

Developing a multi-physics tool for the neutronics-thermal-hydraulics in a molten salt reactor using the lattice Boltzmann method

Master Thesis

Coco Polderman

Developing a multi-physics tool for the neutronics- thermal-hydraulics in a molten salt reactor using the lattice Boltzmann method

Master Thesis

by

Coco Polderman

Student Name	Student Number
Coco Polderman	4604628

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday February 14, 2024 at 13:30 PM.

Thesis committee: Dr. Ir. M. Rohde
Dr. Ir. D. Lathouwers
Dr. D.S.W. Tam
Faculty: Faculty of Applied Sciences, Delft

Highlights

The most important contributions from this research project to existing literature are:

- The development of a novel coupled multi-physics model using the lattice Boltzmann method, specifically designed for resolving the neutron diffusion equation in combination with thermal-hydraulics within a Molten Salt Fast Reactor, and its subsequent validation against well-established numerical tools.*
- A rationale for employing lower, yet sufficiently high Prandtl and Schmidt numbers in scenarios where diffusion effects are negligible. This approach is substantiated with evidence, presenting an effective strategy to reduce numerical complexity while maintaining adequate accuracy.*
- The successful integration of the predictor corrector quasi-static method and lattice Boltzmann method for perturbations in transient coupled multi-physics reactor simulations, a pioneering achievement in the field of describing neutron behaviour with the lattice Boltzmann method.*

Abstract

In Molten Salt Fast Reactors (MSFRs) liquid salt acts as both coolant and fuel. This results in strong coupling of temperature and flow fields to neutronics compared to solid fuel reactors. These temperature feedback mechanisms improve the stability of the reactor, and result in safety mechanisms that are inherent to the strong coupling between neutronics and thermal-hydraulics. However, this strong coupling also makes describing the physics within the core complex. Trying to understand the interactions between all these phenomena results in the necessity for computer tools that are efficient in predicting the physics at hands.

This research proposes a multi-physics tool for MSFRs using the lattice Boltzmann method and validates it against existing models. The primary objective is to integrate the neutronics and thermal-hydraulics components in one numerical scheme.

In this research the Navier-Stokes equation, heat equation, neutron diffusion equation and neutron precursor advection-diffusion equation are all solved using the lattice Boltzmann method (LBM). The equations are solved in two-dimensional space using the LBM with the Bhatnagar-Gross-Krook operator. A multiple distribution function approach is implemented, assigning separate distribution functions to different fields. Two types of vacuum boundary conditions are implemented in the LBM and compared on their capabilities to correctly simulate the distribution of the neutron scalar flux. The coupled steady-state eigenvalue problem is solved using the power method. The coupled transient simulation is performed by integrating the steady-state LBM with the predictor corrector quasi-static method (PC-QSM). The model is validated by comparison to a coupled numerical benchmark that was specifically designed for assessing the capabilities of describing the multi-physics by numerical tools.

This research project reveals that for simulating vacuum boundaries, applying the equilibrium boundary (EB) condition proves more effective than the non-equilibrium extrapolation scheme (NEES). This is evidenced by the fact that the EB yields increasingly lower discrepancies from benchmark codes with an increased number of lattice nodes, whereas the discrepancies from benchmark codes in the NEES increases with a greater number of lattice nodes.

The steady-state coupled model developed in this research project using the LBM proves to be efficient in modelling the complex physics in MSFRs, finding discrepancies from benchmark codes in eigenvalues between 0.000019% and 0.000043% and discrepancies from benchmark codes of macroscopic fields between 0.32% and 5.9%.

The transient coupled model represents a pioneering and successful integration of the PCQSM and LBM for perturbations in coupled reactor simulations, marking a notable first in this field of study. However, numerical issues arise in the determination of the initial reactivity, resulting in a steep initial peak in reactivity and thus in reactor power. Nevertheless, the developed tool delivers promising results, accurately capturing the trends in gain and phase shift across varying perturbation frequencies. This achievement underscores its potential, although there remains scope for further refinement in its numerical implementation to enhance its performance.

Additionally, in this research a rationale is proposed for employing lower, yet sufficiently high Prandtl (Pr) and Schmidt (Sc) numbers in scenarios where diffusion effects are negligible. It is found that discrepancies from simulations that implement higher Pr and Sc numbers are of importance in regions where gradients in temperature and concentration are highest. This approach presents an effective strategy to reduce numerical complexity while maintaining adequate accuracy.

The presented model in this research project integrates the neutronics and thermal-hydraulics components in a single numerical scheme, the LBM. This results in all fields having a similar data structure and it simplifies information exchange between fields. Further, in the LBM solutions for the particle distribution and particle collisions are local and the simulation method is highly parallelizable making the coupled multi-physics tool in this research adequate for GPU simulations, increasing numerical speed.

Contents

Highlights	i
Abstract	ii
Nomenclature	ix
1 Introduction	1
1.1 The Molten Salt Fast Reactor	1
1.2 Review of existing literature and models	3
1.2.1 Existing multi-physics tools for Molten Salt Fast Reactors	3
1.2.2 Models for neutronics using the lattice Boltzmann method	3
1.3 Thesis goals	5
2 Theory	7
2.1 Nuclear reactor physics	7
2.2 Neutron transport and neutron diffusion	8
2.2.1 Steady-state neutronics: reactor criticality calculation	10
2.2.2 Power method for k-eigenvalue problem	11
2.2.3 Transient behaviour in a nuclear reactor core: the quasi-static method	12
2.3 The thermal-hydraulics in a nuclear reactor core	14
2.3.1 Fluid dynamics	14
2.3.2 Temperature field	15
2.4 Interaction between fields	15
2.5 From macroscopic equations to the Boltzmann equation: kinetic theory	15
3 Numerical Method	17
3.1 Lattice Boltzmann Method	17
3.1.1 LBM for the Navier-Stokes equation	18
3.1.2 LBM for the heat equation	19
3.1.3 LBM for the neutron diffusion equation	20
3.1.4 LBM for neutron precursor advection-diffusion equation	21
3.2 Boundary conditions	21
3.2.1 Dirichlet boundary conditions	22
3.2.2 Von Neumann boundary conditions	22
3.2.3 Boundary condition for neutronics	22
3.3 Lattice conversion	24
3.4 Stability and accuracy constraints	24
3.4.1 Dealing with high Prandtl and Schmidt numbers	25
3.4.2 Dealing with high neutron velocities	26
3.4.3 Ensuring stability while choosing lattice units	26
3.5 Scheme for solving coupled eigenvalue problem	26
3.6 Predictor Corrector Quasi-Static Method	27
4 Validation of single-physics models	32
4.1 Description of the Tiberge benchmark	32
4.1.1 The molten salt system	33
4.2 Phase 0.1, Velocity field in lid driven cavity	35
4.3 Neutronics field modelled using several benchmarks	36
4.3.1 Neutronics field in a homogeneous solid fuel reactor core	36
4.3.2 Neutronics field in a multi-region solid fuel reactor core	38
4.3.3 Phase 0.2, Static fuel with six neutron energy groups and eight neutron precursor families	40

4.4	Phase 0.3, Temperature	42
4.5	Phase 1.1, Circulating fuel	44
4.6	Take-aways for the coupling of fields	47
5	Validation of steady-state coupled models	48
5.1	Phase 1.2, Power coupling	48
5.2	Phase 1.3, Buoyancy	51
5.3	Phase 1.4, Full coupling	55
5.4	Comparing the differences between phases	59
5.4.1	Velocity field	59
5.4.2	Temperature field	59
5.4.3	Precursor concentration fields	60
6	Validation of transient coupled models	62
6.1	Transient model with perturbed reactivity	62
6.2	Transient model with perturbed heat transfer coefficient	63
7	Conclusion	68
7.1	Steady-state models	68
7.2	Transient models	69
7.3	Recommendations	69
	References	71
A	Iteration schemes steady-state uncoupled neutronics simulations	75
B	Neutronics and precursor data	78
B.1	Neutronics data	78
B.2	Precursor parameters	79
C	Derivation of non-dimensional governing equations for concentration and temperature	80
C.1	Non-dimensional heat equation	80
C.2	Non-dimensional concentration advection-diffusion equation	81
D	Results for uncoupled neutronics simulations of chapter 4	82
D.1	Results for the neutron fluxes in a homogeneous solid fuel reactor core	82
D.2	Results for the benchmark flux field in the TWIGL reactor	83
E	Results for the fully coupled transient simulations	85
E.1	Results for the transient simulations in first 3 seconds after perturbation	85
E.2	Results of transient simulations displaying outcomes after power jump stabilization period	92
F	Wall-clock times and memory storage	100
F.1	Wall-clock times for selected simulations	100
F.2	Memory usage per node in simulations	100

List of Figures

1.1	Schematic overview of a Molten Salt Fast Reactor [32]	2
2.1	Difference in behaviour of the exact scalar neutron flux and the scalar neutron flux in the diffusion approximation at a vacuum boundary [8].	10
3.1	Two lattice velocity sets implemented in the LBM, $D2Q4$ in black and $D2Q9$ in black and grey [21].	18
3.2	Schematic overview of the collision and streaming step for distribution function f_i	19
3.3	Schematic overview of link-wise and wet node boundary schemes.	21
3.4	Schematic overview of the simulation domains for the neutron fields and thermal-hydraulics fields.	23
3.5	Schematic overview of the extrapolated boundary coordinates, $\tilde{x}_{boundary,1}$ and $\tilde{x}_{boundary,2}$ as prescribed in the diffusion approximation, and the actually implemented extrapolation in the numerical model, x_b	23
3.6	Schematic overview of the different time-scales used in the PCQSM.	28
3.7	Solving scheme for transient simulation using PCQSM and LBM. Part (1/2) <i>Scheme continues on the next page.</i>	30
3.8	Solving scheme for transient simulation using PCQSM and LBM. Part (2/2) <i>Continuing from previous page.</i>	31
4.1	Schematic overview of the simulation domain, with observation center-lines AA' and BB' [35]	34
4.2	Horizontal velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.1.	35
4.3	Vertical velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.1.	36
4.4	Discrepancy from benchmark codes of the horizontal (U_x) and vertical (U_y) velocity fields along center-lines for different numbers of grid-point per dimension in phase 0.1.	36
4.5	Square two-dimensional homogeneous multiplying cavity with two neutron energy groups. 37	
4.6	Analytical and LBM solution for a two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 1.	38
4.7	TWIGL benchmark dimensions: Square two-dimensional multiplying cavity filled with seed regions 1 and 2 and blanket region 3 [12, 48].	39
4.8	LB solution for scalar neutron fluxes of the two energy groups in the TWIGL benchmark [12] in the top-right quarter of the simulation domain.	40
4.9	Discrepancy from benchmark codes of the fission rate density for different numbers of grid-points per dimension, for the EB and NEES boundary schemes in phase 0.2.	41
4.10	Fission rate density along center-line AA' for the various benchmark codes and the LBM simulation in phase 0.2.	41
4.11	Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.3.	42
4.12	Discrepancy from benchmark codes of the temperature fields along center-lines for different implemented Pr numbers in phase 0.3.	43
4.13	Discrepancy from benchmark codes of the temperature fields at different locations in the core along center-lines AA' (left) and BB' (right) for different implemented Pr numbers in phase 0.3.	43
4.14	Discrepancy from benchmark codes of the temperature field along center-lines for different numbers of grid-point per dimension in phase 0.3.	44

4.15	Discrepancy from benchmark codes of the fission rate density for different numbers of grid-points per dimension, for the EB and NEES boundary schemes in phase 1.1. Implemented S_c number is 100.	45
4.16	Delayed neutron source along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.1.	45
4.17	Discrepancy from benchmark codes of the precursor source along center-lines for different implemented S_c numbers in phase 1.1.	46
4.18	Discrepancy from benchmark codes of the precursor source at different locations in the core along center-lines AA' (left) and BB' (right) for different implemented S_c numbers in phase 1.1.	46
5.1	Iteration scheme for phase 1.2	49
5.2	Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.2.	50
5.3	Change in fission rate density (compared to uncoupled fission rate density) along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.2.	50
5.4	Iteration scheme for phase 1.3	52
5.5	Horizontal velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.	53
5.6	Vertical velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.	53
5.7	Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.	54
5.8	Delayed neutron source along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.	54
5.9	Iteration scheme for phase 1.4	56
5.10	Horizontal velocity field along center-lines AA' (left) and BB' (right) in phase 1.4.	57
5.11	Vertical velocity field along center-lines AA' (left) and BB' (right) in phase 1.4.	57
5.12	Temperature field along center-lines AA' (left) and BB' (right) in phase 1.4.	58
5.13	Delayed neutron source along center-lines AA' (left) and BB' (right) in phase 1.4.	58
5.14	Magnitude of the velocity field obtained by LBM simulations for three differently coupled phases.	59
5.15	Temperature field obtained by LBM simulations for three differently coupled phases.	60
5.16	Concentration of the first family of precursors ($T_{1/2} = 55.6s$) obtained by LBM simulations for three differently coupled phases.	61
5.17	Concentration of the fifth family of precursors ($T_{1/2} = 2.37s$) obtained by LBM simulations for three differently coupled phases.	61
6.1	Transient results for sinusoidal perturbation of 2.0 Hz in reactivity for the first 2.5 s after the initial perturbation.	63
6.2	Gain and phase-shift as a function of perturbation frequency in the heat transfer coefficient.	64
6.3	Transient results for perturbation of 0.8 Hz from 35 seconds to 40 seconds	66
6.4	Transient results for perturbation of 0.8 Hz from 0 seconds to 10 seconds	67
A.1	Iteration scheme for phase 0.2	76
A.2	Iteration scheme for phase 1.1	77
D.1	Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 2.	82
D.2	Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 3.	83
D.3	Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 4.	83
E.2	Transient results for perturbed heat coefficient with $f = 0.4Hz$, for first 3 seconds after initial perturbation	86

E.1	Transient results for perturbed heat coefficient with $f = 0.8Hz$, for first 3 seconds after initial perturbation	87
E.3	Transient results for perturbed heat coefficient with $f = 0.2Hz$, for first 3 seconds after initial perturbation	88
E.4	Transient results for perturbed heat coefficient with $f = 0.1Hz$, for first 3 seconds after initial perturbation	89
E.5	Transient results for perturbed heat coefficient with $f = 0.05Hz$, for first 3 seconds after initial perturbation	90
E.6	Transient results for perturbed heat coefficient with $f = 0.025Hz$, for first 3 seconds after initial perturbation	91
E.7	Transient results for perturbed heat coefficient with $f = 0.8Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	93
E.8	Transient results for perturbed heat coefficient with $f = 0.4Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	94
E.9	Transient results for perturbed heat coefficient with $f = 0.2Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	95
E.10	Transient results for perturbed heat coefficient with $f = 0.2Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	96
E.11	Transient results for perturbed heat coefficient with $f = 0.1Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	97
E.12	Transient results for perturbed heat coefficient with $f = 0.05Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.	98
E.13	Transient results for perturbed heat coefficient with $f = 0.025Hz$, for time-interval $t = [15s, 60s]$ after initial perturbation.	99

List of Tables

4.1	Fuel salt composition of salt in reactor core [35]	34
4.2	Definition of the energy groups [35].	34
4.3	Definition of precursor families [35].	35
4.4	Physical parameters used in the velocity simulations [35].	35
4.5	Nuclear data for two-energy group systems [7]	38
4.6	Comparison of the neutronics code with analytical result [7]	38
4.7	Nuclear parameters for the seed and blanket regions in the TWIGL benchmark[48], [12], [43].	39
4.8	Physical parameters used in the neutronics simulations [35].	40
4.9	Multiplication factor k_{eff} and reactivity ρ found for the various benchmark codes and the LBM simulation, together with the discrepancies for the multiplication factors in phase 0.2 [7].	42
4.10	Physical parameters prescribed for the temperature field in the benchmark study [35].	42
4.11	Physical parameters prescribed for the precursor fields in the benchmark study [35].	44
4.12	Multiplication factor k_{eff} , reactivity ρ and reactivity change from static fuel $\rho - \rho_{0.2}$ found for the various benchmark codes and the LBM simulation, together with the discrepancies for the multiplication factors in phase 1.1 [7].	46
5.1	Implemented simulation parameters in phase 1.2	48
5.2	Multiplication factor k_{eff} , reactivity ρ and reactivity change from circulating fuel without temperature feedback $\rho - \rho_{1.1}$ found for the various benchmark codes and the LBM simulation in phase 1.2 [7].	50
5.3	Discrepancies of the different codes for different fields in phase 1.2 [7]	51
5.4	Implemented simulation parameters in phase 1.3	51
5.5	Multiplication factor k_{eff} , reactivity ρ and reactivity change from the fully uncoupled neutronics solution $\rho - \rho_{0.2}$ found for the various benchmark codes and the LBM simulation in phase 1.3 [7].	54
5.6	Discrepancies of the different codes for different fields in phase 1.3 [7]	55
5.7	Discrepancies of the different codes for the effective multiplication factor in phase 1.3 [7]	55
5.8	Implemented simulation parameters in phase 1.4	55
5.9	Multiplication factor k_{eff} , reactivity ρ , reactivity change from the fully uncoupled neutronics solution $\rho - \rho_{0.2}$, found for the various benchmark codes and the LBM simulation in phase 1.4 [7].	58
6.1	Point kinetics parameters	62
6.2	Simulation parameters for transient simulation	64
B.1	Fission cross-sections, total (removal) cross-sections, diffusion constants and velocities for the 6 neutron energy groups as retrieved from Tiberga et al. [35].	78
B.2	P_0 scattering cross-sections for the 6 neutron energy groups as originating group and destination group as retrieved from Tiberga et al. [35].	78
B.3	Average number of neutrons emitted per fission event, prompt neutron spectrum, delayed neutron spectrum and average energy emitted per fission event for the 6 neutron energy groups as retrieved from Tiberga et al. [35].	78
B.4	Decay constant and delayed neutron fraction for the 8 delayed precursor families as retrieved from Tiberga et al. [35].	79
F.1	Wall-clock times for selected simulations	100
F.2	Memory usage per node for selected simulations	101

Nomenclature

Abbreviation	Definition
BE	Boltzmann equation
BGK	Bhatnagar Gross Krook
EB	equilibrium boundary
FB	finite Boltzmann
GPU	graphical processing unit
LBE	lattice Boltzmann equation
LBM	lattice Boltzmann method
MCM	Monte Carlo Method
MSFR	molten salt fast reactor
MSR	molten salt reactor
NDE	neutron diffusion equation
NEES	non-equilibrium extrapolation scheme
NSE	Navier-Stokes equation
NTE	neutron transport equation
ODE	ordinary differential equation
PCQSM	point corrector quasi static method
PK	point kinetics
RRMSE	relative root mean squared error

Symbol	Definition	Unit
C	Precursor concentration	$[m^{-3}]$
C_p	Specific heat	$[Jkg^{-1}K^{-1}]$
Da	Damkohler number	$[-]$
e_i	Discrete lattice velocity	$[ms^{-1}]$
E	Energy	$[J]$
\mathbf{F}	Force	$[N]$
f_i	Distribution function for momentum field	$[-]$
h_i	Distribution function for temperature field	$[-]$
\mathbf{J}	Neutron current	$[m^2s^{-1}]$
k	Multiplication factor	$[-]$
m	Precursor amplitude function	$[-]$
n	Neutron amplitude function	$[-]$
n_i	Distribution function for neutron field	$[-]$
Nu	Nusselt number	$[-]$
P	Pressure	$[Pa]$
Pr	Prandtl number	$[-]$
r	Space coordinate	$[m]$
Re	Reynolds number	$[-]$
Sc	Schmidt number	$[-]$
S_f	Fission source	$[m^3s^{-1}]$
t	Time	$[s]$
T	Temperature	$[K]$
\mathbf{u}	Velocity	$[ms^{-1}]$

Symbol	Definition	Unit
v	Velocity magnitude	$[ms^{-1}]$
c_i	Distribution function for precursor field	$[-]$
c_s	Speed of sound	$[ms^{-1}]$
D_g	Neutron diffusion coefficient	$[m]$
D_p	Precursor diffusion coefficient	$[m^2s^{-1}]$

Greek symbol	Definition	Unit
α	Thermal diffusivity	$[m^2s^{-1}]$
β	Delayed neutron fraction	$[-]$
χ	Neutron spectrum	$[-]$
ϵ	Discrepancy	$[\%]$
γ	Heat transfer coefficient	$[WK^{-1}m^{-3}]$
λ	Decay constant	$[s^{-1}]$
ν	Kinematic viscosity	$[m^2s^{-1}]$
ν	Average number of neutrons produced per fission	$[-]$
Ω	BGK operator	$[-]$
Φ	Conjugate neutron flux	$[m^{-2}s^{-1}]$
ϕ	Scalar neutron flux	$[m^{-2}s^{-1}]$
φ	Angular neutron flux	$[m^{-2}s^{-1}]$
ρ	Density	$[kgm^{-3}]$
ρ	Reactivity	$[-]$
Σ	Cross-section	$[m^{-1}]$
σ	Stress tensor	$[Pa]$
τ	Relaxation time	$[s]$
ξ	Particle velocity	$[ms^{-1}]$
$\hat{\Omega}$	Angular direction	$[-]$

Subscript	Definition
adj	Adjacent
α	Angular direction
b	Boundary
c	For distribution function c
d	Delayed precursor family
eff	Effective
f	For distribution function f
f	Fission
g	Energy group
$g' \rightarrow g$	Scatter from energy group g' to g
h	For distribution function h
i	Lattice direction
n	For distribution function n
p	Precursors
ref	Reference
s	Scattering
t	Total removal
tot	Total
0	Initial

Superscript Definition	
<i>eq</i>	Equilibrium
<i>D</i>	Delayed
<i>P</i>	Prompt

1

Introduction

In the energy transition a shift towards carbon-free electricity generation is of great importance. Nuclear power is not only crucial for power sector decarbonisation but also plays a vital role in enhancing energy security. It stabilises power grids and complements the intermittent nature of renewable energy sources like wind and solar photo-voltaics, which have an inconsistent output. This is particularly important as wind and solar require the support of dispatchable resources to ensure a continuous and reliable power supply. Over the past 50 years nuclear energy has avoided about 70 Gt of CO₂ emissions globally [28], comparable to the total global emissions of energy combustion and industrial processes of about two years. According to the International Energy Agency, nuclear power needs to expand further to stay on track with ambitions for Net Zero Scenarios by 2050.

Nuclear power faces opposition in many countries. This is often due to safety concerns when people think of the major accidents that have happened in the past, such as the nuclear disaster in Chernobyl in 1986. Further, people are concerned with the large half-lives of radioactive waste produced by nuclear reactors.

Luckily, advancement in nuclear technologies have found ways to cope with these reasonable concerns. Technologies in the nuclear sector aim to mitigate these issues, making nuclear energy a more viable and appealing option for clean energy transitions. A promising technology is the Molten Salt Reactor (MSR), that performs well on both these important concerns. MSRs incorporate advanced passive safety mechanisms, automatically stabilizing the reactor in case of malfunctions, thereby enhancing safety significantly. These reactors notably produce radioactive waste with much shorter half-lives, substantially reducing the duration for safe storage. MSRs can utilise and transform long-lived radioactive waste from other reactors into shorter-lived isotopes, addressing a key challenge in nuclear waste management. This innovative approach positions MSRs as a promising technology in the nuclear sector, aligning with global efforts towards cleaner and safer energy solutions.

1.1. The Molten Salt Fast Reactor

Various classes of molten salt reactors exist and they are often characterised by their neutron spectrum. When one includes a moderator, thermal neutron spectra can be obtained, while when excluding a moderator one obtains fast spectra [32]. In the latter case, one refers to the reactor as a Molten Salt Fast Reactors (MSFR). These represent a class of nuclear reactors offering several advanced features compared to traditional nuclear technologies. MSFRs use molten salt as both fuel and coolant. This dual role of molten salt is central to the operation and safety profile of MSFRs. In MSFRs, the fuel, typically a mixture of fissile uranium fluoride (UF₄) and fertile thorium fluoride (ThF₄), is dissolved in the molten salt, typically lithium fluoride (LiF) [17]. Alternatively, not fluoride salts but chloride salts are used, resulting in slightly different neutron spectra. In figure 1.1 a schematic overview of the design of an MSFR is shown.

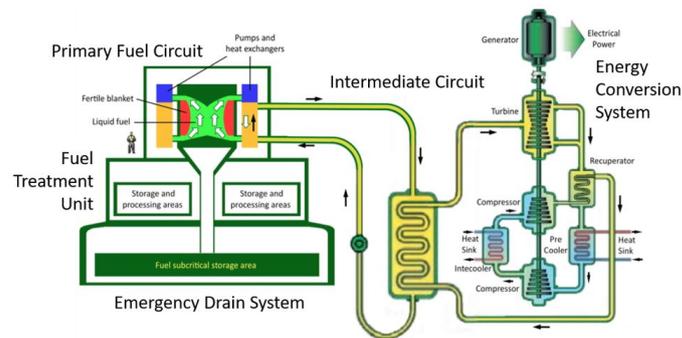


Figure 1.1: Schematic overview of a Molten Salt Fast Reactor [32]

In the SAMOSAFER project, MSFRs have a thermal and electrical power output of 3 GW_{th} and 1.3 GW_e respectively[17]. The average fuel salt temperature is around 1000 K and salt melting point is around 850 K . The core is cylindrical with a radius of around $1 - 1.4\text{ m}$ and height of $1.6 - 2.3\text{ m}$.

The MSFR is a promising technology in the context of both nuclear safety and nuclear waste management [16]. It is unique in its capability of efficiently using thorium due to the fast spectrum of the neutrons in the core. A great advantage of the capability of using thorium results in a substantial reduction in plutonium production, which is long-lived nuclear waste. Consequently, time-spans of waste storage can be highly reduced to only a few centuries, compared to the much longer time-spans required for other types of reactor waste. Further, thorium is abundant in nature, making it an attractive fuel element. These two advantages of using thorium position MSFRs as highly promising and sustainable compared to other reactor types.

Additionally, fuel salt composition can easily be adapted during operation. This results in that the MSFR can operate as both a breeder reactor, when it produces more fissile material than it consumes, and as a burner reactor, when it consumes existing fissile and fertile materials, including waste products from other nuclear reactors such as long-lived plutonium.

Of course, safety is of paramount concern in nuclear reactor design, and MSFRs incorporate several features to enhance safety [17]:

- Draining of the fuel salt (*freeze plug*):
One notable safety feature is the freeze plug. In normal operation, the plug is kept frozen by active cooling. If the reactor overheats, the cooling system fails, causing the plug to melt. This melting allows the molten salt, containing the nuclear fuel, to drain into passively cooled tanks underneath the reactor core, shutting down the nuclear reaction.
- Cleaning of the salt (*extract lanthanide and actinides*):
In an MSFR fuel salt can be purified in two steps. Gaseous and insoluble fission products, are extracted from the reactor core through a technique of gas bubbling. Uranium, actinides, and certain fission products can be separated outside of the reactor core. This leads to a reduction in decay heat as fission products are removed from the core before decay.
- Temperature feedback mechanisms (*thermal expansion and Doppler feedback*):
The liquid salt acts as both coolant and fuel. This results in strong coupling of temperature and flow fields to neutronics compared to solid fuel reactors. Increase in temperature makes the molten salt expand, which leads to a decrease in neutron density and a reduction in the nuclear reaction rate, thereby cooling the reactor. Additionally, increase in temperature results in Doppler effects, that decrease the probability for nuclear reactions to occur, again leading to a reduction in the nuclear reaction rate, thereby cooling the reactor. These temperature feedback mechanisms improve the stability of the reactor, and also eliminate the necessity of control rods in the core [33].

These safety mechanisms are inherent to the strong coupling between neutronics and thermal-hydraulics [35]. However, this strong coupling also makes describing the physics within the core complex, as one should include phenomena such as fluid flow, thermal expansions, cross-section changes, precursor movement, turbulence effects and phase changes. Trying to understand the interactions between all these phenomena results in the necessity for computer tools that are efficient in predicting the

physics at hands. This research aims to develop and validate a model that couples thermal-hydraulics and neutronics through fluid flow, thermal expansions, cross-section changes and precursor movement.

1.2. Review of existing literature and models

Since this project is not only aimed at the development but also the validation of a coupled multi-physics tool for describing the physics in a MSFR, reference results are of great importance. Experimental data for molten salt reactors is available from the results of the Molten Salt Reactor Experiment but this is for the older generation thermal reactor [13]. Limited experimental data is available for the fast spectrum molten salt reactor and thus verification of multi-physics tools for MSFRs is difficult [35]. Results from different tools are therefore validated against each other and their capabilities are compared in a benchmark study by Tiberger et al.: "*Results from a multi-physics numerical benchmark for codes dedicated to molten salt fast reactors*" [35]. This numerical benchmark study will be used throughout this research project to validate the model's performance, as it stands out in its step-by-step approach resulting in easier error identification. The study gradually couples the developed models for the different fields. In phase 0 fields are modeled separately, in phase 1 steady-state coupling is implemented and in phase 2 transient coupling is implemented. In this research project, these phase definitions are also used for consistency. The study will be described in more depth in section 4.1. In the paper, different tools are compared. Before the description of a newly created tool in this research, first the existing coupled tools dedicated to molten-salt reactors are described in section 1.2.1. Afterwards, existing tools that use the lattice Boltzmann method for simulating neutronics are discussed in section 1.2.2.

1.2.1. Existing multi-physics tools for Molten Salt Fast Reactors

Due to the strongly coupled physical fields in a molten salt reactor, conventional codes describing solid fuel reactors are insufficient for describing behaviour in MSFRs. Several research projects have been focused on the development of new, suitable tools that do couple the different phenomena. The codes discussed in this chapter were developed during the SAMOFAR project and their capabilities are compared in the previously described benchmark study by Tiberger et al. [35].

The CNRS code is developed at the LPSC/CNRS-Grenoble. It combines several codes for the different physical phenomena. For solving the neutron transport equations, both the C++ finite volume libraries of OpenFOAM and Monte Carlo Method (MCM) in Serpent are used. For the thermal-hydraulics model, the Navier-Stokes equation is solved using OpenFOAM's PIMPLE algorithm [35].

The PoliMi code, developed by Cervi et al. [5], also uses the finite volume libraries of OpenFOAM. The model implements several methods [5, 6] and solves the neutron diffusion equation using the MCM code in Serpent coupling it to the thermal-hydraulics described by a solver in OpenFOAM. [6].

The PSI code, developed at the Paul Scherrer Institute, couples four sub-solvers. The sub-solvers are dedicated to solving equations for the displacement-based thermal-hydraulics and multi-group neutron diffusion. The first three solvers are based on OpenFOAM. The latter is a MCM simulation in Serpent that exports results to OpenFOAM to couple the physical processes [10].

The last code described in the benchmark by Tiberger et al. [35], developed at the TU Delft, couples the thermal-hydraulics code DGFlows with a radiation transport code PHANTOM- S_N [34].

All codes described above couple the physical phenomena of thermal-hydraulics and neutronics within an MSFR.

1.2.2. Models for neutronics using the lattice Boltzmann method

Recently, steps have been made towards the integration of the multi-physics coupling within one programming method, the lattice Boltzmann method (LBM). LBM is a method widely used for computational fluid dynamics due to its simplicity and parallelizability [46]. Several efforts were made to model the Boltzmann transport equation (BTE) for neutron transport using LBM [3, 46, 42]. Also, frameworks exist for the modelling of the neutron diffusion equation using LBM [44]. Based on previous efforts to use the LBM for radiative transfer problems, Bindra and Patil [3] started their first attempts to use the LBM for both radiative and neutron transport. Disadvantages of their work were the neglect of the collision term and low angular resolution. Wang et al. [46] advance the previous work by regarding the angular directions as the macroscopic velocities specific for the lattice structure. This results in the opportunity to treat the transient BTE as a conservation law, resulting in more accuracy in solving the

multidimensional BTE. Further, using the Chapman-Enskog expansion, the relaxation time of the LBM can be considered a free parameter, that can be used to enhance the stability of the code. The model by Wang et al. [46] is a lattice Boltzmann method for the transient BTE, but the steady-state solution can be obtained when using LBM until convergence. No interactions with other physical phenomena is implemented. The paper can be used as an accurate example of LBM modelling for neutron transport, but the model needs an extension with thermal-hydraulic phenomena.

In LBM the streaming process is only related to adjacent nodes. Solutions for the particle distribution and particle collisions are local and can for each node be computed in parallel [42]. This makes the LBM highly adequate for GPU simulations. A next paper by Wang et al. [42] describes a method that uses both the advantages of GPU and CPU. On CPU the initialisation of the computation is executed, after which the individual computations at the lattice nodes are executed on GPU. The GPU data is stored and CPU performs the judgement of convergence. The paper also describes a similar method for multiple-GPU acceleration. Wang et al. [42] test the model against three benchmarks, all of which only include neutron transport and exclude coupling of neutronics with thermal-hydraulic effects. Further, for the transient simulation the paper uses the predictor-corrector quasi-static method (PCQSM) that comes with the advantage that neutron fluxes, that are solved with high computational costs, have to be solved for larger time-scales (and thus less frequently) as they are considered quasi-static. Time-dependent, zero-dimensional kinetics equations are solved at small time-scales but their computational cost is much lower. The combination of these different time-scales results in the fact that the computational costs can be highly reduced without losing accuracy. This method is performed for neutronics kinetics in static fuel. PCQSM is an established transient calculation method for describing reactor dynamics. Dulla et al. [9] describe the PCQSM and compare its accuracy to a different quasi-static method, and Kooreman et al. [20] showed that the PCQSM can also be accurate when temperature feedback is considered.

In two later papers by Wang et al. [40, 41] the development of a multi-physics code is described. This recent effort to model the Boltzmann transport equations using LBM proves that both physical phenomena playing a part within the reactor core of an MSFR, the thermal-hydraulic fluid dynamics and the neutron transport, can be modelled using the same simulation method. This simplifies the coupling between the phenomena within a reactor core as no switching between modelling methods is required.

The first follow-up paper by Wang et al. [41] describes a method of using LBM to model the coupling between neutronics and heat transfer, ignoring flow effects. A steady-state simulation is considered. Wang et al. have developed a single Lattice Boltzmann Method framework that can accommodate several neutron transport approximations (S_N , SP_3 , and P_1) accurately, coupling the neutronics to heat transport, while neglecting flow effects. This latter approximation is often sufficient for solid fuel reactors, while for MSFRs flow effects play an import role.

The second follow-up paper by Wang et al. [40] develops a method for the full coupling of thermal-hydraulics and neutronics [40]. The method uses transient LBM schemes for the different physical phenomena, but from assessment of the model it is concluded that it only partly validates steady-state simulations. For the neutronics, a finite Boltzmann (FB) scheme is used. This is a special case of LBM where the relaxation parameter is equal to 1 [37]. According to van der Sman et al. [37], the performance of this scheme is not necessarily better than others, but the memory requirements are lower as FB schemes have the advantage that only the conserved fields have to be stored in memory, from which the equilibrium distribution is constructed straightforwardly.

For the neutronics, two groups of neutron particles and six families of delayed neutron precursors are considered in the paper by Wang et al. [40]. The paper does not consider a transient model. The paper documents on results for the fully coupled system in an MSR, using conditions similar to those in the benchmark by Tibergera et al. [35], but simplified. These results are documented but not validated against benchmark results. The deviations in the conditions from the benchmark by Tibergera et al. [35] result in an incomplete validation of the model.

A disadvantage of this last model is that a transport model is used for the neutronics, which results in high computational costs when combining the neutronics with thermal-hydraulics. The model takes into account a separate distribution function for each neutron group with different angular direction α and corresponding to a different energy group g . As an example of the computational expense of this choice: six discrete angular directions and six energy groups already result in 36 separate distribution functions for each lattice direction, not yet taking into account distribution functions for delayed neutron

precursors and thermal-hydraulics. This disadvantage leads to the relevance of examining methods to reduce the number of equations describing the neutronics. Also, the model can be extended to transient situations.

An alternative to solving the neutron transport equation (NTE) is solving the neutron diffusion equation (NDE). A method for solving the NDE using lattice Boltzmann method is described in yet another paper by Wang et al. [44]. Using the NDE has the advantage of lower computational cost compared to solving the NTE since not all angular directions for the neutronics are solved separately [44]. Since the core of an MSFR is homogeneous compared to solid-fueled reactors, diffusion models are expected to provide sufficiently good results [33] as the inaccuracies in the diffusion approximation are most important near interfaces.

In the paper a method for the programming of neutronics using LBM is described that uses a high-order LBM for the multi-group neutron diffusion equation [44]. The model is thorough in giving a general expression for a high-order LBM, and additionally describes both first- and second-order truncations explicitly. Using high-order LBM can result in high accuracy in simulations, while using low-order LBM simplifies the model. The method can be used as a computationally less costly description for the neutronics model compared to the LBM for the NTE. The paper does not consider precursor groups, nor does it consider coupling of neutronics with different fields.

1.3. Thesis goals

This research aims to develop a multi-physics tool for Molten Salt Fast Reactors (MSFRs) using the Lattice Boltzmann Method (LBM) and validate it against existing models. The primary objective is to integrate the neutronics and thermal-hydraulics components. The study utilises the neutronics diffusion LBM from Wang et al. [44] to solve the steady-state NDE, extending it to include precursor sources. Using the NDE facilitates the inclusion of an increased number of neutron energy groups compared to other neutronics LB simulations in literature that use the NTE. The developed steady-state model will simulate conditions detailed in the benchmark Tiberga et al. [35]. Comparative analysis of the outcomes from this newly developed model against those documented in the benchmark study by Tiberga et al. [35] will be conducted. This analysis will highlight deviations and identify potential enhancements for the model.

After validation of the steady-state multi-physics LBM tool, the model will be extended to include transient behaviour in the reactor core, following the approach outlined in Tiberga et al.'s study. This phase includes the implementation of a transient perturbation in the temperature simulation to evaluate the system's transient behavior, facilitating an effective validation against the benchmark study. The transient problem will be simplified by using the PCQSM, to reduce computational efforts.

Research questions

1. **Model development:** *How can a novel multi-physics framework be developed by coupling the newly formulated LBM for the multi-group neutron diffusion equation with the established LBM for Navier-Stokes equation and advection-diffusion equations in laminar conditions, to simulate both steady-state and transient scenarios in a two-dimensional Molten Salt Fast Reactor?*
 - *What stability constraints exist for input parameters of the different physical fields?*
 - *What boundary conditions for neutronics exist and how do their capabilities compare?*
2. **Steady-State Model:** *How does the newly developed LBM framework for neutronics and thermal-hydraulics compare with established benchmarks in Tiberga et al.'s single-physics phase (Phase 0) and steady-state coupling phase (Phase 1) studies?*
 - *How can an iterative scheme for solving the neutron eigenvalue problem be combined with the LBM?*
 - *What deviations exist between the developed model and the benchmark results?*
 - *Which specific physical phenomena contribute to these deviations?*
 - *How can the developed model be refined for improved accuracy in comparison to the benchmarks?*
3. **Transient Model:** *How can the developed LBM framework be extended to include transient behavior?*

- *How can the PCQSM be employed for integrating transient behavior with the LBM?*
- *How does the model align with established benchmark codes in Tiberga et al.'s transient coupling phase (Phase 2)?*

Thesis outline

Chapter 2 of this thesis will describe the theory and governing equations for the physics of neutronics, thermal-hydraulics and their interactions. Methods to solve steady-state and transient neutronics problems are described as well. In Chapter 3 the lattice Boltzmann method is described for numerically solving the multi-physics in a liquid fuel reactor core. Additionally, the numerical implementation of the PCQSM with the LBM is described. The tool is validated for its uncoupled components in Chapter 4 in steady-state situations. In Chapter 5 the components are gradually coupled and the results are discussed in comparison to the results from the multi-physics codes the benchmark by Tiberga et al. [35]. In Chapter 6 the capabilities of the fully coupled tool that integrates the PCQSM and the LBM is discussed for transient situations. Chapter 7 will present the conclusions drawn in this research project and recommend future research.

2

Theory

In the nuclear reactor core of a molten salt reactor, a molten salt is present that acts both as a coolant and as a fuel. This leads to interesting phenomena happening within the salt that differ from the older generation reactors such as heat generation within the coolant, thermal feedback by fuel expansion and convective transport of delayed neutron precursors from flow of the salt. This means that the relevant physical fields, the thermal-hydraulics and the neutronics, are coupled. In this chapter, in section 2.1 general theory on reactor physics is described, after which the governing equations for the separate fields are discussed starting with neutronics in section 2.2 followed by thermal-hydraulics in section 2.3. Subsequently, the physical interaction between the fields is discussed in section 2.4. Finally, the mesoscopic approach to assess the fields is described using kinetic theory in section 2.5, followed by a derivation of the Boltzmann equation.

2.1. Nuclear reactor physics

Within a nuclear reactor core fission reactions occur [8]. These fission reactions occur when heavy nuclei split into, most often, two nuclei releasing energy and several neutrons. This can happen spontaneously but the rate of spontaneous fission is very slow. In a reactor core fission of heavy nuclei is induced by incident neutrons, that make the nuclei unstably heavier, after which the heavy nucleus falls apart. Fission is just one of the types of neutron-nuclear interactions that can occur in the core. Other neutron-nuclear reaction types that are relevant are scattering, where neutrons change their angular direction and/or energy, and absorption, where neutrons are absorbed by processes such as radiative capture.

The probability that neutron-nuclear interactions take place is characterised by the macroscopic nuclear cross-section, Σ_x , that is the product of the microscopic cross-section σ_x , which is medium and energy specific, and of the number density of the isotopes in the medium N . The subscript x represents the reaction types, such as fission, absorption or scattering.

Fission is categorised as a type of absorption. In the absorption process a neutron is absorbed by a nucleus, forming a compound nucleus that subsequently decays while emitting particles. Whether a compound nucleus is formed highly depends on the energy of the incident neutrons. When the incident neutron energy matches an energy level of the compound nucleus the probability of absorption is significantly higher. Therefore, we observe so-called resonance peaks at the matching incident neutron energies. Furthermore, compound nuclei are in thermal motion and the relative speed of the compound nuclei gives rise to the so-called Doppler effect, where the resonance peaks broaden for increasing temperature, while peak magnitudes decrease. In conclusion, the probability of nuclear reactions to occur in the reactor core is affected by temperature through changes in the number density of isotopes, changing Σ_x , and through Doppler effects.

In the nuclear core, neutrons created in a fission reaction induce new fission reactions, resulting in a nuclear chain reaction. The number of neutrons in the core should remain stable for safe and efficient operation, and the reactor core is designed such that one neutron emitted in a previous fission event induces another fission event, and the other neutrons from the previous fission event are absorbed or escape the core. This is where the multiplication factor k plays a role, which represents the number of neutrons in one generation divided by the number of neutrons of a previous generation. Is $k = 1$, the

reactor core is called critical as the number of neutrons in the core is constant and the chain reaction is sustained. For $k > 1$ the number of neutrons grows and for $k < 1$ the number of neutrons decreases and the core is called super- and sub-critical respectively. An important task in reactor design is thus to find for which conditions the core is critical. Another measure related to the multiplication factor is the reactivity ρ represented as

$$\rho = \frac{k - 1}{k} , \quad (2.1)$$

which is a measure for the deviation of the core from criticality [8]. Changing core composition changes the neutron population and thus k and ρ .

