes. The transient therefore mainly affected the magnitude of the coupled variables rather than producing a qualitative reorganisation of their spatial distributions.
Although these results support the use of power only feedback for the specific reference transient, they do not demonstrate that temperature can be neglected in general. More severe cooling degradations, slower or faster disturbances and different operating conditions may produce larger or more localised temperature

excursions. A safety oriented controller should therefore include at least one temperature related indicator or an explicit thermal protection limit.
Several limitations of the present work must consequently be acknowledged. The reactor configuration is based on the simplified two dimensional CNRS benchmark and does not represent the geometry or all the physical phenomena of a complete MSR plant. The external cooling system is described through a spatially uniform effective heat transfer coefficient rather than through explicit models of pumps, heat exchangers, valves and secondary loop dynamics. Consequently, the results demonstrate the capabilities of the numerical framework and the behaviour of an idealised control strategy, but they cannot be used directly to assess the performance or safety of a real emergency cooling system.
Furthermore, only scalar variables are exchanged through the current FMI interface. OpenModelica therefore does not receive the full spatial information calculated by GeN-Foam, and the controller cannot detect local phenomena unless appropriate indicators are specifically defined and transferred. The explicit staggered exchange also introduces a one step coupling delay and does not enforce convergence between the two solvers within each communication interval.

Future work should first extend the OpenModelica model to include physically based representations of the primary, intermediate and secondary circuits, including heat exchangers, pumps, valves, thermal capacities and realistic actuator constraints. This would allow the effective coefficient γ to be replaced by quantities directly related to component operation, such as mass flow rates, pump speeds or heat exchanger boundary conditions.
The FMI interface should also be extended to exchange additional reactor variables, including maximum and volume averaged temperatures, local probe temperatures, flow rates, reactivity indicators and regional power values. Such an extension would enable multi variable control strategies and provide explicit protection against local thermal limits. The use of adaptive communication steps or iterative implicit coupling should also be investigated, particularly for transients whose characteristic time scales are comparable with the communication interval.
A more systematic controller assessment should employ quantitative performance indices, such as the integral of the absolute power error, maximum temperature and power deviations, settling time, overshoot, cooling effort and maximum rate of change of the actuation variable. The robustness of the selected tuning should then be tested against different failure fractions, degradation durations, initial operating conditions, actuator dynamics and communication time steps. More advanced control approaches, including gain scheduling, model predictive control or combined power temperature control, could subsequently be compared with the PI strategy.
Finally, the framework should be applied to a realistic three dimensional reactor

model and coupled to a complete plant level representation. Sensitivity and uncertainty analyses should be performed for nuclear data, thermophysical properties, numerical discretisation and model parameters. Where suitable experimental or integral effect data become available, they should be used to progress from numerical verification toward validation.
Another research direction worth investigating is the development of a reduced-order core model in OpenModelica, calibrated against high-fidelity three-dimensional GeN-Foam simulations. Such a model could reproduce the main multiphysics response of the core at a substantially lower computational cost, facilitating extensive sensitivity, uncertainty, and control analyses.
In conclusion, this thesis established a functioning GeN-Foam/OpenModelica co-simulation framework capable of combining a spatially resolved description of molten salt reactor multiphysics with flexible system-level and control models. The agreement obtained in the CNRS benchmark, the successful verification of the FMI data exchange and the stable operation of the closed loop cooling model demonstrate the potential of the proposed approach. At the same time, the study defines clearly the developments required before the framework can be applied to the safety analysis and control design of a realistic molten salt reactor plant.

Appendices

Appendix A

Neutronics data

Group,g	Upper energy bound (MeV)
1	2.0×10^{1}
2	2.231×10^{0}
3	4.979×10^{-1}
4	2.479×10^{-2}
5	5.531×10^{-3}
6	7.475×10^{-4}

Table A.1: Energy group division

Table A.2: Total and fission cross sections.

