Modelling a Freeze Plug with the Filter Matrix Lattice Boltzmann Method to Investigate effects of Freeze Plug Design Parameters on Melting Times.

Master Thesis of Robin de Waal

RPNM research group TU Delft

R.T. de Waal

Delft University of Technology

Modelling a Freeze Plug with the Filter Matrix Lattice Boltzmann Method to Investigate effects of Freeze Plug Design Parameters on Melting Times.

Master Thesis of Robin de Waal

by

R.T. de Waal

Delivered:	31/06/2025
Student number:	5121671
TU Delft supervisor:	dr. ir. M. Rohde
Thesis committee:	dr. ir. D. Lathouwers dr. S. R. de Roode dr. ir. M. Rohde

Preface

R.T. de Waal
Delft, July 2025

Absract

The Freeze Plug is a passive safety feature considered for molten salt reactors. The freeze plug should melt as quickly as possible when a reactor shuts down. Using a filter matrix lattice Boltzmann method, this thesis aims to model the melting process of a freeze plug. With a working model, the physical parameters of the freeze plug can be researched to minimize melting times. The filter matrix method is a nascent method that can still be optimized and researched.

Several benchmark studies were used to build up to a model for the melting of the freeze plug. The first benchmark simulates natural convection in a square cavity. The second adds a conjugate boundary by adding a solid wall at one of the sides. The third benchmark simulates melting a PCM using natural convection. Simulating the first two benchmarks provided good results except in low Ra number regions $Ra = 10^4$.

The freeze plug model was implemented following a benchmark proposed by Pater and Kaaks. The temperature measurements inside the domain of the freeze plug without natural convection compared well to the benchmark. As well as the melting front of the freeze plug without natural convection. The freeze plug model with natural convection suffered from long compilation times. This made solving issues tricky since large grid size was needed to improve stability. In the end, even for large grid sizes the model did not end up being stable.

To improve the boundary method of the filter matrix, corners were specifically examined. Corners are normally implemented in a special way in lattice Boltzmann wet node boundary methods. To find whether the filter matrix method should also treat these specially, different corner implementations were compared to benchmark results. The results show the most success using no special corner implementation. Diagonal and pure adiabatic boundary orientation of corner normals showed the worst results.

Different grid size options and kinematic viscosity values are tested to try to find a relation between performance / compilation time and accuracy and stability. Overall it was found that the reaction became unstable when the time conversion variable went to a value lower than $1.3 \cdot 10^{-5}$. Stable solutions were still obtained for Courant numbers higher than 1 but produced far more error.

Contents

Preface i

Summary ii

Nomenclature viii

1 Introduction 1
- 1.1 Nuclear reactors 1
- 1.2 Molten salt reactor 2
- 1.3 Freeze plug 3
- 1.4 Earlier work 3
  - 1.4.1 Freeze Plug 3
  - 1.4.2 Numerical method 4
- 1.5 Research questions 4

2 Theoretical Framework 6
- 2.1 Fluid Dynamics 6
- 2.2 Relevant Physical Phenomena for the Freeze Plug 6
  - 2.2.1 Natural Convection 6
  - 2.2.2 Conjugate heat transfer 8
  - 2.2.3 Enthalpy balance for melting 8
- 2.3 Lattice Boltzmann Method 10
  - 2.3.1 Kinetic Theory 10
  - 2.3.2 Discretisation and Velocity Sets 10
  - 2.3.3 Chapman-Enskog analysis 11
  - 2.3.4 Double Distribution Function 11
  - 2.3.5 Advantages and disadvantages LBM 11

3 Numerical Method 13
- 3.1 Lattice units 13
  - 3.1.1 Conversion factors 13
  - 3.1.2 Choice of free parameters 14
- 3.2 FMLBM Collision 14
  - 3.2.1 Collision operators 14
  - 3.2.2 Momentum 15
  - 3.2.3 Temperature 16
  - 3.2.4 Enthalpy 17
- 3.3 Streaming 17
  - 3.3.1 Opposite Streaming direction 17
- 3.4 Boundary Conditions 18
  - 3.4.1 Bounce back methods 18
  - 3.4.2 Filter matrix boundary method 19
  - 3.4.3 Conjugate boundary 21
  - 3.4.4 Corners 22
  - 3.4.5 Phase front 22
- 3.5 Algorithm 23

4 Benchmarking Individual Models 25
- 4.1 Natural convection 25
- 4.2 Conjugate Boundary 28
- 4.3 Enthalpy melting 31

5 Freeze Plug model **39**
5.1 Conjugated conductive heat model . . . 41
5.2 Natural convection model . . . 41

6 Stability and Performance FMLBM **45**
6.1 Corner influence . . . 45
6.1.1 No corners . . . 45
6.1.2 Orthogonal corners . . . 45
6.1.3 Diagonal corners . . . 45
6.1.4 Quantitative results . . . 49
6.2 Free parameters . . . 50

7 Conclusions **53**

8 Recommendations **54**

References **56**

A Appendix **61**
A.1 Algorithm benchmarks . . . 61
A.1.1 Natural convection benchmark . . . 61

List of Figures

1.1 Schematic of neutron life cycle [1] 1
1.2 Schematic of a simple pressurized water reactor (PWR) [3] 2
1.3 A schematic of a molten salt reactor including the freeze plug (here called the freeze valve) and drain tank. 3
1.4 This figure shows a schematic of two freeze plug designs designs that were researched [10] [12]. The grey represents Hastelloy-N, a blue circle represents a freeze plug. 4
1.5 Two snapshots from the transient analysis from van Dijk show that the molten salt is able to reach the containment chamber at t=730 seconds. This is faster than the 0° transient analysis performed by van Dijk which showed an opening at t=1520 seconds. 5

2.1 An example of velocity streamlines in a heated square cavity showing the effects of natural convection.[21] 7
2.2 The enthalpy change with a mushy phase change (a). And the enthalpy change with an isothermal phase change at T_m (b) 9
2.3 Schematic of a D2Q9 velocity set. With e_i denoting the direction of the velocity. [31] . . 11

3.1 Schematic of both boundary methods. The computational boundary of the link wise method is placed at a distance of $\frac{\Delta x}{2}$ from the Physical boundary. 18
3.2 The physical representation of a halfway bounce back boundary interaction. During the collision process both normal and tangential momentum components are reversed. The average particle momentum, before and after collision, is $\langle mv \rangle = (mv(t) + mv(t + \Delta t))/2 = \mathbf{0}$ [28] 19
3.3 A schematic showing both convex and concave corner cells in a LBM simulation. The arrows signify unknown density populations when using the NEBB method. 22

4.1 A schematic of the natural convection benchmark case, as performed by Davis [48] and Zhuo [41]. The type of the boundary condition is noted by both text and color. 25
4.2 The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = 10^4 in this simulation. 26
4.3 The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = 10^5 in this simulation. 27
4.4 The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end).The final time shown is the the time for which the solution has converged. Ra = 10^6 in this simulation. 27
4.5 These figures show the temperature isotherms found by De Vahl Davis [48]. Each figure shows the converged solution for a different Rayleigh number. 27
4.6 The schematic for the conjugate boundary condition benchmark. The right edge of the wall is heated, and the wall and fluid domain are connected through a conjugate boundary. 29
4.7 The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = 10^4 in this simulation. 29
4.8 The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = $0.7 \cdot 10^5$ in this simulation. 30
4.9 The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = $0.7 \cdot 10^6$ 30

4.10 The schematic for the enthalpy melting benchmark. The left edge is heated, while the rest of the walls are adiabatic. The domain is filled completely with PCM. 31
4.11 Temperature gradient for enthalpy melting benchmark at different times for Ra = $5 \cdot 10^4$ 32
4.12 The liquid fraction for the enthalpy benchmark at different times for Ra = $5 \cdot 10^4$ 33
4.13 The temperature gradient for the enthalpy benchmark at different times for Ra = $1.7 \cdot 10^5$ 34
4.14 The liquid fraction for the enthalpy benchmark at different times for Ra = $1.7 \cdot 10^5$ . . . 35
4.15 Different regimes of melting. From conduction dominated melting to natural convection dominated melting. [55] 36
4.16 Quantitative results for enthalpy melting benchmark with Ra = $5 \cdot 10^4$ 37
4.17 Quantitative results for enthalpy melting benchmark with Ra = $1.7 \cdot 10^5$ 37

5.1 Schematic of freeze plug model and boundary conditions used. The Dirichlet boundary condition is dependant on decay heat as well as the volumetric heat source in the PCM domain. The PCM domain is considered a solid salt that is partially molten in the upper part with a height of 0.1 N_Y. The wall is made of Hastelloy-N and spans 0.2 N_X in the x direction.[47] 40
5.2 Results of modelling the freeze plug without natural convection at t = 1000 s 41
5.4 Temperature thermometers displaying temperature at a specific x coordinate over the y position. 43
5.5 Meltfront results compared to benchmark results for the freeze plug without natural convection at t = 2500 s 44
5.6 Liquid fraction freeze plug at approximately 2.5 seconds 44

6.1 A zoomed in representation of the boundary workings near the corner. The corner normal orientation possibilities are shown in different colors. Pink shows vertical orientation towards a Dirichlet BC, orange shows diagonal orientation, both Dirichlet BC and adiabatic BC possible, green shows horizontal orientation towards adiabatic BC. 46
6.2 An example of the no corner method. Both boundary conditions are executed on the same corner node. 46
6.3 Liquid fraction for enthalpy melting with Ra = $1.7 \cdot 10^5$ for corners that were not treated special. 47
6.4 Liquid fraction for enthalpy melting with Ra = $1.7 \cdot 10^5$ for vertically oriented corners that follow dirichlet BC over adiabatic BC 47
6.5 Quantitative results for enthalpy melting benchmark with Ra = $5 \cdot 10^4$ for horizontally oriented corners that always follow adiabatic BC 48
6.6 Quantitative results for enthalpy melting benchmark with Ra = $5 \cdot 10^4$ for diagonally oriented corners that follow adiabatic BC over Dirichlet BC 48
6.7 Qualitative results for enthalpy melting benchmark with Ra = $5 \cdot 10^4$ for diagonally oriented corners that follow Dirichlet BC over adiabatic BC 49
6.8 Quantitative results for enthalpy corner method differences with Ra = $1.7 \cdot 10^5$ 49
6.9 The Nusselt number reached for different grid sizes and viscosity values for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale. 50
6.10 The Nusselt number reached for different grid sizes and C_{time} for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale. Unstable solutions are represented by red dots just below the benchmark Nusselt line. 51
6.11 The Nusselt number for different grid sizes and STPRT for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale.Unstable solutions are represented by red dots just below the benchmark Nusselt line. 52

List of Tables

4.1 Physical and lattice parameters (Natural convection benchmark case) 26
4.2 Comparison of results for different Rayleigh numbers for natural convection benchmark. 28
4.3 Physical and lattice parameters (Conjugated boundary bench mark) 29
4.4 Comparison of results for different Rayleigh numbers for conjugateed heat benchmark. . 30
4.5 Physical and lattice parameters (Case 2 - Phase Change) 32

5.1 Used thermophysical properties for fuel salt and Hastelloy N and some general parameters 40
5.2 Lattice parameters chosen for freeze plug model without natural convection. 41
5.3 Lattice parameters chosen for freeze plug model with natural convection. 42

6.1 Lattice parameters chosen for stability and accuracy measurements. 50
6.2 The associated Λ factor for different grid sizes for the natural convection benchmark. . . 51

Nomenclature

In order of appearance.

Abbreviations

Abbreviation	Definition
PWR	Pressurized Water Reactor
HTR	High Temperature Reactor
MSR	Molten Salt Reactor
MSFR	Molten Salt Fast Reactor
TRU	Transuranic
LBM	Lattice Boltzmann Method
FMLBM	Filter Matrix Lattice Boltzmann Method
FD	Fluid Dynamics
CFD	Computational Fluid Dynamics
BGK	BhatnagarGrossKrook
LBE	Lattice Boltzmann Equation
PCM	Phase Change Material
NEBB	Non-Equilibrium Bounce Back
HBB	Half-Way Bounce Back
BC	Boundary Condition
MDF	Multiple Distribution Function
FBB	Full-way Bounce Back
LUMPS	Lattice Updates Per Second
MLUMPS	Million Lattice Updates Per Second
STPRT	Simulation Time Per Real Time

Symbols

Symbol	Definition	Unit (SI)	Unit (lattice)
σ	Neutron leakage	[-]	
F	Force	$[kg \cdot m/s^2]$	$[kg \cdot ls/lt^2]$
ρ	Density	$[kg/m^3]$	$[1/ls^3]$
V	Volume	$[m^3]$	$[ls^3]$
f	Body force	$[kg/m^2s^2]$	$[lm/ls^2]$
α	Thermal expansion coefficient	$[K^{-1}]$	$[K^{-1}]$
T	Temperature	$[K]$	$[K]$
c_P	Specific heat capacity	$[Ws/kgK]$	$[ls^2/lt^2K]$
L	Length	$[m]$	$[ls]$
$\ddot{q}$	Heat flux	$[W/m^2]$	
λ	Thermal conductivity	$[W/mK]$	$[ls/lt^3K]$
x	Distance	$[m]$	$[ls]$
t	Time	$[s]$	$[lt]$
H	Enthalpy	$[m^2kg/s^2]$	$[ls^2kg/lt^2]$
$\boldsymbol{u}$	Velocity	$[m/s]$	$[ls/lt]$
q_S	Volumetric source term	$[W/m^3]$	
h_S	Sensible enthalpy	$[m^2kg/s^2]$	$[ls^2kg/lt^2]$

Symbol	Definition	Unit (SI)	Unit (lattice)
h_L	Latent enthalpy	$[m^2kg/s^2]$	$[ls^2kg/lt^2]$
f_L	Liquid fraction	[-]	[-]
φ	Phase indicator		
κ	Thermal diffusivity	$[m^2/s]$	$[ls^2/lt]$
μ	Dynamic viscosity	$[kg/(m \cdot s)]$	
p	Pressure	$[kg/s^2m]$	
ζ	Bulk viscosity	$[kg/(m \cdot s)]$	
ξ	Particle velocity	$[m/s]$	$[ls/lt]$
Ω	Collision operator		
f_i	Density distribution function		
g_i	Temperature distribution function		
m_i	Enthalpy distribution function		
C_α	Conversion factor		
ν	Kinematic viscosity	$[m^2/s]$	$[ls^2/lt]$
N_α	Number of cells in α direction		
τ	Relaxation time	$[s]$	$[lt]$
c_s	Lattice speed of sound		$[ls/lt]$
$\mathbf{M}$	Transformation matrix		
$\mathbf{S}$	Relaxation time vector		
c_i	discretized lattice speed		$[ls/lt]$
ω_i	Directional weights vector		
E_{ik}	Filter matrix		
α_k	Density solution vector		
β_k	Density solution vector		
γ_k	Density solution vector		
S	Friction term	$[kg/(m \cdot K \cdot s)]$	$[1/(ls \cdot K \cdot lt)]$
ϕ	Conversion variable		
Γ	N Conversion check time steps		
δ_ϕ	Conversion criterium		
h_T	Convective heat transfer coefficient	$[W/m^2K]$	
k	Thermal conductivity	$[(W/(m\ K)]$	
ϵ_ϕ	Error	[-]	[-]
d	Wall thickness	$[m]$	$[ls]$
θ	Dimensionless time	[-]	[-]
Q	Volumetric heat source	$[W/m^3]$	

Constants

Symbol	Definition	Value (SI)
g	Gravitational constant	9.81 $[m/s^2]$
A_{mush}	Morphological friction constant	10^8 $[kg/(m \cdot K \cdot s]$
σ_1	Computational constant	10^{-3} [-]

Dimensionless numbers

Symbol	Definition	meaning
Pr	Prandtl number	momentum diffusivity / thermal diffusivity
Nu	Nusselt number	total heat transfer / conductive heat transfer

Symbol	Definition	meaning
Ste	Stefan number	sensible heat / latent heat
Co	Courant number	information travel / max cell travel size
Re	Reynolds number	inertial forces / viscous forces
Ra	Rayleigh number	diffusive transport / convective transport
Fo	Fourier number	time / time scale for diffusion

1

Introduction

Some say that nuclear energy is necessary for the transition from fossil fuels to clean energy sources. Whether that is truly the case is beyond the scope of this project/paper. This paper will focus on the design properties of the freeze plug utilized in a molten salt reactor.

1.1. Nuclear reactors

Nuclear reactors come in all shapes and sizes, but most have a few things in common. The object of a reactor is to have neutrons interact with fissile atoms such as uranium-235. This interaction can then cause fission, which also produces neutrons which then interact with more fuel, which causes more fissions, and so on. In order to have a critical reactor, the amount of neutrons that are being produced needs to be equal to the amount of neutrons that are lost due to reactions or leakage. In figure 1.1 this process is represented by a starting number of neutrons n that go through several processes. Each process alters the number of neutrons which is represented by a factor σ_i. So, the number of neutrons after fast fission has occurred would be equal to $\sigma_f n$. So, for the reactor to be critical $\sigma_c = 1$.

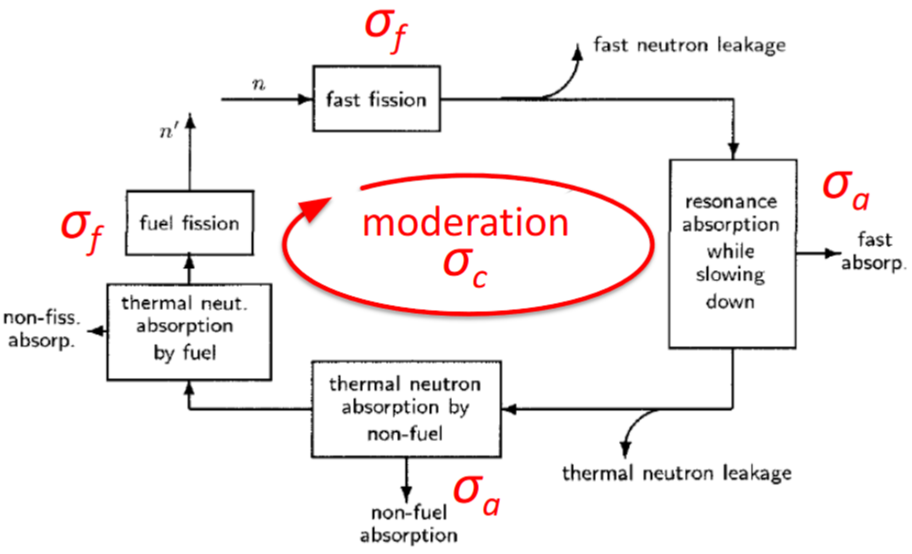


Figure 1.1: Schematic of neutron life cycle [1]

To increase the fission cross section of the neutrons, moderators are used to slow down the neutrons through inelastic collisions. By slowing down the neutrons enough to make them 'thermal', their fission cross-section is increased. [2] Some modern nuclear reactors do not use moderators but instead use richer fuel and/or a higher neutron flux to get enough fission reactions. These reactors are called fast reactors.

A material that is often used as a moderator is water. This material also functions as a coolant.

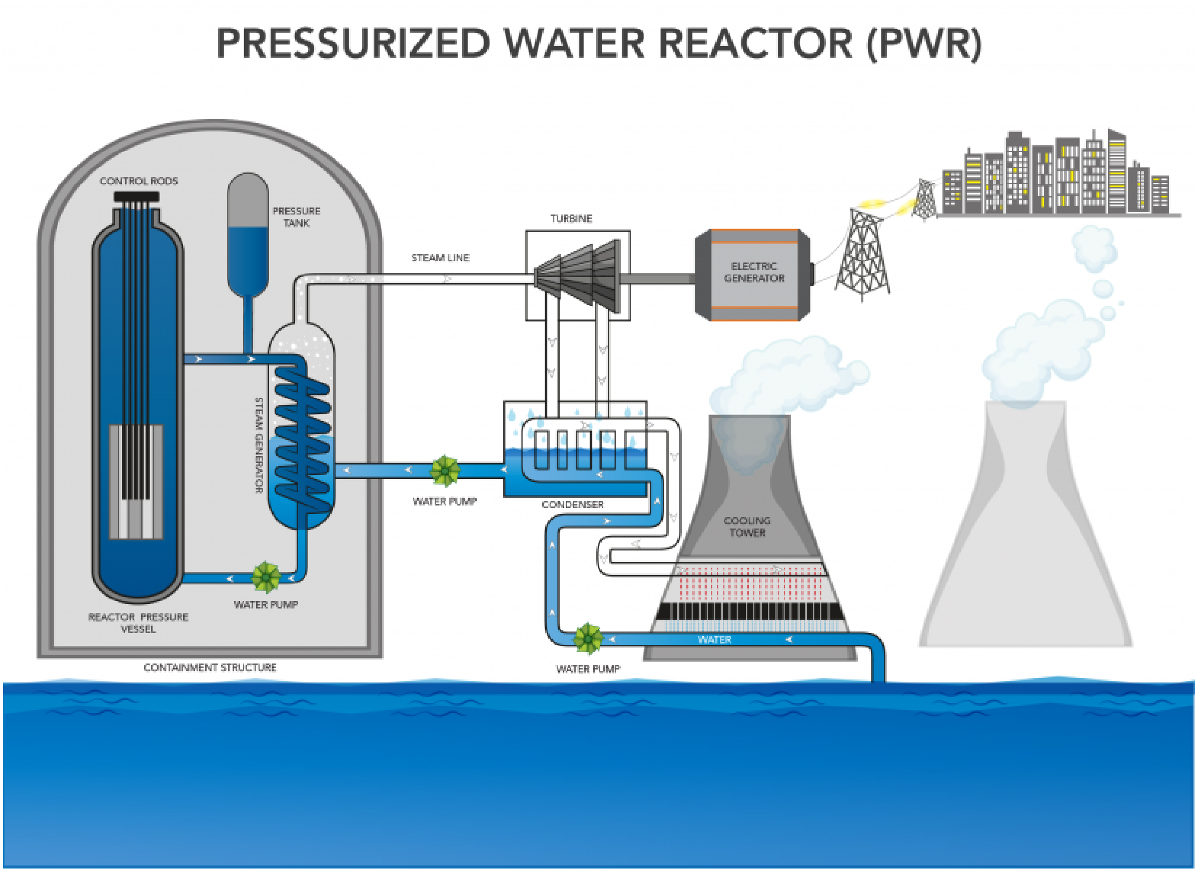


Figure 1.2: Schematic of a simple pressurized water reactor (PWR) [3]

The fission reactions create heat, which in the case of the pressurized water reactor in figure 1.2 is stored in the coolant and then transformed into electricity.

1.2. Molten salt reactor

Molten salt reactors (MSRs) are a type of nuclear reactor in which the primary coolant or even the fuel itself consists of a molten salt mixture. There are two primary subclasses, one in which fissile material is dissolved in the salt mixture, and one in which the molten salt serves as the low pressure coolant to a coated particle-fueled core similar to that employed in high temperature reactors (HTRs). [4] We will focus on liquid-fueled MSRs. Some benefits of these liquid-fueled MSRs or Molten Salt Fast Reactors (MSFRs) are:

- Price, Molten salt reactors should be relatively cheap compared to reactors from previous generations due to the possible lack of a thick containment unit and due to the high thermal efficiency [5] [6]
- The MSFR creates very little TRU (transuranic) waste. This is among other things due to the fast spectrum under which the MSFR operates.

Because of these benefits, some countries are quick in developing these reactors. In September 2018 China started construction of an experimental thorium-powered molten salt reactor. It was supposed to finish construction but, the reactor was completed in 2021 and was granted an operating license in 2023. [7]

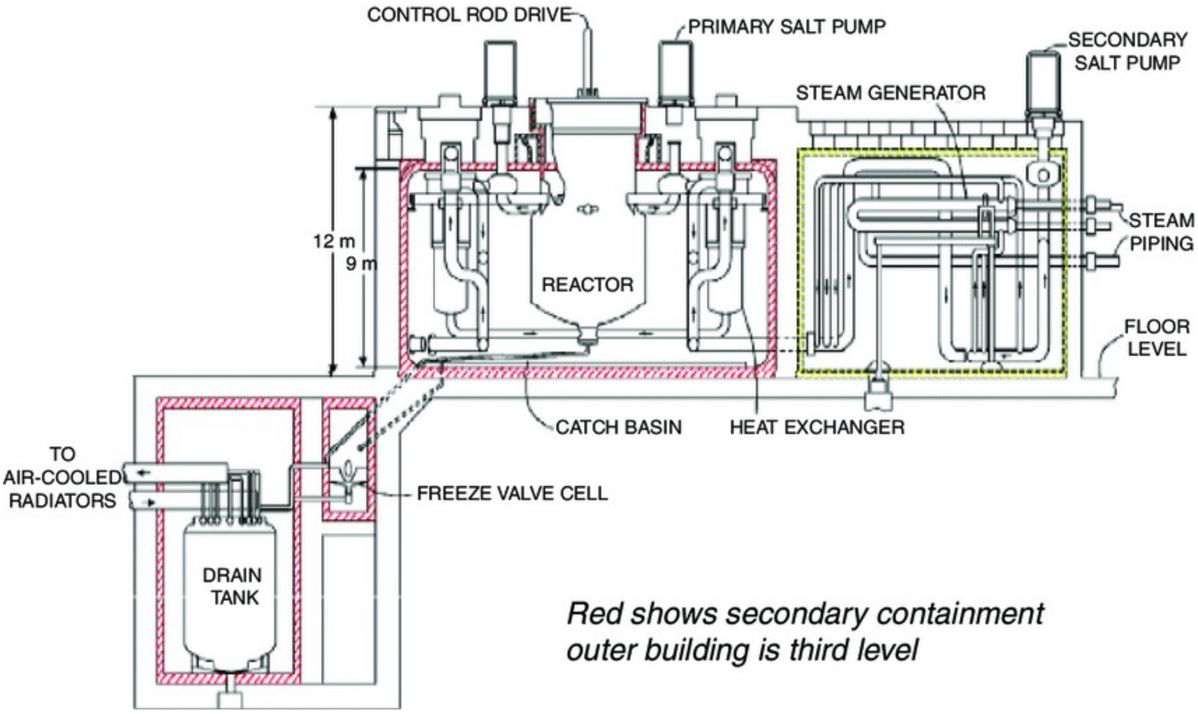


Figure 1.3: A schematic of a molten salt reactor including the freeze plug (here called the freeze valve) and drain tank.