When a fission reaction takes place, neutrons are emitted with a distribution of energies, represented by the fission spectrum $\chi(E)$. The average number of neutrons emitted per fission reaction is represented by ν . Most neutrons appear instantaneously, within 10^{-14} s after the fission event. Some neutrons however, are products of radioactive decay of fission products. These radioactive fission products that still emit neutrons at a delayed rate (ranging from milliseconds to minutes) are referred to as delayed neutron precursors and neutrons emitted are referred to as delayed neutrons. The delayed precursors are often grouped in families that are characterised by their half-lives. The fraction of neutrons per fission event that appear in precursor family d is represented by β_d and total fraction of neutrons that are delayed per fission event is represented by $\beta_{tot} = \sum_d \beta_d$.

In steady-state situations in molten salt reactors, delayed precursors can affect the distribution of neutrons throughout the core, as precursors can be transported through the salt and emit their neutrons at different locations than they were originally created. When one assesses reactor time behaviour delayed neutrons play an important role since they elongate the reactor period making reactor control possible at larger and more practical time-scales.

2.2. Neutron transport and neutron diffusion

To describe the behaviour of neutrons one usually investigates neutron transport, based on balancing the mechanisms playing a role in the creation and removal of neutrons with a certain energy E' , travelling in a certain direction $\hat{\Omega}'$, located within an infinitesimal volume element at location \mathbf{r} at time t [8]. The neutron transport equation (NTE) is given as

$$\frac{1}{v} \frac{\partial \varphi}{\partial t} + \hat{\Omega} \cdot \nabla \varphi + \Sigma_t(\mathbf{r}, E) \varphi(\mathbf{r}, E, \hat{\Omega}, t) = \int_{4\pi} d\hat{\Omega}' \int_0^\infty dE' \Sigma_s(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi(\mathbf{r}, E', \hat{\Omega}', t) + S_f(\mathbf{r}, E', \hat{\Omega}, t) . \quad (2.2)$$

φ is the angular neutron flux, a measure for the number of neutrons crossing unit area of the medium in unit time. In this equation we recognise the terms from left to right as the time rate of change of number of neutrons, the leakage into or out of the volume element, the total removal of neutrons (by absorption or collision), the addition of neutrons of different E or $\hat{\Omega}$ scattering towards the E or $\hat{\Omega}$ of interest and finally the addition of fission neutrons described as

$$S_f(\mathbf{r}, E, \hat{\Omega}, t) = \frac{\chi(E)}{4\pi} \int_{4\pi} d\hat{\Omega}' \int_0^\infty dE' \nu(E') \Sigma_f(E') \varphi(\mathbf{r}, E', \hat{\Omega}', t) . \quad (2.3)$$

The integral represents the total number of neutrons that is emitted when neutrons with energy E' travelling in angular direction $\hat{\Omega}'$ undergo fission, and the prefactor takes only the fraction of neutrons that ends up with energy E travelling in direction $\hat{\Omega}$.

The NTE written in equation 2.2 must be solved for neutrons with all possible energies E travelling in all possible directions $\hat{\Omega}$. This way of describing the neutron transport results in a very large number of equations since each possible E and $\hat{\Omega}$ have their own equation. To simplify the neutronics equations, the neutron energies are discretised into energy bands. Neutrons are classified into groups corresponding to these bands, each group being characterised by a group-energy, denoted as E_g . Within an energy group, the neutrons share the same neutronic parameters, such as cross-sections and diffusion constants.

To further simplify the neutronics equations, the angular directions of the neutrons can also be discretised. This is called a discrete ordinate treatment of angle and gives rise to the S_N equations, where

N represents the number of discretised angular directions. Alternatively, one can use functional expansion to discretise the angular variables, resulting in the P_N equations [8]. A useful expansion of the angular flux is found in the P_1 approximation, where the angular flux is expanded in Legendre polynomials in $\cos\theta$ and omitting high order terms to obtain the P_1 equations. Details on this P_1 approximation and the derivation can be found in literature such as [8].

Combining the P_1 approximation with the assumption that neutron sources in a reactor are isotropic and the assumption that neutron current changes slowly over time compared to collision of neutrons with surrounding medium, one can rewrite the neutron transport equation and introduce the neutron diffusion coefficient D_g . This neutron diffusion coefficient is a function of the total cross-section, the scattering cross-section and the average scattering cosine $\overline{\mu_0}$. Applying these simplifications, it can be found that the NTE can be simplified by the diffusion approximation. In that case, one can use the scalar neutron flux $\phi(E, \mathbf{r}, t)$, independent of angular direction. The motion of neutrons is treated as a diffusion process and the neutron diffusion equation (NDE) is used. This reduces the number of equations significantly, and this approximation is often adequate for reactor applications, more specifically in homogeneous cores. The diffusion approximation follows from four different approximations: neutrons are grouped by their energies, the angular flux is linearly isotropic, sources are isotropic and neutron current density changes slowly compared to the collision time.

Considering discretised energy groups, this leads to the multi-group NDE [8], for each energy group g written as

$$\frac{1}{v_g} \frac{\partial \phi_g}{\partial t} - \nabla \cdot (D_g \nabla \phi_g) = -\Sigma_{t,g} \phi_g + \chi_g \sum_{g'} (\nu \Sigma_f)_{g'} \phi_{g'} + \sum_{g'} \Sigma_{s,g'-g} \phi_{g'} . \quad (2.4)$$

Here, a neutron diffusion coefficient D_g is used. This constant is proportional to the mean free path of transport of a neutron group. One can recognise on the right-hand side simplified versions of the terms written in equation 2.2, namely from left to right a total removal term, by absorption and scattering, a fission term and a scattering term. Equation 2.4 is the transient NDE.

In neutron transport theory a vacuum boundary condition is interpreted as that no neutrons can enter the domain of interest at the specific boundary. It should be noted that in the diffusion approximation the scalar neutron flux is used, for which terms as "incoming" or "outgoing" have no meaning since it is a scalar quantity. Therefore, the vacuum boundary condition for the scalar neutron flux is expressed as that if we would extrapolate the scalar flux beyond the vacuum boundary the scalar flux would go to zero at a point slightly further away. The location in space where the scalar neutron flux vanishes should be extrapolated to a location $\tilde{x}_{boundary}$ proportional to the diffusion constant [8]

$$\tilde{x}_{boundary,g} = x_{boundary} + 2.1312 D_g . \quad (2.5)$$

See the graph below for a representation of this extrapolation. Note that for each energy group the diffusion coefficient might be different and the extrapolated boundary is then thus at a different point in space.

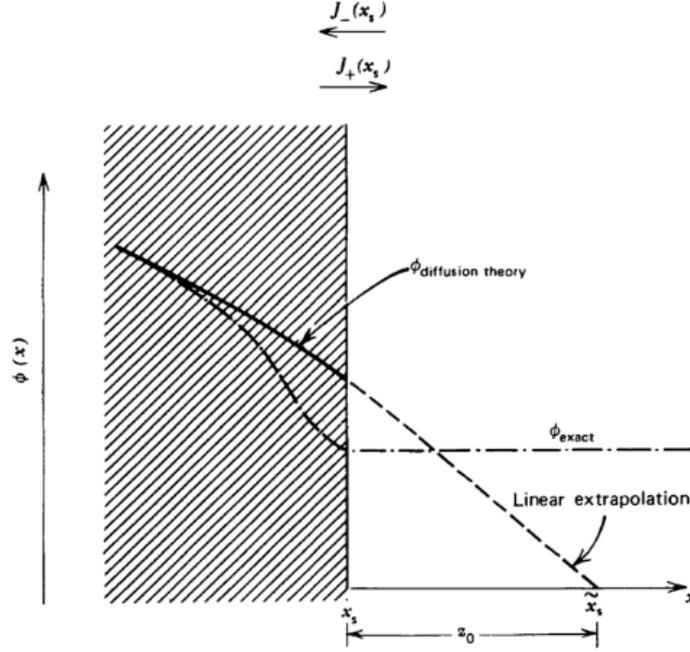


Figure 2.1: Difference in behaviour of the exact scalar neutron flux and the scalar neutron flux in the diffusion approximation at a vacuum boundary [8].

For a vacuum boundary, $\phi(\tilde{x}_{boundary}) = 0$. It is important to emphasise the fact that that does not necessarily mean that the neutron current density \mathbf{J} , the net flow rate of neutrons through a given surface, is zero at the extrapolated boundary as well. In diffusion theory the neutron current density is approximated as

$$\mathbf{J}(\mathbf{r}, t) = -D\nabla\phi(\mathbf{r}, t) . \quad (2.6)$$

For a vacuum condition, while the scalar flux is zero at the extrapolated boundary, there can still be a non-zero gradient of the flux, leading to a non-zero neutron current.

The system of equations above, considers the case where neutrons are only emitted promptly. But as described previously, neutrons are often emitted by precursors after a time-delay. The advection-diffusion equation for the concentration of precursor family d in a non-static fuel field is given by [40]

$$\frac{\partial C_d}{\partial t} = -\nabla \cdot (\mathbf{u}C_d) + \nabla \cdot (D_p\nabla C_d) - \lambda_d C_d + \frac{\beta_d}{k_{eff}} \sum_g (\nu\Sigma_f\phi)_g . \quad (2.7)$$

Here the term on the left-hand side is a convective term, the first term on the right represents diffusion of the precursors, the second term on the right the decay of the precursors and the last term on the right is the creation of precursors by fission reactions. The decay of precursors results in additional neutrons in the core and the NDE in equation 2.4 is adjusted to obtain

$$\frac{1}{v_g} \frac{\partial \phi_g}{\partial t} - \nabla \cdot (D_g\nabla\phi_g) = -\Sigma_{t,g}\phi_g + \frac{\chi_g^p(1 - \beta_{tot})}{k_{eff}} \sum_{g'} (\nu\Sigma_f)_{g'}\phi_{g'} + \chi_g^d \sum_d \lambda_d C_d + \sum_{g'} \Sigma_{s,g'-g}\phi_{g'} . \quad (2.8)$$

2.2.1. Steady-state neutronics: reactor criticality calculation

As stated previously, a reactor core should be designed for criticality. Criticality of the core implies that the behaviour of neutrons does not change over time, and thus the scalar neutron flux is only a function of space. This means that equation 2.8 reduces to

$$-\nabla \cdot (D_g\nabla\phi_g) = -\Sigma_{t,g}\phi_g + \chi_g^p(1 - \beta_{tot}) \sum_{g'} (\nu\Sigma_f)_{g'}\phi_{g'} + \chi_g^d \sum_d \lambda_d C_d + \sum_{g'} \Sigma_{s,g'-g}\phi_{g'} , \quad (2.9)$$

and equation 2.7 reduces to

$$\nabla \cdot (\mathbf{u}C_d) = \nabla \cdot (D_p\nabla C_d) - \lambda_d C_d + \beta_d \sum_g (\nu\Sigma_f\phi)_g . \quad (2.10)$$

This system of equations, together with provided relevant boundary conditions, has a solution when the composition of the core represents a critical system. However, often the problem at hand is to assess what composition results in criticality. For this reason, equation 2.9 is studied as an eigenvalue problem

$$-\nabla \cdot (D_g \nabla \phi_g) = -\Sigma_{t,g} \phi_g + \frac{\chi_g^p (1 - \beta_{tot})}{k_{eff}} \sum_{g'} (\nu \Sigma_f)_{g'} \phi_{g'} + \chi_g^d \sum_d \lambda_d C_d + \sum_{g'} \Sigma_{s,g'-g} \phi_{g'} \quad (2.11)$$

$$\nabla \cdot (\mathbf{u} C_d) = \nabla \cdot (D_p \nabla C_d) - \lambda_d C_d + \frac{\beta_d}{k_{eff}} \sum_g (\nu \Sigma_f \phi)_g \quad (2.12)$$

where the eigenvalue k_{eff} represents the previously described multiplication factor. The found fluxes ϕ_g and concentrations C_d are the eigenvectors of the system of equations.

2.2.2. Power method for k-eigenvalue problem

Together, equations 2.11 and 2.12 form a linear system of equations, that can be solved for eigenvalue k_{eff} and eigenvectors C_d and ϕ_g . It is convenient to write this system of equations in matrix form

$$\mathbf{M}\boldsymbol{\gamma} = \frac{1}{k_{eff}} \mathbf{F}\boldsymbol{\gamma} \quad (2.13)$$

with, for the example of 2 precursor families and 2 neutron energy groups, for matrix \mathbf{M}

$$\mathbf{M} = \begin{bmatrix} \nabla \cdot \mathbf{u} - \nabla \cdot (D_p \nabla) + \lambda_1 & 0 & 0 & 0 \\ 0 & \nabla \cdot \mathbf{u} - \nabla \cdot (D_p \nabla) + \lambda_2 & 0 & 0 \\ -\nabla \cdot D_1 \nabla + \Sigma_{t,1} - \Sigma_{s,1 \rightarrow 1} & 0 & \chi_1^d \lambda_1 & \chi_2^d \lambda_2 \\ \Sigma_{s,1 \rightarrow 2} & -\nabla \cdot D_2 \nabla + \Sigma_{t,2} - \Sigma_{s,2 \rightarrow 2} & \chi_1^d \lambda_1 & \chi_2^d \lambda_2 \end{bmatrix} \quad (2.14)$$

$$\boldsymbol{\gamma} = \begin{bmatrix} C_1 \\ C_2 \\ \phi_1 \\ \phi_2 \end{bmatrix} \quad (2.15)$$

$$\mathbf{F} = \begin{bmatrix} \beta_1 \nu_1 \Sigma_{f,1} & \beta_1 \nu_2 \Sigma_{f,2} & 0 & 0 \\ \beta_2 \nu_1 \Sigma_{f,1} & \beta_2 \nu_2 \Sigma_{f,2} & 0 & 0 \\ \chi_1^p (1 - \beta_{tot}) \nu_1 \Sigma_{f,1} & \chi_2^p (1 - \beta_{tot}) \nu_2 \Sigma_{f,2} & 0 & 0 \\ \chi_1^p (1 - \beta_{tot}) \nu_1 \Sigma_{f,1} & \chi_2^p (1 - \beta_{tot}) \nu_2 \Sigma_{f,2} & 0 & 0 \end{bmatrix} \quad (2.16)$$

These matrices can be rewritten more generally in the form of block matrices in the following way

$$\begin{bmatrix} A_{CC}(\mathbf{u}) & 0 \\ A_{\phi\phi} & A_{\phi C} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi} \\ \mathbf{C} \end{bmatrix} = \frac{1}{k_{eff}} \begin{bmatrix} A_{\phi\phi}^F & 0 \\ A_{C\phi}^F & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi} \\ \mathbf{C} \end{bmatrix} \quad (2.17)$$

with A_{CC} an operator contributing and acting only on the precursor concentrations, $A_{\phi\phi}$ an operator contributing and acting only on neutron fluxes and $A_{\phi C}$ an operator contributing to neutron fluxes but acting on precursor concentrations. Analogously, $A_{\phi\phi}^F$ is a fission operator contributing and acting only on neutron fluxes and $A_{C\phi}^F$ a term contributing to precursor concentration but acting on neutron fluxes. It can now be seen that the matrix on the left-hand side, originally matrix \mathbf{M} , is lower block triangular. We can use the power method to solve this matrix problem. It can be shown that one can find as a solution to this problem a maximum real and positive eigenvalue with a corresponding unique and non-negative eigenvector at every location r . Performing the power method will result in convergence to this maximum eigenvalue and corresponding eigenvectors [8].

The power method starts with initiating the right-hand side of equation 2.17 at iteration number $l = 0$ by setting the multiplication factor k_{eff}^0 and the fission source vector S^0 at a randomly chosen value and solve the in-homogeneous matrix equation

$$\mathbf{M}\boldsymbol{\gamma}^{(l+1)} = \frac{1}{k_{eff}^{(l)}} S^{(l)} \quad (2.18)$$

for $\boldsymbol{\gamma}^{(l+1)}$. This found value for $\boldsymbol{\gamma}^{(l+1)}$ is used to determine a new fission source vector $S^{(l+1)}$ using

$$S^{(l+1)} = \mathbf{F}\gamma^{(l+1)} \quad (2.19)$$

and a new eigenvalue $k_{eff}^{(l+1)}$ using

$$k_{eff}^{(l+1)} = \frac{\int dV S^{(l+1)}}{\frac{1}{k_{eff}^{(l)}} \int dV S^{(l)}} \quad (2.20)$$

This scheme is repeated l iteration steps, until convergence, e.g. when the difference of $k_{eff}^{(l)}$ and $k_{eff}^{(l+1)}$ gets smaller than an imposed criterion. This power method converges to the eigenvalue of largest magnitude. It is sometimes referred to as "fission source iteration" or "outer iteration", because often an additional iteration scheme is used to determine $\gamma^{(l+1)}$, referred to as "flux iteration" or "inner iteration" [38]. Thus, the outer iteration is performed using the power method, while as we will later see the inner iteration will be performed using the lattice Boltzmann method.

2.2.3. Transient behaviour in a nuclear reactor core: the quasi-static method

In time-dependent situations one has to solve the system of equations 2.7 and 2.8. It is common to factorise the neutron flux and precursor concentrations in slowly changing shape functions, $\tilde{\phi}_g(\mathbf{r}, t)$, and fast changing amplitude function, $n(t)$, [43],

$$\phi_g(\mathbf{r}, t) = \tilde{\phi}_g(\mathbf{r}, t)n(t) \quad , \quad (2.21)$$

Both $\tilde{\phi}_g$ and n are functions of time, and this factorisation itself is not an approximation [20]. This factorisation results in the fact that one can solve for the time-dependent amplitude functions separately on short time-scales, and solve for the slowly changing time- and space-dependent shape functions on relatively large time-scales. This is called a quasi-static method, as one assumes the shape function to vary slowly and remains unchanged within the small time-scales. For simplicity the operators that act on the neutron fluxes are shortened as [29, 31]

$$M_g \phi_g = -\nabla \cdot D_g \nabla \phi_g + \Sigma_{t,g} \phi_g - \sum_{g'} \Sigma_{s,g'-g} \phi_{g'} \quad (2.22)$$

$$F_g^P \phi_g = \chi_g^p (1 - \beta_{tot}) \sum_{g'} (\nu \Sigma_f)_{g'} \phi_{g'} \quad (2.23)$$

$$S_g^D = \chi_g^d \sum_d \lambda_d C_d \quad (2.24)$$

$$F_g^D \phi_g = \sum_d F_{d,g}^D \phi_g \quad (2.25)$$

$$F_{d,g}^D \phi_g = \chi_g^d \beta_d \sum_g (\nu \Sigma_f \phi)_g \quad (2.26)$$

$$F_g \phi_g = (F_g^P + F_g^D) \phi_g \quad (2.27)$$

Substituting these operators and the factorised expressions from equation 2.21 for the flux into equation 2.8 and multiplying with a weighting function Φ_g^* , that will be specified later, results in

$$\Phi_g^* \frac{1}{v_g} \tilde{\phi}_g \frac{\partial n}{\partial t} + \Phi_g^* \frac{1}{v_g} n \frac{\partial \tilde{\phi}_g}{\partial t} = \Phi_g^* (F_g^P - M_g) \tilde{\phi}_g n + \Phi_g^* S_g^D \quad . \quad (2.28)$$

Integrating over the entire volume results in the expression below, where notation $\langle x, y \rangle$ is used for the inner product of x and y and integration of this inner product over the entire domain,

$$\langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle \frac{\partial n}{\partial t} + \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle n \frac{\partial \tilde{\phi}_g}{\partial t} = \langle \Phi_g^*, (F_g^D - M_g) \tilde{\phi}_g \rangle n + \langle \Phi_g^*, S_g^D \rangle \quad . \quad (2.29)$$

Performing similar steps for equation 2.7, assuming no convection and diffusion terms [26], results in

$$\frac{\partial \langle \Phi_g^*, \chi_g^d C_d \rangle}{\partial t} = \langle \Phi_g^*, F_g^D \tilde{\phi}_g \rangle n - \lambda_d \langle \Phi_g^*, \chi_g^d C_d \rangle \quad . \quad (2.30)$$

This static fuel assumption is justified by the fact that convection and diffusion time-scales for the precursors are generally much longer than the change of precursor concentrations [22]. Defining m_d as

$$m_d = \frac{\sum_g \langle \Phi_g^*, \chi_g^d C_d \rangle}{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle} , \quad (2.31)$$

and subsequently dividing both equation 2.29 and 2.30 by $\langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle$ and summing over all neutron groups results in

$$\frac{\partial n}{\partial t} = \frac{\sum_g \langle \Phi_g^*, (F_g^P - M_g) \tilde{\phi}_g \rangle}{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle} n + \sum_d \lambda_d m_d \quad \& \quad \frac{\partial m_d}{\partial t} = \frac{\sum_g \langle \Phi_g^*, F_{d,g}^D \tilde{\phi}_g \rangle}{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle} n - \lambda_d m_d \quad (2.32)$$

with the assumption that the change of the flux shape functions with time is negligible compared to the change in amplitude function, eliminating the second term in equation 2.29 using $\frac{\partial \tilde{\phi}_g}{\partial t} = 0$. Here a form of the point kinetics equations (PK equations) is found, where the point kinetics parameters are expressed as functions of the slowly changing shape functions. The equations are rewritten as [43]

$$\frac{dn(t)}{dt} = \frac{\rho(t) - \beta(t)}{\Lambda(t)} n(t) + \sum_d \lambda_d m_d(t) \quad \& \quad \frac{dm_d(t)}{dt} = \frac{\beta_d(t)}{\Lambda(t)} n(t) + \lambda_d m_d(t) , \quad (2.33)$$

and expressions for the point kinetics parameters (PK parameters) $\rho(t)$, $\beta(t)$, $\beta_d(t)$ and $\Lambda(t)$ are defined as [31]

$$\rho(t) = \frac{\sum_g \langle \Phi_g^*, (F_g - M_g) \tilde{\phi}_g \rangle}{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle} , \quad (2.34)$$

$$\beta_d(t) = \frac{\sum_g \langle \Phi_g^*, F_{d,g}^D \tilde{\phi}_g \rangle}{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle} , \quad (2.35)$$

$$\beta(t) = \sum_d \beta_d(t) , \quad (2.36)$$

$$\Lambda(t) = \frac{\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle}{\sum_g \langle \Phi_g^*, F_g \tilde{\phi}_g \rangle} . \quad (2.37)$$

The weighting function $\Phi_0^*(\mathbf{r})$ that one repeatedly finds in the equations above is the conjugate neutron flux, defined as the solution to the steady-state neutron diffusion equation at the initial time [9, 20]

$$-\nabla \cdot (D_g \nabla \Phi_g^*) = -\Sigma_{t,g} \Phi_g^* + \frac{\chi_g^p (1 - \beta_{tot})}{k_{eff}} \sum_{g'} (\nu \Sigma_f)_{g'} \Phi_g^* + \chi_g^d \sum_d \lambda_d C_d + \sum_{g'} \Sigma_{s,g'-g} \Phi_g^* . \quad (2.38)$$

The conjugate neutron flux is used as a weighting function and this ensures the uniqueness of the factorisation in equation 2.21 [9]. The choice for using the initial steady-state neutron flux is made as to reduce the error that comes from solving the shape function at larger time-intervals. The conjugate neutron flux is normalised using the condition

$$\sum_g \langle \Phi_g^*, \frac{1}{v_g} \tilde{\phi}_g \rangle = 1 . \quad (2.39)$$

The time-dependent reactivity is often considered the most important PK parameter as its change in time is generally responsible for the change in reactor power [31]. It not only contains terms for production of neutrons, but also the leakage of neutrons. Often the change in leakage of neutrons can be controlled and thus this term can be tweaked in real-life cases to ensure safe operating conditions, for instance by increasing absorption of neutrons when multiplication rises. Further, the expression in equation 2.34 contains all neutronics parameters that are affected by thermal-hydraulic changes in

the reactor, and these changes are imposed on the neutronics by this key parameter, among other contributing mechanisms that affect the shape-function.

Not surprisingly, for a critical system the transient behaviour is not very interesting as the fluxes are unchanging. However, it becomes interesting when one perturbs the system and assesses the behaviour of the reactor core in response. When assessing this response, several time-scales are important. Prompt neutrons respond almost instantaneously to changes in the core, while precursors with the longest half-lives can respond sometimes only after about 80 seconds. The system is often changed by inserting reactivity, for instance when reducing the absorption of the reactor.

When the reactivity insertion is small, $\rho < \beta$, the so-called prompt term $\frac{\rho - \beta}{\Lambda} n(t)$ is negative. The positive delayed term $\sum_d \lambda_d m_d$ generally balances this reactivity insertion out and the amplitudes change. For small reactivity insertions the response of a reactor core is determined by both prompt and delayed neutrons. When reactivity insertions are large, $\rho \geq \beta$, the prompt term $\frac{\rho - \beta}{\Lambda} n(t)$ is positive and the prompt neutrons are more important than the delayed neutrons. This is called prompt critical, as the reactor is critical already on prompt neutrons alone, without even considering the delayed neutron addition in the core. For very large reactivity insertion $\rho \gg \beta$ the reactor response is almost entirely determined by the behavior of the prompt neutrons. In reactor design prompt criticality should be avoided at all times since in that case reactor periods are way too small for practical safety control.

2.3. The thermal-hydraulics in a nuclear reactor core

The heat generation from fission affects the temperature in a reactor core. Further, in a molten salt reactor fuel is liquid and might thus flow throughout the core. Therefore the thermal-hydraulics in the molten salt reactor should be studied as it can affect the reactor behaviour significantly.

2.3.1. Fluid dynamics

One can describe the behaviour of a fluid through its conservation of mass and momentum while treating the fluid as a continuum. Within this continuum assumption, macroscopic characteristics like density, velocity, and temperature are well-defined within infinitesimal volume elements. In the continuity assumption, this volume is small compared to the size of the entire system, but large compared to the size of the molecules of the system [24].

Let us consider the change of macroscopic properties of such an infinitesimal volume element as described above and assess what phenomena can be responsible for this change. Any change of mass can only be a result of mass flowing into or out of the volume element, as we know mass is a conserved quantity. The continuity equation in fluid dynamics follows from this conservation law and reads

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.40)$$

which in the case of an incompressible fluid reduces to

$$\nabla \cdot \mathbf{u} = 0 \quad (2.41)$$

Any change of net momentum must come from either in- or outflow of momentum, pressure differences or body forces acting on the volume element. Considering each of these contributions to momentum change one obtains the Euler equation for an ideal fluid, i.e. without viscosity and compressibility effects, that reads

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla P + \mathbf{F} \quad (2.42)$$

In a more general form, i.e. for non-ideal fluid, this equation above is written as the Cauchy momentum equation and reads

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla \cdot \boldsymbol{\sigma} + \mathbf{F} \quad (2.43)$$

with $\boldsymbol{\sigma}$ being a total stress tensor, containing pressure and viscosity contributions [21]. Considering incompressible flow and constant viscosity, this can be simplified to the incompressible Navier-Stokes equation (NSE)

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla P + \rho \nu \nabla \cdot (\nabla \mathbf{u}) + \mathbf{F}_{body} \quad (2.44)$$

2.3.2. Temperature field

In addition to the conservation of mass and momentum, energy is also conserved. The overall energy can be divided into internal and mechanical components. Alterations in internal energy occur through the in- or outflow of energy or through the release or removal of internal energy, for instance by phase change or reactions of particles. The total energy conservation in a fluid element then leads to the heat equation [21] given as

$$\frac{\partial T}{\partial t} = \mathbf{u} \cdot \nabla T + \nabla \cdot (\alpha \nabla T) + \frac{q}{\rho C_p} . \quad (2.45)$$

This equation holds for situations in which the following assumptions are made: first, we treat the fluid as incompressible, avoiding the need to consider density changes. Second, we do not take into account viscous dissipation, which while important in certain flow conditions, is not relevant for our current analysis of laminar flows. Third, we assume no presence of shock waves and thus free from rapid, discontinuous changes in pressure and density. Lastly, we apply Fourier's Law for modeling heat diffusion, linking the temperature gradient to the heat flow, which simplifies the understanding of heat transfer in the fluid. In the equation the first term on the right-hand-side describes heat convection, the second term describes heat diffusion and the last term is a heat source.

2.4. Interaction between fields

The neutronics and thermal-hydraulics fields impact each other. Fluid dynamics affect the temperature field and the precursor field by convection, which is seen through the presence of vector \mathbf{u} in equation 2.45 and in equation 2.12 . The temperature field affects fluid dynamics through a buoyancy force when temperature differences are present in the fluid. Buoyancy can be modelled using the Boussinesq approximation written as [21]

$$\mathbf{F}_{buoyancy} = -\beta_{thermal} \rho_0 (T - T_{ref}) \mathbf{g} . \quad (2.46)$$

In this approximation the change of density $\Delta \rho = -\beta_{thermal} \rho_0 (T - T_{ref})$ is linearised. The Navier-Stokes equation then becomes

$$\rho_0 \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla P + \rho_0 \nu \nabla \cdot (\nabla \mathbf{u}) + \mathbf{F}_{buoyancy} , \quad (2.47)$$

with ρ_0 the reference density of the fluid without thermal expansion effects [36].

The temperature field affects the neutronics by salt-expansion feedback, scaling the neutronic parameters diffusion constant D and cross-sections Σ according to [35]

$$\Sigma(T) = \Sigma(T_{ref}) \frac{\rho(T)}{\rho(T_{ref})} , \quad (2.48)$$

and

$$D(T) = D(T_{ref}) \frac{\rho(T_{ref})}{\rho(T)} . \quad (2.49)$$

If the Doppler effect is to be taken into account as well, cross-sections are often numerically corrected by [2, 33, 39]

$$\Sigma(T) = \left[\Sigma(T_{ref}) + \alpha_r \ln \left(\frac{T}{T_{ref}} \right) \right] \frac{\rho(T)}{\rho(T_{ref})} . \quad (2.50)$$

Lastly, the neutronics impact the temperature field by the release of fission heat in fission reactions with as heat source q in equation 2.45

$$q = E_{fission} \sum_g \Sigma_{f,g} \phi_g . \quad (2.51)$$

2.5. From macroscopic equations to the Boltzmann equation: kinetic theory

In kinetic theory particles are considered on a mesoscopic scale analysing particle distributions, in contrast to a microscopic scale analysing individual particles. The equations in kinetic theory describe

dilute gases and were first proposed by Ludwig Boltzmann [8]. To fully describe a system of particles as time t passes one takes into account the particle's rest mass m , its path in space r and its velocity ξ [4]. In kinetic theory we use for this purpose the distribution function $f(r, \xi, t)$ which represents the density of particles at location r , with particle velocity ξ at time t . The distribution function f is connected to the macroscopic variables through moments, which are integrals over velocity space involving f weighted over functions of the particle velocities ξ [21].

Without any interaction with the outside of the particle system after some time the particle system will reach an equilibrium state and the distribution function converges to an equilibrium distribution function f^{eq} .

The distribution function's evolution over time can be described by

$$\frac{df}{dt} = \left(\frac{\partial f}{\partial t} \right) \frac{dt}{dt} + \left(\frac{\partial f}{\partial x_\alpha} \right) \frac{dx_\alpha}{dt} + \left(\frac{\partial f}{\partial \xi_\alpha} \right) \frac{d\xi_\alpha}{dt}, \quad (2.52)$$

where one recognises $\frac{dt}{dt} = 1$, $\frac{dx_\alpha}{dt} = \xi_\alpha$ and $\frac{d\xi_\alpha}{dt} = F_\alpha/\rho$ to obtain the Boltzmann equation (BE)

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \xi_\alpha \frac{\partial f}{\partial x_\alpha} + \frac{F_\alpha}{\rho} \frac{\partial f}{\partial \xi_\alpha}. \quad (2.53)$$

The left-hand side is often considered a source term that represents the effect of collisions on the redistribution of f , and therefore is called the collision-operator, $\frac{df}{dt} = \Omega(f)$. Due to the fact that collisions also conserve certain quantities, such as mass and momentum, there are some constraints on the collision operator. Boltzmann first considered in this collision operator all outcomes of two-particle collisions, resulting in a highly complex term in velocity space [21]. The mathematically simpler operator used is the BGK collision operator,

$$\Omega(f) = -\frac{1}{\tau}(f - f^{eq}) \quad (2.54)$$

named after Bhatnagar, Gross and Krook, describing the relaxation towards the equilibrium distribution, with τ the speed of this equilibration called the relaxation time. Together, equation 2.53 and equation 2.54 result in the Boltzmann equation (BE) in the form

$$\Omega(f) = \frac{\partial f}{\partial t} + \xi_\alpha \frac{\partial f}{\partial x_\alpha} + \frac{F_\alpha}{\rho} \frac{\partial f}{\partial \xi_\alpha}. \quad (2.55)$$

From the BE in equation 2.55, one can derive the macroscopic conservation equations directly, by taking the moments of the equation [24]. By integrating equation 2.55 directly over velocity space, one obtains the continuity equation 2.40. By multiplying equation 2.55 with the particle velocity before integrating over velocity space, one obtains the Cauchy momentum equation, equation 2.43. Finally, by assuming $f \simeq f^{eq}$, one can obtain the Euler momentum equation 2.42. The Navier-Stokes equation seen in equation 2.44 can be derived by the Chapman-Enskog expansion, as described in [24].

The Chapman-Enskog expansion can also be used to show how the heat equation 2.45 can be retrieved from the later discussed lattice Boltzmann equation (LBE) for heat flow (seen in equation 3.10) and the precursor advection-diffusion equation 2.12 from the later discussed LBE for precursor transport (seen in equation 3.22).

Further, the Boltzmann equation can be used when assessing the behaviour of neutrons when one describes the behaviour of a "neutron gas" to obtain the neutron transport equation 2.2 [8]. Wang et al. [44] have shown how to retrieve the NDE from the LBE for neutronics through the Chapman-Enskog expansion.

3

Numerical Method

To solve the macroscopic equations in chapter 2 numerically the lattice Boltzmann method (LBM) is used in this research project. The aim is to integrate the previously described fields in a single model, using equivalent methods for efficient coupling. The lattice Boltzmann method has proven efficient in the modelling of thermal-hydraulics, and recently has proven to be an appropriate method for modelling other fields, such as radiation transport, neutron transport or diffusion as well [44, 46]. The lattice Boltzmann method is first described in section 3.1, after which the implementation of boundary conditions is discussed in section 3.2. Lattice conversion in the LBM is explained in section 3.3 together with the important stability criteria for proper simulations in section 3.4. Finally, the schemes for solving the coupled fields are described, followed by the description of the PCQSM in section 3.6 which is used to accelerate transient simulations.

3.1. Lattice Boltzmann Method

LBM makes use of the discretisation of velocity, space and time and the basic quantity in LBM is the discrete velocity distribution function $f_i(\mathbf{r}, t)$ [21]. Discretising the Boltzmann equation for these quantities (and integrating along certain characteristic) leads to the lattice Boltzmann equation (LBE)

$$f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i(\mathbf{r}, t) + \Omega_i(\mathbf{r}, t), \quad (3.1)$$

which describes that particles travelling in lattice direction i , moving with lattice speed e_i arrives at the next position $\mathbf{r} + \mathbf{e}_i \Delta t$ at time $t + \Delta t$, while being affected by a collision operator Ω_i representing particle collisions and re-distributions [21]. This collision operator can be modelled in several ways. Three common operators are the Bhatnagar-Gross-Krook (BGK) operator, the multiple-relaxation-time operator and the filter-matrix lattice Boltzmann method. In this research project, the BGK operator is used.

The discretised velocities form a discrete set $\{\mathbf{c}_i\}$ with weighting coefficients w_i . The velocity sets are commonly denoted by $DdQq$, with d the number of dimensions in space, and q the number of discretised velocities. The number of dimensions is often prescribed by the physical problem at hand and in this research project two-dimensional simulations are performed and d is always 2. The number of discretised velocities can be chosen depending on the purpose of the simulation [21]. Common choices for two-dimensional lattices are $D2Q9$, $D2Q5$ and $D2Q4$. The first is appropriate in situations where convection is important, while the second is appropriate when the diffusion effect is larger than convection [25]. The last is most appropriate when no convection takes place at all [44]. The $D2Q4$ and $D2Q9$ lattices are seen in figure 3.1.

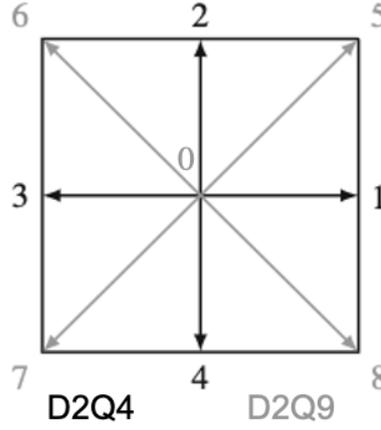


Figure 3.1: Two lattice velocity sets implemented in the LBM, $D2Q4$ in black and $D2Q9$ in black and grey [21].

The LBM is a common tool in fluid dynamic problems, but this research project aims to extend it to all fields. The LBM is also commonly used to solve advection-diffusion equations such as equation 2.7 and 2.45 [21] and will be used for the temperature and precursor concentration fields. The temperature and each precursor family will have its own distribution function. Further, Wang et al. [44] developed a LB scheme that solves the multi-group neutron diffusion equation, in which each neutron energy group has its own distribution function. This method is used and extended to include delayed neutron sources.

For simplicity, the LBM together with the used BGK operator will be first described using its initial application, namely for solving the Boltzmann equation for the Navier-Stokes equation. Afterwards, the LBM for modelling the other relevant fields will be described.

3.1.1. LBM for the Navier-Stokes equation

The collision operator used in this research project is the BGK operator described above, which discretised has the form

$$\Omega_i(f) = -\frac{1}{\tau_f}(f_i - f_i^{eq}) , \quad (3.2)$$

and represents the relaxation towards an equilibrium distribution of particles. The equilibrium distribution function for the momentum field is designed to mimic the Maxwell-Boltzmann equilibrium distribution and is given by

$$f_i^{eq}(\mathbf{r}, t) = w_i \rho \left(1 + \frac{\mathbf{u} \cdot \mathbf{c}_i}{c_s^2} + \frac{(\mathbf{u} \cdot \mathbf{c}_i)^2}{2c_s^4} - \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right) . \quad (3.3)$$

As stated in the previous chapter, the NSE can be retrieved from the BE, which can be retrieved from the LBE. To retrieve the NSE, one performs the Chapman-Enskog expansion. The description of the Chapman-Enskog expansion can be found in literature [21]. Through the Chapman-Enskog expansion, the relation between the kinetic viscosity and the dimensionless relaxation time is found to be [21]

$$\nu = \left(\tau_f - \frac{1}{2} \right) \Delta t c_s^2 . \quad (3.4)$$

In the presence of momentum sources, the discretised Lattice Boltzmann Equation (LBE) receives an additional source term, and the equation becomes

$$f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i(\mathbf{r}, t) - \frac{1}{\tau_f} [f_i(\mathbf{r}, t) - f_i^{eq}(\mathbf{r}, t)] + \Delta t q_i , \quad (3.5)$$

where q_i depends on the force term acting on the fluid. The distribution functions can be related to macroscopic variables by taking moments [21]. The macroscopic mass density and macroscopic momentum density can be obtained from the first and second moments,

$$\rho(\mathbf{r}, t) = \sum_i f_i(\mathbf{r}, t) \quad (3.6)$$

and

$$\rho \mathbf{u}(\mathbf{r}, t) = \sum_i \mathbf{c}_i f_i(\mathbf{r}, t) , \quad (3.7)$$

respectively. When modelling the momentum field, convection is of importance and the $D2Q9$ lattice is used.

Time step in LBM

One time step in LBM can be decomposed in two steps that are successively executed:

1. Collision:

$$f_i^*(\mathbf{r}, t) = f_i(\mathbf{r}, t) - \frac{1}{\tau_f} (f_i(\mathbf{r}, t) - f_i^{eq}(\mathbf{r}, t)) , \quad (3.8)$$

where f_i^* represents the intermediate distribution function during a time step, after collision.

2. Streaming:

$$f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i^*(\mathbf{r}, t) , \quad (3.9)$$

where the distribution obtained in the collision step is streamed towards neighbouring nodes along the discretised lattice directions. A graphical representation of this process is found in figure 3.2

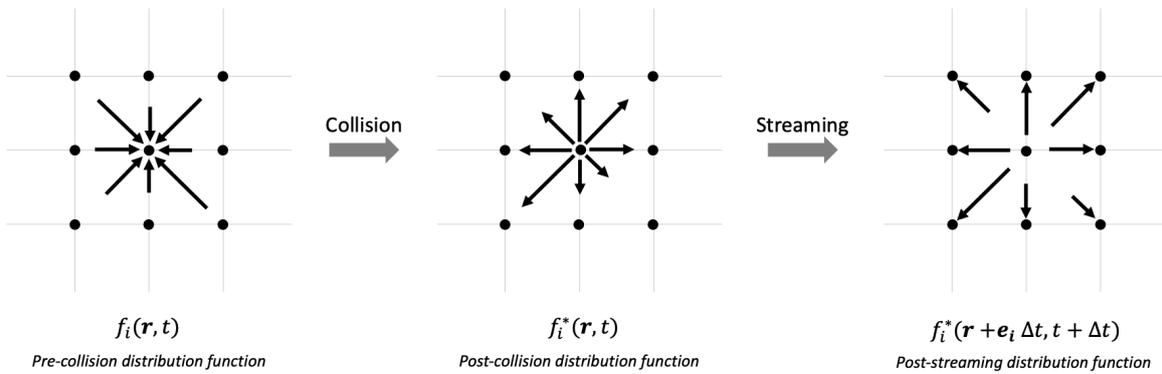


Figure 3.2: Schematic overview of the collision and streaming step for distribution function f_i .

In the LBM algorithm these time-steps can be executed until a maximum number of time steps has passed, or until the fields have converged. This LBM algorithm can be extended to the other fields.

3.1.2. LBM for the heat equation

Heat transfer is described by an advection-diffusion equation, as was shown in the previous chapter, thus the Lattice Boltzmann method for advection-diffusion can be used [15]. This results again in a similar form of equation 3.1 with a BGK operator, but now for the heat distribution function h_i and with a source term Q_i

$$h_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = h_i(\mathbf{r}, t) - \frac{1}{\tau_h} [h_i(\mathbf{r}, t) - h_i^{eq}(\mathbf{r}, t)] + \Delta t Q_i(\mathbf{r}, t) . \quad (3.10)$$

The equilibrium distribution function

$$h_i^{eq}(\mathbf{r}, t) = w_i T \left(1 + \frac{\mathbf{u} \cdot \mathbf{c}_i}{c_s^2} + \frac{(\mathbf{u} \cdot \mathbf{c}_i)^2}{2c_s^4} - \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right) , \quad (3.11)$$

is analogous to that of the particle distribution function. To retrieve the heat equation from equation 3.10, the Chapman-Enskog expansion can be applied and the relation between the dimensionless relaxation time and the thermal diffusivity is obtained as

$$\alpha = \left(\tau_h - \frac{1}{2} \right) c_s^2 \Delta t . \quad (3.12)$$

The source term is also obtained from the Chapman-Enskog expansion [21] for first order accuracy as

$$Q_i = \frac{w_i}{\rho C_p} \left[\gamma(T_{ext} - T) + E_f \sum_g \Sigma_f^g \phi_g \right]. \quad (3.13)$$

The macroscopic temperature T can again be found by the summation of the distribution functions over the discrete directions

$$T(\mathbf{r}, t) = \sum_i h_i(\mathbf{r}, t). \quad (3.14)$$

When modelling the temperature field in this research convection is of importance and the $D2Q9$ lattice is used.