Group g	$\Sigma_{t,g}$ [cm^{-1}]	$\Sigma_{f,g}$ [cm^{-1}]
1	1.65512×10^{-1}	1.11309×10^{-3}
2	2.17253×10^{-1}	1.08682×10^{-3}
3	3.18009×10^{-1}	1.52219×10^{-3}
4	2.42093×10^{-1}	2.58190×10^{-3}
5	2.50351×10^{-1}	5.36326×10^{-3}
6	2.72159×10^{-1}	1.44917×10^{-2}

Table A.3: P_0 scattering cross sections $\Sigma_{s,0,g'\to g}$ [cm^{-1}].

g'	1	2	3	4	5	6
1	1.08476×10^{-1}	5.23316×10^{-2}	4.01805×10^{-3}	1.09869×10^{-4}	2.53290×10^{-5}	3.78334×10^{-6}
2	0	1.83666×10^{-1}	3.19138×10^{-2}	2.34218×10^{-5}	2.25259×10^{-6}	2.00405×10^{-7}
3	0	0	2.98293×10^{-1}	1.63470×10^{-2}	1.70575×10^{-5}	1.24625×10^{-6}
4	0	0	0	2.17472×10^{-1}	1.90243×10^{-2}	1.36858×10^{-8}
5	0	0	0	0	2.27173×10^{-1}	1.05885×10^{-2}
6	0	0	0	0	0	2.37826×10^{-1}

Table A.4: P_1 scattering cross sections $\Sigma_{s,1,g'\to g}$ [cm^{-1}].

g'	1	2	3	4	5	6
1	5.02479×10^{-2}	-5.31822×10^{-3}	4.77967×10^{-4}	3.40533×10^{-5}	8.93761×10^{-6}	1.34789×10^{-6}
2	0	3.78784×10^{-2}	-6.77085×10^{-3}	-1.92591×10^{-6}	-6.99121×10^{-8}	1.59216×10^{-8}
3	0	0	2.42287×10^{-2}	-5.13491×10^{-3}	8.27643×10^{-7}	3.25237×10^{-8}
4	0	0	0	1.75032×10^{-2}	-5.64363×10^{-3}	-5.05232×10^{-10}
5	0	0	0	0	1.49908×10^{-2}	-3.22691×10^{-3}
6	0	0	0	0	0	1.17041×10^{-2}

Table A.5: P_1-corrected diffusion coefficients and inverse neutron velocities.

Group g	D_g [cm]	$1/v_g$ [$\mathrm{s\,cm}^{-1}$]
1	2.80064	4.00367×10^{-10}
2	1.84021	7.39846×10^{-10}
3	1.13110	2.61748×10^{-9}
4	1.44786	6.69270×10^{-9}
5	1.39750	1.55845×10^{-8}
6	1.28252	4.24462×10^{-8}

Appendix B

Codes

B.1 Transient GeN-Foam

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2506                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      fvOptions;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
heatSink
{

    type            scalarCodedSource;
    name            scalarCodedSource;


    scalarCodedSourceCoeffs
    {
    active          true;
    selectionMode   all;
    fields          (h);


    codeInclude
```

```
31          #{
32              #include "mathematicalConstants.H"
33          #};
34
35      codeCorrect
36          #{
37          //Pout<< "**codeCorrect**" << endl;
38          #};
39
40
41       codeAddSupRho
42          #{
43
44              const Time& time = mesh().time();
45              const scalar t = time.value();
46
47              const volScalarField& hField =
48                  mesh().lookupObject<volScalarField>("h");
49              const volScalarField& TField = mesh().lookupObject<volScalarField>(
    "T");
50              const scalarField& T = TField.internalField();
51
52              const scalarField& h = hField.internalField();
53              const scalarField& V = mesh().V();
54              scalarField& heSource = eqn.source();
55
56
57              const scalar gamma0 = 1e6;
58              const scalar Text_eff = 900.0;
59              const scalar Cp_vol = 3075;
60              const scalar f = 0.0125;
61              const scalar pi = Foam::constant::mathematical::pi;
62              scalar maxT = gMax(T);
63
64              //scalar hcentro = 0;
65              scalar gamma = gamma0 * (1.0 + 0.1 * Foam::sin(2.0 * pi * f * t));
66              static autoPtr<OFstream> gammaFilePtr;
67
68      // Se è la prima volta che il codice viene eseguito, crea il file
69              if (!gammaFilePtr.valid())
70              {
71                  gammaFilePtr.reset(new OFstream("gamma_history.dat"));
72                  // Opzionale: aggiungi un'intestazione
73                  gammaFilePtr() << "# Time    Gamma" << endl;
74              }
75
76              // Scrive i dati nel file
77              gammaFilePtr() << t << " " << gamma << endl;
78
```