This research considers an MSFR breeder reactor under ambient pressure and at 750 °C. [10][11]

1.3. Freeze plug

Another advantage of the MSR is the possibility of another safety measure called the freeze plug. The MSFR freeze plug, also called a freeze valve or cold plug, is a vertically oriented cylinder of salt located in the draining pipe of the MSF. The freeze plug prevents the molten salt mixture from falling into a separate containment tank. The plug consists of salt that is kept frozen/solid through the use of external coolants that are placed outside the pipe. When there is a power outage or station blackout, this plug melts, which causes the molten salt mixture to fall into the containment tank. This passive safety feature prevents the reactor core from going critical due to a power outage. It is important that the freeze plug melts fast enough to let the molten salt mixture in the drain tank before the reactor temperature in the reactor core is higher than 1473 K [8]. According to previous research, this can happen between 480 s [8] and 1600 s [9][10].

1.4. Earlier work

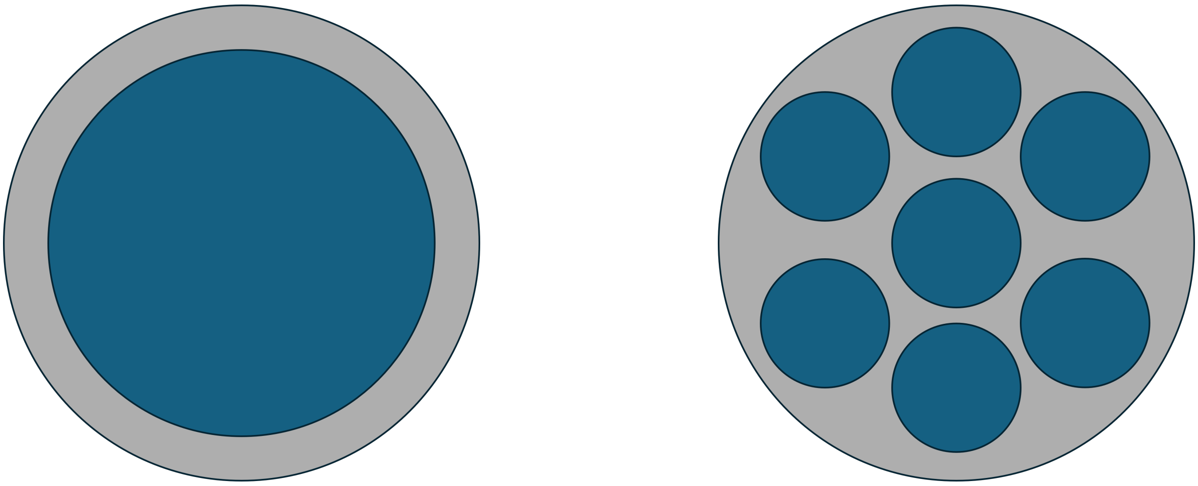
1.4.1. Freeze Plug

Quite a lot of earlier work has been done on the freeze plug.

In 2016 Swaroop had investigated the design of the freeze plug. He modeled the melting process using a deformed geometry method in COMSOL (COMSOL is a finite element analyzer). In his research he found that a simple single freeze plug placed in parallel with respect to gravity did not meet the required melting time. [11]

In 2017, Shafer considered a grated design of the freeze plug. Using COMSOL Shafer found that multiple smaller plugs might result in a faster melting process. [10] The same was also found and confirmed by Makkinje [12] when researching this grated design with the new lattice Boltzmann method, which will not be one of the goals of this paper. But it could be considered when there is a lot of extra time available.

In 2020 Aji considered placing the freeze plug at an angle. If a 'normal' vertical freeze plug would be considered at a 0° angle. Aji found through finite-volume methods that natural convection in the molten salt intensified with an increase in the angle of inclination.[13] That is, when the freeze plug



(a) Single freeze plug standard design **(b)** Seven freeze plugs grated design

Figure 1.4: This figure shows a schematic of two freeze plug designs designs that were researched [10] [12]. The grey represents Hastelloy-N, a blue circle represents a freeze plug.

is tilted more, it melts faster. Aji considered a different salt (FliBe) and significantly different plug dimensions, so the results cannot be directly translated to the MSFR considered in this paper.

In 2022 van Dijk furthered research on the time dependence of the inclination angle of the freeze plug. Van Dijk used a linearized enthalpy method to model the melting process of the freeze plug. His model showed that placing the freeze plug on an incline of 45° results in the molten salt falling into the containment chamber faster compared to the normal 0° incline.[14] However, van Dijk had problems verifying his results, since they did not coincide with the results obtained by Shafer and Reus. [10][15]

In 2023 and 2024 Borstlap continued work on the Freeze plug. She developed a filter-matrix lattice Bolzmann method with double distribution functions and used a wet-node boundary scheme to develop a wet-node-solution-vector boundary condition. Separate models were used to model convective thermal flows, conjugate heat transfer, and phase change. These models were then put together to create a large model to analyze the behavior of the freeze plug. This large model worked well except when thermal flow, conjugated heat transfer, and natural convection were combined. When all these aspects were combined, the model showed severe instabilities. [16]

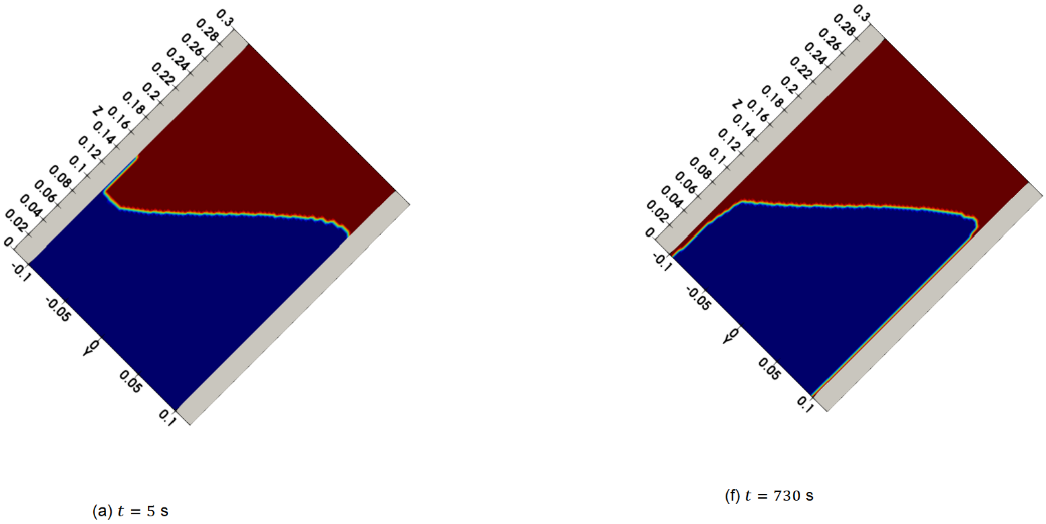
1.4.2. Numerical method

The lattice Boltzmann method has been a popular method for computational fluid dynamics since the 1990s. [17] Between then and now there has been much research on this method. Some relevant research includes the Filter-Matrix method developed by Eggels et al. [18]. It has also been used by Zhuo et al. the filter matrix method is the numerical foundation for this thesis. The further workings of this method are discussed in section 3.2.

1.5. Research questions

This thesis has two main research goals. Goal one is to further develop and explore the filter-matrix lattice Boltzmann method, with the goal to improve the accuracy and stability of this method. Specifically, the boundary method and the stability of this LBM method. The second goal is to be able to model the freeze plug using the FMLBM method and find out which parameters have the greatest effect on minimizing melting times.

- How can the boundary treatment and overall stability of the filter-matrix lattice Boltzmann method be improved or optimized in the case of melting problems that involve natural convection?
- Which physical parameters influence the melting time of a freeze plug, and how can these be adjusted to minimize melting duration?



(a) Transient analysis from van Dijk using a 1.1 million mesh at t=5 seconds and a freeze plug at an inclination angle of 45 °

(b) Transient analysis from van Dijk using a 1.1 million mesh at t=730 seconds and a freeze plug at an inclination angle of 45 °

Figure 1.5: Two snapshots from the transient analysis from van Dijk show that the molten salt is able to reach the containment chamber at t=730 seconds. This is faster than the 0° transient analysis performed by van Dijk which showed an opening at t=1520 seconds.

2

Theoretical Framework

2.1. Fluid Dynamics

The most fundamental equations that are used in Fluid Dynamics (FD) are the conservation laws for mass and momentum. These conservation laws can be transformed into the continuity equation and the Navier-Stokes equations respectively. [19] These equations describe the macroscopic phenomena of fluid motion.

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\boldsymbol{u} = 0 \tag{2.1}$$

$$\rho\frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla\cdot\left[\mu\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T - \frac{2}{3}(\nabla\cdot\boldsymbol{u})\boldsymbol{I}\right)\right] + \nabla\left[\zeta\left(\nabla\cdot\boldsymbol{u}\right)\right] + \rho\boldsymbol{f} \tag{2.2}$$

Equation 2.1 and 2.2 are the Navier Stokes equations. On the left-hand side, the term $\rho\frac{D\boldsymbol{u}}{Dt}$ represents the fluid's inertial response the rate of change of momentum per unit volume. On the right-hand side, there are several actors that influence the fluid. The term $-\nabla p$ determines the pressure gradients. The viscous stress term, involving the dynamic viscosity μ, captures internal friction due to velocity gradients and adjusts for compressibility effects. An additional term involving the bulk viscosity ζ models resistance to uniform compression or expansion, relevant in compressible flows. In this thesis only incompressible flows are considered, and this term is neglected. Since $\nabla\cdot u = 0$ is true when dealing with incompressible flows, the term $\frac{2}{3}(\nabla\cdot\boldsymbol{u})\boldsymbol{I}$ is also neglected. Finally, $\rho\boldsymbol{f}$ includes external body forces such as gravity. Together, these terms describe the balance of momentum in a viscous, compressible fluid. To close this system of equations, the equation of state is used. This equation differs based on the situation that is investigated. These equations cannot be solved analytically for all but the simplest cases. They are not yet proven to be unsolvable analytically, but they are on the millenium prize problem list, so at the very least they are currently neigh impossible to solve.

Solving these equations through conventional methods, i.e. through means of finite difference, finite volume, or finite element, however, is extremely challenging.

2.2. Relevant Physical Phenomena for the Freeze Plug

To accurately model the melting of a freeze plug several forms of heat transfer need to be modeled. These forms can be modeled and tested separate, after which they are combined and implemented in relevant simulations.

2.2.1. Natural Convection

Natural convection describes the effect that occurs when the density of a fluid changes due to the temperature change. This density change combined with gravitational forces causes motion within a fluid. This is due to the buoyancy force acting on the present fluids.

$$F_{buoancy} = \rho V g \tag{2.3}$$

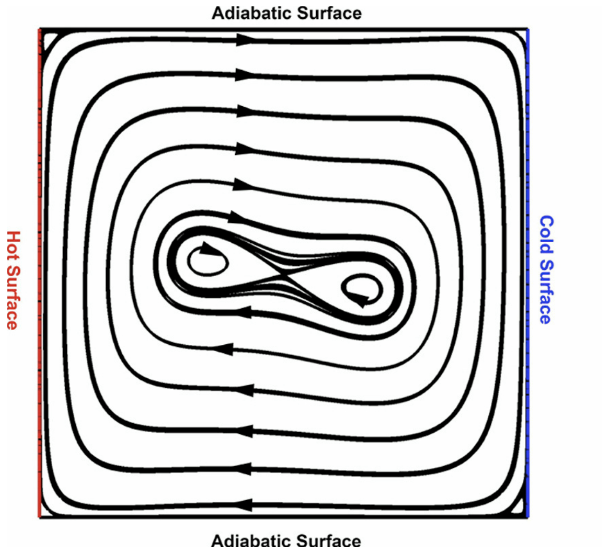


Figure 2.1: An example of velocity streamlines in a heated square cavity showing the effects of natural convection.[21]

When rewriting this to a body force this buoyancy term becomes:

$$f_{buoancy} = \rho g \tag{2.4}$$

The net-buoyancy force that would work on the fluid is

$$f_{buoancy} = \Delta\rho g = (\rho_1 - \rho_0)g \tag{2.5}$$

Where ρ_1 is the density of the fluid or body that is surrounded by a fluid with density ρ_0.

A well known approximation for the density is used in the Boussinesq approximation. It states that density consist of a consistent part, and a part that is linearly dependent on temperature. When combined with a taylor expansion, this leads to the following equation [20]:

$$\rho(T) \approx \rho_0 - \rho_0\alpha\left(T - T_0\right). \tag{2.6}$$

This leads to the following buoyancy force that is applied when dealing with gravity in heated fluids.

$$\mathbf{f}_{buoancy} = \mathbf{g}\rho_0\alpha\left(T - T_0\right) \tag{2.7}$$

When applied to the heating of a fluid inside a square cavity this causes the heated fluid to rise. This causes velocity streams like the one in figure 2.1.

When applied to a phase change melting problem the influence of natural convection is determined by the phase of the melting process and the dimensionless Rayleigh (Ra) and Prandtl (Pr) numbers. [22]

$$\mathrm{Ra} = \frac{\Delta T\alpha N^3 g}{\kappa\nu}, \quad \mathrm{Pr} = \frac{\nu}{\kappa} \tag{2.8}$$

In case of low Rayleigh numbers the basic temperature profile is linear. For high Rayleigh numbers the temperature profile and velocity streams become buoyancy driven.

The Prandtl number characterizes the ratio between momentum diffusivity and thermal diffusivity, thereby determining the relative thickness of the velocity and thermal boundary layers in a natural convection system and influencing the overall heat transfer rate.

The phase of the melting process is also important. Since natural convection is only able to influence heat transfer through moving particles, only molten fluid material experiences the effects of natural convection. So, as more and more material melts, the effects of natural convection become larger and larger. At first there is a conduction regime where conduction is the primary source of heat transfer. Then after some time the melting process enters the convection regime where convection is the primary source of heat transfer.

2.2.2. Conjugate heat transfer

When two different materials transfer heat to each other, typically a form of Fourier's law of heat conduction is used:

$$\ddot{q} = -\lambda \frac{\partial T}{\partial x} \tag{2.9}$$

$\ddot{q}$ represents the heat flux pointed in the direction of the interface that stands between the two materials. At this interface 2 boundary conditions apply. The temperature of material 1 must be equal to the temperature of material 2 at the boundary. And the heat flux of material 1 must be equal to the heat flux of material 2 at the boundary.[23] So for the temperature the following is true at the interface:

$$T_{1,W} = T_{2,W} \tag{2.10}$$

For the heat flux the following boundary condition is derived:

$$\ddot{q}_{1,W} = \ddot{q}_{2,W} \tag{2.11}$$

$$\lambda_{1,W} \left.\frac{\partial T}{\partial x}\right|_{1,W} = \lambda_{2,W} \left.\frac{\partial T}{\partial x}\right|_{2,W} \tag{2.12}$$

How this is solved numerically will be covered in section 3.4.3.

2.2.3. Enthalpy balance for melting

The freeze plug melts when it undergoes a phase change from solid to fluid. To achieve this phase change a certain threshold of energy needs to be reached. This energy can be retrieved from the surroundings of the freeze plug. And since energy is conserved in the domain of the simulation for this thesis, this can best be shown using an energy balance. The energy balance is derived from the Navier-Stokes equations 2.1 and 2.2. [24]

The simplified equations that solve mass and momentum through use of the LBM are

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{2.13}$$

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla P + \nabla \cdot (\rho \nu \nabla \boldsymbol{u}) + \mathbf{F} \tag{2.14}$$

The total enthalpy H of a system can be divided into latent heat and sensible heat

$$H = h_S + h_L = h_S + f_L L \tag{2.15}$$

a sensible part h_S and a latent part $f_L L$. Now the sensible part is the energy that is measurable through measuring the temperature, and thus it fully depends on temperature. The latent part is the energy that a substance gains during phase change, where the temperature does not alter. So, when the substance is solid $f_L = 0$ and when the substance has turned fluid $f_L = 1$. This phenomenon can also be viewed in the subfigure b of figure 2.2 Seperating the enthalpy into two parts is an important step to reach an energy balance that fully depends on enthalpy, while stil relaying information about the temperature. This is called the total-enthalpy based approach. [25]

The total enthalpy of a system is defined as [26]

$$\rho^{\varphi} \frac{DH^{\varphi}}{Dt} = -\nabla \cdot \mathbf{q_S}. \tag{2.16}$$

Here φ indicates the state of the substance when the equation is applied. φ denotes between 3 possible states: liquid, mushy and solid.

The source term for pure substances $\ddot{q}_S$ simplifies to

$$\ddot{q}_S = \frac{\partial(\rho f_L L)}{\partial t}. \tag{2.17}$$

Two more assumptions lead to the final total enthalpy based balance equation. First it is assumed that the density and specific heat of the material are the same in both fluid, solid and mushy conditions. The mushy condition of a material is during the phase change when it is neither solid or fluid. In this regime $0 < f_L < 1$.

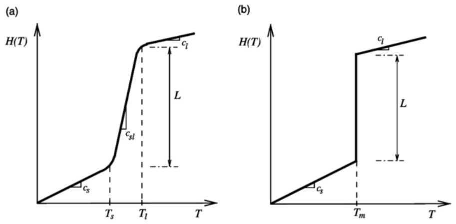


Figure 2.2: The enthalpy change with a mushy phase change (a). And the enthalpy change with an isothermal phase change at T_m (b)

The second assumption is that $\nabla f_L \approx 0$. This assumption can be made since velocities near the mushy zone are approximately zero.

This all leads to the final energy balance

$$\frac{\partial H^{\varphi}}{\partial t} + \mathbf{u} \cdot \nabla h_S^{\varphi} = \nabla \cdot \left(\kappa^{\varphi} \nabla h_S^{\varphi}\right) \tag{2.18}$$

When solving the enthalpy, the temperature can also be recovered through these relations. [25]

$$T = \begin{cases} H/c_{P,S} & T < T_s \\ T_s + \frac{H-H_s}{H_l-H_s}\left(T_l - T_s\right) & T_s \leqslant T \leqslant T_l \\ T_l + \left(H - H_l\right)/c_{P,L} & T > T_l \end{cases} \tag{2.19}$$

Here the subscript s signifies the temperature/enthalpy when the substance becomes completely solid. So that is equal to the maximum temperature at which it is solid. And subscript l signifies the minimum temperature for which the material is considered completely fluid, so $f_L = 1$. And from these relations and equation 2.15 a definition of f_L for different phase regimes can also be gathered.

$$f_L = \begin{cases} 0 & \text{if } H < H_s \\ \frac{H-H_s}{H_l-H_s} & \text{if } H_s \leq H < H_l \\ 1 & \text{if } H \geq H_l \end{cases} \tag{2.20}$$

The same can be done for the specific heat and thermal conductivity, which depends on the material phase state. For the mushy state the macroscopic value is determined through a simple first order extrapolation. Which in method differs from previous research. Previous research used either the mean value [16] $c_P = \frac{c_{P,S}+c_{P,L}}{2}$ or the harmonic mean value [27] $c_{P,\mathrm{ref}} = \frac{2c_{P,L}c_{P,S}}{c_{P,L}+c_{P,S}}$. However due to the nature of of the fluid fraction this method resembles better the correct value for the specific heat and thermal conductivity.

$$c_P = \begin{cases} c_{P,S} & \text{if } H < H_s \\ c_{P,S}(1 - f_L) + c_{P,L} f_L & \text{if } H_s \leq H < H_l \\ c_{P,L} & \text{if } H \geq H_l \end{cases} \tag{2.21}$$

$$\lambda = \begin{cases} \lambda_S & \text{if } H < H_s \\ \lambda_S(1 - f_L) + \lambda_L f_L & \text{if } H_s \leq H < H_l \\ \lambda_L & \text{if } H \geq H_l \end{cases} \tag{2.22}$$

An important dimensionless number for melting problems is the Stefan (Ste) number.

$$\mathrm{Ste} = \frac{c_p \Delta T}{L} \tag{2.23}$$

Physically, Ste compares the thermal energy available to raise the temperature of the material to the energy required to convert solid to liquid. Thus:

- A large Ste number implies abundant sensible heat relative to latent heat, leading to a quickly moving melting interface and strong buoyancy forces.
- A small Ste number indicates most energy is consumed by melting rather than heating, producing a thin, slowly evolving melt layer with weak buoyancy forces.

2.3. Lattice Boltzmann Method

The Lattice Boltzmann method is a particle distribution-based method based on the Boltzmann equation instead of conservation equations 2.1 2.2.

2.3.1. Kinetic Theory

The base of the Lattice Boltzmann method lies in kinetic theory. This theory works on a mesoscopic scale instead of a macroscopic scale. Kinetic theory describes the distribution of particles. The theory relies on the use of the distribution function $f(\boldsymbol{x}, \boldsymbol{\xi}, t)$. The distribution function represents the density of particles with velocity $\boldsymbol{\xi}$ at position $\boldsymbol{x}$ at time t. [28]

This mesoscopic variable can be translated to macroscopic variables through:

$$\rho(\boldsymbol{x}, t) = \int f(\boldsymbol{x}, \boldsymbol{\xi}, t)\, \mathrm{d}^3\xi \tag{2.24}$$

$$\rho(\boldsymbol{x}, t)\boldsymbol{u}(\boldsymbol{x}, t) = \int \xi f(\boldsymbol{x}, \boldsymbol{\xi}, t)\, \mathrm{d}^3\xi. \tag{2.25}$$

$$\rho(\boldsymbol{x}, t)E(\boldsymbol{x}, t) = \frac{1}{2}\int |\boldsymbol{\xi}|^2 f(\boldsymbol{x}, \boldsymbol{\xi}, t)\, \mathrm{d}^3\xi. \tag{2.26}$$

In these equations u represents the fluid velocity. Now we can also derive the Boltzmann equation.

$$\frac{\partial f}{\partial t} + \xi_\beta \frac{\partial f}{\partial x_\beta} + \frac{F_\beta}{\rho}\frac{\partial f}{\partial \xi_\beta} = \Omega(f) \tag{2.27}$$

This equation describes the total differential of the distribution function. $\Omega(f) = \frac{df}{dt}$ and is called the collision operator. The first two terms on the left hand side represent the distribution function changing through motion of the particle velocities $\boldsymbol{xi}$. The third term on the left-hand side represents the forces acting on this velocity. [28]

2.3.2. Discretisation and Velocity Sets

By discretizing the Boltzmann equation we end up with the lattice Boltzmann equation:

$$f_i(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t) = f_i(\mathbf{x}, t) + \Delta t\Omega_i(\mathbf{x}, t). \tag{2.28}$$

This discretized version of the lattice Boltzmann equation can be used to solve numerical problems on a square lattice. The distances between the lattice points are defined by the lattice spacing constant: Δx. The time steps are defined by the lattice time step: Δt, more on both these lattice constants and other lattice constants and parameters can be found in Section 3.1.1. Ω denotes the collision operator, this operator represents the collisions between particles that result in a change in the velocity density. Finally $\mathbf{c}$ indicates the discrete velocity vector.

The i in the lattice Boltzmann equation denotes the direction in which the distribution function is pointing and in which direction the 'particles' will travel on the next lattice time step. In some cases there is also a rest velocity which is often denoted by $i = 0$. This rest velocity signifies the particles that have no intention of traveling to any neighboring lattice points.

There are a number of possible dimensions and directions that can be chosen for use of the lattice Boltzmann equation. All the different options are called velocity sets. These are characterized by their dimension and the amount of directions, respectively. So a 1 dimensional 3 directional (1 rest velocity) velocity set is called: D1Q3. Different velocity sets can be used for different kinds of problems. For example: D3Q27 is more computationally taxing than D3Q19, but some truncation terms used in D3Q19 are not rotationally invariant, leading to problems in non-linear regions, i.e. turbulent regions. [29]

Investigating the effect of the angle of incidence is a two-dimensional problem. Because of that a D2Q9 velocity set suffices and is chosen over D3 velocity sets because of computational reasons. In fact, some research regarding fluid flow use a D2Q4 velocity set for computational efficiency. [30] Using this method could be considered for future research if it proves accurate enough, since it would reduce computation times significantly.

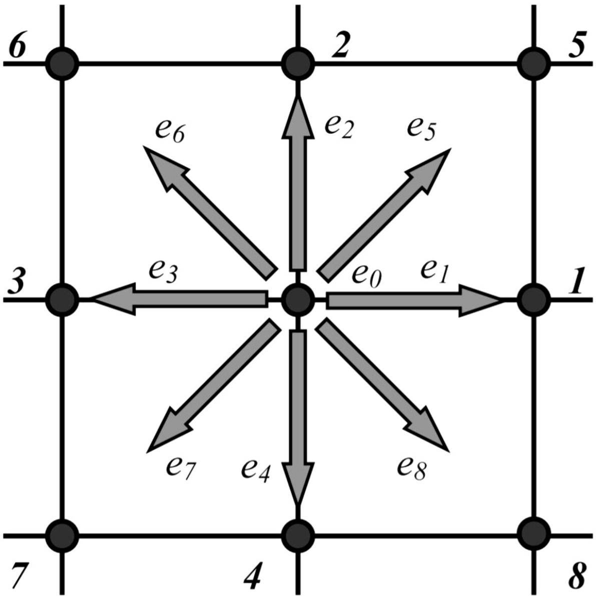


Figure 2.3: Schematic of a D2Q9 velocity set. With e_i denoting the direction of the velocity. [31]

2.3.3. Chapman-Enskog analysis

2.3.4. Double Distribution Function

So far, only the momentum distribution has been taken into account in the LBM. To tackle the heat flow during this project the double distribution function method will be used. This method uses a second distribution function that describes the heat flow. The momentum distribution function f and the heat distribution function g are subsequently coupled in the collision step of the LBM method.

The lattice Boltzmann equation for the heat distribution function also follows the same rules as the lattice Boltzmann equation for the momentum distribution. [32]

$$g_i(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t) = g_i(\mathbf{x}, t) + \Delta t\Omega_i^g(\mathbf{x}, t). \tag{2.29}$$

This can also be done for the enthalpy by creating an enthalpy distribution function called m.

$$m_i(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t) = m_i(\mathbf{x}, t) + \Delta t\Omega_i^m(\mathbf{x}, t). \tag{2.30}$$

The enthalpy and heat distribution functions contain information that can be translated to macroscopic variables as in equations 2.24 2.26 2.25. In a discretized form, the temperature T and the enthalpy H can be retrieved from:

$$T(\mathbf{x}, t) = \sum_i g_i(\mathbf{x}, t) \tag{2.31}$$

$$H(\mathbf{x}, t) = \sum_i m_i(\mathbf{x}, t) \tag{2.32}$$

2.3.5. Advantages and disadvantages LBM

When comparing the LBM method with more traditional methods such as finite volume, there is no one method better than the other. Each method has its own advantages and disadvantages. Here are some of the reasons to choose the LBM method for the research on the freeze plug.

- The heaviest computations of the LBM are local, making some solutions easier and giving the option for parallelisation of the computations. [33] [34]
- LBM is well suited for multiphase and multicomponent flow in complex geometries. Making it well suited for the multiphase flow calculations of the freeze plug.

However, the LBM also comes with some disadvantages when analysing the freeze plug behaviour.