3.1.3. LBM for the neutron diffusion equation

In the same way as for fluid particles, we can discretise the continuous neutron velocities into finite particle speeds and obtain local neutron distribution functions for neutrons in phase space [44]. The lattice Boltzmann equation for neutronics using the BGK operator is then given by

$$n_{i,g}(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = n_{i,g}(\mathbf{r}, t) - \frac{1}{\tau_{n,g}} [n_{i,g}(\mathbf{r}, t) - n_{i,g}^{eq}(\mathbf{r}, t)] + R_{i,g}(\mathbf{r}, t) \quad (3.15)$$

where $\tau_{n,g}$ is the relaxation time for neutronics (subscript n) for neutron energy group g , $n_{i,g}^{eq}(\mathbf{r}, t)$ is the equilibrium distribution function, and $R_{i,g}(\mathbf{r}, t)$ is the neutron source [14, 44]. Summation of the distribution function over the lattice directions results in the scalar neutron flux

$$\phi_g(\mathbf{r}, t) = \sum_i n_{i,g}(\mathbf{r}, t). \quad (3.16)$$

Since the total neutron distribution does not change in the relaxation process of LBM, the summation of the equilibrium distribution and the local distribution should obtain the same result, i.e.

$$\phi_g(\mathbf{r}, t) = \sum_i n_{i,g}(\mathbf{r}, t) = \sum_i n_{i,g}^{eq}(\mathbf{r}, t). \quad (3.17)$$

Since the neutron diffusion approximation does not entail any convection, the equilibrium distribution is uniform in all directions and the relation of the scalar neutron flux and the neutron equilibrium distribution is given by

$$n_{i,g}^{eq}(\mathbf{r}, t) = w_i \phi_g(\mathbf{r}, t). \quad (3.18)$$

To ensure that the original multi-group NDE can be retrieved, the Chapman-Enskog expansion must be applied, and when doing so one finds the expressions for the neutron relaxation time $\tau_{n,g}$ and neutron source function $R_{i,g}(\mathbf{r}, t)$ 3.15. Starting from the LBM model for the neutronics described in equation 3.15, one can derive the NDE and find relations [44]

$$D_g v_g = (\tau_{n,g} - \frac{1}{2}) c_s^2 \Delta t, \quad (3.19)$$

and

$$R_{i,g} = \Delta t R_{i,g}^{(1)} + \Delta t^2 R_{i,g}^{(2)} + \Delta t^3 R_{i,g}^{(3)} + \dots, \quad (3.20)$$

with, using only the first order truncation [44],

$$R_{i,g} \simeq \Delta t R_{i,g}^{(1)} = \Delta t w_i v_g \left[-\Sigma_{t,g} \phi_g + \frac{\chi_g^p (1 - \beta_{tot})}{k_{eff}} \sum_{g'} (\nu \Sigma_f)_{g'} \phi_{g'} + \chi_g^d \sum_d \lambda_d C_d + \sum_{g'} \Sigma_{s,g'-g} \phi_{g'} \right]. \quad (3.21)$$

This full description of the Chapman-Enskog expansion applied on the LBE for neutronics, equation 3.15, is elaborated in [44]. When using the lattice Boltzmann method for the inner iteration in the power method to solve the eigenvalue equation, the solving scheme is an iterative process. That means that transient terms have no meaning. Therefore, the neutron velocity can be eliminated from equations 3.19 and 3.21. This was also seen in the forms of the steady-state NDE in equations 2.9 and 2.11, where

the neutron velocities v_g were eliminated. This is convenient in simulations, since neutron velocities are generally high, which leads to difficulties when trying to achieve stable simulation conditions, as will be explained in section 3.4.2.

As seen in absence of a convection term in the NDE, when modelling neutron transport convection does not play a role and the $D2Q4$ lattice is used.

3.1.4. LBM for neutron precursor advection-diffusion equation

Since the precursor behaviour is prescribed macroscopically by an advection-diffusion equation, analogous to heat transfer, the form of the LBM in equation 3.10 is used and altered for the precursor distribution functions as

$$c_{d,i}(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = c_{d,i}(\mathbf{r}, t) - \frac{1}{\tau_c} [c_{i,d}(\mathbf{r}, t) - c_{i,d}^{eq}(\mathbf{r}, t)] + \Delta t Q_{d,i}(\mathbf{r}, t) . \quad (3.22)$$

The equilibrium distribution function is found as

$$c_{d,i}^{eq}(\mathbf{r}, t) = w_i C_d \left(1 + \frac{\mathbf{u} \cdot \mathbf{c}_i}{c_s^2} + \frac{(\mathbf{u} \cdot \mathbf{c}_i)^2}{2c_s^4} - \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right) . \quad (3.23)$$

By again performing the Chapman-Enskog expansion to retrieve the original governing equation, the relaxation time is found to be related to the diffusion constant through

$$D_p = \left(\tau_c - \frac{1}{2} \right) c_s^2 \Delta t . \quad (3.24)$$

The source term is also obtained from the Chapman-Enskog expansion and is given, for first order accuracy, as

$$Q_{d,i} = w_i \left[\frac{\beta_d}{k_{eff}} \sum_g (\nu \Sigma_f \phi)_g - \lambda_d C_d \right] . \quad (3.25)$$

The macroscopic precursor concentration C_d can be found by the summation of the distribution function over the discrete directions

$$C_d(\mathbf{r}, t) = \sum_i c_i(\mathbf{r}, t) \quad (3.26)$$

When modelling the precursor concentration field, convection is of importance and the $D2Q9$ lattice is used.

3.2. Boundary conditions

In this thesis, walls are modeled using both wet-node boundary schemes and link-wise boundary schemes. The difference is that for wet-node boundary schemes boundary nodes are on the physical boundary, while for link-wise boundary schemes the physical boundary is located half-way between solid and boundary lattice nodes. Solid nodes x_s are nodes located outside of the domain, boundary nodes are located (partly) inside the domain. This is graphically displayed in figure 3.3.

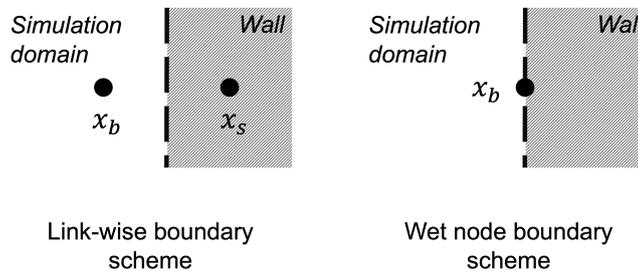


Figure 3.3: Schematic overview of link-wise and wet node boundary schemes.

For simplicity all boundary conditions are expressed using the momentum distribution functions, unless specified otherwise.

3.2.1. Dirichlet boundary conditions

For the momentum distribution function the Dirichlet boundary condition with a prescribed wall velocity is given as [21]

$$f_j(x_b, t + \Delta t) = f_i^*(x_b, t) - 2w_i\rho_w \frac{c_i \cdot \mathbf{u}_w}{c_s^2} = f_i^*(x_b, t) + 2w_j\rho_w \frac{c_j \cdot \mathbf{u}_w}{c_s^2} , \quad (3.27)$$

where $c_j = -c_i$, which is interpreted as that both the tangential and the normal velocity vectors are inverted at the walls. Note that the starred notation, just as in section 3.1 refers to the intermediate distribution function (after collision, but before streaming). In the $D2Q9$ scheme this is a reflection of 180 degrees. For a resting wall, using $\mathbf{u}_w = 0$, this reduces to

$$f_j(x_b, t + \Delta t) = f_i^*(x_b, t) . \quad (3.28)$$

From a macroscopic point of view these equations lead to no-slip conditions for a wall located midway on lattice links. Equation 3.27 considers a boundary that recovers a velocity condition, which is useful when determining the momentum distribution function. For the advection-diffusion equations Dirichlet boundary conditions can be modeled using an anti-bounce-back condition with zero-velocity at the walls as

$$f_j(x_b, t + \Delta t) = -f_i^*(x_b, t) + 2w_i Q_w , \quad (3.29)$$

with Q_w the macroscopic quantity imposed at the wall.

3.2.2. Von Neumann boundary conditions

The von Neumann boundary condition can be implemented as a symmetry boundary condition, expressed as

$$f_j(x_b + c_{j,t}\Delta t, t + \Delta t) = f_i^*(x_b, t) , \quad (3.30)$$

where $c_{j,t} = c_{i,t}$ and $c_{j,n} = -c_{i,n}$, which is interpreted as that the tangential velocities remain constant across streaming, but the normal velocities are reversed. This is called specular reflection. The scheme results from a macroscopic point of view in zero normal velocity for the momentum equation, or zero normal flux for the advection-diffusion equations, but imposes no condition for tangential velocity or flux for the momentum and advection-diffusion equations respectively [21]. This is also referred to as free-slip.

3.2.3. Boundary condition for neutronics

For the neutronics a wet-node approach is used. In effect the simulation domains for the neutron field and the thermal-hydraulics fields are not right on top of each other anymore, as can be seen in the figure 3.4.

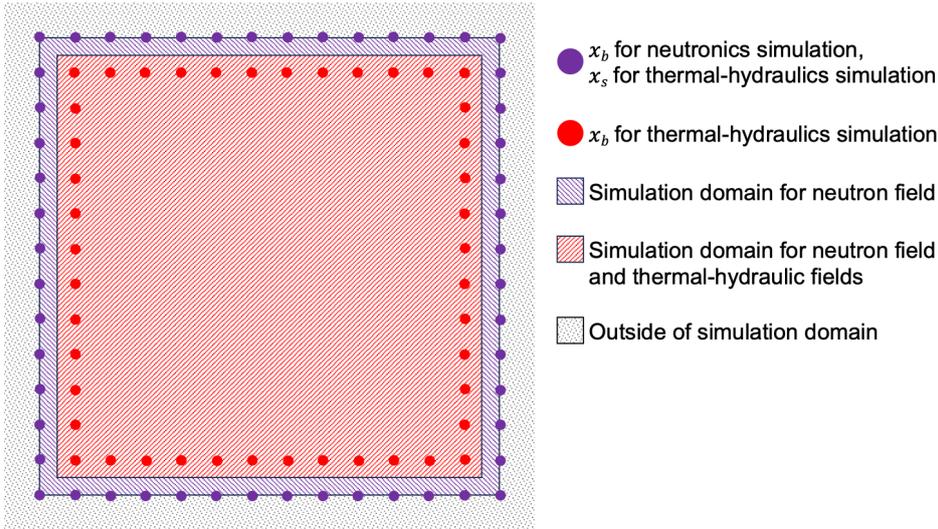


Figure 3.4: Schematic overview of the simulation domains for the neutron fields and thermal-hydraulics fields.

This is justified by the fact that in chapter 2.2 it was described that in diffusion theory the domain where the scalar neutron flux is assessed should be extended by taking extrapolated boundaries as in equation 2.5. This boundary extrapolation would be a complex task as the extrapolated boundary is a function of the diffusion constant, which is different for each energy group. That implies that for each neutron energy group a different lattice grid would have to be implemented with boundaries at different points in space. For simplicity, the extrapolation is therefore limited to the solid nodes in the thermal-hydraulics simulations for all groups represented as x_b in figure 3.5. In the figure, the actual extrapolation that is prescribed in the diffusion approximation as in equation 2.5 is displayed for 2 neutron energy groups, with diffusion constants for which $D_2 > D_1$ holds.

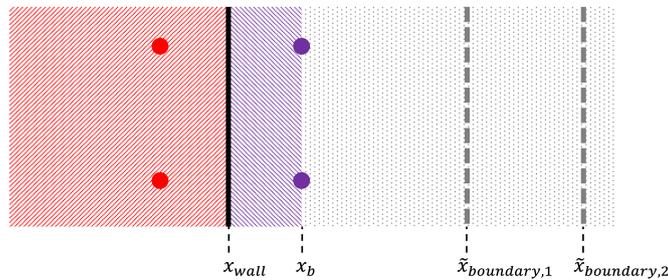


Figure 3.5: Schematic overview of the extrapolated boundary coordinates, $\tilde{x}_{boundary,1}$ and $\tilde{x}_{boundary,2}$ as prescribed in the diffusion approximation, and the actually implemented extrapolation in the numerical model, x_b .

Two boundary conditions are used in literature. Both these boundary conditions will be modelled and assessed on their performance. The first is the non-equilibrium extrapolation scheme (NEES) is implemented [44, 45, 46]. The NEES is expressed as

$$n_{i,g}(x_b, t) = n_{i,g}^{eq}(x_b, t) + [n_{i,g}(x_{adj}, t) - n_{i,g}^{eq}(x_{adj}, t)] \quad (3.31)$$

The other boundary condition used is the equilibrium boundary condition [27]. This is one of the more simple boundary conditions found in the theory of LBM. The EB condition is represented as

$$n_{i,g}(x_b, t) = n_{i,g}^{eq}(x_b, t) \quad (3.32)$$

If the equilibrium distribution function at the walls is zero (such as for vacuum boundary conditions, $\phi^{eq}(x_b, t) = 0$), this boundary condition leads to a zero neutron current density J at the boundary node.

At the boundary, while the scalar flux is zero, there can still be a non-zero gradient of the flux, leading to a non-zero neutron current. This is thus an approximation, since apart from assuming zero scalar neutron flux at the boundary also zero current density at the boundary is assumed.

3.3. Lattice conversion

Lattice Boltzmann simulations are generally performed in dimensionless numbers, called lattice units. This means that parameters need to be scaled from physical units to lattice units and after the simulation back to lattice units. This is done using a conversion factor [21], which is the ratio between physical units and lattice units of a quantity Q as

$$C_Q = \frac{Q_P}{Q^*} . \quad (3.33)$$

The lattice quantities are represented with a star superscript, whereas physical quantities obtain subscript P . Conversion factors of quantities can be combined to obtain conversion factors of other quantities. For instance, when scaling velocity (units m/s) one can use the conversion factors for distance and time and write

$$C_u = \frac{C_x}{C_t} . \quad (3.34)$$

Because the physical units for ν , α , D_p and $v_g D_g$ are equal, they share the same conversion factor

$$C_\nu = C_\alpha = C_{D_p} = C_{v_g D_g} . \quad (3.35)$$

The most important quantities to scale are mass, distance, time and temperature, since from these one can scale all other, in this research project, relevant quantities.

3.4. Stability and accuracy constraints

The lattice Boltzmann method works properly for most cases. However, there are some stability constraints for the lattice units. Firstly, flow must be incompressible. Compressibility of a flow can be quantified using the dimensionless Mach number Ma , which must be smaller than 0.1 in the incompressibility limit and is defined as the ratio of velocity of particles in a medium u and the speed of sound in the medium c_s [21] following

$$Ma = \frac{u^*}{c_s} < 0.1 . \quad (3.36)$$

There are situations where achieving this small Ma leads to high computational costs, from the necessity of increasing the number of grid-points or time-steps. It can then be tested whether higher Ma is already accurate, since the relation between the simulation error and Ma is not straightforward. It should be noted that Ma higher than 0.7 should always be avoided according to literature [21]. Next, for a stable simulation when using the BGK operator non-negativity of the equilibrium distribution functions need to be ensured. This generally holds for $\tau > 0.5$. Values near 0.5 might already result in instability, thus we restrict ourselves to the following condition

$$\tau^* > 0.505 . \quad (3.37)$$

The constraints mentioned above result in restrictions when choosing lattice parameters.

Any physical situation prescribes certain dimensionless parameters, that are consistent throughout unit conversions to obtain accurate simulations. Other than the previously described Mach numbers, in this research project the relevant dimensionless parameters are the Reynolds number Re , Schmidt number Sc and the Prandtl number Pr . The Reynolds number describes the type of flow considered, e.g. laminar or turbulent flow, defined as

$$Re = \frac{u_{max}^* \cdot x^*}{\nu^*} . \quad (3.38)$$

The Schmidt number is a scale for convective and diffusional concentration transport, defined as

$$Sc = \frac{\nu^*}{D_p^*} . \quad (3.39)$$

The Prandtl number is a scale for convectonal and diffusional heat transport, defined as

$$Pr = \frac{\nu^*}{\alpha^*} . \quad (3.40)$$

Combining accuracy and stability constraints results in criteria that must hold in all simulations. Using the first constraint in equation 3.36 and the physically prescribed relation for the type of flow in equation 3.38 results in incompressibility criterion

$$\frac{Re \cdot \nu^*}{x^*} \leq Ma \cdot c_s , \quad (3.41)$$

where ν^* and x^* can be chose freely. The combination of these two must be chosen such that the incompressibility criterion holds.

For the second constraint in equation 3.37, the relaxation times of the different fields, equation 3.4, 3.12, 3.19, 3.24, are assessed. As seen in equation 3.35 conversion parameters are shared amongst parameters determining relaxation times, and the choice of one influences all. Combining this with the prescribed dimensionless numbers from equation 3.39 and 3.40 we obtain for the non-negativity constraint

$$\nu^* \geq 0.005 \Delta t c_s^2 , \quad (3.42)$$

$$\frac{\nu^*}{Pr} \geq 0.005 \Delta t c_s^2 , \quad (3.43)$$

$$\frac{v_g D_g \cdot \nu^*}{\nu} \geq 0.005 \Delta t c_s^2 , \quad (3.44)$$

$$\frac{\nu^*}{Sc} \geq 0.005 \Delta t c_s^2 . \quad (3.45)$$

Note that parameters Pr , Sc , $v_g D_G$ and ν are prescribed by the physical problem and only the choice of ν^* is free. In summary, choice of lattice parameters ν^* and x^* should be chosen carefully for stable simulations.

3.4.1. Dealing with high Prandtl and Schmidt numbers

Typically the Pr number for molten salts is between 1 and 35, depending on temperature and salt composition [47]. These orders of magnitude for Pr and Sc are easily handled by LBM, as they can be implemented easily within stability criteria in equations 3.43 and 3.45. However, high Pr and Sc numbers lead to the necessity to choose a high ν^* to ensure that the left-hand sides remain bigger or equal than the right-hand sides. Difficulty occurs when combining this with 3.41, as increasing ν^* leads to the necessity of increasing x^* . This essentially means increasing the grid refinement, leading to high computational costs.

A way to avoid this issue is to think about what the high Pr and Sc numbers physically mean. A high Prandtl number means that thermal diffusivity has a much lower effect than momentum diffusivity and convection has bigger effect on the heat distribution than conduction. Similarly, a high Schmidt number means that mass diffusivity has a much lower effect than momentum diffusivity and convection has bigger effect on the mass distribution than diffusion. To understand what high Pr and Sc numbers mean for the model, dimensionless analysis is performed on the relevant macroscopic equations. The non-dimensional form of the steady-state heat equation is found in the form

$$Pr Re [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{T}] = -\tilde{\nabla}^2 \tilde{T} + K (Gr Pr)^{1/4} (1 - \tilde{T}) , \quad (3.46)$$

with K a constant that relates to the Nusselt number, a measure for the ratio between convective and conductive heat transfer. The derivation can be found in appendix C.1. The effect of a high Pr number can be seen from this equation. When the divergence and Laplacian of \tilde{T} are low, increasing the Pr number results in dominance of the convection term (left-hand side) and heat sink term (first of the right-hand side), while the diffusion term (second term of the right-hand side) becomes relatively insignificant. At some high Pr number, diffusion becomes so insignificant that increasing the Pr number has little effect on the temperature distribution. Thus, in the simulation, it can be tested whether the implementation of a lower Pr , that suffices to the criterion in equation 3.43 more easily (that is, without too refined grid), already leads to accurate results. This reasoning breaks down if the divergence and

Laplacian of \tilde{T} are high and the terms containing the Pr number do not necessarily dominate. A similar line of reasoning can be followed for the Sc number, where the non-dimensional form of the steady-state mass concentration equation is found in the form

$$ScRe[\tilde{\mathbf{u}} \cdot \tilde{\nabla}\tilde{C}] = \tilde{\nabla}^2\tilde{C} - Da\tilde{C} \quad , \quad (3.47)$$

with Da the Damköhler number, that relates to the decay rate of the precursors. The derivation can be found in appendix C.2. Again, the diffusion term becomes less and less significant for high Sc numbers, when the divergence and Laplacian of \tilde{C} are low. It should be tested whether the implementation of a lower Sc , that suffices to the criterion in equation 3.45 more easily (that is, without too refined grid), already leads to accurate results.

3.4.2. Dealing with high neutron velocities

In neutronics simulations utilizing the Lattice Boltzmann Method (LBM), a notable challenge arises from the fact that high neutron velocities result in high relaxation times, as seen in equation 3.19. Neutron velocities in MSFRs are typically within the interval $10^5 m/s - 10^7 m/s$. High relaxation times are undesirable, as the relaxation time directly scales with the simulation error [21]. While this issue can be mitigated in steady-state scenarios by omitting neutron velocities from the relaxation time equation 3.19, as mentioned in section 3.1.3, this strategy is not applicable in transient situations. Therefore, in transient situations of using the LBM for neutronics simulations, one has to scale down the velocity in lattice units using a very high number of time-steps, which leads to increased numerical costs. A way to avoid this is by remembering from section 2.2.3 that one can split the neutron fluxes in the slowly changing shape function and fast changing amplitude function. The shape function can be solved using the steady-state LBM until convergence, where once again the neutron velocities can be omitted. When one subsequently solves for the fast changing amplitude equation, one can obtain the neutron scalar group fluxes, without enormously increasing numerical times. This is a quasi-static method, as one assumes the shape function to be static at the small time-scales of the amplitude function. The combination of the LBM with the quasi-static method will be explained further in section 3.6.

3.4.3. Ensuring stability while choosing lattice units

For the mass and temperature conversion factors C_m and C_T , one is free to choose what suits best for the simulation or is easy to handle, since these quantities do not have to comply with criteria for stability. For the distance and time conversion factors, C_x and C_t , on the other hand, one has to be more careful since quantities related to these, such as ν^* , α^* , D^* , $v_g^* D_g^*$ and u_{max}^* , must comply with stability criteria. A way to ensure stability is choosing units as follows:

1. Choose ν^* such that equations 3.42, 3.43, 3.44 and 3.45 hold.
2. Using ν^* from step 1 choose the number of lattice nodes, x^* , in a physical length scale, x , such that equation 3.41 holds. Now one can find the maximum lattice velocity u_{max}^* using equation 3.38 and using the maximum physical velocity u_{max} can determine C_u using equation 3.33.
3. Determine C_x , using equation 3.33, inserting the physical length scale x and the number of lattice nodes in that physical length scale x^* . Use equation 3.34 to determine the conversion factor for time, C_t , by

$$C_t = \frac{C_x}{C_u} \quad \text{or} \quad C_t = \frac{C_x^2}{C_\nu} \quad , \quad (3.48)$$

ensuring that these two expressions for C_t are consistent.

4. Choose C_m and C_T adequate for simulation, without restrictions.
5. Scale other quantities using conversion factors for mass, distance, time and temperature using variations of equation equation 3.34 based on the physical units of the to be scaled quantity.

3.5. Scheme for solving coupled eigenvalue problem

For the coupled system, the thermal-hydraulics and neutronics fields are solved simultaneously and the system of lattice Boltzmann equations becomes

$$\mathbf{F}_i^{*(l)} = \mathbf{F}_i^{(l)} + \Omega_i^{(l)} + \mathbf{Q}_i^{(l)} + \frac{1}{k_{eff}^{(l-1)}} \mathbf{Q}_{f,i}^{(l-1)} \quad , \quad (3.49)$$

in which the vectors are given as

$$\mathbf{F}^*_i = \begin{bmatrix} f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) \\ h_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) \\ n_{g,i}(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) \\ c_{d,i}(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) \end{bmatrix}, \quad (3.50)$$

$$\mathbf{F}_i = \begin{bmatrix} f_i(\mathbf{r}, t) \\ h_i(\mathbf{r}, t) \\ n_{g,i}(\mathbf{r}, t) \\ c_{d,i}(\mathbf{r}, t) \end{bmatrix}, \quad (3.51)$$

$$\mathbf{\Omega}_i = \begin{bmatrix} -\frac{\Delta t}{\tau_f} (f_i - f_i^{eq}) \\ -\frac{\Delta t}{\tau_h} (h_i - h_i^{eq}) \\ -\frac{\Delta t}{\tau_{n,g}} (n_{i,g} - n_{i,g}^{eq}) \\ -\frac{\Delta t}{\tau_c} (c_{d,i} - c_{d,i}^{eq}) \end{bmatrix}, \quad (3.52)$$

$$\mathbf{Q}_i = \Delta t w_i \begin{bmatrix} -\beta_{th} \rho_0 (T - T_{ref}) (\mathbf{g} \cdot \mathbf{c}_i) \\ \frac{1}{\rho C_p} [\gamma (T_{ext} - T) + E_f \sum_g \Sigma_{f,g} \phi_g] \\ v_g [-\Sigma_{t,g} \phi_g + \chi_g^d \sum_{d=1}^D \lambda_d C_d + \sum_{g'}^G \Sigma_{s,g' \rightarrow g} \phi_{g'}] \\ -\lambda_d C_d \end{bmatrix} \quad (3.53)$$

$$\mathbf{Q}_{f,i} = \Delta t w_i \begin{bmatrix} 0 \\ 0 \\ v_g [\chi_g^p (1 - \beta^{tot})] \\ \beta_d \end{bmatrix} \cdot S_f, \quad (3.54)$$

and

$$S_f = \sum_{g=1}^6 (\nu \Sigma_f \phi)_g. \quad (3.55)$$

The superscript (l) represents the outer iteration number, as described in 2.2.2. The inner iteration entails the subsequent execution of the collision and streaming steps in the lattice Boltzmann method for all the different fields simultaneously, until all distribution functions and thus macroscopic properties converge. Distribution functions and macroscopic quantities are updated throughout the inner iterations, except for the fission terms. In the inner iterations, the fission source term $\mathbf{Q}_{f,i}$ and effective multiplication factor k_{eff} , are not altered. As will become relevant later, for different coupling situations, different fields have to be solved actively. When a field is not solved, the entire row in equation 3.49 for that field can be omitted in the inner iteration.

In the inner iterations the lattice Boltzmann method is used repeatedly to determine the distribution functions of the several fields, until convergence. Once the distribution functions of all fields converge, the neutron fluxes and precursor concentrations are determined and used to update the fission source term $\mathbf{Q}_{f,i}$ and effective multiplication factor k_{eff} using equations 2.19 and 2.20. For each different stage of the steady-state coupled simulations (referred to as phase 1) the iteration schemes are displayed in the corresponding sections in chapter 5.

3.6. Predictor Corrector Quasi-Static Method

The transient behaviour is assessed with the initial system being critical. That means that the critical system must first be determined before continuing to the transient system. This is done by applying the LBM until convergence using the power method, determining the steady-state eigenvectors and eigenvalue k_{eff} . The determined eigenvalue for the steady-state system is used in the transient calculation such that the fission cross-sections are scaled for criticality if no perturbation is applied.

As described in section 2.2.3, two types of functions are to be solved: shape functions and amplitude functions. The amplitude functions are solved using the Euler method, described later. The shape functions are solved using the LBM. Since the LBM does not solve for the neutron fluxes and precursor concentrations directly but solves for distribution functions, some adjustments are made to the regular

PCQSM scheme. From the factorisation in equation 2.21 a similar factorisation for the amplitude follows as

$$n_{i,g}(\mathbf{r}, t) = \tilde{n}_{i,g}(\mathbf{r}, t)n(t) \quad , \quad (3.56)$$

where $\tilde{n}_{i,g}(\mathbf{r}, t)$ is the neutron distribution shape function.

The amplitude function is solved on the shortest time scale Δt_S , and the shape function on a longer time scale Δt_B . The PK parameters ρ , Λ and β are determined on an intermediate time-scale Δt_M . Within the intermediate time scale, the PK parameters are considered constant. This is represented in figure 3.6.

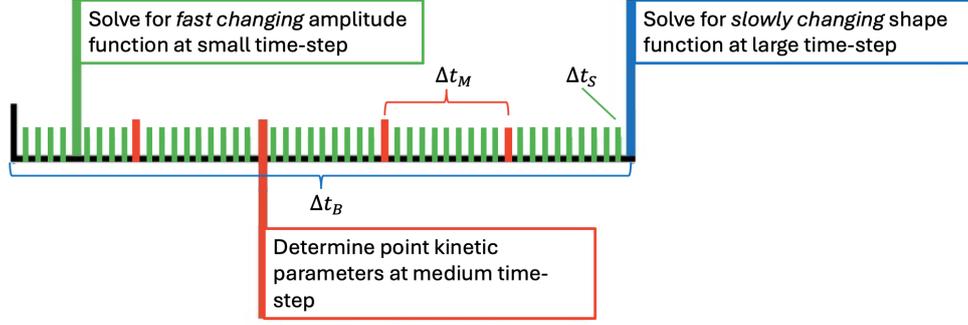


Figure 3.6: Schematic overview of the different time-scales used in the PCQSM.

The steady-state lattice Boltzmann method is used to determine predicted distribution functions $n_{i,g}^p(\mathbf{r}, t + \Delta t_B)$ and $c_{i,d}^p(\mathbf{r}, t + \Delta t_B)$ first, and later sums over these distribution functions to obtain the predicted neutron flux $\phi^p(\mathbf{r}, t + \Delta t_B)$ and predicted precursor concentration $C_d^p(\mathbf{r}, t + \Delta t_B)$.

From the predicted neutron distribution functions $n_{i,g}^p(\mathbf{r}, t + \Delta t_B)$ the neutron distribution shape functions $\tilde{n}_{i,g}(\mathbf{r}, t + \Delta t_B)$ are determined using

$$\tilde{n}_{i,g}(\mathbf{r}, t) = \frac{n_{i,g}^p(\mathbf{r}, t)}{\sum_{g=1}^G \int_V \frac{1}{v_g} \Phi_g^*(\mathbf{r}) \phi_g^p(\mathbf{r}, t) dV} \quad . \quad (3.57)$$

From the predicted neutron fluxes the neutron flux phase functions $\tilde{\phi}_g(\mathbf{r}, t + \Delta t_B)$ can be determined, using

$$\tilde{\phi}_g(\mathbf{r}, t) = \frac{\phi_g^p(\mathbf{r}, t)}{\sum_{g=1}^G \int_V \frac{1}{v_g} \Phi_g^*(\mathbf{r}) \phi_g^p(\mathbf{r}, t) dV} \quad . \quad (3.58)$$

The flux phase function is linearly interpolated in interval $[t, t + \Delta t_B]$ in time-steps of Δt_M , such that at each intermediate time step Δt_M a phase function is known.

At each medium time step the PK parameters are determined, using the interpolated flux phase function and equations 2.35, 2.36 and 2.37.

Within the medium time interval $[t, t + \Delta t_M]$ the PK parameters are considered constant, and the point kinetics equations 2.33 reduce to ordinary differential equations (ODEs) as

$$\frac{dn(t)}{dt} = \frac{\rho - \beta}{\Lambda} n(t) + \sum_d \lambda_d m_d(t) \equiv N(n(t), m_d(t)) \quad , \quad (3.59)$$

$$\frac{dm_d(t)}{dt} = \frac{\beta_d}{\Lambda} n(t) + \lambda_d m_d(t) \equiv M_d(n(t), m_d(t)) \quad . \quad (3.60)$$

where $N(n(t), m_d(t))$ and $M_d(n(t), m_d(t))$ are functions that are used in solving the equations.

In the model these ODEs are numerically solved using the Euler method [18]. This method allows one to determine an approximation of $n(t + \Delta t_S)$ and $m_d(t + \Delta t_S)$, provided that $n(t)$ and $m_d(t)$ are known, as

$$n(t + \Delta t_S) = n(t) + \Delta t_S \cdot N(n(t), m_d(t)) \quad (3.61)$$

$$m_d(t + \Delta t_S) = m_d(t) + \Delta t_S \cdot M_d(n(t), m_d(t)) \quad (3.62)$$

At $t = 0$, the amplitude functions are given as [9]

$$n(0) = 1 \quad \& \quad m_d(0) = \frac{\beta_d}{\Lambda \lambda_d} . \quad (3.63)$$

When the big time-step is reached, the predicted neutron fluxes $\phi^p(\mathbf{r}, t + \Delta t_B)$ are corrected by using the amplitude function, $n(t + \Delta t_B)$, and the phase functions, $\tilde{\phi}_g(\mathbf{r}, t + \Delta t_B)$ as formulated in equation 2.21. This is repeated until the total time span of the simulation is reached. The scheme is represented in figures 3.7 and 3.8.

The PK parameters are solved using equations 2.34, 2.35, 2.36 and 2.37 where volume integrals are numerically implemented as summing over all lattice nodes. In equation 2.34 taking the gradient and divergence is numerically implemented using the central difference method. The central difference method estimates the gradient and divergence by looking at values of the function at points on either side of the point of interest. Numerically the gradient is implemented as

$$\nabla f(x, y) = \left[\begin{array}{c} \frac{f(x+\Delta x, y) - f(x-\Delta x, y)}{2\Delta x} \\ \frac{f(x, y+\Delta y) - f(x, y-\Delta y)}{2\Delta y} \end{array} \right] , \quad (3.64)$$

where Δx and Δy are the spacings between lattice nodes and f is the function of interest, for instance ϕ . The divergence is implemented as

$$\nabla \cdot \mathbf{F}(x, y) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = \frac{u(x + \Delta x, y) - u(x - \Delta x, y)}{2\Delta x} + \frac{v(x, y + \Delta y) - v(x, y - \Delta y)}{2\Delta y}, \quad (3.65)$$

where Δx and Δy are again the spacings between lattice nodes and \mathbf{F} is the vector field of interest, for instance $\nabla \phi$. It should be noted that in this research the gradient and divergence near walls led to numerical errors and these are set to zero. In further research it is suggested to improve the gradient and divergence calculation near walls.

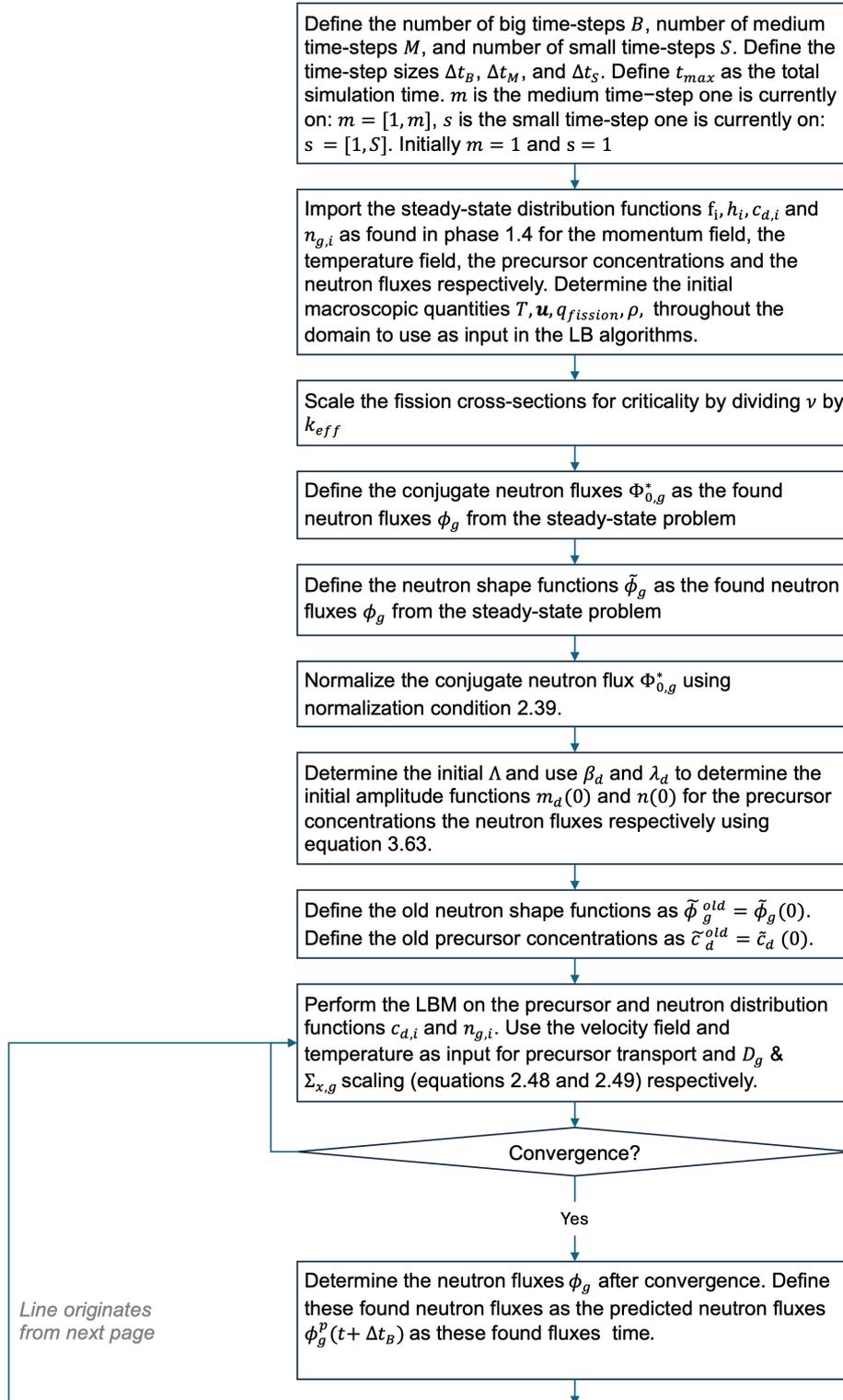


Figure 3.7: Solving scheme for transient simulation using PCQSM and LBM. Part (1/2) Scheme continues on the next page.

4

Validation of single-physics models

The aim of this research is to build a coupled multi-physics tool for several physical fields. To ensure that sources of errors in the tool are effectively traced, the coupling between the different physical fields is gradually increased. The benchmark study by M. Tiberga et al.[35] is used for the validation of the model. The neutronics model is assessed against additional, simpler benchmark studies and analytical solutions before extending the model to the complexity of the benchmark by M. Tiberga et al. [35]. In section 4.1 the physical conditions of Tiberga benchmark[35] is described. The validation of the uncoupled LBM for the velocity field is described in section 4.2 by comparison to the Tiberga benchmark. Subsequently the validation of the uncoupled LBM for the neutronics field in static fuel is described in section 4.3 in which one analytical and two numerical benchmark studies are used for comparison, including the Tiberga benchmark. In section 4.4 the validation of the uncoupled LBM for the temperature field is described and in section 4.5 the validation of the uncoupled LBM for the neutronics field in non-static fuel is described, both in comparison with the Tiberga benchmark. Finally, some intermediate conclusions are drawn in section 4.6 that are used in the simulations for the coupled fields in the next chapter.

To compare the obtained results against an analytical solution, the accuracy is represented by the Relative Root Means Squared Error (RRMSE) as

$$RRMSE = \sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} \left(\frac{Q(\mathbf{r}_i) - Q_{analytic}(\mathbf{r}_i)}{Q_{analytic}(\mathbf{r}_i)} \right)^2}, \quad (4.1)$$

where $Q(\mathbf{r}_i)$ is the found value following from the simulation using LBM at sampling point \mathbf{r}_i and $Q_{analytic}(\mathbf{r}_i)$ is the analytical solution at that sampling point. N_p represents the number of sampling points. When no analytical solutions can be used, the accuracy the results is assessed by computing the discrepancy from other codes as

$$\epsilon_c = \sqrt{\frac{\sum_{i=1}^{N_p} (Q_c(\mathbf{r}_i) - Q_{avg}(\mathbf{r}_i))^2}{\sum_{i=1}^{N_p} Q_{avg}^2(\mathbf{r}_i)}}, \quad (4.2)$$

where the subscript c is for the assessed code, $Q_c(\mathbf{r}_i)$ is the value of the computed quantity at sampling point \mathbf{r}_i and Q_{avg} is the average value of that quantity computed in the different codes. N_p is the number of sampling points [35]. An interpretation of the obtained physical fields and the effects of coupling is described in section 5.4 and is not included in this chapter.

4.1. Description of the Tiberga benchmark

The Tiberga benchmark is developed as a "coupled numerical benchmark specifically designed to assess the physics-coupling capabilities" of numerical tools for simulating the fields in a Molten Salt Fast Reactor [35]. The benchmark study compares codes developed during the SAMOFAR project at four different institutions: Delft University of Technology, Paul Scherrer Institute, Politecnico di Milano and

LPSC/CNRS-Grenoble. The interesting feature of the Tiberga benchmark is that it considers a step-by-step approach in the sense that coupling between fields is gradually increased. Initially, the neutronics, momentum and temperature fields are modelled separately, and with each phase in the Tiberga benchmark more fields are coupled. This is predominantly useful to identify mistakes and assess the capabilities of handling the coupling of the different fields. The gradual coupling is expressed by subdividing the simulations into phases. Phase 0 entails simulations of the uncoupled fields, with phases 0.1, 0.2 and 0.3 simulations for the velocity, neutronics and temperature fields respectively. Any inter-field influence is passively implemented, i.e. in phase 0.3 a fixed velocity and heat source influence the temperature field. Phase 1 entails simulations of gradually coupled fields. In phase 1.1 the velocity and precursor concentration fields are coupled, but still passively. For this reason, phase 1.1 in this research project is considered uncoupled. In phases 1.2, 1.3 and 1.4 active coupling is increased gradually, meaning that the fields are not fixed but impact each other throughout simulations. The Tiberga benchmarks provides results of codes developed by the four aforementioned institutions against which the results can be assessed.

4.1.1. The molten salt system

The multi-physics benchmark describes a system using $LiF - BeF_2 - UF_4$ as a fuel salt, with fluid properties constant with temperature and uniform in space. Fuel salt composition is found in table 4.1. The flow is considered incompressible and laminar, buoyancy forces are modelled using the Boussinesq approximation and the reactor is considered homogeneous. The reactor core is a two-dimensional lid-driven cavity, with sides of 2 meters. The boundaries of the core are bare, meaning zero neutron flux, adiabatic, meaning zero normal heat flux, and no-slip, with zero velocity everywhere except at the top lid. For the precursors, von Neumann boundary conditions are implemented, resulting in a zero normal precursor flux at the boundaries. Salt cooling in the core is simulated as a volumetric heat sink, as

$$q(\mathbf{r}) = \gamma(T_{ext} - T(\mathbf{r})) . \quad (4.3)$$

The neutrons are grouped into six energy groups, and the precursors are grouped into eight families. No upscattering is assumed, meaning that neutrons can only scatter from higher energy groups to lower energy groups. Cross sections and neutron diffusion coefficients for diffusion codes are affected only by the salt expansion feedback and scale with temperature. Excluded are turbulence, decay heat and Doppler effects on microscopic cross sections. To still include a temperature dependency of the cross-sections and diffusion coefficients they are scaled with fuel density as described in equations 2.48 and 2.49. A graphical representation of the reactor core is found in figure 4.1. Quantities are observed along the displayed center-lines AA' and BB' .