```
            scalar Su = gamma * Text_eff;
            scalar Sp = -gamma / Cp_vol;

            forAll(heSource, i)
            {
               heSource[i] += - gamma * (Text_eff - T[i]) * V[i];

            }
            scalar hmax = gamma * (Text_eff - maxT) * 2.5e-5;
            Info << gamma << hmax << endl;
        #};
    codeConstrain
        #{

        #};

      }
}
// ************************************************************************* //
```

Listing B.1: OpenFOAM heat sink source term for GeNfoam transient

B.2 Transient Co-simulation

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  v2506                                 |
|   \\  /    A nd           | Website:  www.openfoam.com                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      fvOptions;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
heatSink
{

    type            scalarCodedSource;
    name            scalarCodedSource;


    scalarCodedSourceCoeffs
```

```
    {
    active          true;
    selectionMode   all;
    fields          (h);

    Gamma
        {
            type            fmi;
            nameFromFMU     Gamma; // Name of the variable in OpenModelica
            initialValue    1e6;
        }

        codeInclude
        #{
            #include "mathematicalConstants.H"
            #include "timeProfile.H"
        #include "volFields.H"
        #};
        codeOptions
        #{
            -I/home/lareddia/foamForNuclear/src/profiles/lnInclude
        #};
        codeCorrect
        #{
        //Pout<< "**codeCorrect**" << endl;
        #};


        codeAddSupRho
        #{
            const Time& time = mesh().time();
            const scalar t = time.value();

            // Get the dictionary block containing "Gamma"
            const dictionary& dict = coeffs();

            // ------------------------------------------------------------
            // INITIALIZE THE FMI CONNECTION (ONLY ONCE)
            // ------------------------------------------------------------
            static autoPtr<timeProfile> gammaProfilePtr;

            if (!gammaProfilePtr.valid())
            {
                // This calls the specific constructor:
                // timeProfile(const dictionary& object, word timeProfileName,
    const Time& runTime)
                gammaProfilePtr.reset(new timeProfile(dict, "Gamma", time));

                if (Pstream::master())
```

```
72                 {
73                     Info<< "Initialized FMI connection for Gamma." << endl;
74                 }
75             }
76 
77             // ------------------------------------------------------
78             // FETCH THE DYNAMIC VALUE FROM OPENMODELICA
79             // ------------------------------------------------------
80             // This pulls the current value via GeN-Foam's commDataLayer
81             scalar gamma = gammaProfilePtr->value(t);
82             const volScalarField& hField =
83                 mesh().lookupObject<volScalarField>("h");
84             const volScalarField& TField = mesh().lookupObject<volScalarField>(
    "T");
85             const scalarField& T = TField.internalField();
86 
87             const scalarField& h = hField.internalField();
88             const scalarField& V = mesh().V();
89             scalarField& heSource = eqn.source();
90             const scalar Text_eff = 900.0;
91             const scalar Cp_vol = 3075;
92             const scalar f = 0.0125;
93             const scalar pi = Foam::constant::mathematical::pi;
94             scalar maxT = gMax(T);
95 
96             //scalar hcentro = 0;
97             static autoPtr<OFstream> gammaFilePtr;
98 
99     // Se   la prima volta che il codice viene eseguito, crea il file
100            if (!gammaFilePtr.valid())
101            {
102                gammaFilePtr.reset(new OFstream("gamma_history.dat"));
103                // Opzionale: aggiungi un'intestazione
104                gammaFilePtr() << "# Time    Gamma" << endl;
105            }
106 
107            // Scrive i dati nel file
108            gammaFilePtr() << t << " " << gamma << endl;
109 
110            //scalar Su = gamma * Text_eff;
111            //scalar Sp = -gamma / Cp_vol;
112 
113            forAll(heSource, i)
114            {
115               heSource[i] += - gamma * (Text_eff - T[i]) * V[i];
116 
117            }
118            scalar hmax = gamma * (Text_eff - maxT) * 1e-4;
119            Info << gamma << hmax << endl;
```

```
        #};
    codeConstrain
        #{

        #};

      }
}
// ************************************************************************* //
```

