Imperial College of Science, Technology and Medicine
Department of Mechanical Engineering

Computational Fluid Dynamics Simulation of Corium Spreading

Hugo Raphaël Perrier

Thesis submitted for the degree of
Doctor of Philosophy at Imperial College London
July 2018

Abstract

In the event of a severe accident in a nuclear reactor involving reactor core degradation, molten core materials called Corium can form. The interaction of Corium with the concrete of reactor containment buildings may threaten the integrity of nuclear power plants. A technical solution to prevent the penetration of Corium into the concrete base-mat is the implementation of a core-catcher designed to collect the Corium and spread it over a large area to be cooled efficiently.

Due to the complexity of Corium spreading simulations, standard numerical methods used in Computational Fluid Dynamics (CFD) may not be applicable to Corium flows. Therefore, the work presented in this thesis investigates in detail the errors associated with the numerical simulation of Corium flows.

In particular, the numerical representation of the interface between two immiscible fluids, such as Corium and air, is challenging. This thesis identifies numerical diffusion of temperature, which is a numerical error associated with the spatial discretisation process, in the vicinity of the Corium/air interface, as a source of error for Corium spreading simulations. A numerical strategy is proposed and evaluated on several test cases with known solutions. It is shown that, compared to standard numerical methods, the proposed strategy significantly reduces numerical diffusion.

The numerical methods studied in this thesis are validated against experimental results from the VULCANO VE-U7 Corium spreading experiment. The simulation results obtained are compared to experimental results as well as other simulation results available in the scientific literature. It is shown that the choice of numerical methods affects the spreading predictions, however, the uncertainty of Corium properties is the major source of uncertainty in the simulation results.

Acknowledgements

I would like to acknowledge EPSRC and Rolls-Royce for their support of my industrial CASE studentship.

My thanks go to my supervisors Berend van Wachem and Matthew Eaton.

I owe a special gratitude to Fabian Denner. The completion of my thesis would not have been possible without his support, feedback and guidance throughout the past four years.

I am grateful to everyone in the MultiFlow Lab and in the Nuclear Engineering groups for the interesting discussions we had and the fun times we spent together. I hope we will be able to keep in touch for the years to come.

I want to acknowledge CEA and Christophe Journeau for sharing the VULCANO VE-U7 experimental results and helpful comments on the modeling of Corium spreading.

The many clubs and societies make the life at Imperial College very enjoyable and provide so much opportunities for the students. I wish the best of luck to everyone who has been involved in the Imperial College Data Science Society.

Finally, I wish to thank the Tresette team for their friendliness and cheerfulness. The deep conversations we had and the regular technical meetings we held drastically helped me develop my CFD skills. Their unconditional support have been critical to help me go through tough times.

Copyright Declaration

The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives license. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the license terms of this work.

Statement of Originality

I certify that the content of this thesis is the result of my own work expressed in my own words. To the best of my knowledge, this thesis does not contain material previously published or written by another person except where due references are made.

Abbreviations

BE Bercovier-Engelman

CDF Central Differencing Scheme

CL Contact Line

CEA Commissariat à l'énergie atomique et aux énergies alternatives

CFD Computational Fluid Dynamics

CFL CourantFriedrichsLewy

CICSAM Compressive Interface Capturing Scheme for Arbitrary Meshes

EPR European Pressurized Reactor / Evolutionary Pressurized Reactor

FVM Finite Volume Method

HC Hyper-C

IRSN Institut de Radioprotection et de Sûreté Nucléaire

IRWST Internal Refuelling Water Storage Tank

IVR In Vessel Retention

NPP Nuclear Power Plant

RHS Right hand side

RPV Reactor Pressure Vessel

TMI Three Mile Island

TVD Total Variation Diminishing

UDS Upwind Differencing Scheme

UQ Ultimate Quickest

VOF Volume-Of-Fluid

List of presentations and publications

The following is a list of presentations and publications that relate to this work.

- H. Perrier, M. D. Eaton, B G. M. van Wachem, *Review of in-vessel Corium melt pool modelling capabilities with commercial CFD codes.* Presented to the research group of Prof. Jeong, Nuclear Energy Conversion Lab, KAIST, Korea, 2015.

- H. Perrier, *Computational Fluid Dynamics Simulation of Corium Spreading.* Early Stage Assessment, presented at Imperial College London, 2016.

- H. Perrier, M. D. Eaton, V. Badalassi, B G. M. van Wachem, *Modelling of Corium Spreading in a Core Catcher.* Presented at the 9th International Conference on Multiphase Flow, Florence, 2016.

- H. Perrier, M. D. Eaton, V. Badalassi, B G. M. van Wachem, *New VOF Method with Two Temperature Fields.* Presented at the UK Fluids Conference 2016, Imperial College London, 2016.

- H. Perrier, M. D. Eaton, B G. M. van Wachem, *Numerical Modelling of Corium Spreading.* In Proc. 17th International Topical Meeting on Nuclear Thermal Hydraulics (NURETH-17), Xi'an, Shaanxi, China, 2017.

- H. Perrier, F. Denner, M. D. Eaton, B G. M. van Wachem, *On the numerical modelling of Corium spreading using Volume-of-Fluid methods.* Nuclear Engineering and Design 345, 216232, 2019. doi:10.1016/j.nucengdes.2019.02.013

- H. Perrier, F. Denner, M. D. Eaton, B G. M. van Wachem, *Numerical simulation of the VULCANO VE-U7 Corium spreading experiment using Volume-of-Fluid methods.* Publication prepared for submission to the Journal Nuclear Engineering and Design, 2018.

Contents

Abstract **i**

Acknowledgements **iii**

Copyright Declaration **iii**

Statement of Originality **v**

1 Introduction **1**

1.1 Introduction to Computational Fluid Dynamics 1

1.2 Introduction to corium spreading 3

1.3 The core-catcher concept 4

1.4 Motivation and objectives 8

1.5 Thesis outline 9

1.6 Contributions 10

2 Review of Corium spreading studies **13**

2.1 Experimental studies of prototypical corium 13

2.1.1 COMAS series of experiments 14

2.1.2 FARO series of experiments 15

2.1.3 VULCANO series of experiment 15

2.2 Numerical modelling of corium spreading 16

2.2.1 MELTSPREAD 18

2.2.2 THEMA 19

2.2.3 LAVA 20

2.2.4 CROCO 20

2.2.5 CORFLOW 21

3 Multiphase Computational Fluid Dynamics Framework 22

3.1 Governing equations 23

3.2 Discretisation of the governing equations 26

3.2.1 Time discretisation 27

3.2.2 Space discretisation 28

3.2.3 Interface discretisation 34

3.2.4 Discretisation of the energy equation 37

3.3 Solution procedure 38

3.3.1 Solution of a linear system 38

3.3.2 Solution of a non-linear system 41

3.3.3 Fully coupled solver 42

3.3.4 Solution of the energy equation 45

4 CFD simulation of Bingham Fluid 48

4.1 Bingham fluid model 49

4.2 Implementation of a Bingham fluid model in a CFD code 52

4.3 Test case: lid driven cavity 53

4.3.1 Newtonian fluid in a lid driven cavity 54

4.3.2 Bingham fluid in a lid driven cavity 56

5 Spreading of a fluid with temperature dependent properties 61

5.1 Solution procedure 63

5.2 Viscosity averaging methods 64

5.3 Advection of a hot fluid into a cold channel 66

5.3.1 Discretisation and solution of the energy equation 69

5.3.2 Comparison of analytical and computed viscosity at interface cells 73

5.3.3 Viscosity averaging 76

5.3.4 Viscosity smoothing 77

5.4 Spreading test case 81

5.4.1 Discretisation of the energy equation 83

5.4.2 Viscosity averaging and computation 86

5.4.3 Spreading with heat transfer between fluids 88

5.5 Conclusions 91

6 CFD simulation of the Vulcano Corium spreading experiment **93**

6.1 Description of the VULCANO VE-U7 experiment 97

6.2 Modelling of corium spreading 98

6.3 Numerical methods 102

6.3.1 Simulation domain 103

6.4 Initial and boundary conditions 105

6.5 Results 106

6.5.1 Simulation of a Couette Flow 106

6.5.2 VE-U7 simulation results 107

6.5.3 Comparison with the VULCANO VE-U7 experimental results 112

6.6 Estimation of heat loss by radiation 115

6.7 Conclusions 119

Thesis conclusions **121**

Bibliography **126**

List of Tables

4.1 Position and intensity of the principal vortex for the simulation of a Newtonian fluid in a lid driven cavity using several mesh resolutions and comparison with results from the literature 56

4.2 Results of the simulation of a Bingham fluid in a lid-driven cavity. The position and value of the velocity extrema along the centre lines $x = 0.5$ and $y = 0.5$ are given for different value of the yield stress τ_S. 60

5.1 Fluid properties of the two-fluids considered in the test case "advection of a hot fluid in a cold channel" 67

5.2 Description of the interface shape and calculation time for simulations using the three considered solution procedures. The flow time-step Δt_{Flow} required to reach convergence is also given. 71

5.3 Considered smoothing strategies for the computation of the viscosity. 78

6.1 Reference properties of corium and properties of Nitrogen at room temperature . 101

List of Figures

1.1 Schematic of the EPR core catcher. In the event of a nuclear severe accident, the reactor pressure vessel (in blue) can be breached. With the EPR core-catcher concept, the molten core material, called corium, falling down from the breached vessel is accumulated and mixed with sacrificial concrete. When corium melts through the melt plug, it is spread and cooled on a dedicated area. Image from Fischer [27]. 5

1.2 Schematic of the EPR core catcher spreading area. In the event of a nuclear severe accident involving core melt-down and vessel failure, with the EPR core-catcher concept, the molten material is spread on a dedicated spreading area to be cooled. Image from Fischer [27]. 7

3.1 Schematic illustration of the spatial discretisation of a 2D rectangular domain into a numerical mesh focusing on two cells P and Q separated by a face f. . . . 26

3.2 Figure from Denner and van Wachem [21]. Mesh cell P with neighboring cell Q and shared face f of an equidistant Cartesian (left) and a triangular (right) two-dimensional mesh including relevant interpolation entities. 30

3.3 Notation used for the discretisation of a general variable ϕ at face f given the flow direction. 32

3.4 Illustration of the notation used to identify cells. Cell P is the cell under consideration and cells W and E are the western and eastern neighbour cells respectively. 39

3.5 Flowchart describing the procedure to solve a non linear system of equation using a linear numerical solver. 41

3.6 Flowchart describing a solution procedure where the VOF equation is solved first with time step Δt_{VOF} and the fully coupled system $[\boldsymbol{u}, p]$ is, then, solved with time step Δt_{Flow}. 43

3.7 Focus on Part I (left) and Part II (right) of the flowchart presented in figure 3.6. Part I describes the procedure to solve the VOF equation and Part II describes the solution procedure to solve the fully coupled system formed with the momentum and continuity equations. 44

3.8 Solution procedure A to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF equation is solved separately but the flow and energy equations are solved in the same time loop. 46

3.9 Solution procedure B to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where flow, VOF and energy equations are solved in three separate time loops. 46

3.10 Solution procedure C to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF and energy equation are solved in the same time loop and the flow equations are solved in a second time loop. . 47

4.1 Stress rate as a function of strain rate for a Newtonian fluid and a Bingham fluid. 50

4.2 Stress as a function of strain rate for a Bingham fluid and different regularised Bingham models. The regularised models (Bercovier-Engelman model, Papanoustiou model and bi-viscosity models) make use of model parameters (ϵ_{BE}, ϵ_{Papa} and C_{BV} respectively) that control the strength of the regularisation. 52

4.3 Schematic description of the lid driven cavity test case. The top wall is moving with velocity 1 m/s and driving the fluid movement inside the cavity 54

4.4 Velocity streamlines for the simulation of a Newtonian fluid in a lid driven cavity 55

4.5 Evolution of the intensity of the principal vortex with time for the simulation of a Newtonian fluid in a lid driven cavity 55

4.6 Streamlines of the velocity for a Bingham fluid in a lid driven cavity using different yield stress values (the case with $\tau_S = 0$ corresponds to a Newtonian fluid). 57

4.7 Evolution of the intensity of the principal vortex with time, for the lid driven cavity test. Simulations are run for a Newtonian fluid and for a Bingham fluid, with yield stress $\tau_S = 10$ Pa, using the BE model with different value of the model constant ϵ_{BE}. 58

4.8 Evolution of the intensity of the principal vortex with time, lid driven cavity test. Simulations are run for a Bingham fluid, with yield stress $\tau_S = 10$ Pa, using the BE, Papanoustiou and bi-viscosity regularised Bingham models. 59

5.1 Solution procedure A to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where the flow and energy equations are solved in the same coupled linear system of equations, and the VOF advection equation is solved separately. 63

5.2 Solution procedure B to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where flow, VOF and energy equations are solved in three separate time loops. . 63

5.3 Solution procedure C to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where the VOF and energy equations are solved in the same time loop and the flow equations are solved in a separate time loop. 64

5.4 Cell viscosity for a cell containing two fluids with viscosities $1Pa.s$ and $100Pa.s$ using different cell viscosity model 65

5.5 Illustration of the surface (2D) or volume (3D) occupied by interface cells (hatched area) on a coarse mesh (left) and a fine mesh (right) 66

5.6 Schematic illustration of the advection of a hot fluid in a cold channel, including the initial values (denoted with subscript "init") and the applied boundary conditions (BC). Initially a hot fluid is present on the left hand side of the domain and a cold fluid occupies the right hand side of the domain. Hot fluid is being introduced in the system from the left boundary of the system with velocity 1 $m.s^{-1}$. Gravity is neglected. 67

5.7 Viscosity of the hot fluid, simulated in the test case "advection of a hot fluid in a cold channel", as a function of temperature 68

5.8 Colour function profile along the x-axis corresponding to the test case "advection of a hot fluid in a cold channel at two different times: initial colour function profile (dashed line) and colour function profile after one second of simulation (solid line). 69

5.9 Profiles of the cell temperature (left) and cell viscosity (right) along the x-axis after one second of simulation, using different schemes for the discretisation of the energy equation (upwind, van Leer and SUPERBEE). 69

5.10 Profiles of the colour function (left) and the temperature (right) along the x-axis after one second of simulation using the CICSAM scheme for the discretisation of the energy equation. 72

5.11 Profile of the viscosity along the x-axis after one second of simulation using the CICSAM scheme for the discretisation of the energy equation. 73

5.12 Analytical value for the viscosity in the interfacial cell as a function of the cell colour in absence of numerical diffusion. 74

5.13 Schematic of the evolution of the colour function value in the interfacial cell with time. 75

5.14 Time evolution of the colour function value in the interfacial cell. 75

5.15 Time evolution of the maximum viscosity using different advection schemes for the discretisation of the energy equation. 76

5.16 Time evolution of the maximum viscosity value using different viscosity averaging methods. 77

5.17 Profile of the colour function along the x-axis in the interface region after one second of simulation obtained using different strategies. 79

5.18 Profiles of viscosity (left) and temperature (right) along the x-axis in the interface region after one second of simulation obtained using different smoothing strategies for the viscosity. 79

5.19 Flowchart describing the proposed algorithm to calculate the temperature-dependent viscosity. 80

5.20 Profile of viscosity along the x-axis in the interfacial region after one second of simulation obtained using two methods to calculate the temperature-dependent viscosity of the hot fluid: using the temperature in the interfacial cell (plain line) and calculating the temperature using the algorithm proposed in figure 5.19 (dashed line). 81

5.21 Functions describing the temperature-dependent viscosity of the hot fluid in the spreading test case. 82

5.22 Schematic illustration of the spreading of a hot dense fluid in a rectangular domain filled with a cold light fluid. No fluid is injected in the system and the hot dense fluid spreads along the bottom wall under the influence of gravity. . . 83

5.23 Predicted spreading length as a function of time for the spreading test case using different advection schemes for the discretisation of the energy equation, compared against the reference simulation with constant viscosity. Two different functions are used for the temperature-dependency of the hot fluid: the linear function given in equation 5.8 (left) and the quadratic function given in equation 5.9 (right). 84

5.24 Maximum viscosity as a function of time for the spreading test case using different advection schemes for the discretisation of the energy equation, compared against the reference simulation with constant viscosity. Two different functions are used for the temperature-dependency of the hot fluid: the linear function given in equation 5.8 (left) and the quadratic function given in equation 5.9 (right) . . . 84

5.25 Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case using different viscosity averaging methods. The function describing the temperature-dependency of the viscosity of the hot fluid is the quadratic function given in equation 5.9. 86

5.26 Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case using different strategies to calculate the temperature-dependent viscosity of the hot fluid and using an arithmetic averaging to calculate the cell viscosity from the viscosities of the individual fluids. . 87

5.27 Predicted spreading length as a function of time for simulations using the proposed numerical methods with different mesh resolutions. 88

5.28 Predicted spreading length as a function of time for simulations using the proposed the algorithm proposed in section 5.3.4 with different mesh resolutions. . . 88

5.29 Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case with heat diffusion, using the upwind and CICSAM schemes for the discretisation of the energy equation and using the arithmetic ("Arith") and harmonic ("Harmo") averaging for the viscosity. 90

5.30 Contours of the colour function (left) and the temperature (right) after one second of simulation of the spreading test case with heat diffusion. The CICSAM scheme is used for the discretisation of the energy equation. 90

5.31 Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case with heat diffusion using different strategies to calculate the temperature-dependent viscosity of the hot fluid. 91

6.1 Geometry of the spreading area of the VULCANO VE-U7 corium spreading experiment. During the experiment, corium is heated-up, poured into the retention cup and it spreads toward the two spreading channels which are made of zirconia bricks and covered with concrete and ceramic respectively. 97

6.2 Solution procedure to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF and energy equation are solved in the same time loop and the flow equations are solved in a separate time loop. 102

6.3 Geometry used to run numerical simulations of the VULCANO VE-U7 corium spreading experiment. Compared to the experiment, the retention cup, where corium is poured, is not represented in the numerical model and only one of the two spreading channels is considered. 104

6.4 3D view of a mesh used to run numerical simulations of the VULCANO VE-U7 experiment. 104

6.5 Top view of the 180 000 cells mesh used to run numerical simulations of the VULCANO VE-U7 experiment. 104

6.6 Schematic illustration of a two dimensional Couette flow including the applied boundary conditions (BC). Initially the fluid is at rest. A constant velocity U is then apply at the top boundary which drives the flow motion. 106

6.7 Analytical and simulation results for a two-dimensional Couette flow problem. The chart presents the adimensional velocity u/U as a function of the adimensioanl height y/h where U is the velocity applied at the top boundary and h is the spacing between the plates. 107

6.8 Spreading length as a function of time for different mesh refinement levels. . . . 108

6.9 Spreading length as a function of time for different mesh refinement levels using the CICSAM scheme and the Upwind scheme for the discretisation of the energy equation. 109

6.10 Spreading distance as a function of time for simulations using different values for the initial viscosity of corium. 109

6.11 Spreading distance as a function of time for simulations using different values for the specific heat capacity values of corium. 111

6.12 Spreading distance as a function of time for simulations using different corium density (left) and different corium thermal diffusion coefficient (right). 111

6.13 Mass of corium in a spreading channel as a function of time. 112

6.14 Corium temperature at the end of the simulation on a slice of the mesh normal to the y-axis. 113

6.15 Height of the corium layer as a function of the x-distance. 114

6.16 Spreading distance as a function of time predicted by the numerical simulation and experimental results. 114

6.17 Illustration of the energy balance in the system considered for radiation heat loss estimation. 115

6.18 Illustration of the VULCANO VE-U7 experimental setup. The corium volume is simplified assuming a constant height from the inlet up to the corium front. The variables presented in blue are used to compute the corium/air interface area. 117

6.19 Analysis of the decrease in corium bulk temperature as a function of time using a zero-dimensional model. 118

Chapter 1

Introduction

This thesis deals with the numerical simulation of corium spreading flows using Computational Fluid Dynamics (CFD). The basics of CFD are introduced in section 1.1, then, corium is defined and its formation during a nuclear accident is described in section 1.2. A strategy to mitigate the consequences of a nuclear accident by spreading corium is presented in section 1.3. The motivation and objective of this thesis are presented in section 1.4 and an outline of the thesis is shown in section 1.5. Finally, section 1.6 discusses the contributions of this thesis to the field of corium modelling.

1.1 Introduction to Computational Fluid Dynamics

Fluid dynamics is a discipline that describes the motion of fluids, such as gases or liquids. The study of fluid flows, encountered in nature and in industrial systems, has many applications, such as weather forecasting, or the design and improvement of products and industrial processes. The motion of fluids is governed by conservation laws of physics:

- Mass conservation : The mass of a fluid in a closed system remains constant
- Newton's second law : The rate of change of momentum equals the sum of the forces acting on a fluid

- First law of thermodynamics: The rate of change of internal energy in a closed system equals the difference between the heat addition to a fluid and the rate of work done on the fluid.

The conversation laws can be represented mathematically by conservation equations and solved to predict the evolution of a fluid system. The equations governing the movement of fluids can only be solved analytically for a limited number of simple flow configurations. To study more complex flows, different methods have been developed, such as experimental and numerical methods.

Computational Fluid Dynamics (CFD) uses computer programs to solve the equations governing fluid flows numerically. Compared to experimental methods, which usually require the set-up of an experimental facility, CFD reduces the time and the cost of the study of fluid flows. Numerical simulations of large scale systems can be performed and potentially hazardous or accidental situation can be investigated. CFD also has the advantage to yield very detailed results while experimental methods are limited by the availability of measurement systems. The main disadvantage of numerical methods is that they are restricted by the mathematical model used to represent the fluid system. If the mathematical model does not accurately describe the physics of the flow investigated, simulation results are not representative of the actual flow.

This thesis uses CFD to investigate the flow of molten material, called corium, encountered in the event of a nuclear accident in a reactor core. The study of accidental situation in existing nuclear reactors is, for obvious reasons, impractical and the design of experiments with molten nuclear material is complex and costly. CFD offers a convenient alternative, however, the simulation of corium flows is extremely complex and standard numerical methods are not always applicable.

1.2 Introduction to corium spreading

In the event of an accident in a nuclear power plant, the balance between heating and cooling of the nuclear reactor core may be lost, leading to an increase of the temperature in the reactor. If the cooling is not restored soon enough, the core starts to degrade and the accident is classified as severe. Severe accidents in light water reactors have been an active research topic for several decades, and, the interested reader is referred to Sehgal [57] for a detailed description of the phenomenology of severe accidents. The focus of the present work is on the late stage of a nuclear accident, after the core degradation has lead to the melting of the reactor internals (e.g fuel, fuel cladding and steel structure) and the failure of the reactor pressure vessel (illustrated in blue in figure 1.1). The composition of the mixture of molten material that forms, called corium, depends on the accident scenario and changes as the accident progresses. Corium is essentially composed of uranium oxide, zirconium oxide, zirconium, iron, fission products and it may include concrete in the later stages of the accident.

After the reactor pressure vessel is breached, corium falls on the concrete base-mat of the reactor containment building. The interaction between corium and concrete threatens the integrity of the nuclear plant by melting through the concrete base-mat or through the release of combustible gases, such as hydrogen, which pressurize the containment building. If the corium is cooled sustainably to a temperature below that of concrete ablation, the severe accident may be terminated before the containment building is breached, preventing or limiting radioactive releases to populations and the environment.

The core-catcher concept ([13], [27]), described in section 1.3, has been developed to limit the consequences of a nuclear severe accident, involving in-vessel core degradation and failure of the reactor pressure vessel, by spreading corium on a dedicated spreading area where it can be cooled efficiently. Spreading corium on a large area also helps reduce the criticallity and, thus, prevent nuclear fission reactions which would increase the severity of the accident.

1.3 The core-catcher concept

The safety objectives set for the design and construction of new nuclear reactors imply the requirement for mitigation of low pressure core melt accident situations. The detailed requirements for the design and construction of new pressurised water reactors are described in the document "Technical guidelines for the design and construction of the EPR" available on the website of the French nuclear regulation authority [9]. In particular, one of the requirements stated in this document is:
"*The penetration of the base-mat of the containment building by a corium must be avoided, as this phenomenon could imply significant releases and durable contamination of underground waters and of the sub-soil. Moreover, adequate provisions have to be implemented to prevent leakage of contaminated water and gases to the sub-soil via cracks in the base-mat.*"

The technical solution retained in the design of the EPR reactor to fulfil this requirement is the implementation of a so-called core-catcher ([13], [27]) to spread the corium on a large area and cool it through active and passive systems. The EPR core-catcher is depicted in figure 1.1.

The EPR core-catcher concept, as described by Bouteille et al. [13] and Fischer [27], distinguishes between two phases:

- in the retention phase, corium is accumulated in a dedicated cavity until the melting of the underlying plug,
- in the spreading phase, corium falls down into a spreading compartment and it is spread over a large area, depicted in figure 1.2, where it is cooled efficiently.

The cavity, below the reactor pressure vessel, is covered with sacrificial concrete, so that, during the retention phase, corium incorporates concrete, which enhances oxidation of metallic uranium and zirconium and lowers the liquidus temperature of the corium [42]. The composition of corium after mixing with sacrificial concrete is expected to promote spreading. The retention phase is also helpful for severe accident studies as it ensures a relatively "standardised" composition of corium which reduces the accident scenario dependency of the corium composition.

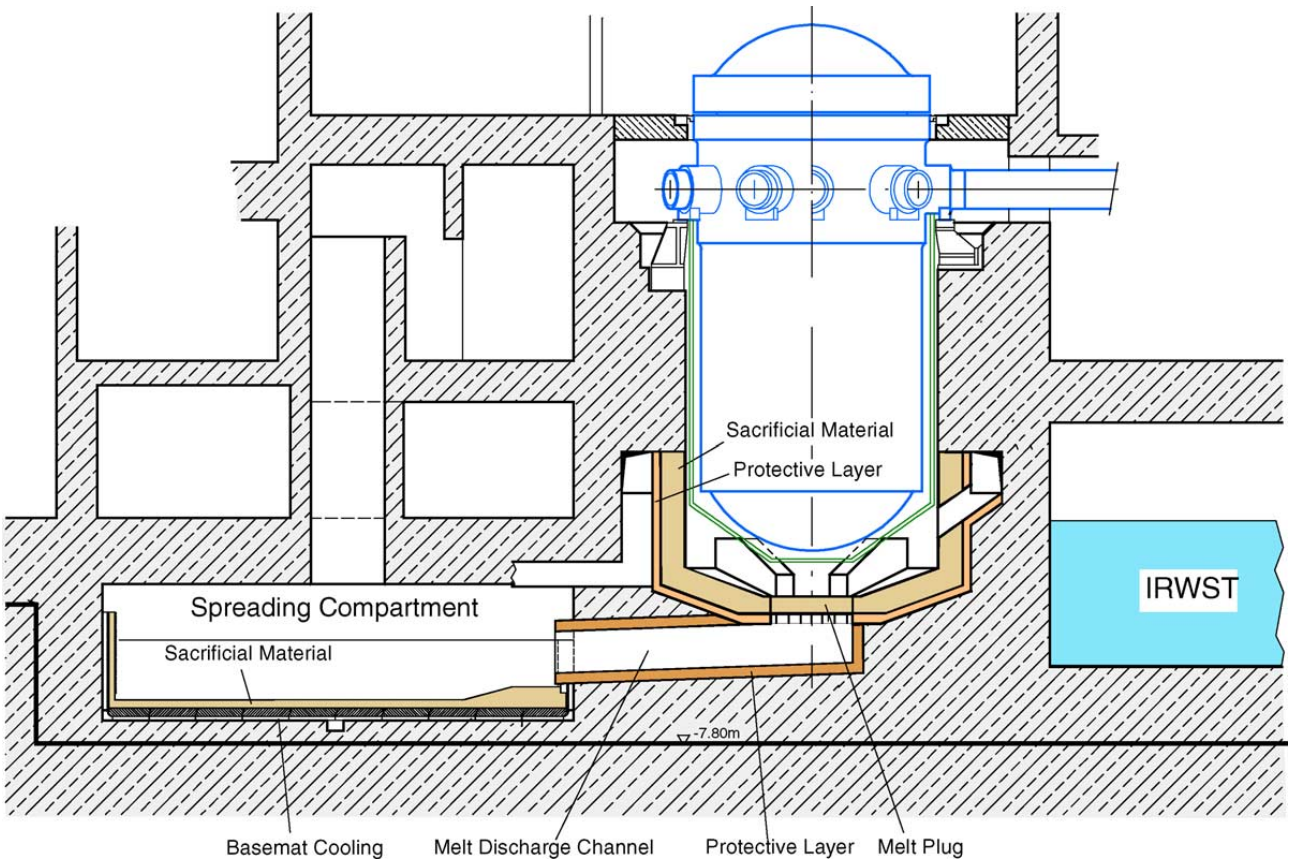


Figure 1.1: Schematic of the EPR core catcher. In the event of a nuclear severe accident, the reactor pressure vessel (in blue) can be breached. With the EPR core-catcher concept, the molten core material, called corium, falling down from the breached vessel is accumulated and mixed with sacrificial concrete. When corium melts through the melt plug, it is spread and cooled on a dedicated area. Image from Fischer [27].