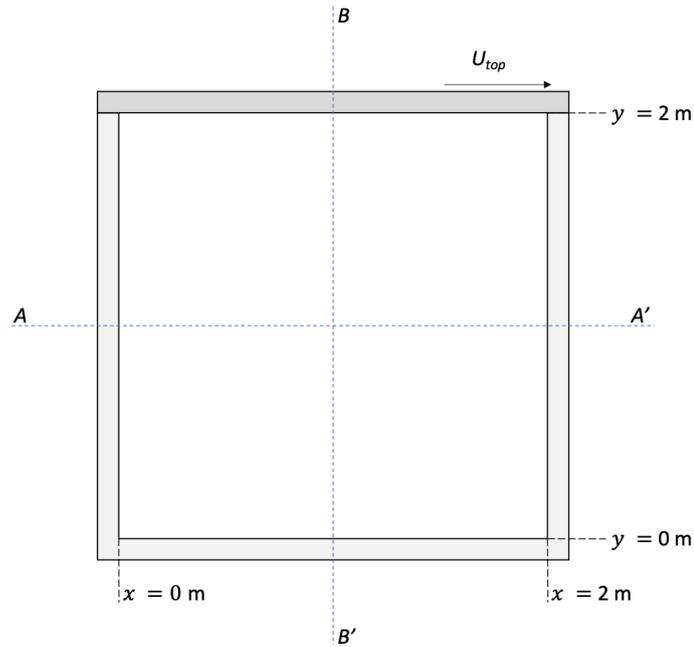


Figure 4.1: Schematic overview of the simulation domain, with observation center-lines AA' and BB' [35]

Isotope	Li-6	Li-7	Be-9	F-19	U-235
Atomic fraction [%]	2.11488	26.0836	14.0992	56.3969	1.30545

Table 4.1: Fuel salt composition of salt in reactor core [35]

The Tiberga benchmark prescribes nuclear data-sets, retrieved from the JEFF-3.1 library [19] at $T = 900K$ generated with Serpent [23] and then grouped into six energy groups that were already tested in previous works to resemble the real spectrum of an MSFR. The definition of the energy groups is found in table 4.2.

Group number, g	Upper energy bound [MeV]
1	$2.000 \cdot 10^1$
2	$2.231 \cdot 10^0$
3	$4.979 \cdot 10^{-1}$
4	$4.479 \cdot 10^{-2}$
5	$5.531 \cdot 10^{-3}$
6	$7.485 \cdot 10^{-4}$

Table 4.2: Definition of the energy groups [35].

The delayed precursor families are grouped by their decay constants. The definition of the families is found in table 4.3

Family number, d	Decay constant [s^{-1}]
1	$1.24667 \cdot 10^{-2}$
2	$2.82917 \cdot 10^{-2}$
3	$4.25244 \cdot 10^{-2}$
4	$1.33042 \cdot 10^{-1}$
5	$2.92467 \cdot 10^{-1}$
6	$6.66488 \cdot 10^{-1}$
7	$1.63478 \cdot 10^0$
7	$3.55460 \cdot 10^0$

Table 4.3: Definition of precursor families [35].

All other nuclear data for the six energy groups and 8 families can be found in appendix B.

4.2. Phase 0.1, Velocity field in lid driven cavity

To assess the capability of describing the velocity field of the code a steady-state simulation is performed that only models the velocity field. The momentum source is a moving lid, that drags the fluid at the top. The relevant physical parameters are found in table 4.4.

Property	Units	Value
ρ	$kg\ m^{-2}$	$2.0 \cdot 10^3$
ν	$m^2\ s^{-1}$	$2.5 \cdot 10^{-2}$
U_{lid}	$m\ s^{-1}$	0.5
Re	–	40

Table 4.4: Physical parameters used in the velocity simulations [35].

The velocity field is modelled using the LB equations as described in section 3.1.1. The simulation is performed until convergence of the velocity and density fields. The only momentum source, the moving lid, is numerically implemented at the top boundary as described in section 3.2.1. The horizontal and vertical components of the velocity field along center-lines AA' and BB' , as shown in figure 4.1, are the observables for the performance of the code, in line with the benchmark by M. Tiberga et al. [35], and displayed in figure 4.2 and 4.3 respectively.

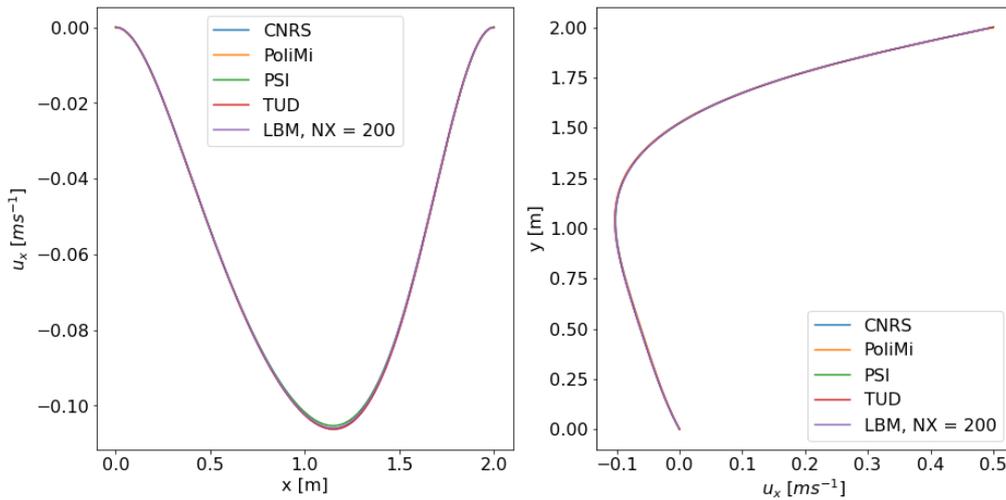


Figure 4.2: Horizontal velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.1.

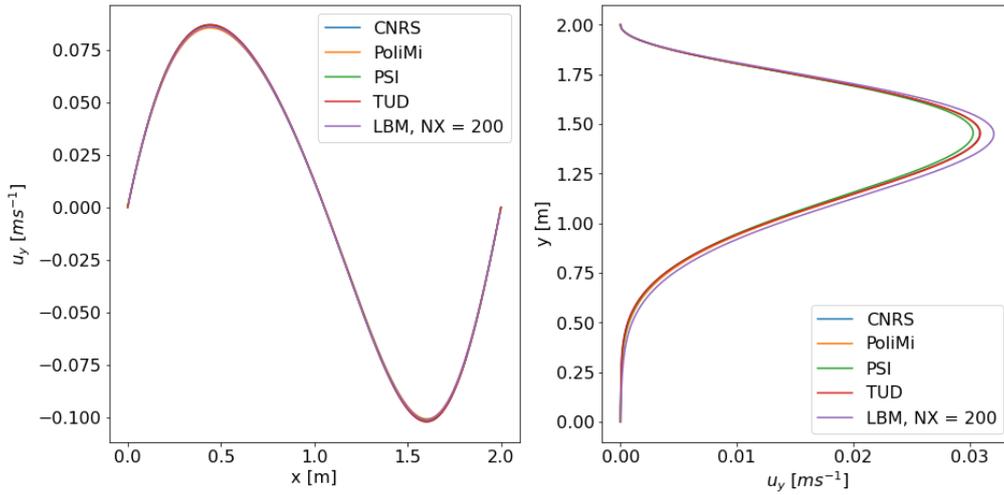


Figure 4.3: Vertical velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.1.

To assess the results on grid convergence, the grid is refined and the discrepancy from the benchmark results is determined using equation 4.2. The results for this convergence study is found in figure 4.4. The discrepancies converge to approximately constant values, and grid convergence is assumed to be achieved at 200 grid-points. At that grid refinement found discrepancies are for the vertical and horizontal velocity along center-line BB' are 5.6% and 1.6% respectively and for the the vertical and horizontal velocity along center-line AA' are 1.1% and 0.40% respectively.

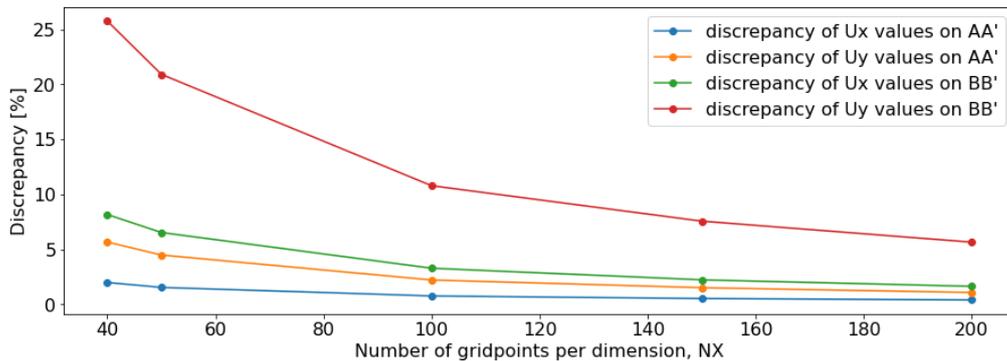


Figure 4.4: Discrepancy from benchmark codes of the horizontal (U_x) and vertical (U_y) velocity fields along center-lines for different numbers of grid-point per dimension in phase 0.1.

4.3. Neutronics field modelled using several benchmarks

To assess the capability of modelling the neutronics field, the neutronics is also modelled separately. In the benchmark by M. Tibergera et al. [35] six groups and eight precursor families are prescribed. To ensure accuracy, first two simpler situations are simulated. The most simple case is a model of two neutron groups, neglecting precursors. Subsequently, a case is assessed with two neutron groups and one precursor family. Finally, the benchmark situation, with six energy groups and eight families is simulated and assessed. For this latter simulation the difference in accuracy of the results between the NEES and EB conditions are discussed.

4.3.1. Neutronics field in a homogeneous solid fuel reactor core

In the relatively simple two-dimensional case where the neutrons can be grouped in one fast and one thermal group, neglecting upscattering and considering zero flux at the boundaries, the eigenvalue problem can be solved analytically. For the fast and thermal groups respectively, one finds the following equations

$$-D_1 \nabla^2 \phi_1(x, y) + \Sigma_{a1} \phi_1(x, y) = \frac{1}{k_{eff}} \nu \Sigma_f \phi_1(x, y) , \quad (4.4)$$

$$-D_2 \nabla^2 \phi_2(x, y) + \Sigma_{a2} \phi_2(x, y) = \Sigma_{s,1 \rightarrow 2} \phi_1(x, y) . \quad (4.5)$$

Considering the fundamental modes and implementing that the scalar neutron fluxes are zero at the boundaries, one finds the general solutions

$$\phi_1(x, y) = \Phi_1 \cos\left(\frac{\pi x}{L}\right) \cos\left(\frac{\pi y}{L}\right) \quad (4.6)$$

$$\phi_2(x, y) = \Phi_2 \cos\left(\frac{\pi x}{L}\right) \cos\left(\frac{\pi y}{L}\right) \quad (4.7)$$

with Φ_1 and Φ_2 constants for the magnitude of the fluxes. Substituting these into the equations above, expressions for the flux magnitudes can be found as

$$\left(2D_1 \left(\frac{\pi}{L}\right)^2 + \Sigma_{a1} - \frac{1}{k_{eff}} \nu \Sigma_f\right) \Phi_1 = 0 \quad (4.8)$$

$$\left(2D_2 \left(\frac{\pi}{L}\right)^2 + \Sigma_{a2}\right) \Phi_2 = \Sigma_{s,1 \rightarrow 2} \Phi_1 \quad (4.9)$$

and considering only non-trivial solutions, solving for k_{eff} results in

$$k_{eff} = \frac{\nu \Sigma_f}{2D_1 \left(\frac{\pi}{L}\right)^2 + \Sigma_{a1}} , \quad (4.10)$$

and for the relation between Φ_1 and Φ_2

$$\Phi_1 = \frac{2D_2 \left(\frac{\pi}{L}\right)^2 + \Sigma_{a2}}{\Sigma_{s,1 \rightarrow 2}} \Phi_2 . \quad (4.11)$$

The developed code for the steady-state neutronics k-eigenvalue is first tested against a simple 2-group benchmark study [7], that uses the analytical solutions described above as a reference. The study considers a square bare reactor core of 140cm sides, as can be seen in 4.5.

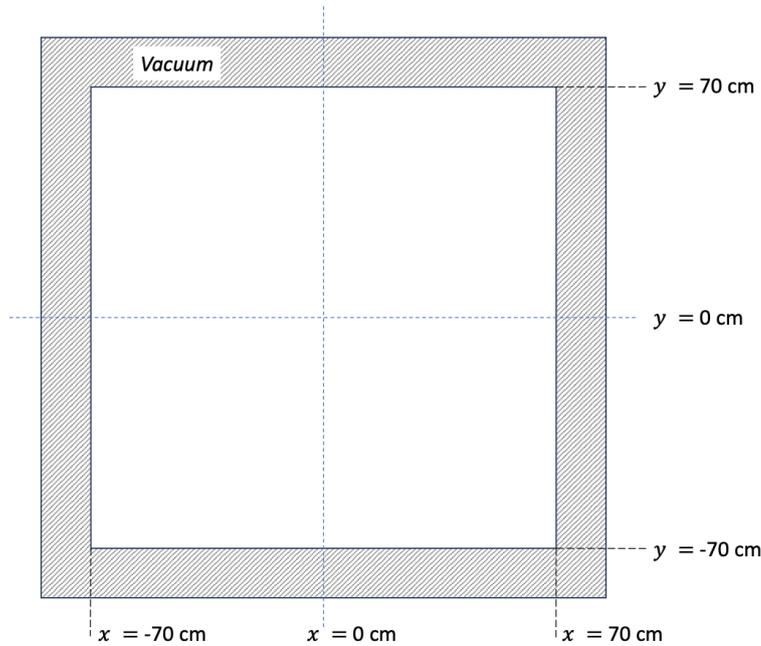


Figure 4.5: Square two-dimensional homogeneous multiplying cavity with two neutron energy groups.

The k-eigenvalue problem is solved using the LB equations as described in section 3.1.3 and the power method as in 3.5 for a reactor core using four different two-energy group nuclear data-sets. The EB condition was used for modelling the vacuum boundaries. The nuclear parameters are displayed in table 4.5.

Material zone	$\Sigma_{r,1}$	$\Sigma_{r,2}$	$\Sigma_{s,1-2}$	$\nu\Sigma_{f,1}$	$\nu\Sigma_{f,2}$	D_1	D_2
1	0.02935	0.2029	0.01563	0.0181	0.2117	1.438	0.2976
2	0.03562	0.24321	0.0205	0.0138	0.2243	1.124	0.2456
3	0.03620	0.1210	0.0241	0.0085	0.1851	1.267	0.3540
4	0.03562	0.24321	0.0205	0.0148	0.2673	1.0	0.2456

Table 4.5: Nuclear data for two-energy group systems [7]

The results are displayed in table 4.6, together with the analytical results [7].

Material zone	k_{eff} (anal. ref.)	k_{eff} (LBM)	RRMSE [%]
1	1.116423	1.116403	0.0018
2	0.889391	0.889352	0.0044
3	1.207672	1.207628	0.0037
4	1.018580	1.018530	0.0049

Table 4.6: Comparison of the neutronics code with analytical result [7]

For the material zone 1, the flux profiles are displayed in figure 4.6 together with the analytical solutions as obtained above. As can be seen in the figure, the results of the developed model correspond well to the sinusoidal analytical profile, with an RRMSE of 1.3%. The normalised flux profiles of material zone 2, 3 and 4 can be found in appendix D.1, with RRMSE of 1.0%, 1.1% and 0.89% respectively.

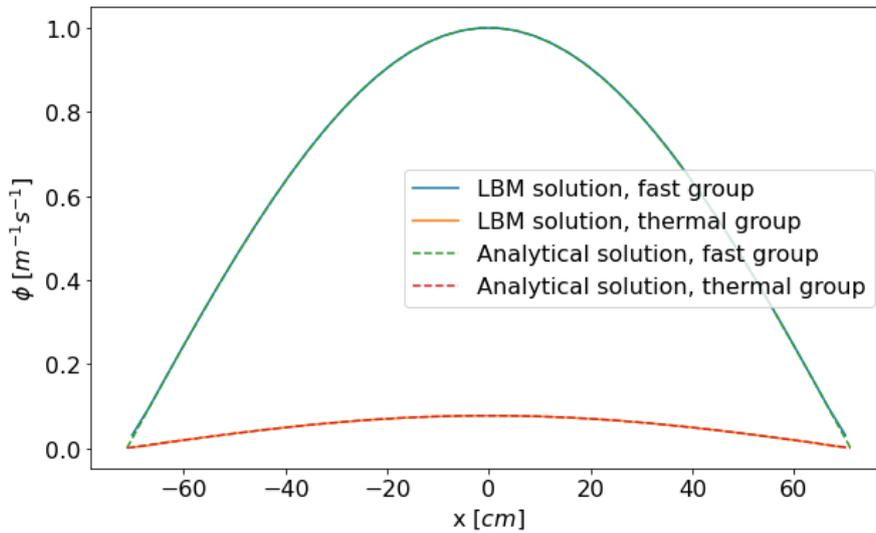


Figure 4.6: Analytical and LBM solution for a two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 1.

The scalar neutron flux does not go to zero at the walls, as the boundary node lies outside the reactor core and goes to zero at that boundary node.

4.3.2. Neutronics field in a multi-region solid fuel reactor core

The second simulation for the neutronics uses the TWIGL benchmark problem dating from 1969 [12]. It describes a non-homogeneous reactor core divided into five regions of three different materials, as can be seen in figure 4.7.

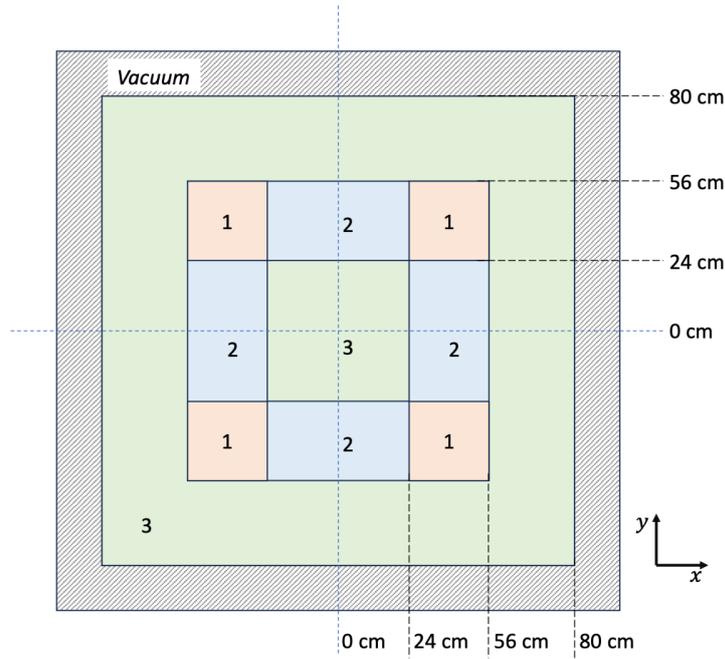


Figure 4.7: TWIGL benchmark dimensions: Square two-dimensional multiplying cavity filled with seed regions 1 and 2 and blanket region 3 [12, 48].

Regions 1 and 2 are core regions containing primary fissile material, region 3 is a blanket region that also contains fissile material. The k-eigenvalue problem is solved using the LB equations as described in section 3.1.3 and the power method as in 3.5. The EB condition was used for modelling the vacuum boundaries. In this situation, one precursor group is considered. In the core no liquid fuel is present leading to no precursor transport by convection. Further, diffusion effects for the precursors are neglected. The treatment of the precursors in the LB equations is therefore simplified from the equations in section 3.1.4 through omitting the collision term, resulting in

$$c_{d,i}(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = c_{d,i}(\mathbf{r}, t) + \Delta t Q_{d,i}(\mathbf{r}, t) . \quad (4.12)$$

The rest of the equations in section 3.1.4 remain the same. The nuclear parameters are found in table 4.7.

Material	Group	$\Sigma_{r,g}$	$\Sigma_{s,1-2}$	$\nu \Sigma_{f,g}$	D_g	v_g
1	1	0.02	0.01	0.007	1.4	$1.0 \cdot 10^7$
	2	0.15	—	0.2	0.4	$2.0 \cdot 10^5$
2	1	0.02	0.01	0.007	1.4	$1.0 \cdot 10^7$
	2	0.15	—	0.2	0.4	$2.0 \cdot 10^5$
3	1	0.018	0.01	0.003	1.3	$1.0 \cdot 10^7$
	2	0.05	—	0.06	0.5	$2.0 \cdot 10^5$

Table 4.7: Nuclear parameters for the seed and blanket regions in the TWIGL benchmark [48], [12], [43].

The k-eigenvalue problem is solved and the results for the scalar neutron fluxes are found in figure 4.8. The eigenvalue is found as 0.908258, whereas the reference solution for the eigenvalue was 0.913237. This results in an RRMSE of 0.545% for the eigenvalue. The reference solution for the neutron scalar fluxes of the two energy groups are found in appendix D.2 and correspond well to the figure below.

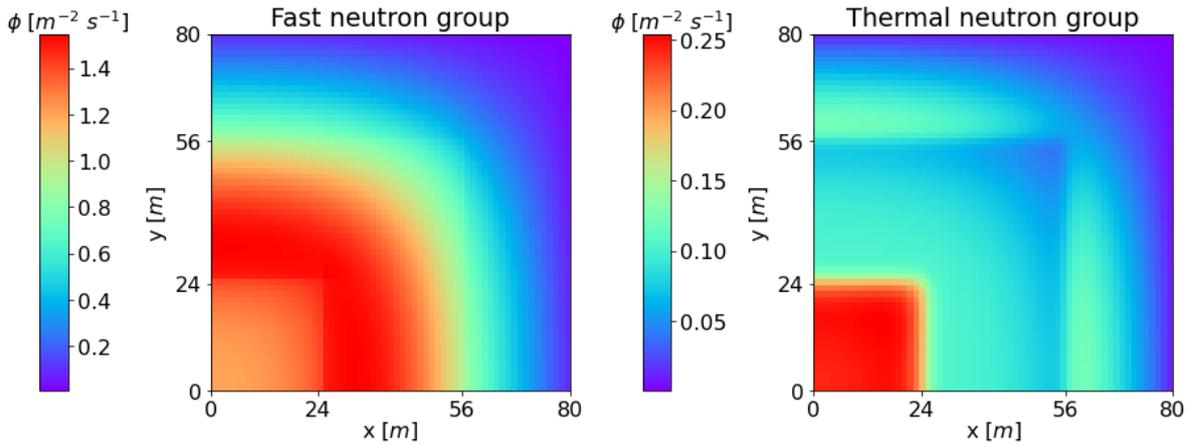


Figure 4.8: LB solution for scalar neutron fluxes of the two energy groups in the TWIGL benchmark [12] in the top-right quarter of the simulation domain.

The flux profile across the domain is assessed over the diagonal, that is in regions 1 and 3. The flux is lowest at the walls in the blanket region, increases in the seed region and decreases again in the blanket region.

4.3.3. Phase 0.2, Static fuel with six neutron energy groups and eight neutron precursor families

In this phase, the reactor is filled with a static, isothermal fuel salt. Static flow results in no convection affecting precursor distribution and the generated heat does not affect the flow of the salt or the neutronic parameters. Further, diffusion effects for the precursors in this phase are neglected. The physical properties are displayed in table 4.8. After convergence of the model the scalar neutron fluxes are normalised for the total reactor power of 1 *GW*. The solving scheme is found in appendix D.

Property	Units	Value
P	W	$1.0 \cdot 10^6$
T	K	900

Table 4.8: Physical parameters used in the neutronics simulations [35].

The nuclear parameters for the simulation are found in appendix B. The equations as described in section 3.1.3 and 3.1.4 are used for the neutron group fluxes and the precursor concentrations respectively, but for the precursors again the collision operator is omitted to exclude diffusion effects, as was expressed in equation 4.12. The k-eigenvalue problem is solved using the power method as in 3.5. The used numerical scheme for this uncoupled system is found in appendix A. The NEES and EB schemes are both used and their results are compared for different grid refinements. This is displayed in figure 4.9.

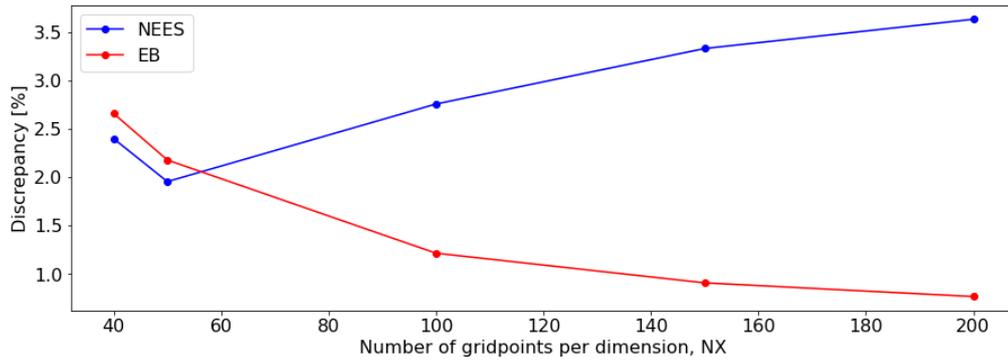


Figure 4.9: Discrepancy from benchmark codes of the fission rate density for different numbers of grid-points per dimension, for the EB and NEES boundary schemes in phase 0.2.

The discrepancy from the benchmark results is determined using equation 4.2. As seen, the discrepancy for the NEES boundaries increases with increasing grid refinement. The EB is more capable of resembling the vacuum boundary condition and the discrepancy seems to converge to a constant value. Grid convergence is assumed to be achieved at 200 grid-points. At this refinement the discrepancy for the fission rate density is found as 0.77% and for k_{eff} as 0.065%. The results corresponding to the simulation with this lowest discrepancy (EB scheme, grid refinement of $NX = 200$) are found in figure 4.10, compared with the different codes from the benchmark by M. Tiberga et al. [35].

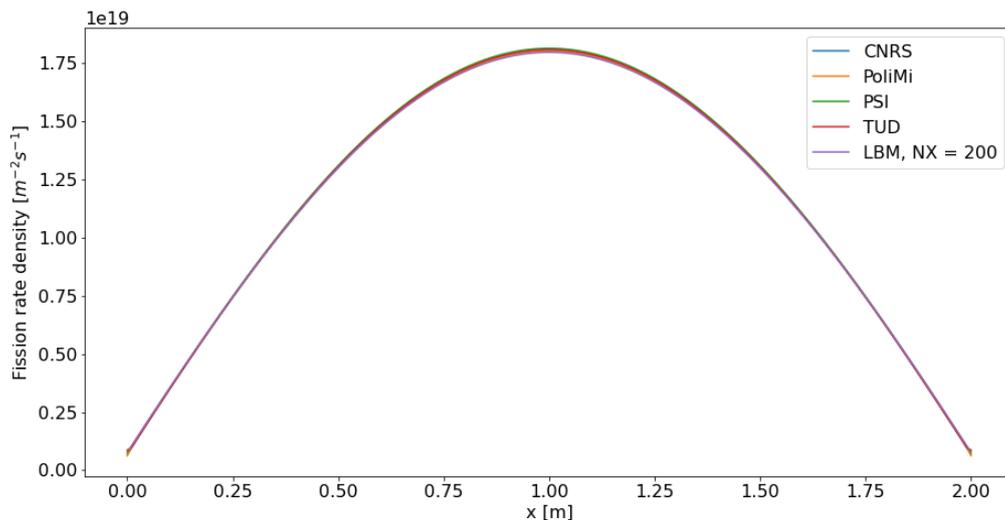


Figure 4.10: Fission rate density along center-line AA' for the various benchmark codes and the LBM simulation in phase 0.2.

One again recognises the non-zero flux at the reactor walls. For the flux profile we recognise the sinusoidal shape that we found in the simpler case for the 2-group neutronics problem as well. The found multiplication factors that correspond to the flux profiles in figure 4.10 are found in table 4.9.

Code	k_{eff}	ρ [pcm]	$\epsilon_{k_{eff}}$
CNRS-SP ₁	1.00412999	411.3	0.014%
CNRS-SP ₃	1.00354955	353.7	0.013%
PSI	1.00422982	421.2	0.014%
PoliMi	1.00413402	411.7	0.014%
TUD-S ₂	1.0048494	482.6	0.016%
TUD-S ₆	1.00581461	578.1	0.018%
LBM	1.00523287	520.6	0.017%

Table 4.9: Multiplication factor k_{eff} and reactivity ρ found for the various benchmark codes and the LBM simulation, together with the discrepancies for the multiplication factors in phase 0.2 [7].

4.4. Phase 0.3, Temperature

To assess the capability of modelling the temperature field a simulation is performed that uses a fixed flow field and heat source. The velocity field found in section 4.2 is used as static input in this phase. The fission rate density found in section 4.3.3 results in fission heat that is used as a heat source in the current temperature simulation. Note that the flow field affects heat convection, but the effect of convection on precursors is neglected. The temperature field is modelled using the LB equations as described in 3.1.2 until convergence of the temperature. The physical properties as prescribed in the benchmark are displayed in table 4.10.

Property	Units	Value
γ	$W m^{-2} K^{-1}$	$1.0 \cdot 10^6$
c_p	$J K^{-1}$	$3.075 \cdot 10^3$
α	$m^2 s^{-1}$	$8.13 \cdot 10^{-8}$
ρ	m^{-2}	$2.0 \cdot 10^3$
Pr	–	$3.075 \cdot 10^5$

Table 4.10: Physical parameters prescribed for the temperature field in the benchmark study [35].

As described in section 3.4.1, these prescribed respectively high and low values for Pr and α result in the necessity for extremely small grid refinement to ensure the stability condition 3.43. As described in section 3.4.1, the high Pr results in dominance of convection over diffusion, even in regions with large gradients in temperature. To assess the effect of the implementation of lower Pr numbers than prescribed in this simulation, the Pr is varied and the discrepancy from the benchmark codes is assessed using equation 4.2. Further, the discrepancies along the center-lines are assessed for different Pr numbers, for equal grids refinement.

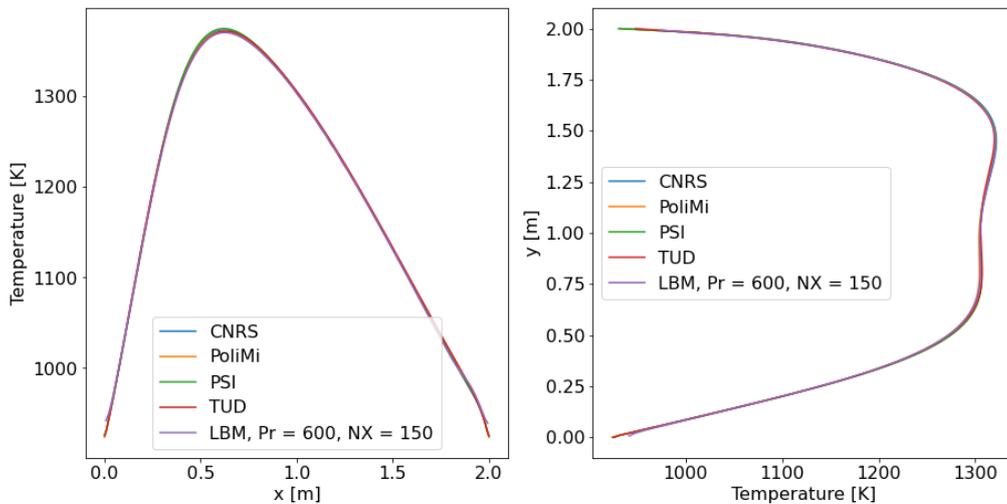


Figure 4.11: Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 0.3.

The discrepancy of the with the LBM determined temperature field from the other codes is found for the horizontal (AA') and vertical (BB') center-line as 0.34% and 0.38% respectively.

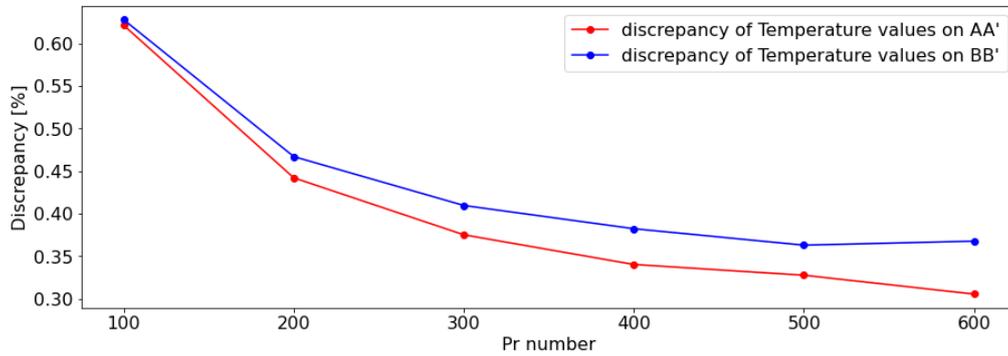


Figure 4.12: Discrepancy from benchmark codes of the temperature fields along center-lines for different implemented Pr numbers in phase 0.3.

As can be seen in figure 4.12, the discrepancy of the result compared to the benchmark codes converges to a constant value for increasing Pr . Further increasing Pr results in the necessity of higher grid refinement, increasing numerical effort, while not significantly increasing accuracy.

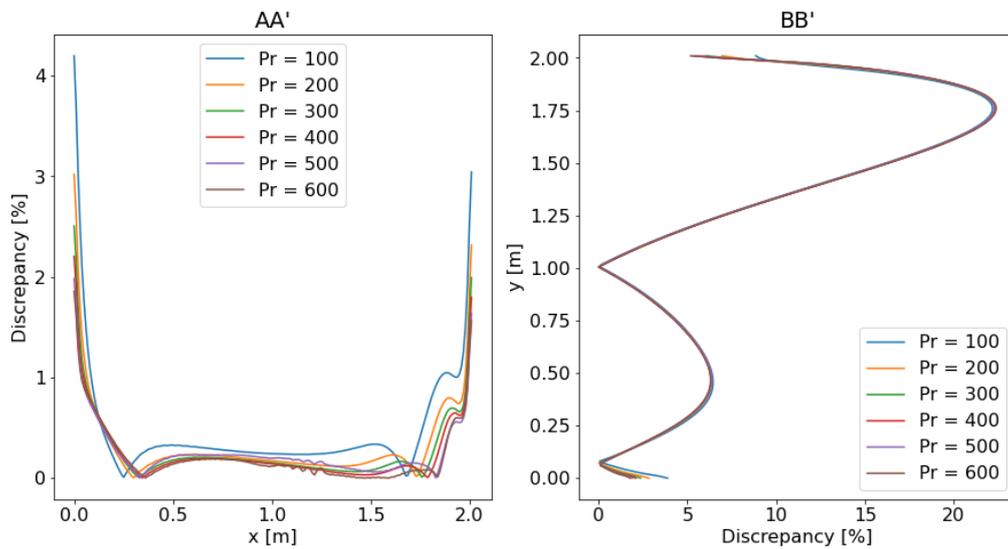


Figure 4.13: Discrepancy from benchmark codes of the temperature fields at different locations in the core along center-lines AA' (left) and BB' (right) for different implemented Pr numbers in phase 0.3.

Further, the discrepancy across the core was assessed based on the same simulations as for the results in figure 4.12. Two remarks have to be made. The first is to conclude that indeed the discrepancies from the benchmark codes, that were modelled using $Pr = 3.075 \cdot 10^5$, are most significant where temperature gradients are highest, for instance at the walls. Similarly, when gradients are closest to zero (e.g. at $y = 1.00m$), discrepancies approach zero as well. This can be explained by remembering that lower values for Pr are implemented in the simulations than what was prescribed in the benchmark by M. Tiberger et al. [35]. As stated in section 3.4.1, when gradients in temperature are high while the Pr number is low the convection and heat sink terms do not necessarily dominate. In the case with lower Pr numbers the gradient in temperature, responsible for diffusion, is more significant than it would be in the high Pr number case. Secondly, it is seen that the discrepancies from the benchmark codes converge to one line in figure 4.13. Apparently, for a high enough chosen Pr number, the diffusion effect is overcome even for the highest temperature gradients in this simulation.

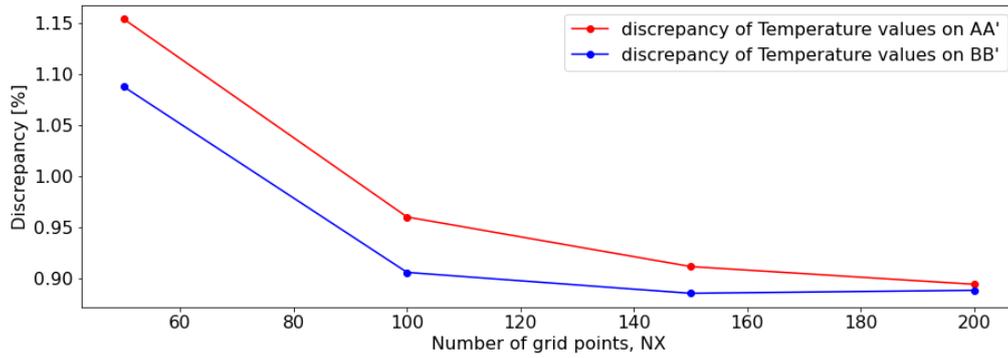


Figure 4.14: Discrepancy from benchmark codes of the temperature field along center-lines for different numbers of grid-point per dimension in phase 0.3.

Again, to assess the results on grid convergence, the grid is refined and the discrepancy from the benchmark results is determined using equation 4.2. The results of this convergence study is found in figure 4.14. The discrepancies converge to approximately constant values, and grid convergence is assumed to be achieved at 200 grid-points, where the found discrepancies for the horizontal (AA') and vertical (BB') center-line are 0.89% and 0.89% respectively.

4.5. Phase 1.1, Circulating fuel

This phase simulates a cavity with a fixed velocity field as in the first phase with uniform temperature. This means that the heat generated through fission does not affect the velocity, temperature or neutronics field. In this situation, the effect of convection on the precursor distribution is assessed, which in turn has an effect on the fission rate in the core. The k-eigenvalue problem is solved using the LB equations as described in section 3.1.3 and 3.1.4 and the power method as in 3.5. The velocity field is passively implemented and is not solved throughout the simulation. The fixed velocity field has its effect in the calculation of the precursor concentrations. The reactor power is normalised at $1GW$. The physical properties as prescribed in the benchmark are displayed in table 4.11. The solving scheme is found in appendix D.

Property	Units	Value
T	K	900
D_p	$m^2 s^{-1}$	$1.25 \cdot 10^{-10}$
Sc	–	$2.0 \cdot 10^8$

Table 4.11: Physical parameters prescribed for the precursor fields in the benchmark study [35].

Again, as described in section 3.4.1, these prescribed respectively high and low values for Sc and D_p result in the necessity for extremely small grid refinement to ensure the stability condition 3.45. Lower Sc numbers are implemented and the effect of this implementation is discussed later. First, the NEES and EB schemes are both used and their results are compared for different grid refinements. This is displayed in figure 4.15

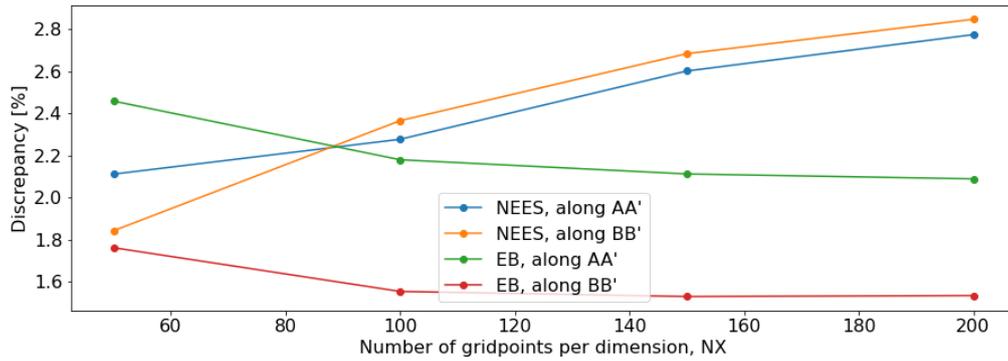


Figure 4.15: Discrepancy from benchmark codes of the fission rate density for different numbers of grid-points per dimension, for the EB and NEES boundary schemes in phase 1.1. Implemented Sc number is 100.

The discrepancy from the benchmark results is determined using equation 4.2. As seen, the discrepancy for the NEES boundaries again increases with increasing grid refinement. The EB is more capable of resembling the vacuum boundary condition and the discrepancy seems to converge to a constant value. Grid convergence is assumed to be achieved at 200 grid-points for $Sc = 100$.

As described in section 3.4.1 and similar to the reasoning for the high Pr number, high Sc results in dominance of convection over diffusion, even in regions with large gradients in mass concentration. To assess the effect of the implementation of lower Sc in this simulation, the Sc is varied and the discrepancy from the benchmark codes is assessed for different Sc values using equation 4.2. The results for $NX = 150$ and $Sc = 400$ are shown in figure 4.16, where the discrepancy for the delayed neutron source over AA' and BB' are 1.6% and 1.3% respectively.

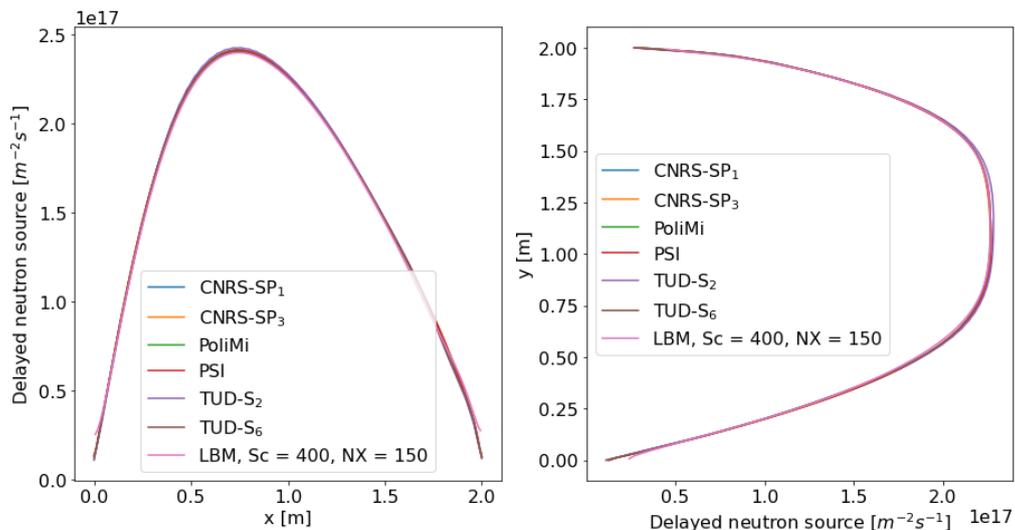


Figure 4.16: Delayed neutron source along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.1.

Code	k_{eff}	ρ	$\rho - \rho_{0.2}$	$\epsilon_{k_{eff}}$
CNRS-SP ₁	1.00350021	348.8	-62.5	0.013%
CNRS-SP ₃	1.00291950	291.1	-62.2	0.011%
PSI	1.00360495	359.2	-62.0	0.013%
PoliMi	1.00349920	348.7	-63.0	0.013%
TUD-S ₂	1.00422377	420.6	-62.0	0.014%
TUD-S ₆	1.00520091	517.4	-60.7	0.017%
LBM, $Sc = 400$	1.00457311	455.2	-65.3	0.015%

Table 4.12: Multiplication factor k_{eff} , reactivity ρ and reactivity change from static fuel $\rho - \rho_{0.2}$ found for the various benchmark codes and the LBM simulation, together with the discrepancies for the multiplication factors in phase 1.1 [7].