Listing B.2: OpenFOAM heat sink source term for Co-simulation transient

Listing B.3: OpenModelica implementation of the emergency cooling control system.

```
model GammaEmergencyCoolingBlocks

  parameter Real P0 = 1e9;
  parameter Real PReference = P0
;
  parameter Real gamma0 = 1e6;

  parameter Real failureFraction(min=0, max=1) = 0.30;
  parameter Modelica.Units.SI.Time failureStart = 0;
  parameter Modelica.Units.SI.Time failureDuration = 20;
  parameter Boolean useRampFailure = true;

  parameter Real Kp = 5
    "PI proportional gain: relative backup cooling per relative power error";
  parameter Modelica.Units.SI.Time Ti = 10
    "PI integral time; must be tuned with the coupled GeN−Foam model";
  parameter Real Ni = 0.5;
  parameter Real backupMax(min=0) = 0.50;

  parameter Real gammaMinRelative(min=0) = 0.10;
  parameter Real gammaMaxRelative(min=0) = 2.00;
  parameter Modelica.Units.SI.Time actuatorTimeConstant = 2;

  Modelica.Blocks.Interfaces.RealInput power annotation(
      Placement(transformation(origin={−180,80}, extent={{−20,−20},{20,20}}),
        iconTransformation(origin={−120,60}, extent={{−20,−20},{20,20}})));

  Modelica.Blocks.Interfaces.RealOutput Gamma annotation(
      Placement(transformation(origin={180,20}, extent={{−10,−10},{10,10}}),
        iconTransformation(origin={120,50}, extent={{−10,−10},{10,10}})));

  Modelica.Blocks.Math.Gain normalizePower(k=1/P0) annotation(
    Placement(transformation(origin={−130,80}, extent={{−10,−10},{10,10}})));