- Adding the ability to simulate compressable flows is not straightforward since this would result in the collision operator having to be rewritten. [33]
- The range of viscosities and densities are somewhat limited in multiphase and multicomponent simulations. [33]

3

Numerical Method

For this thesis the Lattice Boltzmann Method is used for all simulations. How these simulations function and more is discussed in this chapter.

3.1. Lattice units

3.1.1. Conversion factors

The parameters used by the lattice Boltzmann method are in lattice units instead of physical units. Physical units can be related to lattice units through the use of conversion factors and dimensionless numbers.

The conversion factors can be found using the same process that is used in non-dimensionalisation. A quantity can be made non-dimensional by dividing that quantity by a given reference quantity of the same dimension. In the Lattice Boltzmann method the resulting dimensionless factor is called the conversion factor. For length we get:

$$C_l = \frac{\Delta x}{\Delta x^*} \tag{3.1}$$

Here Δx is the ordinary distance between cells of the system with unit $[m]$, Δx^* is the chosen reference distance, and C_l is the resulting length conversion factor. The same can be done for the density conversion factor and the viscosity conversion factor.

$$C_\rho = \frac{\rho}{\rho^*}, \quad C_\nu = \frac{\nu}{\nu^*} \tag{3.2}$$

Having 3 independent conversion factors is enough to generate the dimension of any physical quantity. [28] So with the conversion factors C_l, C_ρ and C_ν any other physical quantity can be transformed into lattice units. The other conversion factors that are useful are the following. In this simulation the temperature is not set to Lattice units and thus $C_T = 1$

$$C_t = \frac{C_l^2}{C_\nu}, \quad C_\kappa = C_\nu, \quad C_\lambda = \frac{C_\nu^3}{C_l^5 C_T}, \quad C_{Cp} = \frac{C_\nu^2}{C_l^2 C_T}, \quad C_g = \frac{C_\nu^2}{C_l^3}, \quad C_L = \frac{C_l^2}{C_\nu} \tag{3.3}$$

Where t stands for time κ is the heat capacity, λ is the thermal conductivity, Cp is the thermal diffusivity, g is the gravitational acceleration and L is the latent heat. In the LBM C_l is also defined by $C_l = L_x/N_x$, where L_x is the physical length of the system in x and N_x is the amount of cells used on the x axis.

It is standard in the LBM to set Δx^* and $\Delta\rho^*$ to 1. This leaves 2 free parameters that can be chosen. System size N_x and reference viscosity ν^*.

3.1.2. Choice of free parameters

A typical stability restraint in CFD (Computational Fluid Dynamics) is the Courant number. [35]

$$C = |\boldsymbol{u}|\frac{\Delta t}{\Delta x} \tag{3.4}$$

From a numerical viewpoint, the time integration schemes can be broadly divided into three groups: explicit, implicit and semi-implicit. Roughly speaking, in explicit schemes the solution at the time step n+1 is calculated using the known values of the solution at time n. Conversely, in an implicit scheme, the solution at time n+1 is calculated using unknown values of the solution at time n+1, hence requiring the solution of an ODE. Lastly, semi-implicit schemes use both the solution at time n and n+1. [36] The method used in this thesis is explicit as can be seen at the end of the previous chapter. For explicit schemes, the CourantFriedrichsLewy (CFL) condition applies. [36]

$$\mathrm{Co} = \frac{\mathrm{u}\Delta \mathrm{t}}{\Delta \mathrm{x}} \leq \mathrm{Co}_{\mathrm{max}} \tag{3.5}$$

The Courant number is so useful since it both contains the physical speed $\boldsymbol{u}|$ at which the fluid is advected and the speed at which the information passes through the simulation $\frac{\Delta x}{\Delta t}$. If the physical speed is greater than the information carry speed the simulation becomes unstable. Because of this it is standard practice to keep $C \leq 1$. Since the Δx in equation 3.4 is the free parameter Δx^* in the LBM this limits how small Δx^* can be based on the max velocity of the simulation.

Something that is closely related to the free lattice parameters Δx^* and ν^* is the grid Reynolds number Re_g. This is defined as:

$$\mathrm{Re}_g = \frac{U^*_{max}\Delta x^*}{\nu^*} \tag{3.6}$$

For the BGK operator method parameters that are used are the relaxation parameter τ and the lattice speed of sound c_s. The kinematic viscosity can be translated into these parameters by the following equation.

$$\nu = c_s^2(\tau - \Delta t/2) \tag{3.7}$$

In the BGK method a necessary condition is $\frac{\tau}{\Delta t} \geq \frac{1}{2}$. This is due to the way the lattice Boltzmann Equation solves the Navier Stokes equations, and otherwise the BGK method causes negative viscosities resulting in instability.[28] This is not directly useful information for the stability of the FMLBM, but it can be used in combination with the grid Reynolds number.

$$\mathrm{Re}_g = \frac{U^\star_{\mathrm{max}}\,\Delta x^\star}{\nu^\star} = \frac{U^\star_{\mathrm{max}}}{c_s^{\star 2}\left(\tau^\star - \frac{1}{2}\right)} \quad \Longrightarrow \quad \tau^\star = \frac{1}{2} + \frac{U^\star_{\mathrm{max}}}{c_s^{\star 2}\mathrm{Re}_g}. \tag{3.8}$$

Combining this with the relaxation time condition it can be deduced that the Re_g should be of $\mathcal{O}(10)$ or smaller.[28] The physical interpretation of this is the following: This also gives more restrictions to the free parameters Δx^* and ν^* dependent on U^*_{max}

3.2. FMLBM Collision

3.2.1. Collision operators

In 2.3 a basis for the LBM is introduced. The collision operator inside the LBE can be defined in many different ways. The BhatnagarGrossKrook collision operator (BGK)[37] proposed by Bhatnagar, Gross and Krook is one of the simpler operators.

$$\Omega_{BGK,i}\left(f_i\right) = -\tau^{-1}\left(f_i - f_i^{eq}\right) \tag{3.9}$$

Here τ is the relaxation time, and f_i^{eq} is the Maxwell-Boltzmann distribution of $\mathbf{c}_i$ for a given macroscopic velocity $\mathbf{u}$ and density ρ [38]. Physically this can be interpreted as the tendency of the distribution f_i to reach it's equilibrium state f_i^{eq} in time τ. [28] Inputting this definition of the collision operator in equation 2.27 gives the following result which is called the lattice BGK equation.

$$f_i\left(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t\right) = f_i(\boldsymbol{x}, t) - \frac{\Delta t}{\tau}\left(f_i(\boldsymbol{x}, t) - f_i^{\mathrm{eq}}(\boldsymbol{x}, t)\right) \tag{3.10}$$

. Even though the BGK operator is quite simple it works well. With this operator the LBE is able to reproduce the continuity and Navier-Stokes equations. The main drawbacks of the BGK operator are accuracy and stability issues. There are other operators that try to tackle these issues with more than one relaxation time, these are called two-relaxation time and multiple-relaxation-time operators. The basic idea behind a multiple-relaxation time operator is that there are multiple relaxation times which combine with a certain transformation matrix.

$$\boldsymbol{\Omega}_{MRT}(f) = -\left(\mathbf{M}^{-1}\mathbf{S}\mathbf{M}\right)(\mathbf{f} - \mathbf{f}^{eq}) \tag{3.11}$$

Where $\mathbf{S} = \mathrm{diag}\left(\tau_\rho, \tau_e, \tau_\epsilon, \tau_j, \tau_q, \tau_j, \tau_q, \tau_s, \tau_s\right)^{-1}$, and $\mathbf{M}$ is just an orthogonal transform matrix. Use of an MRT operator results in more stability and accuracy due to more degrees of freedom. [39]

The collision operator for the filter matrix method looks somewhat different, since it does not rely on the equilibrium distribution directly.

$$\Omega_{FM,i} = \rho\omega_i \left(\frac{(c_{i\alpha}\partial_\alpha)\,(c_{i\beta}u_\beta)}{c_s^2} - \partial_\alpha u_\alpha \right) \tag{3.12}$$

.

The lattice Boltzmann equation using this collision operator can be performed using a staggered, explicit, first-order scheme. [40]

$$\begin{aligned} f_i\left(x_\alpha \pm c_{i\alpha}\tfrac{\Delta t}{2}, t \pm \tfrac{\Delta t}{2}\right) &= f_i\left(x_\alpha, t\right) \pm \tfrac{1}{2}\Delta t \Omega_i\left(f_i\right) + \mathcal{O}\left(\Delta t^2\right) = \\ \rho\omega_i &\left(1 + \tfrac{c_{i\alpha}u_\alpha}{c_s^2} + \tfrac{(c_{i\alpha}u_\alpha)(c_{i\beta}u_\beta)}{2c_s^4} - \tfrac{u_\alpha^2}{2c_s^2} + \left(\tfrac{-\nu}{c_s^2} \pm \tfrac{\Delta t}{2}\right)\left[\tfrac{(c_{i\alpha}\partial_\alpha)(c_{i\beta}u_\beta)}{c_s^2} - \partial_\alpha u_\alpha\right]\right) + \mathcal{O}\left(\Delta t^2\right) \end{aligned} \tag{3.13}$$

This can (and should) be more concisely written as

$$f_i\left(x_\alpha \pm c_{i\alpha}\frac{\Delta t}{2}, t \pm \frac{\Delta t}{2}\right) = \omega_i E_{ik} \alpha_k^\pm\left(x_\alpha, t\right) \tag{3.14}$$

ω_i is a weights vector, E_{ik} is the filter matrix and $\alpha_k^\pm\left(x_\alpha, t\right)$ is the solution vector. The weights vector is defined as:

$$\omega = [\frac{4}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}], \tag{3.15}$$

for a D2Q9 scheme. The filter matrix and the solution vector are distribution scheme dependent and will be further derived below.

3.2.2. Momentum

Equation 3.14 is the basis for the use of the FMLBM in momentum and density simulations. The filter matrix is square and reversible. Because of this the solution vector can be gained from

$$\alpha_k^\pm(\mathbf{x}, t) = \sum_i E_{ki} f_i\left(\mathbf{x} \pm \frac{\mathbf{c}_i\Delta t}{2}, t \pm \frac{\Delta t}{2}\right) \tag{3.16}$$

For a D2Q9 scheme the solution vector results in

$$\alpha_k^\pm(\mathbf{x}_\alpha, t) = \begin{bmatrix} \rho \\ \rho u_x \pm \frac{\Delta t}{2} f_x \\ \rho u_y \pm \frac{\Delta t}{2} f_y \\ 3\rho u_x u_x + \rho(-6\nu \pm \Delta t)\frac{\partial u_x}{\partial x} \\ 3\rho u_x u_y + \rho\frac{-6\nu\pm\Delta t}{2}\left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}\right) \\ 3\rho u_x u_y + \rho(-6\nu \pm \Delta t)\frac{\partial u_y}{\partial y} \\ T_1^\pm \\ T_2^\pm \\ F^\pm \end{bmatrix} \tag{3.17}$$

Here the ± sign indicates whether the solution vector is from the preceding timestep $(-)$ or from the subsequent timestep $(+)$. The last three terms $T_1^\pm$, $T_2^\pm$ and $F^\pm$ are higher order terms that originate from the $\mathcal{O}\left(\Delta t^2\right)$ term and are thus unphysical. These terms can be neglected and in calculations are set to 0. The filter matrix is defined as

$$E_{ik} = \begin{bmatrix} 1 \\ 3c_{ix} \\ 3c_{iy} \\ \frac{3c_{ix}^2-1}{2} \\ 3c_{ix}c_{iy} \\ \frac{3c_{iy}^2-1}{2} \\ \frac{3c_{ix}(3c_{iy}^2-1)}{2} \\ \frac{3c_{iy}(3c_{ix}^2-1)}{2} \\ \frac{(3c_{ix}^2-1)(3c_{iy}^2-1)}{2} \end{bmatrix} \tag{3.18}$$

In practice the algorithm to perform timesteps is as follows:

1. Determine $\alpha_k^-(\mathbf{x},t)$ through use of the inverse filter matrix.

$$\alpha_k^-\left(\mathbf{x}_\alpha,t\right) = \left(\omega_i E_{ik}\right)^{-1} f_i\left(\mathbf{x}_\alpha - c_{i\alpha}\frac{\Delta t}{2}, t - \frac{\Delta t}{2}\right) \tag{3.19}$$

2. Through $\alpha_k^-(\mathbf{x},t)$ it is easy to obtain $\alpha_k^+(\mathbf{x},t)$ through use of equation 3.17 and changing the necessary - signs to + signs.
3. And by using the obtained $\alpha_k^+(\mathbf{x},t)$ the momentum distribution function for the new timestep can be gained.

$$f_i\left(x_\alpha + c_{i\alpha}\frac{\Delta t}{2}, t + \frac{\Delta t}{2}\right) = \omega_i E_{ik}\alpha_k^+\left(x_\alpha,t\right) \tag{3.20}$$

3.2.3. Temperature

The lattice Boltzmann equation for the temperature is very similar to the lattice Boltzmann equation for momentum. The differences are the use of the heat distribution function $g_i(\mathbf{x},t)$ and the accompanied change in solution vector. So now the Filter Matrix Lattice Boltzmann equation looks like this

$$g_i\left(x_\alpha \pm c_{i\alpha}\frac{\Delta t}{2}, t \pm \frac{\Delta t}{2}\right) = \omega_i E_{ik}\beta_k^\pm\left(x_\alpha,t\right) \tag{3.21}$$

Now $\beta_k^\pm\left(x_\alpha,t\right)$ represents the solution vector and can be obtained similarly through

$$\beta_k^\pm(\mathbf{x},t) = \sum_i E_{ki} g_i\left(\mathbf{x} \pm \frac{\mathbf{c}_i\Delta t}{2}, t \pm \frac{\Delta t}{2}\right) \tag{3.22}$$

In vector form this ends up being equivalent to [41]

$$\beta_k^\pm(\mathbf{x}_\alpha,t) = \begin{bmatrix} T \\ Tu_x + \frac{-6\kappa\pm\Delta t}{6}\frac{\partial T}{\partial x} \\ Tu_y + \frac{-6\kappa\pm\Delta t}{6}\frac{\partial T}{\partial y} \\ S_1^\pm \\ S_2^\pm \\ S_3^\pm \\ T_1^\pm \\ T_2^\pm \\ F^\pm \end{bmatrix} \tag{3.23}$$

$S_{1,2,3}$, $T_{1,2}$ and F can once again be neglected and set to 0 in simulations since they are second-, third-, and fourth-order terms respectively. The filter matrix remains the same for temperature since the same D2Q9 scheme is used.

3.2.4. Enthalpy

The enthalpy variation is slightly different. First we divide the total Enthalpy into 2 parts

$$H = c_P T + f_L L \tag{3.24}$$

a sensible part $C_P T$ and a latent part $f_L L$.

The enthalpy distribution function is also divided into a sensible and latent part.

$$m_i = m_i^s + m_i^l \tag{3.25}$$

The latent part only has a stationary component which is equal to $f_L L$. This is under the assumption that the latent heat does not diffuse (it does not disperse energy to it's surroundings). [42]

For m_i^s the same process is applied that was used for the temperature and momentum distributions.

$$m_i^s \left(x_\alpha \pm c_{i\alpha} \frac{\Delta t}{2}, t \pm \frac{\Delta t}{2} \right) = \omega_i E_{ik} \gamma_k^{\pm} \left(x_\alpha, t \right) \tag{3.26}$$

And the new solution vector for the sensible enthalpy$\gamma_k^{\pm} \left(x_\alpha, t \right)$ can be found in the same manner as well.

$$\gamma_k^{\pm}(\mathbf{x}_\alpha, t) = \begin{bmatrix} h \\ hu_x + \frac{-6\kappa \pm \Delta t}{6} \frac{\partial h}{\partial x} \\ hu_y + \frac{-6\kappa \pm \Delta t}{6} \frac{\partial h}{\partial y} \\ S_1^{\pm} \\ S_2^{\pm} \\ S_3^{\pm} \\ T_1^{\pm} \\ T_2^{\pm} \\ F^{\pm} \end{bmatrix} \tag{3.27}$$

And once again all unphysical higher order terms are negligible and set to 0 in calculations.

3.3. Streaming

The collision step of the Lattice Boltzmann Method is the step where the collision operator is alters the distribution vector on the current cell. In formula form this looks like this

$$f_i^* \left(x_\alpha, t \right) = f_i(x_\alpha, t) + \Delta t \Omega_i^g (x_\alpha, t) \tag{3.28}$$

To change the distribution function values on other cells another step is required

$$f \left(x_\alpha + c_{i\alpha} \Delta t, t + \Delta t \right) = f_i^* \left(x_\alpha, t \right) \tag{3.29}$$

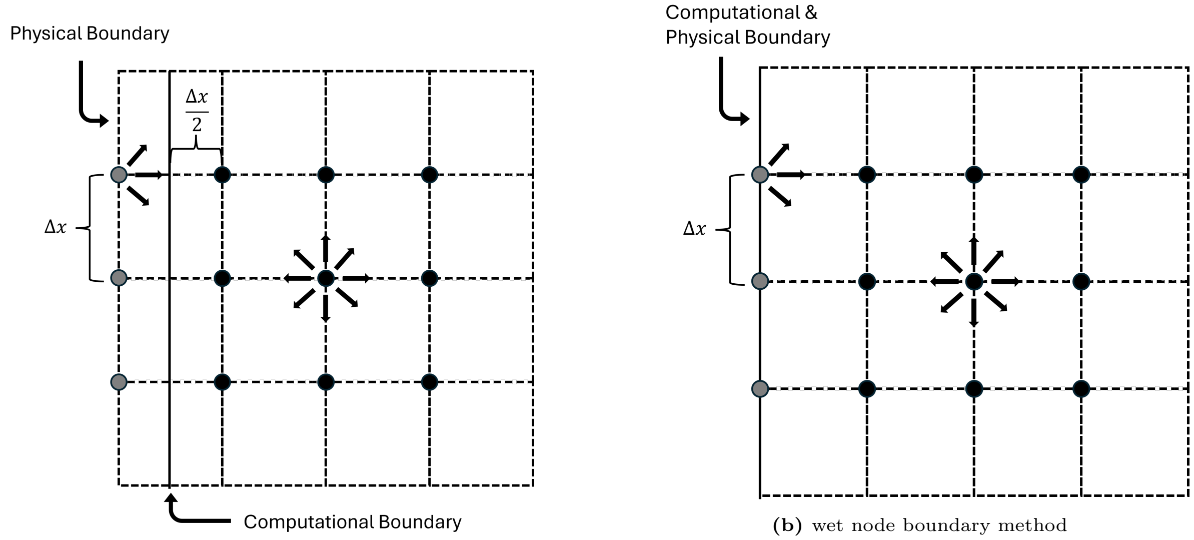
So practically speaking the act of streaming is the act of copying the memory content of $f_i^* \left(x_\alpha, t \right)$ to the lattice site $x_\alpha + c_{i\alpha} \frac{\Delta t}{2}$. One thing to be careful of when rewriting the memory at the new lattice site is to no change memory that is still needed in the other streaming steps. For example, if the memory at site $(3, 3)$ is changed from site $(3, 2)$ then the data given by site $(3, 3)$ onward might be faulty. There are ways to tackle this problem, one might two data sets while streaming, one for reading data and one for writing the new data.

3.3.1. Opposite Streaming direction

The implementation that is used is to take the 3d distribution array `g[d,i,j]`. Here d stands for the velocity direction and i and j are the coordinates of the lattice cell or α. For each lattice direction $d \in \{2, \ldots, 9\}$, and for each cell (i, j) , the destination location $(i_{\text{dest}}, j_{\text{dest}})$ is computed and a source location $(i_{\text{src}}, j_{\text{src}})$ is computed through use of the following index calculators.

$$i_1^{(\mathrm{p})} = N_x + 1 - i, \quad i_2^{(\mathrm{p})} = i_1^{(\mathrm{p})} - 1, \quad i_1^{(\mathrm{n})} = i, \quad i_2^{(\mathrm{n})} = i + 1,$$

and



(a) link wise boundary method.

(b) wet node boundary method

Figure 3.1: Schematic of both boundary methods. The computational boundary of the link wise method is placed at a distance of $\frac{\Delta x}{2}$ from the Physical boundary.

$$j_1^{(\mathrm{p})} = N_y + 1 - j, \quad j_2^{(\mathrm{p})} = j_1^{(\mathrm{p})} - 1, \quad j_1^{(\mathrm{n})} = j, \quad j_2^{(\mathrm{n})} = j + 1,$$

For each direction, the destination and source indices are chosen according to the direction of propagation. For example, in direction $d = 2$ (eastward), the destination index is (i_1^{p}, j) and the source index is (i_2^{p}, j). Similarly, for $d = 3$ (westward), the destination is (i_1^{n}, j) and the source is (i_2^{n}, j). In diagonal directions like $d = 6$ (northeast), we write:

$$g[6, i_1^{\mathrm{p}}, j_1^{\mathrm{p}}] = g[6, i_2^{\mathrm{p}}, j_2^{\mathrm{p}}],$$

where the source cell lies ahead in both i and j directions.

Because the destination indices mirror the grid along both axes (e.g., $i_{\mathrm{dest}} = x_2 + 2 - i$), and because the outer loops increment i and j from the lower to upper bounds, each write operation targets a memory location that has not yet been visited by the loop. Simultaneously, the source cell lies ahead in the loop order and thus still contains the unmodified data from the previous timestep. This ensures that no cell is read after it has been overwritten, enabling correct and efficient in-place propagation without the need for temporary storage. This approach is conceptually similar to the *Esoteric Twist* algorithm [43] which uses mirrored indexing and loop reordering to achieve conflict-free streaming in a single pass.

3.4. Boundary Conditions

There are 2 different main methods of applying boundary conditions for the FMLBM. There are link wise methods and wet node methods.

3.4.1. Bounce back methods

The prime example of link-wise boundary conditions are bounce back methods.

The principle of bounce back methods is that populations hitting a solid wall are reflected back to where they originally came from. This would work for all three relevant distributions. The bounce back implies no flux across boundaries since no populations get through the boundary. It also implies a no slip boundary since there is no relative transverse motion.

Halfway bounce back (HBB) is implemented through the following rule at the boundary

$$f_i\left(x, t\right) = f_j^{\star}\left(x, t + \Delta t\right) \tag{3.30}$$

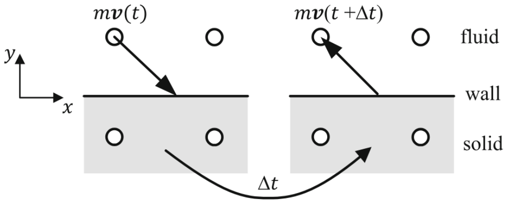


Figure 3.2: The physical representation of a halfway bounce back boundary interaction. During the collision process both normal and tangential momentum components are reversed. The average particle momentum, before and after collision, is $\langle mv \rangle = (mv(t) + mv(t+\Delta t))/2 = \mathbf{0}$ [28]

Where i notates the opposite directional index of j. Since the difference between f_i and $f_j^\star$ is Δt, the bounce occurs at $t + \frac{\Delta t}{2}$ at the computational boundary.

Another popular bounce back method is the full way bounce back method (FBB). The full bounce back method changes the boundary rule from HBB slightly to

$$f_i\left(x, t - \Delta t\right) = f_j^\star\left(x, t + \Delta t\right) \tag{3.31}$$

Between $t - \Delta t$ and $t + \Delta t$ (so at t) the information aboutthe distribution is stored in a solid node beyond the computational boundary. This method thus requires nodes that are set slightly in the solid domain.

Neither of these bounce back methods were used to generate the results of this thesis. However, these methods were used to verify the working of the filter matrix boundary method.

3.4.2. Filter matrix boundary method

The wet node boundary technique is a little different. Instead of the physical boundary being outside of the computational domain, it is instead the last node inside the computational domain which lies on the physical boundary. [44] The main idea of the wet node approach is to guesstimate what the distribution function of the boundary wet node should be. This estimation is made using the surrounding non boundary nodes and extrapolating and by setting specific rules depending on what type of oundary condition is present. There are 3 different types of boundary conditions that are used in this thesis, a Dirichlet boundary condition, a Neumann boundary condition and a conjugated boundary condition.

Now for this thesis a specific wet node boundary method is used that was originally proposed by M. Rohde. [40] The proposed method integrates the filter matrix method with a wet node boundary method to create the filter matrix boundary method, or lattice boltzmann filter matrix boundary method (LBFMBM) for long. The filter matrix method uses solution vectors from which macroscopic quantities such as temperature and density can be easily translated. However, these macroscopic quantities can also be easily translated back into the solution vectors. This makes setting boundary conditions using the filter matrix boundary method more straightforward.

The following are the steps to perform the filter matrix boundary method for the temperature field.

Step 1 Estimate the solution vector at the boundary. This is done using a first order spatial extrapolation.

$$(\beta_k^{i=0})_{extr} = 2\beta_k^{i=1} - \beta_k^{i=2} \tag{3.32}$$

Previous research done by L. Borstlap investigated the accuracy of other orders of spatial extrapolation as well as temporal extrapolation. The results found by L. Borstlap showed inaccurate non converging results for 0th order spatial extrapolation, $(\beta_k^{i=0})_{extr} = \beta_k^{i=1}$, and first order temporal extrapolation, $(\beta_k^{i=0,t})_{extr} = \beta_k^{i=1,t} - \beta_k^{i=1,t-1} + \beta_k^{i=0,t-1}$. It also showed little difference in result accuracy between first order spatial extrapolation and higher orders of extrapolation. [16]

Step 2 Determine the temperature at the boundary according to the boundary condition. *Dirichlet* For the Dirichlet boundary condition the temperature at the boundary is a set temperature.

$$T_{BC} = T_{Dir} \tag{3.33}$$

Neumann The Neumann boundary condition sets the flux through the boundary to a known value. In the case of this thesis all Neumann boundaries are adiabatic boundaries.

$$\frac{dT}{dx_i} = 0 \tag{3.34}$$

At the boundary a 2nd order extrapolation is used to determine the temperature at the boundary and transform the Neumann boundary condition into a Dirichlet boundary condition.

$$T_{BC} = \frac{4}{3}\beta^{i=1} - \frac{1}{3}\beta^{i=2} \tag{3.35}$$

Step 3 Set up the final estimation of the solution vector.

$$\beta_k^{i=0} = \left(\beta_k^{i=0}\right)_{\text{extrp}} - \begin{bmatrix} T_{\text{extrp}} \\ u_x T_{\text{extrp}} \\ u_y T_{\text{extrp}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} T_{BC} \\ u_x T_{BC} \\ u_y T_{BC} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.36}$$

The initial estimation of $\left(\beta_k^{i=0}\right)_{\text{extrp}}$ is adjusted with the T_{BC} found using the Boundary conditions. T_{extr} is defined as $\left(\beta_0^{i=0}\right)_{\text{extrp}}$ and represents the initial guess of the temperature based on extrapolation.