When the melt plug (see figure 1.1) is totally molten, the corium accumulated in the cavity starts spreading on the dedicated spreading area.

The core catcher is equipped with a water cooling system underneath the spreading area. The water comes from the Internal Refuelling Water Storage Tank (IRWST), illustrated in figure 1.1, which collects water condensing in the containment building. The water flows underneath the spreading area and a chimney is there to redirect the generated steam to the containment building. This system allows for passive cooling of corium, however, active water injection systems provide extra cooling.

With the EPR core-catcher, corium spreading is expected to lead to a quasi-homogeneous corium thickness over the spreading compartment. The cooling of corium spread on a large area could lead to a stable state within hours, and, complete corium solidification could be

achieved after a few days [27]. Another purpose of spreading corium on a large area is to limit the criticality of corium. Indeed, corium is made of fissile materials and if the radioactive spread reaches a critical configuration, energetic fission reactions could worsen the nuclear accident.

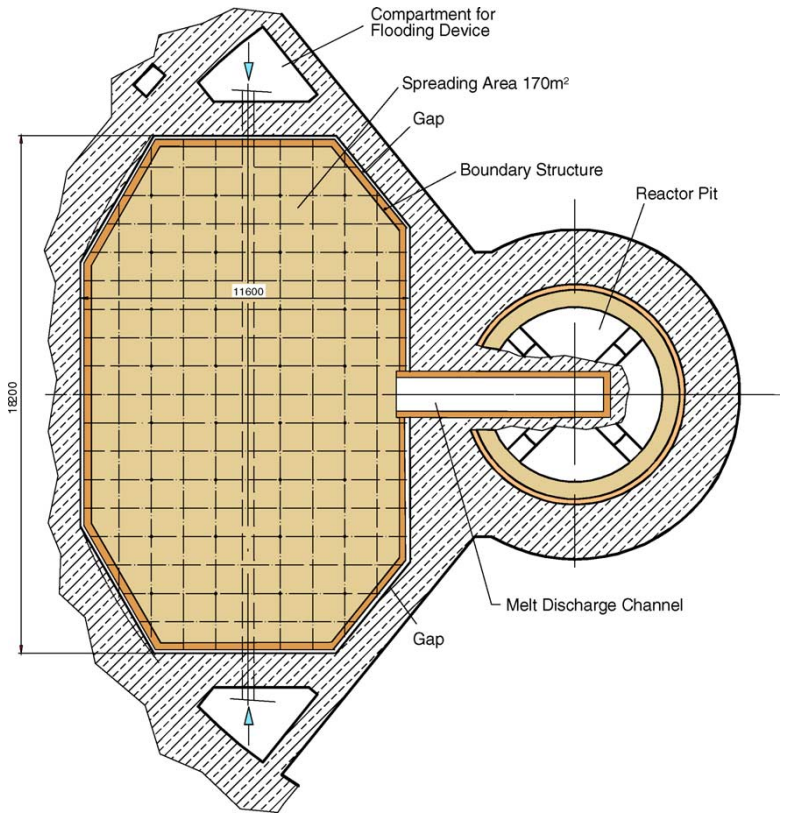


Figure 1.2: Schematic of the EPR core catcher spreading area. In the event of a nuclear severe accident involving core melt-down and vessel failure, with the EPR core-catcher concept, the molten material is spread on a dedicated spreading area to be cooled. Image from Fischer [27].

1.4 Motivation and objectives

A good understanding of corium spreading helps with the design of efficient core-catchers and thus improves the safety of reactors. However, corium spreading flows are very complex to study due to large uncertainties on corium properties and multiple phenomena at different scales occurring at the same time [57]. The composition of corium depends on the type of reactor, the accident scenario and it changes as the accident progresses. Moreover, corium is a non-eutectic mixture solidifying over a large range of temperature (of the order of 1000 K). The modelling of corium rheology has been an active research topic ([34], [50], [54]), but, it remains complex and largely uncertain. The spreading is also affected by the formation of a corium crust, changes in the thermochemical state of corium (e.g. corium oxidation) and interactions between corium and the underlying substrate. During corium spreading on concrete substrates, concrete melting, concrete ablation and gas release from the concrete decomposition may occur and affect the spreading.

One of the many challenges related to the simulation of corium spreading flows with CFD is the representation of the corium/air interface (also referred to as corium free surface) in a numerical framework. The main objective of this thesis is the detailed investigation of the numerical methods best suited for the simulation of corium spreading flows. In particular, the Volume-Of-Fluid method (VOF), commonly used to simulate interfacial flows, is investigated and the effect of numerical errors, associated with the numerical solution of the equations governing the flow, on spreading predictions is evaluated.

The work presented in this thesis makes use of the progress made in the field of multiphase flow modelling in the past decades, as well as, novel numerical methods to simulate corium spreading flows. The objective of this thesis is not to perform corium spreading simulations that can be used for the assessment of nuclear safety in nuclear reactors but, rather, to develop, test and propose numerical methods suitable for such simulations.

1.5 Thesis outline

In the introduction of this thesis, the laws of physics describing fluid flows are presented, the formation of corium during a nuclear severe accident is discussed and the concept of core-catcher is introduced.

Chapter 2 reviews previous studies investigating corium spreading. The experimental and numerical studies of corium spreading available in the scientific literature are presented and the key findings are discussed.

The numerical framework used as a basis for the numerical simulation of corium flows is presented in chapter 3. The equations that govern the fluid dynamics are introduced and the numerical methods used, in this thesis, to solve the equations are described. Chapter 3 also presents a new solution procedure to solve the governing equations, developed, as part of this thesis, to limit numerical errors associated with the simulation of interfacial flows.

Chapter 4 presents the Bingham fluid model that can be used to model flows of solidifying liquids. The constitutive equations of a Bingham fluid are introduced, the implementation of the Bingham fluid model in a CFD code is described and the implemented model is tested on a benchmark problem. The benchmark problem is a Bingham fluid in a lid-driven square cavity, for which benchmark results are available in the scientific literature.

Chapter 5 is the main chapter of this thesis. It focuses on the numerical methods to simulate a two-fluid flow where one of the fluids has temperature dependent properties (such as the temperature-dependent viscosity of corium). The numerical errors that affect the simulation of such flows are investigated in details and methods to minimize such errors are proposed. It is shown in chapter 5 that numerical errors, such as numerical diffusion of temperature, can lead to significant errors in the prediction of the spreading of a fluid. The numerical methods are investigated using two test cases. The first test case simulates a basic channel flow where a hot fluid is injected in a channel that contains a cold fluid. The second test case simulates the spreading of a hot dense fluid with a temperature-dependent viscosity (representing corium) in a domain filled with a cold light fluid (representing air).

Chapter 6 builds up on the different developments and studies presented in the previous chapters to simulate the VULCANO VE-U7 corium spreading experiment [35]. The effect of numerical methods and fluid properties of corium on the predicted spreading length are investigated and the simulation results are compared to the experimental results.

1.6 Contributions

The research presented in this thesis focuses on numerical methods for the simulation of corium spreading flows. As part of this thesis, contributions to the numerical framework developed by Denner and van Wachem [21], such as the implementation of models and numerical methods, have been performed. This thesis discusses in detail the numerical errors arising in the vicinity of the interface, for simulations of two-phase flows of immiscible fluids (such as corium and air) with temperature-dependent fluid properties. In particular, the numerical errors that lead to an erroneous evaluation of the temperature-dependent viscosity of corium are investigated. In this thesis, simulations of an experiment, involving a prototypical corium composition, are performed and the results are compared to experimental results as well as results from other numerical simulations available in the scientific literature. The major contributions of this thesis to the field of research in corium modelling are:

- The detailed description of the implementation of a regularised Bingham fluid model, in a computational fluid dynamics framework based on the finite volume method, to simulate the non-Newtonian behaviour of corium below liquidus temperature. The study of Bingham fluids using numerical methods has been going on for many decades ([11], [43], [45]), however, the implementation of the regularised Bingham fluid model in a finite volume framework is rarely described in great details. In the work presented in this thesis, all the steps involved in the implementation of the Bingham fluid model are explained, and, thus, the results presented in this thesis could be easily replicated and improved in future research work.

- A new numerical method to minimise the adverse effect of numerical diffusion of tem-

perature on the simulation of two-phase flows of immiscible liquids with temperature-dependent properties. Numerical diffusion of temperature, in the vicinity of the corium/air interface, causes an artificial cooling of the corium phase. Due to the temperature-dependency of the viscosity of corium, the lower temperatures cause an artificial increase in the corium viscosity, therefore, the predicted corium spreading length is reduced. In this thesis, the use of the CICSAM scheme [68], developed for the discretisation of sharp interfaces, for the discretisation of the energy equation is suggested. The use of the CICSAM scheme to discretise the energy equation requires modifications of the procedure used to solve the equations governing the fluid flow. The research presented in this thesis investigates different solution procedures and a procedure best suited for the simulation of corium spreading flows, described in chapter (5), is proposed. The proposed method is evaluated on different test cases, and, compared to standard methods commonly used in the CFD literature, numerical diffusion of temperature is significantly reduced.

- A new algorithm to calculate, in a Volume-of-Fluid (VOF) framework, the temperature-dependent properties of a fluid in interfacial cells. When VOF methods are used to capture the dynamics of the interface between immiscible fluids, mesh cells containing two fluids, referred to as interfacial cells, are encountered. Simulations using the Volume-of-Fluid method solve a single set of governing equations in every mesh cells for the phase-averaged variables (velocities, temperature). Therefore, even if interfacial cells contain two fluids with potentially different temperatures, a single cell-averaged temperature value is available in each grid cell as an output of the simulation. The averaged temperature value might be very different from the individual temperatures of the two fluids present in interfacial cells, therefore, the evaluation of a temperature-dependent fluid property based on the available cell-averaged temperature is prone to errors. With the new algorithm proposed in this thesis, if a fluid "A" has strongly temperature-dependent fluid properties, to evaluate the temperature-dependent property of fluid "A" in an interfacial cell, a loop on neighbouring mesh cells is performed, and, the temperature of the neighbour cell containing the highest ratio of fluid "A" is used for the temperature-dependent property calculation. The proposed algorithm is evaluated on different test cases with known

solutions, and, the calculated temperature-dependent fluid properties at interfacial cells is significantly improved.

- Results of numerical simulations of the VULCANO VE-U7 corium spreading experiment [35]. In the VULCANO VE-U7 corium spreading experiment, a corium mixture is heated to high temperatures and spread on two spreading channels. The corium spreading length is measured and the results are used as a benchmark to compare the performances of several simulation codes. In this thesis, new simulation results of the VULCANO VE-U7 experiment are obtained and compared to experimental results found in the scientific literature.

Chapter 2

Review of Corium spreading studies

Ex-vessel corium stabilization has been an active research topic in nuclear safety over the past decades. In particular, two large European projects, involving collaborations between several research laboratories, were dedicated to the study of ex-vessel corium: the Ex-vessel Core Melt STabilization Research (ECOSTAR) project [5] and the corium Spreading and Coolability (CSC) project [15]. The experiments presented in this chapter, see section 2.1, were conducted in the framework of these projects and the computation codes presented in section 2.2 were developed and validated using the data obtained through these experiments. This chapter reviews some of the experiments, developments and findings associated with these projects.

2.1 Experimental studies of prototypical corium

Corium spreading experiments using prototypical corium (corium with a realistic composition including depleted uranium oxide) are complex and costly, therefore, the first spreading experiment were conducted with non-prototypical corium, later on referred to as simulant material (CORINE [72], S3E [23], SPREAD [63], KATS [26], ECOKATS [3] [4]). Some of the key findings of experiment using simulant materials are reported in [15], such as the significant difference between 1D spreading and 2D spreading or the role of crust formation in spread immobilization. To validate the findings of experiments using simulant material, Piluso et al.

[49] highlight the necessity to conduct experiments using prototypical corium compositions. The COMAS [62], FARO [65] and VULCANO [33] series of experiments, described in this section, use prototypical corium compositions to study corium spreading flows. In this thesis, the VULCANO VE-U7 [35] experiment, which is part of the VULCANO series of experiment is simulated using CFD and the results are reported in chapter 6.

2.1.1 COMAS series of experiments

The objective of the COMAS experiments [62] is to perform spreading tests with prototypical corium representative of EPR corium melts to provide a basis for the development of core-catchers. The tests are carried out at Siempelkamps CARLA melting facility for radioactively contaminated metal scrap. The COMAS experiments mainly focus on high melt release rates (above 150 kg/s).

The experiments investigate different substrate material for the spreading section: concrete, ceramics and cast iron. In the spreading results reported, the dependency of the spreading on the substrate material is not significant, however, the melt release rate and initial corium temperature have a significant effect on the spreading distance.

In the EPR core catcher concept, corium is mixed with sacrificial concrete and as a result, corium properties are modified. In the COMAS experiment the influence of concrete incorporation in corium composition on spreading distance is investigated by adding silicate to the reference corium composition. The presence of silicate lowers the solidus temperature of the mixture while the liquidus temperature is hardly affected. The resulting mixture has a solidification range of about 1000 K, so, corium can be cooled down to lower temperatures before it completely solidifies. In the reported tests results, the spreading distance is increased by adding silicate to the corium mixture. This finding highlights the role of solidification and crust formation in the melt immobilisation.

2.1.2 FARO series of experiments

The FARO series of experiments [65] uses corium composed of binary mixtures of uranium oxide and zirconium oxide, representative of corium before it is mixed with sacrificial concrete. The solidification range for such mixtures is only 50 K, compared to a range around 1000 K after mixing with concrete which lowers the solidus temperature. The FARO experiments investigate corium spreading on 2D sector geometries on a metallic substrate and at relatively low melt release rates (about 16.5 kg/s) compared to the COMAS experiments.

The spreading observed, in the FARO experiments, is characterised by an irregular flow with local stops of the spread. The stops are attributed to the formation of a crust and flow continuation is sometimes observed after the break of the crust. Corium spreading on a wet substrate is investigated in the FARO experiments and the reported results are not significantly different from the results of spreading on a dry surface. The spread thickness observed with the FARO experiments (about 60 mm) is comparable to the thickness observed with the COMAS experiments (about 70 mm), however, non-symmetric spreading is reported for the FARO experiments. The spread stopped due to crust formation and corium remelting occurred over only half of the width of the spreading channel, leading to a non-symmetric spreading profile.

The initial corium temperature reported for the experiments is close to the liquidus temperature of corium, but, the uncertainty range ($\pm$50 K), is in the order of magnitude of the solidification range. This makes the modelling of the experiment challenging and the FARO experiments are, thus, not a good candidate to validate spreading predictions using numerical methods.

2.1.3 VULCANO series of experiment

The VULCANO series of experiments [33] are part of a research effort dedicated to the study of corium spreading and coolability [15], motivated by the design and assessment of a reliable core-catcher. The VULCANO experiments investigate a large range of possible prototypical corium compositions spreading at low mass flow rates.

The results of the VULCANO experiments proved that corium-concrete mixtures, such as the mixture expected with the EPR core-catcher concept, have good spreading characteristics, which can lead to homogeneous corium thickness over the spreading area, and, thus, promote efficient corium cooling. Journeau et al. [33] report that the ablation of concrete substrates occurs on a time scale longer than the spreading time scale. Therefore, concrete ablation does not significantly affect the spreading and these results suggest that concrete ablation may be neglected for corium spreading studies.

The experiments performed using prototypical corium are used for the validation of spreading models, such as models to describe corium viscosity, and to validate codes for corium spreading simulations. The validation of simulation codes is useful as validated codes may be used to investigate corium flows in realistic reactor conditions.

Journeau et al. [33] report that the validation of models and codes is limited by the large uncertainty on the fluid properties of corium, and, they suggest that a major research effort to improve the knowledge of corium properties is required to perform more precise validation of corium models.

The VULCANO VE-U7 experiment, which investigates the spreading of corium on concrete and ceramic substrates, is simulated as part of the work presented in this thesis. The VULCANO VE-U7 experiment is presented in chapter 6 and the experimental results are compared with the simulations performed in the present work.

2.2 Numerical modelling of corium spreading

Compared to experimental studies, numerical simulations of corium spreading offer several advantages such as a low cost and the possibility to simulate large-scale systems. The availability of variables throughout the simulated system is also an advantage compared to experiments, where the number and type of sensors that can be used is limited. However, a major drawback of numerical simulations is the representativeness of the results. Corium spreading is complex and it is affected by many different physical phenomena. Unlike experiments, which naturally

includes all the phenomena affecting the flow, simulations are restricted to the mathematical model used to represent the simulated system. Simulations are also affected by simplifications and errors in the solution of the system of equations that describes the flow evolution. Experiments may be used to validate numerical simulations and numerical methods may be used to expand the range of flows that can be studied experimentally. Experimental and numerical methods are, thus, complementary to investigate corium spreading flows.

Among the research performed towards the numerical simulation of corium spreading flows, two types of codes can be distinguished: codes developed specifically for corium simulations, such as the codes MELTSPREAD [24], THEMA [61], LAVA [2] [58], and codes suitable to simulate a large range of three-dimensional flow conditions, such as the codes CORFLOW [77], CROCO [39] [48] [74] or commercially available CFD codes.

The codes MELTSPREAD, THEMA and LAVA take advantage of a characteristic of corium spreading flows, the vertical length scale is small compared to the horizontal length scale, to use the shallow water assumption and simplify the modelling. With the shallow water assumptions, a fixed vertical velocity profile is assumed and the governing equations are integrated along the vertical axis. The shallow water equations for a fluid with constant density [32] are given in non-conservative form by

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left(\overline{h} + h\right) u_x + \frac{\partial}{\partial y}\left(\overline{h} + h\right) u_y = 0, \tag{2.1}$$

$$\frac{\partial u_x}{\partial t} + u_x \frac{\partial u_x}{\partial x} + u_y \frac{\partial u_x}{\partial y} = -g\frac{\partial h}{\partial x} + \frac{\mu}{\rho}\left(\frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2}\right), \tag{2.2}$$

$$\frac{\partial u_y}{\partial t} + u_x \frac{\partial u_y}{\partial x} + u_y \frac{\partial u_y}{\partial y} = -g\frac{\partial h}{\partial y} + \frac{\mu}{\rho}\left(\frac{\partial^2 u_y}{\partial x^2} + \frac{\partial^2 u_y}{\partial y^2}\right), \tag{2.3}$$

where t is the time, x and y are the horizontal Cartesian coordinates, $\overline{h}$ is mean height of the fluid layer (vertical direction), h is the local deviation from the mean layer height, $\boldsymbol{u}$ is the fluid velocity, g is the acceleration due to gravity, μ is the fluid dynamic viscosity and ρ is the fluid

density.

Codes using integrated equations have the advantage of numerical simplicity, robustness and low computational cost, and, given the large uncertainties on the physical properties of corium, detailed modelling may not necessarily yield more accurate results [59].

This section describes different codes that have been used, in the literature, to investigate corium spreading and some of the major findings.

2.2.1 MELTSPREAD

The MELTSPREAD code [24] is a one-dimensional code using the shallow water assumption to predict gravity driven spreading flows and thermal interactions between corium and the underlying, concrete or steel, substrate.

The code includes models to simulate corium freezing, immobilization and remelting; heat transfer in the substrate; decomposition, gas release, melting and ablation of concrete substrates; and; corium oxidation by the gas produced by corium decomposition.

The fluid velocity is calculated by solving a one-dimensional momentum equation including a transient term, a convection term, a hydrostatic pressure term and a friction term. Corium immobilization is enforced when the corium height at the leading edge falls below the minimum height at which surface tension balances gravity. A variety of boundary conditions can be prescribed, at the corium/substrate interface, to account for the presence of a corium crust and the ablation of the substrate. At the corium top surface, the heat transfer is calculated either assuming the presence of an overlying water layer or assuming a dry atmosphere. Different heat transfer models can be prescribed, at the top surface, to account the presence of a top crust.

In the work of Farmer et al. [24], the MELTSPREAD code is validated against the analytical solution of a "dam break" problem, and, against an experiment using water as a corium simulant in a scaled model of a Mk I boiling-water nuclear reactor containment building. Simulations of corium spreading on a concrete substrate have been performed [24] and the simulation results

suggest that the time scale of spreading is shorter than the time scales of concrete ablation and corium oxidation by the gas from concrete decomposition. Farmer et al. [24], therefore, suggest that the effects of concrete ablation and corium oxidation could be neglected in corium spreading studies.

2.2.2 THEMA

The THEMA code [61] is devoted to the simulation of corium spreading flows, in one-dimensional or two-dimensional geometries, taking into account corium solidification and possible melting of the base-mat. The possible stratification of corium into an oxidic and a metallic phase with an overlying crust layer can be simulated. The THEMA code includes a module to solve the heat transfer in the substrate. The heat transfer equation is solved in a three-dimensional numerical mesh of the substrate with models to take into account gas release from concrete decomposition. The THEMA code has been validated against several spreading experiments, such as the experiments CORINE [72], KATS [26], ECOKATS [3] [4], COMAS [62], FARO [65] and VULCANO [33].

The code solves the equations governing the evolution of a two-phase flow, with an oxidic and a metallic fluid phases, integrated over the melt thickness. Similarly to the METSPREAD code, corium immobilization is enforced when the height of corium at the leading edge falls below the minimum height at which surface tension balances gravity. The viscosity of solidifying corium is modelled using an apparent viscosity model, where the viscosity of corium is a function of its solid fraction. Several correlations have been investigated by Spindler and Veteau [61] and after validation against experiments using prototypical corium, the Ramacciotti correlation [54] is shown to yield better agreement with the experiments. A Newtonian and a Bingham model are implemented in the THEMA code, and, simulations using the two models have been compared to experiments. The difference observed between the Bingham and the Newtonian models are not significant. Therefore, given that the Bingham model requires a yield stress parameter which is uncertain, Spindler and Veteau [61] recommends the use of a Newtonian model for corium spreading simulations.

2.2.3 LAVA

The LAVA code [2] [58] is developed for the simulation of corium spreading and relocation in the reactor containment building. The code computes the fluid velocity by solving the momentum equations integrated over the melt thickness and using the creeping flow assumption so that inertia terms are not taken into account. Corium rheology is modelled using a Bingham fluid model and an effective viscosity model, where the solidifying corium viscosity is a function of the solid fraction. The LAVA code solves a one-dimensional heat transfer equation which accounts for the radiation heat transfer at the corium free surface, latent heat associated with crust formation, heat transfer by conduction into the corium phase and heat transfer by conduction into the substrate. The code has been validated against spreading experiment using prototypical corium, such as COMAS [62] and VULCANO [33]. The LAVA code has been used to study spreading in the reactor cavity of a pressurised water reactor. Spengler [58] report that, in most situation considered, with a direct melt release from the breached reactor pressure vessel into the reactor cavity, the spreading will lead to a corium distribution in the cavity which allows for efficient cooling.

2.2.4 CROCO

The CROCO code [39] [48] [74] is developed for the numerical simulation of gravity currents of non-Newtonian fluids on a two-dimensional computational grid. The corium free-surface is described as a polygonal interface and advanced in time by a Lagrangian technique. The Navier-Stokes equations are solved using a Galerkin method. With this method, non-differentiable constitutive laws, such as the constitutive laws of a Bingham fluid, can be simulated without the need for regularisation [73].

The CROCO code has been validated against benchmark problems, such as a lid driven cavity and the differentially heated cavity, for which numerical and experimental results are available in the scientific literature ([11], [12], [17], [40]). The CROCO code has been used to simulate corium spreading flows, however, the simulation results have not been published in a journal.

2.2.5 CORFLOW

The CORFLOW code [77] is developed for the study of free-surface flows and structural material in three-dimensional geometries, using a finite volume framework, and, it is applicable to corium flows. The fluid dynamics are described by the continuity and the momentum conservation equations with the Boussinesq approximation. The free surface is represented by a height function and a free-surface pressure boundary condition is applied at the free surface. The fluid temperature is computed by solving an energy equation taking into account internal heat generation by decay heat, heat transfer by conduction into the underlying substrate, and, radiative and convective heat transfer at the corium free-surface. To model corium rheology, the solidification phase-change model introduced by Voller ([75], [76]) and the Bingham fluid model introduced by Papanastasiou [45] are implemented in the CORFLOW code.

In the work of Wittmaack [78], the CORFLOW code is validated against several spreading experiments and the author identifies corium dynamic viscosity and inlet conditions for spreading as major sources of uncertainty of spreading length predictions. Wittmaack [78] investigated the corium spreading on the dedicated spreading area of the EPR core-catcher using the CORFLOW code. The calculations performed use conservative assumptions (assumptions that prevent the spreading), such as the use of a purely oxidic corium or a conservative choice of corium properties. The results indicate that, even in the most conservative situation, an almost homogeneous corium thickness in the spreading compartment is obtained, which enables efficient cooling of corium.

Chapter 3

Multiphase Computational Fluid Dynamics Framework

The development of numerical methods to accurately predict the displacement of an interface with time has been an active research topic for many decades. The main objective of this thesis is to make use of the recent progress made in the field of multiphase flow modelling to simulate corium spreading flows. The CFD framework developed by Denner and van Wachem [21] is used in this thesis and the main characteristics of this framework are presented in this chapter. Contributions to this framework are made to tackle the challenges of corium spreading flow simulations. In particular, in the present thesis, different solution procedures to solve the governing equations are implemented and investigated (see section 3.3).

Throughout this thesis, corium spreading is modelled as the incompressible flow of two immiscible homogeneous fluids (corium and air) separated by an interface. The simplifications and assumptions used to model corium spreading are discussed in detail in chapter 6. In this chapter, section 3.1 introduces the equations governing the general flow of two immiscible fluids separated by an interface. Section 3.2 describes the discretisation process required to solve the governing equations using numerical methods and section 3.3 describes the procedure followed to solve the governing equations.

3.1 Governing equations

The flow of a fluid can be described mathematically by a set of partial differential equations derived from the conservation of physical quantities: mass, momentum and energy. These equations can appear in different forms [8] but this section presents the equation in a form commonly used in Computational Fluid Dynamics frameworks. Detailed derivation of these governing equations can be found in Versteeg and Malalasekera [71]. Throughout this thesis, all equations are solved in a fixed Cartesian coordinate system with axes x, y and z.