```

```
35  Modelica.Blocks.Sources.Constant referenceRelative(k=PReference/P0) annotation(
36    Placement(transformation(origin={−130,120}, extent={{−10,−10},{10,10}})));
37
38  Modelica.Blocks.Continuous.LimPID powerController(
39    controllerType=Modelica.Blocks.Types.SimpleController.PI,
40    k=Kp,
41    Ti=Ti,
42    Ni=Ni,
43    yMax=backupMax,
44    yMin=0,
45    initType=Modelica.Blocks.Types.Init.InitialOutput,
46    y_start=0,
47    strict=true) annotation(
48      Placement(transformation(origin={−52,98}, extent={{−12,−12},{12,12}})));
49
50  Modelica.Blocks.Sources.Ramp mainCoolingRamp(
51    height=−failureFraction,
52    duration=failureDuration,
53    offset=1,
54    startTime=failureStart) annotation(
55      Placement(transformation(origin={−164,−52}, extent={{−10,−10},{10,10}})));
56
57  Modelica.Blocks.Sources.Step mainCoolingStep(
58    height=−failureFraction,
59    offset=1,
60    startTime=failureStart) annotation(
61      Placement(transformation(origin={−130,−100}, extent={{−10,−10},{10,10}})));
62
63  Modelica.Blocks.Sources.BooleanConstant selectRamp(k= true) annotation(
64    Placement(transformation(origin={−130,−20}, extent={{−10,−10},{10,10}})));
65
66  Modelica.Blocks.Logical.Switch failureSelector annotation(
67    Placement(transformation(origin={−70,−60}, extent={{−10,−10},{10,10}})));
68
69  Modelica.Blocks.Math.Add addMainAndBackup(k1=1, k2=1) annotation(
70    Placement(transformation(origin={10,10}, extent={{−10,−10},{10,10}})));
71
72  Modelica.Blocks.Nonlinear.Limiter gammaCommandLimiter(
73    uMax=gammaMaxRelative,
74    uMin=gammaMinRelative,
75    strict=true) annotation(
76      Placement(transformation(origin={50,10}, extent={{−10,−10},{10,10}})));
77
78  Modelica.Blocks.Continuous.FirstOrder actuator(
79    k=1,
80    T=actuatorTimeConstant,
81    initType=Modelica.Blocks.Types.Init.InitialOutput,
82    y_start=1) annotation(
83      Placement(transformation(origin={90,10}, extent={{−10,−10},{10,10}})));
```

```
84
85    Modelica.Blocks.Math.Gain dimensionalGamma(k=gamma0) annotation(
86      Placement(transformation(origin={130,20}, extent={{−10,−10},{10,10}})));
87  initial equation
88    assert(P0 > 0);
89    assert(PReference > 0);
90    assert(gamma0 > 0);
91    assert(Ti > 0);
92    assert(Ni > 0);
93    assert(actuatorTimeConstant > 0);
94    assert(failureDuration > 0);
95    assert(gammaMaxRelative > gammaMinRelative);
96
97  equation
98    connect(power, normalizePower.u) annotation(
99      Line(points = {{−180, 80}, {−142, 80}}, color = {0, 0, 127}));
100   connect(normalizePower.y, powerController.u_m) annotation(
101     Line(points = {{−119, 80}, {−100, 80}, {−100, 84}, {−52, 84}}, color = {0, 0, 127}));
102   connect(referenceRelative.y, powerController.u_s) annotation(
103     Line(points = {{−119, 120}, {−100, 120}, {−100, 98}, {−66, 98}}, color = {0, 0, 127}));
104   connect(mainCoolingRamp.y, failureSelector.u1) annotation(
105     Line(points = {{−153, −52}, {−82, −52}}, color = {0, 0, 127}));
106   connect(selectRamp.y, failureSelector.u2) annotation(
107     Line(points = {{−119, −20}, {−100, −20}, {−100, −60}, {−82, −60}}, color = {255, 0,
        255}));
108   connect(mainCoolingStep.y, failureSelector.u3) annotation(
109     Line(points = {{−119, −100}, {−100, −100}, {−100, −68}, {−82, −68}}, color = {0, 0,
        127}));
110   connect(failureSelector.y, addMainAndBackup.u2) annotation(
111     Line(points = {{−59, −60}, {−20, −60}, {−20, 4}, {−2, 4}}, color = {0, 0, 127}));
112   connect(powerController.y, addMainAndBackup.u1) annotation(
113     Line(points = {{−39, 98}, {−20, 98}, {−20, 16}, {−2, 16}}, color = {0, 0, 127}));
114   connect(addMainAndBackup.y, gammaCommandLimiter.u) annotation(
115     Line(points = {{21, 10}, {38, 10}}, color = {0, 0, 127}));
116   connect(gammaCommandLimiter.y, actuator.u) annotation(
117     Line(points = {{61, 10}, {78, 10}}, color = {0, 0, 127}));
118   connect(actuator.y, dimensionalGamma.u) annotation(
119     Line(points = {{101, 10}, {110, 10}, {110, 20}, {118, 20}}, color = {0, 0, 127}));
120   connect(dimensionalGamma.y, Gamma) annotation(
121     Line(points = {{142, 20}, {180, 20}}, color = {0, 0, 127}));
122   annotation(
123     uses(Modelica(version="4.0.0")));
124 end GammaEmergencyCoolingBlocks;
```

Bibliography

[1] Our World in Data. *Global primary energy consumption by source*. URL: https://ourworldindata.org/grapher/global-energy-consumption-source?time=1974..2024 (cit. on p. 2).