Step 4 Finally for the collision and streaming steps the distribution functions at the boundary are needed, not the solution vectors. So the solution vectors are translated back to the distribution functions using the FM matrix multiplication.

$$g_i(x=0,t) = \sum_k \omega_i E_{ik} \beta_k^{i=0} \tag{3.37}$$

These steps work similar for the density distribution and the enthalpy distribution.

Changes for density distribution The most important difference is the difference in boundary condition. All boundaries are assumed to be no-slip boundaries. This can be assumed since the Rayleigh number will be kept small enough so that all flow is laminair during the simulations.

Step 1 Remains unchanged.

$$(\alpha_k^{i=0})_{extr} = 2\alpha_k^{i=1} - \alpha_k^{i=2} \tag{3.38}$$

Step 2 Now the boundary condition that is applied states:

$$u_{iBC} = 0 \tag{3.39}$$

Where i is the direction parallel to the boundary, signifying a no-slip boundary.

Step 3 The operation performed is similar to the temperature case, it only looks slightly different since the form of the solution vector is different.

$$\left(\alpha_k^{i=0}\right)_{BC} = \left(\alpha_k^{i=0}\right)_{extrp} - \begin{bmatrix} \rho \\ \rho\left(u_x\right)_{extrp} \\ \rho\left(u_y\right)_{extrp} \\ 3\rho\left(u_x^2\right)_{extrp} \\ 3\rho\left(u_x u_y\right)_{extrp} \\ \rho\left(u_y^2\right)_{extrp} \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \rho \\ \left(\rho u_x\right)_{BC} \\ \rho\left(u_y\right)_{BC} \\ 3\rho\left(u_x^2\right)_{BC} \\ 3\rho\left(u_x u_y\right)_{BC} \\ \rho\left(u_y^2\right)_{BC} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.40}$$

Step 4 Step 4 remains unchanged.

$$f_i(x=0,t) = \sum_k \omega_i E_{ik} \alpha_k^{i=0} \tag{3.41}$$

Changes for enthalpy distribution The most important difference when handling the boundary for the enthalpy case is that the Dirichlet/Neumann boundary temperature needs to be translated into Enthalpy.

Step 1 Remains unchanged.

$$(\gamma_k^{i=0})_{extr} = 2\gamma_k^{i=1} - \gamma_k^{i=2} \tag{3.42}$$

Step 2 First steps are the same to determine T_{BC}, then this is translated into H_{BC}.

$$H_{BC} = \begin{cases} c_{p,s}T_{BC} & \text{if } T_{BC} \leq T_s \\ c_{p,s}T_s + f_L^{i=0}L & \text{if } T_s < T_{BC} < T_l \\ c_{p,s}T_s + c_{p,l}\,(T_{BC} - T_l) + L & \text{if } T_{BC} \geq T_l \end{cases} \tag{3.43}$$

Step 3 Remains unchanged.

$$\gamma_k^{i=0} = \left(\gamma_k^{i=0}\right)_{\text{extrp}} - \begin{bmatrix} H_{\text{extrp}} \\ u_x H_{\text{extrp}} \\ u_y H_{\text{extrp}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} H_{BC} \\ u_x H_{BC} \\ u_y H_{BC} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.44}$$

Step 4 Remains unchanged.

$$m_i(x=0,t) = \sum_k \omega_i E_{ik} \gamma_k^{i=0} \tag{3.45}$$

3.4.3. Conjugate boundary

At the conjugate boundary the boundary condition also needs to be translated into a Dirichlet boundary condition. There can also be 2 relevant domains carrying different information. The PCM (phase change material) domain which uses enthalpy distribution functions, and the solid wall domain which uses temperature distribution functions. For one benchmark case handled in section 4.2 both domains use the temperature distribution function. All said below can still be applied there with less steps, so this case is not handled seperately.

The start point are equations 2.12 and 2.10. These need to be translated into a Dirichlet boundary condition.

For the derivation the fluidsolid interface (wall) is located at $x = 0$. The fluid PCM domain occupies $x > 0$, the solid domain occupies $x < 0$.

First, a first-order one-sided difference on each side is used:

$$\left.\frac{\partial T}{\partial x}\right|_{0+} \approx \frac{\beta_1^{1,l} - T_{BC}}{\Delta x}. \tag{3.46}$$

$$\left.\frac{\partial T}{\partial x}\right|_{0-} \approx \frac{T_{BC} - \beta_1^{1,s}}{\Delta x}. \tag{3.47}$$

By then imposing.

$$\frac{\lambda_l(\beta_1^{1,l} - T_{BC})}{\Delta x} = \frac{\lambda_s(T_0 - \beta_1^{1,s})}{\Delta x} \tag{3.48}$$

The first order interface temperature is found.

$$T_{BC} \;=\; T_0 \;=\; \frac{\lambda_l\,\beta_1^{1,l} \;+\; \lambda_s\,\beta_1^{1,s}}{\lambda_l + \lambda_s}. \tag{3.49}$$

For second order this derivation comes out to

$$\left.\frac{\partial T}{\partial x}\right|_{0+} \approx \frac{-3T_{BC} + 4\beta_1^{1,l} - \beta_1^{2,l}}{2\Delta x}. \tag{3.50}$$

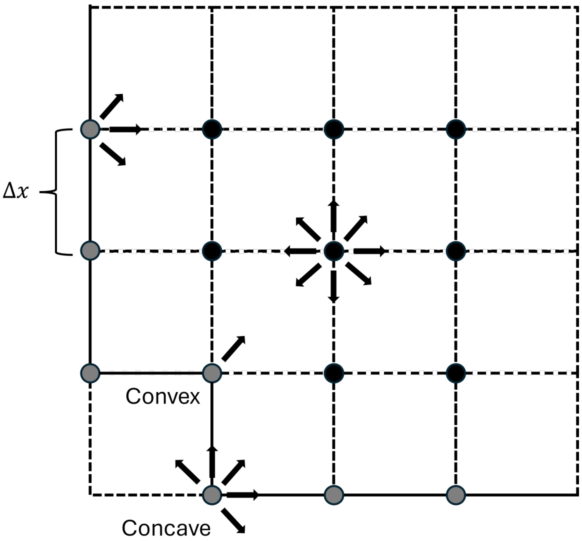


Figure 3.3: A schematic showing both convex and concave corner cells in a LBM simulation. The arrows signify unknown density populations when using the NEBB method.

$$\left.\frac{\partial T}{\partial x}\right|_{0^-} \approx \frac{3T_0 - 4\beta_1^{1,s} + \beta_1^{2,s}}{2\Delta x}. \tag{3.51}$$

$$T_{BC} = \frac{\lambda_l(4\beta_1^{1,l} - \beta_1^{2,l}) + \lambda_s(4\beta_1^{1,s} - \beta_1^{2,s})}{3(\lambda_l + \lambda_s)} \tag{3.52}$$

For the PCM domain, the enthalpy solution vectors can be translated to the temperature solution vectors using the relations from equation 2.19.

$$\beta_1^i = \begin{cases} \gamma_1^i / C_p & \gamma_1^i < H_s \\ T_s + \frac{\gamma_1^i - H_s}{H_l - H_s}\left(T_l - T_s\right) & H_s \leqslant \gamma_1^i \leqslant H_l \\ T_l + \left(\gamma_1^i - H_l\right) / C_p & H > T_l \end{cases} \tag{3.53}$$

3.4.4. Corners

In LBM methods corners can be considered a special case. [45] For Bounce back methods, no special treatment is necessary and both convex and concave corners can be treated as straight walls.

A more developed wet node approach is called the non-equilibrium bounce-back (NEBB). This method has to treat corners different from straight walls, and also discriminates between concave and convex corners. [45] In short, the concave corner has extra unknown populations compared to straight walls. Therefore the corner needs extra constraints. The issue surrounding corners becomes more complex when handling more dimensions and directions. [46]

Regarding the Filter Matrix method there has not yet been a lot of research done on how corners should be handled. There are 2 key points to consider for the FM method.

Connecting seperate boundary conditions At a corner point it is possible for that cell to be subjected to two seperate boundary conditions. An example that will come up is a corner that is the connecting point of a Neumann BC and a Dirichlet BC.

Extrapolation direction corners When extrapolating from a straight wall to form an initial guess of the solution vector, it is clear what direction needs to be extrapolated in. When dealing with convex corners there are 3 possible directions that could be considered viable. There are the 2 orthogonal choices (directions $\pm[1,0]$ and $\pm[0,1]$), and there is the diagonal choice (directions $[\pm1,\pm1]$).

3.4.5. Phase front

During each time step the melting front should have its own no slip boundary conditions since a no slip interface is also assumed there. A rudimentary method of accomplishing this would be to track the melting front and apply a no slip HBB BC. This is considered a computationally heavy method compared to the more elegant enthalpy-porosity method. The enthalpy-porosity method uses a friction component S to simulate the no slip boundary. The term S, represents an equivalent frictional resistance force per unit mass, which originates from the consideration that the morphology of the phase-changing domain can be treated as an equivalent porous medium that offers a frictional resistance towards fluid flow in that region. Hence it is called the enthalpy porosity method. This friction term can be introduced as a Darcy momentum sink in the Navier-Stokes equation. [24]

$$\rho\left(\frac{\partial u_{x,y}}{\partial t}+u_{x,y}\nabla\mathbf{u}\right)=-\frac{\partial P}{\partial x,y}+\rho\nabla\left(\nu\nabla u_{x,y}\right)-S_{x,y}u_{x,y} \tag{3.54}$$

The term S is modeled as

$$S_{x,y}=A_{mush}\frac{\left(1-f_L\right)^2}{f_L^3+\sigma_1} \tag{3.55}$$

Here A_{mush} is a morphological constant set to $10^8\,\frac{\mathrm{kg}}{\mathrm{m\ K\ s}}$ and σ_1 is a very small computational constant introduced to avoid division by zero. This method was successfully adapted for FMLBM use by C. Bus.[27] This was done by modifying the solution vector for density distributions.

$$\alpha^{\pm}_{1,2}=\left(\rho\mp\frac{\Delta t}{2}S_{x,y}\right)u_{x,y}\pm\frac{\Delta t}{2}f_{x,y} \tag{3.56}$$

C. Bus also redefined how to find the velocities when using the enthalpy-porosity method.[27]

$$u_{x,y}=\frac{\alpha^{-}_{1,2}+\frac{\Delta t}{2}f_{x,y}}{\rho+\frac{\Delta t}{2}S_{x,y}} \tag{3.57}$$

3.5. Algorithm

This section will explain which steps the algorithm takes to compute a melting simulation. The algorithm will be for the Pater & Kaaks study [47]. This case handles everything discussed prior. A slightly simpler version of the algorithm that just handles temperature and momentum distributions can be found in the appendix. It handles a simpler benchmark case based on studies from de Vahl Davis [48] and Zhuo & Zong.[41]

- Initialize the distribution functions in the PCM domain for density $f_i(x,t)$ and enthalpy $m_i(x,t)$ using chosen macroscopic quantities.
- Initialize the distribution functions in the Wall domain for density $f_i(x,t)$ and temperature $g_i(x,t)$ using chosen macroscopic quantities.
- Initialize enthalpy properties such as the liquid fraction f_L and the total enthalpy of all cells using $m_i(x,t)$.
- For each time step `t` from `0` to `T`, do:
    1. **Collision step PCM domain**
        - Substract latent enthalpy $(f_L(x,t)*L)$ from $m_i(x,t)$
        - Determine solution vector $\gamma_i^-(x,t)$ using equation 3.19 and $m_i(x,t)$
        - Translate $\gamma_i^-(x,t)$ into $\gamma_i^+(x,t)$ 3.27
        - Get the post collision enthalpy distributions from $\gamma_i^+(x,t)$ 3.26
        - re apply latent heat to $m_i(x,t)$
    2. **Propogate** $m_i(x,t)$
    3. **Update enthalpy properties**
        - Find total enthalpy in all cells 2.32
        - Find temperature in the PCM domain 2.19
        - Determine liquid fraction 2.20
        - Determine specific heat 2.21
        - Determine thermal conductivity 2.22
    4. **Collision step Wall domain**
        - Determine solution vector $\beta_i^-(x,t)$ using equation 3.19 and $g_i(x,t)$
        - Translate $\beta_i^-(x,t)$ into $\beta_i^+(x,t)$ 3.23

   - Get the post collision enthalpy distributions from $\beta_i^+(x,t)$ 3.22

5. **Propogate** $g_i(x,t)$
6. **Apply enthalpy Neumann boundary condition edges PCM domain** (See steps from section 3.4.2
7. **Apply enthalpy Dirichlet boundary condition edge PCM domain**
8. **Apply enthalpy boundary conditions to corners on the PCM domain**
9. **Apply temperature Neumann boundary condition edges Wall domain**
10. **Apply temperature boundary conditions to corners on the Wall domain**
11. **Update enthalpy properties**
    - Find total enthalpy in all cells 2.32
    - Find temperature in the PCM domain 2.19
    - Determine liquid fraction 2.20
    - Determine specific heat 2.21
    - Determine thermal conductivity 2.22
12. **Apply conjugate boundary condition on interface edge as specified in section 3.4.3**
13. **Apply conjugate boundary condition on corners**
14. **Update enthalpy properties**
    - Find total enthalpy in all cells 2.32
    - Find temperature in the PCM domain 2.19
    - Determine liquid fraction 2.20
    - Determine specific heat 2.21
    - Determine thermal conductivity 2.22
15. **Collision step PCM domain for density distributions**
    - Determine friction component $S(x,t)$ 3.55
    - Determine solution vector $\alpha_i^-(x,t)$ using equation 3.19 and $m_i(x,t)$
    - Determine acting body forces due to natural convection 2.7
    - Determine velocities 3.57
    - Translate $\alpha_i^-(x,t)$ into $\alpha_i^+(x,t)$ 3.56
    - get the post collision enthalpy distributions from $\alpha_i^+(x,t)$ 3.20
16. **Apply density boundary conditions to edges on the PCM domain**
17. **Apply density boundary conditions to corners on the PCM domain**
18. **For each nth time step t**
    (a) retrieve all relevant macroscopic values for data and intermediary plots

- End simulation when past maximum time steps OR when a converging solution is found OR when the part of the freeze plug has melted completely through in freeze plug simulations.

The converging criteria is stated as follows:

$$\sqrt{\frac{\sum_i \left(\phi\left(\mathbf{x_i},t\right)-\phi\left(\mathbf{x_i},t-\Gamma\Delta t\right)\right)^2}{\sum_i \left(\phi\left(\mathbf{x_i},t-\Gamma\Delta t\right)^2\right.}} < \delta_\phi \tag{3.58}$$

Where $\phi\left(\mathbf{x_i},t\right)$ is the chosen indicator variable and δ_ϕ is the converging criterium. i.e. how stable the solution must be. Γ is the number of timesteps between each instance that the converging criteria is checked.

4

Benchmarking Individual Models

Before being able to properly simulate and research the influence of design parameter on the melting times of the freeze plug, it is important to first benchmark the simulation. By testing each individual thermodynamic problem before proceeding to the final step, it becomes easier to expose flaws during development. Below are the results from benchmarking against individual models.

4.1. Natural convection

The first benchmark follows papers from Vahl Davis [48] and Zhuo and Zhong [41]. Both papers simulate a square cavity that is heated from the side for different Rayleigh numbers. Vahl Davis used a finite difference method that used a stream function-vorticity formulation. This is just a practical way to describe the velocities of the fluid. Zhuo and Zhong used a FM-LBM. However, instead of a FM boundary method they used bounce back schemes at the boundary.

Simulation benchmark goals

Show workings of

- base code (LBM collision scheme, translating macro variables into LBM variables and vice versa, etc.)
- Temperature field collision, propagation and boundary conditions
- Density field collision, propagation and boundary conditions.

The square cavity contains a Boussinesq fluid with properties that can be seen in table 4.1. The Rayleigh number dictates how large the natural convection effect is. This benchmark aims to show the

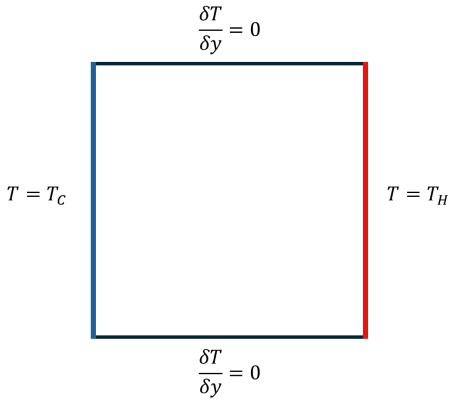


Figure 4.1: A schematic of the natural convection benchmark case, as performed by Davis [48] and Zhuo [41]. The type of the boundary condition is noted by both text and color.

effects of natural convection in fluid flow. The Rayleigh number in all simulations was kept below a threshold where the fluid would undergo turbulence. For all simulations, the Rayleigh number does not go higher than 10^6, which should keep it in the laminar regime. [49]

Table 4.1: Physical and lattice parameters (Natural convection benchmark case)

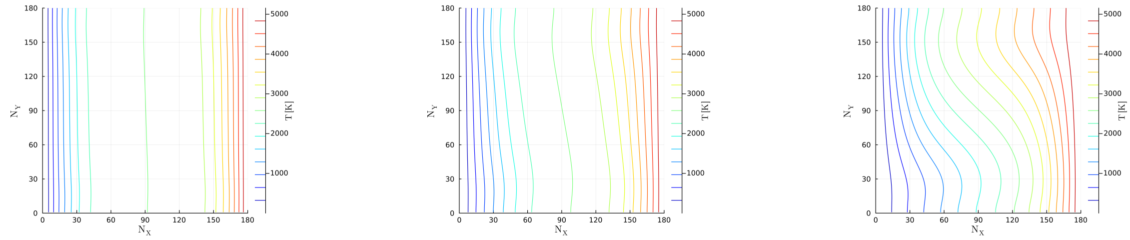
Description	*Variable*	*Physical value*	*Unit*	*Lattice value*	*Unit*
Kinematic viscosity	ν	$8.56 \cdot 10^{-7}$	$[\mathrm{m}^2/\mathrm{s}]$	$\frac{1}{100}, \frac{1}{100}, \frac{1}{150}$	$[ls^2/lt]$
Thermal diffusivity	κ	$1.23 \cdot 10^{-6}$	$[\mathrm{m}^2/\mathrm{s}]$	$1.23 \cdot 10^{-6}/C_\kappa$	$[ls^2/lt]$
Cubic thermal expansion coeff.	α	$2.07 \cdot 10^{-4}$	$[K^{-1}]$	$2.07 \cdot 10^{-4}$	$[K^{-1}]$
Gravitational acceleration	g	9.81	$[\mathrm{m}/\mathrm{s}^2]$	$9.81/C_g$	$[ls/lt^2]$
Initial density	ρ_0	1.293	$[\mathrm{kg}/\mathrm{m}^3]$	1.0	$[ls^{-3}]$
Rayleigh number	Ra	$10^4, 10^5, 10^6$	–	same	–
Prandtl number	Pr	0.7	–	0.7	–
Wall temperature	$T_{H,C}$	$T_0 \pm \Delta T/2$	$[K]$	same	$[K]$
Initial temperature	T_0	1	$[K]$	1	$[K]$
Height & Width cavity	$H \times H$	0.001×0.001	[m]	$N \times N$	$[ls]$
Number of cells in 1 direction	N	-	–	$180, 180, 220$	-

The temperature difference between the walls is calculated from the initial value given by Ra.

$$\mathrm{Ra} = \frac{\Delta T \alpha N^3 g}{\kappa \nu} \quad \Rightarrow \quad \Delta T = \frac{\mathrm{Ra}\, \kappa \nu}{\alpha N^3 g} \tag{4.1}$$

For different Rayleigh numbers, different values were taken for the kinematic viscosity and the grid size. ν and N were chosen to ensure that the simulations run stable and produce accurate results. These specific numbers were also chosen to compare exact results with the results from Borstlap [16].

Figures 4.2, 4.3 and 4.4 show the temperature isolines ascertained in the simulation during specific time stamps. A qualitative reference can be made by comparing the flow patterns during the same time stamps from the source papers. The results compare well qualitatively as can be seen from the isolines from the source material. They also make physical sense. The heated fluid rises next to the heated wall, flows over to the cold wall, and cools down. Finally, the steady-state solution is reached when both $\delta_p = 10^{-4}$ for $\phi =$Nu and $\delta_p = 10^{-5}$ for $\phi = T$. The amount of time steps Γ between each conversion check was set to 500.



(a) t = 0.008 s **(b)** t = 0.016 s **(c)** t= 0.033 s

Figure 4.2: The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = 10^4 in this simulation.

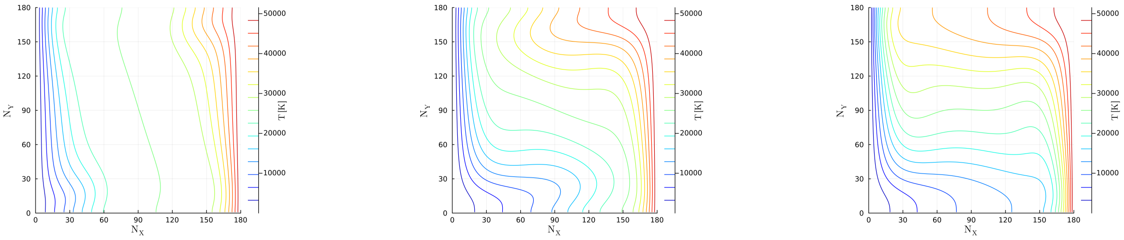
A good quantitative way to compare results in thermodynamic fluid flow is the Nusselt number. The Nusselt number is defined by:

$$\mathrm{Nu}_L = \frac{\text{Convective heat transfer}}{\text{Conductive heat transfer}} = \frac{\Delta T\ h_T}{\Delta T\ k/L} = \frac{h_T L}{k}, \tag{4.2}$$

where h is the convective heat transfer coefficient of the flow, L is the characteristic length, and k is the thermal conductivity of the fluid.

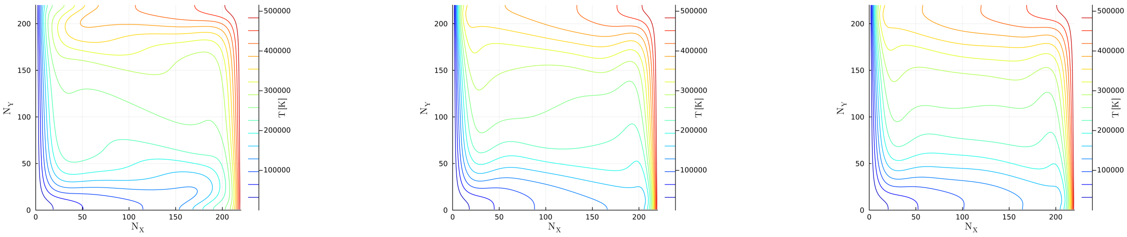
The Nusselt number at the cool wall can be translated to [41] [50]

$$\mathrm{Nu} = -\frac{H}{\Delta T}\left(\frac{\partial T}{\partial x}\right)_w \tag{4.3}$$



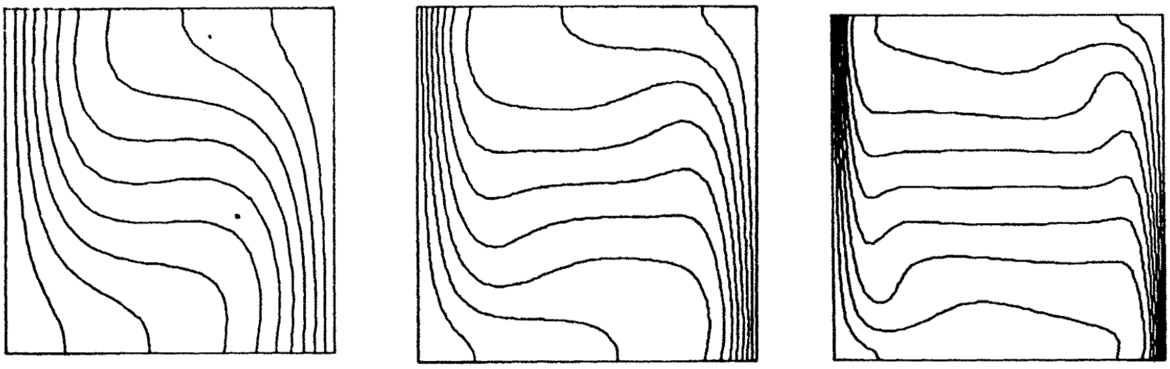
(a) t = 0.008 s **(b)** t = 0.016 s **(c)** t = 0.043 s

Figure 4.3: The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra = 10^5 in this simulation.



(a) t = 0.008 s **(b)** t = 0.016 s **(c)** t = 0.0179 s

Figure 4.4: The temperature isotherms of the natural convection benchmark shown at 3 different times (beginning, middle and end).The final time shown is the the time for which the solution has converged. Ra = 10^6 in this simulation.



(a) Ra = 10^4 **(b)** Ra = 10^5 **(c)** Ra = 10^6

Figure 4.5: These figures show the temperature isotherms found by De Vahl Davis [48]. Each figure shows the converged solution for a different Rayleigh number.

To take the average Nusselt number at the cool wall Nu_0 a simple integration is enough.

$$Nu_0 = -\frac{1}{\Delta T}\int_0^H \left(\frac{\partial T}{\partial x}\right)_{x=0} dy \tag{4.4}$$

To calculate this numerically, Simpson's rule is used. [51]

$$\int_a^b f(x)dx \approx \frac{1}{3}h\left[f(x_0) + 4\sum_{i=1}^{n/2} f(x_{2i-1}) + 2\sum_{i=1}^{n/2-1} f(x_{2i}) + f(x_n)\right] \tag{4.5}$$

$\frac{\partial T}{\partial x}$ is obtained using a second-order finite difference method. The reason why this option is chosen and not taking it directly from the temperature solution vector is because the solution vector at the boundary is only first order accurate, since only first order extrapolation is used as is discussed in section 3.4.2. The third-order finite difference method should produce more accurate Nusselt number results.

$$\frac{\partial T}{\partial x} \approx \frac{1}{12}\left(-25T_i + 48T_{i+1} - 36T_{i+2} + 16T_{i+3} - 3T_{i+4}\right) \tag{4.6}$$

The error produced by the quantitative test is calculated using L_2 since it is more rigorous than L_1

$$\epsilon_\phi = \sqrt{\frac{\sum_i \left(\phi_i - \phi_{i,ref}\right)^2}{\sum_i \left(\phi_{i,ref}\right)^2}} \tag{4.7}$$

Ra	**Study**	**$\mathbf{Nu_0}$**
$\mathbf{10^4}$	Davis	2.238
	Zhuo	2.245
	Borstlap	2.232
	Current	1.625
	Error [%]	27 - 27.6
$\mathbf{10^5}$	Davis	4.509
	Zhuo	4.521
	Borstlap	4.543
	Current	4.532
	Error [%]	0.24 - 0.5
$\mathbf{10^6}$	Davis	8.817
	Zhuo	8.819
	Borstlap	8.890
	Current	8.807
	Error [%]	0.11 - 0.93

Table 4.2: Comparison of results for different Rayleigh numbers for natural convection benchmark.