Following the one fluid formulation described in Kataoka [36], which allows the implementation of classical single-phase numerical tools for solving multiphase flow problems, the two fluids in the corium/air system considered, are modelled as a single continuum and a single set of governing equations describes the evolution of phase-weighted variables. The different fluids are identified locally by a volume fraction field. The system of governing equations to be solved consists of a continuity equation, momentum conservation equations and an energy conservation equation. In the present study, the fluids are assumed to be Newtonian and incompressible. Corium density variation due to thermal expansion are supposed to be slow and are thus neglected throughout the present thesis. Similar assumptions are also made in other numerical studies of corium flows such as the study of Wittmaack [77].

The continuity equation for a compressible fluid is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0, \tag{3.1}$$

where subscripts i refers to Cartesian coordinates, ρ is the fluid density and $\boldsymbol{u}$ is the fluid velocity. For an incompressible fluid, the change in density with time and temperature are negligible and the continuity is satisfied if the divergence of the velocity field is zero. The continuity equations simplifies to

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{3.2}$$

For a two-phase flow of incompressible fluids, density may vary in space. Equation 3.1 can be rewritten as

$$\frac{\partial \rho}{\partial t} + \rho \frac{\partial u_i}{\partial x_i} + u_i \frac{\partial \rho}{\partial x_i} = 0. \tag{3.3}$$

Inserting the material derivative of the density, given by

$$\frac{D\rho}{Dt} = \frac{\partial \rho}{\partial t} + u_i \frac{\partial \rho}{\partial x_i} = 0, \tag{3.4}$$

in equation 3.3, the two-phase flow continuity constraint follows as

$$\rho \frac{\partial u_i}{\partial x_i} = 0. \tag{3.5}$$

Therefore, for a two-phase flow of incompressible fluids, continuity is satisfied if the divergence of the velocity field is zero. The momentum equations, in non-conservative form, for such fluids are given by

$$\rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}(\mu S_{ij}) + g_i, \tag{3.6}$$

where subscripts i and j refer to Cartesian coordinates, p is the pressure, μ is the fluid viscosity, $\boldsymbol{S}$ is the rate of strain

$$S_{ij} = \frac{1}{2} \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right), \tag{3.7}$$

and $\boldsymbol{g}$ is the gravity force acting on the fluid of interest.

The first law of thermodynamics, which governs the heat exchange in a system, can be expressed using several formulations. Since the flow considered in this work is incompressible, with constant density and specific heat capacity, the temperature formulation is most convenient. In the present work, the heat exchange due to viscous dissipation is assumed to be small and

is neglected. The energy conservation equation is then given by

$$\rho c_P \left(\frac{\partial T}{\partial t} + u_j \frac{\partial T}{\partial x_j} \right) = -\frac{\partial}{\partial x_j} \left(\lambda \frac{\partial T}{\partial x_j} \right), \tag{3.8}$$

where c_P is the specific heat capacity, T is the fluid temperature and λ is the thermal conductivity.

The set of equations above needs to be complemented with an equation to track the evolution of the interface in the domain. In the present study, the VOF method [31] is used to capture the evolution of the interface between two immiscible fluids A and B. With the VOF method, the interface is represented implicitly through a colour function C defined as the volume fraction of fluid A in a cell

$$\begin{aligned} C = 0 \quad & \text{Cell contains fluid A only} \\ C = 1 \quad & \text{Cell contains fluid B only} \\ 0 < C < 1 \quad & \text{Cell contains an interface.} \end{aligned} \tag{3.9}$$

The colour function is used to capture the interface position and its evolution is described by the VOF equation

$$\frac{\partial C}{\partial t} + u_j \frac{\partial C}{\partial x_j} = 0. \tag{3.10}$$

The density ρ, viscosity μ, specific heat capacity c_P and thermal conductivity λ are defined in each cell as a colour function weighted average of the values of the individual fluids. The most commonly used averaging method to compute the cell viscosity from the viscosities of individual fluids is a colour weighted arithmetic average (simply referred to as arithmetic average), given, for fluid property ϕ as

$$\phi = C\phi_A + (1 - C)\phi_B. \tag{3.11}$$

3.2 Discretisation of the governing equations

The numerical simulation of fluid flows in a finite volume framework requires to compute the alue of the variables that govern the flow (e.g. velocity, pressure, temperature) at discrete points in space and time. The discrete times are obtained by dividing the total length of the simulation into a finite number of time steps. The space discretisation is performed by dividing the total domain to be simulated into a finite number of cells and forming a numerical mesh. Figure 3.1 illustrates the spacial discretisation of a 2D rectangular domain indicating cells and cell faces.

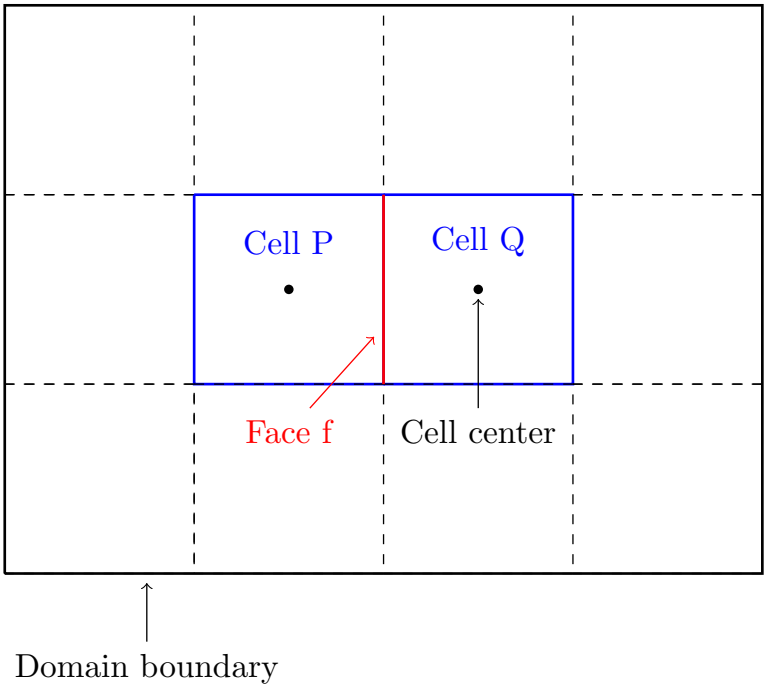


Figure 3.1: Schematic illustration of the spatial discretisation of a 2D rectangular domain into a numerical mesh focusing on two cells P and Q separated by a face f.

The mathematical model presented in section 3.1, which consists of a set of partial differential equations, is the starting point for the numerical simulation of the flows of interest and flow predictions are made by solving these equations. To solve the governing equations, in each cell and at every discrete time, using a numerical solver, the equations need to be transformed in a process called discretisation. This section describes the discretisation process used in the present work.

3.2.1 Time discretisation

In a fluid simulation, the initial value of variables (initial conditions) are known and specified as an input parameter for the simulation. The initial condition are used to compute the values of the variables at the first time step, which, in turn, can be used to compute the next time steps. The values of variables at each time step are thus computed sequentially until the final time t_{end} is reached. The time discretisation process is presented for the transport equation of a general variable ϕ in a simplified form

$$\frac{\partial \phi}{\partial t} = f\left(t, \phi(t)\right), \tag{3.12}$$

where $f\left(t, \phi(t)\right)$ contains the advection and diffusion terms.

An expression of variable ϕ at a new time step $t + \Delta t$ from a known solution at time t can be obtained by integrating equation 3.12 over a time step

$$\int_t^{t+\Delta t} \frac{\partial \phi}{\partial t} dt = \phi^{t+\Delta t} - \phi^t = \int_t^{t+\Delta t} f\left(t, \phi(t)\right) dt, \tag{3.13}$$

where $\phi^{t+\Delta t}$ is a shorthand notation for the value of variable ϕ at time $t + \Delta t$.

The evaluation of the integral in the right hand side of equation 3.13 requires the knowledge of the unknown ϕ, therefore, an approximation is necessary. With the backward Euler method, the integral on the right hand side of equation 3.13 is evaluated using the value of the integrand at the final point

$$\int_t^{t+\Delta t} f\left(t, \phi(t)\right) dt \approx f\left(t + \Delta t, \phi^{t+\Delta t}\right) \Delta t. \tag{3.14}$$

Combining equations 3.13 and 3.14, the value of ϕ at discrete time $t + \Delta t$ is

$$\phi^{t+\Delta t} = \phi^t + f\left(t + \Delta t, \phi^{t+\Delta t}\right) \Delta t. \tag{3.15}$$

The backward Euler method is first order accurate and it is unconditionally stable. In this thesis, the backward Euler scheme is used for the discretisation of the flow equations (momentum and continuity). The Crank Nicolson scheme, presented in section 3.2.3, is applied for the discretisation of the VOF and energy equations.

3.2.2 Space discretisation

With the finite volume framework, the governing equations are integrated over mesh cells. In this section, the spatial discretisation process is described for the transport equation of a general variable ϕ in integral form

$$\frac{\partial}{\partial t}\int_{V_P}\phi dV + \int_{V_P} u_j \frac{\partial \phi}{\partial x_j} dV = \int_{V_P} \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial \phi}{\partial x_j}\right) dV, \tag{3.16}$$

where V_P represents the volume of cell P and ϕ_P represents the discrete value for variable ϕ at the centre of cell P. The volume integral of the terms, in equation 3.16, that contain gradients of dependent variables can be turned into surface integrals using the Gauss theorem

$$\int_{V_P}\left(\frac{\partial \phi}{\partial x_i}\right) dV = \int_S (\phi \boldsymbol{e}_i.\boldsymbol{n})\, dS, \tag{3.17}$$

where S is the outer surface of control volume V_P, $\boldsymbol{e}$ is the unit vector of Cartesian coordinates, dS in an infinitesimal surface element on S and $\boldsymbol{n}$ is an outward pointing vector normal to dS. Each cell is bounded by a finite number of flat faces f so the integration over the surface, in equation 3.17, can be expressed as a sum of integrals over each face

$$\int_S (\phi \boldsymbol{e}_i.\boldsymbol{n})\, dS = \sum_f \int_f (\phi \boldsymbol{n}_f)\, dS. \tag{3.18}$$

The value of ϕ along a boundary face is not known therefore an approximation is required. With the so-called midpoint rule, the integral over a face is approximated by the value at

the face centre multiplied by the face area. On a uniform grid, this leads to the second-order approximation at face centres

$$\sum_f \int_f (\phi \boldsymbol{n}_f)\, d\mathbf{S} \approx \sum_f \phi_f \boldsymbol{n}_f A_f, \tag{3.19}$$

where subscript f denotes values at face centres, $\boldsymbol{n}_f$ is the outward-pointing unit vector normal to face f and A_f is the area of face f.

Continuity equation

The numerical framework used in the present thesis is based on the work of Denner and van Wachem [21]. In this framework, a linear equation system is constructed containing the three discretised momentum equations (assuming a three-dimensional flow), and a modified continuity equation providing a relationship between pressure and velocity. Unlike many CFD frameworks, the numerical framework developed by Denner and van Wachem [21] does not require the solution of a Poisson type equation. The fourth equation is based on the continuity equation

$$\frac{\partial u_i}{\partial x_j} = 0, \tag{3.20}$$

where an advection velocity u_f^n derived using the momentum interpolation method, first proposed by Rhie and Chow [55], is introduced

$$\frac{\partial u_i}{\partial x_j} = \frac{1}{V_P} \int_{V_P} \frac{\partial u_i}{\partial x_j} dV = \frac{1}{V_P} \sum_f \boldsymbol{u}_f \mathbf{n}_f A_f \approx \frac{1}{V_P} \sum_f u_f^n A_f = 0. \tag{3.21}$$

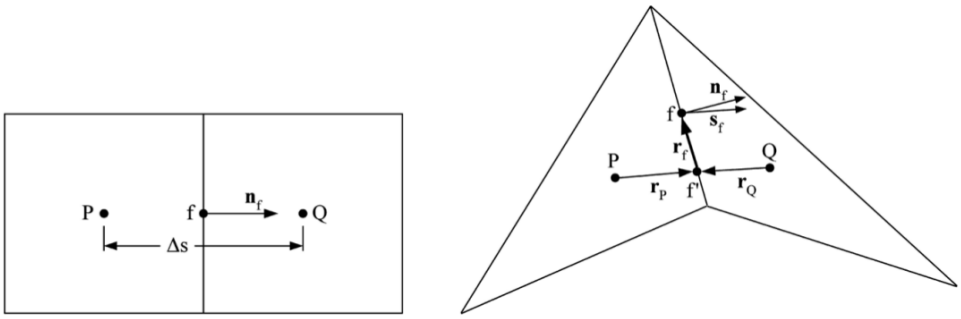


Figure 3.2: Figure from Denner and van Wachem [21]. Mesh cell P with neighboring cell Q and shared face f of an equidistant Cartesian (left) and a triangular (right) two-dimensional mesh including relevant interpolation entities.

Following Denner and van Wachem [21] and using the notation illustrated in figure 3.2 the advection velocity is computed for a two phase flow as

$$\begin{aligned} u_f^n = \boldsymbol{u}_f \boldsymbol{n}_f - \alpha \hat{\boldsymbol{d}}_f \left[\frac{p_Q - p_P}{\Delta s} - \frac{\rho_f}{2} \left(\frac{\nabla p|_P}{\rho_P} + \frac{\nabla p|_Q}{\rho_Q} \right) \right] + c_f \hat{\boldsymbol{d}} \left(u_f^{n,t-\Delta t} - \boldsymbol{u}_f^{t-\Delta t} \boldsymbol{n}_f \right) \\ + \alpha \hat{\boldsymbol{d}}_f \sigma \left[\kappa \frac{C_Q - C_P}{\Delta s} - \frac{\rho_f}{2} \left(\frac{\kappa_P \nabla C|_P}{\rho_P} + \frac{\kappa \nabla C|_Q}{\rho_Q} \right) \boldsymbol{s}_f \right] \\ + \alpha \hat{\boldsymbol{d}}_f \left[\rho_f (\boldsymbol{g} \cdot \boldsymbol{s}_f) - \frac{\rho_f}{2} \left(\frac{\boldsymbol{S}_{g,P}}{\rho_P} + \frac{\boldsymbol{S}_{g,Q}}{\rho_Q} \right) \boldsymbol{s}_f \right] \end{aligned} \tag{3.22}$$

where $\boldsymbol{u}_f$ is the velocity vector at a cell face, $\boldsymbol{n}_f$ is the unit vector normal to a cell face, κ is the interface curvature, σ is the surface tension coefficient, a_P denotes the coefficient associated with the velocity at cell P when the momentum equation is written in semi-discretised form (see section 3.3 or Ferziger and Perić [25])

$$a_P u_P + \sum_{nb} a_{nb} u_{nb} = b, \tag{3.23}$$

where the right hand side term b contains all the terms independent from the velocity and the index nb denotes cells neighbouring cell P. The other terms appearing in equation 3.22 are either illustrated in figure 3.2 or defined below.

$$\alpha = \frac{1}{(\boldsymbol{n}_f \cdot \boldsymbol{s}_f)} \tag{3.24}$$

$$c = \frac{\rho}{\Delta t} \tag{3.25}$$

$$d_P = \frac{V_P}{a_P} \tag{3.26}$$

$$d_f = \frac{d_P + d_Q}{2} = \frac{1}{2}\left(\frac{V_P}{a_P} + \frac{V_Q}{a_Q}\right) \tag{3.27}$$

$$\hat{d}_f = \frac{d_f}{1 + c d_f} \tag{3.28}$$

$$\rho_f = \frac{2}{\rho_P^{-1} + \rho_Q^{-1}} \tag{3.29}$$

The advecting velocity defined in equation 3.22 ensures that the continuity equation 3.20 is fulfilled and it couples pressure and velocity thus preventing the need to solve a Poisson type equation frequently encountered in other CFD frameworks.

Discretisation of advection terms

The advection term in equation 3.16, which represents the transport equation for a general variable ϕ, is given as

$$\int_{V_P} \left(u_j \frac{\partial \phi}{\partial x_j}\right) dV. \tag{3.30}$$

The Gauss theorem, midpoint rule and advection velocity, introduced in the previous section,

are applied to transform the integral over a control volume into a sum over bounding faces

$$\int_{V_P} \left(u_j \frac{\partial \phi}{\partial x_j} \right) dV = \sum \phi_f \boldsymbol{n}_f \boldsymbol{u}_f A_f \approx \sum_f \phi_f u_f^n A_f. \tag{3.31}$$

The value of ϕ at cell faces, in equation is 3.31, is not known, therefore, ϕ_f has to be interpolated from the value of ϕ at neighbouring cell centres. Interpolation methods are also referred to as advection schemes. Several schemes are investigated in the present thesis and an overview of the schemes used is presented below using the notation presented in figure 3.3 and an equidistant Cartesian grid.

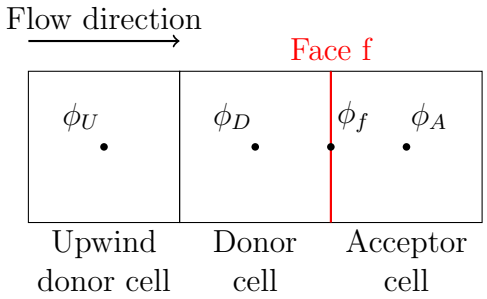


Figure 3.3: Notation used for the discretisation of a general variable ϕ at face f given the flow direction.

The Upwind Differencing Scheme (UDS) is a first-order accurate differencing scheme where, the face value of a variable is defined based on the flow direction, using the notation introduced in figure 3.3, as

$$\phi_f = \phi_D \tag{3.32}$$

The UDS is widely used in convection-dominated flows, it is simple to implement and it guarantees a bounded solution [25]. However, it results in considerable numerical diffusion.

Total Variation Diminishing (TVD) schemes [30, 64] are a class of schemes with higher-order accuracy designed to prevent unphysical oscillations and to minimise numerical diffusion. Using a TVD scheme, the variable ϕ at face f is generally given as

$$\phi_f = \left(1.0 - \frac{\beta_{TVD}(r_f)}{2} \right) \phi_D + \frac{\beta_{TVD}(r_f)}{2} \phi_A, \tag{3.33}$$

where β_{TVD} is the flux limiter that is defined by the applied TVD scheme and r_f is the ratio of the gradients of consecutive cells

$$r_f = \frac{\phi_D - \phi_U}{\phi_A - \phi_D}. \tag{3.34}$$

In the context of this study, the SUPERBEE scheme [56] and the van Leer scheme [70] are considered. The SUPERBEE scheme is the most compressive second-order accurate TVD scheme, with the flux limiter given as

$$\beta_{SB}(r_f) = \max(0, \min(2r_f, 1), \min(r_f, 2)). \tag{3.35}$$

The van Leer scheme is also second-order accurate but more diffusive than the SUPERBEE scheme, with the flux limiter given as

$$\beta_{VL}(r_f) = \frac{r_f + |r_f|}{1 + r_f}. \tag{3.36}$$

Discretisation of diffusion terms

To discretise the diffusion term in equation 3.16, the Gauss theorem and the midpoint rule are used to transform the volume integral into a sum over bounding faces of the control volume

$$\int_{V_P} \frac{\partial}{\partial x_j} \left(\lambda \frac{\partial \phi}{\partial x_j} \right) dV \approx \sum_f \left(\lambda \left(\frac{\partial \phi}{\partial \boldsymbol{x}} \right)_f \boldsymbol{n}_f A_f \right). \tag{3.37}$$

Equation 3.37 requires variable gradients at cell faces $\left(\frac{\partial \phi}{\partial \boldsymbol{x}}\right)_f$. On an equidistant Cartesian grid, these are simply computed as

$$\left(\frac{\partial \phi}{\partial x_i} \right)_f \approx \frac{\phi_A - \phi_D}{x_{i,A} - x_{i,D}}, \tag{3.38}$$

where $x_{i,A}$ and $x_{i,D}$ are the positions of the centres of cell A and cell D along the i-axis. On a

uniform equidistant Cartesian mesh assuming a linear spatial variation of ϕ, this approximation is second order accurate.

3.2.3 Interface discretisation

The numerical representation of a thin interface is challenging and numerical methods have been specifically developed to simulate interfacial flows. The present work uses the compressive VOF method developed in Denner and van Wachem [22] to model the dynamic of the interface. An overview of this method is presented in this section.

Time discretisation

Maintaining a sharp interface is a major challenge for compressive VOF methods and studies have shown that first-order accurate schemes for time discretisation are too diffusive to preserve the sharpness of the interface [16]. Following previous studies ([22, 68]), a second-order accurate Crank Nicolson scheme is used for the time discretisation. The transport equation for a general variable ϕ discretised using the Crank Nicolson scheme reads

$$\frac{(\phi^{t+\Delta t} - \phi^t)V}{\Delta t} = -\sum_f \frac{\phi_f^{t+\Delta t} F_f^{t+\Delta t} + \phi_f^t F_f^t}{2}, \tag{3.39}$$

where superscripts t and $t + \Delta t$ refer to discrete times for two consecutive time steps, Δt is the duration of a time step, f is the index of a face and F is the flux through face f given by

$$F_f = u_f^n A_f, \tag{3.40}$$

where A_f is the area of face f and u_f^n is the advection velocity calculated using momentum interpolation method as detailed in Denner and van Wachem [21]. As proposed byUbbink [67], equation 3.40 can be simplified for small time-steps by assuming $F_f^t = F_f^{t-\Delta t}$, which leads to

$$\frac{(\phi^{t+\Delta t} - \phi^t)V}{\Delta t} = -\sum_f F_f^t \frac{\left(\phi_f^{t+\Delta t} + \phi_f^t\right)}{2}. \tag{3.41}$$

Spatial discretisation of the VOF equation

The CICSAM scheme [68] is used to discretise the advection term of the VOF advection equation. Using the CICSAM scheme, the colour function C at cell face f is given as

$$C_f = (1 - \beta_{f,C})C_D + \beta_{f,C}C_A, \tag{3.42}$$

where subscripts D and A denote the donor and acceptor cells, respectively, as schematically illustrated in figure 3.3. The subscript C indicates that the β-function is related to the discretisation of variable C. The weighting function β_f is defined by the CICSAM scheme and is given as

$$\beta_{f,C} = \frac{\tilde{C}_f - \tilde{C}_D}{1 - \tilde{C}_D}. \tag{3.43}$$

The β-function defined in equation 3.43 uses the normalised colour function

$$\tilde{C}_D = \frac{C_D - C_U}{C_A - C_U}, \tag{3.44}$$

with subscript U denoting the upwind cell, and

$$\tilde{C}_f = \psi_f \tilde{C}_f^{HC} + (1 - \psi_f)\tilde{C}_f^{UQ}. \tag{3.45}$$

The face value C_f^{HC} is obtained using the Hyper-C scheme and C_f^{UQ} is obtained using the ULTIMATE QUICKEST scheme [38]. The blending function ψ_f is defined based on the angle between the normal vector of the interface and the normal vector of face f, as proposed by

Ubbink and Issa [68]. When the interface normal vector is orthogonal to the vector that connects the centres of the acceptor and the donor cells, the value of ψ_f is close to zero and the upwind scheme ULTIMATE QUICKEST is favoured. When the interface normal vector and the vector that connects the centres of the acceptor and the donor cells, a compressive scheme is desired and the Hyper-C scheme is thus favoured.

Choice of time step

In the absence of surface tension, the time step used to solve the flow equations (Δt_{Flow}) is chosen so that the CFL number (also referred to as Courant number Co) is

$$CFL = ||\boldsymbol{u}|| \frac{\Delta t_{\text{Flow}}}{\Delta x} < 1, \tag{3.46}$$

where Δx is the mesh spacing, in the entire computational domain and $||\boldsymbol{u}||$ is the norm of the velocity. However, to preserve the interface sharpness, the CICSAM scheme imposes further constraints on the applied time-step because it accounts for the volume of a given fluid in the donor cell. Previous studies [29, 22] recommended a time-step that satisfies the condition $CFL < 0.01$. Because the discretised VOF advection equation is solved separate from the discretised flow equations in the employed numerical framework, different time-steps can be applied. To this end, the VOF time-step is chosen to be an integer multiple of the flow time-step, given as

$$\Delta t_{\text{VOF}} = \frac{\Delta t_{\text{Flow}}}{N_{\Delta tVOF}}, \tag{3.47}$$

where $N_{\Delta tVOF}$ is the number of VOF time steps Δt_{VOF} within one flow time step Δt_{Flow} required to satisfy the condition $CFL < 0.01$.

3.2.4 Discretisation of the energy equation

The viscosity of corium is strongly temperature-dependent, therefore, errors in temperature prediction lead to large error in viscosity calculation and thus affect the flow predictions. A detailed study of numerical errors related to the discretisation of the energy equation is presented in chapter 5.

In this thesis, the author proposes to apply the CICSAM scheme commonly used for the discretisation of the VOF equation to the discretisation of the energy equation. The discretisation of the advection term in the energy equation using the CICSAM scheme, which was originally developed for the discretisation of the VOF colour function, is similar to the process described in section 3.2.3, merely inserting the temperature T in equation 3.42. As demonstrated in chapter 5, the choice of a suitable solution procedure is critical to obtain accurate results when using the CICSAM scheme to discretise the energy equation. Section 3.3 describes the different solution procedures investigated as part of this thesis.

As a matter of comparison, the study presented in this thesis also includes simulations using the different advection schemes introduced in section 3.2.2 for the discretisation of the energy equation.

3.3 Solution procedure

The governing equation are introduced in section 3.1 and the discretisation process to transform the partial differential equations into systems of algebraic equation is presented in section 3.2. In this section the procedure followed to solve the discretised equations is described.

Section 3.3.1 details how to write a linear system of algebraic equations into a matrix form to be solved by a linear solver, section 3.3.2 describes the general method to solve a non linear system of equation and section 3.3.3 describes the fully coupled methodology, developed by Denner and van Wachem [21], to solve the system of coupled equations.

It is shown, in chapter 5, that the CICSAM scheme, with an appropriate solution procedure, can be used for the discretisation of the energy equation in order to minimise numerical diffusion temperature. Section 3.3.4 presents different strategies, investigated in the present work, to include the energy equation in the solution procedure.

3.3.1 Solution of a linear system

After the discretisation of an equation has been performed, the partial differential equation is replaced by a system of algebraic equations. The system can be written in matrix form and solved using appropriate numerical solution methods. In this section, the steps to reach a matrix formulation of the algebraic equation system are illustrated using a simple linear partial differential equation solved on a equidistant one-dimensional mesh

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = RHS, \tag{3.48}$$

where ϕ is the dependent variable to be computed, u is a steady advection velocity and RHS represent terms independent of ϕ.

Equation 3.48 is discretised using a backward Euler scheme for temporal discretisation and a Central Differencing Scheme for the spatial discretisation. Using the notation illustrated in

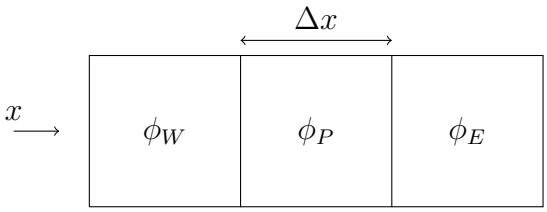


Figure 3.4: Illustration of the notation used to identify cells. Cell P is the cell under consideration and cells W and E are the western and eastern neighbour cells respectively.

figure 3.4, the discretised equation in a particular cell P is given by

$$\Delta x(\phi_P^{t+\Delta t} - \phi_P^t) + u_P \Delta t \frac{\phi_E^{t+\Delta t} - \phi_W^{t+\Delta t}}{2\Delta x} = \Delta x \Delta t RHS. \tag{3.49}$$

The different terms in equation 3.49 can be grouped in three categories

- Terms containing $\phi^{t+\Delta t}$ at cell P
- Terms containing $\phi^{t+\Delta t}$ at cells neighbouring P
- Terms that are independent of $\phi_P^{t+\Delta t}$.