[2] IAEA. *What is Nuclear Energy? The Science of Nuclear Power*. 2022. URL: https://www.iaea.org/newscenter/news/what-is-nuclear-energy-the-science-of-nuclear-power (cit. on p. 3).

[3] Roman Mouginot and Hannu Hänninen. *Microstructures of nickel-base alloy dissimilar metal welds*. 2013. URL: https://www.researchgate.net/publication/260576574_Microstructures_of_nickel-base_alloy_dissimilar_metal_welds (cit. on p. 4).

[4] Office of Nuclear Energy. *What is Generation Capacity?* U.S. Department of Energy. 2025. URL: https://www.energy.gov/ne/articles/what-generation-capacity (cit. on p. 5).

[5] Stephen M. Goldberg and Robert Rosner. *Nuclear Reactors: Generation to Generation*. American Nuclear Society, 2011 (cit. on pp. 5, 6).

[6] *The Generation IV International Forum*. URL: https://www.gen-4.org/about-gif/history-and-achievements-gif (cit. on p. 7).

[7] World Nuclear Association. *GIF Annual Reports*. 2024. URL: https://world-nuclear.org/information-library/nuclear-fuel-cycle/nuclear-power-reactors/generation-iv-nuclear-reactors (cit. on pp. 7, 8).

[8] ENDURANCE. *D1.1 Review of the EU MSR designs*. 2025. URL: https://www.endurance-msr-project.eu/publications-outreach/ (cit. on p. 9).

[9] Sustainable Nuclear Energy Technology Platform (SNETP). *ENDURANCE project*. 2025. URL: https://snetp.eu/portfolio-items/endurance/ (cit. on pp. 9, 10).

[10] CEA and ORANO. «Opportunities of molten salt fuel for actinides management». In: 2022. URL: https://conferences.iaea.org/event/321/attachments/13273/20588/D2_24_France_Molten%20Salt%20Reactor_Pascal.pdf (cit. on pp. 9, 10).

[11] Paul N. Haubenreich and J. R. Engel. «Experience with the Molten-Salt Reactor Experiment». In: *Nuclear Applications and Technology* 8.2 (1970), pp. 118–136. DOI: 10.13182/NT8-2-118 (cit. on p. 12).

[12] Marco Tiberga et al. «Results from a multi-physics numerical benchmark for codes dedicated to molten salt fast reactors». In: *Annals of Nuclear Energy* 142 (2020), p. 107428. ISSN: 0306-4549. DOI: https://doi.org/10.1016/j.anucene.2020.107428. URL: https://www.sciencedirect.com/science/article/pii/S0306454920301262 (cit. on pp. 13, 14, 31).

[13] CFD Direct. *The PIMPLE algorithm*. URL: https://doc.cfd.direct/notes/cfd-general-principles/the-pimple-algorithm (cit. on p. 16).

[14] Tanay Mazumdar and S.B. Degweker. «Solution of the neutron transport equation by the Method of Characteristics using a linear representation of the source within a mesh». In: *Annals of Nuclear Energy* 108 (Oct. 2017), pp. 132–150. DOI: 10.1016/j.anucene.2017.04.011 (cit. on p. 19).

[15] Hugh Coleman and Committee Members. *ASME V and V 20-2009 Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer (V and V20 Committee Chair and principal author)*. Jan. 2009 (cit. on pp. 19, 20).

[16] *GeN-Foam User Guide*. URL: https://foam-for-nuclear.gitlab.io/GeN-Foam/user_guide/fmu/ (cit. on p. 42).

[17] Richard Courant, Kurt Friedrichs, and Hans Lewy. «Über die partiellen Differenzengleichungen der mathematischen Physik». In: *Mathematische Annalen* 100 (1928), pp. 32–74. DOI: 10.1007/BF01448839 (cit. on p. 45).