From both the qualitative and quantitative tests it can be seen that for Ra $= 10^5$ and Ra $= 10^6$ there is good agreement with the benchmark studies. However, the results for Ra $= 10^4$ show a lack of heat transfer. Ra $= 10^4$ has a smaller ΔT and in consequence also a smaller natural convection effect. It could be a possibility that there is an error in the conduction regime where the effects of natural convection are not yet large enough. This error gets diminished for higher heat transfer and body forces. Increasing grid size and changing the kinematic viscosity did not yield better results. Also changing from wet node boundary conditions to bounce back methods did not improve the results generated.

4.2. Conjugate Boundary

Following simulations performed by Kaminski [52] and Misra [53]. Neither of them used the LBM method. Kaminski used a finite difference model, while Misra used a finite elements model. The set-up is similar to the previous benchmark. The exception is a solid wall that is positioned to the right of the square cavity. The wall itsself is heated on the right side. And on the left side it is connected to the fluid domain through a conjugate boundary.

Simulation benchmark goals

Show workings of

- Conjugate boundary condition

Most physical and lattice parameters were also kept the same as in the previous benchmark. Relevant parameters that were changed can be found in the table below. Also some new parameters of the wall can be found in the table. One might notice that the relation between κ_W and κ and the relation between λ and λ_W is the same. This is because the specific heat capacity and the density in both domains are equal.

$$\rho c_P = \frac{\lambda}{\kappa} \tag{4.8}$$

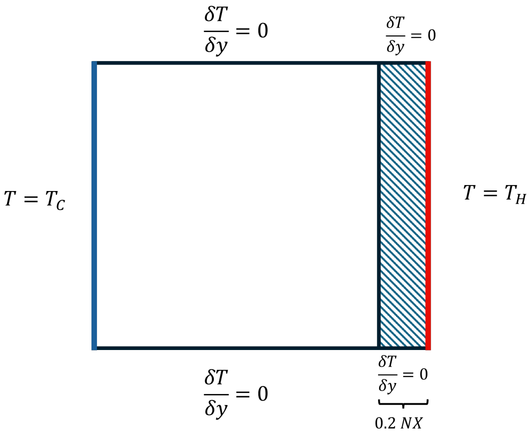


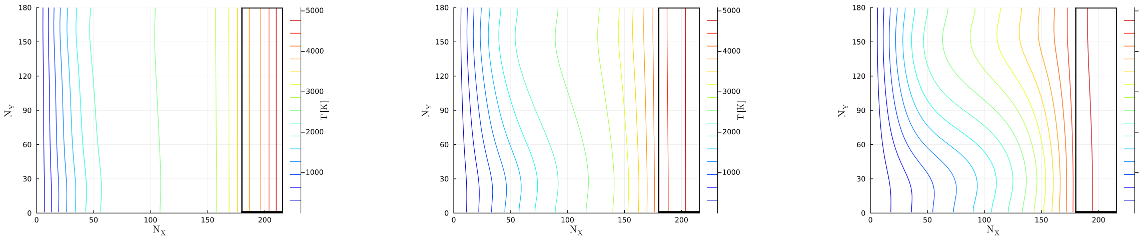
Figure 4.6: The schematic for the conjugate boundary condition benchmark. The right edge of the wall is heated, and the wall and fluid domain are connected through a conjugate boundary.

This implies that while both materials store thermal energy similarly, they transport it at different rates. As a result, temperature fields evolve at different speeds through each domain, influencing the thermal coupling across the interface.

Table 4.3: Physical and lattice parameters (Conjugated boundary bench mark)

Description	*Variable*	*Physical value*	*Unit*	*Lattice value*	*Unit*
Prandtl number	Pr	0.71	–	0.71	–
Rayleigh number	Ra	10^4,$0.7 \cdot 10^5$,$0.7 \cdot 10^6$	–	same	–
Wall thickness ratio	$\frac{d}{H}$	0.2	$\frac{N_W}{N}$	0.2	
Kinematic viscosity	ν	$8.56 \cdot 10^{-7}$	$[\mathrm{m}^2/\mathrm{s}]$	$\frac{1}{100}, \frac{1}{100}, \frac{1}{150}$	$[ls^2/lt]$
Thermal diffusivity wall	κ_W	$6.15 \cdot 10^{-6}$	$[\mathrm{m}^2/\mathrm{s}]$	$6.15 \cdot 10^{-6}/C_\kappa$	$[ls^2/lt]$
Thermal conductivity fluid	λ	$51.74 \cdot 10^-3$	$[\mathrm{kgm/s^3K}]$	$51.74 \cdot 10^-3/C_\lambda$	$[ls^{-1}lt^3K]$
Thermal conductivity wall	λ_W	$258.7 \cdot 10^-3$	$[\mathrm{kgm/s^3K}]$	$258.7 \cdot 10^-3/C_\lambda$	$[ls^{-1}lt^3K]$
Number of cells in 1 direction	N	-	–	$180, 180, 220$	–

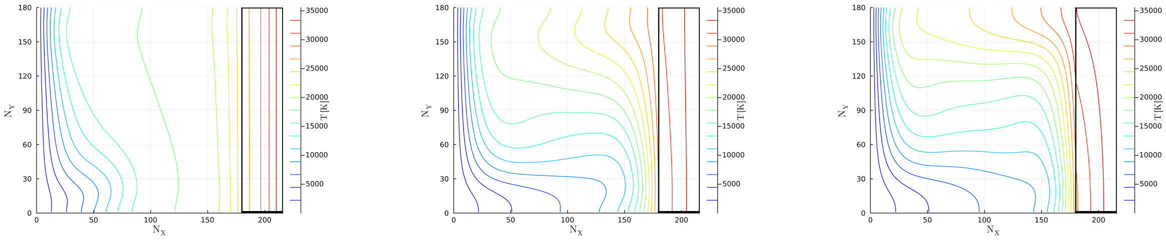
First the temperature isotherms are plotted to see if they make sense physically. After looking at the figures 4.7, 4.8 and 4.9 it can be seen that they do. In the converged solutions the fluid domain shows the same properties as it did in the original benchmark. Which makes sense, since the Boussinesq fluid is still getting heated, just with an solid wall between the fluid and the heat source. To analyze the effects that the wall has on the domain it is easier to also view the Nusselt numbers . The steady state solution is once again reached when both $\delta_p = 10^{-4}$ for $\phi =$Nu, $\delta_p = 10^{-5}$ for $\phi = T$ but for this case Γ was set to 1000 due to longer converging times.



(a) t = 0.007 s **(b)** t = 0.014 s **(c)** t = 0.020 s

Figure 4.7: The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra $= 10^4$ in this simulation.

Once again the Nusselt number is used to verify quantitative results. For this results comparison the Nusselt number is taken at the intersection between the wall and the fluid domain $\mathbf{Nu_W}$. Where $\frac{\partial T}{\partial x}$

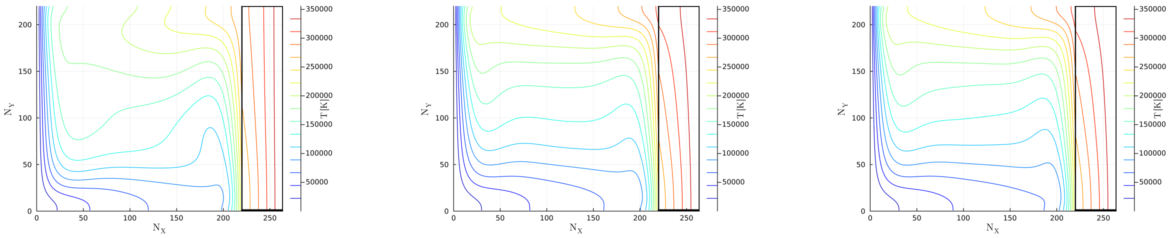


(a) t = 0.007 s

(b) t = 0.014 s

(c) t = 0.027 s

Figure 4.8: The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra $= 0.7 \cdot 10^5$ in this simulation.



(a) t = 0.007 s

(b) t = 0.014 s

(c) t = 0.016 s

Figure 4.9: The temperature isotherms of the conjugated boundary benchmark shown at 3 different times (beginning, middle and end). The final time shown is the the time for which the solution has converged. Ra $= 0.7 \cdot 10^6$

is taken in the direction of the fluid domain.

Ra	**Study**	**$\mathrm{Nu_W}$**
10^4	Kaminski	–
	Misra	2.021
	Borstlap	2.025
	Current	1.602
	Error [%]	20.7 - 21.0
$0.7 \cdot 10^5$	Kaminski	3.42
	Misra	3.436
	Borstlap	3.423
	Current	3.422
	Error [%]	0.03 - 0.41
$0.7 \cdot 10^6$	Kaminski	5.89
	Misra	5.910
	Borstlap	5.857
	Current	5.832
	Error [%]	0.43 - 1.3

Table 4.4: Comparison of results for different Rayleigh numbers for conjugateed heat benchmark.

The results show the same pattern as in the previous benchmark. The results for the relatively low Rayleigh number Ra $= 10^4$ show a major deviation from the benchmark results. While the deviation is unfortunate, it is in line with the previous results. Also the higher Rayleigh number results once again seem to be correct. By comparing the resulting Nusselt numbers it is also easy to see the effect that the wall has on the converging solution. Even though the Rayleigh numbers are only slightly lower, the Nusselt number is substantially lower in relative terms. This means that in this simulation the heat transfer is lower at the conjugate boundary than it was at the Dirichlet boundary. This makes

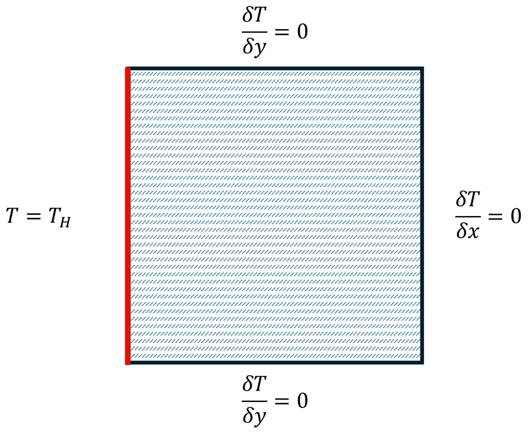


Figure 4.10: The schematic for the enthalpy melting benchmark. The left edge is heated, while the rest of the walls are adiabatic. The domain is filled completely with PCM.

sense, since the wall can be seen as a sort of barrier for the heat transfer because of the higher thermal conductivity.

4.3. Enthalpy melting

The enthalpy melting bench mark is an adaptation on the 2 phase Stefan problem [54] that includes the effects of natural convection. The benchmark studies chosen are from Huber [55] and Darzi [56]. Huber and Darzi both used the Lattice Boltzmann method. However, they used an MDF (Multiple Distribution Function) approach [57] [58]. The melting method Huber and Darzi used was similar since they also used an enthalpy based melting method [59] instead of a phase field method. [60].

Simulation benchmark goals

Show workings of

- Enthalpy field collision, propagation and boundary conditions
- Enthalpy porosity method

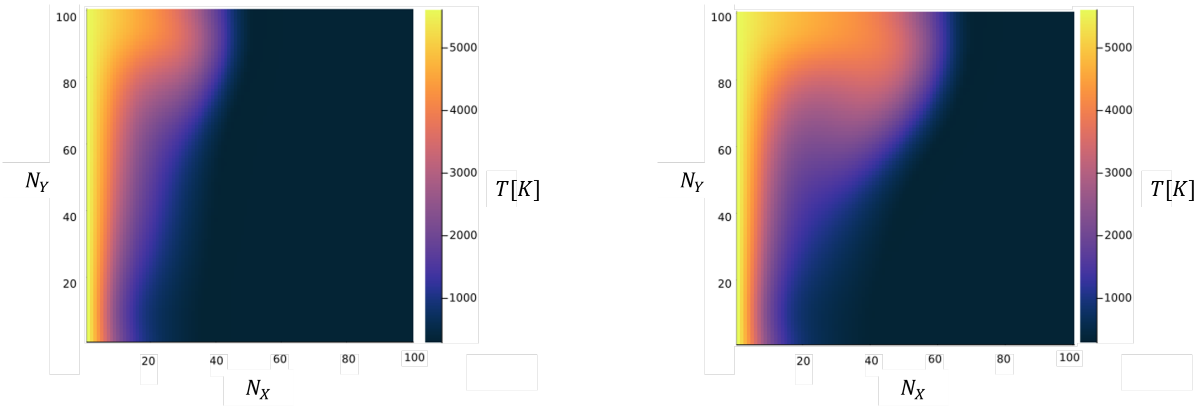
The set up for the enthalpy melting benchmark is as follows. There is a solid in a square cavity at T_0. The solid has a melting temperature of T_l which is slightly higher than the solidifying temperature T_s. The left wall is constantly heated at a temperature T_H where $T_H > T_L > T_0$. The melting temperature is slightly higher than the solidifying temperature. The solidification temperature is taken to be higher than the melting temperature to account for phase change hysteresis. [61] This distinction can also aid in numerical stability by preventing ambiguity at the phase interface during thermal transitions.

The figures 4.11, 4.12, 4.13 and 4.14 show the melting process in 2 different ways. The first set shows a temperature heatmap throughout different stages of the process. The second set of images shows the liquid fraction of the substance. These images provide a qualitative view over the benchmark experiment. They can not be directly compared to results from Huber and Darzi. But, they aim to show physical validity.

It can be seen that the liquid fraction is mostly either 1 or 0. This makes sense, since a liquid fraction between 1 and 0 only signifies how much of the particular cell has turned into liquid. So the fact that this can only be seen at the melting border is a good sign. Furthermore, the melting process at different time steps, show the different phases properly as well according to figure 4.15. First there is a dominant conduction regime i). Here the temperature difference between the heated boundary and the solid phase drives heat transfer primarily through thermal conduction. In this stage, the liquid phase formed at the heated surface is relatively thin, and buoyancy-induced fluid motion is negligible due to the limited volume and minimal temperature gradients in the fluid. The melting front advances predominantly by conduction of heat into the solid, leading to a relatively uniform and smooth phase interface progression.

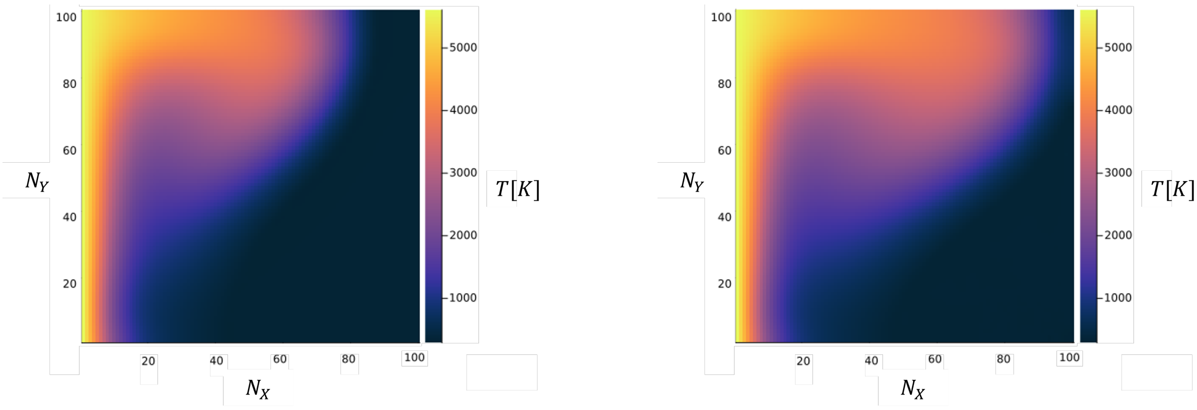
Table 4.5: Physical and lattice parameters (Case 2 - Phase Change)

Description	*Variable*	*Physical value*	*Unit*	*Lattice value*	*Unit*
Thermal conductivity	λ	1.5	[W/m/K]	$1.5/C_\lambda$	$[ls/lt^3/K]$
Thermal diffusivity	κ	$7.5 \cdot 10^{-3}$	[m^2/s]	$7.5 \cdot 10^{-3}/C_\kappa$	$[ls^2/lt]$
Specific heat	c_p	0.0456	[Ws/kg/K]	$0.0456/C_p$	$[ls^2/lt^2/K]$
Latent heat	L	$159 \cdot 10^3$	[Ws/kg]	$159 \cdot 10^3/C_{Lat}$	$[ls^2/lt^2]$
Thermal expansion coeff.	α	$2.5 \cdot 10^{-4}$	$[K^{-1}]$	same	$[K^{-1}]$
Kinematic viscosity	ν	0.0075	[m^2/s]	$1/45, 1/80$	$[ls^2/lt]$
Prandtl number	Pr	1.0	–	1.0	–
Stefan number	Ste	10	–	10	–
Rayleigh number	Ra	$5 \cdot 10^4, 1.7 \cdot 10^5$	–	same	–
Wall temperature	T_H	$T_0 + \Delta T$	$[K]$	same	$[K]$
Initial temperature	T_0	301.14	$[K]$	301.14	$[K]$
Solidifying temperature	T_s	301.14	$[K]$	301.14	$[K]$
Melting temperature	T_l	$T_s + 0.002$	$[K]$	same	$[K]$
Gravitational acceleration	g	9.81	[m/s^2]	$9.81/C_g$	$[ls/lt^2]$
Height & Width cavity	$H \times H$	0.6×0.6	[m]	$N \times N$	$[ls]$
Number of cells in 1 direction	N	–	–	100,100	–



(a) t = 1.05 s

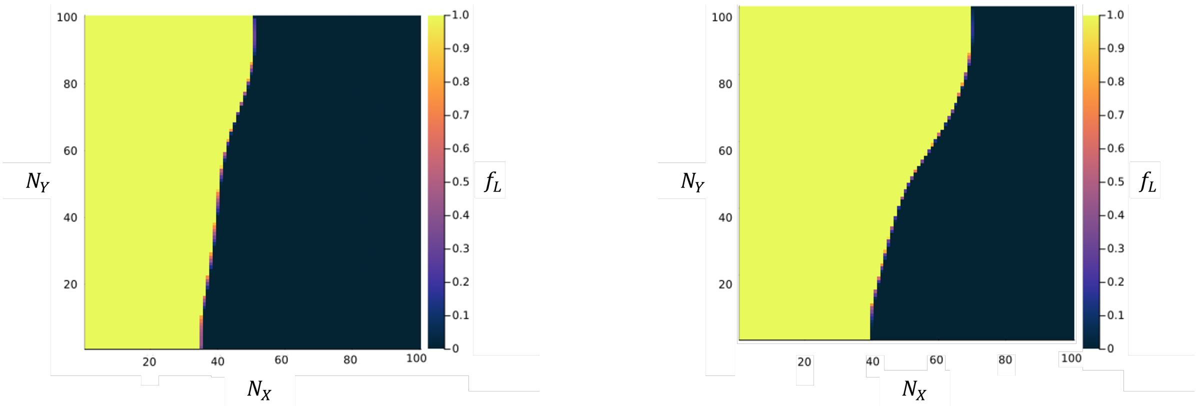
(b) t = 1.57 s



(c) t = 2.09 s

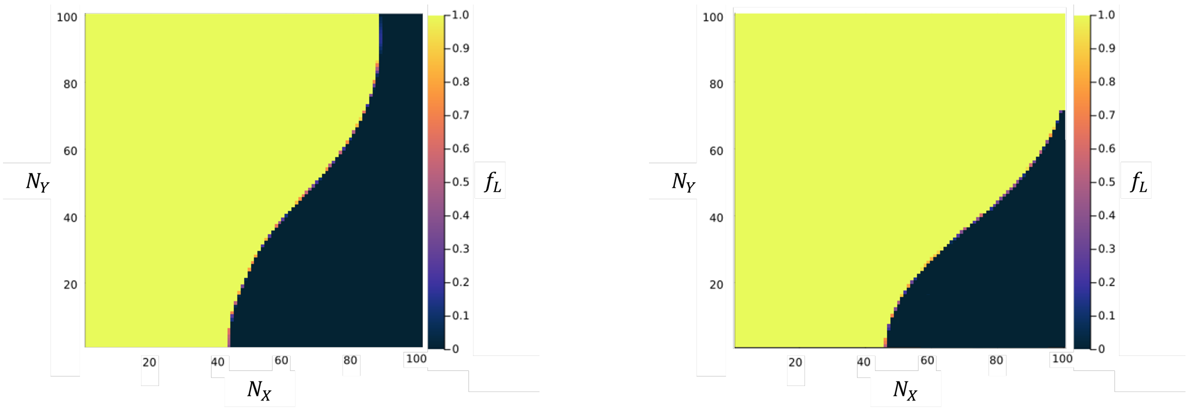
(d) t = 2.61 s

Figure 4.11: Temperature gradient for enthalpy melting benchmark at different times for Ra $= 5 \cdot 10^4$



(a) t = 1.05 s

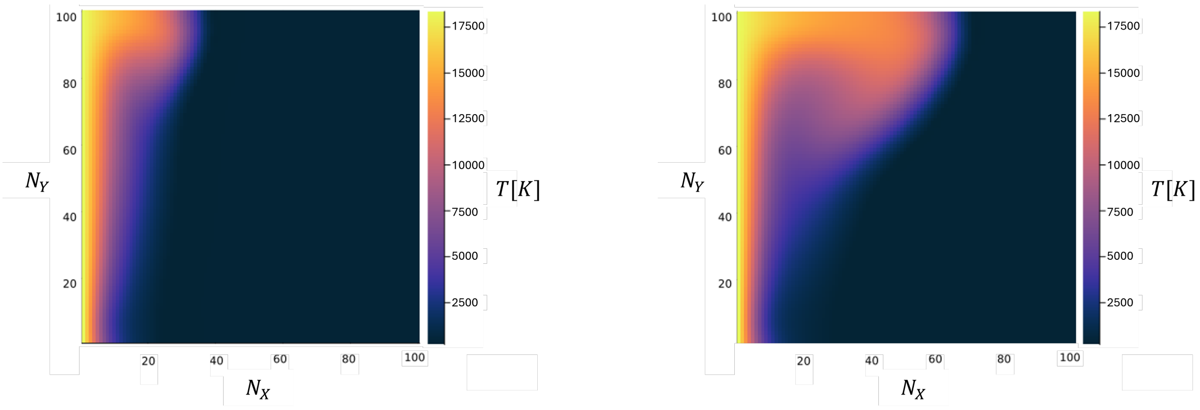
(b) t = 1.57 s



(c) t = 2.09 s

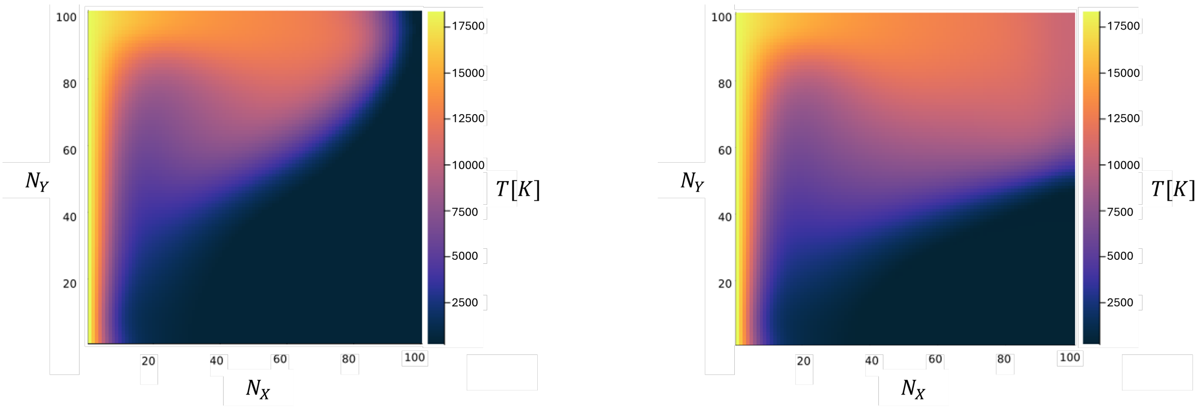
(d) t = 2.61 s

Figure 4.12: The liquid fraction for the enthalpy benchmark at different times for $\mathrm{Ra} = 5 \cdot 10^4$



(a) t = 0.037 s

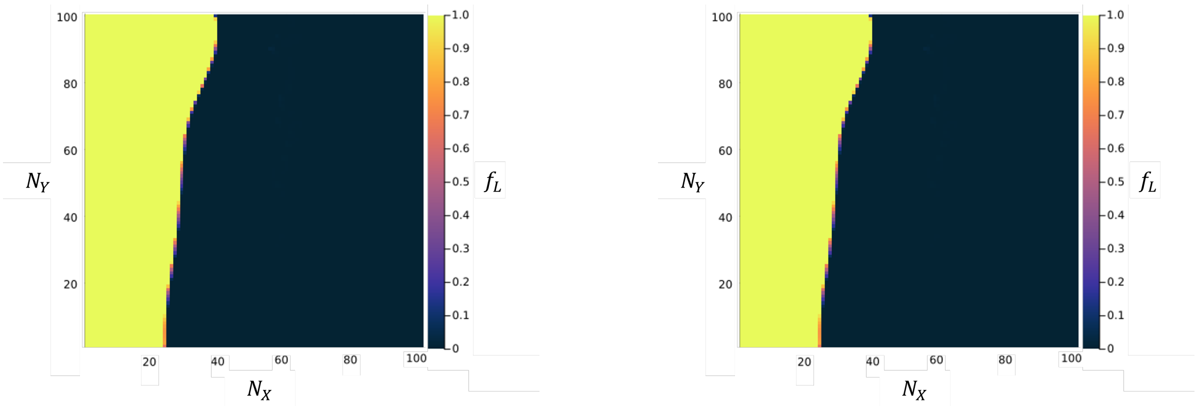
(b) t= 0.074 s



(c) t = 0.111 s

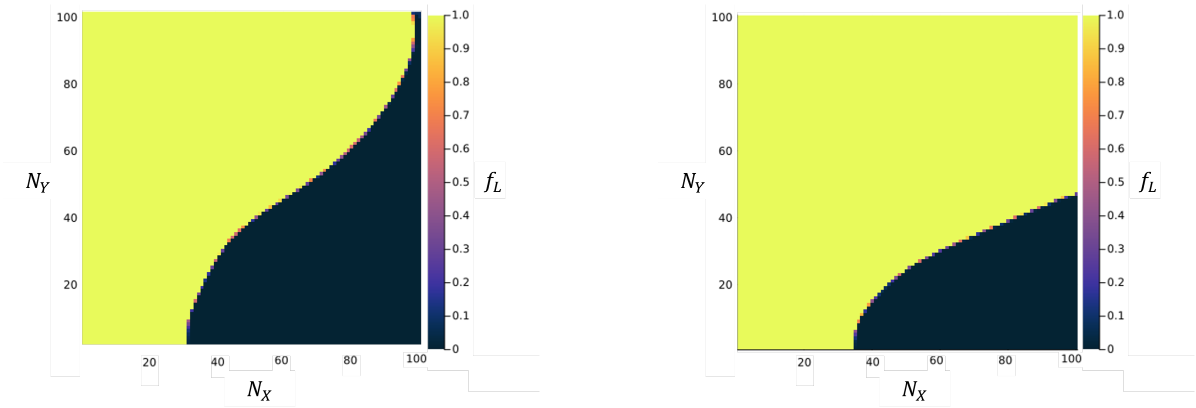
(d) t = 0.149 s

Figure 4.13: The temperature gradient for the enthalpy benchmark at different times for Ra $= 1.7 \cdot 10^5$



(a) t = 0.037 s

(b) t = 0.074 s



(c) t = 0.111 s

(d) t = 0.149 s

Figure 4.14: The liquid fraction for the enthalpy benchmark at different times for $\mathrm{Ra} = 1.7 \cdot 10^5$

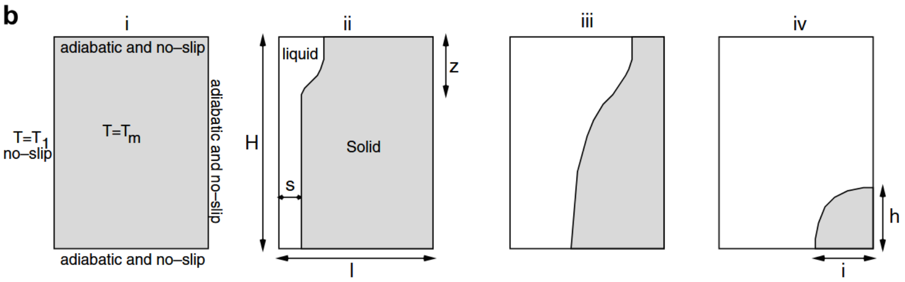


Figure 4.15: Different regimes of melting. From conduction dominated melting to natural convection dominated melting. [55]

Notably, the heat flux preferentially pushes the melting front sideways along the heated wall because the temperature gradient normal to the wall is largest near the boundary. This anisotropic propagation results in a lateral spreading of the melted region before any significant convective motion develops.