When running a simulation, values of variable ϕ at each time step are computed sequentially therefore, when solving for $\phi^{t+\Delta t}$, the value of ϕ^t is known. The terms in the equation containing ϕ^t are thus treated as right hand side terms. The rearranged equation then reads

$$\Delta x \cdot \phi_P^{t+\Delta t} + \frac{u_P \Delta t}{2\Delta x} \cdot \left(\phi_E^{t+\Delta t} - \phi_W^{t+\Delta t}\right) = \Delta x \Delta t \cdot RHS + \Delta t \cdot \phi^t. \tag{3.50}$$

Introducing a_P, a_{nb} and b coefficients (the coefficients are dependent on the numerical grid and the discretisation schemes used), the equation is reformulated as

$$a_P \phi_P + \sum_{nb} a_{nb} \phi_{nb} = b. \tag{3.51}$$

Considering all the cells in the solution domain, a system of linear algebraic equation can be written in matrix form as

$$A_\phi \boldsymbol{\phi} = \mathbf{b}_\phi, \tag{3.52}$$

where $\boldsymbol{A}_\phi$ is a matrix of coefficients and $\boldsymbol{b}_\phi$ is a vector with all the ϕ independent terms for each cell. The expanded matrix is shown in equation 3.53.

$$\underbrace{\begin{pmatrix} a_{0,0} & a_{0,1} & 0 & \cdots & & 0 \\ a_{1,0} & a_{1,1} & a_{1,2} & & & \\ 0 & a_{2,1} & a_{2,2} & \ddots & & \vdots \\ & \ddots & \ddots & a_{i-1,i} & & \\ \vdots & & a_{i,i-1} & a_{i,i} & \ddots & 0 \\ & & & \ddots & \ddots & a_{N-1,N} \\ 0 & \cdots & & 0 & a_{N,N-1} & a_{NN} \end{pmatrix}}_{A} \cdot \underbrace{\begin{pmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \phi_i \\ \vdots \\ \phi_N \end{pmatrix}}_{\boldsymbol{\phi}} = \underbrace{\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_i \\ \vdots \\ b_N \end{pmatrix}}_{\boldsymbol{b}} \tag{3.53}$$

The coefficients in red are the coefficients related to ϕ_i in the i^{th} equation which correspond to the ϕ_P terms in equation 3.51. They are referred to as diagonal coefficients. The coefficients in blue correspond to the ϕ_{nb} terms in equation 3.51. They are referred to as off-diagonal coefficients. The b_i coefficients are the terms independent from ϕ and are referred to as right hand side coefficients or RHS coefficients.

The structure of matrix A differs depending on the type of computational mesh (structured mesh, arbitrary mesh) and the number of spatial dimension but the sparsity of the matrix (many zero elements) and the distinction between diagonal and off-diagonal coefficients remains.

Once the matrix form of the algebraic equation system is obtained, it can be solved using efficient numerical solvers and the values of ϕ at the next time steps can be computed. The solution methods are not described in the present work but the interested reader can learn about the most common solvers in [25].

3.3.2 Solution of a non-linear system

In a non-linear problem, some terms can not be written as a linear combination of the dependent variable as $a \cdot \phi$ where a is a coefficient independent of ϕ. Instead, the non linear term are written as $a(\phi) \cdot \phi$ where the $a(\phi)$ is a coefficient dependent on ϕ. The system of equation can not be solved as such therefore the equation needs to be linearised by making an approximation for all the non-linear coefficients. Typically, the value of the non-linear coefficients is evaluated using the latest available value of ϕ. The linearised system is then solved and a new solution for ϕ is obtained. This new solution is used in turn to update the non-linear coefficient. The sequence of update of the non-linear coefficients and solution of the linearised system is repeated until the value of ϕ converges. The convergence is assessed by evaluating the residual of the non-linear problem and comparing it with a predefined convergence criterion. Once convergence is reached, the solution of the problem at the next time step can start. The process is illustrated in figure 3.5.

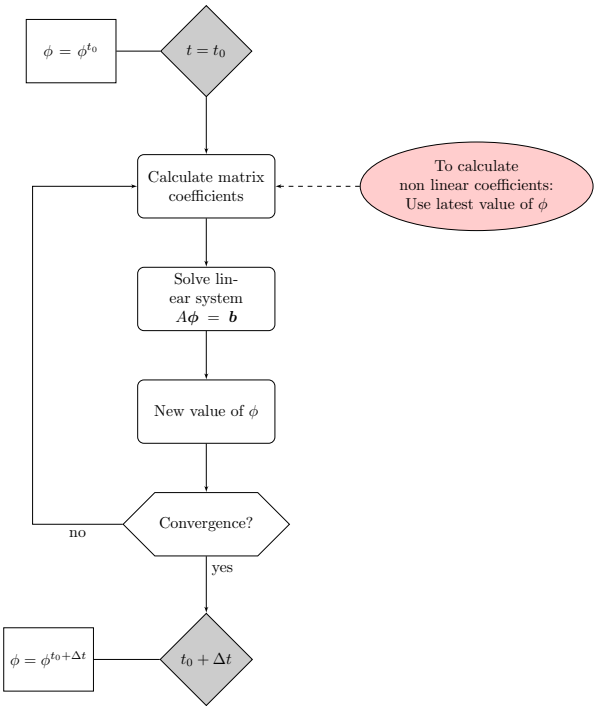


Figure 3.5: Flowchart describing the procedure to solve a non linear system of equation using a linear numerical solver.

3.3.3 Fully coupled solver

The equations presented in section 3.1 are coupled: the solution of the continuity equation requires values for the velocities u, v and w, which are computed by solving the momentum equations. To solve the coupled system of equation, the present work follows a coupled implicit approach for the fluid flow as presented by Denner and van Wachem [21]. A linearised equation system describing the flow is constructed with the three momentum equations and the continuity equation. The linear system is composed of four independent equations and four unknown variables (three velocities u, v, w and pressure p) and it needs to be solved at every discrete point in space. The assembled equation system describing the fluid flow is given by

$$\underbrace{\begin{pmatrix} A_u^x & 0 & 0 & A_p^x \\ 0 & A_v^y & 0 & A_p^y \\ 0 & 0 & A_w^z & A_p^z \\ A_u^c & A_v^c & A_w^c & A_p^c \end{pmatrix}}_{\boldsymbol{A}_{fc}} \cdot \begin{pmatrix} u \\ v \\ w \\ p \end{pmatrix} = \boldsymbol{b} \ , \tag{3.54}$$

where $\boldsymbol{A}_{fc}$ is the coefficient matrix of the fully coupled system and b is the right hand side vector. The elements A_j^i of matrix $\boldsymbol{A}_{fc}$ correspond to the coefficient related to variable j in the ith system of equation (indexes x, y and z denote the momentum equations and index c denotes the last equation). For example A_u^x represent the matrix of coefficients related to variable u in the x-momentum equation. It should be noted that the same time step (Δt_{Flow}) has to be used for the solution of the different equations in the system 3.54.

There are two more equations to be solved: the energy and VOF equations. Different strategies for the solution of the energy equation are investigated in this thesis: the energy equation can be added to the linear system 3.54 as a fifth equation or the energy equation can be solved separately. The solution of the energy equation is particularly important for the simulation of corium spreading flows because of the strong temperature-dependency of the corium viscosity. The different solution procedures investigated are detailed in Section 3.3.4.

The solution of the VOF equation follows the methodology developed by Denner and van Wachem [21]. The VOF equation is solved separately, prior to the solution of the fully-coupled system 3.54, by constructing a separate linear system. Solving the VOF equation separately allows the use of a smaller time step (Δt_{VOF}).

The steps to compute the value of the dependent variables C, $\boldsymbol{u}$ and p at discrete time $t_0+\Delta t_{Flow}$ starting from discrete time t_0 are illustrated in figure 3.6.

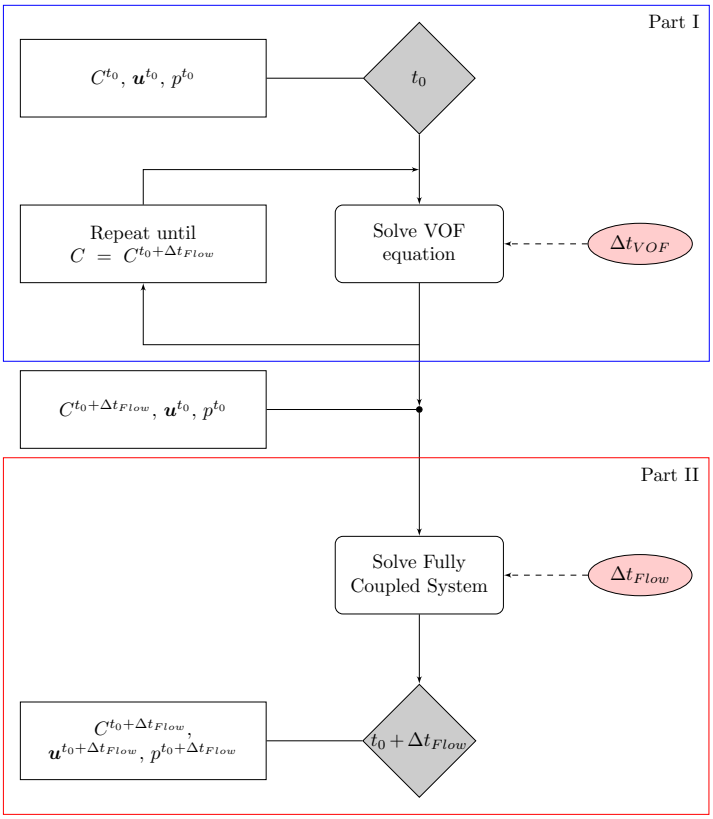


Figure 3.6: Flowchart describing a solution procedure where the VOF equation is solved first with time step Δt_{VOF} and the fully coupled system $[\boldsymbol{u}, p]$ is, then, solved with time step Δt_{Flow}.

In the figure 3.6, Part I corresponds to the solution of the VOF equation and Part II corresponds to the solution of the fully coupled system. These two parts are detailed in the flowcharts in figure 3.7.

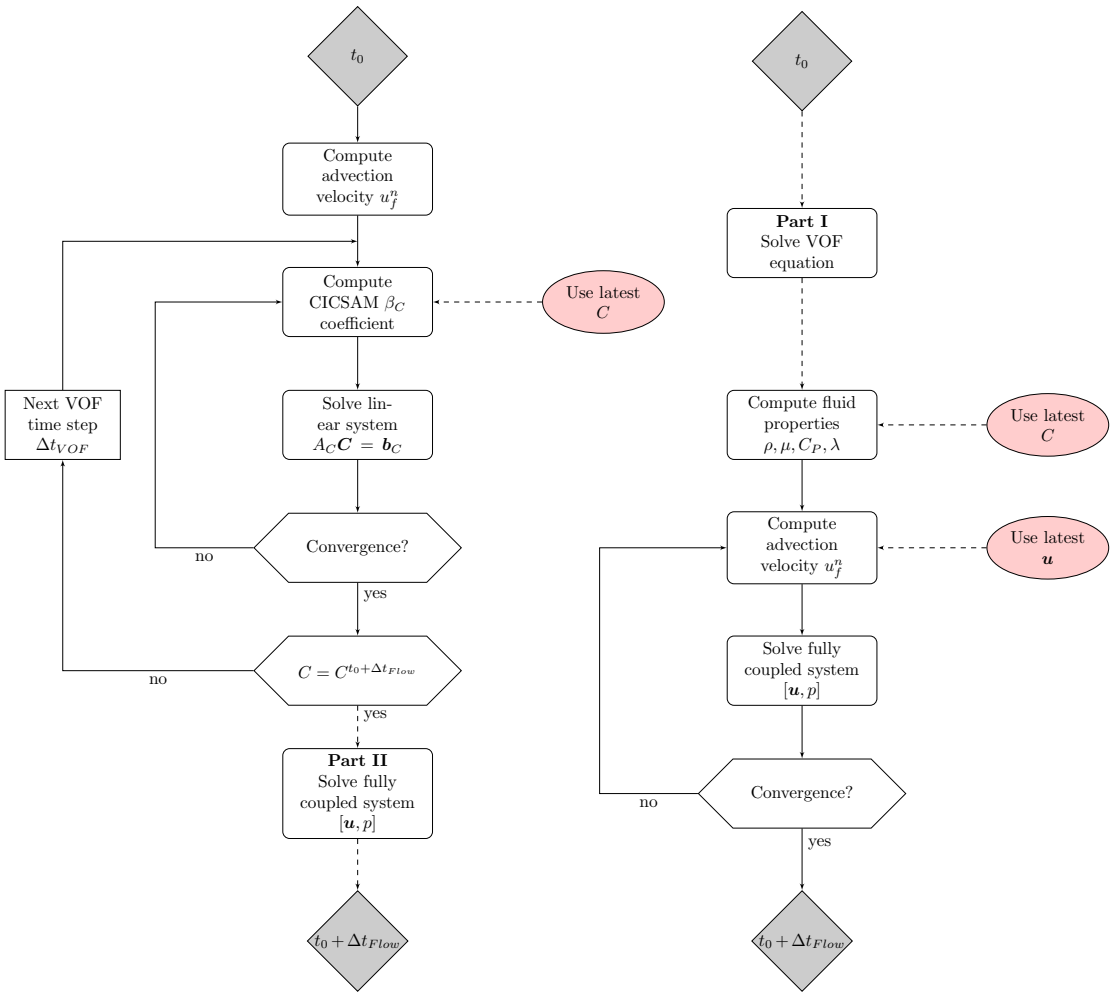


Figure 3.7: Focus on Part I (left) and Part II (right) of the flowchart presented in figure 3.6. Part I describes the procedure to solve the VOF equation and Part II describes the solution procedure to solve the fully coupled system formed with the momentum and continuity equations.

Description of Part I: In figure 3.7, the advection velocity introduced in Section 3.2.2 is computed using the current velocity values $\boldsymbol{u}^{t_0}$. Then, the CICSAM advection scheme requires to compute the β_C coefficients. The β_C coefficients are dependent on the colour function C therefore the VOF equation is non-linear and a non-linear loop is required to reach convergence. As discussed previously, the time step used to solve the VOF equation (Δt_{VOF}) is smaller than the flow time step (Δt_{Flow}), therefore, the VOF equation system has to be solved repeatedly until the the colour function values at time $t = t_0 + \Delta t_{Flow}$ are obtained.

Description of Part II: Figure 3.7 details the procedure to solve the fully-coupled system 3.54. First, the fluid properties are calculated for the individual fluids of the two-fluid system. Then, the phase averaged fluid properties are computed as a function of the individual fluid

properties and the colour function. Different methods to calculate the phase averaged fluid properties from the individual fluid properties are investigated in this thesis. These methods are presented in chapter 5. The next step in the solution procedure is the calculation of an advection velocity using the latest available velocity values. Since the advection velocity is dependent on the variables being computed ($\boldsymbol{u}$), the problem is non-linear and a non-linear loop is necessary to reach a converged solution.

3.3.4 Solution of the energy equation

The solution of the energy equation is particularly important for the simulation of corium spreading flows because of the strong temperature-dependency of the corium viscosity. Small errors in the temperature prediction can cause large errors in the calculation of corium viscosity and thus strongly affect the flow.

In this section different approaches to include the solution of the energy equation into the solution procedure described in section 3.3.3 are presented. The different solution procedures are named A, B and C, and are illustrated in figures 3.8, 3.9 and 3.10 respectively.

Solution procedure A is shown in figure 3.8 and follows traditional solution methods associated with compressive VOF calculations [29], [22]. With this solution procedure, the VOF equation is solved repeatedly using a small time step Δt_{VOF} until the colour function at time $t = t^{n+1}$ is obtained. The new value of the colour function is then used to update the fluid properties ρ, λ and c_P (see equation 5.1). In the present work, viscosity μ is the only temperature-dependent property updated using the currently available values of colour function and temperature (values at time $t = t^{n+1}$ for the colour function and values at time $t = t^n$ for the temperature). A fully-coupled system, with the flow and energy equations, is then solved using a larger time step Δt_{Flow} to compute the values of the pressure, the velocity and the temperature at time $t = t^{n+1}$.

With solution procedure B described in figure 3.9, the VOF equation is solved first with a small time step Δt_{VOF} until the colour function at time $t = t^{n+1}$ is obtained. The new value

of the colour function is then used to update the fluid properties ρ, λ and c_P. The energy equation is then solved with the same small time step Δt_{VOF} until the temperature at time $t = t^{n+1}$ is obtained. The viscosity μ is updated using the new values of the colour function and temperature. Finally the fully-coupled system, with the flow equations, is solved using a larger time step Δt_{Flow}.

With solution procedure C described in figure 3.10, the VOF equation and the energy equation are solved at the same time with a small time step Δt_{VOF}. The fluid properties ρ, λ and c_P are recalculated at every time step using the latest value of the colour function. When the values of the colour function and temperature at time $t = t^{n+1}$ are obtained, they are used to calculate μ and the fully-coupled system, with the flow equations, is solved using a larger time step Δt_{Flow}.

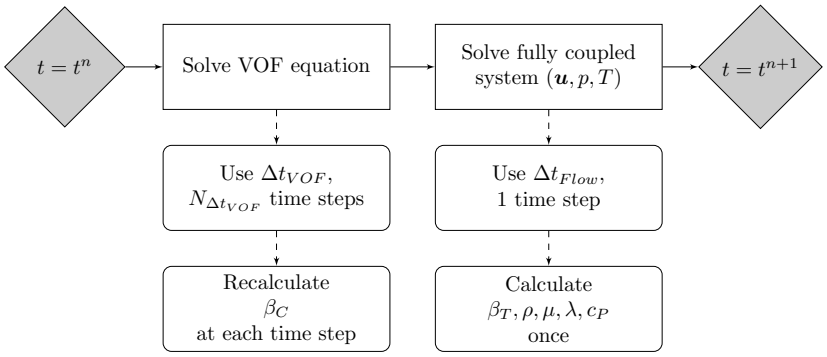


Figure 3.8: Solution procedure A to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF equation is solved separately but the flow and energy equations are solved in the same time loop.

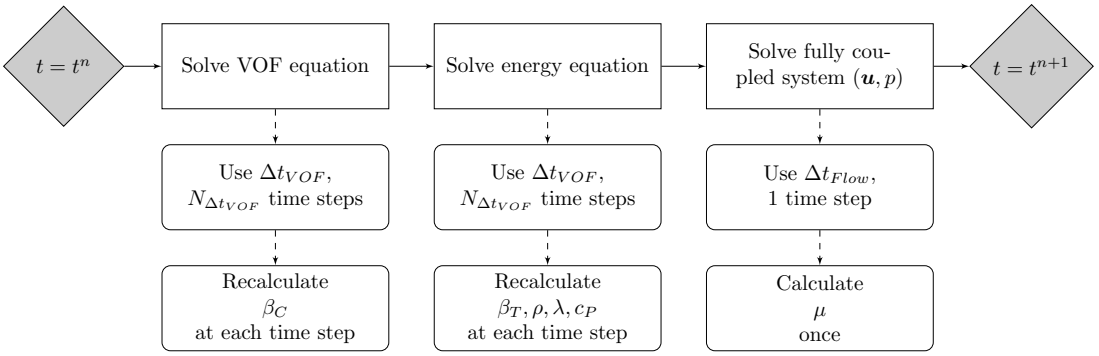


Figure 3.9: Solution procedure B to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where flow, VOF and energy equations are solved in three separate time loops.

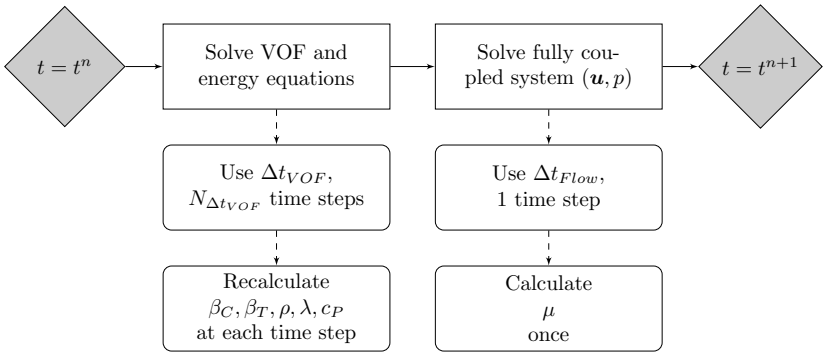


Figure 3.10: Solution procedure C to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF and energy equation are solved in the same time loop and the flow equations are solved in a second time loop.

Chapter 4

CFD simulation of Bingham Fluid

The rheology of corium is very complex due to large temperature variations, corium solidification and changes in corium composition during a nuclear severe accident. Corium modelling is also adversely affected by the large uncertainties on corium properties. Experimental programs, such as the experiments performed in the VITI facility at CEA Cadarache [50], dedicated to the measurements of corium properties have been conducted to improve corium understanding.

Based on experimental and analytical investigations of corium properties, Piluso et al. [50] proposes guidelines for the modelling of corium rheology. The suggested modelling strategy distinguishes between corium above liquidus temperature (purely liquid) and corium below liquidus temperature (solidifying). For solidifying corium, the fluid is modelled as a Bingham non-Newtonian fluid and using an apparent viscosity model where corium viscosity is a function of corium solid fraction. For purely liquid corium, different models are suggested depending on the corium composition. If the corium doesn't contains silicate (prior to mixing with concrete), the corium is modelled as a Newtonian fluid using the model proposed by Andrade [6]. If the corium contains silicate, it is modelled using the model developed by Urbain [69]. The work presented in this thesis aims at simulating corium spreading flows associated with the EPR core-catcher concept ([13] [27]), described in chapter 1. With the EPR core-catcher, during the spreading phase, corium temperature is expected to be below the liquidus temperature and, thus, solidifying.

This chapter focuses on the implementation of a Bingham fluid model in a CFD code. The study of Bingham fluids using numerical methods has been going on for many decades ([11], [43], [45]), however, the implementation of a regularised Bingham fluid model, see section 4.1, in a finite volume framework is rarely described in great details. In the work presented in this thesis, all the steps involved in the implementation of the Bingham fluid model are explained, and, thus, the results presented in this thesis could be easily replicated and improved in future research work. Section 4.1 describes the constitutive law of a Bingham fluid and the regularisation methods used to implement it in a CFD code. Section 4.2 describes how the Bingham fluid model is implemented in finite volume CFD framework used in this thesis. Section 4.3 shows simulation results using the implemented Bingham model on a lid driven cavity test case.

4.1 Bingham fluid model

Bingham fluids are a type of fluid, with a non-Newtonian behaviour, characterized by a yield stress τ_S. If the shear stress applied on the fluid is lower than the yield stress, the fluid behaves as a rigid body, if the yield stress is exceeded, the fluid behaves as a viscous incompressible fluid. This behaviour is described by the constitutive law

$$\begin{cases} |\tau| \geq \tau_S & \iff \tau_{ij} = (2\mu + \frac{\tau_S}{|S|})S_{ij} \\ |\tau| < \tau_S & \iff S_{ij} = 0. \end{cases} \tag{4.1}$$

The constitutive law 4.1 relates the stress tensor τ, introduced in chapter 3, to the strain rate tensor S defined as:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{4.2}$$

The norm of the strain rate tensor S is given by

$$|S| = \left(\sum_{1<i,j<3} (S_{ij})^2 \right)^{1/2}. \tag{4.3}$$

Figure 4.1 presents the stress as a function of the strain rate for a Newtonian and a Bingham fluid.

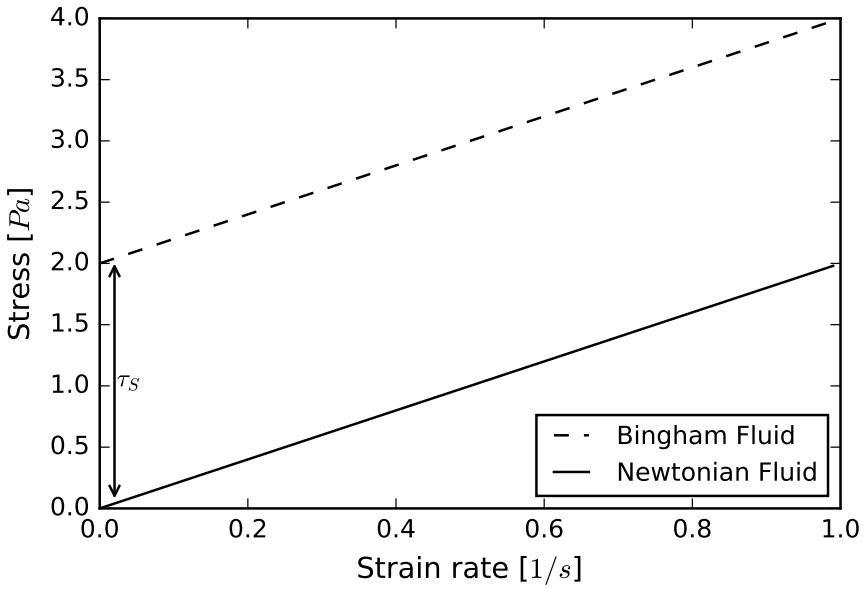


Figure 4.1: Stress rate as a function of strain rate for a Newtonian fluid and a Bingham fluid.

In a CFD framework, the governing equations are usually solved to compute the fluid velocity, and, the velocity is then used to compute the strain rate tensor. Due to the finite precision of numerical solvers, the calculation of a null velocity never occurs and, therefore, the rigid body behaviour of the fluid (second part of the constitutive law described in by the system 4.1) can't be modelled. A common strategy to overcome this issue is to use regularised models approaching the constitutive law of Bingham fluids. With the regularisation strategies presented in this chapter, the strain rate tensor S is proportional to an effective viscosity μ_{eff}, which depends on the fluid viscosity and the norm of the stress rate tensor, given as

$$\tau_{ij} = 2\mu_{eff}(\mu, |S|)S_{ij}. \tag{4.4}$$

Several models for the effective viscosity are found in the literature. In the present work, the

BE model proposed by Bercovier and Engelman [11], the Papanoustiou model proposed by Papanastasiou [45] and the bi-viscosity model proposed by O'Donovan and Tanner [43] are considered.

The **Bercovier-Engelman** model [11]:

$$\mu_{eff} = \mu + \frac{1}{2}\tau_S \left(\frac{1}{(\epsilon_{\text{BE}} + |S|^2)^{1/2}} \right) \tag{4.5}$$

The **Papanastasiou** model [45]:

$$\mu_{eff} = \mu + \frac{1}{2}\tau_S \left(\frac{1 - e^{-|S|/\epsilon_{\text{Papa}}}}{|S|} \right) \tag{4.6}$$

The **bi-viscosity** model [43]:

$$\begin{cases} \mu_{eff} = \mu + \frac{1}{2}\frac{\tau_S}{|S_{ij}|} & |S| \geq \frac{\tau_S}{(C_{BV}-1)\mu} \\ \mu_{eff} = \mu * C_{\text{BV}} & |S| < \frac{\tau_S}{(C_{\text{BV}}-1)\mu} \end{cases} \tag{4.7}$$

The models described in equations 4.5, 4.6 and 4.7 make use of model parameters (ϵ_{BE}, ϵ_{Papa} and C_{BV} respectively) that control the strength of the regularisation. For the Bercovier-Engelman and Papanoustiou models, when the ϵ_{BE} and $\epsilon_{\text{Papanoustiou}}$ parameter tend toward zero or for the bi-viscosity model when the C_{BV} value becomes large, the models approach the Bingham behaviour. Figure 4.2 illustrates the regularised models with different parameters as well as the original Bingham constitutive law.

An alternative to regularisation is to solve the equations using variational methods. Variational methods are not described in this thesis, but, the interested reader is referred to the work of Dean et al. [18] for a review of Bingham fluid modelling using variational methods.