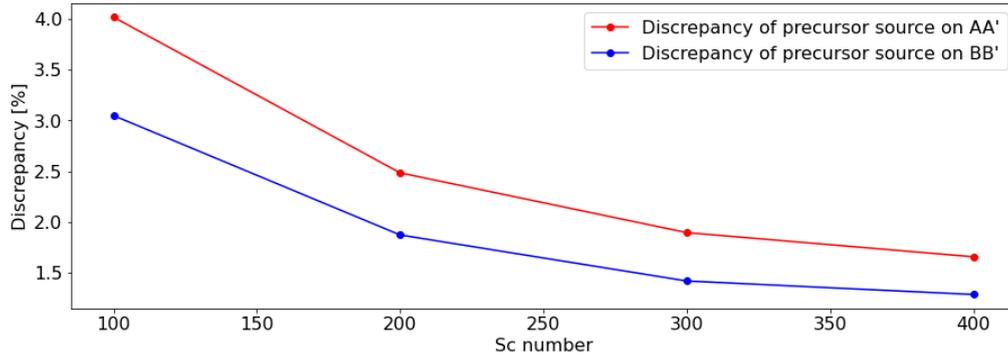


Figure 4.17: Discrepancy from benchmark codes of the precursor source along center-lines for different implemented Sc numbers in phase 1.1.

As can be seen in figure 4.17, the discrepancy of the LBM result compared to the benchmark codes converges to a constant value for increasing Sc . Further increasing Sc results in the necessity of higher grid refinement, increasing numerical effort, while not significantly increasing accuracy.

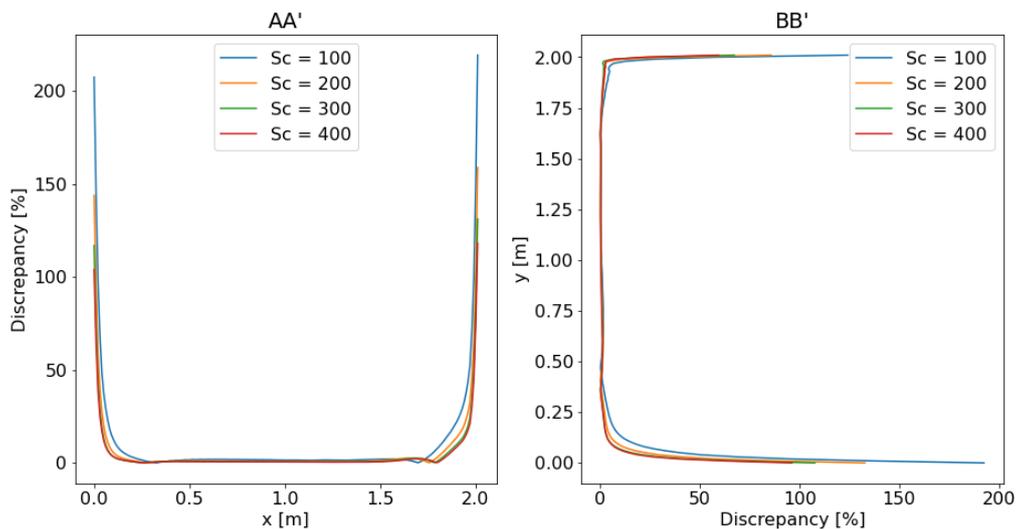


Figure 4.18: Discrepancy from benchmark codes of the precursor source at different locations in the core along center-lines AA' (left) and BB' (right) for different implemented Sc numbers in phase 1.1.

Again the discrepancy across the core was assessed. The discrepancies from the benchmark codes, that were modelled using $Sc = 2.0 \cdot 10^8$, are most significant where the concentration gradients changes are highest, i.e. at the walls. Similarly, when gradients are closest to zero (e.g. at $y = 1.00m$), discrepancies go to zero. Again, it is important to remember that lower values for Sc are implemented

in the simulations than what was prescribed in the benchmark by M. Tiberga et al. [35]. As stated in section 3.4.1, when concentration gradients are high while the Sc number is low the convection and reaction rate terms do not necessarily dominate. In cases with lower Sc numbers the relatively high concentration gradients impose more significant diffusion effects than it would be in the high Sc number case. Again, it is also seen that the discrepancies from the benchmark codes converge to one line in figure 4.18. Apparently, for a high enough chosen Sc number, the diffusion effect is overcome even for the highest concentration gradients in this simulation.

4.6. Take-aways for the coupling of fields

From these previously described uncoupled simulations, it can be concluded that certain simulation conditions ensure accurate results. The following intermediate conclusions are drawn:

- For the neutronics simulations, the implementation of equilibrium boundary conditions results in lower discrepancy from the benchmark code, especially for more grid-points per dimension, than the implementation of the non-equilibrium extrapolation scheme. Therefore, for the neutronics field minimum grid-refinement of 200 lattice points per dimension is advised and the equilibrium boundary conditions should be implemented.
- For all fields, a minimum grid-refinement of 200 lattice points per dimension is advised.
- Pr and Sc numbers of approximately 400 result in a practical balance between accuracy and numerical costs. Increasing Pr and Sc results in the necessity of increased grid-refinement to ensure stability.

5

Validation of steady-state coupled models

The results obtained from the models for steady-state coupling are displayed here. In all cases, the criticality eigenvalue problem is solved. For the neutronics, the EB condition is used at the boundaries, due to the found lower discrepancies for this scheme in the previous chapter. In this chapter, we also include the simulation parameters. This is crucial because the interaction between the fields introduces multiple stability and accuracy criteria, which may potentially conflict with each other when chosen incorrectly. In section 5.1 a simulation is performed with power coupling, in which temperature effects are included in the scaling of neutronics parameters. In section 5.2, a coupled simulation is performed with power coupling extended with buoyancy effects. In section 5.3, a coupled simulation is performed with power coupling, buoyancy effects and an external momentum source. In section 5.4 the fields are discussed on their differences between how fields interact.

5.1. Phase 1.2, Power coupling

In this phase, the temperature feedback on the neutronics is simulated. The fuel temperature is non-uniform and the fission heat is taken into account. This fission heat affects the neutron diffusion constants and the neutron reaction cross-sections. The flow field is again fixed as in the first phase and the reactor power is normalised at $1GW$. Physical properties for the temperature field are implemented as displayed in table 4.10. Physical properties for the precursor fields are implemented as displayed in table 4.11. The implemented scheme can be found in figure 5.1. The stability criteria described in section 3.4 are taken into account. The chosen combination of grid refinement x^* , Pr , Sc , lattice viscosity ν^* and the resulting Ma and t^* are displayed in table 5.1.

Simulation parameters	Value
x^*	0.01
ν^*	0.7
Ma	0.24
Pr	200
Sc	200
τ_f^*	2.6
τ_h^*	0.5105
τ_c^*	0.5105

Table 5.1: Implemented simulation parameters in phase 1.2

Note that the simulation is performed with higher Ma than prescribed in equation 3.36. The temperature field and fission rate change compared to phase 0.2 for all codes are found in figures 5.2 and 5.3. It is expected that the higher Ma does not contribute too much to the simulation error, since the velocity field is in this case not actively solved, but implemented as a static field. The temperature field

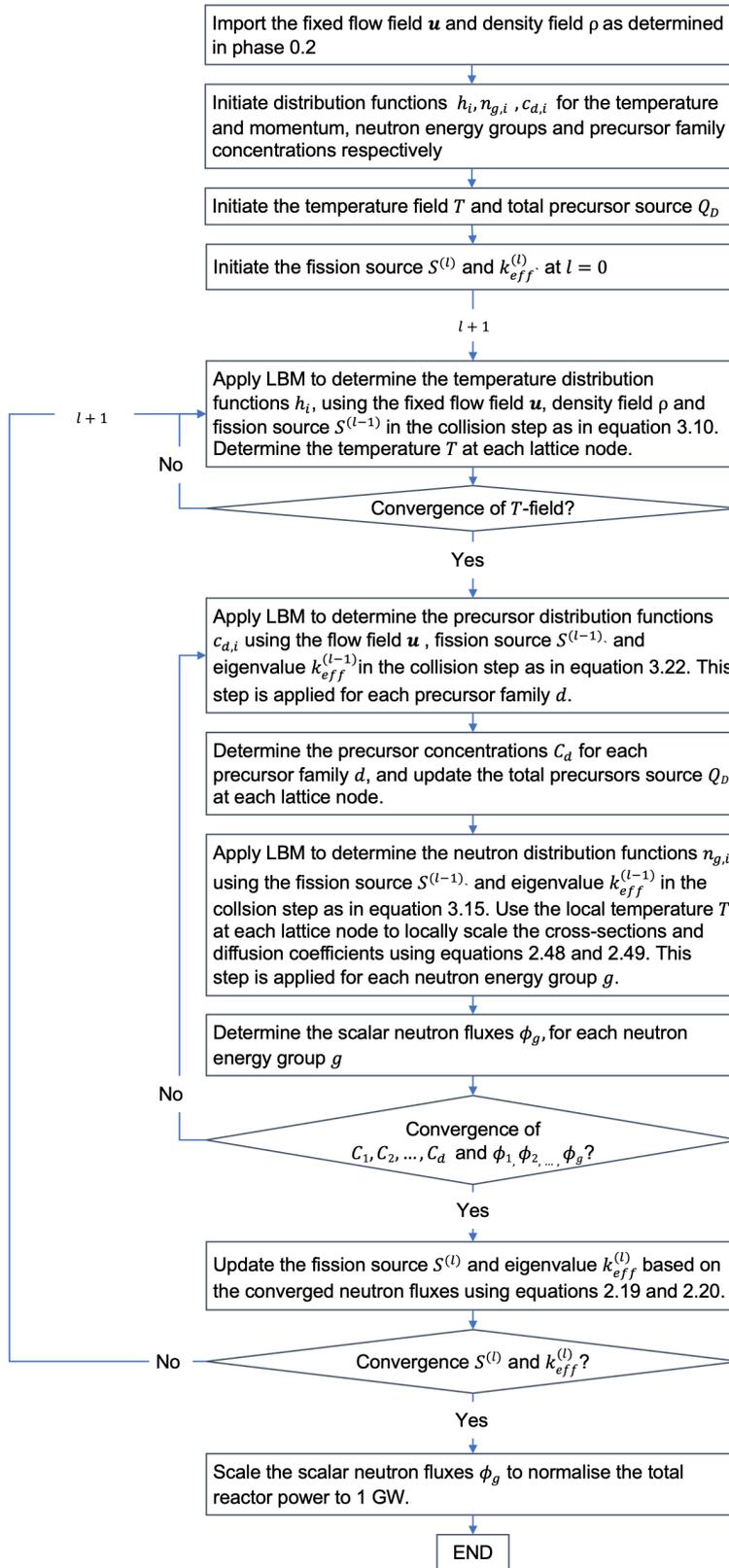


Figure 5.1: Iteration scheme for phase 1.2

and change in fission rate density (from phase 0.2) along center-lines are displayed in figures 5.2 and 5.3 respectively.

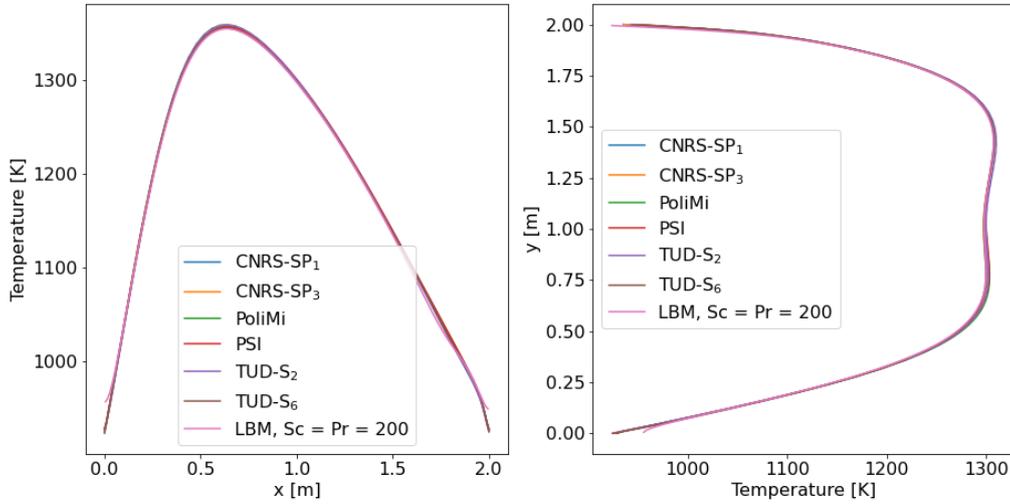


Figure 5.2: Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.2.

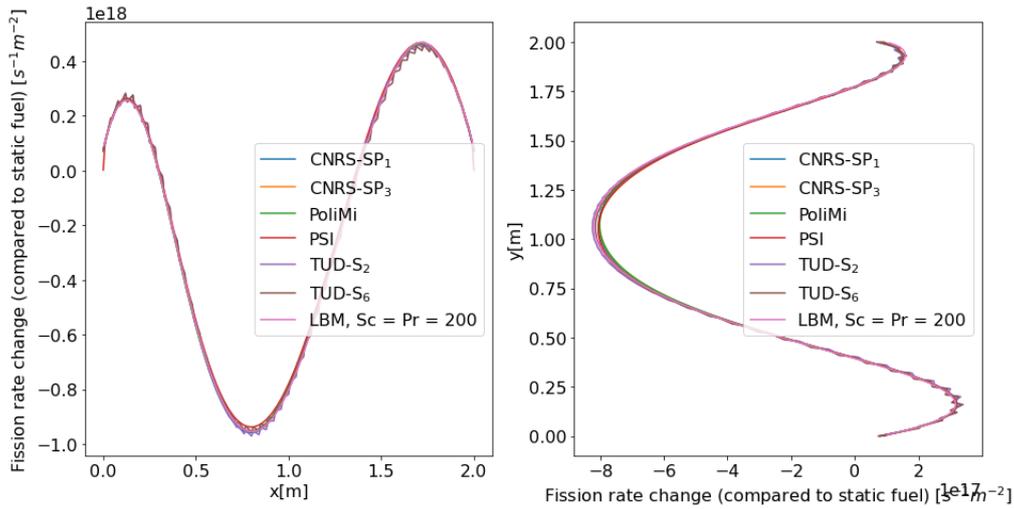


Figure 5.3: Change in fission rate density (compared to uncoupled fission rate density) along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.2.

The effective multiplication factor k_{eff} , the reactivity ρ and the change in reactivity from phase 1.1 $\rho_{1.2} - \rho_{1.1}$ for all codes are found in table 5.2.

Code	k_{eff}	ρ	$\rho - \rho_{1.1}$
CNRS-SP ₁	0.9920320	-803.2	-1152.0
CNRS-SP ₂	0.9914576	-861.6	-1152.7
PSI	0.9920458	-801.8	-1161.0
PoliMi	0.9920035	-806.1	-1154.8
TUD-S ₂	0.9928062	-724.6	-1145.2
TUD-S ₆	0.9939903	-604.6	-1122.0
LBM, $Pr = Sc = 200$	0.99288174	-716.9	-1172.1

Table 5.2: Multiplication factor k_{eff} , reactivity ρ and reactivity change from circulating fuel without temperature feedback $\rho - \rho_{1.1}$ found for the various benchmark codes and the LBM simulation in phase 1.2 [7].

The results are shown for a simulation with 200 grid-points and $Pr = Sc = 200$. The discrepancies are determined by equation 4.2 and are found in table 5.3. The discrepancies for the LB method are an order of magnitude larger than the discrepancies of the different codes. This is most likely due to the use of $Pr = Sc = 200$, while in the previous chapter it was seen that Pr and Sc convergence are at $Pr = Sc = 400$.

Code	$\epsilon_{T,AA'}$	$\epsilon_{T,BB'}$	$\epsilon_{FRC,AA'}$	$\epsilon_{FRC,BB'}$	$\epsilon_{k_{eff}}$
CNRS-SP ₁	0.080%	0.066%	1.2%	6.8%	0.000013%
CNRS-SP ₃	0.079%	0.071%	9.7%	8.7%	0.000088%
PSI	0.096%	0.11%	0.99%	2.4%	0.000012%
PoliMi	0.099%	0.10%	1.0%	1.1%	0.000015%
TUD-S ₂	0.098%	0.10%	1.9%	1.8%	0.000018%
TUD-S ₆	0.094%	0.098%	1.2%	2.2%	0.000260%
LBM, $Pr = Sc = 200$	0.32%	0.42%	2.0%	2.1%	0.000025%

Table 5.3: Discrepancies of the different codes for different fields in phase 1.2 [7]

5.2. Phase 1.3, Buoyancy

In this phase, again the temperature feedback on the neutronics is simulated and additionally the temperature and velocity field are coupled through buoyancy. The fuel temperature is non-uniform and the fission heat affects the neutron diffusion constants and the neutron reaction cross-sections. The top lid is not moving and the reactor power is again normalised at $1GW$. Physical properties for the fluid field are implemented as displayed in table 4.4. Physical properties for the temperature field are implemented as displayed in table 4.10. Physical properties for the precursor fields are implemented as displayed in table 4.11. The implemented scheme can be found in figure 5.4.

The stability criteria described in section 3.4 are taken into account. The chosen combination of grid refinement x^* , Pr , Sc , lattice viscosity ν^* and the resulting Ma and τ^* are displayed in table 5.4. In this simulation, the low Ma was necessary for the results to be accurate. It is expected that this is the case since in this simulation the velocity field is actively solved, contrarily to the previous phase.

Simulation parameters	Value
x^*	0.004
ν^*	0.44
Ma	0.10
Pr	300
Sc	300
τ_f^*	1.82
τ_h^*	0.5066
τ_c^*	0.5066

Table 5.4: Implemented simulation parameters in phase 1.3

The horizontal and vertical velocities along the center-lines are displayed in figures 5.5 and 5.6 respectively. The temperature field and delayed precursor source along center-lines are displayed in figures 5.7 and 5.8 respectively.

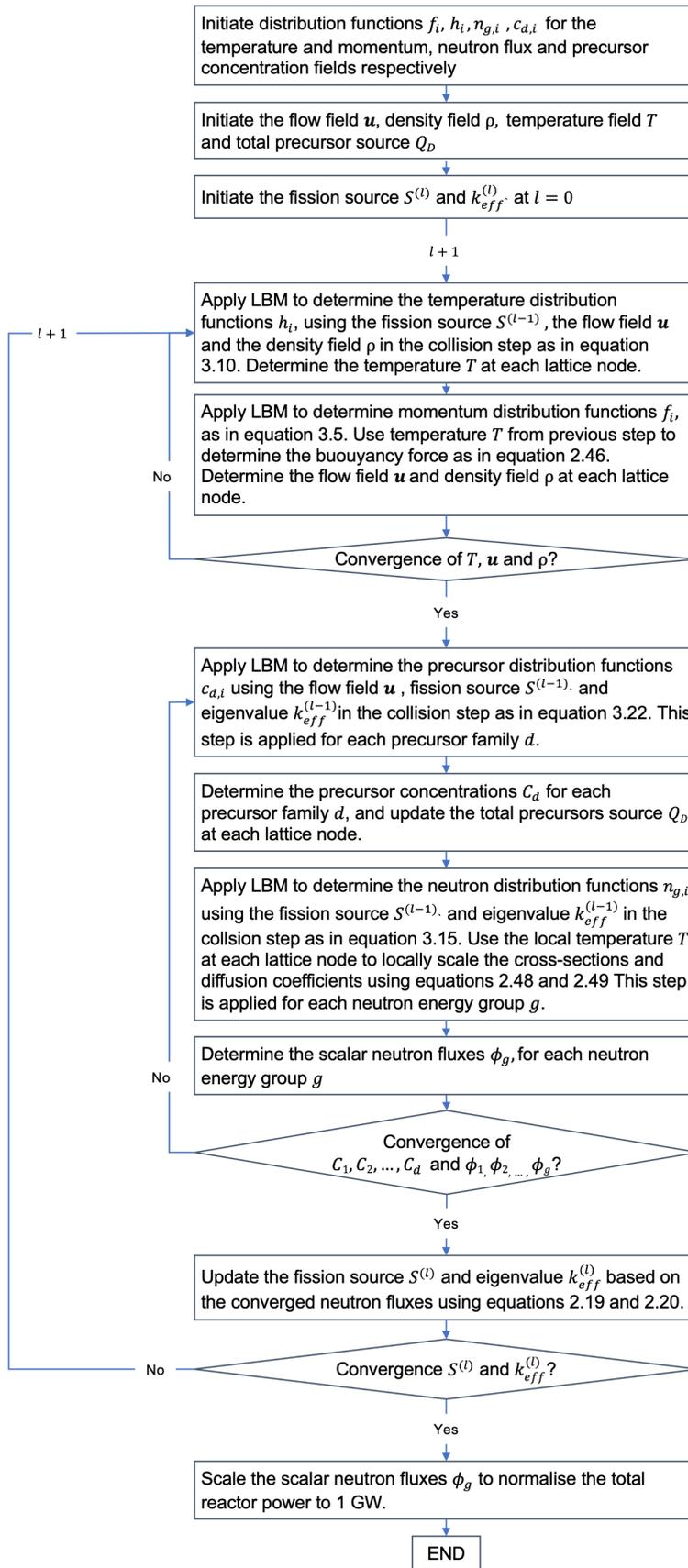


Figure 5.4: Iteration scheme for phase 1.3

Phase 1.3, Different models

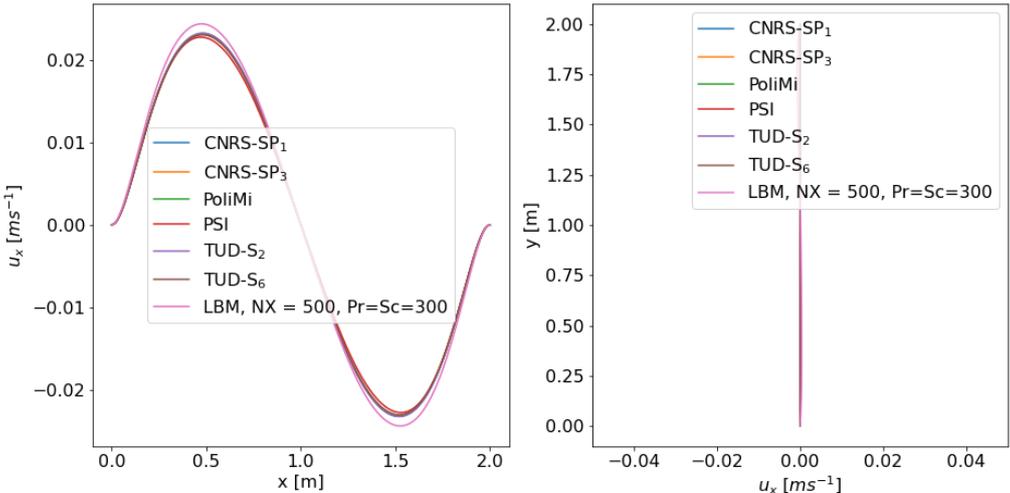


Figure 5.5: Horizontal velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.

Phase 1.3, Different models

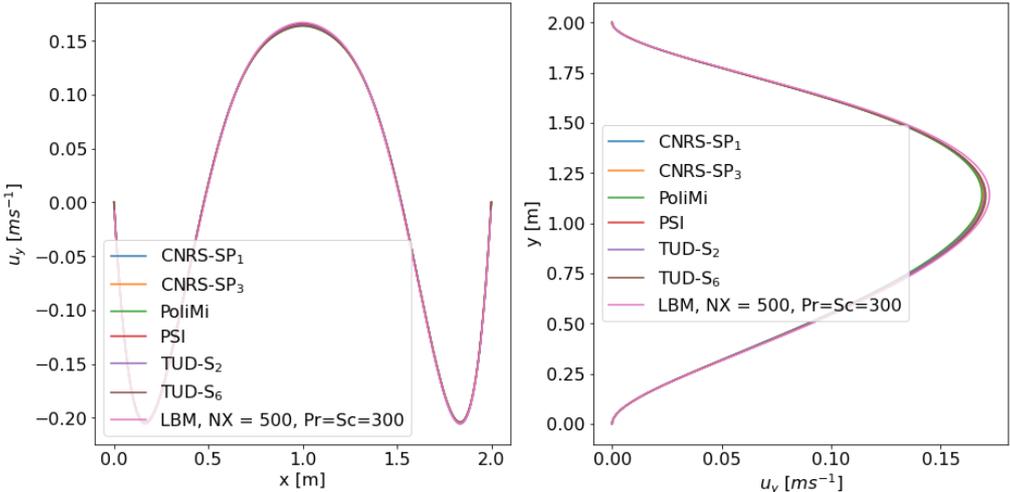


Figure 5.6: Vertical velocity field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.

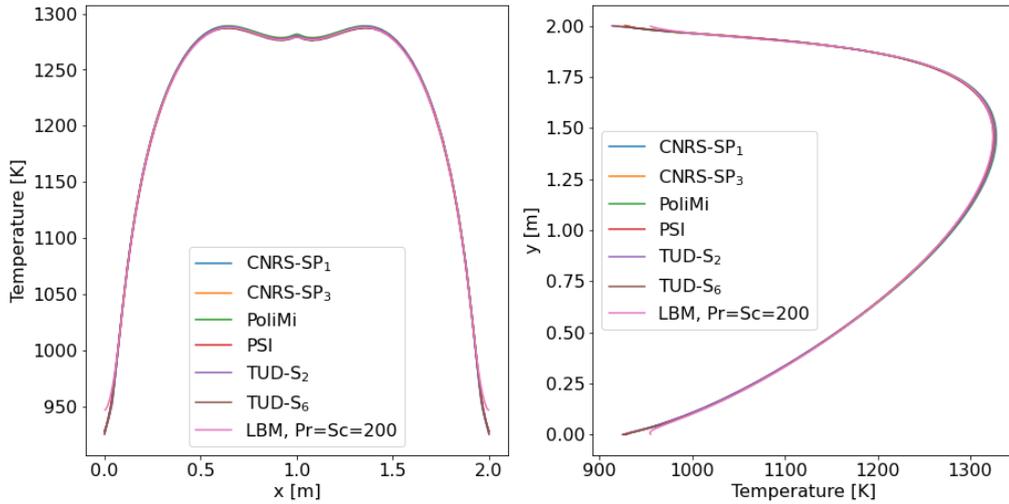


Figure 5.7: Temperature field along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.

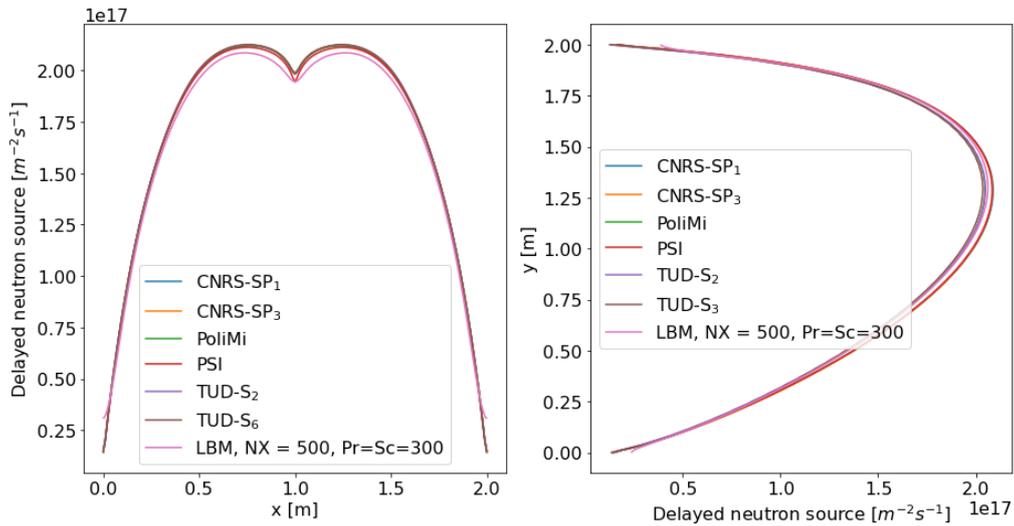


Figure 5.8: Delayed neutron source along center-lines AA' (left) and BB' (right) for the various benchmark codes and the LBM simulation in phase 1.3.

The effective multiplication factor k_{eff} , the reactivity ρ and the change in reactivity from phase 0.2 $\rho_{1.3} - \rho_{0.2}$ for all codes are found in table 5.2.

Code	k_{eff}	ρ	$\rho_{1.3} - \rho_{0.2}$
CNRS-SP ₁	0.9919730	-809.2	-1220.5
CNRS-SP ₃	0.9914045	-867.0	-1220.7
PSI	0.9920064	-805.8	-1227.0
PoliMi	0.9919857	-807.9	-1219.6
TUD-S ₂	0.9927933	-725.9	-1208.5
TUD-S ₆	0.9939735	-606.3	-1184.4
LBM, $Pr = Sc = 300$	0.99278435	-726.8	-1247.4

Table 5.5: Multiplication factor k_{eff} , reactivity ρ and reactivity change from the fully uncoupled neutronics solution $\rho - \rho_{0.2}$ found for the various benchmark codes and the LBM simulation in phase 1.3 [7].

The results are shown for a simulation with 500 grid-points and $P = Sc = 300$. The discrepancies are

determined by equation 4.2 and are found in table 5.6. The discrepancies for the effective multiplication factor are found in table 5.7. The discrepancies for the LB method are an order of magnitude larger than the discrepancies of the different codes. This is most likely due to the use of $Pr = Sc = 300$, while in the previous chapter it was seen that Pr and Sc convergence are at $Pr = Sc = 400$.

Code	$\epsilon_{u_x,AA'}$	$\epsilon_{u_y,AA'}$	$\epsilon_{u_y,BB'}$	$\epsilon_{T,AA'}$	$\epsilon_{T,BB'}$	$\epsilon_{DNP,AA'}$	$\epsilon_{DNP,BB'}$
CNRS-SP ₁	0.53%	0.20%	0.14%	0.025%	0.051%	0.12%	1.3%
CNRS-SP ₃	0.19%	0.10%	0.33%	0.030%	0.044%	0.12%	1.2%
PSI	1.9%	0.46%	0.55%	0.078%	0.089%	5.3%	1.1%
PoliMi	0.53%	0.39%	0.64%	0.085%	0.11%	0.14%	0.92%
TUD-S ₂	0.78%	0.40%	0.14%	0.056%	0.092%	0.17%	1.2%
TUD-S ₆	0.23%	0.42%	0.55%	0.071%	0.096%	0.17%	1.5%
LBM, $Pr = Sc = 300$	5.9%	1.5%	1.5%	0.30%	0.39%	2.4%	2.6%

Table 5.6: Discrepancies of the different codes for different fields in phase 1.3 [7]

Code	$\epsilon_{k_{eff}}$
CNRS-SP ₁	0.000015%
CNRS-SP ₃	0.000092%
PSI	0.000012%
PoliMi	0.000014%
TUD-S ₂	0.000019%
TUD-S ₆	0.000266%
LBM, $Pr = Sc = 300$	0.000019%

Table 5.7: Discrepancies of the different codes for the effective multiplication factor in phase 1.3 [7]

5.3. Phase 1.4, Full coupling

In this phase, the coupling scheme of phase 1.3 is repeated, but with the external momentum source of the moving top lid at a velocity of $0.5ms^{-1}$. The temperature feedback on the neutronics is simulated and the temperature and velocity field are coupled through buoyancy. The fuel temperature is non-uniform and the fission heat affects the neutron diffusion constants and the neutron reaction cross-sections. The reactor power is again normalised at $1GW$. Physical properties for the fluid field are implemented as displayed in table 4.4. Physical properties for the temperature field are implemented as displayed in table 4.10. Physical properties for the precursor fields are implemented as displayed in table 4.11. The implemented scheme can be found in figure 5.9.

The stability criteria described in section 3.4 are taken into account. The chosen combination of grid refinement x^* , Pr , Sc , lattice viscosity ν^* and the resulting Ma and t^* are displayed in table 5.8.

The stability criteria described in section 3.4 are taken into account. The chosen combination of grid refinement x^* , Pr , Sc , lattice viscosity ν^* and the resulting Ma and τ^* are displayed in table 5.8. In this simulation, the low Ma was necessary for the results to be accurate. It is expected that this is the case since in this simulation the velocity field is actively solved.

Simulation parameters	Value
x^*	0.004
ν^*	0.35
Ma	0.048
Pr	100
Sc	100
τ_f^*	1.55
τ_h^*	0.5105
τ_c^*	0.5105

Table 5.8: Implemented simulation parameters in phase 1.4

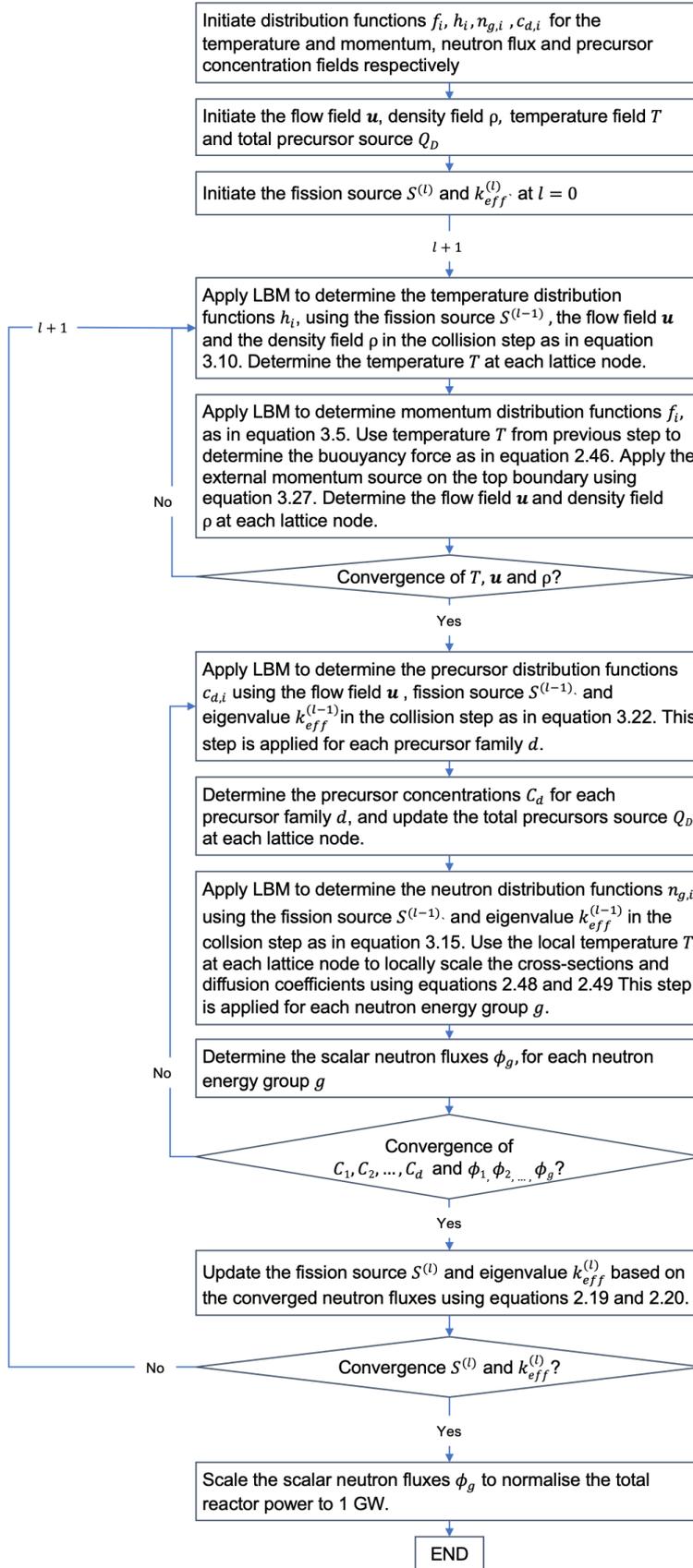


Figure 5.9: Iteration scheme for phase 1.4

The horizontal and vertical velocities along the center-lines are displayed in figures 5.10 and 5.11 respectively. The temperature field, delayed precursor source and fission rate density along center-lines are displayed in figures 5.12 and 5.13 respectively. Note that the profiles are not compared against the other codes due to a lack of available data.

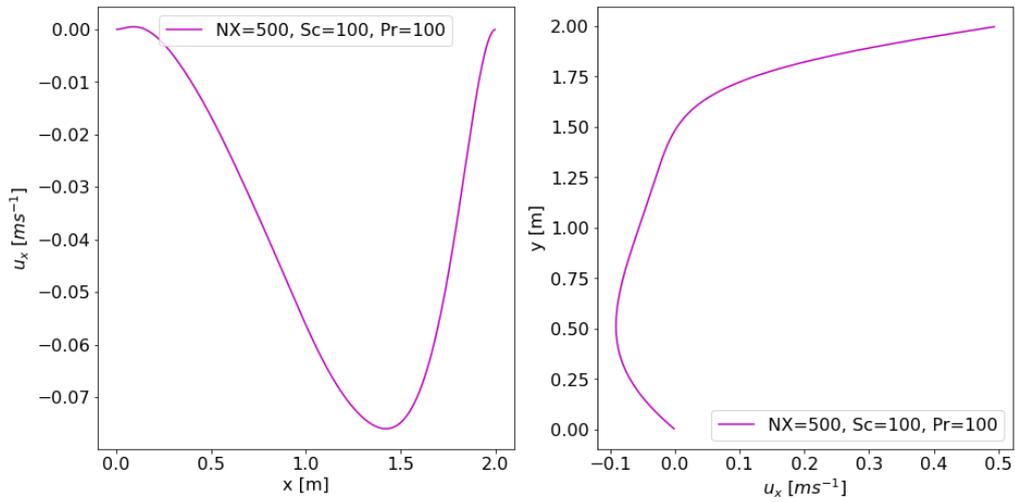


Figure 5.10: Horizontal velocity field along center-lines AA' (left) and BB' (right) in phase 1.4.

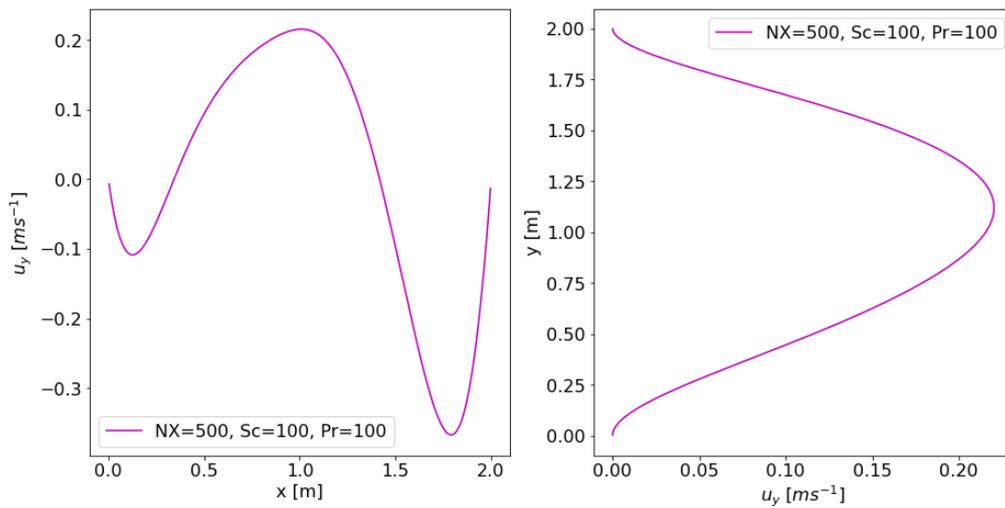


Figure 5.11: Vertical velocity field along center-lines AA' (left) and BB' (right) in phase 1.4.

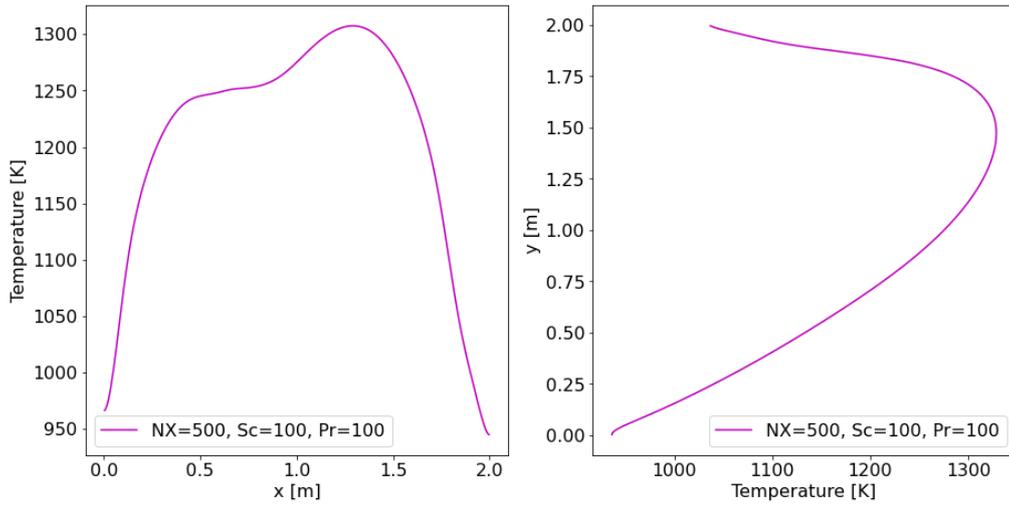


Figure 5.12: Temperature field along center-lines AA' (left) and BB' (right) in phase 1.4.

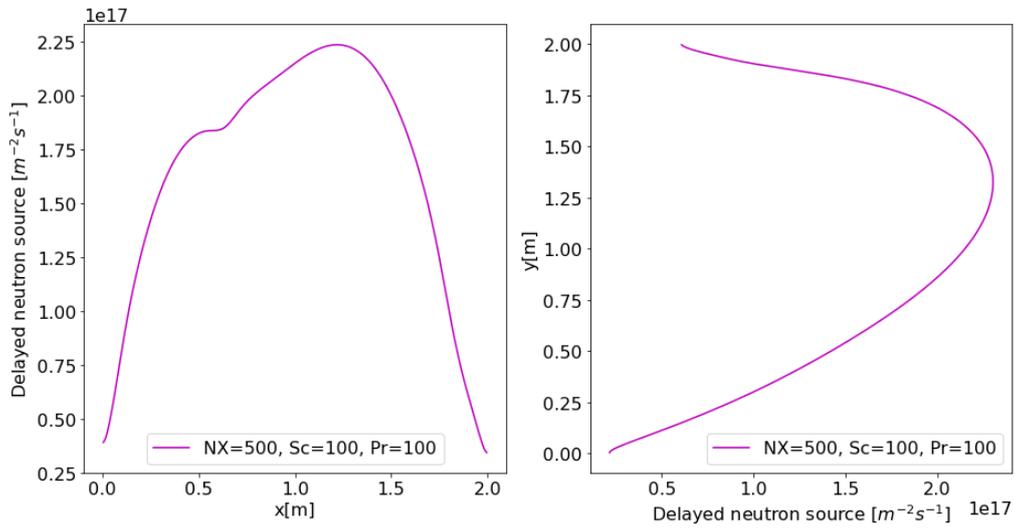


Figure 5.13: Delayed neutron source along center-lines AA' (left) and BB' (right) in phase 1.4.

The effective multiplication factor k_{eff} , the reactivity ρ and the change in reactivity from phase 0.2 $\rho_{1.4} - \rho_{0.2}$ for all codes are found in table 5.2.