As the melted liquid layer thickens, temperature gradients within the fluid increase as well as the available volume and buoyancy effects become non-negligible.

Eventually, the buoyancy forces fully dominate the heat transfer process, giving rise to classical natural convection patterns within the liquid phase. In this regime, warm fluid near the heated boundary rises due to its lower density, while cooler fluid descends, establishing circulating flow cells. This convective circulation enhances heat transport from the hot boundary into the melt, accelerating melting.

The quantitative check is done once again by the Nusselt number. But now it is taken over dimensionless time. The dimensionless time is defined as

$$\theta = \mathrm{Fo} \cdot \mathrm{Ste} \tag{4.9}$$

Where

$$\mathrm{Fo} = \frac{\kappa t}{N^2} \qquad \mathrm{Ste} = \frac{c_P \Delta T}{H} \tag{4.10}$$

The reason a dimensionless time is used here, is because Jany found the Nusselt number to scale with the dimensionless time [62]. The Nusselt number scales differently in different regimes. The different regimes can be found in figure 4.15, and the Nusselt number scales as follows at the start in regime i.

$$\mathrm{Nu} \propto \theta^{-1/2} + \mathrm{Ra}\, \theta^{3/2} \tag{4.11}$$

This ends at a time θ_1 which coincides with regime iii. At the end the Nusselt number evens out at

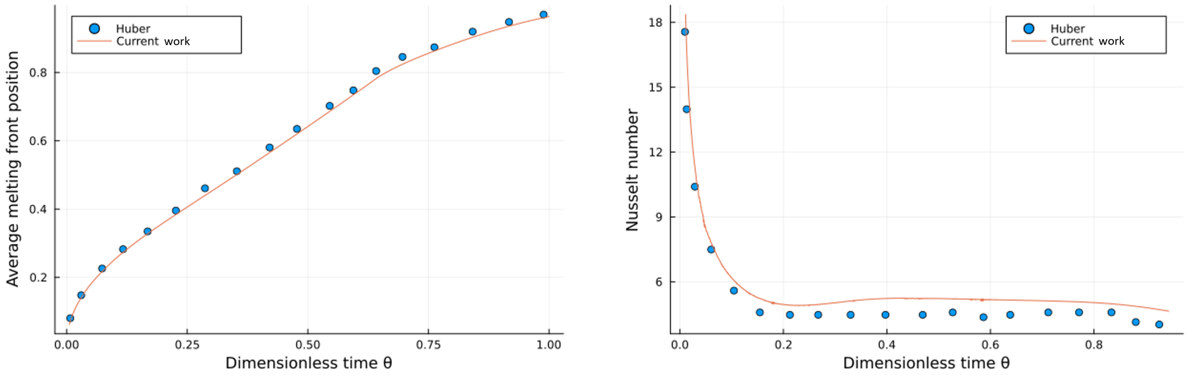
$$\mathrm{Nu} \propto \theta^{-\frac{1}{4}} \tag{4.12}$$

One more quantitative measurement is taken for the average melting front. This is calculated using

$$s_{av}(\theta) = \frac{1}{H} \int_0^H x_{\text{front}}\,(y, \theta) dy \tag{4.13}$$

Here x_{front} is determined by taking the first cell that is not liquid. From that x coordinate $(1 - f_L)$ is distracted. f_L here has the value of the cell to the left of the solid cell. The integral is once again calculated using Simpson's rule from equation 4.5.

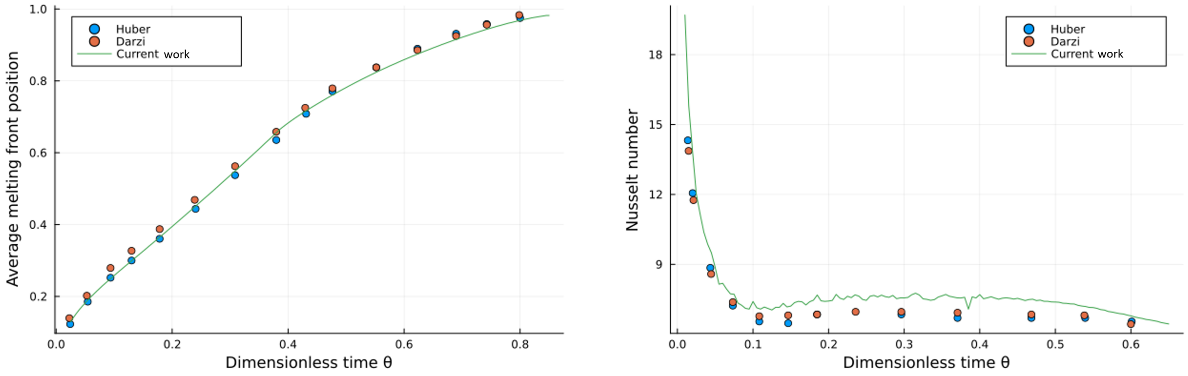
The melting front and Nusselt number coincide quite well with the benchmark. However, the Nusselt number is slightly to high. This does not however, coincide with the results seen in the average melting front. In this case it is better to look at the Nusselt number and not at the average melting front, namely with wrong results the melting front can take different shapes that might still produce a correct looking average melting front. For improvement in the results different solutions were attempted. Increasing grid size and changing the kinematic viscosity did not yield definitive positive results. Also changing the Dirichlet and Adiabatic boundary conditions to HBB and FBB (Full way Bounce Back) respectively did not improve the results generated. Also change in corner use had hardly any effects on quantitative results. This can further be seen in section 6.1. When looking at the results in different regimes, it seems



(a) The results for the average melting front over dimensionless time θ.

(b) The results for the Nusselt number at the left wall over dimensionless time θ.

Figure 4.16: Quantitative results for enthalpy melting benchmark with $\mathrm{Ra} = 5 \cdot 10^4$



(a) The results for the average melting front over dimensionless time θ.

(b) The results for the Nusselt number at the left wall over dimensionless time θ.

Figure 4.17: Quantitative results for enthalpy melting benchmark with $\mathrm{Ra} = 1.7 \cdot 10^5$

that the conduction regime produces good results, also corresponding well with the literature value for Nu.[62][55]. However, the convective effect seems overestimated. This does not seem related to the issue with the previous bench mark as that error undershot the Nusselt number instead of overshooting it. One possibility for the discrepancy is the change in numerical method from the benchmark that just produces slightly different results.

5

Freeze Plug model

The goal testing the previous benchmark models was to work towards a set up that can handle the freeze plug melting process. Altough by taking different steps a study by Pater & Kaaks [47] also modelled the melting process of the freeze plug. They started by modelling a square cavity enclosed with adiabatic boundary conditions with a transient dirichlet boundary condition at the top. This is similar to the final benchmark of the previous chapter.

The transient boundary condition simulates the decay heat from a shut-down reactor. The transient temperature is based on a study by Tiberga.[63] The study used

$$Q_R(t) = 6.45908 \cdot 10^6 - 6.9200 \cdot 10^5 \ln(t[s]) \quad \left(\mathrm{W\ m^{-3}}\right) \tag{5.1}$$

for the time-trend of the residual heat of the reactor [64]. Then, by using a lumped capacitance model and assuming no external heat losses

$$Q_R(t) = \rho c_p \frac{\mathrm{d}T(t)}{\mathrm{d}t} \tag{5.2}$$

This gives the following transient boundary condition used by Pater $ Kaaks.

$$T(t) = -0.0001t^2 + 0.5244t + 923 \quad (\mathrm{K}) \tag{5.3}$$

The second stage of the study done by Pater $ Kaaks adds a time-dependent volumetric heat source to the entire domain. This heat comes from energetic neutrons and photons that originate from the fuel. These neutrons and photons can also interact with the molten salt, and the not (yet) molten salt in the freeze plug. The heat is approximated to be 1% of the decay heat of the reactor. 1% of equation 5.1 gives the following transient heat source term.

$$Q(t) = 6.45908 \cdot 10^4 - 6.92 \cdot 10^3 \cdot \ln(t) \quad \left(\mathrm{W\ m^{-3}}\right) \tag{5.4}$$

Implementing the source term is easy enough. By converting the source term to lattice units

$$Q_{lat}(t) = Q(t) \frac{C_{time}^3}{C_L^2} \tag{5.5}$$

And adding it to the solution vector during the collisison step.

$$\gamma_0^{\pm}(\mathbf{x}, t) = h \pm \frac{\Delta t}{2} \frac{Q_{lat}(t)}{\rho_{lat}} \tag{5.6}$$

The top of the PCM domain is put in an already molten state. This shortens the period of the first melting regime when natural convection is activated. The height of the already molten part h is equal to $0.1H$. There is once again a difference in the melting temperature T_l and the solidification temperature T_s for the same reasons as before. The salt is modeled after LiF-ThF_4 and the wall is considered to be made of Hastelloy-N.

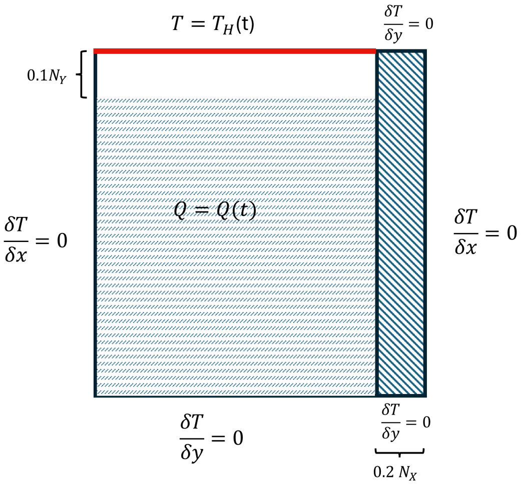


Figure 5.1: Schematic of freeze plug model and boundary conditions used. The Dirichlet boundary condition is dependant on decay heat as well as the volumetric heat source in the PCM domain. The PCM domain is considered a solid salt that is partially molten in the upper part with a height of 0.1 N_Y. The wall is made of Hastelloy-N and spans 0.2 N_X in the x direction.[47]

Table 5.1: Used thermophysical properties for fuel salt and Hastelloy N and some general parameters

Material	*Property*	*Physical value*	*Unit*	*Lattice value*	*Unit*
Salt	$\rho_s = \rho_f$	4390	kg/m^3	1.0	ls^{-3}
	$\lambda_s = \lambda_f$	1.5	W/m/K	$1.5/C_\lambda$	$ls/lt^3/K$
	$c_{p,s}$	815	J/kg/K	$815/C_{cp}$	$ls^2/lt^2/K$
	$c_{p,l}$	1000	J/kg/K	$1000/C_{cp}$	$ls^2/lt^2/K$
	T_s	841	K	841	K
	T_l	841.002	K	841.002	K
	L	$159 \cdot 10^3$	J/kg	$159 \cdot 10^3/C_{Lat}$	ls^2/lt^2
	α	$2.5 \cdot 10^{-4}$	K^{-1}	$2.5 \cdot 10^{-4}$	K^{-1}
	ν	$7.5 \cdot 10^{-4}/\rho$	m^2/s	$1/45$	ls^2/lt
	Pr	5	–	5	–
	Ste	0.063	–	0.063	–
	$T_{S,0}$	831	K	831	K
	$T_{L,0}$	841.001	K	841.001	K
	h/H	0.1	–	–	ls
Hastelloy-N	ρ	8860	kg/m^3	$8860/C_\rho$	ls^{-3}
	λ	23.6	W/m/K	$23.6/C_\lambda$	$ls/lt^2/K$
	c_p	578	J/kg/K	$578/C_{cp}$	$ls^2/lt^2/K$
	d/H	0.2	–	0.2	–
	T_0	831	K	831	K
General	H	0.2	m	–	ls
	L	0.1	m	–	ls

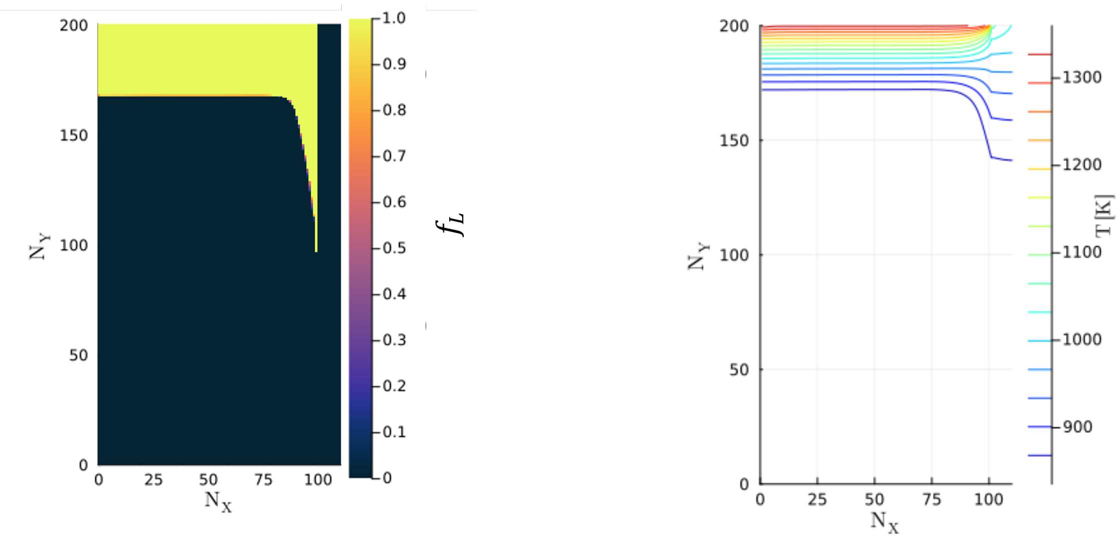
5.1. Conjugated conductive heat model

In the first case, the natural convection is not yet added. This follows stage 3 from Pater. The specific lattice parameters that were used can be found in table 6.1.

Lattice parameter	*unit*	*Value*
ν	$[ls^2/lt]$	1/40
N_X	$[ls]$	100
N_Y	$[ls]$	200

Table 5.2: Lattice parameters chosen for freeze plug model without natural convection.

The qualitative results can be compared with the benchmark at t = 2500 s. When looking at the way the isolines at t = 1000 s compare to the isolines at t = 2500 s. It looks like the results will perform well. The isolines make physical sense as well as the melting pattern. The wall conducts heat efficiently to the PCM due to the higher thermal conductivity that the Hastelloy-N wall has. The isolines also compare qualitively well with the figure 5.3a from OpenFOAM by Pater.



(a) Liquid fraction of freeze plug showing melting front at t = 1000 s

(b) Temperature isolines at t = 1000 s

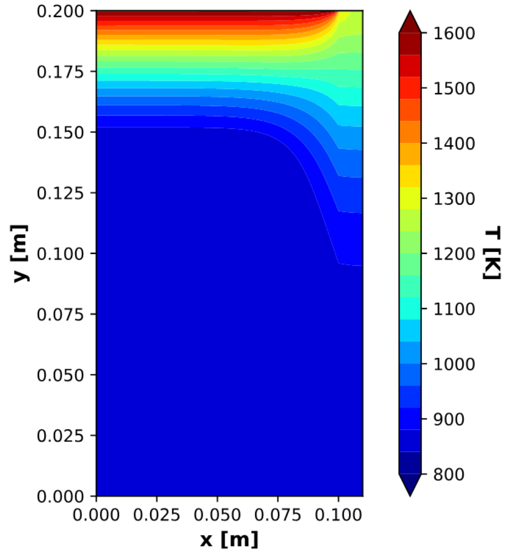
Figure 5.2: Results of modelling the freeze plug without natural convection at t = 1000 s

At t = 2500 the benchmark solution has been reached. The temperature heatmap still looks good compared to the qualitative benchmark. The benchmark 3 temperature measurements performed over the entire y coordinate. One at $x = 0$ m, one at $x = 0.075$ m and one at $x = 0.09$ m. The comparison between the results and the benchmarks can be viewed in figure 5.4. These results compare good to the benchmark. Especially compared to OpenFOAM. The difference between the methods is that the three codes adopt different modelling strategies for the conjugate heat transfer. OpenFOAM and Star CCM solve the heat equation in the salt and metal domains sequentially (i.e. first the heat equation in the salt is solved, then the heat equation in the metal), whereas in DGFlows the heat equation is solved for the full domain and no distinction is made between the wall and the PCM regions. [47] The melting front compares quite well with all benchmarks, as can be seen in figure 5.5.

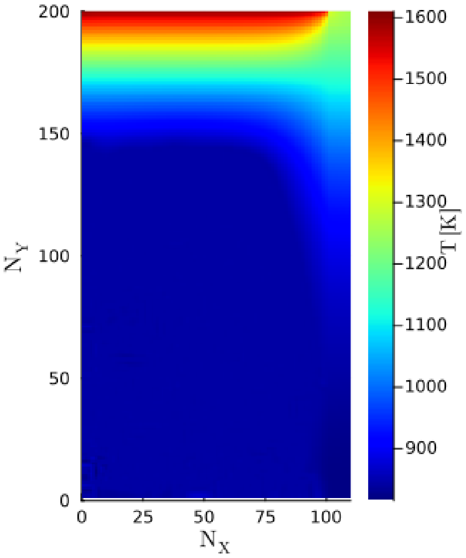
5.2. Natural convection model

For this part natural convection is applied by setting g to 9.81 ms^{-2}. To keep the Courant number low enough the grid size was increased and kinematic viscosity was decreased. The latter change results in smaller relative time steps.

These changes put a significant strain on the computational power needed. Simulating 2.5 seconds took the laptop used 500 seconds. The laptop used used a 13th generation Intel(R) Core(TM) i9-13900



(a) Temperature heat map from Pater at t=2500 s,using an 880x440 mesh with a timestep of $\Delta t = 0.5$ in OpenFOAM[47]



(b) Temperature heat map from the current FMLBM method at t=2500 s,using a 200x100 mesh with a kinematic viscosity of $\nu = 1/40$

Lattice parameter	*unit*	*Value*
ν	$[ls^2/lt]$	1/150
N_X	$[ls]$	250
N_Y	$[ls]$	500

Table 5.3: Lattice parameters chosen for freeze plug model with natural convection.

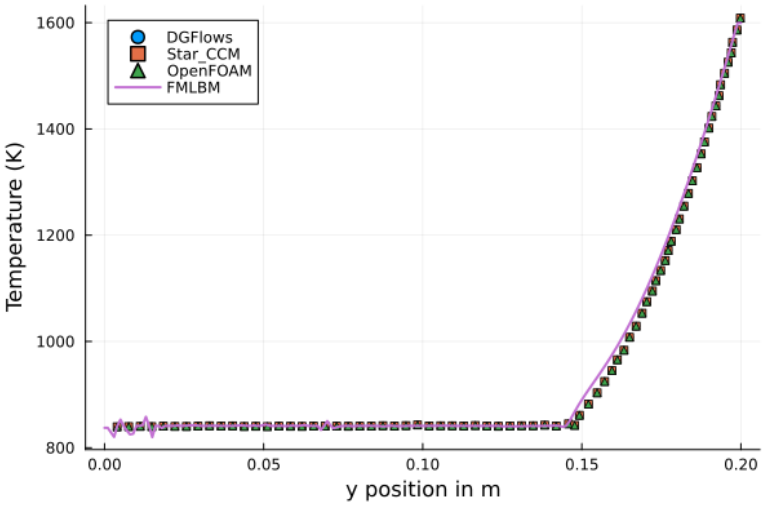
with a clock speed of $2.6GHz$. Simulating the total 2500 seconds would have taken approximately 139 hours. The computation speed can better be illustrated using LUPS (Lattice Updates Per Second) or for higher calculation speeds, MLUPS (Million Lattie Updates per Second). Using the above mentioned processing speeds this statistic can be calculated using [65]

$$\text{MLUPS} = \frac{N_{\text{grid}}\, N_T}{T} \times 10^{-6} \tag{5.7}$$

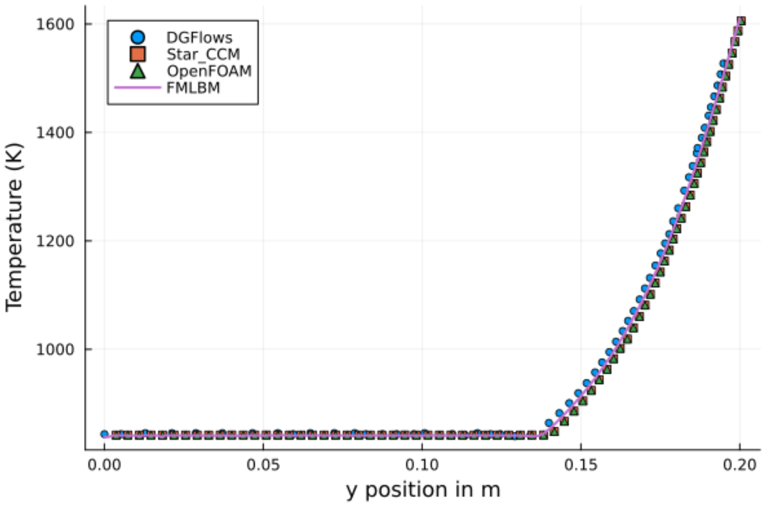
Where $N_{\text{grid}} = 250 \cdot 500$, $N_T = 4000$ and $T = 522.1$. This all comes together to about 1.2 MLUPS.

In this work, computation speed was not the main constraint. Somewhere after the aforementioned 2.5 seconds simulations became unstable. Figure 5.6 shows the liquid fraction at 125 seconds. One can directly see that results are nonphysical. This is further backed up by temperatures that went to $-2 \cdot 10^{290} K$ near the wall. This shows clear signs of instability. Creating an even larger grid size showed inconclusive results since the simulations took to long to gather any significant results.

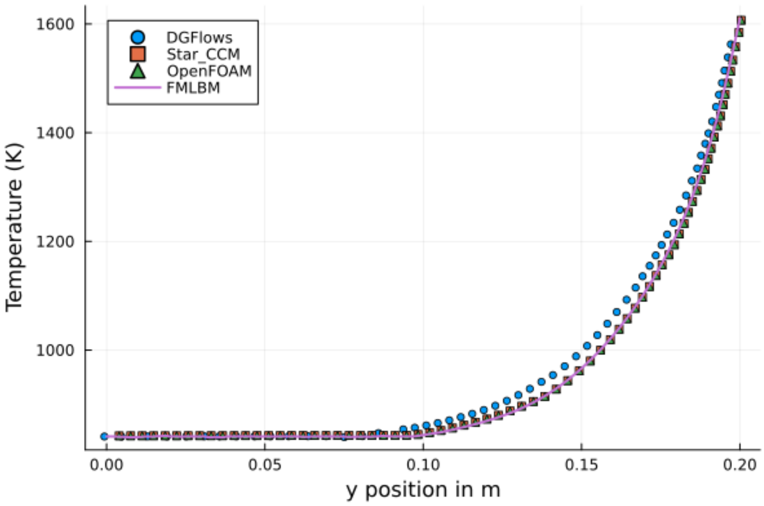
The reference benchmark paper shows that with natural convection implemented the melting would be accelerated [47], but only at the top side. The natural convection slows the melting at the side of the freeze plug adjacent to the metal wall. This makes sense since the heated fluid would rise causing the temperatures at lower depths to decrease. This causes opening times to be slower than without natural convection being implemented.



(a) $x = 0$ m



(b) $x = 0.075$ m



(c) $x = 0.09$ m

Figure 5.4: Temperature thermometers displaying temperature at a specific x coordinate over the y position.