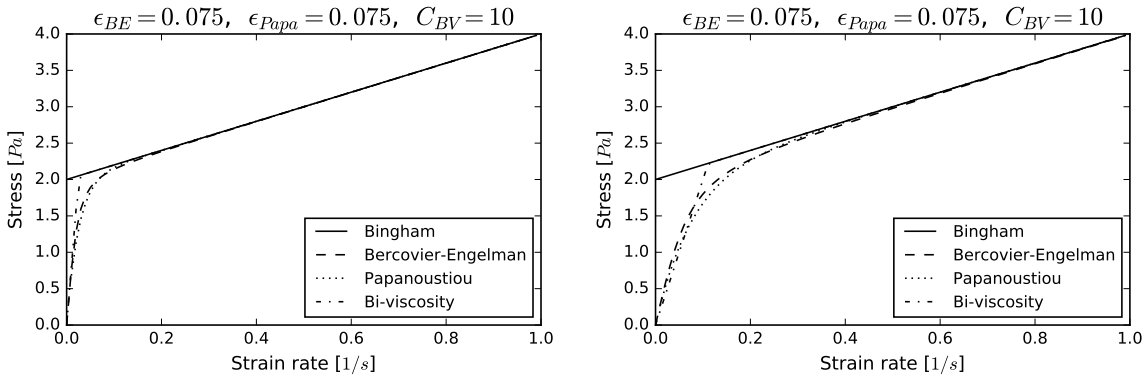


Figure 4.2: Stress as a function of strain rate for a Bingham fluid and different regularised Bingham models. The regularised models (Bercovier-Engelman model, Papanoustiou model and bi-viscosity models) make use of model parameters (ϵ_{BE}, ϵ_{Papa} and C_{BV} respectively) that control the strength of the regularisation.

4.2 Implementation of a Bingham fluid model in a CFD code

The regularised Bingham model affects the diffusion term in the momentum equations (viscous term). Following the finite volume framework described in section 3, the viscous term in the momentum equation is given for a Newtonian flow as

$$\int_{V_P} \frac{\partial}{\partial x_j}\left(\mu \frac{\partial u_i}{\partial x_j}\right) dV \approx \sum_f \left(\mu_f \left(\frac{\partial u_i}{\partial \boldsymbol{x}}\right)_f \boldsymbol{n}_f A_f\right) \tag{4.8}$$

This term requires values of the viscosity and of the velocity gradient at cell faces. As recommended by Patankar [46], the face viscosity is calculated using the harmonic average of the viscosities at adjacent cell centres. For a face f at the intersection of cell P and cell Q, the harmonic average is given by

$$\mu_f = \frac{\mu_P \cdot \mu_Q}{\mu_P + \mu_Q}. \tag{4.9}$$

Assuming a Cartesian equidistant mesh, the value of the velocity gradients at cell faces is computed by linear interpolation as

$$\left(\frac{\partial u_i}{\partial x_j}\right)_f \approx \frac{u_P - u_Q}{x_{j,P} - x_{j,Q}}, \tag{4.10}$$

where $x_{j,P}$ and $x_{j,Q}$ are the j-coordinate of cell P and cell Q respectively. To simulate a Bingham fluid using the regularisation methods presented in section 4.1, the fluid viscosity is replaced by a Bingham viscosity μ_{Bingham}. Equation 4.11 presents the general formula to compute the Bingham viscosity

$$\mu_{f,\text{Bingham}} = f_{\text{Bingham}}(\mu_f, |S_f|), \tag{4.11}$$

where $f_{Bingham}$ represents one of the models presented in section 4.1. The values of the strain rate tensor at cell faces is computed using the velocity gradients at cell faces from equation 4.10.

$$S_{ij,f} = \frac{1}{2}\left(\left(\frac{\partial u_i}{\partial x_j}\right)_f + \left(\frac{\partial u_j}{\partial x_i}\right)_f\right) \tag{4.12}$$

The Bingham viscosity is then recalculated at each time step based on the new values of the velocity.

4.3 Test case: lid driven cavity

The lid driven cavity test case is used to validate the implementation of the Bingham model in the CFD framework. The lid driven cavity test case is an isothermal test case that considers a two-dimensional square geometry with three static wall and a moving wall at the top. A constant velocity of 1 m/s is prescribed at the top boundary, which drives the fluid movement inside the cavity. The fluid in the cavity is initially at rest and a no slip boundary condition is applied on the static walls. The fluid density is $\rho = 1000$ kg.m^{-3} and the fluid viscosity is $\mu = 1$ Pa.s corresponding to a Reynolds number of $Re = 1000$. The choice of time step and mesh size is discussed in sections 4.3.1 and 4.3.2.

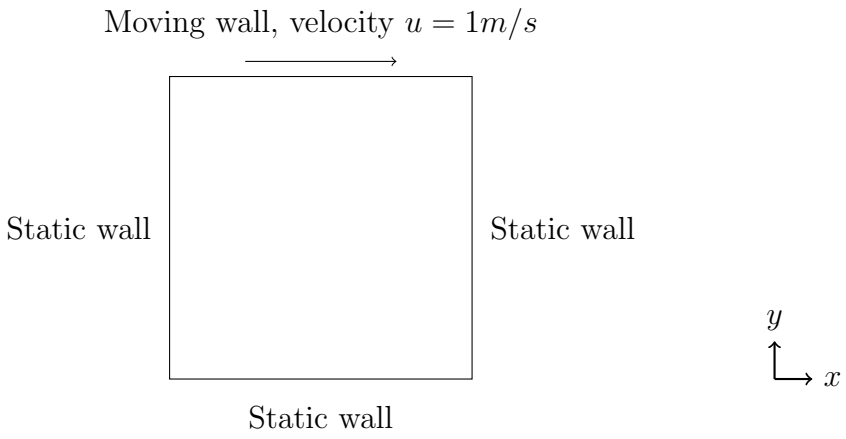


Figure 4.3: Schematic description of the lid driven cavity test case. The top wall is moving with velocity 1 m/s and driving the fluid movement inside the cavity

The main quantitative results of the lid driven cavity test case, which can be compared to values from the literature ([12], [73]), are the features of the vortices that form in the cavity. The position and intensity of the vortices is obtained by finding the value and position of the local maxima of the stream function ψ, which is computed by solving the partial differential equation

$$\nabla^2\psi(x,y) = \frac{\partial u_y}{\partial x} - \frac{\partial u_x}{\partial y} \tag{4.13}$$

where u_x and u_y are the two components of the velocity in the two-dimensional simulation.

4.3.1 Newtonian fluid in a lid driven cavity

The lid driven cavity has been extensively studied in the literature to benchmark CFD codes on Newtonian fluid simulation. In this section, simulations of a Newtonian fluid in a lid driven cavity are used to benchmark the present CFD framework with results from the literature and to run grid sensibility studies. The simulation is run on a 80x80 mesh with a time step size of $0.25s$.

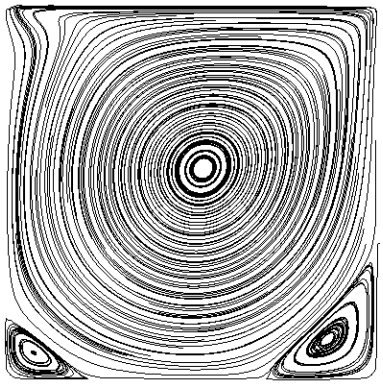


Figure 4.4: Velocity streamlines for the simulation of a Newtonian fluid in a lid driven cavity

In figure 4.4, which shows the velocity streamlines, several vortices are visible. There is a principal vortex in the centre and two secondary vortices in the bottom left and bottom right corners. The convergence of the simulation is assessed by investigating the evolution of the quantities of interest (principal vortex position and intensity) with time. The evolution of the principal vortex intensity with time is plotted in figure 4.5.

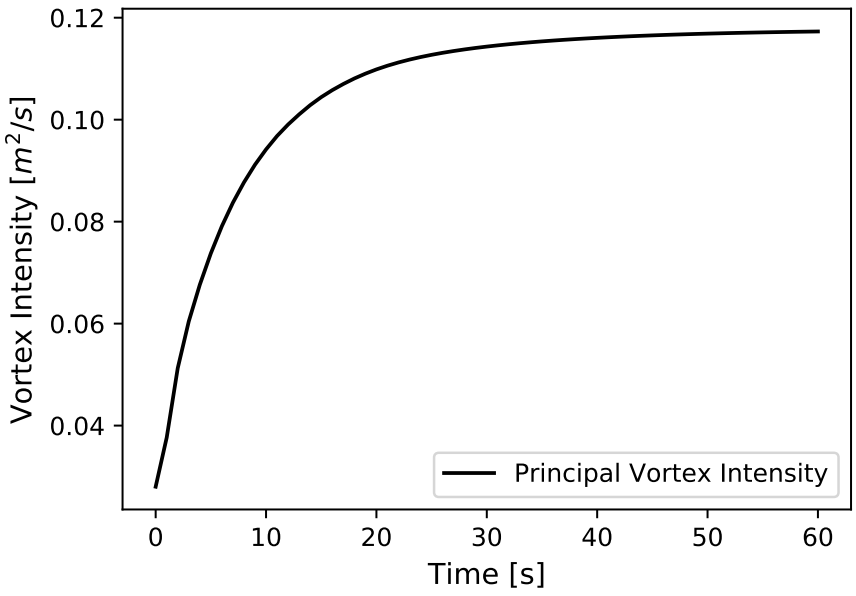


Figure 4.5: Evolution of the intensity of the principal vortex with time for the simulation of a Newtonian fluid in a lid driven cavity

In figure 4.5, after about thirty seconds of simulation, the principal vortex intensity reaches a

converged value. The position of the centre of the principal vortex converges quickly, so, only the principal vortex intensity is used to assess the convergence of the simulation. Table 4.1 reports the results, for the intensity and position of the vortices, obtained in the present work using several grid resolutions as well as other results from the literature.

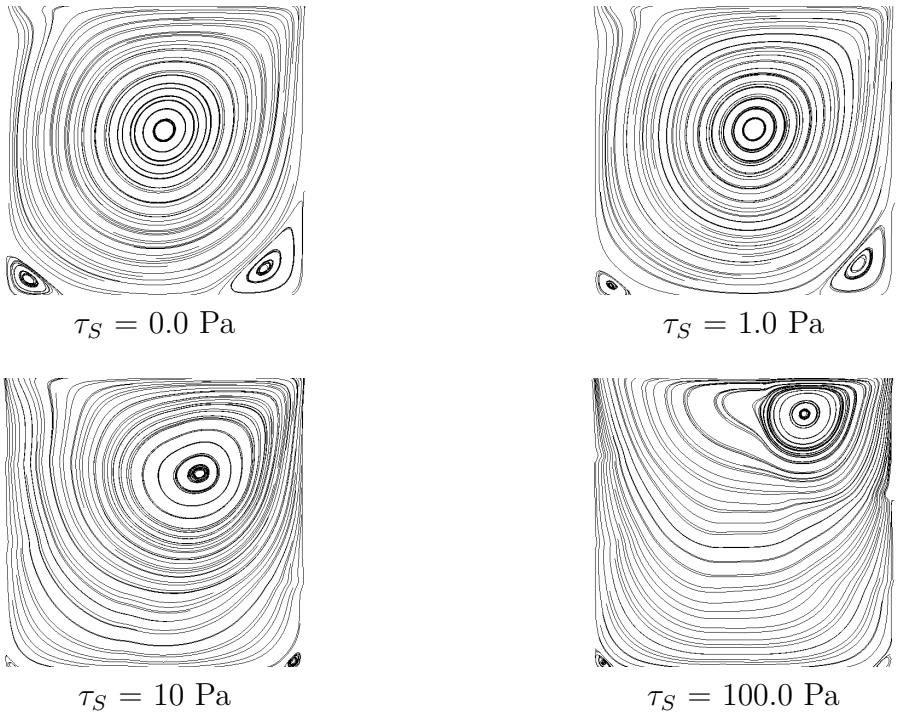
Table 4.1: Position and intensity of the principal vortex for the simulation of a Newtonian fluid in a lid driven cavity using several mesh resolutions and comparison with results from the literature

Reference (Grid size)	**Intensity** $[m^2/s]$	**Position** (x,y) $[m]$
	Principal vortex	
Present Work (40x40)	0.1144	(0.5375, 0.5875)
Present Work (60x60)	0.1174	(0.5250, 0.5750)
Present Work (80x80)	0.1181	(0.5312, 0.5688)
Present Work (100x100)	0.1185	(0.5350, 0.5750)
Vola et al. [73] (80x80)	0.1162	(0.525, 0.5625)
Botella and Peyret [12]	0.1189	(0.531, 0.5652)
	Bottom right vortex	
Present Work (40x40)	0.00007	(0.8375, 0.0875)
Present Work (60x60)	0.00056	(0.8417, 0.0917)
Present Work (80x80)	0.00093	(0.8438, 0.0938)
Present Work (100x100)	0.00109	(0.8550, 0.0950)
Vola et al. [73] (80x80)	0.00170	(0.8625, 0.1125)
Botella and Peyret [12]	0.00173	(0.864, 0.118)

The features of the principal vortex converge towards the values from the literature, both in terms of intensity and position, as the mesh is refined. The position of the centre of the bottom right vortex matches literature results, but, there are discrepancies in the vortex intensity.

4.3.2 Bingham fluid in a lid driven cavity

In this section, a Bingham fluid in a lid driven cavity is simulated. Except for the Bingham fluid model, all the simulation parameters are unchanged compared to section 4.3.1. Figure 4.6 shows results, on a 100x100 mesh with a time step of 0.25 s, using a Newtonian fluid and a Bingham fluid with yield stress values 1 Pa, 10 Pa and 100 Pa. For the Bingham fluid simulation, the Papanoustiou model is used to calculate the effective viscosity with model parameter $\epsilon = 0.01$.



$\tau_S = 0.0$ Pa $\tau_S = 1.0$ Pa

$\tau_S = 10$ Pa $\tau_S = 100.0$ Pa

Figure 4.6: Streamlines of the velocity for a Bingham fluid in a lid driven cavity using different yield stress values (the case with $\tau_S = 0$ corresponds to a Newtonian fluid).

In figure 4.6, it appears that the principal vortex is shifted towards the top right corner as the yield stress is increased. When the yield stress value is high, the region at the bottom, where the velocity is small, the Bingham viscosity is high and the fluid behaves as a solid. In the top region, where the velocity is higher, the fluid is still flowing. Therefore, the principal vortex is shifted towards the top and the bottom vortices disappear.

The effect of the model constants of the regularised models is investigated using the lid-driven cavity test case and a Bingham fluid with yield stress $\tau_S = 10$ Pa. In particular, figure 4.7 presents the evolution of the intensity of the principal vortex intensity with time. Simulations are run for a Newtonian fluid and a Bingham fluid, with yield stress $\tau_S = 10$ Pa, using the BE model and different values of the ϵ_{BE} constant.

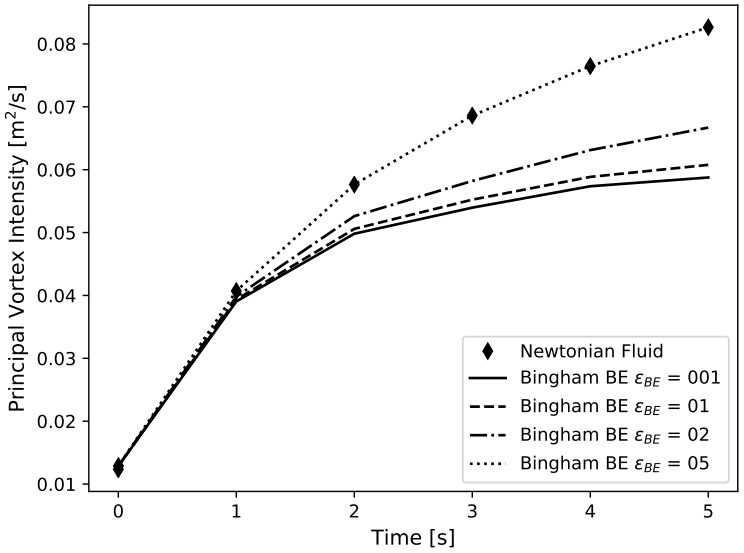


Figure 4.7: Evolution of the intensity of the principal vortex with time, for the lid driven cavity test. Simulations are run for a Newtonian fluid and for a Bingham fluid, with yield stress $\tau_S = 10$ Pa, using the BE model with different value of the model constant ϵ_{BE}.

For large values of the ϵ_{BE} constant, the Bingham viscosity tends toward the Newtonian fluid viscosity, and, the results are, thus, similar to the results for a simulation of a Newtonian fluid. When the value of the ϵ_{BE} constant decreases, there is a convergence of the principal vortex intensity value. However, small values of the ϵ_{BE} constant increase the non-linearity of the simulation and smaller time steps are required to avoid the divergence of the simulation.

To investigate the effect of the choice of regularised model on the results, simulations, illustrated in figure 4.8, are run using the BE, Papanoustiou and bi-viscosity models with values of the constant that yield converged results ($\epsilon_{\mathrm{BE}} = 0.01$, $\epsilon_{\mathrm{Papa}} = 0.01$ and $C_{\mathrm{BV}} = 100$ respectively).

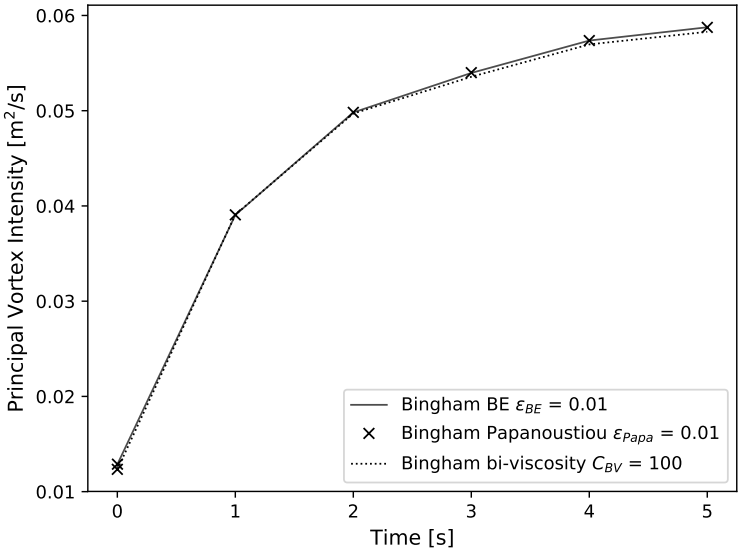


Figure 4.8: Evolution of the intensity of the principal vortex with time, lid driven cavity test. Simulations are run for a Bingham fluid, with yield stress $\tau_S = 10$ Pa, using the BE, Papanoustiou and bi-viscosity regularised Bingham models.

Figure 4.8 shows that, using model constants that yield converged results, the predicted principal vortex intensity is similar for the three regularised models investigated.

The lid driven cavity test case is simulated using the Papanoustiou model and the parameter $\epsilon_{\text{Papa}} = 0.01$. The extrema of the velocity along the the centre lines ($x = 0.5$ and $y = 0.5$) are computed and compared to the results of Vola et al. [73]. Table 4.2 presents the minimum u-velocity along the $x = 0.5$ centre line ($min(u)$), the position where the minimum, $min(u)$, is encountered ($y_{min(u)}$), the minimum and maximum v-velocity along the $y = 0.5$ centre line ($min(v)$ and $max(v)$) and the position where the minimum and maximum, $min(v)$ and $max(v)$, are encountered ($x_{min(v)}$) and ($x_{max(v)}$). The results are given for different values of the yield stress τ_S. Table 4.2 presents the results obtained in the present thesis as well as the results obtained by Vola et al. [73] (values between brackets).

Table 4.2: Results of the simulation of a Bingham fluid in a lid-driven cavity. The position and value of the velocity extrema along the centre lines $x = 0.5$ and $y = 0.5$ are given for different value of the yield stress τ_S.

	Velocity extrema (Literature value from Vola et al. [73])					
	Along x=0.5 [m]		Along y=0.5 [m]			
$\boldsymbol{\tau_S}$	$\boldsymbol{y_{min(u)}}$	**min(u)** $[m/s]$	$\boldsymbol{x_{min(v)}}$	**min(v)** $[m/s]$	$\boldsymbol{x_{max(v)}}$	**max(v)** $[m/s]$
0 Pa	0.16875 (0.175)	-0.36967 (-0.3813)	0.90625 (0.9125)	-0.50177 (-0.5181)	0.15625 (0.1625)	0.35787 (0.3723)
0.1 Pa	0.16875 (0.175)	-0.36724 (-0.3796)	0.90625 (0.9125)	-0.50006 (-0.5144)	0.15625 (0.1625)	0.35520 (0.364)
1 Pa	0.18125 (0.175)	-0.34470 (-0.3438)	0.90625 (0.9)	-0.48274 (-0.482)	0.16875 (0.1625)	0.33083 (0.3286)
5 Pa	0.19375 (0.2625)	-0.23772 (-0.2149)	0.89375 (0.8875)	-0.39922 (-0.3373)	0.19375 (0.225)	0.22500 (0.1786)
10 Pa	0.34375 (0.5)	-0.22706 (-0.1156)	0.88125 (0.825)	-0.33094 (-0.1258)	0.30625 (0.3375)	0.176553 (0.0572)
100 Pa	0.58125 (0.6)	-0.07824 (-0.0762)	0.83125 (0.6125)	-0.05690 (-0.0002)	0.21875 (0.4625)	0.06034 (0.0002)

There is a good correspondence of the minimum u-velocity along the $x = 0.5$ centre line, obtained in the present thesis, compared with the results of Vola et al. [73]. For the minimum and maximum v-velocity along the $y = 0.5$ axis, there is a good match for yield stress values below $\tau_S = 5$ Pa, however, for higher values of the yield stress, significant discrepancies are observed. In particular, for simulations using a yield stress of 100 Pa, the results obtained differ significantly from the literature results. With a yield stress of 100 Pa, Vola et al. [73] report a principal vortex located centrally at the top of the cavity whereas in the present work the principal vortex is located at the top right corner of the cavity. The discrepancies could be explained by the numerical methods adopted. The present work uses regularised models, which can be conveniently implemented in general-purpose CFD codes, to simulate the Bingham fluid behaviour while Vola et al. [73] use variational methods specifically developed to model Bingham fluid flows.

Chapter 5

Spreading of a fluid with temperature dependent properties

Part of this chapter has been submitted to a peer-reviewed scientific conference and presented:

H. Perrier, M. D. Eaton, B G. M. van Wachem, *Numerical Modelling of corium Spreading.* In Proc. 17th International Topical Meeting on Nuclear Thermal Hydraulics (NURETH-17), Xi'an, Shaanxi, China, 2017.

Part of this chapter has been prepared for submission to a peer-reviewed scientific journal:

H. Perrier, F. Denner, M. D. Eaton, B G. M. van Wachem, *Numerical simulation of corium spreading using Volume-of-Fluid methods.* Publication submitted to the Journal Nuclear Engineering and Design, 2018.

This chapter investigates the numerical simulation of nuclear corium spreading, focussing on modelling the two fluid corium/air system. The complex corium spreading flow is simplified by assuming the corium to be a homogeneous non-reactive fluid with a strongly temperature-dependent viscosity simulating corium solidification. A detailed investigation of the numerical methods most suitable to simulate corium spreading flows is conducted, focussing on the study of the numerical errors, such as numerical diffusion, caused by the spatial discretisation of the energy equation. Numerical errors in the prediction of temperature may strongly affect the flow prediction because of the temperature-dependency of the viscosity.

The focus of this chapter is the study of the numerical methods, based on a VOF framework, suitable to simulate spreading flows with strongly temperature-dependent fluid properties, such as corium spreading flows. Particular attention is placed on the schemes used to discretise the energy equation and a strategy is proposed to limit the numerical errors associated with the overall discretisation process.

While numerical diffusion can be limited using suitable discretisation schemes, it is shown that numerical errors in the calculation of the temperature-dependent viscosity arise as a result of the implicit temperature-averaging inherent to the VOF method.

Various viscosity averaging method, presented in section 5.2, and smoothing strategies, presented in section 5.3.4 are investigated to limit the effect of the inaccuracies in viscosity calculation on the flow prediction. An algorithm, presented in section 5.3.4, is proposed to limit the inaccuracies related to the calculation of the viscosity at cells that contain the interface.

Two test cases are used in this chapter to investigate the numerical methods suitable to simulate two-phase flows with temperature-dependent properties. The first test case described in section 5.3 considers two fluids separated by a flat interface, which is advected in a channel at constant velocity in the absence of gravity force. This test case is useful to isolate and investigate in detail the numerical errors occurring in the vicinity of the interface between the two fluids. Different numerical methods for energy equation discretisation, viscosity smoothing and viscosity averaging are scrutinised. In section 5.4 the consequences of the numerical errors discussed in section 5.3 on a spreading flow simulation are investigated. The test case consists

of a hot dense fluid spreading in a domain, filled with a cold and light fluid, under the influence of gravity. This test case is useful to describe how numerical errors in temperature and temperature-dependent viscosity predictions can affect the spreading length.

5.1 Solution procedure

The solution procedure determines the order of the operations performed to compute and advance the solution from discrete time $t = t^n$ to discrete time $t = t^{n+1} = t^n + \Delta t_{\text{Flow}}$. A detailed discussion of solution procedures is presented in chapter 3. Figures 5.1, 5.2 and 5.3 illustrate the three solution procedures, named procedures A, B and C, investigated in this chapter.

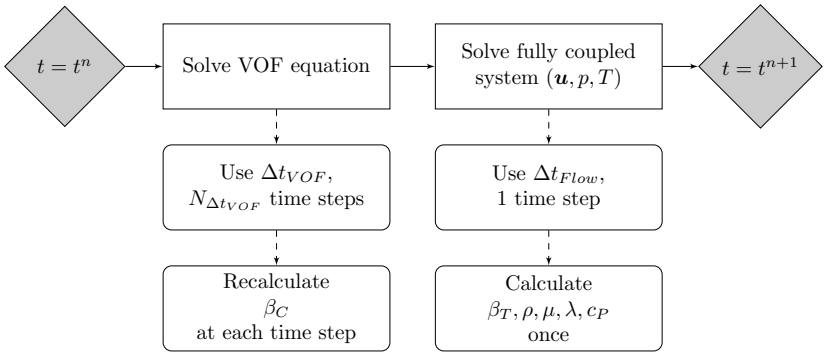


Figure 5.1: Solution procedure A to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where the flow and energy equations are solved in the same coupled linear system of equations, and the VOF advection equation is solved separately.

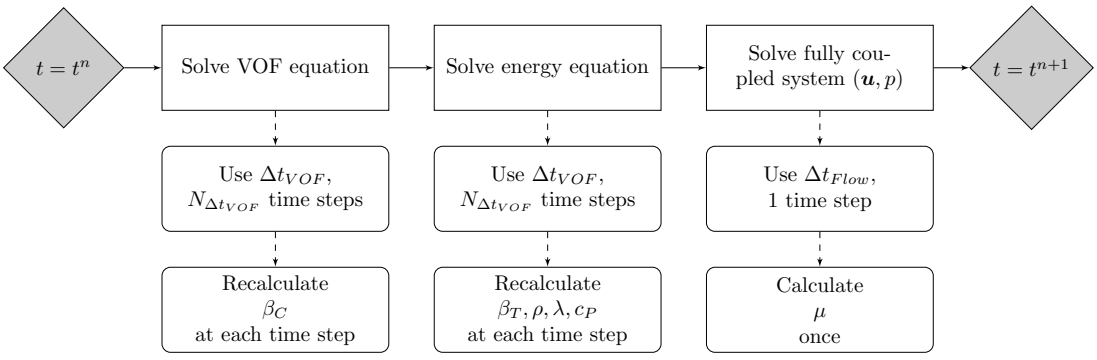


Figure 5.2: Solution procedure B to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where flow, VOF and energy equations are solved in three separate time loops.