Code	k_{eff}	ρ	$\rho_{1.4} - \rho_{0.2}$	$\epsilon_{k_{eff}}$
CNRS-SP ₁	0.9921275	-793,5	-1204.8	0.000015%
CNRS-SP ₁	0.9915569	-851,5	-1205.2	0.000092%
PSI	0.9921344	-792,8	-1214.0	0.000014%
PoliMi	0.9921806	-788,1	-1199.8	0.000011%
TUD-S ₂	0.9929461	-710,4	-1193.0	0.000019%
TUD-S ₆	0.99411879	-591,6	-1169.7	0.000263%
LBM, $Pr = Sc = 100$	0.99316295	-688.4	-1085.3	0.000043%

Table 5.9: Multiplication factor k_{eff} , reactivity ρ , reactivity change from the fully uncoupled neutronics solution $\rho - \rho_{0.2}$, found for the various benchmark codes and the LBM simulation in phase 1.4 [7].

5.4. Comparing the differences between phases

After we have validated the gradually coupled tools, it is interesting to assess what physics is responsible for the differences between the results of the different fields. The different velocity profiles are analysed first, after which the temperature profiles are looked at. Finally, concentration profiles of two precursor families are compared. The reader is referred to the benchmark by M. Tiberga et al. [35] for the obtained fields by the benchmark codes.

5.4.1. Velocity field

A comparison is made between velocity fields that are differently induced. In phase 0.1, the velocity field was driven by a moving top lid. In phase 1.3, the velocity was induced by buoyancy effects from rising temperatures due to fission. In phase 1.4, the velocity was both induced by a moving top lid and buoyancy effects.

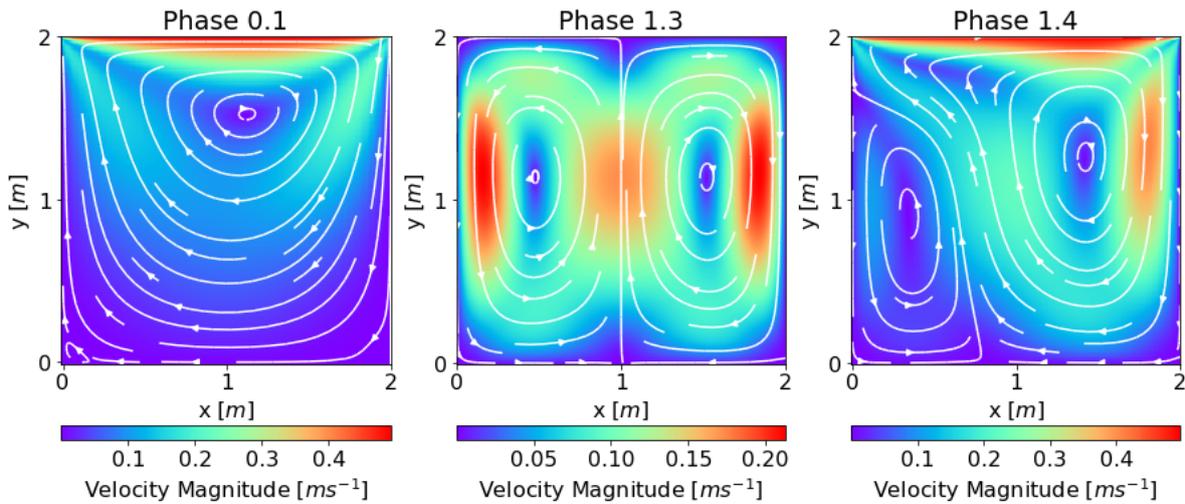


Figure 5.14: Magnitude of the velocity field obtained by LBM simulations for three differently coupled phases.

As can be seen by comparing the different velocity fields found in the different phases of coupling, the velocity field changes when different fields interact. In the left graph of figure 5.14, the uncoupled velocity field is displayed, with the moving lid as a momentum source. The lid is moving to the right with a speed of 0.5ms^{-1} , and it is seen that indeed at the top this maximum speed is achieved. The fluid moves clockwise along with the top lid moving right.

In the middle graph of figure 5.14, no external source of momentum is present. The temperature field affects the velocity field through buoyancy effects. Fission heat adds energy to the fluid resulting in a density difference in the fluid. This density difference imposes a pressure gradient that results in fluid motion. Hot and low density fluid ascends, while cold and high density fluid descends. At the cavity center fission rates peak and temperatures are increased most. Consequently, fluid in that location ascends, to cool down at the top and move left and right towards the boundaries to flow down again. The result is two vertical convection cells with opposite flow direction, the left moving counter-clockwise and the right moving clockwise.

The right graph of figure 5.14 displays a situation where both buoyancy and movement of the top wall affect the velocity field. One still sees two convection cells, but the displacement of the top wall introduces a significant effect, causing a notable velocity shift at the top towards the right.

5.4.2. Temperature field

A comparison is made between temperature fields that are differently affected by the other fields. In phase 0.3, temperature rises due to fission heat, and the velocity field is static as in phase 0.1. Flow affects the temperature distribution through convection. In phase 1.3, temperature again rises due to fission energy and additionally the velocity and temperature field are coupled through buoyancy effects. In phase 1.4, the coupling is the same as in phase 1.3, but with the velocity field driven by both buoyancy effects and the moving top lid.

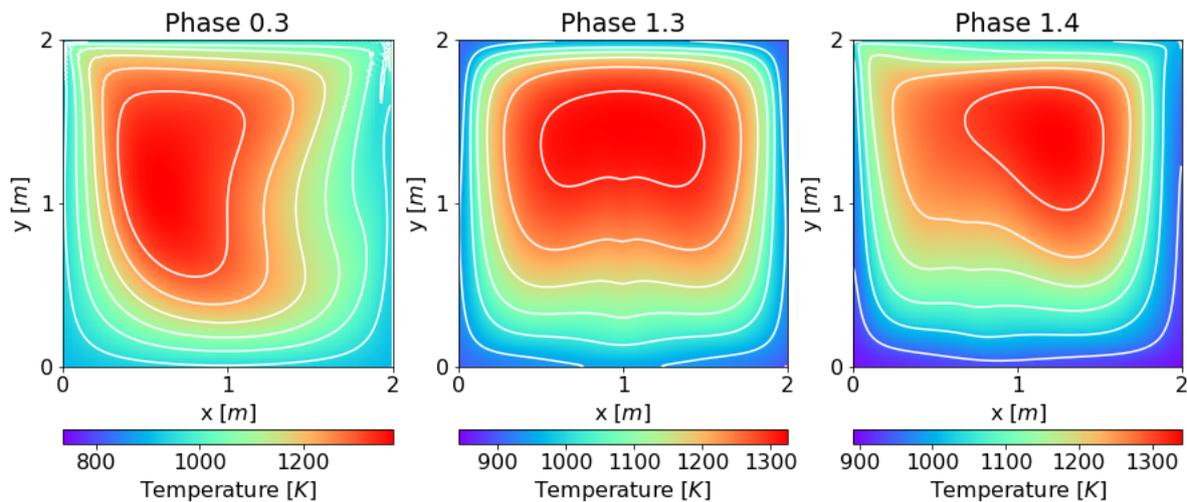


Figure 5.15: Temperature field obtained by LBM simulations for three differently coupled phases.

The left graph of figure 5.15 displays the temperature field in a situation where heat is generated from fission which as stated before peaks at the center of the cavity. A source of momentum is again imposed at the top wall. One can recognise the high temperatures at the center being spread by convection, resulting in an asymmetric temperature profile. In this left situation, the temperature does not in turn affect the fluid flow.

The middle graph of figure 5.15 displays the temperature field in the situation where heat again is generated from fission, but the flow field is not imposed by a momentum source at the top but by buoyancy effects. As previously described, hot fluid ascends, while cold fluid descends resulting in the higher temperatures at the top. Due to the absence of a momentum source at the top, the temperature field is symmetric.

The right graph of figure 5.15 is the result of a temperature field with both buoyancy effects and the external momentum source of the moving top lid. Comparing the temperature field with the velocity field, one sees that the larger convection cell on the right falls together with the high temperature region suggesting that the fluid in that region is less prone to mixing with the rest of the fluid. Further, hot low-density fuel is transported upwards, resulting in higher temperatures at the top.

5.4.3. Precursor concentration fields

A comparison is made between the effect of the velocity field in different phases on the precursor concentrations throughout the core. Two precursor families are assessed. The half-lives of family 1 and 5 are given as 55.6 seconds and 2.37 seconds respectively. The comparison of these two energy groups is chosen to analyse the effect of their different half-lives.

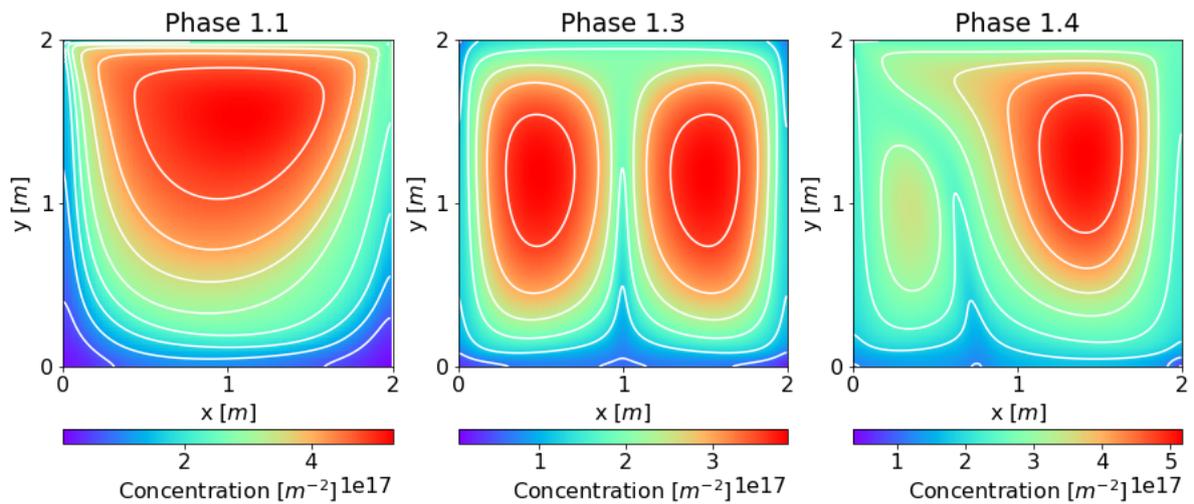


Figure 5.16: Concentration of the first family of precursors ($T_{1/2} = 55.6s$) obtained by LBM simulations for three differently coupled phases.

The result of this difference is clearly seen in the concentration profiles. For all situations, the concentration profiles of family 1 have much more time to shape itself into the velocity profile as described above since the lifetime of the precursors is much larger. For family 5 on the other hand, the precursors have already decayed before the full convection profile is mimicked.

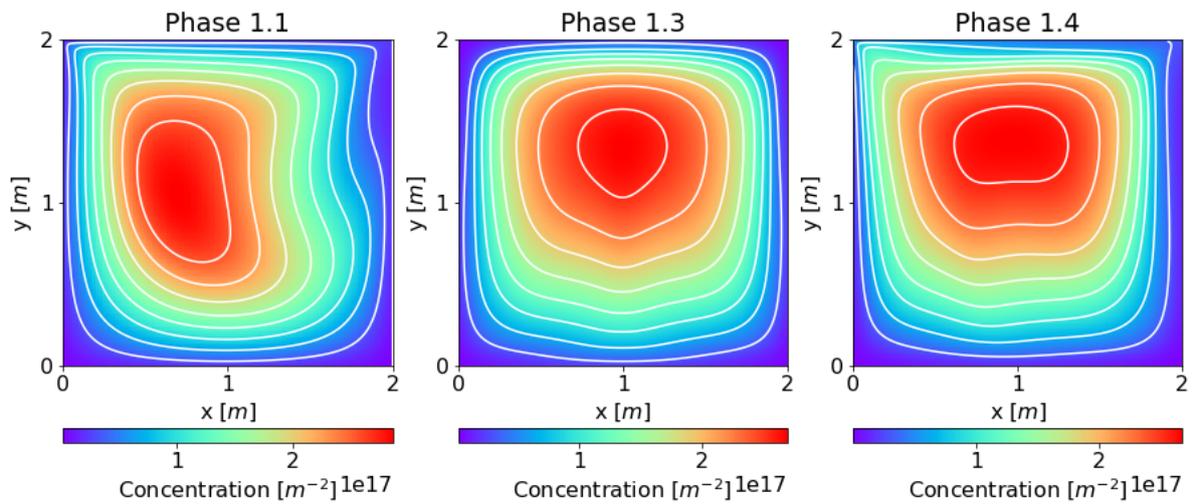


Figure 5.17: Concentration of the fifth family of precursors ($T_{1/2} = 2.37s$) obtained by LBM simulations for three differently coupled phases.

6

Validation of transient coupled models

In this section, results are displayed for the simulation of transient reactor behaviour making use of the integration of the predictor reactor quasi-static method and the lattice Boltzmann method. Three primary initiators of transient behavior can be identified in MSFRs: reactivity insertions, variations in primary pump operations, and alterations in heat exchanger cooling capabilities [11]. In this research reactivity insertions are used to initiate transient behaviour. In the benchmark by M. Tiberger et al. [35] the perturbation in the system is imposed on the heat transfer coefficient that ultimately affects the reactivity of the core. In this chapter, first a simpler perturbed system is simulated using a perturbation directly in the reactivity and the results are displayed in section 6.1. This initial simpler simulation is performed to assess the capability of the integration of PCQSM and LBM, without the solution being affected by computational errors in the PK equations. After, a situation is modelled where the perturbation is imposed in the heat transfer coefficient, of which the results are shown in section 6.2.

6.1. Transient model with perturbed reactivity

First the tool's capability of handling the integration of LBM with PCQSM is tested, without determining the PK parameters. Instead, a transient situation is assessed where a periodically changing reactivity affects the reactor power. Starting from the critical situation of phase 1.4, a reactivity is inserted. The reactivity as a function of time is given by:

$$\rho = \rho_{max} \sin(2\pi ft) . \quad (6.1)$$

It is assessed how the reactor power is affected by the changing reactivity.

Simulation parameters	Units	Value
ρ_{max}	-	0.001
Λ	s	$2.9 \cdot 10^{-5}$ s
f	Hz	2.0

Table 6.1: Point kinetics parameters

The results are shown in figure 6.1.

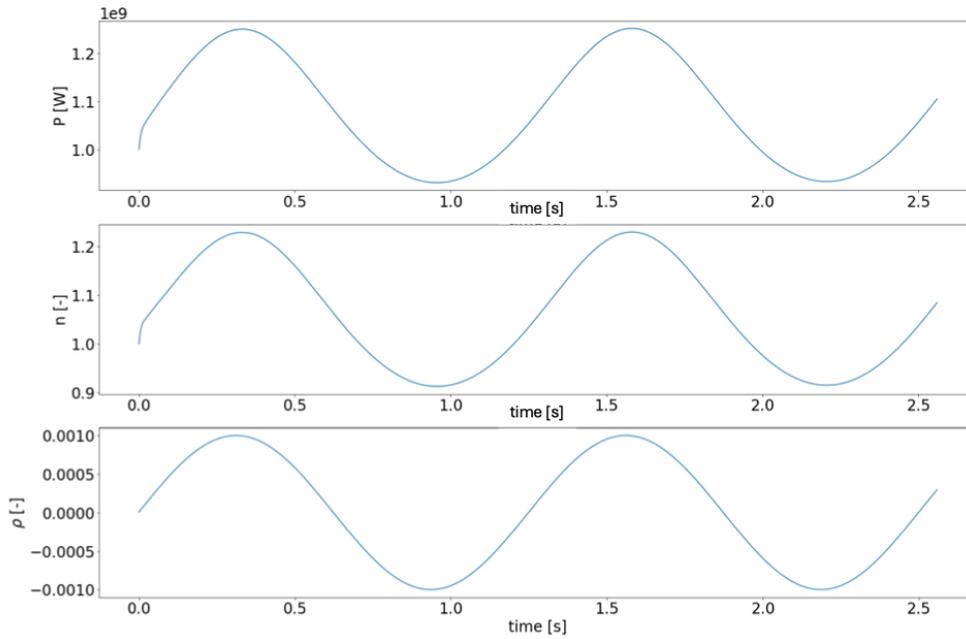


Figure 6.1: Transient results for sinusoidal perturbation of 2.0 Hz in reactivity for the first 2.5 s after the initial perturbation.

As can be seen, the amplitude function and reactor power are affected by the reactivity instantaneously. A prompt jump is observed in the reactor power and amplitude function at the beginning of the simulation, after which the simulation follows the sinusoidal shape of the reactivity. From this simulation it can be concluded that the integration of PCQSM with the LBM was successful and can be expanded with integrating the time-varying PK parameter ρ and Λ .

6.2. Transient model with perturbed heat transfer coefficient

In line with the benchmark by M. Tiberga et al. [35], the transient reactor behaviour is analysed by perturbing the volumetric heat transfer coefficient γ . Starting from the critical situation of phase 1.4, the volumetric heat transfer coefficient is perturbed. The change in γ results in a change in the heat sink as in equation 4.3. Higher γ means that the reactor loses heat more efficiently, while lower γ means that the reactor loses heat less efficiently. The perturbation of γ is uniform throughout the reactor and changes with time according to a sine wave of amplitude 10%. The heat transfer coefficient as a function of time is given as

$$\gamma = \gamma_{initial} + 0.1 \cdot \gamma \cdot \sin(2\pi ft) . \quad (6.2)$$

The frequency is varied, and for each frequency the gain and phase shift is observed. When γ is lower, less heat is removed, and the reactor core's temperature will rise. Conversely, when γ is higher, more heat is removed, and the core's temperature will decrease. The temperature gradient within the reactor will change, impacting the flow and behavior of the molten salt. Changes in the temperature profile of the reactor core will affect neutron moderation and absorption. Higher temperatures typically mean less dense fuel and moderator, leading to a lower likelihood of neutron absorption. As the temperature increases, the reactor may become less reactive, leading to a decrease in neutron production. Conversely, a decrease in temperature increases the reactivity. The gain is defined according to the benchmark by M. Tiberga et al. [35] as

$$Gain = \frac{(P_{max} - P_{avg})/P_{avg}}{(\gamma_{max} - \gamma_{avg})/\gamma_{avg}} . \quad (6.3)$$

The simulation parameters are found in table 6.2.

Simulation parameters	Units	Value
x^*	[-]	0.04
ν^*	[-]	0.02
Ma	[-]	0.014
Pr	[-]	10
Sc	[-]	10
β_{tot}	[-]	0.006882528
Δt_B	s	$1.28 \cdot 10^{-3}$
Δt_M	s	$1.28 \cdot 10^{-4}$
Δt_S	s	$1.28 \cdot 10^{-6}$
τ_f^*	[-]	0.54
τ_h^*	[-]	0.504
τ_c^*	[-]	0.504

Table 6.2: Simulation parameters for transient simulation

In this phase, the mean generation time Λ and the reactivity ρ are determined at each time-step based on the neutron fluxes and neutronic parameters at that time-step through equations 2.34 and 2.37. It is noted that in this simulation β_{tot} and β_d are considered constant throughout the simulation. Neutronic parameters are affected by the temperature in the core using equations 2.48 and 2.49. Results for the gain and phase shift for the different frequencies can be found in figure 6.2.

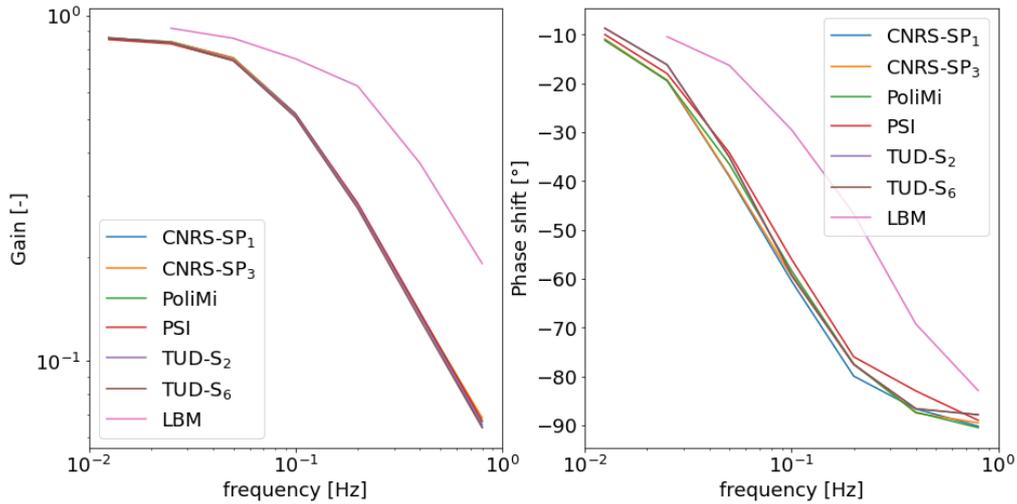


Figure 6.2: Gain and phase-shift as a function of perturbation frequency in the heat transfer coefficient.

The observed parameters from in interval $t = [35 \text{ s}, 40 \text{ s}]$ are displayed in figure 6.3 for a frequency of 0.8 Hz . In that interval, the results are considered to have converged. The results for the other frequencies can be found in appendix E. It can be seen that the implemented perturbation in γ , seen in the bottom graph, does not instantly result in changes in the other parameters. When looking at the graphs along the red vertical line in figure 6.3, the delay in reactivity change and power change are clearly visible.

The entire domain of simulation is shown in figure 6.4, for a frequency of 0.8 Hz . When looking at the initial response to the perturbation, a rapid initial rise in amplitude function and reactor power is observed. It is expected that this is not caused by the initial perturbation but rather by the high initial reactivity, that is in the order of magnitude of 2β . This is considered an extremely high reactivity, and the initial reactivity calculation might include numerical errors. This rapid rise in power is counterbalanced by temperature feedback mechanisms that reduce fission cross-sections and thus fission rates, as can be seen by the rapid drop in reactivity and reactor power. After, the reactivity seems to asymptotically go to fluctuate around a more physical value.

Overall, the gain observed by the LBM tool developed in this research project is observed higher than the gain observed in the $TUD - S_6$ benchmark model. Further, the phase shift observed by the

LBM tool is less negative than the phase shift observed in the $TUD - S_6$ benchmark model. A possible way to explain these discrepancies is the different Pr number implemented in the LBM tool.

The Pr number might play a role in the fluid's responsiveness. Fluids with lower Pr numbers might demonstrate a more rapid response to temperature changes, because heat diffuses more quickly through the fluid, leading to a more uniform temperature distribution and a prompter adjustment to thermal variations. Consequently, in lower Pr number fluids, temperature gradients might adjust more swiftly in response to the reactor's thermal fluctuations, enhancing the reactor's overall thermal regulation. Conversely, fluids with higher Pr numbers exhibit slower thermal response due to the lower thermal diffusivity relative to momentum diffusivity. In such fluids, heat transfer within the convection cells is less efficient, resulting in more pronounced temperature gradients. The slower diffusion of heat means that temperature changes in the reactor core might take longer to propagate through the fluid, potentially leading to less effective thermal regulation. To affirm the validity of this statement, it is essential to simulate the transient response of reactors with varying Pr and, more importantly, to compare the alterations in their temperature profiles. This comparison is critical as the evolving temperature profiles fundamentally underlie the source of reactivity changes in the reactors and thus the reactor's response to the perturbation.

Deviations can also come from numerical errors in determining the PK parameters that may lead to deviations. These errors can arise from approximations in numerical methods or inaccuracies in computational algorithms. Additionally, the assumption that β_d and β_{tot} , representing the fraction of delayed neutrons in family d and the total fraction of the families combined respectively, is constant with time can lead to deviations. In real reactor systems, these parameters might vary due to changes in fuel composition. Lastly, using a small number of lattice nodes in the computational models might also contribute to deviations.

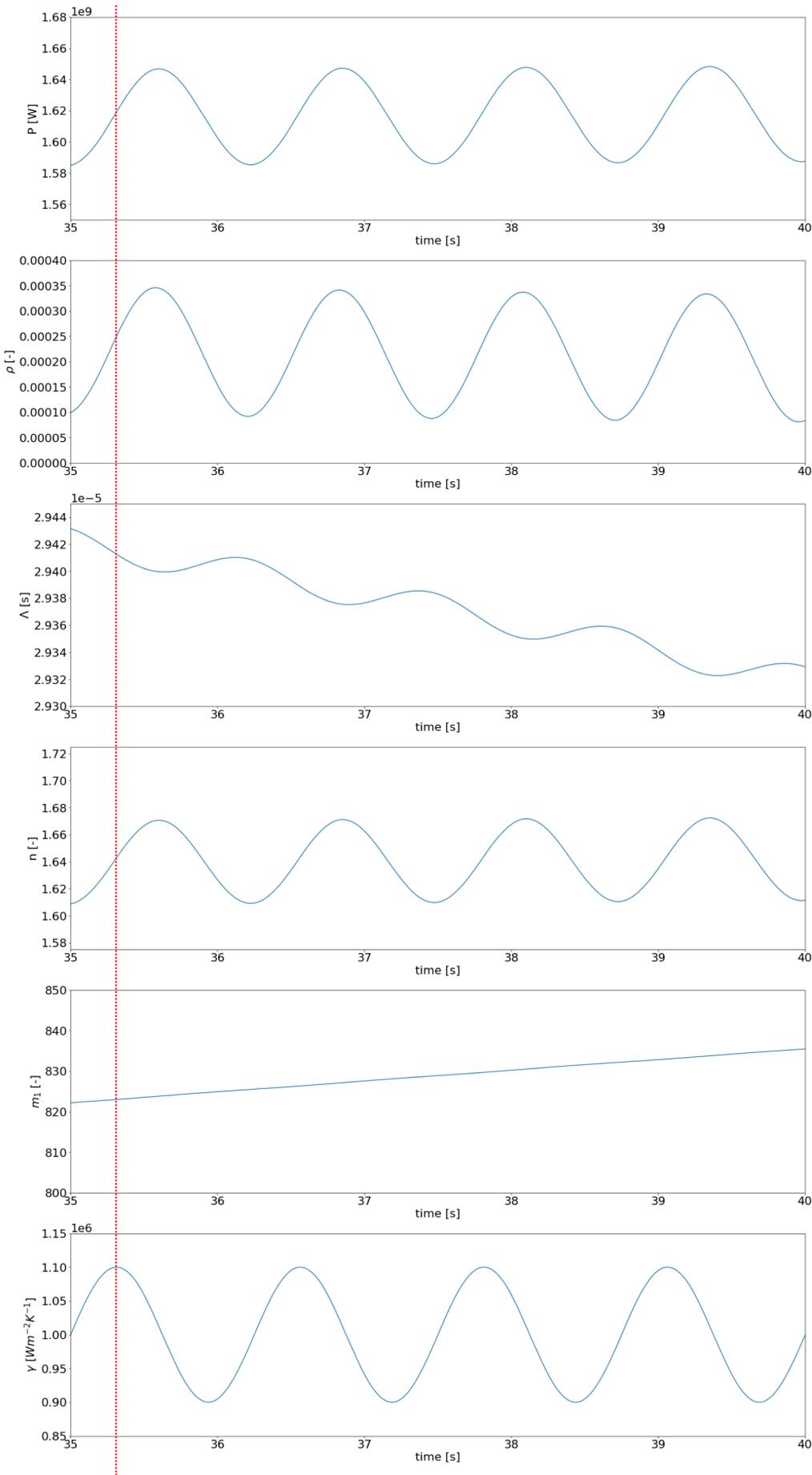


Figure 6.3: Transient results for perturbation of 0.8 Hz from 35 seconds to 40 seconds

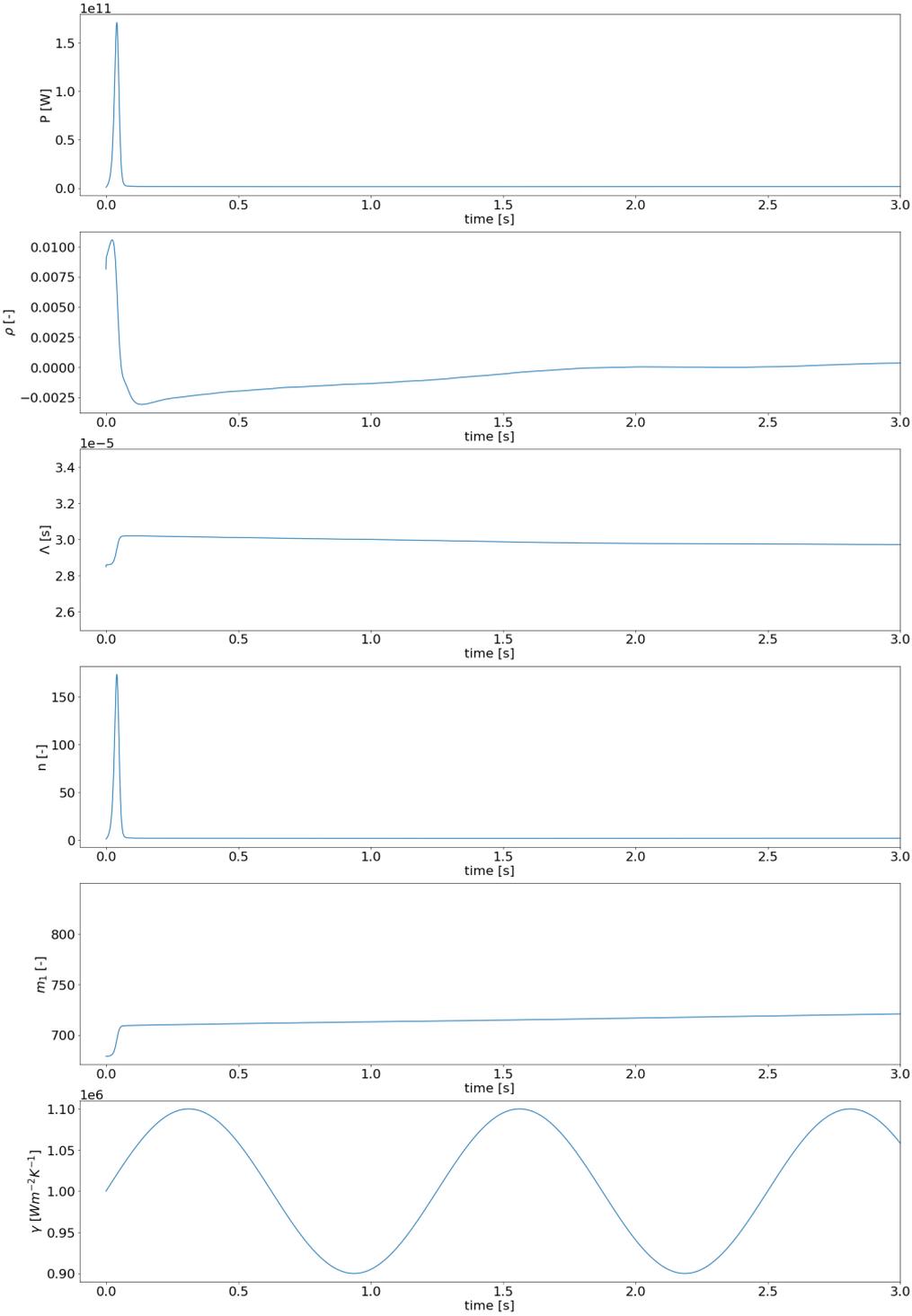


Figure 6.4: Transient results for perturbation of 0.8 Hz from 0 seconds to 10 seconds

7

Conclusion

This research has been directed towards developing tools that can simulate the interaction between thermal-hydraulics and neutronics within MSFRs, a key factor in understanding and predicting their operational behavior. The aim was to develop and validate a tool capable of accurately simulating the interactions between thermal-hydraulic processes and neutron behavior in MSFRs using LBM. For this purpose, the coupling was gradually increased. The capability of simulating the fields and their interactions was assessed using a leading numerical benchmark study by M. Tiberge et al. [35] that compares coupling capabilities of codes using several different numerical methods. Additionally, some simpler benchmarks were used for validation in the uncoupled neutronics phase.

7.1. Steady-state models

All fields in this research are solved in two-dimensional space using the lattice Boltzmann method with the BGK operator. A multiple distribution function approach is implemented, assigning separate distribution functions to different fields. For the velocity simulation, Dirichlet boundary conditions are numerically implemented using the half-way bounce back boundary condition. The results for the uncoupled velocity field is validated by a lid-driven cavity benchmark case, with left bottom and right walls resting and the top wall moving with $0.5ms^{-1}$ [35]. Grid convergence is tested and it is concluded that grid convergence is achieved at 200 grid points per dimension.

For the neutronics simulation, the LBM is used to solve the NDE. The equilibrium boundary condition is used for simulating the vacuum boundaries. For the precursor concentration field, the LBM is used to solve the advection-diffusion equation. Von Neumann boundary conditions are implemented as free-slip boundary conditions. The power method was used to solve the eigenvalue problem for the multiplication factor k_{eff} , and the scalar neutron energy group fluxes ϕ_g . The uncoupled neutronics simulation is tested against several benchmarks, both analytical and numerical.

First the neutronics model was validated against a two-group analytical solution for a homogeneous square reactor core for four different core compositions. Subsequently, the neutronics model was validated against the TWIGL benchmark, a two-group benchmark that also includes one precursor family inside a heterogeneous core [12, 30, 48]. Since diffusion effects were neglected in the benchmark, the collision operator in the LBM for the precursors was omitted. Finally, the uncoupled neutronics model was validated against the six-group and eight-family benchmark [35]. Again, since diffusion effects were neglected in this benchmark phase, the collision operator in the LBM for the precursors was omitted. In this latter, most complex uncoupled benchmark study the effect of implementation of different boundary conditions for the neutronics was assessed. Both the equilibrium boundary condition and the non-equilibrium extrapolation scheme are used for different grid refinements. The discrepancy increases with number of grid points per dimension for the NEES scheme, while it decreases for the EB scheme. It was concluded that the EB scheme is more appropriate for simulating vacuum boundary conditions. Further, it was concluded that the lattice Boltzmann method in combination with the power method can accurately solve the neutronics eigenvalue problem without coupling to thermal-hydraulics.

For the temperature field, the LBM is used to solve the advection-diffusion equation. Von Neumann boundary conditions were implemented as free-slip boundary conditions. Due to the difficulty of imple-

menting high Pr numbers in the LBM, lower Pr numbers are implemented and the effect on accuracy is assessed. Deviations from benchmark codes are highest in regions with highest temperature gradients. At high Pr numbers, increasing Pr further does not significantly lead to increased accuracy. However, increasing Pr does lead to the necessity of using a high number of grid points per dimension, increasing numerical cost. Thus, increasing Pr numbers does not increase accuracy significantly, while rapidly increasing numerical cost.

For the coupled neutronics simulations that include the effect of non-static fuel on the neutronics and precursor fields, the diffusion and convection of precursors plays a role, and the collision operator for the precursors can not be omitted. Due to the difficulty of implementing high Sc numbers in the LBM, lower Sc numbers are implemented and the effect on accuracy is assessed. In a similar reasoning as for the implementation of lower Pr numbers, increasing Sc numbers does not increase accuracy significantly, while rapidly increasing numerical cost. Again, the NEES and EB boundary conditions were implemented and compared on their capabilities. Again, the discrepancy increases with number of grid points per dimension for the NEES scheme, while it decreases for the EB scheme. It was concluded that the EB scheme is more appropriate for simulating vacuum boundary conditions and this scheme was used in the next coupled phases.

For both the simulations for power coupling, i.e. with and without buoyancy effects, the LBM in combination with the power method are capable of accurately simulating the physics in the reactor core, even when implementing lower Sc and Pr than prescribed. As expected from the previous results, deviations are mostly seen at boundaries, where gradients of the precursor and temperature fields are highest. It should be noted that for both simulations the results of the integrated LBM tool does not outperform the other codes for all the fields observed. However, it should also be noted that the integrated LBM tool can still be improved when implementing the prescribed Pr and Sc numbers but this goes at the expense of numerical costs. The steady-state coupled model developed in this research project using the LBM proves to be efficient in modelling the complex physics in MSFRs.

7.2. Transient models

To simulate the transient behaviour in a nuclear reactor core the multi-physics LBM is integrated with the PCQSM. In the PCQSM the steady-state multi-physics LBM is used to solve for the neutron flux shape functions and the Euler method is used to solve for the neutron and delayed precursor amplitude functions. Each time-step, the PCQSM outputs the neutron fluxes after which the precursor concentrations are determined based on these found fluxes using the LBM. Subsequently, the LBM is used to determine the temperature and momentum fields at the time-step, before repeating the process for the next time-step. In the PCQSM, interpolated neutron shape functions are used to determine the point kinetics parameters ρ and Λ .

The transient model that integrates the PCQSM and the LBM is validated against the transient coupled results from the benchmark by M. Tiberga et al. [35]. Although most simulation conditions are similar, a much lower Pr and Sc are implemented to mitigate numerical costs. Further, numerical issues arise in the determination of the initial reactivity, resulting in a steep initial peak in reactivity and thus in reactor power. Nevertheless, the developed tool delivers promising results, accurately capturing the trends in gain and phase shift across varying perturbation frequencies. This achievement underscores its potential, although there remains scope for further refinement in its numerical implementation to enhance its performance.

The transient model represents a pioneering and successful integration of the PCQSM and the LBM for perturbations in coupled reactor simulations, marking a notable first in this field of study.

7.3. Recommendations

- The LBM is highly parallelizable, making it well-suited for implementation on Graphics Processing Units (GPUs). Future research could leverage this advantage to simulate flows with higher Pr and Sc numbers more efficiently. GPU acceleration has the potential to significantly reduce computation time, allowing for more realistic simulations. This could lead to more accurate simulations of thermal and mass transfer processes.
- Although the tool developed in this research already incorporates the complex coupling of physical fields, it does not account for all phenomena observed in reality, such as turbulence, Doppler effects, phase transitions and heat radiation. The Lattice Boltzmann method is capable of mod-

eling turbulence, phase transitions and radiation, but their integration into the current framework has not yet been applied. Future developments could focus on integrating additional physical phenomena into the LBM framework. This would enhance the realism and applicability of the simulations.

- The current approach uses the power method to solve the eigenvalue problem, which may not always be the fastest. Alternative iterative schemes, such as the Chebyshev method, could potentially offer faster convergence. Research could be directed towards comparing different iterative schemes against the in this framework used power method for solving eigenvalue problems. This comparative analysis could lead to more efficient algorithms, potentially reducing computation times for large-scale simulations. Additionally, it is important to dedicate research efforts to exploring the feasibility and effectiveness of integrating the alternative iterative methods (e.g. the Chebyshev method) with the LBM.
- The current research is focused on two-dimensional simulations. Extending this to three dimensions can significantly enhance the realism and applicability of the simulations.
- The current simulation is less effective in regions where temperature and concentration gradients are high. Implementing adaptive grid refinement in LBM simulations particularly in regions exhibiting high temperature or concentration gradients is useful in capturing the nuances of heat transfer and fluid flow. The technique involves dynamically adjusting the grid resolution based on the local thermal or concentration characteristics of the fluid.
- Further research is suggested to investigate the role of the Pr in determining the thermal response of fluids in nuclear reactors. The relevance of this research lies in its potential to enhance understanding of how different Pr fluids affect reactor thermal regulation. Observations of more rapid temperature gradient adjustments in low Pr fluids, versus slower thermal responses in high Pr fluids, could significantly impact simulation results.
- Further research is recommended to improve the numerical method used in determining the PK parameters used in the PCQSM. The current approach treats β_d and β_{tot} as constants over time. However, employing the PCQSM suggests that these parameters should be dynamically solved at certain medium time-steps throughout the simulation. Adopting this methodology could lead to enhanced accuracy in the simulation results. Further, a deeper assessment of the determination of the initial reactivity ρ is suggested, as the current method seems to show numerical inaccuracies in its calculation. It is important to investigate the sources of these inaccuracies to improve the model's reliability.

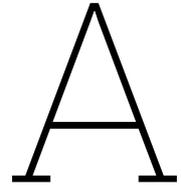
References

- [1] H.E.A. van den Akker and R.F. Mudde. *Fysische Transportverschijnselen: denken in balansen*. 5e druk. CC BY licentie. TU Delft OPEN Publishing, 2023. ISBN: 978-94-6366-680-0. DOI: 10.5074/t.2023.002. URL: <http://resolver.tudelft.nl/uuid:48cbddd8-0575-49b3-aa19-e76ff0ce52b5>.
- [2] M. Aufiero et al. "Development of an OpenFOAM model for the Molten Salt Fast Reactor transient analysis". In: *Chemical Engineering Science* 111 (2014), pp. 390–401. ISSN: 0009-2509. DOI: <https://doi.org/10.1016/j.ces.2014.03.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0009250914001146>.
- [3] H. Bindra and D.V. Patil. "Radiative or neutron transport modeling using a lattice Boltzmann equation framework". In: *Phys. Rev. E* 86 (1 July 2012), p. 016706. DOI: 10.1103/PhysRevE.86.016706. URL: <https://link.aps.org/doi/10.1103/PhysRevE.86.016706>.
- [4] R. D. Blandford and K. S. Thorne. "APPLICATIONS OF CLASSICAL PHYSICS". In: (2013). URL: <https://api.semanticscholar.org/CorpusID:118651591>.
- [5] E. Cervi et al. "Development of a multiphysics model for the study of fuel compressibility effects in the Molten Salt Fast Reactor". In: *Chemical Engineering Science* 193 (2019), pp. 379–393. ISSN: 0009-2509. DOI: <https://doi.org/10.1016/j.ces.2018.09.025>. URL: <https://www.sciencedirect.com/science/article/pii/S0009250918306742>.
- [6] E. Cervi et al. "Multiphysics analysis of the MSFR helium bubbling system: A comparison between neutron diffusion, SP3 neutron transport and Monte Carlo approaches". In: *Annals of Nuclear Energy* 132 (2019), pp. 227–235. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2019.04.029>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454919302130>.
- [7] D. S. Dominguez, C. R.G. Hernandez, and R. C. Barros. "Spectral nodal method for numerically solving two-energy group X,Y geometry neutron diffusion eigenvalue problems". In: *International Journal of Nuclear Energy Science and Technology* 5.1 (2010), pp. 66–79. DOI: 10.1504/IJNEST.2010.030307. eprint: <https://www.inderscienceonline.com/doi/pdf/10.1504/IJNEST.2010.030307>. URL: <https://www.inderscienceonline.com/doi/abs/10.1504/IJNEST.2010.030307>.
- [8] J.J. Duderstadt and L.J. Hamilton. *Nuclear Reactor Analysis*. John Wiley & Sons, Inc., 1976. ISBN: 9788126541218.
- [9] S. Dulla, E.H. Mund, and P. Ravetto. "Accuracy of a Predictor-Corrector Quasi-Static Method for Space-Time Reactor Dynamics". In: *x x* 2006 (Jan. 2006).
- [10] C. Fiorina et al. "GeN-Foam: a novel OpenFOAM® based multi-physics solver for 2D/3D transient analysis of nuclear reactors". In: *Nuclear Engineering and Design* 294 (2015), pp. 24–37. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2015.05.035>. URL: <https://www.sciencedirect.com/science/article/pii/S0029549315003829>.
- [11] C. Fiorina et al. "Modelling and analysis of the MSFR transient behaviour". In: *Annals of Nuclear Energy* 64 (2014), pp. 485–498. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2013.08.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454913004118>.
- [12] L. A. Hageman and J. B. Yasinsky. "Comparison of Alternating-Direction Time-Differencing Methods with Other Implicit Methods for the Solution of the Neutron Group-Diffusion Equations". In: *Nuclear Science and Engineering* 38.1 (1969), pp. 8–32. DOI: 10.13182/NSE38-8. eprint: <https://doi.org/10.13182/NSE38-8>. URL: <https://doi.org/10.13182/NSE38-8>.