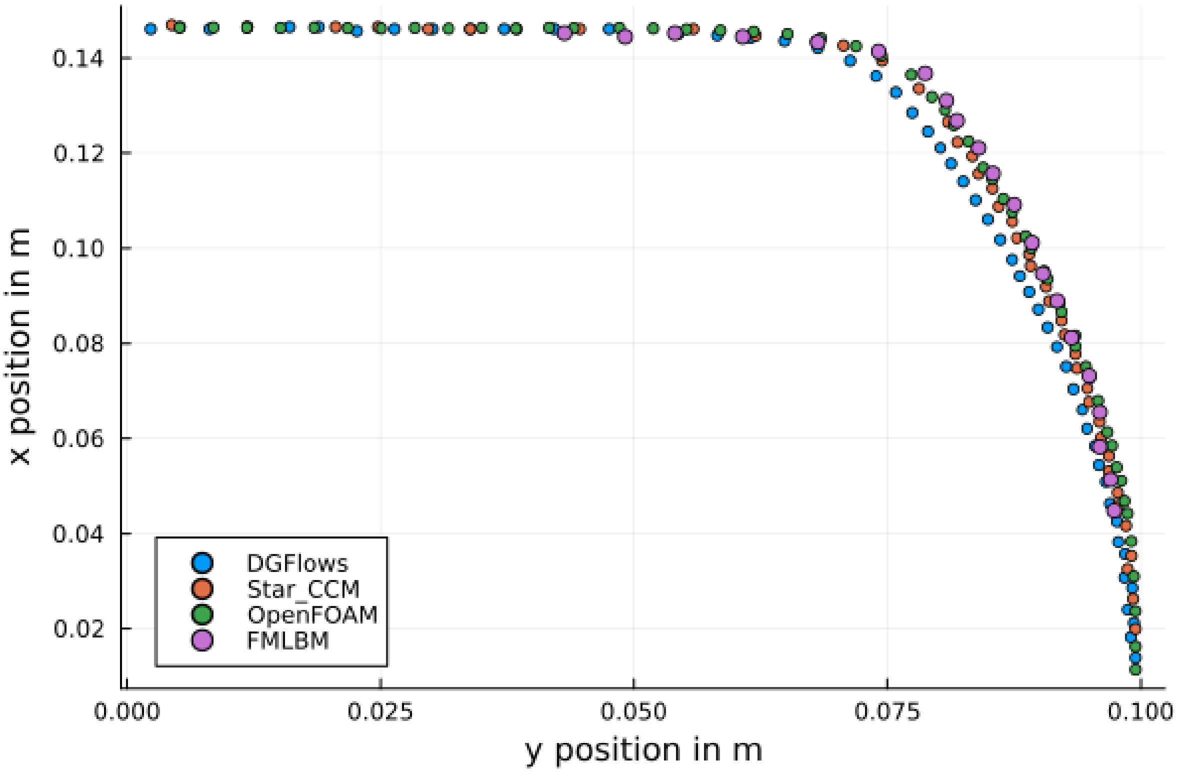


Figure 5.5: Meltfront results compared to benchmark results for the freeze plug without natural convection at t = 2500 s

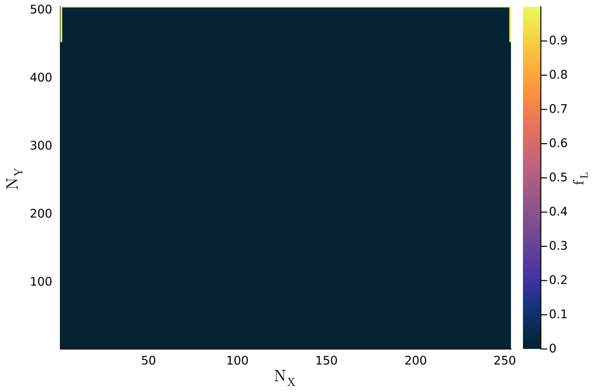


Figure 5.6: Liquid fraction freeze plug at approximately 2.5 seconds

6

Stability and Performance FMLBM

6.1. Corner influence

The filter matrix boundary method is still quite new. Previous work done by Borstlap did some research on the influence of order of extrapolation on the boundaries. Still a relevant question is: what is the influence of different corner methods on the filter matrix boundary method. That is the subject of this chapter. For all simulations a small grid size was chosen as to give the corner nodes more influence. Since it is assumed that changes in the corner method will only show tiny effects.

In figure 6.1 the possible orientations of the corner normal are shown. These orientations determine the extrapolation nodes that are considered when determining the temperature that the corner node should have.

Because of the nature of the Dirichlet and adiabatic boundary conditions and how the extrapolation from the corner is taken, it could make sense to not treat the corners separately at all. The main concern of treating corners differently is how 2 different boundary conditions are satisfied at one place. When looking at how the temperature at the boundary is calculated from extrapolation in the filter matrix method, one sees that equation 3.33 must be satisfied as well as equation 3.35. Even though it might feel instinctual to do so this can not be true when a diagonal extrapolation is applied. The diagonal method will still be tested, but one can see where the choice might already go wrong as having to freely choose between applying equation 3.33 or 3.35, which give different results for T_{BC}, starts to feel wrong.

For the enthalpy boundary condition this works exactly the same, as the boundary condition for enthalpy is translated into a temperature boundary condition anyways. For the density boundary condition it does not much matter as the boundary condition is always the same: no-slip; $ux = 0$, $uy = 0$.

6.1.1. No corners

For the no corners method, corners are not treated special. Each corner is part of each adjacent boundary. This does mean that corners are subjected to the boundary condition step twice. Once for each boundary. In all simulations, the Dirichlet BC were applied last.

Figure 6.3 shows that the no special corner method remains stable and seems to produce physical results similar to the benchmark results.

6.1.2. Orthogonal corners

Vertically directed corners that include the Dirichlet boundary when possible, show instability in figure 6.4 for high θ i.e. $\theta > 0.9$. The interesting thing is that the instability seems to come from the right bottom corner. In this corner there should not

Horizontally oriented corners that only include an adiabatic boundary condition show quick instability in figure 6.5. It did however, not become entirely unstable, unlike some other corner configurations.

6.1.3. Diagonal corners

Figure 6.6 shows the results for Diagonal corners that include only adiabatic boundary conditions.

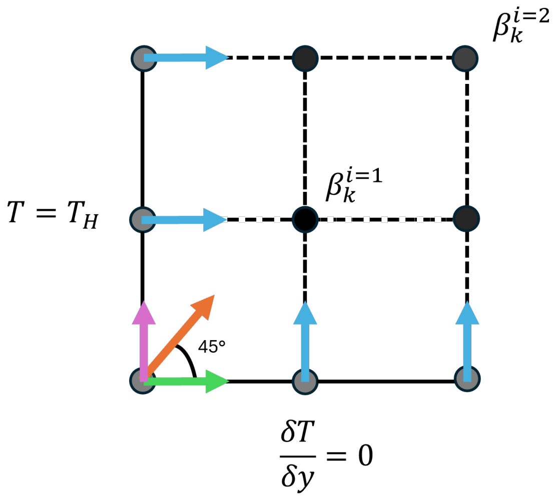


Figure 6.1: A zoomed in representation of the boundary workings near the corner. The corner normal orientation possibilities are shown in different colors. Pink shows vertical orientation towards a Dirichlet BC, orange shows diagonal orientation, both Dirichlet BC and adiabatic BC possible, green shows horizontal orientation towards adiabatic BC.

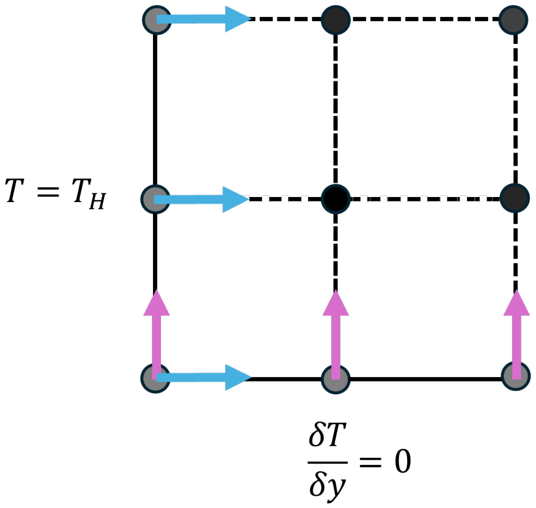
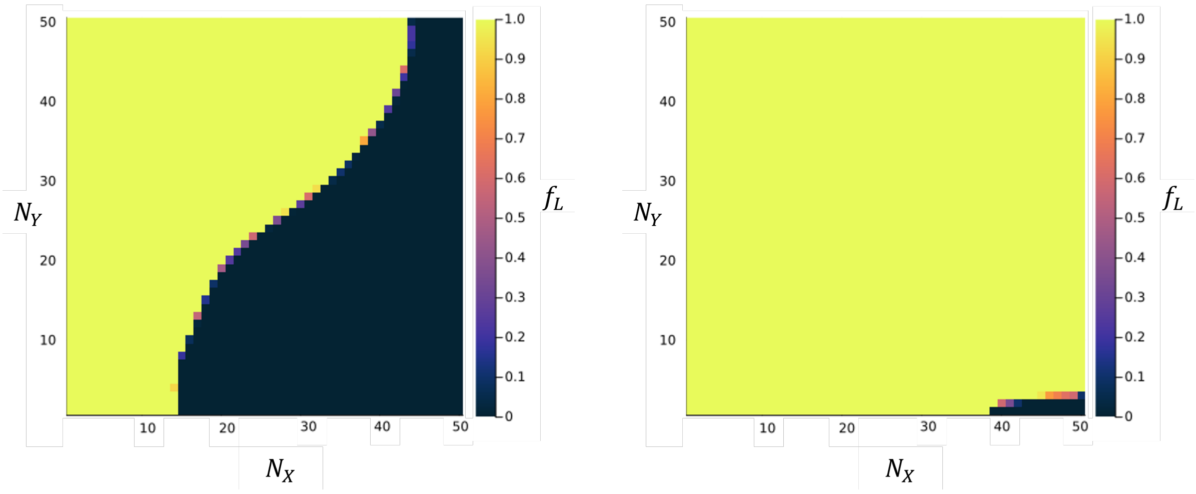
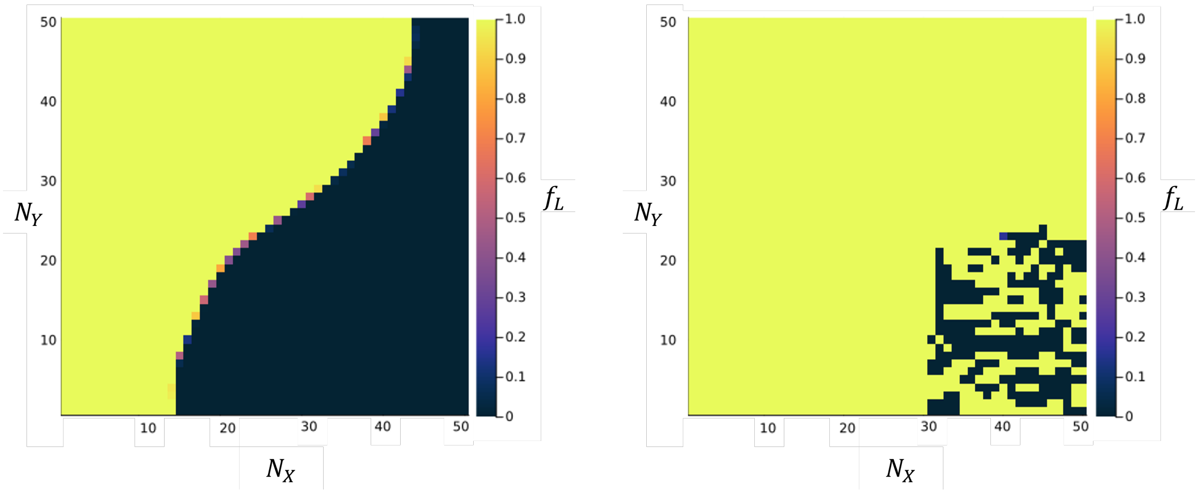


Figure 6.2: An example of the no corner method. Both boundary conditions are executed on the same corner node.



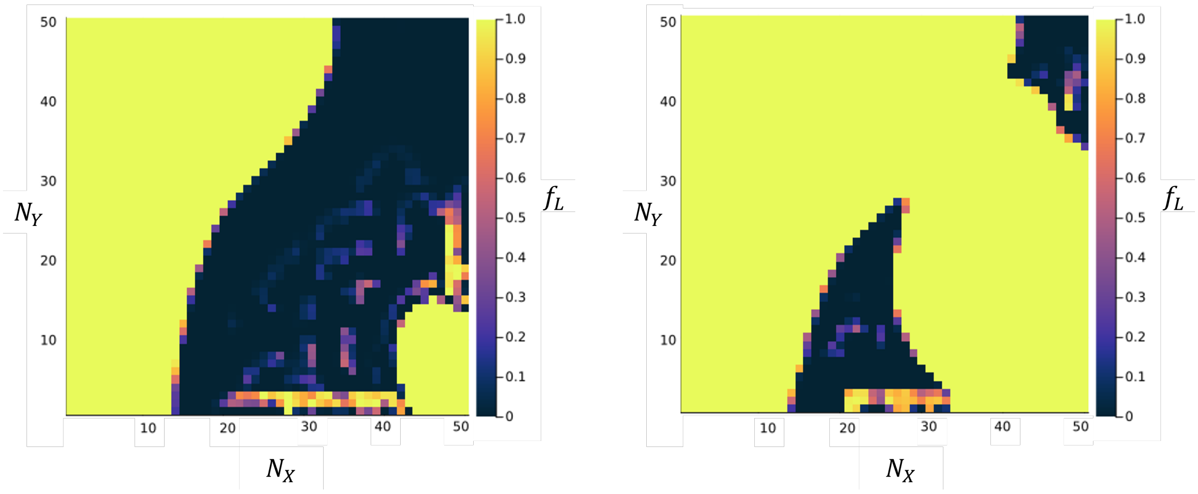
(a) $\theta = 0.5$ **(b)** $\theta = 1$

Figure 6.3: Liquid fraction for enthalpy melting with $\mathrm{Ra} = 1.7 \cdot 10^5$ for corners that were not treated special.



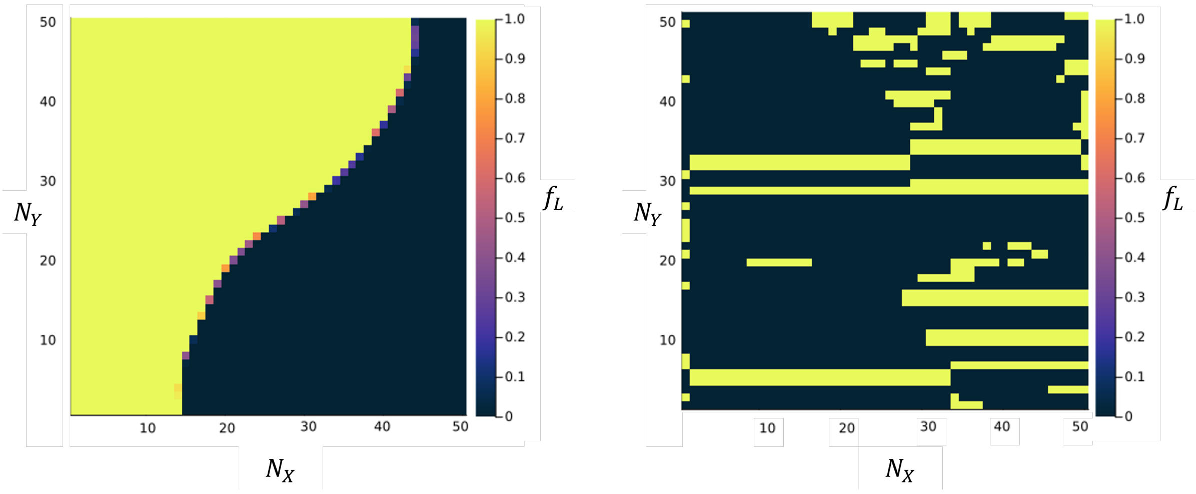
(a) $\theta = 0.5$ **(b)** $\theta = 0.91$

Figure 6.4: Liquid fraction for enthalpy melting with $\mathrm{Ra} = 1.7 \cdot 10^5$ for vertically oriented corners that follow dirichlet BC over adiabatic BC



(a) $\theta = 0.34$ **(b)** $\theta = 0.5$

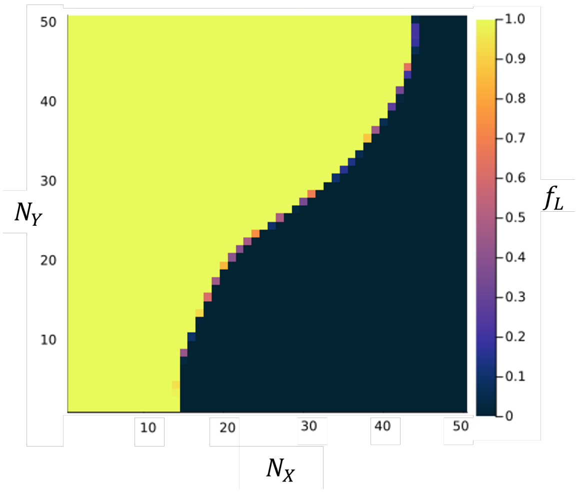
Figure 6.5: Quantitative results for enthalpy melting benchmark with $\mathrm{Ra} = 5 \cdot 10^4$ for horizontally oriented corners that always follow adiabatic BC



(a) $\theta = 0.5$ **(b)** $\theta = 0.82$

Figure 6.6: Quantitative results for enthalpy melting benchmark with $\mathrm{Ra} = 5 \cdot 10^4$ for diagonally oriented corners that follow adiabatic BC over Dirichlet BC

Figure 6.7 shows the results for diagonal corners that include Dirichlet boundary condition on the left side.

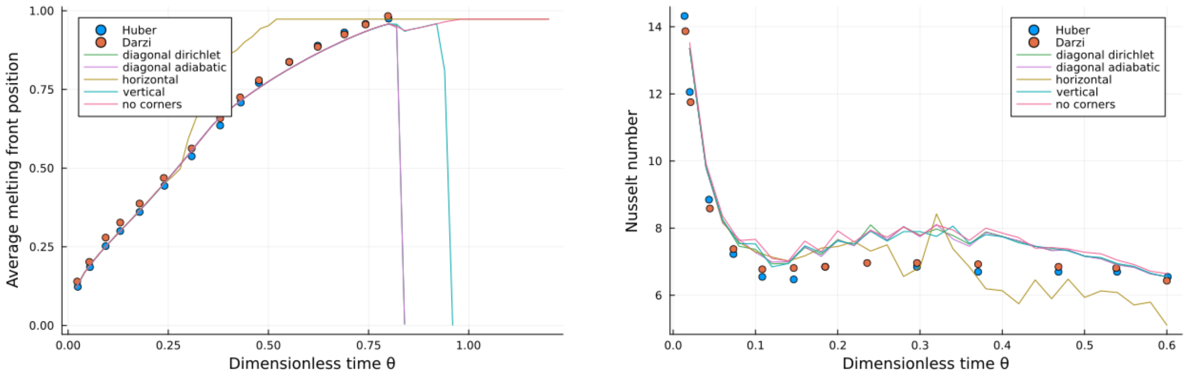


(a) $\theta = 0.5$ **(b)** $\theta = 0.82$

Figure 6.7: Qualitative results for enthalpy melting benchmark with $\mathrm{Ra} = 5 \cdot 10^4$ for diagonally oriented corners that follow Dirichlet BC over adiabatic BC

Both versions of the diagonally oriented corners become unstable at the exact same time. This seems to point at a problem with diagonal orientation, regardless of how it is implemented. Early experimental use of diagonal orientation was performed on the first benchmark as well since it seemed to be the logical option. In the first benchmark the diagonal corner orientation method produced more unstable results as well.

6.1.4. Quantitative results



(a) The results for the average melting front over dimensionless time θ.

(b) The results for the Nusselt number at the left wall over dimensionless time θ.

Figure 6.8: Quantitative results for enthalpy corner method differences with $\mathrm{Ra} = 1.7 \cdot 10^5$

In figure 6.8 a the instabilities and their point of incursion can be seen quickly. Both diagonal methods and the vertical method become completely unstable, but only for high θ. The reason for diagonal and horizontal failing seems logical since theses corner methods are not able to account for both BC correctly. The reason why the vertically oriented method fails is more peculiar. The only difference the vertical method has with the no corner method is only the order in which the boundaries are computed. If this is enough difference, perhaps the experiment was too volatile for concrete results.

The quantitative Nusselt results show not much difference in results, except for the horizontal method which became unstable quickly enough to show up as unstable in figure 6.8 b as well. All other simulations were run using no special corner implementation, since it was deemed the most stable.

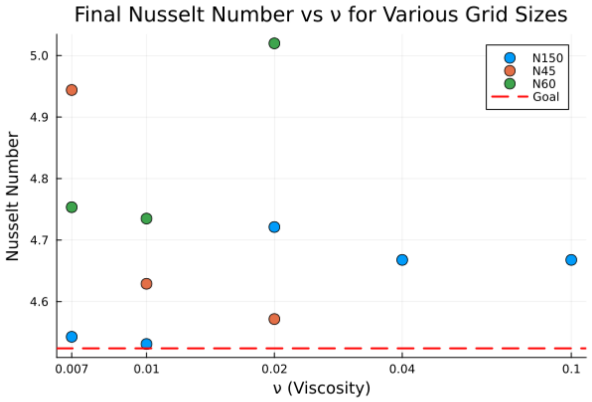


Figure 6.9: The Nusselt number reached for different grid sizes and viscosity values for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale.

6.2. Free parameters

In section 3.1.2 some background regarding important numerical stability numbers is given. These numbers can be influenced with 2 settings. The grid size N and the lattice kinematic viscosity ν. For testing the limits of the FMLBM stability, the temperature NC benchmark was used with a Rayleigh number of 10^5. The results showed little error with the benchmark results and showed no read sources of instability in the qualitative results.

To test the stability and performance of choosing the free parameters, the first benchmark is revisited from section 4.1 The following table shows grid sizes and viscosity values used to test stability, accuracy and performance of the FMLBM.

Variable name	*Lattice parameter*	*Possible values*
Kinemati viscosity	ν	$1/10, 1/25, 1/50, 1/100, 1/150$
Grid size	N	$45, 60, 150$

Table 6.1: Lattice parameters chosen for stability and accuracy measurements.

Data that was omitted from figure 6.9 was unstable and produced very nonphysical results. This was the case for N=60 and N=45 for viscosity $\nu \geq 0.04$. When looking at the figure it clearly shows that a higher grid size performs better. Coming closer to the goal Nusselt number set by the benchmark. It does seem that there is not a clear correlation in viscosity and accuracy. Although for higher viscosity the simulation does become unstable. A parameter that is influenced by both the kinematic viscosity and the grid size is the time conversion constant. This constant is defined as

$$C_{\text{time}} = \frac{\left(\frac{L_X}{N_X}\right)^2}{\frac{\nu_{\text{phys}}}{\nu}} = \frac{L_X^2 \cdot \nu}{N_X^2 \cdot \nu_{\text{phys}}} \tag{6.1}$$

Where L_X is the length of the cavity, N_X is the grid size and ν and ν_{phys} are the lattice viscosity and real viscosity respectively. In figure 6.10 the results are plotted against C_{time} to see if there is any clear correlation there.

Even though there is not a clear correlation that can be found between the error and C_{time}, there is a clear cutoff from where the simulation starts to become unstable. This occurs first for $N = 60$ and $\nu = 0.04$ with a $C_{time} = 1.30 \cdot 10^{-5}$. This can be seen as the minimum Δt time step before the Courant number would get too high. The Courant number from equation 3.4 depends on the timestep, the maximum velocity v and the cell size Δx. All unstable results obtained a Courant number higher than 41.2. The Courant numbers were calculated using a maximum velocity of 68.1 as this was the maximum velocity obtained by Vahl Davis for $Ra = 10^{-5}$ [48]. Most results that obtained a stable solution had a Courant number higher than 1. Only a grid size 150 × 150 with a kinematic viscosity lower than 1/100 had a Courant number lower than 1. This shows that one can reach stable solutions

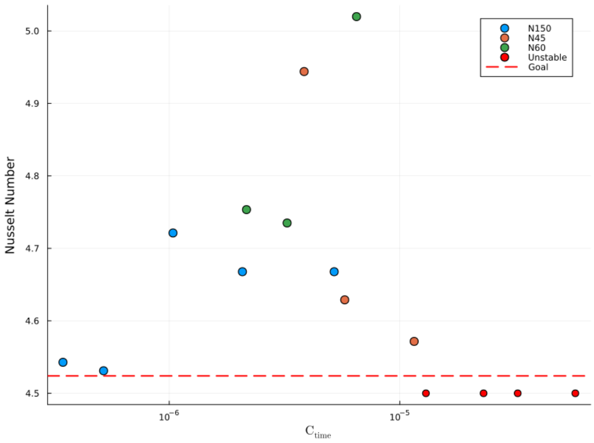


Figure 6.10: The Nusselt number reached for different grid sizes and C_{time} for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale. Unstable solutions are represented by red dots just below the benchmark Nusselt line.

while having a Courant higher than 1. It can be seen that the simulations using a Courant number lower than 1 scored significantly better.

All of these grid sizes and viscosity values also have a calculation speed associated with them. Instead of LUMPS which calculate the lattice updates per second, a better statistic would be Simulation Time Per Real Time or (STPRT).

$$STPRT = \frac{T_{sim}}{t_{real}} C_{time} = \Lambda C_{time} \tag{6.2}$$

Here T_{sim} is the amount of time steps the simulation has taken in an allotted time t_{real} which is simplified as the factor Λ. Λ is dependent on the grid size and is hardware dependent.

Table 6.2 shows the Λ factor per grid size.

Grid size	Λ
45×45	210.5
60×60	123.5
150×150	32.3

Table 6.2: The associated Λ factor for different grid sizes for the natural convection benchmark.

Figure 6.11 shows the STPRT compared to the Nusselt number obtained. Once again, there is no clear relation.

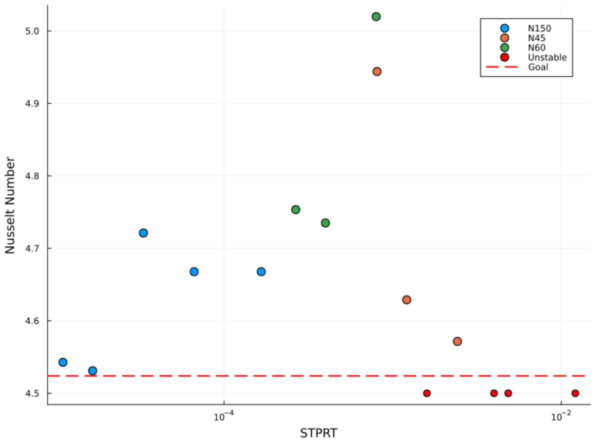


Figure 6.11: The Nusselt number for different grid sizes and STPRT for a heated square cavity following the temperature natural convection benchmark with $Ra = 10^5$. The x-axis is on a logarithmic scale.Unstable solutions are represented by red dots just below the benchmark Nusselt line.

7

Conclusions

The two original research questions were:

- Which physical parameters influence the melting time of a freeze plug, and how can these be adjusted to minimize melting duration?
- How can the boundary treatment and overall stability of the filter-matrix lattice Boltzmann method be improved or optimized?