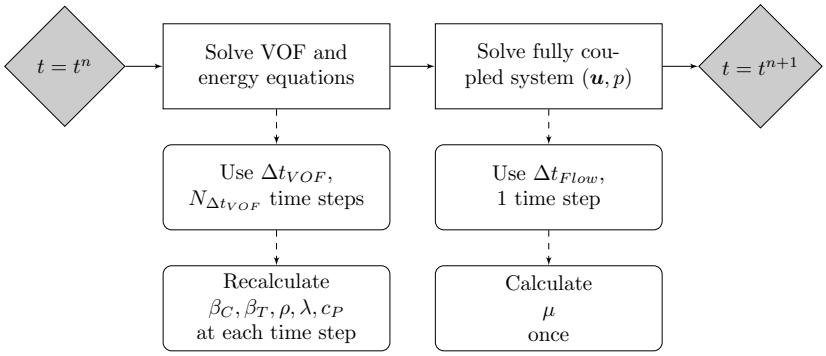


Figure 5.3: Solution procedure C to calculate variables $(\boldsymbol{u}, p, T, C)$ in each time-step Δt_{Flow}, where the VOF and energy equations are solved in the same time loop and the flow equations are solved in a separate time loop.

5.2 Viscosity averaging methods

The study of multiphase flows using numerical methods may involve the discretisation of the space into control volumes (cells) that contain several phases with different fluid properties. With the VOF method, two fluids can be present in a cell and the colour function indicates the volume ratio of each fluid in the cell. In a VOF framework, the discretised momentum equations that need to be solved in every cell involve a unique cell viscosity, that has to be computed from the individual viscosities of the different fluids present in the cell. The methods used to compute the cell viscosity from individual fluid viscosities are later on referred to as viscosity averaging methods. The most commonly used averaging method to compute the cell viscosity from the viscosities of individual fluids is a colour weighted arithmetic average (simply referred to as arithmetic average), given by

$$\mu = C\mu_A + (1 - C)\mu_B \tag{5.1}$$

where μ is the cell viscosity and subscripts A and B relate to the two fluids in the system under consideration. An alternative averaging method is the harmonic average which, compared to an arithmetic average, gives more weight to the fluid with the lowest value of the property of interest. The harmonic average is given by

$$\frac{1}{\mu} = \frac{1-C}{\mu_A} + \frac{C}{\mu_B}. \tag{5.2}$$

The different averaging methods are illustrated in figure 5.4 for two fluids with constant viscosities $\mu = 1$ Pa.s and $\mu = 100$ Pa.s respectively. Colour function value C=0 correspond to a cell full of the fluid with low viscosity ($\mu = 1$ Pa.s) and C=1 corresponds to a cell full of the fluid with high viscosity ($\mu = 100$ Pa.s). Compared to an arithmetic average, the harmonic average gives more weight to the fluid with the lowest value of the property of interest (viscosity), the cell viscosity only increases significantly when the colour function is close to C=1 which mean the cell is almost entirely filled with the fluid with high viscosity.

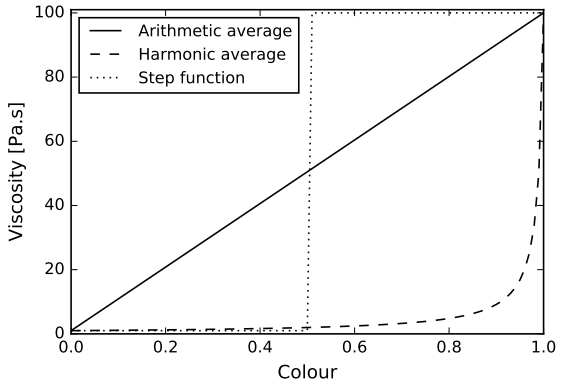


Figure 5.4: Cell viscosity for a cell containing two fluids with viscosities $1Pa.s$ and $100Pa.s$ using different cell viscosity model

It should be noted that, for the sake of clarity, the illustrated situation in figure 5.4 considers fluids with constant individual viscosities. The situation studied in the rest of this chapter is more complex to visualise as one of the fluid has a temperature-dependent viscosity.

In a system with two immiscible fluids separated by an interface, the situation with two fluids in one cell only occurs at cells that contain an interface (later on referred to as interface cells). The total volume occupied by interface cells decreases when the mesh is refined, as illustrated in figure 5.2. In figure 5.2, the volume (surface in two dimensions) occupied by interface cell is hatched.

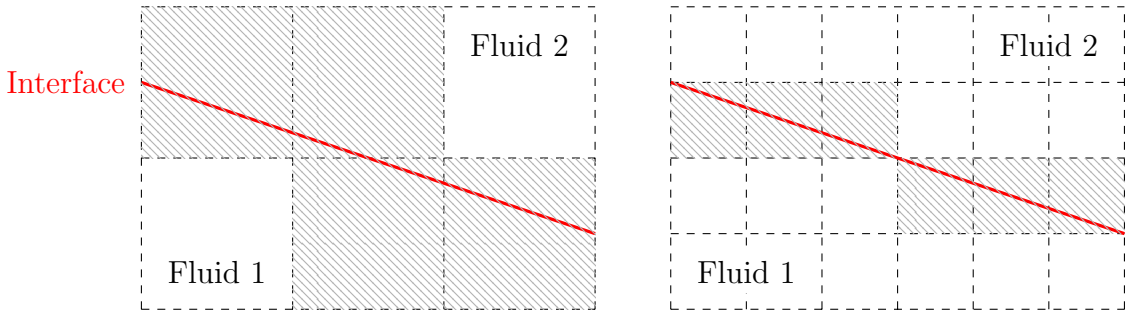


Figure 5.5: Illustration of the surface (2D) or volume (3D) occupied by interface cells (hatched area) on a coarse mesh (left) and a fine mesh (right)

The impact of the viscosity averaging method on flow predictions is expected to be small if the mesh can be made very fine. However, fine grids require a lot of computational power to run simulations, which can be prohibitive. A study of the effect of grid refinement on spreading predictions is presented in section 5.4.2.

5.3 Advection of a hot fluid into a cold channel

The test case investigated in this section is depicted in figure 5.6. It considers a multiphase flow of Newtonian fluids in a rectangular 2D domain of 2.5 m length and 0.5 m height, filled with a hot fluid ($C = 1$) on the left side and a cold fluid ($C = 0$) on the right side. The computational domain is represented by an equidistant Cartesian mesh with 125×25 cells. Hot fluid is introduced from the left boundary of the system, while the cold fluid leaves the system through the right boundary. Free-slip boundary conditions for both fluids are applied at the top and bottom walls to avoid effects of contact line dynamics or wall friction.

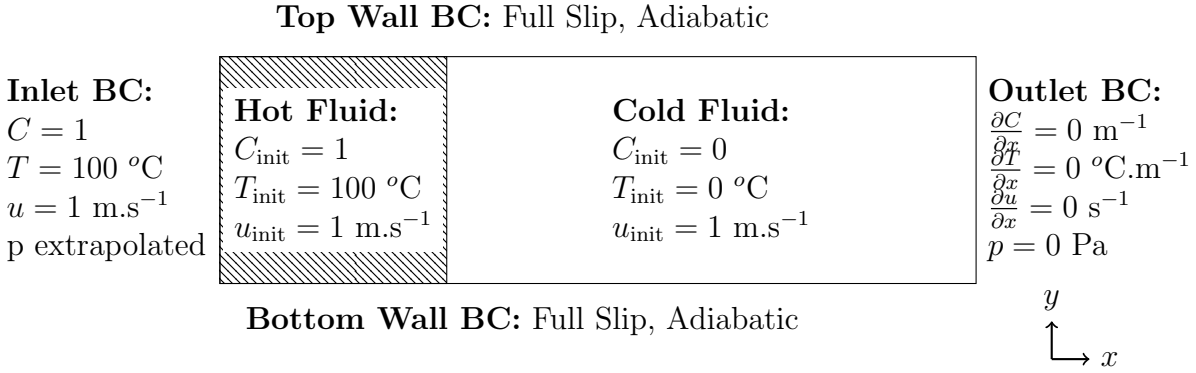


Figure 5.6: Schematic illustration of the advection of a hot fluid in a cold channel, including the initial values (denoted with subscript "init") and the applied boundary conditions (BC). Initially a hot fluid is present on the left hand side of the domain and a cold fluid occupies the right hand side of the domain. Hot fluid is being introduced in the system from the left boundary of the system with velocity 1 $m.s^{-1}$. Gravity is neglected.

Table 5.1: Fluid properties of the two-fluids considered in the test case "advection of a hot fluid in a cold channel"

	Hot Fluid	**Cold Fluid**
Density [kg.m^{-3}]	100.0	1.0
Viscosity [Pa.s]	**T dependent**	1.0
Heat Capacity [J.kg^{-1}.m^{-3}]	1.0	1.0
Thermal Conductivity [W.m^{-1}.K^{-1}]	0.0	0.0

The viscosity of the cold fluid (on the right in figure 5.6) has a constant value of 1 Pa.s. The temperature-dependent viscosity profile used for the hot fluid (on the left in figure 5.6) is shown in figure 5.7 and is given by

$$\mu_{\text{hot}}(T) = -\frac{1}{100}T^2 + \frac{2}{100}T + 100 \text{ (Pa.s)}, \tag{5.3}$$

and is graphically illustrated in Fig. 5.7. The values are chosen so that the viscosity varies as a quadratic function of the temperature and so that the viscosity is 2 Pa.s at 100^{o}C (initial temperature of the hot fluid) and 100 Pa.s at 0^oC. The chosen profile for the temperature-dependent viscosity allows a detailed assessment of the numerical diffusion associated with the applied advection schemes and averaging methods. The thermal conductivity λ of both fluids

is set to zero so that, according to Newton's law of cooling, heat transfer between the fluids should not occur. Any heat transfer observed between the fluids is, therefore, the result of numerical diffusion. The default fluid properties are summarised in table 5.1.

The relationship between temperature and viscosity simulates the solidification process as the viscosity increases when the fluid is cooled down. However, because in theory there is no heat transfer between the fluids (thermal conductivity set to zero), the hot fluid viscosity should remain constant at 2 Pa.s. Any observed increase in viscosity of the hot fluid is, thus, the result of numerical errors in the temperature prediction. It should also be emphasised that due to the relation between hot fluid viscosity and temperature described by equation 5.3, small errors in temperature prediction lead to very large variations in viscosity.

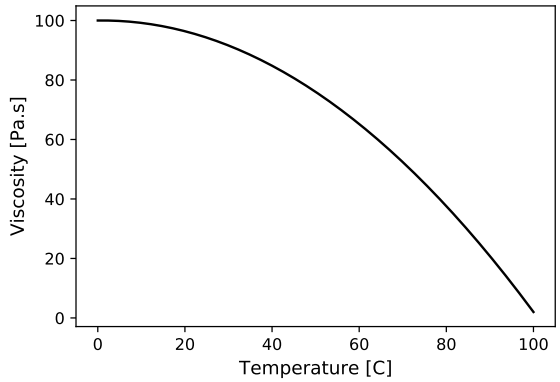


Figure 5.7: Viscosity of the hot fluid, simulated in the test case "advection of a hot fluid in a cold channel", as a function of temperature

Figure 5.8 shows how the profile of colour function C along the x-axis evolves with time, using the CICSAM scheme for the discretisation of the VOF advection equation. The profile of the colour function along the x-axis retains a sharp transition between the hot fluid (C = 1) and the cold fluid (C = 0) at the end of the simulation, demonstrating that the interface sharpness is preserved.

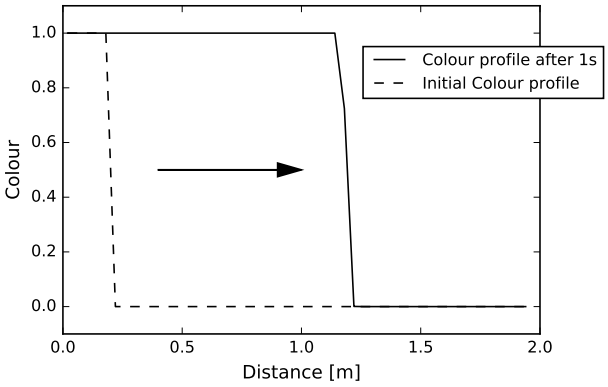


Figure 5.8: Colour function profile along the x-axis corresponding to the test case "advection of a hot fluid in a cold channel at two different times: initial colour function profile (dashed line) and colour function profile after one second of simulation (solid line).

5.3.1 Discretisation and solution of the energy equation

Figure 5.9 presents temperature and viscosity profiles along the horizontal axis after one second of simulation using solution procedure A and with the Upwind, SUPERBEE and Van Leer advection schemes introduced in chapter 3 for the discretisation of the energy equation.

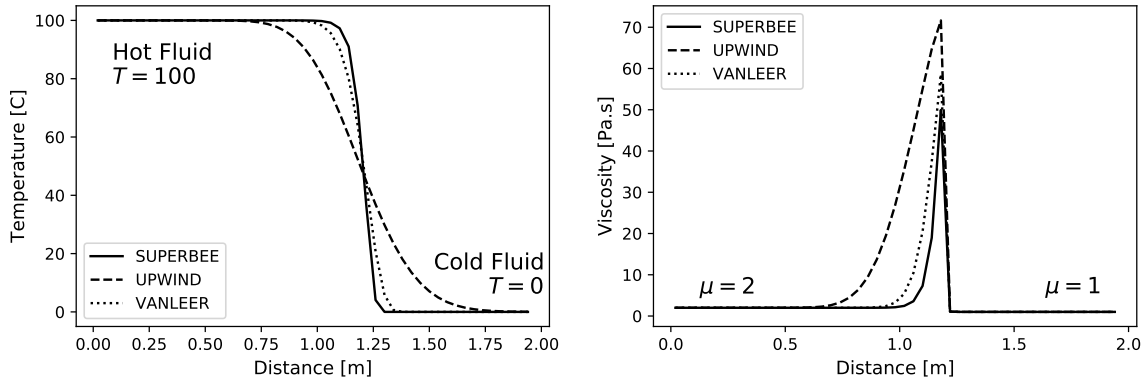


Figure 5.9: Profiles of the cell temperature (left) and cell viscosity (right) along the x-axis after one second of simulation, using different schemes for the discretisation of the energy equation (upwind, van Leer and SUPERBEE).

In figure 5.9 the temperature profile is more diffused than the colour function profile in figure 5.8. there are several mesh cells with temperatures in the range 0^oC – 100^oC in the vicinity of the interface as a result of numerical diffusion of temperature. As expected, the upwind scheme yields the most diffusive temperature profile, whereas the SUPERBEE scheme best preserves the sharp temperature profile at the interface.

The viscosity profiles in figure 5.9 shows a viscosity equal to the hot fluid viscosity ($\mu = 2$ Pa.s) on the left side far from the interface and a viscosity equal to the cold fluid viscosity ($\mu = 1$ Pa.s) on the right side far from the interface. In the vicinity of the interface, however, viscosity is much higher than in either bulk phase. This increase in viscosity occurs in the hot fluid as a result of lower temperatures in the mesh cells containing the interface, leading to an increased temperature-dependent viscosity. Consequently, for a fluid with a strongly temperature-dependent viscosity, numerical diffusion of the temperature field artificially modifies the viscosity close to the interface, which may have a substantial influence on the evolution of the flow.

Considering the significant temperature dependency of the viscosity of the investigated hot fluid, an advection scheme that minimises numerical diffusion is desired. The CICSAM scheme, commonly used for the advection of the colour function in VOF frameworks, is designed specifically to represent sharp discontinuities, such as fluid interfaces.

Using the CICSAM scheme for the discretisation of the energy equation in conjunction with solution procedure A, see figure 5.1, the sharp interface is preserved using the CICSAM scheme and a flow time-step Δt_{Flow} corresponding to a CFL number CFL = 0.025, whereas the simulation diverges if Δt_{Flow} corresponds to CFL = 0.25. This divergence, which severely limits the size of the applied time-step and the efficiency of the solution algorithm, is attributed to the strong coupling between the discretised flow and energy equations, as they are solved simultaneously in a fully-coupled system of linear equations using solution procedure A. Applying solution procedures B (see figure 5.2) or C (see figure 5.3), the flow and energy equations are not solved simultaneously in a fully-coupled system of equations, and thus, a different time-step can be used for the energy equation and for the flow equations. Table 5.2 compares the execution times obtained with the three considered solution procedures; the sharp discontinuity of the temperature profile at the interface is preserved at much lower computational cost using solution procedures B or C than with solution procedure A.

When solution procedure B is adopted, a large time step Δt_{Flow} is used to solve the flow equations and a smaller time step Δt_{VOF} is used to solve the energy and VOF equations. The

Table 5.2: Description of the interface shape and calculation time for simulations using the three considered solution procedures. The flow time-step Δt_{Flow} required to reach convergence is also given.

Solution procedure	$\boldsymbol{\Delta t}_{\text{Flow}}$	$\frac{\boldsymbol{\Delta t}_{\text{Flow}}}{\boldsymbol{\Delta t}_{\text{Energy}}}$	**max T** $[^oC]$	**Calculation time**	**Interface**
A	10^{-3}	1	100	350s	Sharp
B	10^{-2}	10	100.02	90s	Sharp
C	10^{-2}	10	100	90s	Sharp

computation time is reduced compared to solution procedure A, however, temperature overshoots arise close to the interface (temperatures higher than the initial hot fluid temperature) that are not physical. These overshoots are explained by the lag between the solution of the VOF equation and the energy equation. With solution procedure B, the VOF equation is solved first for several time steps Δt_{VOF}), then the energy equation is solved also with time step Δt_{VOF} and finally the flow equations are solved with time step Δt_{Flow}. The density and specific heat capacity values in each mesh cell required to solve the energy equation are calculated based on the available colour function field. However, the solution of the VOF advection equation precedes the solution of the energy equation and, thus, the colour function field available is further advanced in time than the temperature field, leading to the temperature overshoots.

When solution procedure C is adopted, temperature overshoots are prevented and the computation time is kept to a minimum. Indeed, with this solution procedure, the energy equation and VOF equation are solved at the same time, so the colour function field available to update the cell density and specific heat capacity has the same time advancement as the temperature field.

Figure 5.10 shows the profiles of the colour function C and the temperature T obtained using the CICSAM scheme and the SUPERBEE scheme, with Δt VOF corresponding to CFL = 0.025 in conjunction with solution procedure C.

The interface is contained in only one cell ($0 < C < 1$), where both the hot fluid and the cold fluid are present. Similarly, when the CICSAM scheme is used for the discretisation of the energy equation, this interfacial cell is the only one cell with $0^oC < T < 100^oC$. The temperature values upstream and downstream of the interface correspond to the initial temperature of the

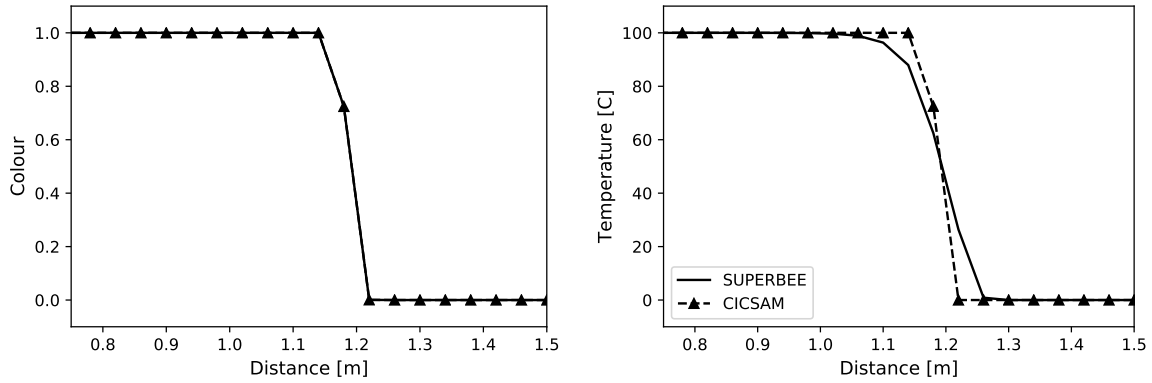


Figure 5.10: Profiles of the colour function (left) and the temperature (right) along the x-axis after one second of simulation using the CICSAM scheme for the discretisation of the energy equation.

bulk phases (100 ^{o}C and 0 ^{o}C, respectively), and no cooling of the bulk phases as a result of numerical diffusion occurs. If instead the SUPERBEE scheme is used for the discretisation of the energy equation, an artificially diffuse temperature profile is obtained at the interface.

Because the CICSAM scheme is able to minimise numerical diffusion of the temperature field at the interface, see figure 5.10, the artificial increase of viscosity at the interface observed in figure 5.9 is also reduced. Figure 5.11 shows the viscosity profile obtained when the CICSAM scheme is used to discretise the energy equation, in comparison with the corresponding viscosity profile obtained using the SUPERBEE scheme. While the CICSAM scheme clearly reduces the unphysical increase in viscosity in the interfacial cells, and limits it to a single mesh cell, it is not eliminated completely. This remaining viscosity peak is, however, not due to numerical diffusion of the temperature field, but is associated with the averaging procedure applied to the viscosity at the interface. Because the temperature-dependent viscosity is based on this cell-averaged temperature, the temperature-dependent hot fluid viscosity is overestimated in the interfacial cell.

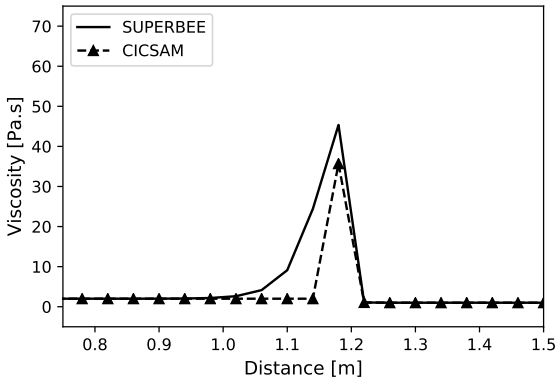


Figure 5.11: Profile of the viscosity along the x-axis after one second of simulation using the CICSAM scheme for the discretisation of the energy equation.

5.3.2 Comparison of analytical and computed viscosity at interface cells

In a VOF framework, only a single value of the temperature is available in each cell. If a cell contains the interface, then, two fluids, with possibly distinct individual temperatures, are present in the cell, so the cell temperature lies between the individual temperatures of each fluid. The temperature in interfacial cells is, thus, lower than the temperature of the hot fluid and, consequently, the viscosity calculated for the hot fluid is overestimated

When thermal diffusion is neglected and the domain boundaries are assumed to be adiabatic, the temperature of each fluid remains equal to its initial temperature, and the temperature in an interfacial cell solely depends on the volume fraction (i.e. the colour function) of each fluid in the cell as well as the initial temperature of each fluid. The analytical cell viscosity can, thus, be computed and compared to the simulation results. In this subsection, the values for the density and specific capacity are set to 1 $\mathrm{kg.m^{-3}}$ and 1 $\mathrm{J.kg^{-1}.K^{-1}}$ for both fluids to simplify the calculations.

The temperature T_P of cell P as a function of the colour function C_P, the physical properties of the fluids and the initial temperatures of the fluids, is given, using equation 5.4, as

$$T_P = \frac{C_P \rho_{\mathrm{Hot}} c_{p,\mathrm{Hot}} T_{\mathrm{Hot}} + (1 - C_P) \rho_{\mathrm{Cold}} c_{p,\mathrm{Cold}} T_{\mathrm{Cold}}}{\rho_P c_{p,P}} \tag{5.4}$$

where the subscripts Cell, Hot and Cold refer to cell values, values for the hot fluid and values for the cold fluid respectively. Combining the arithmetic average given in equation 5.1 with the quadratic function for the temperature-dependent viscosity defined in equation to compute the temperature-dependent viscosity of the hot fluid, the analytical cell viscosity follows from equation 5.4, shown as a function of the colour function in figure 5.12, reaches a maximum of $\mu \approx 39.5$ Pa.s for $C \approx 0.58$.

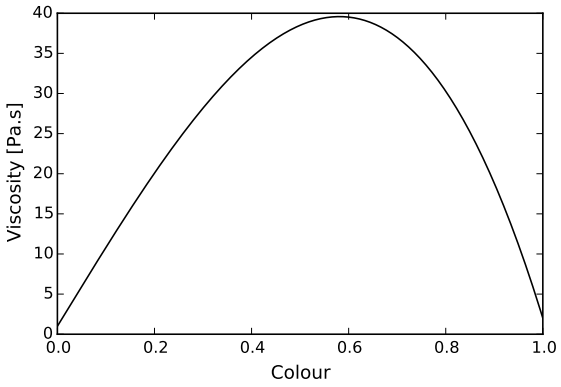


Figure 5.12: Analytical value for the viscosity in the interfacial cell as a function of the cell colour in absence of numerical diffusion.

The advection of a hot fluid in a cold channel test case is a quasi one-dimensional problem, and the interface between the hot fluid and the cold fluid is a straight line moving toward the right, as seen in figure 5.13 where the evolution of the colour function in the interfacial cell is depicted schematically. Interface cell are gradually filled with the hot fluid until the cell is completely filled (i.e. the colour function goes from C = 0 to C = 1). Then the next cell on the right becomes the new interfacial cell and is also gradually filled with the hot fluid. Figure 5.14 shows the colour function value in the interfacial cell obtained from the simulation, where the repeated increase of the colour function in the interfacial cells explains the oscillatory pattern.

Figure 5.15 shows the evolution of the maximum cell viscosity in the system corresponding to the colour function shown in figure 5.14, using the CICSAM scheme and the SUPERBEE scheme for the discretisation of the energy equation. The maximum viscosity in the system corresponds to the viscosity in the interfacial cell. Using the CICSAM scheme, the oscillations of the maximum viscosity are constant in amplitude, ranging from $\mu = 2$ Pa.s to $\mu \approx 40$ Pa.s. The oscillation pattern corresponds to the analytical value of the interfacial cell viscosity

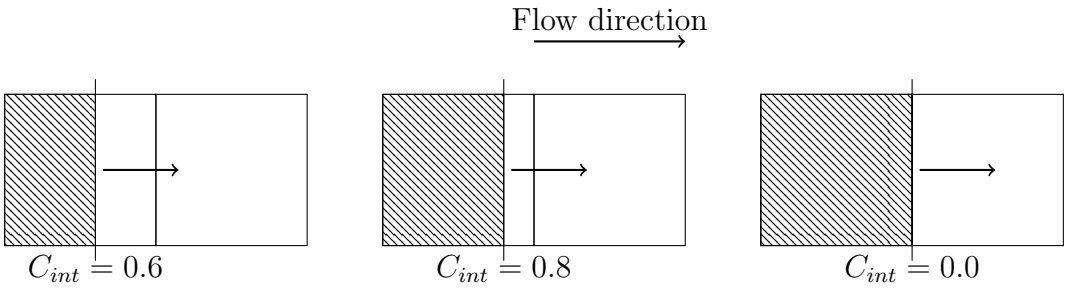


Figure 5.13: Schematic of the evolution of the colour function value in the interfacial cell with time.