- [13] P. 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. eprint: <https://doi.org/10.13182/NT8-2-118>. URL: <https://doi.org/10.13182/NT8-2-118>.
- [14] X. He and L. Luo. "Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation". In: *PHYSICAL REVIEW E* 56 (Dec. 1997), pp. 6811–6817. DOI: 10.1103/PhysRevE.56.6811.
- [15] J. Huang et al. "A general approach of unit conversion system in lattice Boltzmann method and applications for convective heat transfer in tube banks". In: *International Journal of Heat and Mass Transfer* 135 (2019), pp. 873–884. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2019.02.014>. URL: <https://www.sciencedirect.com/science/article/pii/S001793101834835X>.
- [16] J. L. Kloosterman. "Safety assessment of the molten salt fast reactor (SAMOFAR)". In: *Molten Salt Reactors and Thorium Energy*. Ed. by T. J. Dolan. Woodhead Publishing, 2017, pp. 565–570. ISBN: 978-0-08-101126-3. DOI: <https://doi.org/10.1016/B978-0-08-101126-3.00020-8>. URL: <https://www.sciencedirect.com/science/article/pii/B9780081011263000208>.
- [17] J. L. Kloosterman. "Simulation Challenges of the Molten Salt Fast reactor". Presentation at TU Delft, Netherlands on behalf of the SAMOFAR consortium. URL: https://www.janleenkloosterman.nl/reports/technet_20171007.pdf.
- [18] Q. Kong, T. Siau, and A. Bayen. *Python Programming and Numerical Methods: A Guide for Engineers and Scientists*. Elsevier Science, 2020. ISBN: 9780128195505. URL: <https://books.google.nl/books?id=GWzqDwAAQBAJ>.
- [19] A. Koning et al. *The JEFF-3.1 nuclear data library*. Jan. 2006.
- [20] G. Kooreman and D. Griesheimer. "PREDICTOR-CORRECTOR QUASI-STATIC METHOD FOR TIGHTLY-COUPLED REACTOR MULTIPHYSICS TRANSIENT CALCULATIONS". In: *EPJ Web of Conferences* 247 (Jan. 2021), p. 07011. DOI: 10.1051/epjconf/202124707011.
- [21] T. Krueger et al. *The Lattice Boltzmann Method: Principles and Practice*. English. Graduate Texts in Physics. Springer, 2016. ISBN: 978-3-319-44647-9.
- [22] G. Lapenta, F. Mattioda, and P. Ravetto. "Point kinetic model for fluid fuel systems". In: *Annals of Nuclear Energy* 28.17 (2001), pp. 1759–1772. ISSN: 0306-4549. DOI: [https://doi.org/10.1016/S0306-4549\(01\)00012-3](https://doi.org/10.1016/S0306-4549(01)00012-3). URL: <https://www.sciencedirect.com/science/article/pii/S0306454901000123>.
- [23] J. Leppänen et al. "The Serpent Monte Carlo code: Status, development and applications in 2013". In: *Annals of Nuclear Energy* 82 (2015). Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms, pp. 142–150. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2014.08.024>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454914004095>.
- [24] J. Li. *Multiscale and Multiphysics Flow Simulations of Using the Boltzmann Equation: Applications to Porous Media and MEMS*. English. Publisher Copyright: © Springer Nature Switzerland AG 2020. Springer International Publishing, Jan. 2019. ISBN: 9783030264659. DOI: 10.1007/978-3-030-26466-6.
- [25] L. Li, R. Mei, and J. F. Klausner. "Lattice Boltzmann models for the convection-diffusion equation: D2Q5 vs D2Q9". In: *International Journal of Heat and Mass Transfer* 108 (2017), pp. 41–62. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2016.11.092>. URL: <https://www.sciencedirect.com/science/article/pii/S0017931016326047>.
- [26] S. Marguet. *The Physics of Nuclear Reactors*. Springer International Publishing, 2018. ISBN: 9783319595603. URL: <https://books.google.nl/books?id=9Dl0DwAAQBAJ>.
- [27] A. Mohamad and S. Succi. "A note on equilibrium boundary conditions in lattice Boltzmann fluid dynamic simulations". In: *European Physical Journal: Special Topics* 171 (Apr. 2009), pp. 213–221. DOI: 10.1140/epjst/e2009-01031-9.

- [28] *Nuclear Power*. URL: <https://www.iaea.org/energy-system/electricity/nuclear-power> (visited on 01/19/2024).
- [29] K.O. Ott and R.J. Neuhold. *Introductory Nuclear Reactor Dynamics*. Purdue University, 1976. ISBN: 9780894480294.
- [30] Z. Prince, J. Ragusa, and Y. Wang. "IMPLEMENTATION OF THE IMPROVED QUASI-STATIC METHOD IN RATTLESNAKE/MOOSE FOR TIME-DEPENDENT RADIATION TRANSPORT MODELING". In: May 2016.
- [31] D. Rozon. *Introduction to Nuclear Reactor Kinetics*. Trans. by B. Rouben. Translated by Benjamin Rouben. Éditions de l'École Polytechnique de Montréal.
- [32] SAMOSA FER. *Project*. URL: <https://samosafer.eu/project/> (visited on 01/19/2024).
- [33] M. Tiberger. "Development of a high-fidelity multi-physics simulation tool for liquid-fuel fast nuclear reactors". Doctoral Thesis. Delft, The Netherlands: Delft University of Technology, Aug. 2020. ISBN: 978-94-028-2142-0. DOI: 10.4233/uuid:f59f8f98-ebdd-4266-99ab-e70e8e6433ee. URL: <https://doi.org/10.4233/uuid:f59f8f98-ebdd-4266-99ab-e70e8e6433ee>.
- [34] M. Tiberger, D. Lathouwers, and J.L. Kloosterman. "A Discontinuous Galerkin FEM Multi-Physics Solver for the Molten Salt Fast Reactor". In: *Proceedings of the 2019 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2019)*. Ed. by J. Wagner and J. Rathkopf. International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2019. Green Open Access added to TU Delft Institutional Repository 'You share, we take care!' – Taverne project. TU Delft RST/Reactor Physics and Nuclear Materials. Portland, United States: American Nuclear Society, Aug. 25–29, 2019, specific pages if available. ISBN: 978-089448769-9. URL: <http://resolver.tudelft.nl/uuid:d955be49-dfb7-476d-a30a-d008c560752f>.
- [35] M. Tiberger 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.
- [36] D. J. Tritton. "Thermal Flows: Basic Equations and Concepts". In: *Physical Fluid Dynamics*. Dordrecht: Springer Netherlands, 1977, pp. 127–134. ISBN: 978-94-009-9992-3. DOI: 10.1007/978-94-009-9992-3_13. URL: https://doi.org/10.1007/978-94-009-9992-3_13.
- [37] R.G.M. van der Sman. "Finite Boltzmann schemes". In: *Computers & Fluids* 35.8 (2006). Proceedings of the First International Conference for Mesoscopic Methods in Engineering and Science, pp. 849–854. ISSN: 0045-7930. DOI: <https://doi.org/10.1016/j.compfluid.2005.07.019>. URL: <https://www.sciencedirect.com/science/article/pii/S0045793005001465>.
- [38] E.L. Wachspress. *Iterative Solution of Elliptic Systems: And Applications to the Neutron Diffusion Equations of Reactor Physics*. Prentice-Hall international series in applied mathematics. Prentice-Hall, 1966. URL: <https://books.google.nl/books?id=9e9QAAAAMAAJ>.
- [39] A.E. Waltar, D.R. Todd, and P. Tsvetkov. *Fast Spectrum Reactors*. Oct. 2012, pp. 1–720. ISBN: 978-1-4419-9571-1. DOI: 10.1007/978-1-4419-9572-8.
- [40] Y. Wang and Y. Ma. "IbmNTH: A unified lattice Boltzmann framework for coupled neutronics-thermal-hydraulics analysis". In: *Annals of Nuclear Energy* 166 (2022), p. 108750. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2021.108750>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454921006265>.
- [41] Y. Wang, Y. Ma, and M. Xie. "Analysis of the multi-physics approach using the unified lattice Boltzmann framework". In: *Annals of Nuclear Energy* 143 (2020), p. 107500. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2020.107500>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454920301985>.
- [42] Y. Wang, Y. Ma, and M. Xie. "GPU accelerated lattice Boltzmann method in neutron kinetics problems". In: *Annals of Nuclear Energy* 129 (2019), pp. 350–365. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2019.02.009>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454919300763>.

- [43] Y. Wang, Y. Ma, and M. Xie. "GPU accelerated lattice Boltzmann method in neutron kinetics problems II: Neutron transport calculation". In: *Annals of Nuclear Energy* 134 (2019), pp. 305–317. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2019.06.020>. URL: <https://www.sciencedirect.com/science/article/pii/S030645491930338X>.
- [44] Y. Wang, Y. Ma, and M. Xie. "High-order lattice Boltzmann method for multi-group neutron diffusion solution". In: *Progress in Nuclear Energy* 110 (2019), pp. 341–353. ISSN: 0149-1970. DOI: <https://doi.org/10.1016/j.pnucene.2018.10.014>. URL: <https://www.sciencedirect.com/science/article/pii/S014919701830266X>.
- [45] Y. Wang, Y. Ma, and M. Xie. "Neutron transport solution of lattice Boltzmann method and streaming-based block-structured adaptive mesh refinement". In: *Annals of Nuclear Energy* 118 (2018), pp. 249–259. ISSN: 0306-4549. DOI: <https://doi.org/10.1016/j.anucene.2018.04.013>. URL: <https://www.sciencedirect.com/science/article/pii/S0306454918301920>.
- [46] Y. Wang, L. Yan, and Y. Ma. "Lattice Boltzmann solution of the transient Boltzmann transport equation in radiative and neutron transport". In: *Phys. Rev. E* 95 (6 June 2017), p. 063313. DOI: [10.1103/PhysRevE.95.063313](https://doi.org/10.1103/PhysRevE.95.063313). URL: <https://link.aps.org/doi/10.1103/PhysRevE.95.063313>.
- [47] S. Zhang. "Mixed convective heat transfer of medium-Prandtl-number fluids in horizontal circular tubes". In: *International Journal of Heat and Mass Transfer* 190 (2022), p. 122740. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2022.122740>. URL: <https://www.sciencedirect.com/science/article/pii/S0017931022002228>.
- [48] S. Zhang et al. "A Nuclear Reactor Transient Methodology Based on Discrete Ordinates Method". In: *Science and Technology of Nuclear Installations* 2014 (Dec. 2014). DOI: [10.1155/2014/491048](https://doi.org/10.1155/2014/491048).



Iteration schemes steady-state
uncoupled neutronics simulations

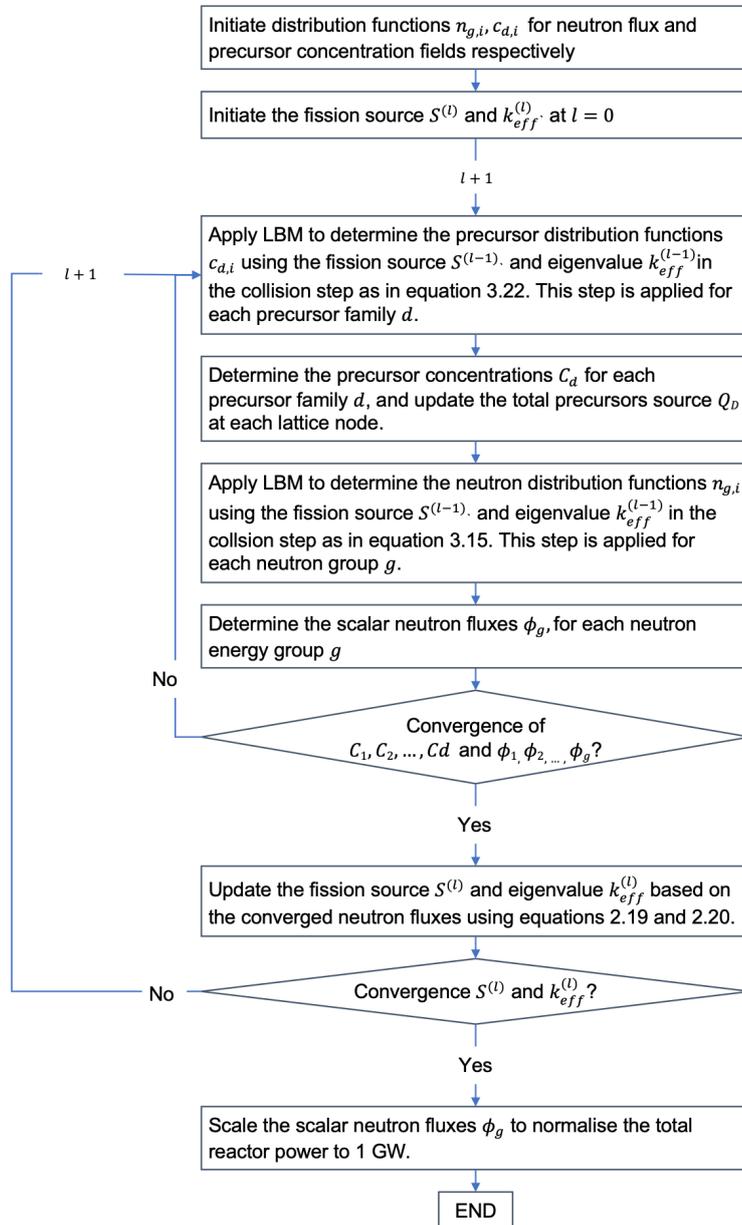


Figure A.1: Iteration scheme for phase 0.2

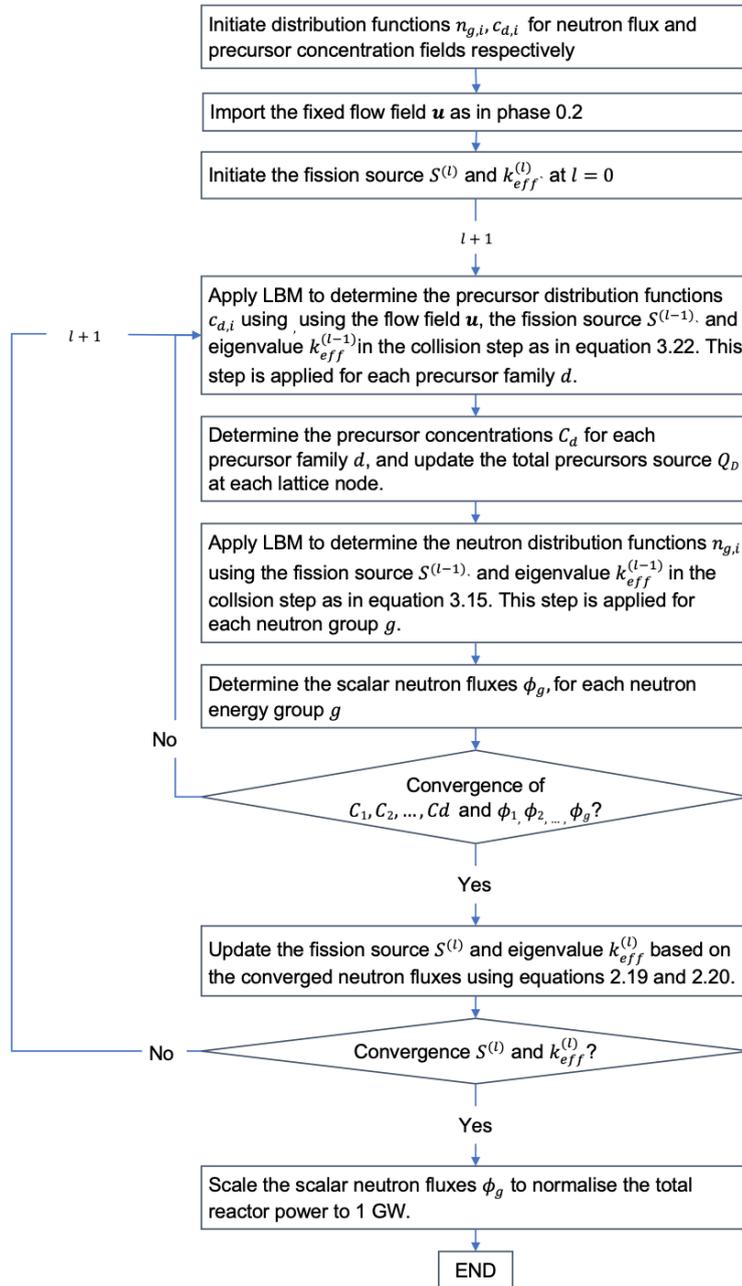


Figure A.2: Iteration scheme for phase 1.1

B

Neutronics and precursor data

The data in this appendix is used in the simulations with 6 neutron energy groups and 8 precursor families, and is retrieved from the benchmark by Tiberge et al. [35].

B.1. Neutronics data

Group, g	$\Sigma_{f,g} [m^{-1}]$	$\Sigma_{t,g} [m^{-1}]$	$D_g [m]$	$v_g [m/s]$
1	0.111309	16.5512	0.0280064	$2.49771 \cdot 10^7$
2	0.108682	21.7253	0.0184021	$1.35163 \cdot 10^7$
3	0.152219	31.8009	0.011311	$3.82047 \cdot 10^6$
4	0.25819	24.2093	0.0144786	$1.49417 \cdot 10^6$
5	0.536326	25.0351	0.013975	$6.41663 \cdot 10^5$
6	1.44917	27.2079	0.0128252	$2.35592 \cdot 10^4$

Table B.1: Fission cross-sections, total (removal) cross-sections, diffusion constants and velocities for the 6 neutron energy groups as retrieved from Tiberge et al. [35].

$\Sigma_{s,g' \rightarrow g}$ [m^{-1}]	g'	g					
		1	2	3	4	5	6
1	1	$1.08476 \cdot 10^1$	0	0	0	0	0
2	1	$5.23318 \cdot 10^0$	$1.83666 \cdot 10^1$	0	0	0	0
3	1	$4.01805 \cdot 10^{-1}$	$3.19138 \cdot 10^0$	$2.98293 \cdot 10^1$	0	0	0
4	1	$1.09869 \cdot 10^{-2}$	$2.34218 \cdot 10^{-3}$	$1.6347 \cdot 10^0$	$2.17472 \cdot 10^1$	0	0
5	1	$2.5329 \cdot 10^{-3}$	$2.25259 \cdot 10^{-4}$	$1.70575 \cdot 10^{-3}$	$1.90243 \cdot 10^0$	$2.27173 \cdot 10^1$	0
6	1	$3.78334 \cdot 10^{-4}$	$2.00405 \cdot 10^{-5}$	$1.24625 \cdot 10^{-4}$	$1.36858 \cdot 10^{-6}$	$1.05885 \cdot 10^0$	$2.37826 \cdot 10^1$

Table B.2: P_0 scattering cross-sections for the 6 neutron energy groups as originating group and destination group as retrieved from Tiberge et al. [35].

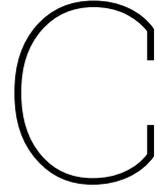
Group, g	$\nu_{tot,g} [-]$	$\chi_g^p [-]$	$\chi_g^d [-]$	$E_{fiss} [J]$
1	2.85517	0.353812	0.00430325	$3.240722 \cdot 10^{11}$
2	2.54532	0.523642	0.387734	$3.240722 \cdot 10^{11}$
3	2.43328	0.121033	0.581848	$3.240722 \cdot 10^{11}$
4	2.43127	0.00135457	0.0227947	$3.240722 \cdot 10^{11}$
5	2.4333	0.000151226	0.0028913	$3.240722 \cdot 10^{11}$
6	2.4333	$7.37236e - 06$	0.000428935	$3.240722 \cdot 10^{11}$

Table B.3: Average number of neutrons emitted per fission event, prompt neutron spectrum, delayed neutron spectrum and average energy emitted per fission event for the 6 neutron energy groups as retrieved from Tiberge et al. [35].

B.2. Precursor parameters

Family, d	$\lambda_d [s^{-1}]$	$\beta_d [-]$
1	0.0124667	0.000233102
2	0.0282917	0.00103262
3	0.0425244	0.000681878
4	0.133042	0.00137726
5	0.292467	0.00214493
6	0.666488	0.000640917
7	1.63478	0.000605805
8	3.5546	0.000166016

Table B.4: Decay constant and delayed neutron fraction for the 8 delayed precursor families as retrieved from Tiberga et al. [35].



Derivation of non-dimensional governing equations for concentration and temperature

C.1. Non-dimensional heat equation

One defines the dimensionless quantities

$$\tilde{x} = \frac{x}{L} \quad \tilde{y} = \frac{y}{L} \quad \tilde{T} = \frac{T}{T_0} \quad \tilde{\mathbf{u}} = \frac{\mathbf{u}}{v} , \quad (\text{C.1})$$

with L the characteristic length, T_0 the characteristic temperature and v the characteristic magnitude of velocity. The dimensionless nabla-operator is defined as

$$\tilde{\nabla} = \begin{bmatrix} \frac{\partial}{\partial \tilde{x}} \\ \frac{\partial}{\partial \tilde{y}} \end{bmatrix} . \quad (\text{C.2})$$

The relevant non-dimensional numbers are

$$Re = \frac{vL}{\nu} \quad \& \quad Pr = \frac{\nu}{\alpha} \quad \& \quad Nu = \frac{\gamma L^2}{\lambda} . \quad (\text{C.3})$$

Starting from the steady-state heat equation without the fission heat source,

$$\mathbf{u} \cdot \nabla T = -\alpha \nabla^2 T + \gamma(T_{ext} - T) , \quad (\text{C.4})$$

substituting the dimensionless quantities and operator above results in

$$\frac{\rho c_p v T_0}{L} [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{T}] = -\frac{\lambda T_0}{L^2} \tilde{\nabla}^2 \tilde{T} + \gamma T_0 (1 - \tilde{T}) . \quad (\text{C.5})$$

Multiplying with L^2 , dividing by ν and T_0 and substituting $\alpha = \frac{\lambda}{\rho c_p}$ results in

$$\frac{vL}{\nu} [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{T}] = -\frac{\alpha}{\nu} \tilde{\nabla}^2 \tilde{T} + \frac{\alpha}{\nu} \frac{\gamma L^2}{\lambda} (1 - \tilde{T}) . \quad (\text{C.6})$$

Recognising the non-dimensional numbers Re , Pr and Nu and after some rewriting one obtains

$$Pr Re [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{T}] = -\tilde{\nabla}^2 \tilde{T} + Nu (1 - \tilde{T}) . \quad (\text{C.7})$$

The Nu number is measure for the ratio between convective and conductive heat transfer. In the case of free convection in laminar flow conditions it is a function of the Grashof number Gr and the Prandtl number Pr in the form of

$$Nu = K (Gr Pr)^{1/4} \quad (\text{C.8})$$

with K a constant, in literature given as 0.52 [1] for $10^4 < Gr Pr < 10^8$ resulting in

$$Pr Re [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{T}] = -\tilde{\nabla}^2 \tilde{T} + K (Gr Pr)^{1/4} (1 - \tilde{T}) . \quad (\text{C.9})$$

C.2. Non-dimensional concentration advection-diffusion equation

One defines the dimensionless quantities

$$\tilde{x} = \frac{x}{L} \quad \tilde{y} = \frac{y}{L} \quad \tilde{C} = \frac{C}{C_0} \quad \tilde{\mathbf{u}} = \frac{\mathbf{u}}{v} , \quad (\text{C.10})$$

with L the characteristic length, C_0 the characteristic concentration and v the characteristic magnitude of velocity. The dimensionless nabla-operator is defined as

$$\tilde{\nabla} = \left[\begin{array}{c} \frac{\partial}{\partial \tilde{x}} \\ \frac{\partial}{\partial \tilde{y}} \end{array} \right] . \quad (\text{C.11})$$

The relevant non-dimensional numbers are

$$Re = \frac{vL}{\nu} \quad \& \quad Sc = \frac{D_p}{\alpha} \quad \& \quad Da = \frac{\lambda_d L^2}{D_p} . \quad (\text{C.12})$$

Starting from the steady-state precursor concentration equation without the fission source,

$$\nabla \cdot (\mathbf{u}C_d) = \nabla \cdot (D_p \nabla C_d) - \lambda_d C_d . \quad (\text{C.13})$$

substituting the dimensionless quantities and operator above results in

$$\frac{vC_0}{L} [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{C}] = \frac{D_p C_0}{L^2} \tilde{\nabla}^2 \tilde{C} - \lambda_d C_0 \tilde{C} . \quad (\text{C.14})$$

Multiplying with L^2 and dividing by ν and C_0 results in

$$\frac{vL}{\nu} [\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{C}] = \frac{D_p}{\nu} \tilde{\nabla}^2 \tilde{C} - \frac{D_p}{\nu} \frac{\lambda_d L^2}{D_p} \tilde{C} . \quad (\text{C.15})$$

Recognising the non-dimensional numbers Re , Pr and Da and after some rewriting one obtains

$$ScRe[\tilde{\mathbf{u}} \cdot \tilde{\nabla} \tilde{C}] = \tilde{\nabla}^2 \tilde{C} - Da \tilde{C} . \quad (\text{C.16})$$

The Da number is used in reaction engineering to relate the timescale of reactions to the timescale of transport processes, in this case the timescale of diffusion. It is defined as the ratio of the characteristic time of reaction to the characteristic time of transport.

D

Results for uncoupled neutronics simulations of chapter 4

D.1. Results for the neutron fluxes in a homogeneous solid fuel reactor core

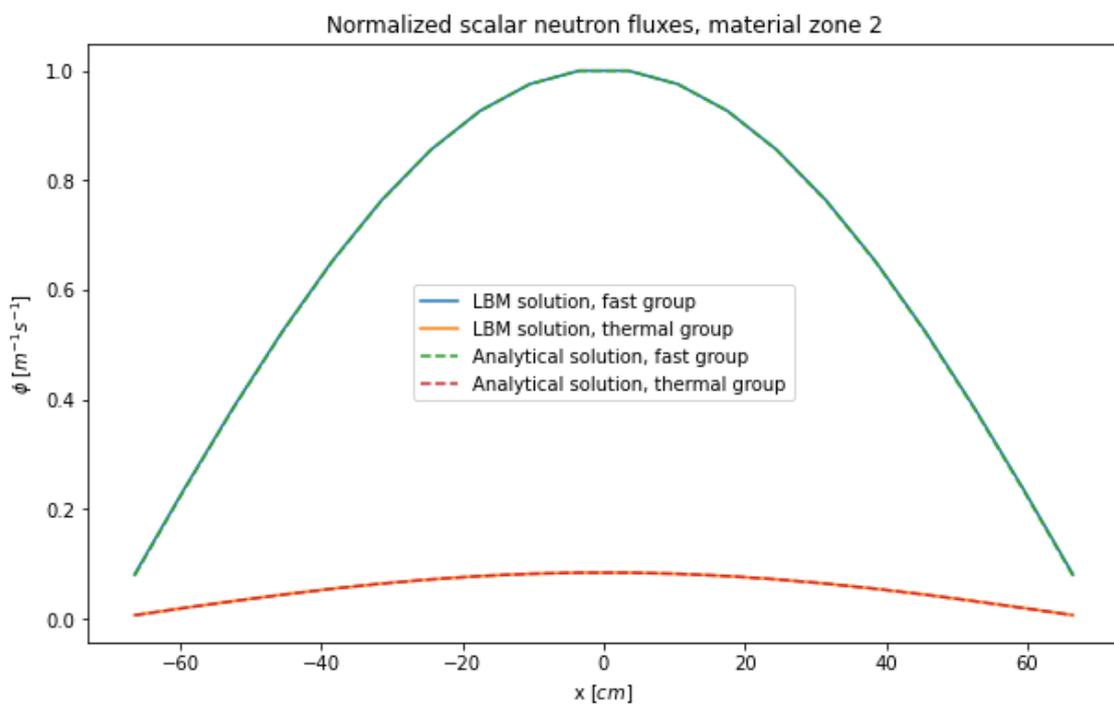


Figure D.1: Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 2.

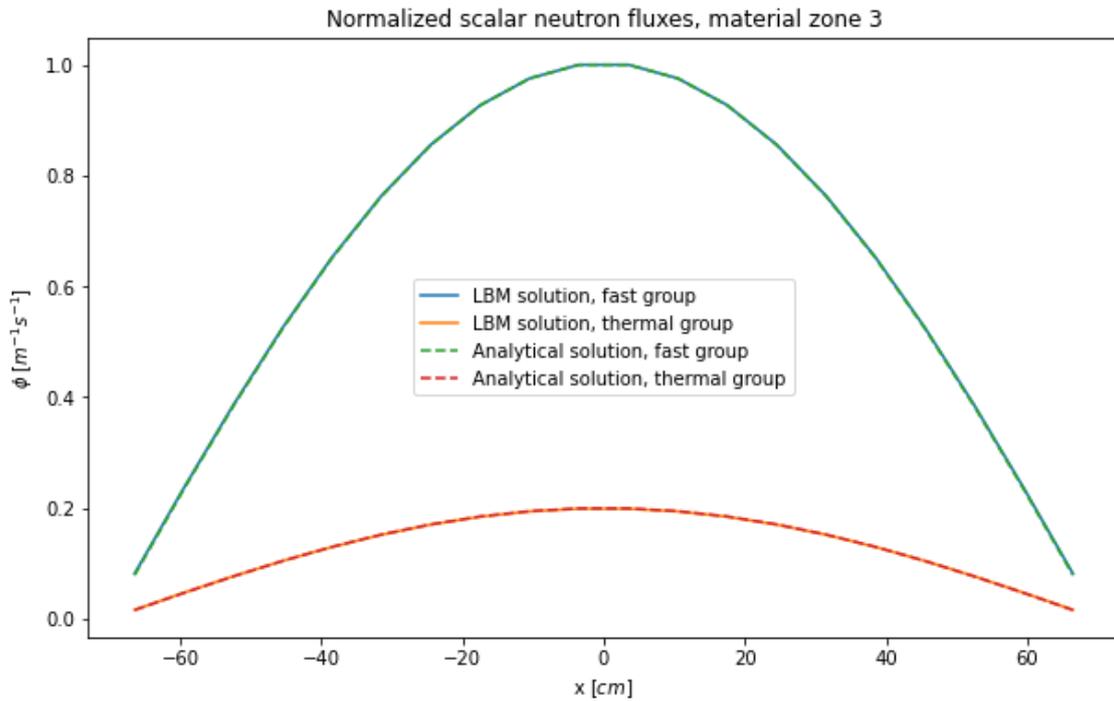


Figure D.2: Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 3.

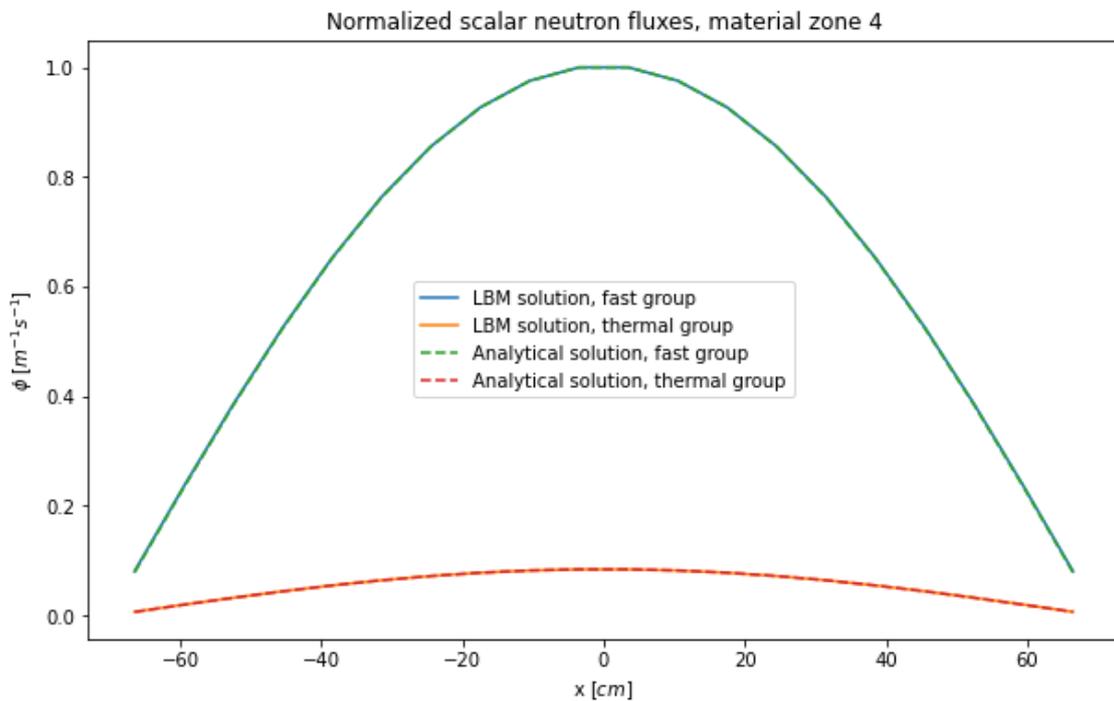
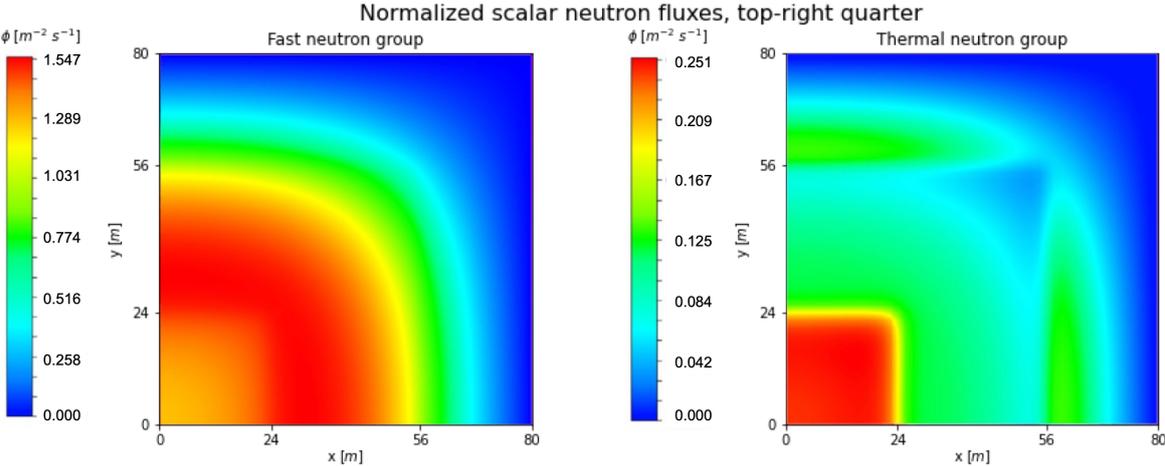


Figure D.3: Analytical and LBM solution for two-energy group simulation in homogeneous square cavity with bare boundaries for material zone 4.

D.2. Results for the benchmark flux field in the TWIGL reactor

These results are obtained from [30]



E

Results for the fully coupled transient simulations

E.1. Results for the transient simulations in first 3 seconds after perturbation

In this section the results for the transient situation are displayed in the first 3 seconds after the initiation of the heat transfer coefficient perturbation.

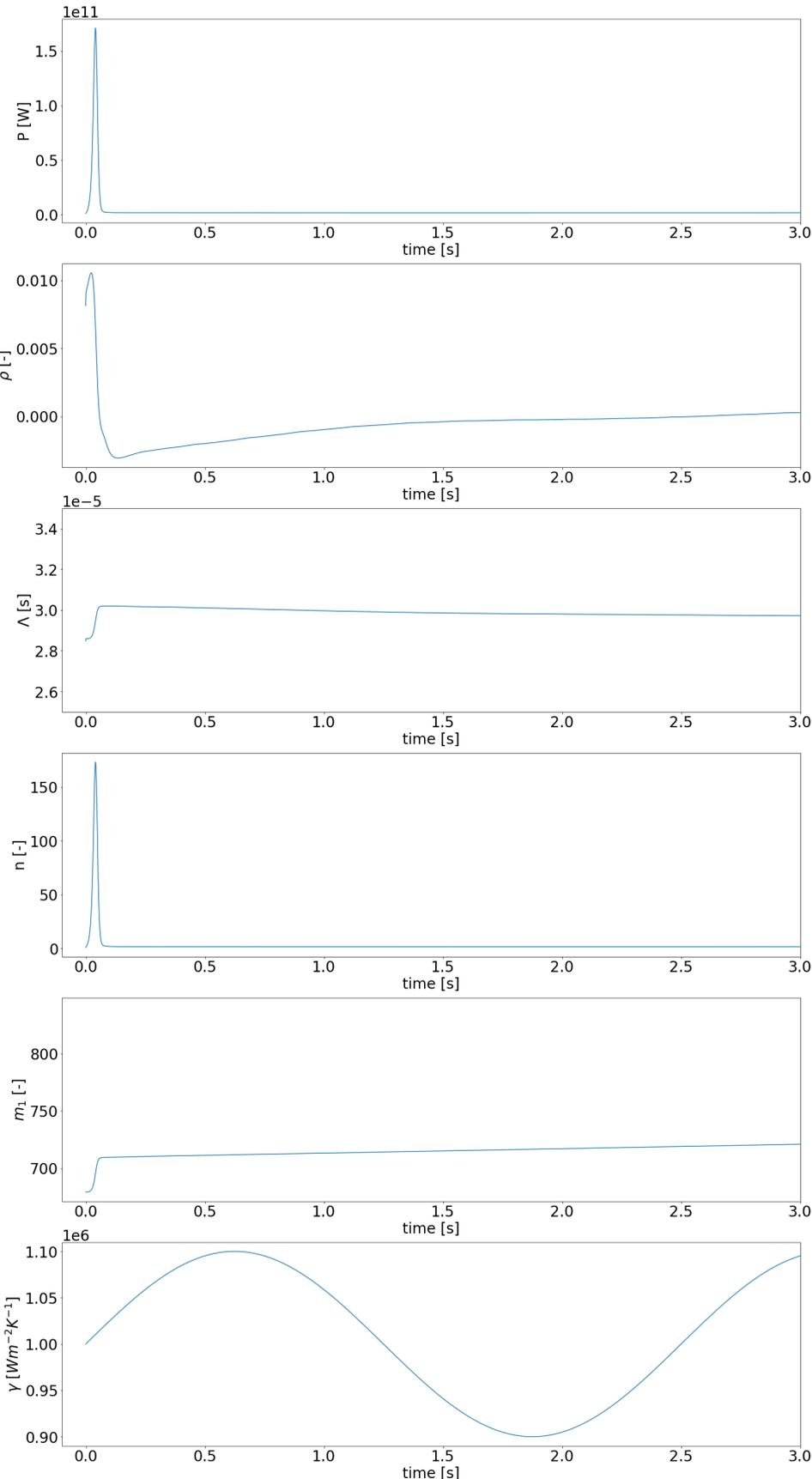


Figure E.2: Transient results for perturbed heat coefficient with $f = 0.4Hz$, for first 3 seconds after initial perturbation

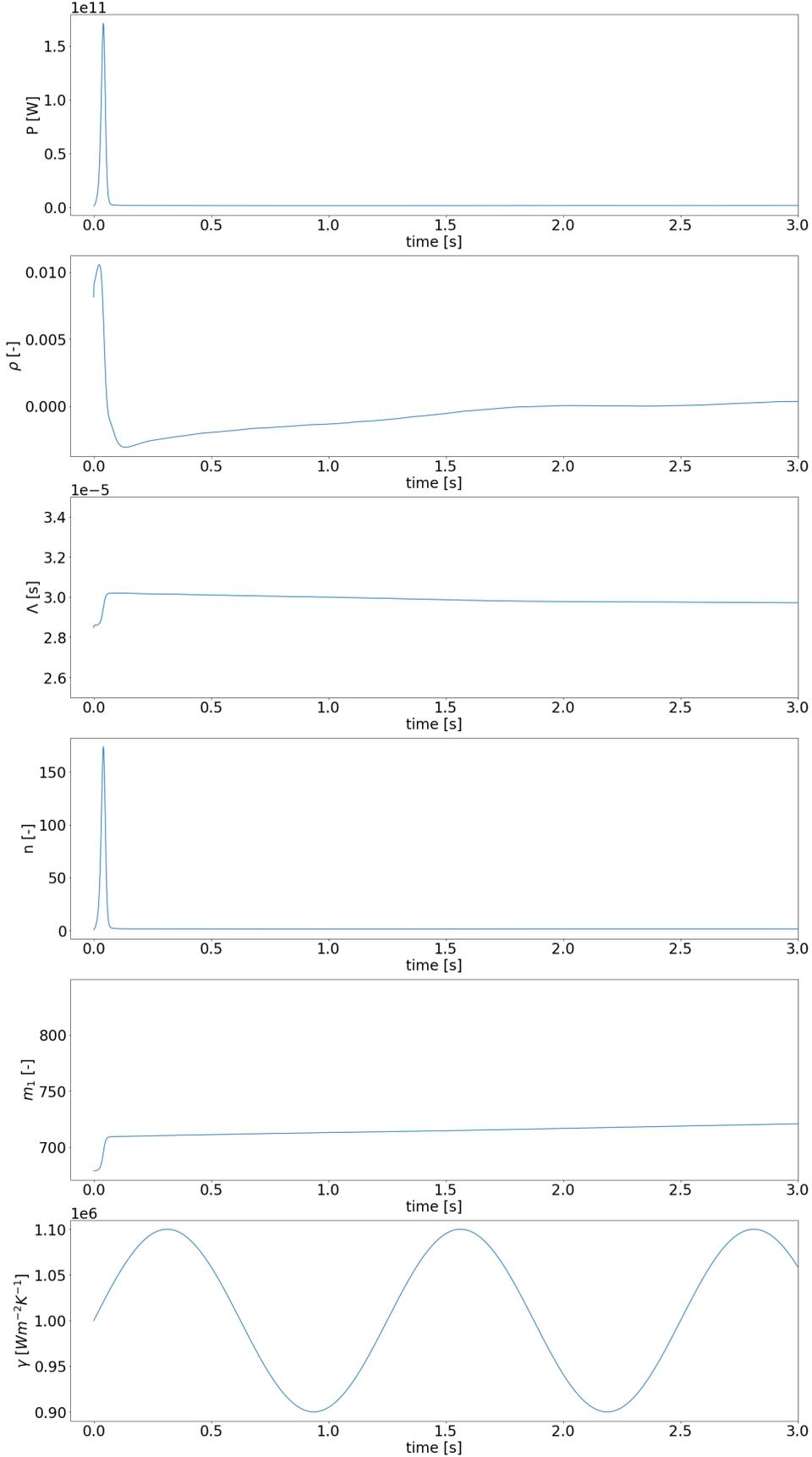


Figure E.1: Transient results for perturbed heat coefficient with $f = 0.8Hz$, for first 3 seconds after initial perturbation