The first question could sadly not be answered properly. Most used benchmarks gave results that were not completely valid. The first benchmark produced good results except for low Rayleigh numbers with little convective driving forces. The conjugated heat benchmark showed a well working conjugated heat boundary condition, since it showed good corresponding results with the benchmark except for again, the low Rayleigh numbers.

The results for the third benchmark for melting with natural convection showed no issues in lower Rayleigh domains. This suggest that the issue lies somewhere in the function for the collision/propogation/boundary with the temperature distribution function. Slight deviations in Nusselt number with the benchmark results occur, even though results for melting front were accurate. A slightly higher Nusselt number was also found by L. Borstlap, she commented on this phenomenom by suggesting that there exists a heat sink at the boundary due to the phase change. The LBM method spreads this sink to neighbouring cells resulting in a higher temperature difference, more flow and thus a higher Nusselt number.

The FMLBM freeze plug model without natural convection showed good qualitative results in the temperature isolines which compared well to those of the benchmark. The more precise temperature thermometers also gave good results. However, when the natural convection was implemented. The compilation times became extremely long. In the end the model did not achieve stable solutions.

The results for using different corner methods showed corners behaved generally unstable when enforced with explicit rules. The worst results came from the corners with diagonal normals (both cases) and the corners that behaved only as adiabatic extensions. This is backed up by mostly by the mathematical way the boundary temperature is decided at the boundary. It is advised to not set special rules for corners, but include them in the boundaries just as is. Since it is not necessary for accuracy and when applied otherwise instability is quickly found.

Different grid resolutions and kinematic viscosity values were tested to investigate their impact on performance, accuracy, and numerical stability. It was observed that the simulation consistently became unstable when the time conversion parameter C_{time} dropped below $1.3 \cdot 10^{-5}$. Although stable solutions could still be obtained for Courant numbers greater than 1, these cases generally exhibited significantly larger errors.

8

Recommendations

The following is a list of reccomendations for future research regarding both the Freeze Plug, and the FMLBM method. First some reccomendations that could be considered when researching the freeze plug for use in a molten salt reactor.

- The boundary conditions applied to model the freeze plug are adequate for modelling but could be made more realistic. For example, currently all boundaries are considered to be no-slip. However, the upper boundary that is currently a no-slip transient dirichlet boundary is not actually no-slip in practice. The point where the boundary is should contain more heated salt. Currently a small layer of salt is already molten at $t = 0$, however, this layer may be too small and the no-slip boundaries might induce a velocity dead-point in the corner where there should be none. So it could be researched if this layer is big enough / should be bigger. Or if the upper boundary should not be made no-slip. This change was also made by Pater and Kaaks in their final stage of modelling the freeze plug. [47]
- The freeze plug model still lacks many attributes before becoming entirely realistic. Currently the PCM is assumed to not change volume and thermal conductivity with a change in temperature. Generally the thermal conductivity can either increase or decrease with temperature depending on the composition. [66] The PCM salt currently has the same density when liquid as when solid, but the salt it is modeled after has a solid density of 4502 kgm^{-3} and a fluid density of 4390 kgm^{-3}.[63] Furthermore, lack of turbulence modelling makes the model less realistic as well. Methods for simulating turbulent flow with the FMLBM do already exist. [67] [68]
- Still, physical freeze plug parameters need to be researched. Possible physical parameters include: angle of incidence, choice of grated design, size of the freeze plug, material choice and many more options are still available.

Second, there are some reccomendations for use of the filter-matrix lattice boltzmann method for use in fluid dynamics simulations.

- When building a code base that is based on the one from this thesis, one should consider revisiting the early numerical benchmark to figure out issues with convective heat transport in low Rayleigh numbers while using the temperature distribution function. The discrepency between the benchmark cases and obtained results indicates possible further issues. This should first be solved.
- The performance, stability and accuracy indicators of chapter 6 can be used to compare the FMLBM with other lattice Boltzmann methods such as the SRT or MRT methods. This would allow for a good numerical comparison between these different collision schemes to see which scheme operates better and by how much.
- GPU parallelization. When dealing with larger grid sizes like the one used for the freeze plug natural convection case, it would be a lot faster to use GPU parallelization. GPU parallelization causes the computations to be done on all cells at once instead of one by one. This is especially helpful when running larger grids. When running the first natural convection benchmark GPU

parallelization allowed for 100 MLUPS to be reached with a grid of 300x300 [65]. This study only reached 1.4 MLUPS on the same grid and benchmark.

- To further speed up the process another scheme could be considered, such as D2Q5. Besides being faster due to less directions needing to be computated, it also shows better accuracy and convergence when the convection term is small. [69]

References

[1] Martin Rohde. *AP 3352, Rohde 2*. 2023.

[2] D. M. Barton et al. “Measurement of the Uranium-235 Fission Cross Section over the Neutron Energy Range 1 to 6 MeV”. In: *Nuclear Science and Engineering* 60.4 (1976), pp. 369–382. DOI: 10.13182/NSE76-A26898.

[3] Office of nuclear Energy. 2019. URL: https://www.energy.gov/ne/articles/infographic-how-does-pressurized-water-reactor-work.

[4] Jérôme Serp et al. “The molten salt reactor (MSR) in generation IV: Overview and perspectives”. In: *Progress in Nuclear Energy* 77 (2014), pp. 308–319. ISSN: 0149-1970. DOI: https://doi.org/10.1016/j.pnucene.2014.02.014. URL: https://www.sciencedirect.com/science/article/pii/S0149197014000456.

[5] Darryl D. Siemer. “Why the molten salt fast reactor (MSFR) is the best Gen IV reactor”. In: *Energy Science & Engineering* 3.2 (2015), pp. 83–97. DOI: https://doi.org/10.1002/ese3.59. URL: https://scijournals.onlinelibrary.wiley.com/doi/abs/10.1002/ese3.59.

[6] Benito Mignacca and Giorgio Locatelli. “Economics and finance of Molten Salt Reactors”. In: *Progress in Nuclear Energy* 129 (2020), p. 103503. ISSN: 0149-1970. DOI: https://doi.org/10.1016/j.pnucene.2020.103503. URL: https://www.sciencedirect.com/science/article/pii/S0149197020302511.

[7] World Nuclear News. 2023. URL: https://www.world-nuclear-news.org/Articles/Operating-permit-issued-for-Chinese-molten-salt-re.

[8] M. Brovchenko et al. “Design-Related Studies for the Preliminary Safety Assessment of the Molten Salt Fast Reactor”. In: *Nuclear Science and Engineering* 175.3 (2013), pp. 329–339. DOI: 10.13182/NSE12-70. URL: https://doi.org/10.13182/NSE12-70.

[9] Julien Giraud et al. “Development of a cold plug valve with fluoride salt”. In: *EPJ N - Nuclear Sciences & Technologies* 5 (2019), p. 9. DOI: 10.1051/epjn/2019005. URL: https://hal.science/hal-02277836.

[10] T.D. Schafer. *Design and Melting Behavior of the MSFR Freeze Plug*. 2018. URL: https://repository.tudelft.nl/record/uuid:a2032f51-6948-40f5-ba35-5b40309d1fa2.

[11] P. Swaroop. *IDesign of a Freeze Plug for the Molten Salt Fast Reactor(MSFR)*. 2016. URL: https://d1wqtxts1xzle7.cloudfront.net/81470592/MSc_Parth_Swaroop-libre.pdf?1646070042=&response-content-disposition=inline%3B+filename%3DDesign_of_a_Freeze_Plug_for_the_Molten_S.pdf&Expires=1725487062&Signature=0FTN2XGLXSchXFRwuKtef3fER2Otca9Vyt2KNooMAb0NFwUS-UhZOR4pMXzpyJpnKazUdn-tbiFv5AePh6DA9Sa4dVL4aT7dlxlnlm60a6tvZhklaKOTVODvD26OUG25if6NAYrHqAFs0qBkqsPf7fFHh-F7HjAqp41J6UH7AX~F~9lu2wXLA7K~fH7T7Ij~2nlxh5Xg0IYv-EOQCp1LN8ezDo9EKzS7JuEZjYmHbf38pLnKIzOiQmMTmR9-021g7fCdakUDjDHq-XOLbNW95LwMDI-jc8oA7uQMAQQOIrQudB-zYctqtaDx00qsbvZVgFVe1kmVxZsjbgQUUayJoA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA.

[12] André Makkinje. 2017. URL: https://filelist.tudelft.nl/TNW/Afdelingen/Radiation%20Science%20and%20Technology/Research%20Groups/RPNM/Publications/BSc_Andre_Makkinje.pdf.

[13] I.K. Aji. *INVESTIGATION OF BASIC PARAMETERS IN DEVELOPING HIGH- PERFORMANCE FREEZE VALVE FOR MOLTEN SALT REACTOR*. 2020.

[14] C.Z. van Dijk. *Investigating the Effect of Freeze Plug Inclination Angle Melting Behaviour with the Linearised Enthalpy Method*. 2023. URL: https://repository.tudelft.nl/record/uuid:be35538b-cb4b-4317-be22-f165a461ecab.

[15] J.W.A. Reus. *A numerical investigation of freeze-valve melting based on a linearised enthalpy method.* 2021.

[16] Lotte Borstlap. *Developing a Filter-Matrix Lattice Boltzmann Boundary Method for Conjugate Heat Transfer and Phase Change.* 2024. URL: `https://martinrohde.nl/theses/msc/MSc_Lotte_Borstlap.pdf`.

[17] Aniruddha Majumder. "Chapter One - LBM for separation processes". In: *Lattice Boltzmann Modeling for Chemical Engineering.* Ed. by Harry E.A. Van den Akker. Vol. 55. Advances in Chemical Engineering 1. Academic Press, 2020, pp. 1–79. DOI: `https://doi.org/10.1016/bs.ache.2020.04.002`. URL: `https://www.sciencedirect.com/science/article/pii/S0065237720300028`.

[18] J.G.M. Eggels and J.A. Somers. "Numerical simulation of free convective flow using the lattice-Boltzmann scheme". In: *International Journal of Heat and Fluid Flow* 16.5 (1995), pp. 357–364. ISSN: 0142-727X. DOI: `https://doi.org/10.1016/0142-727X(95)00052-R`. URL: `https://www.sciencedirect.com/science/article/pii/0142727X9500052R`.

[19] Bruce Roy Munson. *Fundamentals of Fluid Mechanics 7th Ed.* John Wiley and Sons, 2013.

[20] Kamil Wichterle and Marek Vecer. *Transport and surface phenomena.* Elsevier, 2020.

[21] Mahmoud M. El-Gendi. "Numerical simulation of unsteady natural convection flow inside a pattern of connected open square cavities". In: *International Journal of Thermal Sciences* 127 (2018), pp. 373–383. ISSN: 1290-0729. DOI: `https://doi.org/10.1016/j.ijthermalsci.2018.02.008`. URL: `https://www.sciencedirect.com/science/article/pii/S1290072917310049`.

[22] Min Chan Kim and Sin Kim. "The onset of natural convection and heat transfer correlation in horizontal fluid layer heated uniformly from below". In: *KSME International Journal* 15 (2001), pp. 1451–1460. DOI: `10.1007/BF03185687`.

[23] J. H. Lienhard V and J. H. Lienhard IV. *A Heat Transfer Textbook.* 6th. Version 6.00. Cambridge, MA: Phlogiston Press, 2024. 810 pp. URL: `https://ahtt.mit.edu`.

[24] SUMAN CHAKRABORTY and DIPANKAR CHATTERJEE. "An enthalpy-based hybrid lattice-Boltzmann method for modelling solidliquid phase transition in the presence of convective transport". In: *Journal of Fluid Mechanics* 592 (2007), pp. 155–175. DOI: `10.1017/S0022112007008555`.

[25] Rongzong Huang, Huiying Wu, and Ping Cheng. "A new lattice Boltzmann model for solidliquid phase change". In: *International Journal of Heat and Mass Transfer* 59 (2013), pp. 295–301. ISSN: 0017-9310. DOI: `https://doi.org/10.1016/j.ijheatmasstransfer.2012.12.027`. URL: `https://www.sciencedirect.com/science/article/pii/S0017931012009830`.

[26] Celeke Bus, Thorben Besseling, and Martin Rohde. *A Filter-Matrix Lattice-Boltzmann Methodology for Convective Melting and Solidification.* 2025. DOI: `https://doi.org/10.1002/fld.70001`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.70001`.

[27] Celeke Bus. *Simulating the transient freezing in cooled non-eutectic molten salt channel flow.* Master's thesis, TU Delft, Reactor Institute Delft. Available from TU Delft library or Reactor Institute Delft repository. 2022.

[28] Timm Krüger et al. *The Lattice Boltzmann method.* Springer, 2016.

[29] Goncalo Silva and Viriato Semiao. "Truncation errors and the rotational invariance of three-dimensional lattice models in the lattice Boltzmann method". In: *Journal of Computational Physics* 269 (2014), pp. 259–279. ISSN: 0021-9991. DOI: `https://doi.org/10.1016/j.jcp.2014.03.027`. URL: `https://www.sciencedirect.com/science/article/pii/S0021999114002083`.

[30] Qinjun Kang, Peter C. Lichtner, and Dongxiao Zhang. "An improved lattice Boltzmann model for multicomponent reactive transport in porous media at the pore scale". In: *Water Resources Research* 43.12 (2007). DOI: `https://doi.org/10.1029/2006WR005551`. URL: `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2006WR005551`.

[31] Li Wang, Zhengliang Liu, and Methma Rajamuni. "Recent progress of lattice Boltzmann method and its applications in fluid-structure interaction". In: *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 237.11 (2023), pp. 2461–2484. DOI: `10.1177/09544062221077583`. URL: `https://doi.org/10.1177/09544062221077583`.

[32] Q. Li et al. "Coupled double-distribution-function lattice Boltzmann method for the compressible Navier-Stokes equations". In: *Phys. Rev. E* 76 (5 2007), p. 056705. DOI: `10.1103/PhysRevE.76.056705`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.76.056705`.

[33] R.R. Nourgaliev et al. "The lattice Boltzmann equation method: theoretical interpretation, numerics and implications". In: *International Journal of Multiphase Flow* 29.1 (2003), pp. 117–169. ISSN: 0301-9322. DOI: `https://doi.org/10.1016/S0301-9322(02)00108-8`. URL: `https://www.sciencedirect.com/science/article/pii/S0301932202001088`.

[34] Sauro Succi. *The Lattice Boltzmann equation for fluid dynamics and beyond.* Oxford University Press, 2001. DOI: `10.1093/oso/9780198503989.001.0001`. URL: `https://doi.org/10.1093/oso/9780198503989.001.0001`.

[35] Randall LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems (Classics in Applied Mathematics Classics in Applied Mathemat).* SIAM, 2007. URL: `https://dl.acm.org/citation.cfm?id=1355322`.

[36] R. Courant, K. Friedrichs, and H. Lewy. "Uber die partiellen Differenzengleichungen der mathematischen Physik". In: *Mathematische Annalen* 100.1 (1928), pp. 32–74. DOI: `10.1007/bf01448839`. URL: `https://doi.org/10.1007/bf01448839`.

[37] P. L. Bhatnagar, E. P. Gross, and M. Krook. "A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems". In: *Phys. Rev.* 94 (3 1954), pp. 511–525. DOI: `10.1103/PhysRev.94.511`. URL: `https://link.aps.org/doi/10.1103/PhysRev.94.511`.

[38] Junjian Wang et al. "The lattice Boltzmann method for isothermal micro-gaseous flow and its application in shale gas flow: A review". In: *International Journal of Heat and Mass Transfer* 95 (2016), pp. 94–108. ISSN: 0017-9310. DOI: `https://doi.org/10.1016/j.ijheatmasstransfer.2015.12.009`. URL: `https://www.sciencedirect.com/science/article/pii/S0017931015306566`.

[39] Rui Du, Baochang Shi, and Xingwang Chen. "Multi-relaxation-time lattice Boltzmann model for incompressible flow". In: *Physics Letters A* 359.6 (2006), pp. 564–572. ISSN: 0375-9601. DOI: `https://doi.org/10.1016/j.physleta.2006.07.074`. URL: `https://www.sciencedirect.com/science/article/pii/S037596010601173X`.

[40] Martin Rohde. *The filter-matrix lattice-Boltzmann method.* 2025.

[41] Congshan Zhuo, Chengwen Zhong, and Jun Cao. "Filter-matrix lattice Boltzmann model for incompressible thermal flows". In: *Phys. Rev. E* 85 (4 2012), p. 046703. DOI: `10.1103/PhysRevE.85.046703`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.85.046703`.

[42] Thorben Besseling. *Investigating Adaptive Mesh Refinement Criteria for a Double-Distribution FMLB Scheme in Melting and Solidifaction Processes.* 2022.

[43] Martin Geier and Martin Schönherr. "Esoteric Twist: An Efficient In-Place Streaming Algorithmus for the Lattice Boltzmann Method on Massively Parallel Hardware". In: *Computation* 5.2 (2017), p. 19. DOI: `10.3390/computation5020019`.

[44] P. A. Skordos. "Initial and boundary conditions for the lattice Boltzmann method". In: *Phys. Rev. E* 48 (6 1993), pp. 4823–4842. DOI: `10.1103/PhysRevE.48.4823`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.48.4823`.

[45] Robert S. Maier, Robert S. Bernard, and Daryl W. Grunau. "Boundary conditions for the lattice Boltzmann method". In: *Physics of Fluids* 8.7 (1996), pp. 1788–1801. ISSN: 1070-6631. DOI: `10.1063/1.868961`. eprint: `https://pubs.aip.org/aip/pof/article-pdf/8/7/1788/19301012/1788\_1\_online.pdf`. URL: `https://doi.org/10.1063/1.868961`.

[46] Martin Hecht and Jens Harting. "Implementation of on-site velocity boundary conditions for D3Q19 lattice Boltzmann simulations". In: *Journal of Statistical Mechanics: Theory and Experiment* 2010.01 (2010), P01018. DOI: `10.1088/1742-5468/2010/01/P01018`. URL: `https://dx.doi.org/10.1088/1742-5468/2010/01/P01018`.

[47] Mateusz Pater et al. “A numerical benchmark for modelling phase change in molten salt reactors”. In: *Annals of Nuclear Energy* 194 (2023), p. 110093. ISSN: 0306-4549. DOI: `https://doi.org/10.1016/j.anucene.2023.110093`. URL: `https://www.sciencedirect.com/science/article/pii/S0306454923004127`.

[48] G. De Vahl Davis. “Natural convection of air in a square cavity: A bench mark numerical solution”. In: *International Journal for Numerical Methods in Fluids* 3.3 (1983), pp. 249–264. DOI: `https://doi.org/10.1002/fld.1650030305`. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.1650030305`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1650030305`.

[49] Robert M Kerr. “Rayleigh number scaling in numerical convection”. In: *Journal of Fluid Mechanics* 310 (1996), pp. 139–179.

[50] Yunus A. Cengel. *Introduction to Thermodynamics and Heat Transfer*. McGraw-Hill Higher Education, 2009.

[51] Philip J. Davis and Philip Rabinowitz. *Methods of Numerical Integration*. Academic Press, 1975.

[52] D.A. Kaminski and C. Prakash. “Conjugate natural convection in a square enclosure: effect of conduction in one of the vertical walls”. In: *International Journal of Heat and Mass Transfer* 29.12 (1986), pp. 1979–1988. ISSN: 0017-9310. DOI: `https://doi.org/10.1016/0017-9310(86)90017-7`. URL: `https://www.sciencedirect.com/science/article/pii/0017931086900177`.

[53] D. Misra and A. Sarkar. “Finite element analysis of conjugate natural convection in a square enclosure with a conducting vertical wall”. In: *Computer Methods in Applied Mechanics and Engineering* 141.3 (1997), pp. 205–219. ISSN: 0045-7825. DOI: `https://doi.org/10.1016/S0045-7825(96)01109-7`. URL: `https://www.sciencedirect.com/science/article/pii/S0045782596011097`.

[54] R. E. White. “An Enthalpty Formulation of the Stefan Problem”. In: *SIAM Journal on Numerical Analysis* 19.6 (1982), pp. 1129–1157. DOI: `10.1137/0719082`. eprint: `https://doi.org/10.1137/0719082`. URL: `https://doi.org/10.1137/0719082`.

[55] Christian Huber et al. “Lattice Boltzmann model for melting with natural convection”. In: *International Journal of Heat and Fluid Flow* 29.5 (2008), pp. 1469–1480. ISSN: 0142-727X. DOI: `https://doi.org/10.1016/j.ijheatfluidflow.2008.05.002`. URL: `https://www.sciencedirect.com/science/article/pii/S0142727X08000957`.

[56] Mahmoud Jourabian, Mousa Farhadi, and Ahmad Ali Rabienataj Darzi. “Convection-dominated melting of phase change material in partially heated cavity: lattice Boltzmann study”. In: *Heat and Mass Transfer* 49.4 (2012), pp. 555–565. DOI: `10.1007/s00231-012-1102-y`. URL: `https://doi.org/10.1007/s00231-012-1102-y`.

[57] Xiaowen Shan. “Simulation of Rayleigh-Benard Convection Using a Lattice Boltzmann Method”. In: *Phys. Rev. E* 55 (1997). DOI: `10.1103/PhysRevE.55.2780`.

[58] Zhaoli Guo and T. Zhao. “Lattice Boltzmann simulation of natural convection with temperature-dependent viscosity in a porous cavity”. In: *Progress in Computational Fluid Dynamics - PROG COMPUT FLUID DYN* 5 (2005). DOI: `10.1504/PCFD.2005.005823`.

[59] Chun-Pao Kuo and Wen-Shu Jiaung Jeng-Rong Ho. “LATTICE BOLTZMANN METHOD FOR THE HEAT CONDUCTION PROBLEM WITH PHASE CHANGE”. In: *Numerical Heat Transfer, Part B: Fundamentals* 39.2 (2001), pp. 167–187. DOI: `10.1080/10407790150503495`. eprint: `https://doi.org/10.1080/10407790150503495`. URL: `https://doi.org/10.1080/10407790150503495`.

[60] W. Miller and S. Succi. “A Lattice Boltzmann Model for Anisotropic Crystal Growth from Melt.” In: *Journal of Statistical Physics* 107.1/2 (2002), pp. 173–186. DOI: `10.1023/a:1014510704701`. URL: `https://doi.org/10.1023/a:1014510704701`.

[61] Dingyu Lu et al. “Study on Influencing Factors of Phase Transition Hysteresis in the Phase Change Energy Storage”. In: *Materials* 15.8 (2022). ISSN: 1996-1944. DOI: `10.3390/ma15082775`. URL: `https://www.mdpi.com/1996-1944/15/8/2775`.

[62] Peter Jany and Adrian Bejan. “Scaling theory of melting with natural convection in an enclosure”. In: *International Journal of Heat and Mass Transfer* 31.6 (1988), pp. 1221–1235. ISSN: 0017-9310. DOI: `https://doi.org/10.1016/0017-9310(88)90065-8`. URL: `https://www.sciencedirect.com/science/article/pii/0017931088900658`.

[63] Marco Tiberga et al. “Preliminary investigation on the melting behavior of a freeze-valve for the Molten Salt Fast Reactor”. In: *Annals of Nuclear Energy* 132 (2019), pp. 544–554. ISSN: 0306-4549. DOI: `https://doi.org/10.1016/j.anucene.2019.06.039`. URL: `https://www.sciencedirect.com/science/article/pii/S0306454919303573`.

[64] Mauricio Tano et al. “Multiphysics study of the draining transients in the Molten Salt Fast Reactor”. In: *Proceedings of the 2018 International Congress on Advances in Nuclear Power Plants, ICAPP*. 2018, pp. 469–478.

[65] Thomas Pijls. *Developing a GPU-Accelerated Filter-Matrix Lattice Boltzmann Multiphysics Tool for the Transient Thermal-Hydraulics and Neutronics of a Molten Salt Fast Reactor Core*. 2025. URL: `https://martinrohde.nl/theses/msc/MSc_Thomas_Pijls.pdf`.

[66] Huiqiang Yang et al. “Development of a molten salt thermal conductivity model and database for advanced energy systems”. In: *Solar Energy* 256 (2023), pp. 158–178. ISSN: 0038-092X. DOI: `https://doi.org/10.1016/j.solener.2023.04.009`. URL: `https://www.sciencedirect.com/science/article/pii/S0038092X23002359`.

[67] Congshan Zhuo and Chengwen Zhong. “LES-based filter-matrix lattice Boltzmann model for simulating turbulent natural convection in a square cavity”. In: *International Journal of Heat and Fluid Flow* 42 (2013), pp. 10–22. ISSN: 0142-727X. DOI: `https://doi.org/10.1016/j.ijheatfluidflow.2013.03.013`. URL: `https://www.sciencedirect.com/science/article/pii/S0142727X13000702`.

[68] Pieter van der Spek. *GPU-accelerated Large Eddy Simulation of non-eutectic MSFR Salt Freezing in Turbulent Channel Flow*. 2024. URL: `https://martinrohde.nl/theses/msc/MSc_Pieter_van_der_Spek.pdf`.

[69] Like Li, Renwei Mei, and James 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`.

A

Appendix

A.1. Algorithm benchmarks

A.1.1. Natural convection benchmark

- Initialize the distribution functions in the PCM domain for density $f_i(x,t)$ and enthalpy $m_i(x,t)$ using chosen macroscopic quantities.
- Initialize the distribution functions in the wall domain for density $f_i(x,t)$ and temperature $g_i(x,t)$ using chosen macroscopic quantities.
- Initialize enthalpy properties such as the liquid fraction f_L and the total enthalpy of all cells using $m_i(x,t)$.
- For each time step `t` from `0` to `T`, do:

  1. **Collision step PCM domain**
     - Determine solution vector $\beta_i^-(x,t)$ using equation 3.19 and $g_i(x,t)$
     - Translate $\beta_i^-(x,t)$ into $\beta_i^+(x,t)$ 3.23
     - Get the post collision enthalpy distributions from $\beta_i^+(x,t)$ 3.22
  2. **Propogate** $g_i(x,t)$
  3. **Apply temperature Neumann boundary condition edges PCM domain** (See steps from section 3.4.2
  4. **Apply temperature Dirichlet boundary condition edge PCM domain**
  5. **Apply temperature boundary conditions to corners on the PCM domain**
  6. **Collision step PCM domain for density distributions**
     - Determine solution vector $\alpha_i^-(x,t)$ using equation 3.19 and $m_i(x,t)$
     - Determine acting body forces due to natural convection 2.7
     - Determine velocities 3.57
     - Translate $\alpha_i^-(x,t)$ into $\alpha_i^+(x,t)$ 3.56
  7. **Apply density boundary conditions to edges on the PCM domain**
  8. **Apply density boundary conditions to corners on the PCM domain**
  9. **For each nth time step `t`**
     (a) retrieve all relevant macroscopic values for data and intermediary plots

- End simulation when past maximum time steps OR when a converging solution is found OR when the part of the freeze plug has melted completely through in freeze plug simulations.