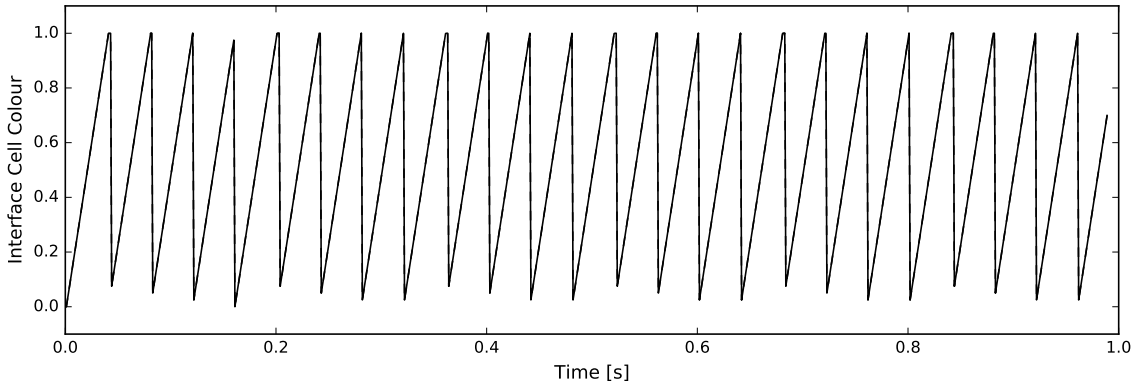


Figure 5.14: Time evolution of the colour function value in the interfacial cell.

given by the colour function, see figure 5.12. This suggests that the peak viscosity observed with the CICSAM scheme for the discretisation of the energy equation is associated with the temperature-averaging process at the interface. When the SUPERBEE scheme is used, the maximum viscosity increases over time and the viscosity does not decrease below $\mu \approx 25$ Pa.s. This behaviour is associated with numerical diffusion that gradually cools down the hot fluid and results in an increase of the temperature-dependent viscosity.

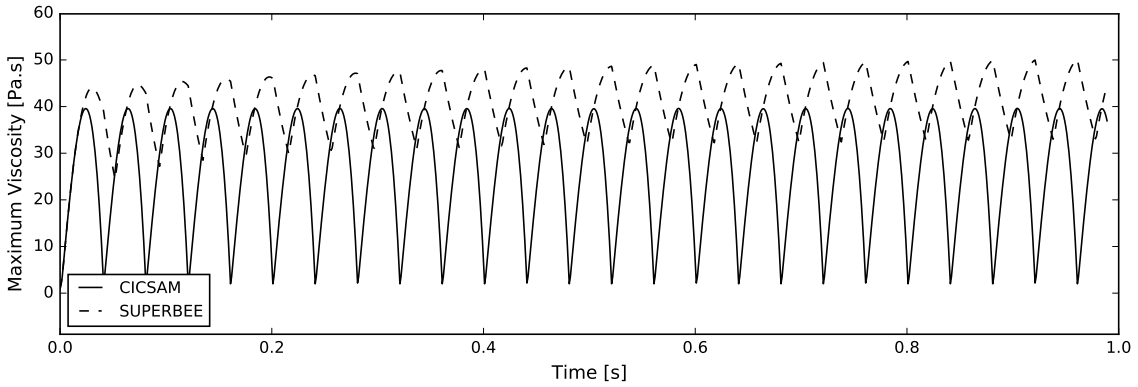


Figure 5.15: Time evolution of the maximum viscosity using different advection schemes for the discretisation of the energy equation.

5.3.3 Viscosity averaging

In the previous section it is shown that the difference between the actual temperature of the hot fluid and the temperature in the interfacial cell is responsible for the peak in viscosity observed in the interfacial cells. This occurs even when numerical diffusion of temperature is prevented by using the CICSAM scheme for the discretisation of the energy equation. Therefore, to limit the maximum viscosity in the interfacial cell, different viscosity averaging methods are compared, using the CICSAM scheme and solution procedure C (see figure 5.3).

Arithmetic averaging, equation 5.1), and harmonic averaging, equation 5.2, are applied to calculate the cell viscosity from individual fluid viscosities. Figure 5.16 presents the time evolution of the maximum viscosity, which corresponds to the viscosity at the interface cell, using both the arithmetic and harmonic averaging. The harmonic average yields a much smaller maximum viscosity in interfacial cells compared to the arithmetic average. The reason for this is that the harmonic average of a small and a large value is closer to the small value, while the arithmetic average gives the middle value of the range bounded by the small and the large values; the harmonic average of the viscosities of the hot fluid and the cold fluid is, thus, closer to the viscosity of the cold fluid.

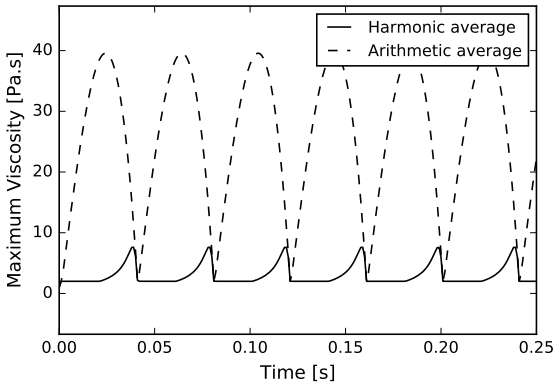


Figure 5.16: Time evolution of the maximum viscosity value using different viscosity averaging methods.

5.3.4 Viscosity smoothing

A common practice used to improve the convergence of numerical solvers, when simulating multiphase flows using interface capturing methods, is convolution (also called smoothing) of fluid properties ([14, 20, 53]). In the context of interfacial flow modelling, convolution consists in applying a transformation to the discontinuous colour function field (the discontinuity lies at the interface) to obtain a smoothed colour field. The smoothed colour function is then commonly used, instead of the original colour function, in equation 5.1 to obtain smooth variation of fluid properties across the interface. Various convolution methods can be found in the literature, see *e.g.* [19, 28, 47], but the present work focuses on the method used in Lafaurie et al. [37] referred to as Laplacian smoothing (or Laplacian filter). The smoothed colour field is obtain as

$$\tilde{C}_P = \frac{\sum_f C_f A_f}{\sum_f A_f}, \tag{5.5}$$

where $\tilde{C}_P$ is the smoothed colour function at cell P, the subscript f relates to the faces of cell P, C_f is the original colour function at cell face f and A_f is the area of face f. It is possible to apply equation 5.5 in an iterative manner to obtain an ever smoother colour function field, but, in the present study, the smoothing operation is applied only once. The different smoothing strategies investigated are summarised in table 5.3.

Smoothing strategy A is the reference situation where no smoothing is applied. With smoothing

strategy B, the smoothed viscosity is defined based on the smoothed colour function

$$\mu_{\tilde{C}_P} = \tilde{C}_P \cdot \mu_{\text{Hot}} + (1 - \tilde{C}_P) \cdot \mu_{\text{Cold}}. \tag{5.6}$$

In equation 5.6, the $\tilde{C}_P$ subscript indicates that the smoothed colour function is used to calculate the viscosity. With smoothing strategy C, the smoothing operation is applied directly to the viscosity field

$$\tilde{\mu}_P = \frac{\sum_f \mu_f A_f}{\sum_f A_f}, \tag{5.7}$$

where $\tilde{\mu}_P$ is the smoothed viscosity at cell P and μ_f is the viscosity at cell face f.

Table 5.3: Considered smoothing strategies for the computation of the viscosity.

Smoothing strategy	**Smoothing A**	**Smoothing B**	**Smoothing C**
Colour smoothing	No smoothing	Equation 5.6	No smoothing
Cell viscosity calculation	Equation 5.1 using C	Equation 5.1 using $\tilde{C}$	Equation 5.1 using C
Viscosity smoothing	No smoothing	No smoothing	Equation 5.7

Figure 5.17 shows the colour function obtained for the advection of a hot fluid in a cold channel, using the CICSAM scheme for the discretisation of the energy equation and different smoothing strategies. A sharp temperature profile for the simulation with no smoothing (smoothing strategy A) is observed; the value of the colour function is $C = 1$ on the left of the interface, $C = 0$ on the right of the interface and $C \approx 0.6$ in the interfacial cell. The colour function profile obtained with smoothing strategy B, however, yields $C < 1$ in the first cell left of the interface and $C > 0$ in the first cell to the right of the interface.

The viscosity profile obtained with the three considered smoothing strategies, alongside the underlying temperature profile, are shown in figure 5.18. With smoothing strategy A, the viscosity in cells that do not contain the interface corresponds to the initial values ($\mu = 2$ Pa.s and $\mu = 1$ Pa.s for the hot fluid and the cold fluid, respectively), but a peak is observed in

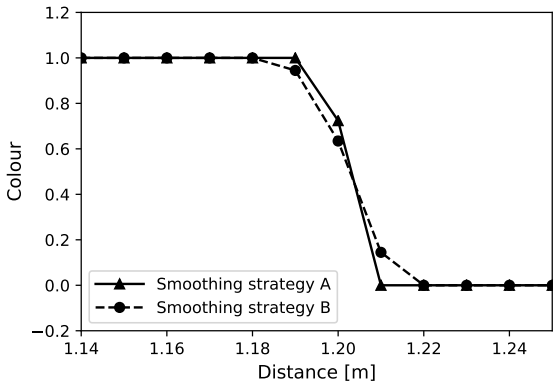


Figure 5.17: Profile of the colour function along the x-axis in the interface region after one second of simulation obtained using different strategies.

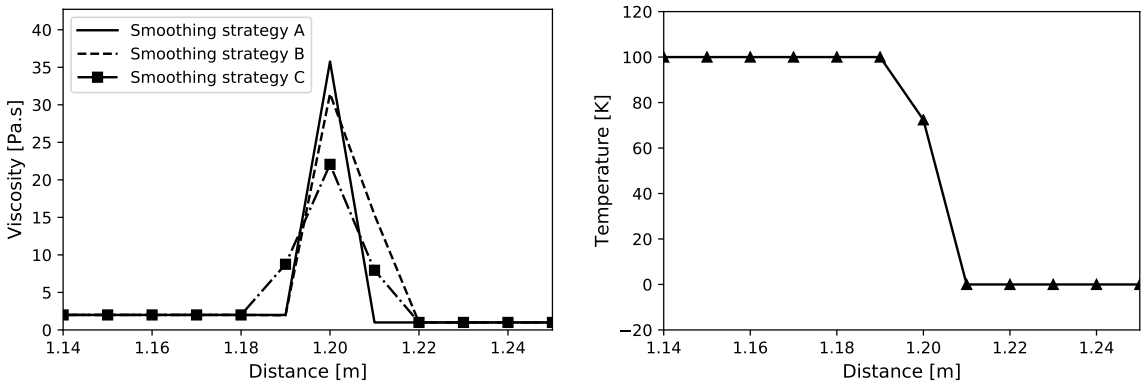


Figure 5.18: Profiles of viscosity (left) and temperature (right) along the x-axis in the interface region after one second of simulation obtained using different smoothing strategies for the viscosity.

the interfacial cell. Smoothing strategy B yields a viscosity peak in the interfacial cell, with an additional increase in viscosity in the first cell to the right of the interface, because the smoothed colour function in this cell is not zero and, according to equation 5.1, the hot fluid viscosity has a contribution to the cell viscosity. Since the temperature in that cell is cold, the viscosity of the hot fluid obtained from equation 5.3 is very high, causing the observed increase in cell viscosity. With smoothing strategy C, the magnitude of the viscosity peak in the interfacial cell is smaller compared to the other smoothing strategies, and the viscosity values in the cells neighbouring the interface are only slightly increased due to the smoothing process. The main difference between smoothing strategies B and C is that with smoothing strategy B, the temperature of the cold fluid may be used to evaluate the temperature-dependent viscosity and, thus, leads to higher viscosity values.

Algorithm to improve the evaluation of the temperature-dependent viscosity

Section 5.3.2 shows that the numerical diffusion of temperature is minimised using the CICSAM scheme for the discretisation of the energy equation, but the evaluation of the temperature-dependent viscosity using the cell-averaged temperature causes a viscosity peak in the interfacial cell. This unphysical increase in viscosity can be remedied by applying an harmonic averaging procedure and a suitable smoothing strategy, as discussed and demonstrated in sections 5.3.3 and 5.3.4, respectively. Based on these findings, the algorithm illustrated in figure 5.19 is proposed to limit the observed unphysical viscosity peak at the interface. With this algorithm, when a cell contains the interface, the temperature of the neighbour cell with the highest value of the colour function C is used to calculate the temperature-dependent viscosity of the hot fluid. If several neighbour cells have the same colour function value, which corresponds to the maximum in the neighbourhood, the average temperature among these cells is used.

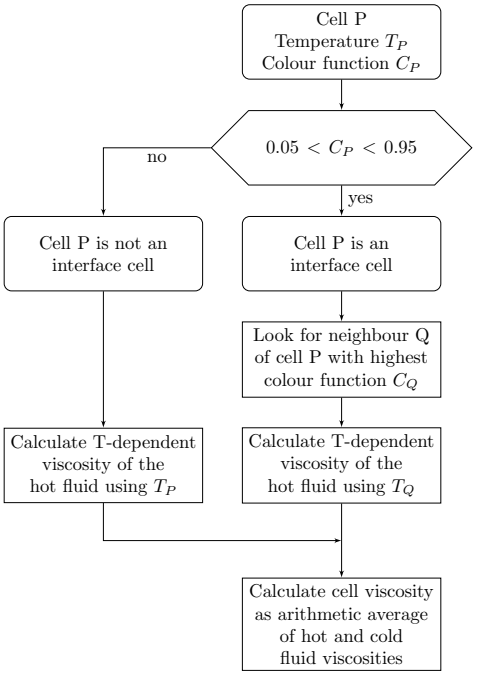


Figure 5.19: Flowchart describing the proposed algorithm to calculate the temperature-dependent viscosity.

Figure 5.20 presents the results for the advection of a hot fluid in a cold channel test case, using the CICSAM scheme for the discretisation of the energy equation and two different strategies to calculate the temperature-dependent viscosity of the hot fluid. The first strategy, represented by the solid line in figure 5.20, calculates the temperature-dependent viscosity in the interfacial

cells using the temperature of the interface cells; the second strategy, represented by the dashed line in figure 5.20, follows the proposed algorithm illustrated in figure 5.19. When the proposed algorithm is used, the viscosity peak at the cell containing the interface is prevented and the viscosity in the interfacial cell is between the bulk viscosity of the hot fluid and of the cold fluid.

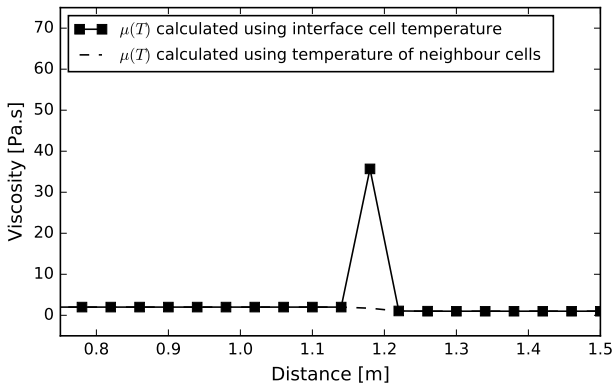


Figure 5.20: Profile of viscosity along the x-axis in the interfacial region after one second of simulation obtained using two methods to calculate the temperature-dependent viscosity of the hot fluid: using the temperature in the interfacial cell (plain line) and calculating the temperature using the algorithm proposed in figure 5.19 (dashed line).

5.4 Spreading test case

The test case discussed in section 5.3 is convenient to study numerical errors affecting the calculation of the cell viscosity in the vicinity of the interface between immiscible fluids. However, it does not describe the effect of erroneous temperature-dependent viscosity calculation on spreading flows. In this section, a more complex spreading test case, depicted schematically in figure 5.22, is studied, where errors in the calculation of viscosity are likely to affect the predicted spreading length.

Initially, a dense hot fluid at rest occupies the lower left side of the rectangular 2D domain, which is 2.5 m in length and 0.5 m in height, and the rest of the domain is filled with a light cold fluid. A free-slip boundary condition is applied at the left and bottom walls, so that the dense fluid can spread freely under the influence of gravity. All domain boundaries are adiabatic. The

domain is represented by an equidistant Cartesian mesh with 500 × 100 cells and the applied flow time-step is $\Delta t_{\text{Flow}} = 5 \times 10^{-3}$ s.

Similarly to the previous test case, the heat conductivity of both fluids is neglected to prevent heat transfer between the fluids. Two viscosity models for the hot fluid, illustrated in figure 5.21, are considered in this section: a linear model

$$\mu_{\text{Linear}}(T) = 100 - 0.98T \text{ (Pa.s)}, \tag{5.8}$$

and a quadratic model

$$\mu_{\text{Quadratic}}(T) = -\frac{1}{100}T^2 + \frac{2}{100}T + 100 \text{ (Pa.s)}. \tag{5.9}$$

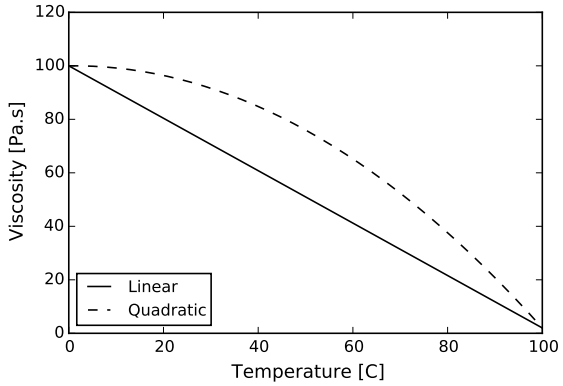


Figure 5.21: Functions describing the temperature-dependent viscosity of the hot fluid in the spreading test case.

The viscosity obtained with these models as a function of temperature is shown figure 5.21. These viscosity models are simplified models simulating the strong temperature dependency of the corium viscosity. Viscosity laws describing the viscosity of semi-solid mixtures, such as solidifying corium, as a function of temperature are available in the literature and the interested reader is referred to the work of Ramacciotti et al. [54]. The viscosity of the cold fluid is $\mu_{\text{Cold}} = 1.0$ Pa.s, and the densities of the hot fluid and the cold fluid are $\rho_{\text{Hot}} = 1000$ kg.m^{-3} and $\rho_{\text{Cold}} = 1000$ kg.m^{-3}, respectively. Both fluids have a specific heat capacity of $c_{p,\text{Hot}} = c_{p,\text{Cold}} = 1.0$ J.kg^{-1}.K^{-1}.

The viscosity of the hot fluid is temperature-dependent, however, since heat transfer between the two fluids should not occur, the temperature of the hot fluid should remain constant. Any changes of the temperature-dependent viscosity is hence not physical but rather reflects numerical errors.

Following the different numerical schemes tested in section 5.3, the effect of numerical errors on the flow is quantified by comparing how far the hot dense fluid spreads when running simulations with different advection schemes applied to discretise the energy equation, as well as different averaging methods for the viscosity. The reference case has a constant viscosity for both fluids, so that the flow is not affected by errors in the calculation of the viscosity of the hot fluid.

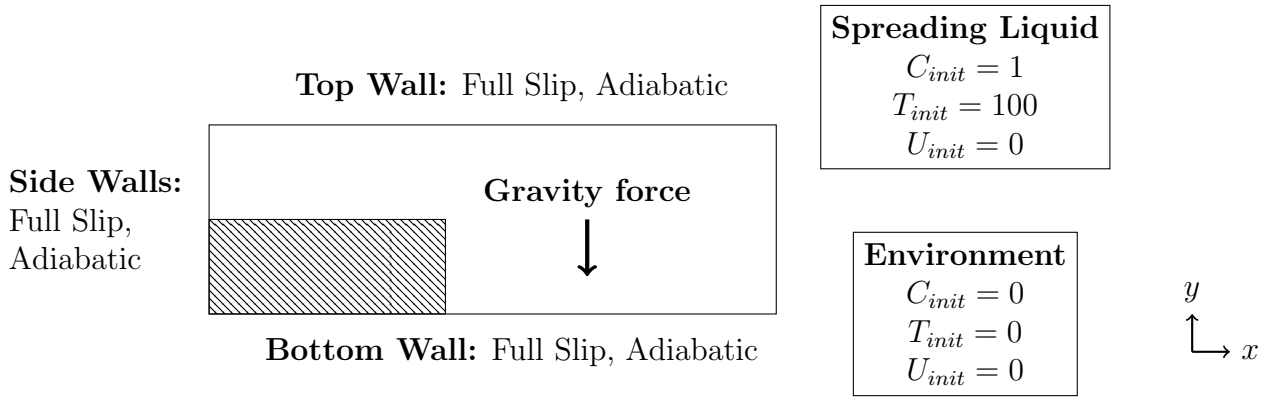


Figure 5.22: Schematic illustration of the spreading of a hot dense fluid in a rectangular domain filled with a cold light fluid. No fluid is injected in the system and the hot dense fluid spreads along the bottom wall under the influence of gravity.

5.4.1 Discretisation of the energy equation

The spreading lengths as a function of time obtained using different advection schemes applied to the energy equation are shown in figure 5.23. The graph on the left presents results using a linearly temperature-dependent viscosity and the graph on the right presents results using a quadratic function for the temperature dependency of the hot fluid. The spreading lengths obtained with both the linear and quadratic function of the temperature-dependent viscosity are shorter compared to the reference simulation with constant viscosity. This difference in spreading length is the result of numerical errors that affect the temperature prediction and,

thus, the temperature-dependent viscosity of the hot fluid, which is more pronounced using the quadratic function, because the temperature dependence of the viscosity is stronger and the errors in the calculation of the hot fluid viscosity are thus bigger. Figure 5.23 also shows that the advection scheme used for the discretisation of the energy equation has a considerable influence on the spreading length. Amongst the advection scheme tested, the upwind scheme always predicts the shortest spreading length and the CICSAM scheme has the lowest impact on the spreading length compared to the reference simulation.

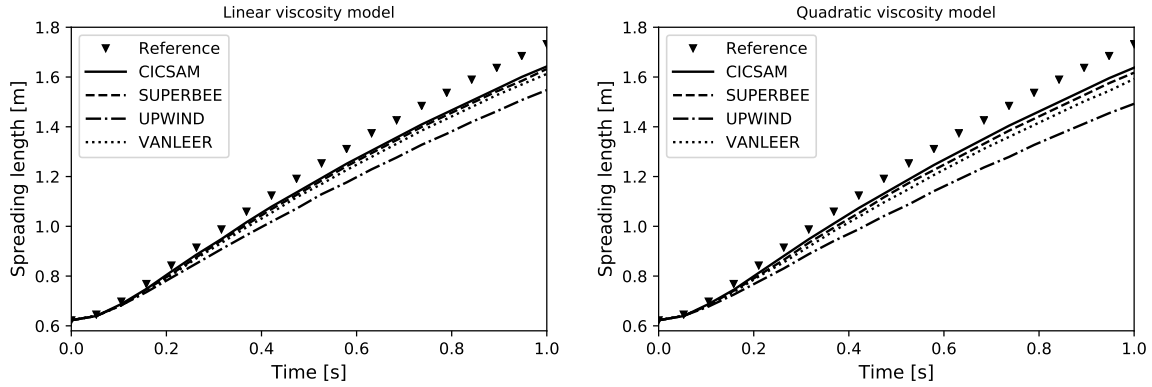


Figure 5.23: Predicted spreading length as a function of time for the spreading test case using different advection schemes for the discretisation of the energy equation, compared against the reference simulation with constant viscosity. Two different functions are used for the temperature-dependency of the hot fluid: the linear function given in equation 5.8 (left) and the quadratic function given in equation 5.9 (right).

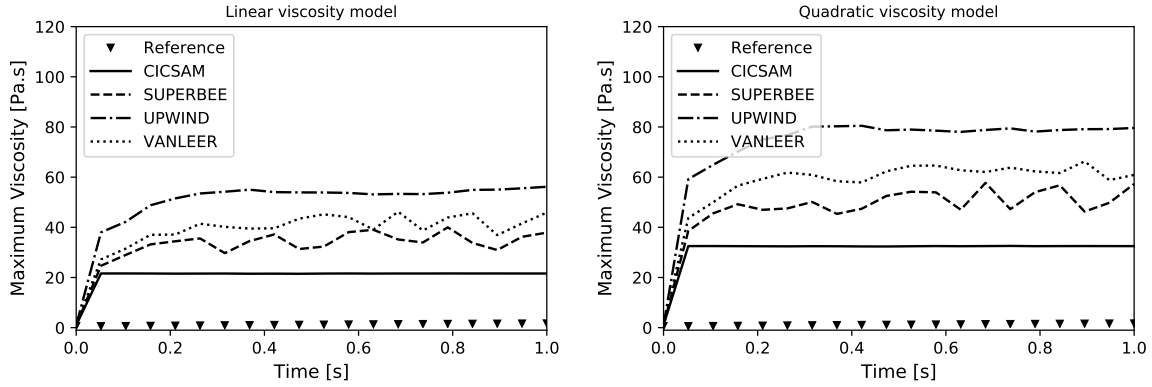


Figure 5.24: Maximum viscosity as a function of time for the spreading test case using different advection schemes for the discretisation of the energy equation, compared against the reference simulation with constant viscosity. Two different functions are used for the temperature-dependency of the hot fluid: the linear function given in equation 5.8 (left) and the quadratic function given in equation 5.9 (right)

Figure 5.24 presents the temporal evolution of the maximum viscosity in the domain for the

linear and quadratic viscosity models using different advection schemes for the discretisation of the energy equation. The results using linear and quadratic functions for the temperature dependency of the hot fluid are qualitatively similar but with different scales; the maximum viscosity values obtain with a linearly temperature-dependent viscosity of the hot fluid are lower than with the quadratic temperature-dependent viscosity. Initially, the maximum viscosity is $\mu = 2$ Pa.s in every configuration, then, when the hot fluid starts to spread, the maximum viscosity remains constant for the reference simulation as there is no temperature dependence of the viscosity, while it increases for all other simulations.

Using the CICSAM scheme, the maximum viscosity first increases for a few time steps and then stays constant. The rise in maximum viscosity is due to the evaluation of the hot fluid viscosity using the temperature in the interfacial cell, which is lower than the hot fluid temperature, as discussed in section 5.3.2, but the oscillatory pattern observed on figure 5.15 does not occur in this case. With the simple test case discussed in section 5.3, the interface is a vertical line moving to the right, which means that the colour function varies from $C = 0$ to $C = 1$ with the regular pattern observed in figure 5.14. However, with the spreading test case, the interface topology is more complex and many different values for the colour function are encountered in interfacial cells at the same time. Among the cells that contain the interface, there is always a cell with a colour function close to the value that corresponds to the maximum viscosity, *e.g.* a maximum viscosity of $\mu \approx 33$ Pa.s is reached when $C \approx 0.5$ for the quadratic viscosity model, which explains why the maximum viscosity remains roughly constant over time.

The results obtained using the standard advection schemes (Upwind, van Leer and SUPERBEE) show a large increase of the maximum viscosity after the first time step and a further increase of maximum viscosity with time for the rest of the simulation. The increased viscosity arises from the two effects highlighted in section 5.3: numerical diffusion and the calculation of the viscosity of the hot fluid viscosity using the cell temperature in interfacial cells. During the course of the simulation, numerical diffusion occurs and the hot fluid is cooled, which explains the continuous increase of the maximum viscosity over time. With the CICSAM scheme the viscosity remains constant after an initial increase.

5.4.2 Viscosity averaging and computation

Following the study of the influence of the advection schemes applied to the energy equation, the effect of the applied viscosity averaging method on the predicted spreading length is investigated. The quadratic model for the temperature-dependent viscosity of the hot fluid, see equation 5.9, is applied. The predicted spreading length obtained using the arithmetic average and harmonic average, in conjunction with the CICSAM scheme for the discretisation of the energy equation and solution procedure C, is shown in figure 5.25.

The results using the harmonic average are closer to the reference results both in terms of spreading length and maximum viscosity. This is consistent with the results obtained for the advection test case in section 5.3.3, as the harmonic average limits the viscosity in interfacial cells and, thus, has a smaller impact on the spreading of the hot fluid.