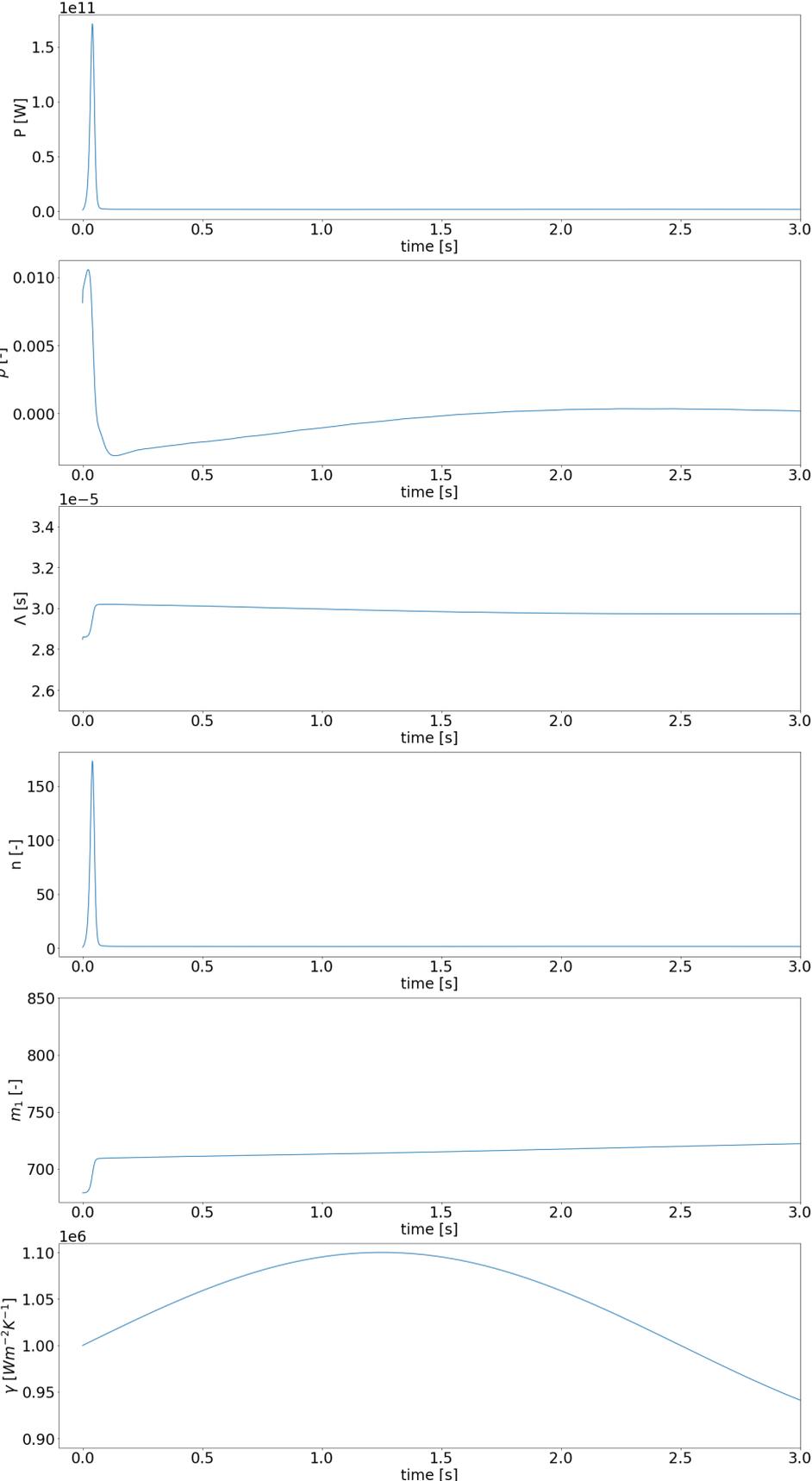


Figure E.3: Transient results for perturbed heat coefficient with $f = 0.2Hz$, for first 3 seconds after initial perturbation

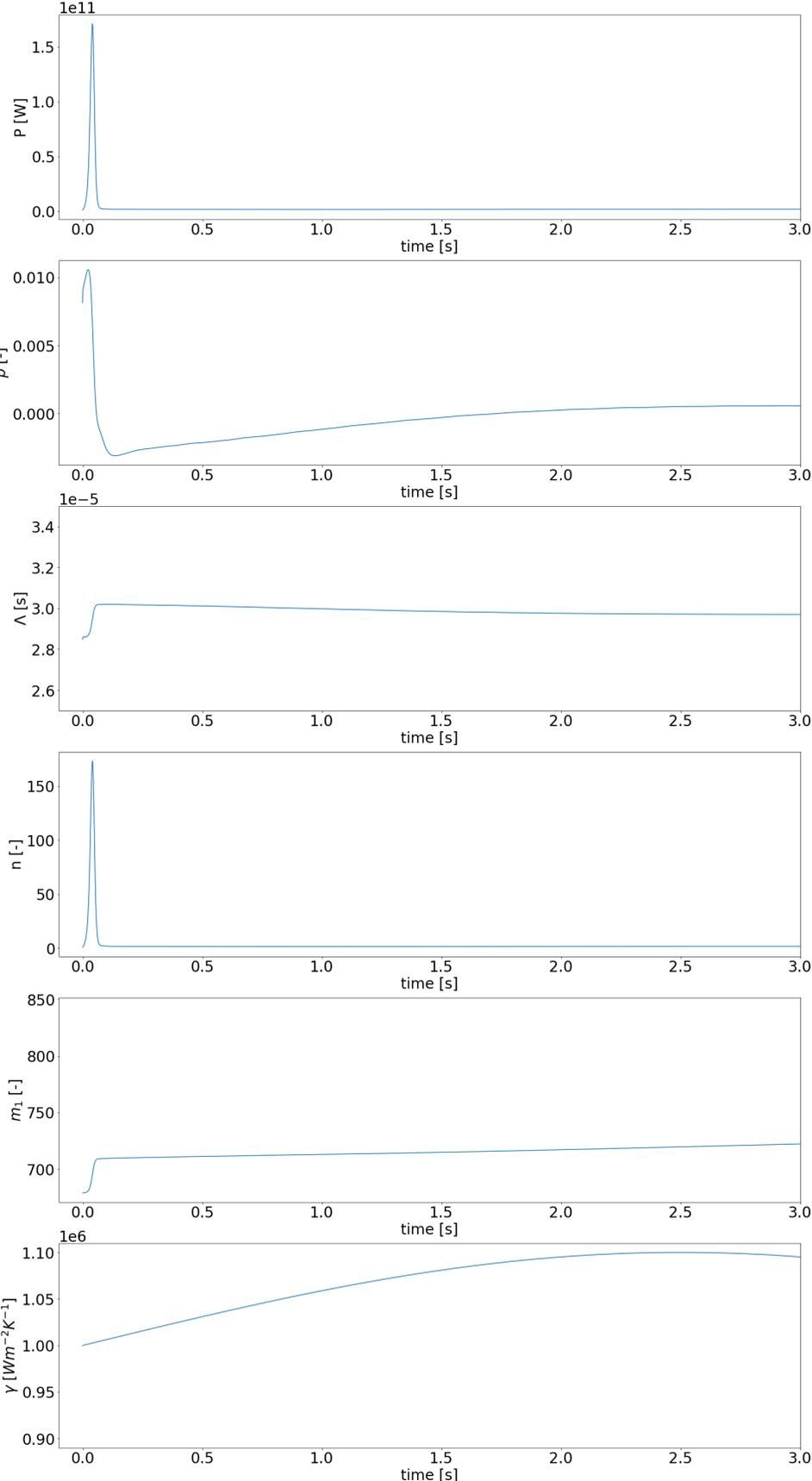


Figure E.4: Transient results for perturbed heat coefficient with $f = 0.1Hz$, for first 3 seconds after initial perturbation

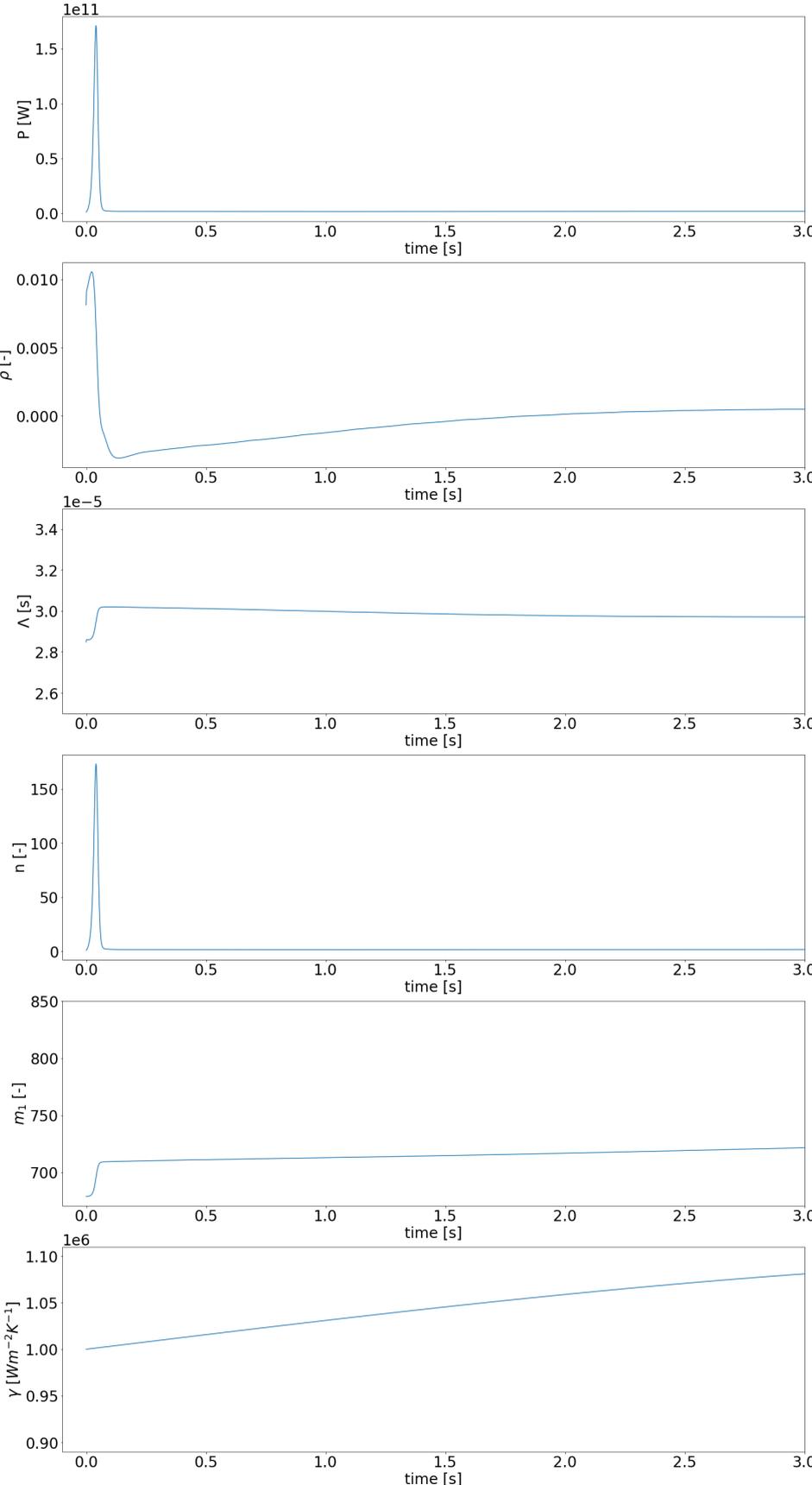


Figure E.5: Transient results for perturbed heat coefficient with $f = 0.05Hz$, for first 3 seconds after initial perturbation

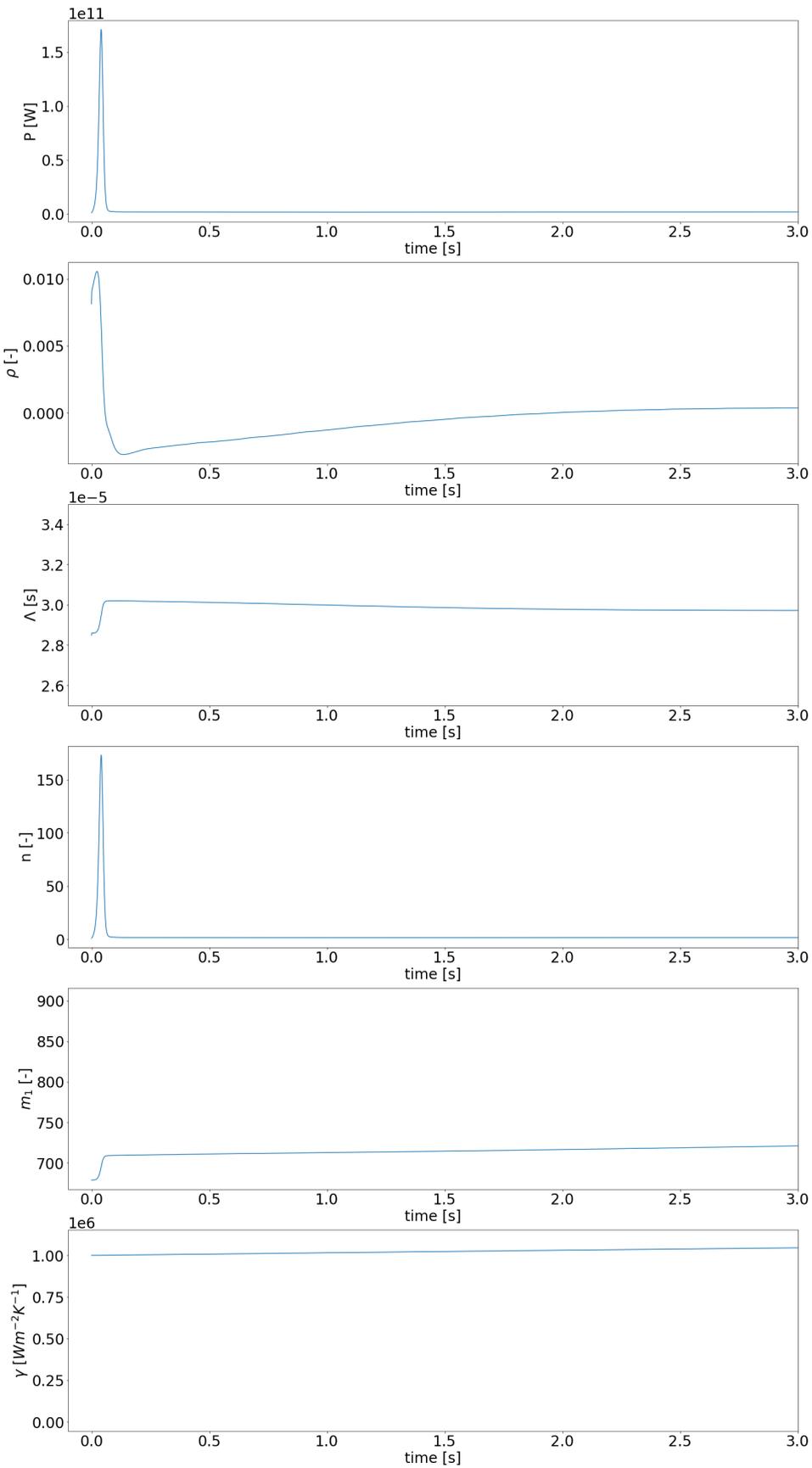


Figure E.6: Transient results for perturbed heat coefficient with $f = 0.025Hz$, for first 3 seconds after initial perturbation

E.2. Results of transient simulations displaying outcomes after power jump stabilization period

In this section the results for the transient situation are displayed after the results have converged. If the simulation terminated before convergence the last time interval with at least one period is displayed.

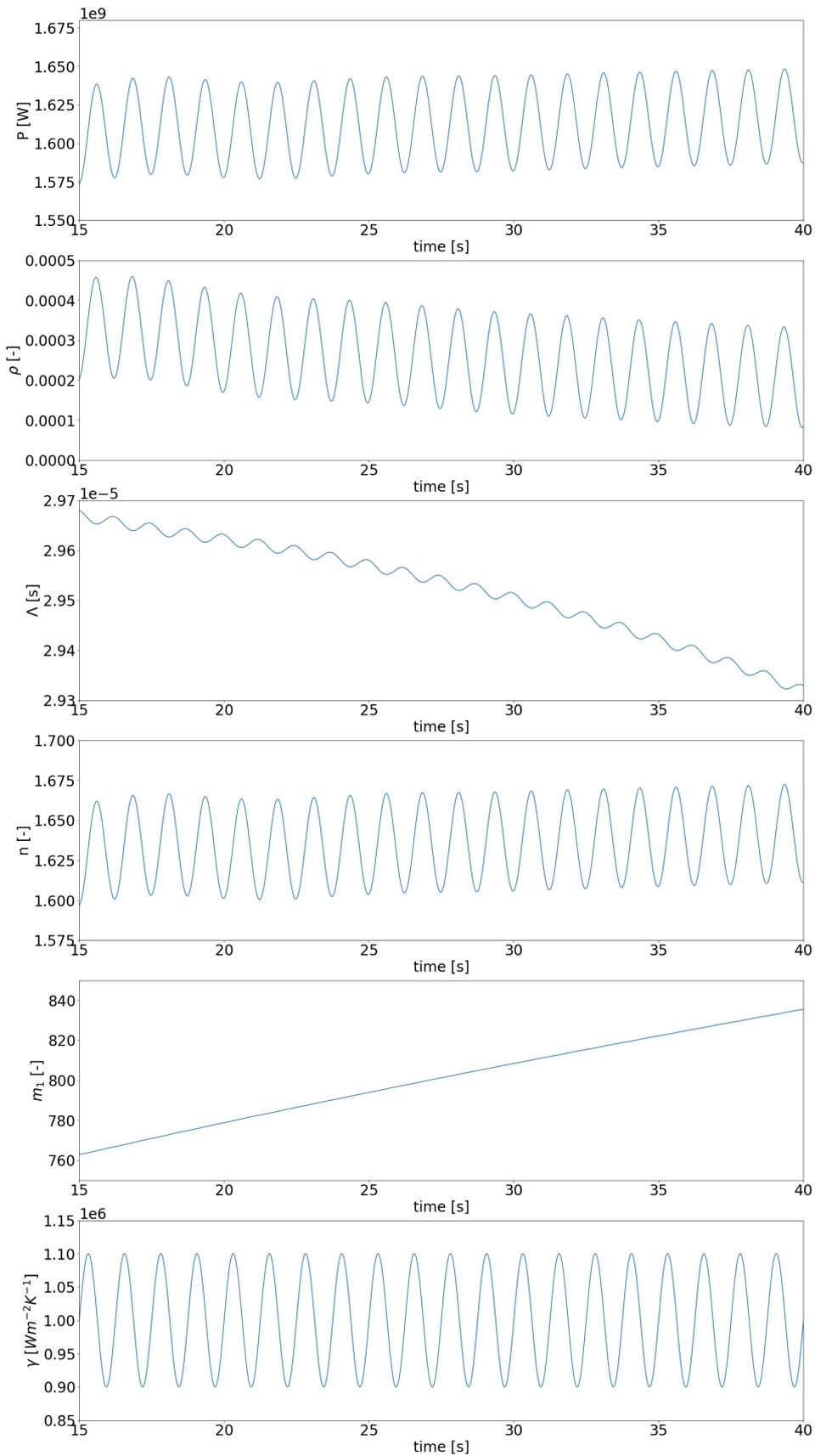


Figure E.7: Transient results for perturbed heat coefficient with $f = 0.8Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

E.2. Results of transient simulations displaying outcomes after power jump stabilization period

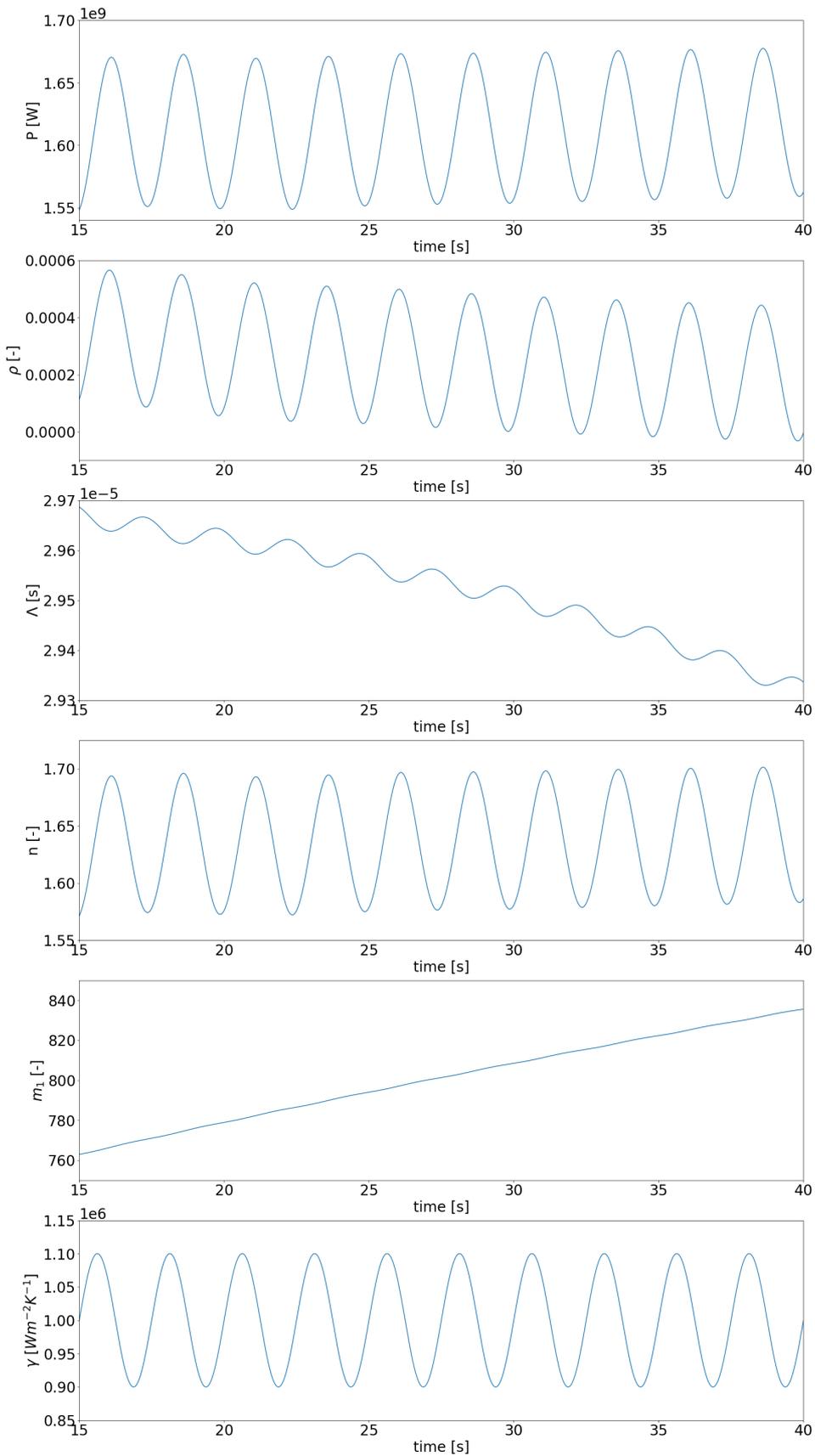


Figure E.8: Transient results for perturbed heat coefficient with $f = 0.4Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

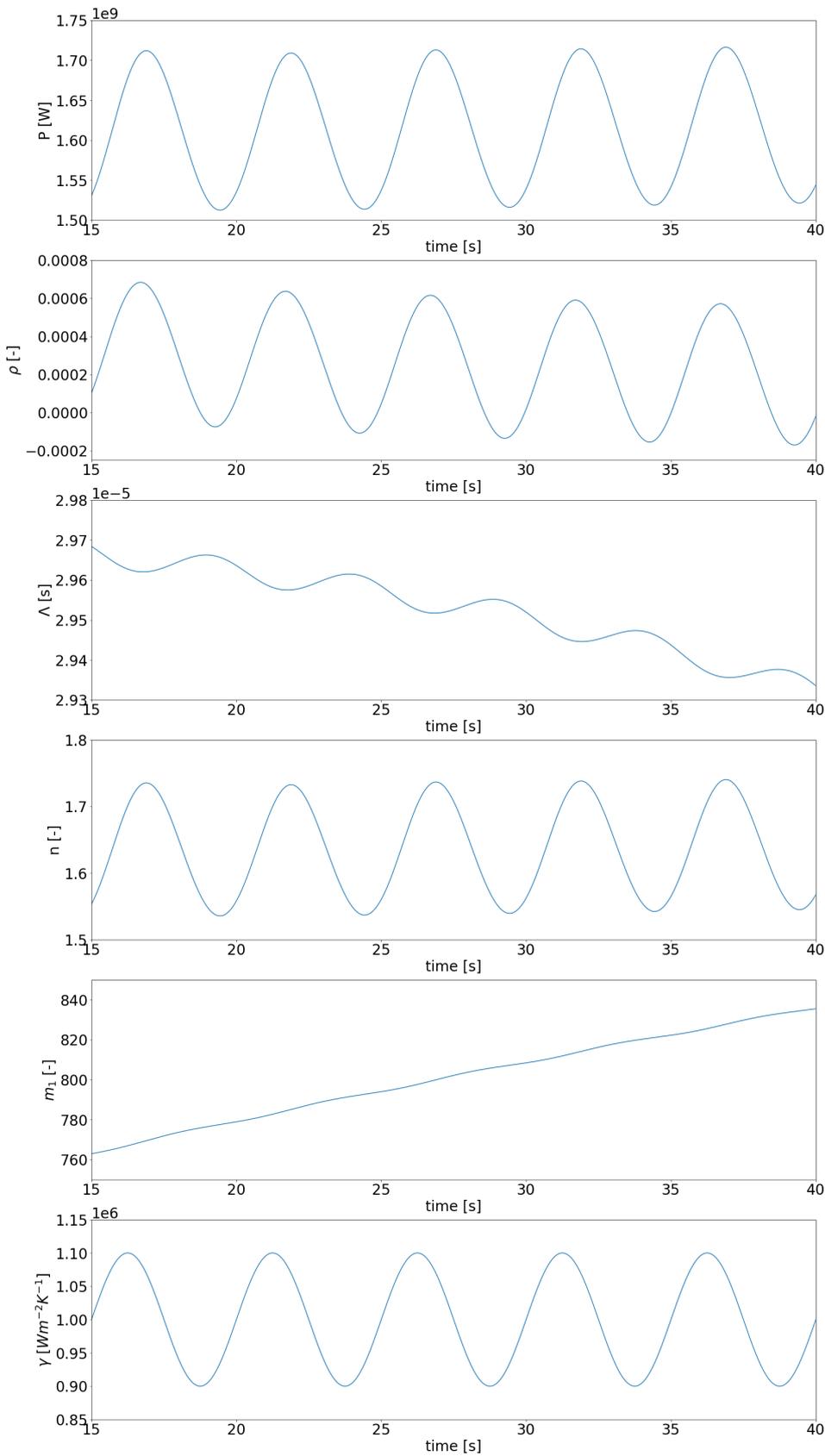


Figure E.9: Transient results for perturbed heat coefficient with $f = 0.2Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

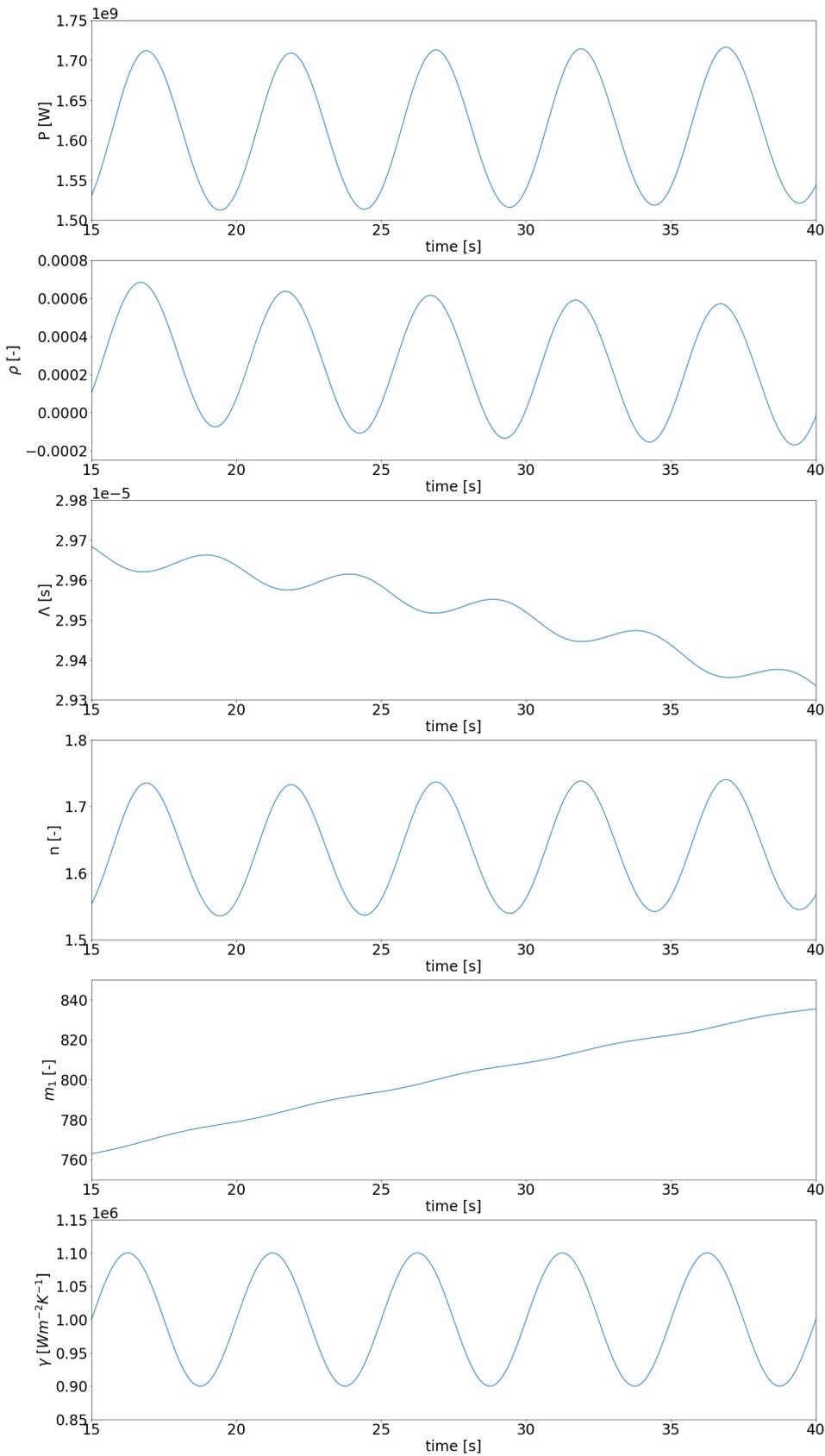


Figure E.10: Transient results for perturbed heat coefficient with $f = 0.2Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

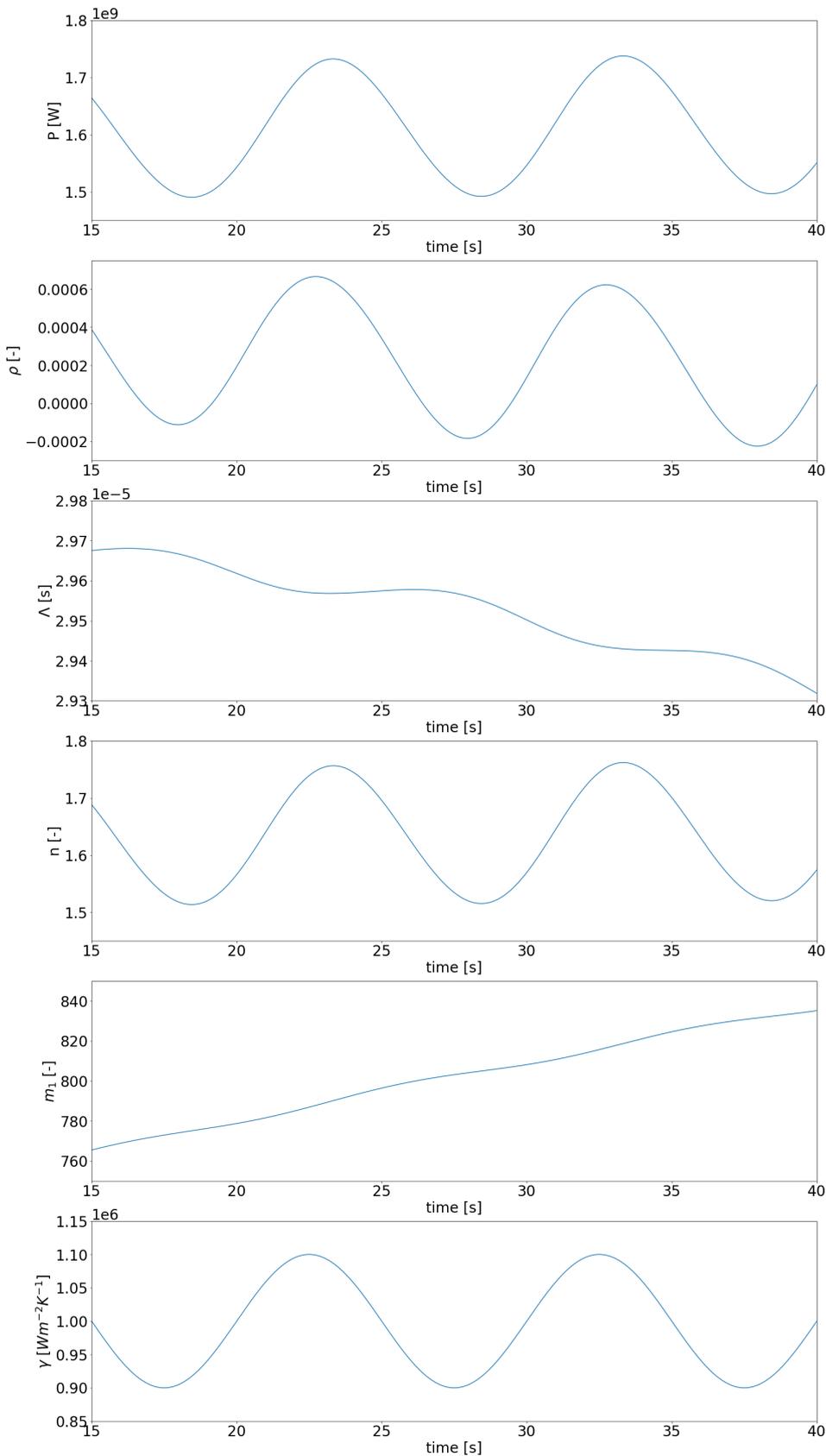


Figure E.11: Transient results for perturbed heat coefficient with $f = 0.1Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

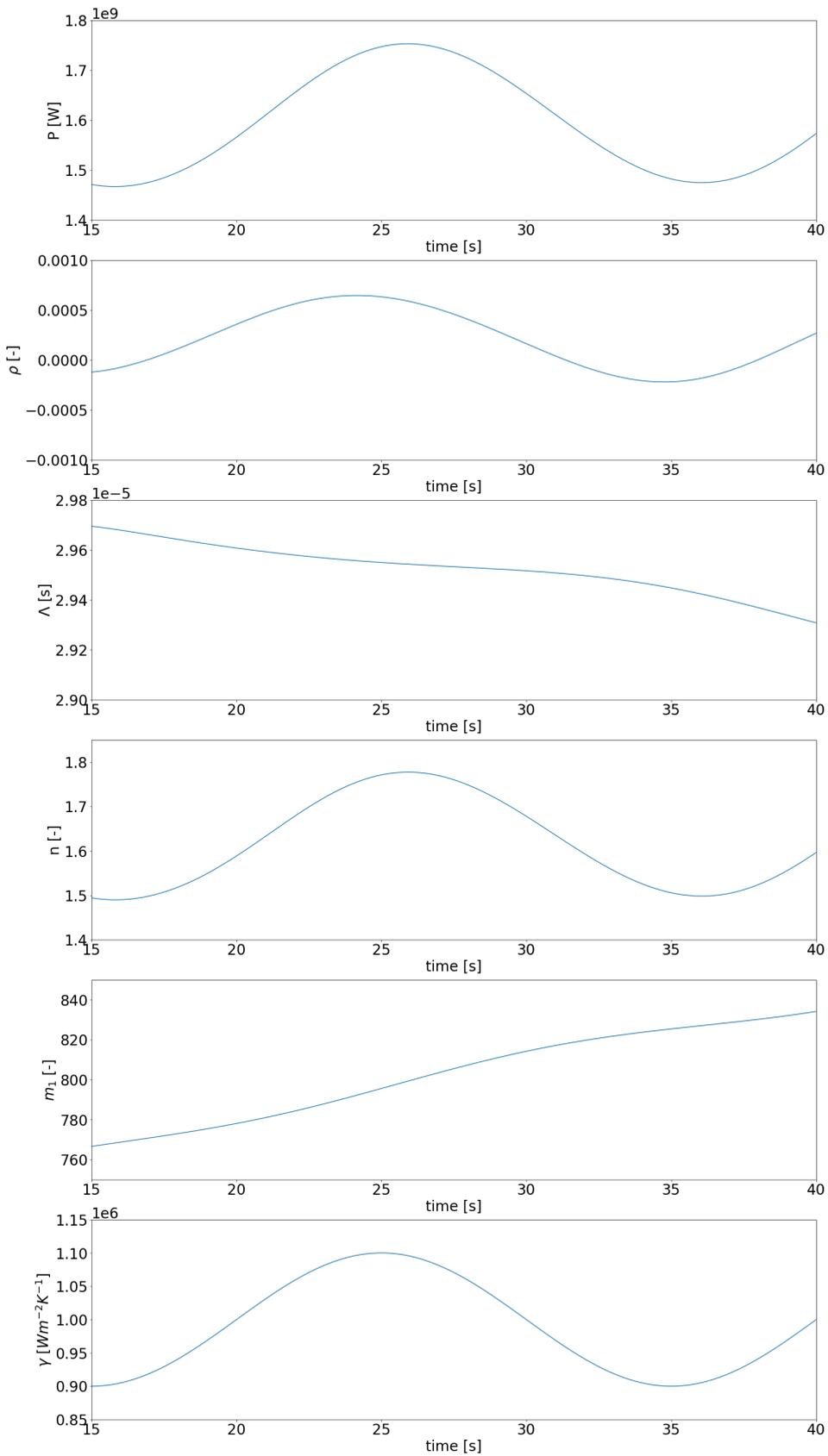


Figure E.12: Transient results for perturbed heat coefficient with $f = 0.05Hz$, for time-interval $t = [15s, 40s]$ after initial perturbation.

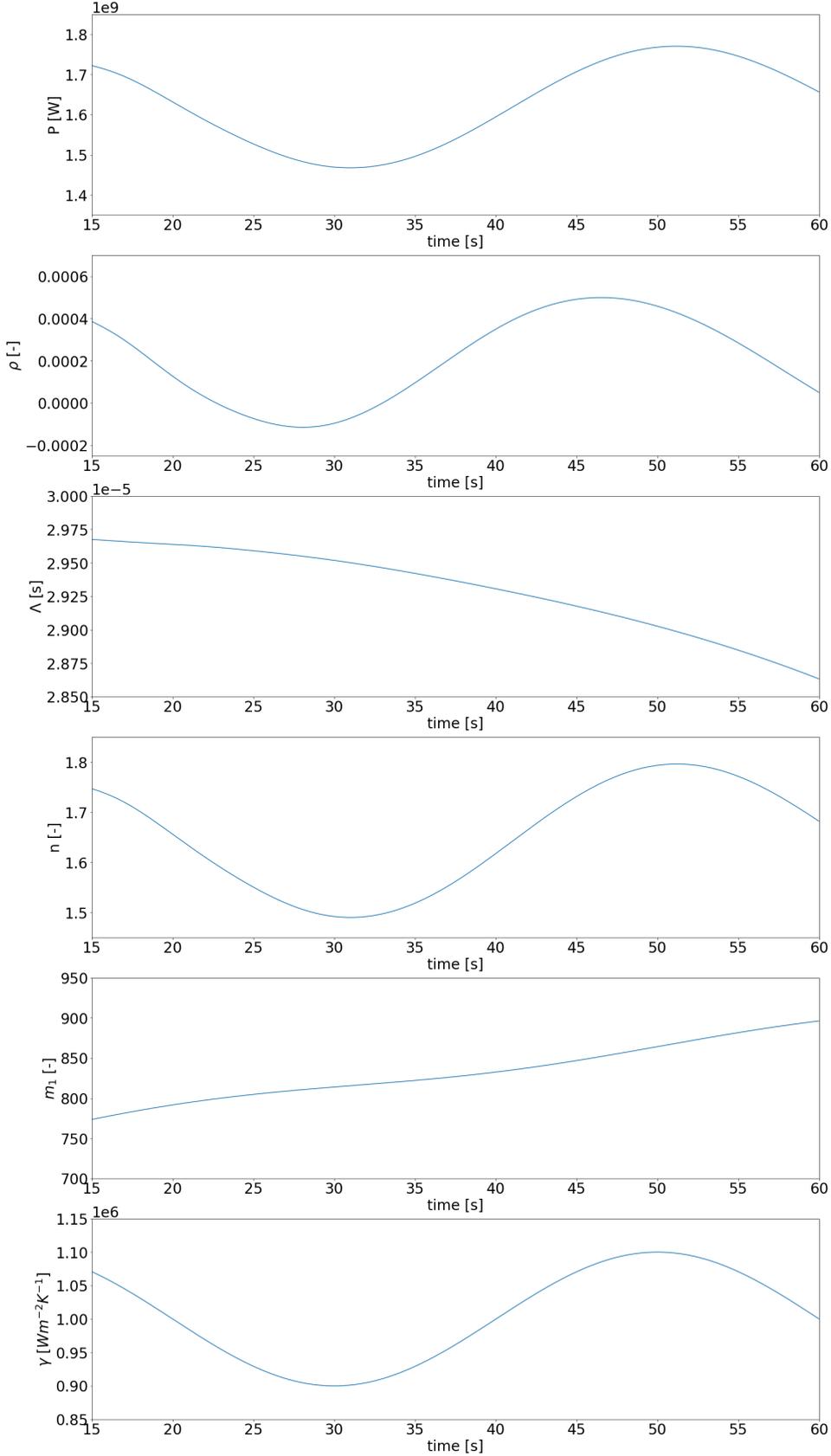


Figure E.13: Transient results for perturbed heat coefficient with $f = 0.025Hz$, for time-interval $t = [15s, 60s]$ after initial perturbation.

F

Wall-clock times and memory storage

F.1. Wall-clock times for selected simulations

In this chapter, the wall-clock times for three simulation types are represented. These three are chosen as they represent the 3 different simulation types that were performed in this research: an uncoupled steady-state simulation (phase 1.1), a fully coupled steady-state simulation and a fully coupled transient simulation.

Phase	Description of conditions	Wall-clock time [hrs]
1.1	Simulation of phase 1.1 for 200x200 lattice nodes	42
1.1	Simulation of phase 1.1 for 150x150 lattice nodes	25
1.1	Simulation of phase 1.1 for 100x100 lattice nodes	12
1.1	Simulation of phase 1.1 for 50x50 lattice nodes	3
1.3	Simulation of phase 1.1 for 300x300 lattice nodes	65
2.1	Simulation of response to periodic perturbation for first 100,000 time-steps after first perturbation for 50x50 lattice nodes	23

Table F.1: Wall-clock times for selected simulations

F.2. Memory usage per node in simulations

The memory usage for each node is calculated by evaluating the size of the arrays maintained at that node. Since the array elements are float64 values, the total memory requirement is determined by multiplying the count of stored parameters per node by 64 bits.

Phase	Memory usage [bits / node]
0.1	1472
0.2	14592
0.3	1408
1.1	16256
1.2	16384
1.3	18752
1.4	18752
2.1	19072

Table F.2: Memory usage per node for selected simulations