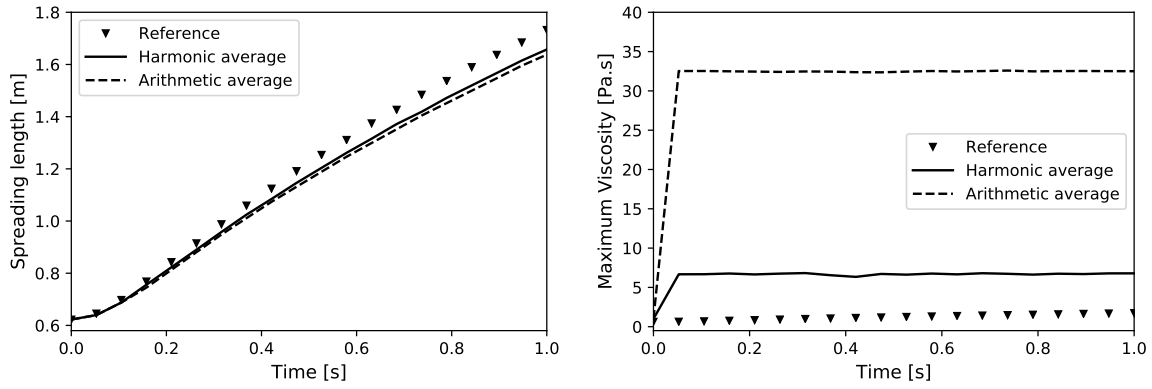


Figure 5.25: Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case using different viscosity averaging methods. The function describing the temperature-dependency of the viscosity of the hot fluid is the quadratic function given in equation 5.9.

The algorithm to calculate the temperature-dependent viscosity of the hot fluid, described in figure 5.19, is applied to the spreading test case. The quadratic model for the temperature-dependent viscosity of the hot fluid, given in equation 5.9, is applied. The CICSAM scheme is used for the discretisation of the energy equation and solution procedure C is adopted. Figure 5.26 presents the results of the reference simulation with constant viscosity of the hot fluid and results using different strategies to calculate the temperature-dependent viscosity of the hot fluid and using an arithmetic averaging to calculate the cell viscosity from the individual

viscosities of the fluids.

The spreading length obtained using the algorithm described in figure 5.19 is closer to the reference simulation compared to results using the original temperature-dependent viscosity calculation method. The proposed algorithm limits the increase in viscosity at interface cells, as seen in figure 5.26, and therefore, the flow is less affected by errors originating from the calculation of the temperature-dependent viscosity.

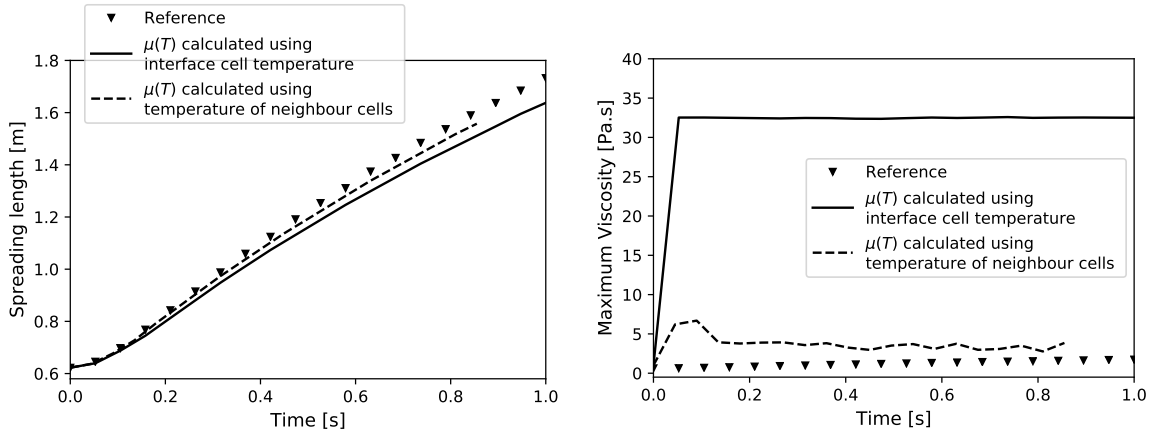


Figure 5.26: Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case using different strategies to calculate the temperature-dependent viscosity of the hot fluid and using an arithmetic averaging to calculate the cell viscosity from the viscosities of the individual fluids.

Mesh resolution

The effect of mesh refinement on the spreading length is investigated for the reference case and using the algorithm proposed in the previous sections, which applies the CICSAM scheme for the discretisation of the energy equation, solution procedure C and the algorithm described in section 5.3.4 to calculate the temperature-dependent viscosity of the hot fluid. Figure 5.27 presents the spreading length on equidistant Cartesian meshes ranging from 125 × 25 to 1000 × 200 cells. The predicted spreading length using the mesh with 125 × 25 is significantly shorter compared to the other simulations. Using a mesh with 250 × 50 cells, or a mesh with higher resolution, yields a similar prediction of the spreading length in the reference case.

Figure 5.27 presents the spreading length predicted for the converged reference simulation (on a

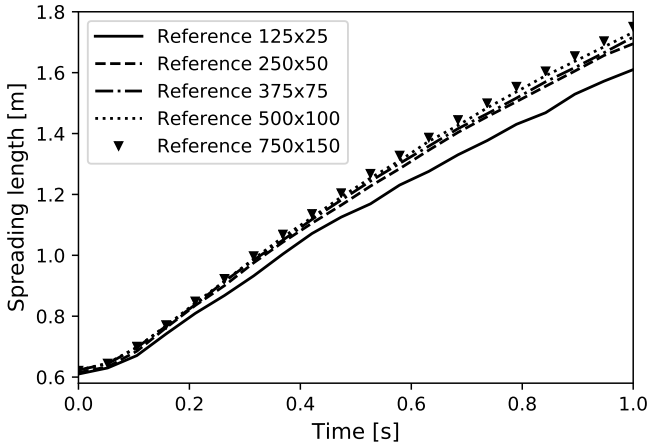


Figure 5.27: Predicted spreading length as a function of time for simulations using the proposed numerical methods with different mesh resolutions.

750 × 150 cells mesh), and, the predicted spreading length, with different mesh refinements, for simulations using the algorithm proposed in section 5.3.4 to evaluate the viscosity in interfacial cells. The results converge to the reference simulation as the mesh is refined.

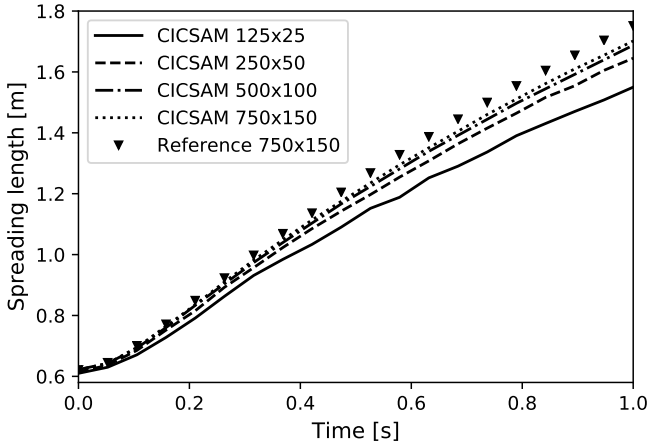


Figure 5.28: Predicted spreading length as a function of time for simulations using the proposed the algorithm proposed in section 5.3.4 with different mesh resolutions.

5.4.3 Spreading with heat transfer between fluids

In the simulations discussed above, the heat transfer between the fluids is prevented by neglecting the thermal conductivity of both fluids ($\lambda = 0$). As result, both fluids should remain at their initial temperature and the viscosity should not change.

However, neglecting the heat transfer between the fluids is not representative of a realistic

situation. To this end, the influence of the advection scheme applied for the discretisation of the energy equation and the applied viscosity averaging method on the spreading of a hot fluid with temperature-dependent viscosity is tested with a finite thermal conductivity of both fluids, with the ensuing conductive heat transfer between the fluids.

The simulations are conducted using the quadratic function for the temperature-dependent viscosity of the hot fluid, equation 5.9, and solution procedure C, see figure 5.3. The thermal conductivity of the hot fluid is $\lambda_{\text{Hot}} = 1$ W.m^{-1}.K^{-1} and of the cold fluid is $\lambda_{\text{Cold}} = 10^{-5}$ W.m^{-1}.K^{-1}. In this situation, the hot fluid is expected to cool down and its temperature-dependent viscosity is expected to increase. Since the viscosity of the hot fluid is expected to change, it is no longer possible to compare these results against a reference simulation with constant viscosity, and the results obtained using different advection schemes and averaging methods can only be compared to each other.

The spreading length and maximum viscosity are shown in figure 5.29 as a function of time, using different advection schemes for the discretisation of the energy equation and different viscosity averaging methods. The spreading length predicted using the upwind scheme is shorter than using the CICSAM scheme for the discretisation of the energy equation, as previously observed in section 5.3, and the maximum viscosity is significantly higher when using the upwind scheme. The viscosity averaging method, however, does not have a significant influence on the results, with the result obtained using the arithmetic average and the harmonic average being almost identical.

As discussed above, when the CICSAM scheme is applied for the discretisation of the energy equation, the peak viscosity in interfacial cells is the result of the difference between the actual temperature of the hot fluid and the available cell temperature, which is an average of the temperatures of the two fluids in the cell. When the thermal conductivity is finite, and because the heat capacity per unit mass is higher for the hot fluid, the cold fluid heats up rapidly, while the temperature of the hot fluid decreases slowly with time. As a consequence, the cell temperature in interfacial cells is close to the hot fluid temperature, as observed in figure 5.30, and the calculation of the temperature-dependent viscosity based on the cell-averaged

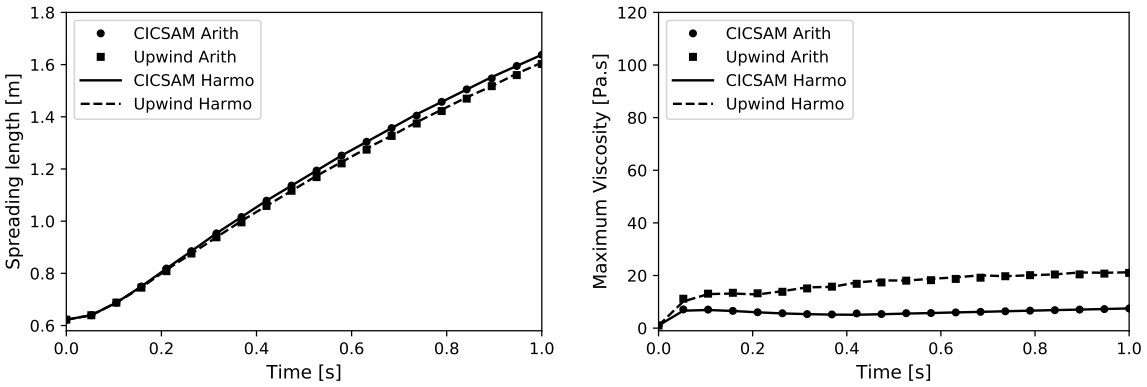


Figure 5.29: Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case with heat diffusion, using the upwind and CICSAM schemes for the discretisation of the energy equation and using the arithmetic ("Arith") and harmonic ("Harmo") averaging for the viscosity.

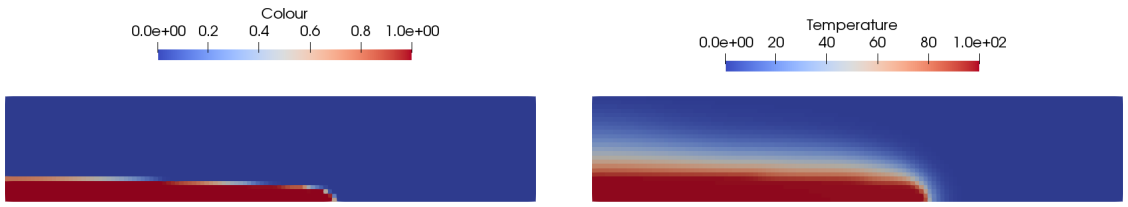


Figure 5.30: Contours of the colour function (left) and the temperature (right) after one second of simulation of the spreading test case with heat diffusion. The CICSAM scheme is used for the discretisation of the energy equation.

temperature does not lead to a significant increase in viscosity. The peak viscosity in this case is around 5 Pa.s compared to 33 Pa.s for the case without heat diffusion. In section 5.4.2 it is observed that for spreading simulations without heat diffusion the averaging method has a significant impact on the spreading length. However, with heat diffusion, the peak viscosity in the interfacial cells is smaller and the impact of the averaging method on the spreading behaviour is smaller.

The spreading length and maximum viscosity as a function of time using different strategies to calculate the temperature-dependent viscosity of the hot fluid are shown in figure 5.31. The results obtained with both methods are similar, with a slightly smaller maximum viscosity and, consequently, larger spreading length using the algorithm proposed in section 5.3.4. Since the thermal conductivity of the considered fluids is non-zero, the temperature in the interfacial cells is close to the hot fluid temperature, and using the interface cell temperature to calculate the

temperature-dependent viscosity does not lead to a significant increase in viscosity, contrary to the cases without thermal conduction.

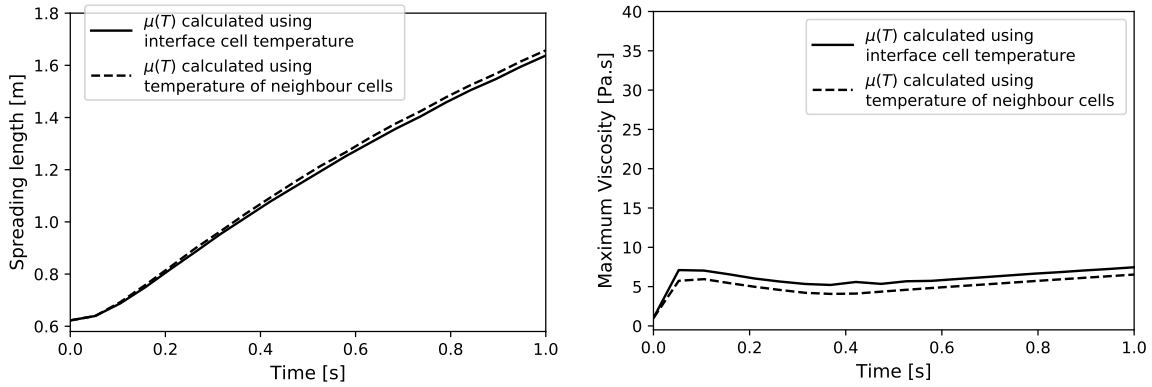


Figure 5.31: Predicted spreading length (left) and maximum viscosity (right) as a function of time for the spreading test case with heat diffusion using different strategies to calculate the temperature-dependent viscosity of the hot fluid.

5.5 Conclusions

In the study presented in chapter 5, the application of different numerical schemes based on the VOF method are applied to simulate the spreading of a simplified model of corium. The numerical models presented consider two immiscible fluids separated by a thin interface, where one fluid is hot and dense with a strongly temperature-dependent viscosity (corium), and the other fluid is light and cold (air). Neglecting thermal conductivity and the associated heat transfer between the fluids, the presented results have demonstrated that standard advection schemes applied to discretise the energy equation, such as the first-order upwind differencing scheme or second-order TVD schemes, generate numerical diffusion of the temperature field in the vicinity of the interface and lead to an erroneous estimation of the temperature-dependent viscosity of the hot fluid. Applying the CICSAM scheme, which was originally developed for the advection of sharp interfaces, to discretise the energy equation has been shown to minimise numerical diffusion of the temperature at the interface. However, when the CICSAM scheme is applied to discretise the energy equation, the results are sensitive to the solution procedure used to solve the discretised governing equations, in particular the VOF advection equation and the energy equation. Different solution procedures have been compared and an optimal

procedure has been identified, in which the momentum and continuity equations are solved simultaneously in a fully-coupled linear system of equations, while the VOF advection equation and the energy equation are solved in separate systems of equations. Furthermore, the VOF advection equation and the energy equation are solved with a smaller time-step than the flow equations, in order to retain a sharp interface.

Even when numerical diffusion is minimised through the application of suitable discretisation schemes, errors in the evaluation of the temperature-dependent viscosity in cells that contain the interface occur, due to the cell temperature averaging inherent to VOF methods. The cell-averaged temperature used to evaluate the temperature-dependent viscosity is colder than the temperature of the fluid representing corium, which leads to an increased value of the calculated viscosity in the interfacial cells. To this end, different methods to limit this unphysical increase in viscosity in interfacial cells have been investigated. Applying a harmonic averaging to calculate the cell viscosity from the individual fluid viscosities, instead of the commonly used arithmetic averaging, as well as smoothing the viscosity field in the vicinity of the interface, have both been shown to limit the unphysical increase in viscosity in interfacial cells. In addition, an algorithm has been proposed to calculate the temperature-dependent viscosity of the hot fluid in interfacial cells, using the temperature of neighbouring cells that contain mostly hot fluid, instead of using the cell averaged temperature at interface cells. This algorithm has been demonstrated to further reduce the unphysical increase in viscosity in interfacial cells.

The work presented has demonstrated that special care is required for an accurate simulation of interfacial flows with strongly temperature-dependent viscosity, such as the spreading of corium in a core catcher. The application of inappropriate numerical schemes leads to an artificial heat diffusion that yields an unphysically large viscosity, in particular in the vicinity of the interface, that can easily be confused with the actual cooling and ensuing viscosity increase (and solidification) of corium. The artificial cooling as a result of numerical diffusion has been shown to affect the predicted spreading length of the hot fluid considerably, which is of particular significance for the design of the safety-critical core catcher, and the proposed numerical methods have been demonstrated to improve the accuracy of the predicted spreading length compared to simulations using standard numerical methods.

Chapter 6

CFD simulation of the Vulcano Corium spreading experiment

Chapter 2 presented a review of the different experimental studies of corium spreading, and, the studies and development presented in the chapters 4 and 5 determined a strategy suited for the simulation of corium spreading flows. In this chapter, this strategy is validated by running simulations of the VULCANO VE-U7 [35] experiment performed in the VULCANO facility at CEA Cadarache [33]. The VULCANO series of experiment are part of an experimental program to test the EPR core-catcher concept and validate its design. The VULCANO VE-U7 experiment investigates spreading on concrete and ceramic substrates, however, it is also used to validate fluid dynamics codes for corium spreading flow simulations [35].

Journeau et al. [35] reported the results of the numerical simulations of the VULCANO VE-U7 experiment using the codes CORFLOW [77, 78], LAVA [2, 58] and THEMA [61]. These codes are described in chapter 2, and, in this section, a summary of the key findings of the numerical simulations of the VULCANO VE-U7 experiment is presented. In the work of Journeau et al. [35], the spreading lengths predicted by these three codes are compared with the experimental results. The authors suggest that the codes provide satisfactory results which justify the use of numerical simulations for larger scale applications, such as simulation of reactor configurations. The LAVA and THEMA codes also calculate the temperature in the substrate and the predicted temperatures are in good agreement with the experiment.

The LAVA and THEMA codes are specifically developed for the simulation of corium flows as opposed to the CORFLOW code or the commercially available code FLUENT, which are suitable for the simulation of a large range of three-dimensional flow conditions. The codes THEMA and LAVA take advantage of a characteristic of corium spreading flows, the vertical length scale is small compared to the horizontal length scale, to use the shallow water assumption and simplify the modelling. With the shallow water assumptions, a fixed vertical velocity profile is assumed and the governing equations are integrated along the vertical axis (the shallow water equations are introduced in chapter 2). Codes using integrated equations have the advantage of numerical simplicity, robustness and low computational cost, and, given the large uncertainties on the physical properties of corium, detailed modelling may not necessarily yield more accurate results [59]. The CORFLOW and FLUENT code use a fine meshing of the corium vertical dimension, which provides more details of the spreading characteristics.

For simulations of the VULCANO VE-U7 experiment using the CORFLOW code, a 15% over-prediction of the spreading length is reported in the work of Journeau et al. [35]. Wittmaack [78] identifies corium dynamic viscosity and inlet conditions for spreading as major sources of uncertainty of spreading length predictions.

Several correlations linking corium temperature and viscosity have been investigated numerically by Spindler and Veteau [61], using the THEMA code, and after validation against experiments using prototypical corium, including the VULCANO VE-U7 experiment, the Ramacciotti correlation [54] is shown to yield better agreement with the experiments. Simulations using the Newtonian and the Bingham fluid models implemented in the THEMA code have been compared to experiments and the auhtors did not report a significant difference between the Bingham and the Newtonian models. Therefore, given that the Bingham model requires a yield stress parameter which is uncertain, Spindler and Veteau [61] recommends the use of a Newtonian model for corium spreading simulations. The assessment of the THEMA code against several spreading experiments [60] indicates that spreading flow simulations are significantly affected by the inlet temperature of the spreading fluid and the prescribed corium viscosity laws.

Ye et al. [79] conducted numerical simulations of the VULCANO VE-U7 experiment using the CFD code Fluent [7]. The simulations consider two immiscible fluids: air and corium, and a Volume-of-Fluid (VOF) method with a geometric interface reconstruction scheme is used to treat the multiphase aspect. A radiative heat transfer model using the discrete ordinate method is adopted. The temperature-dependency of the corium properties is specified through user defined functions. The authors conducted a numerical study of the effect of corium properties on corium spreading. It is reported that heat generation by decay heat in the corium phase does not affect spreading significantly, whereas, the temperature-dependent viscosity of corium and the inlet temperature of corium have a significant effect on spreading predictions. The authors suggest that further development is required regarding the modelling of crust formation.

The work presented in this chapter aims to use the recent progress in two-phase flow modelling as well as the results of the studies conducted in this thesis (see chapter 5) to simulate the

VULCANO VE-U7 experiment. In chapter 5, a detailed investigation of the numerical errors that affect corium spreading simulations has been conducted and a set of numerical methods best suited for such simulations has been proposed. In particular, it is shown that, for the simulation of spreading flows using a Volume-of-Fluid (VOF) methodology to capture the position of the interface between spreading liquid and surrounding gas, numerical diffusion of temperature significantly affect spreading predictions due to the temperature-dependency of corium viscosity. The use of the CICSAM scheme [68], which is specifically designed to represent sharp discontinuities, such as fluid interfaces, is thus recommended for the discretisation of the energy equation to minimise numerical diffusion of temperature.

Compared to previous numerical investigation of the VULCANO VE-U7 experiment, the simulations presented in this section are run on much finer numerical meshes. As a matter of comparison, Ye et al. [79] report a mesh size of 100 000 cells, while, in the work presented in this chapter, a minimum mesh size of 600 000 cells is shown to yield a converged value of the corium spreading length.

The experimental set-up of the VULCANO VE-U7 experiment is described in section 6.1. Section 6.2 describes the modelling strategy and simplifications used to simulate the VULCANO VE-U7 experiment. The numerical methods chosen for the simulations are presented in section 6.3 and section 6.4 describes the initial and boundary conditions of the simulations.

Section 6.5 presents the result of the different simulations performed. The code performance are first validated in section 6.5.1 on a simple Couette flow problem, for which analytical solutions are available. Then, section 6.5.2 presents sensibility studies performed on simulations of the VULCANO VE-U7 experiment. In particular, a grid refinement study is performed, simulation results using different advection schemes for the energy equation discretisation are compared and simulations using different specific heat capacity for the corium are presented. Finally the simulation results are compared with experimental data in section 6.5.3.

6.1 Description of the VULCANO VE-U7 experiment

In the VULCANO VE-U7 experiment [35], a corium mixture composed of UO2 (61 wt.%), ZrO2 (30 wt.%) and other oxides is heated up using plasma torches and poured into a retention cup (illustrated in figure 6.1). As corium is poured in the retention cup, it starts spreading towards two spreading channels made of zirconia bricks and covered with concrete and ceramic respectively. The spreading area is mounted on a weighting scale so that, the mass flow rate of corium entering the channels is measured throughout the experiment. The data collected during the experiment include temperatures at different depth in the substrates, corium mass in the channels, post-mortem height of the spread and corium front position (also referred to as spreading length).

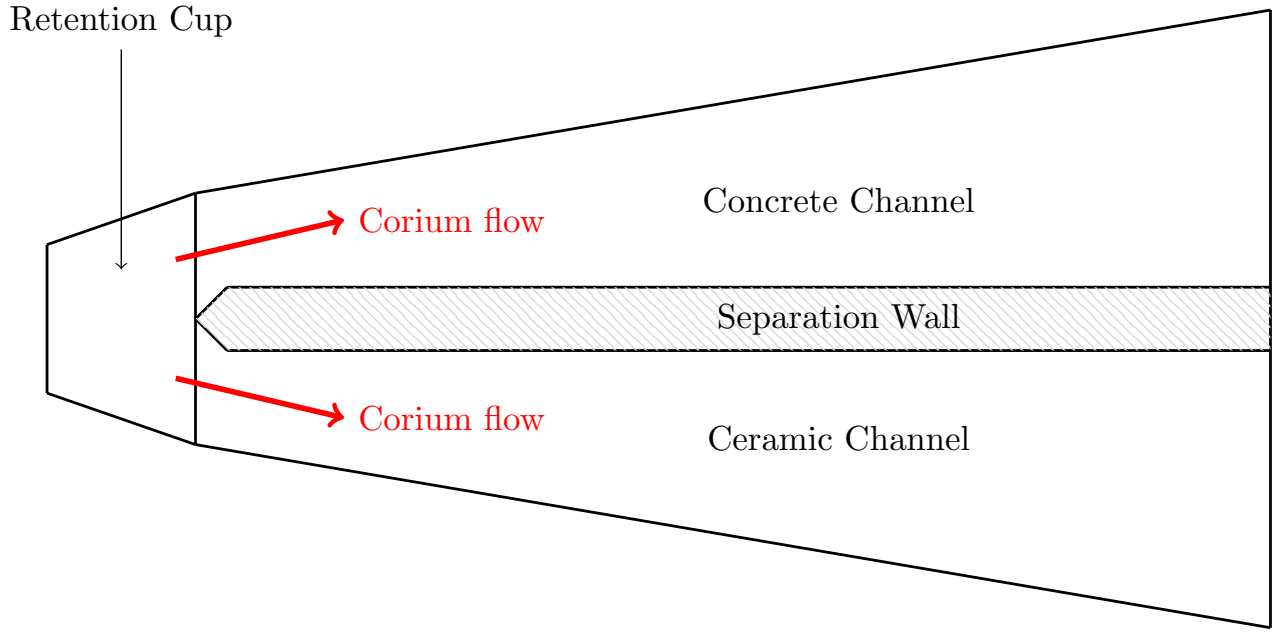


Figure 6.1: Geometry of the spreading area of the VULCANO VE-U7 corium spreading experiment. During the experiment, corium is heated-up, poured into the retention cup and it spreads toward the two spreading channels which are made of zirconia bricks and covered with concrete and ceramic respectively.

In the experiment [33], a total corium mass of approximately 40 kg entered the spreading area and it took between 7 seconds (concrete channel) and 12 seconds (ceramic channel) for the corium to reach immobilisation. On the concrete channel 12.6 kg of corium flowed over a distance of about 350 mm and on the ceramic channel 14 kg of corium flowed over a distance of about 470 mm. The height of the corium spread after immobilisation was about 65 mm.

6.2 Modelling of corium spreading

Numerical simulation is a cost-efficient way to study corium spreading compared to experiments, however, some extreme characteristics of corium spreading flows make the simulations very challenging using classical Computational Fluid Dynamics (CFD). In this section, models and simplifications used to limit the complexity of the CFD simulations are discussed.

The VULCANO VE-U7 experiment investigates corium spreading associated with the core-catcher concept of the EPR. With the EPR core-catcher, corium falling from the pressure vessel is first retained in a cavity covered with sacrificial concrete and then spread over a dedicated spreading area [27]. During the retention phase, corium incorporates concrete, which promotes spreading and ensures a relatively "standardised" composition of corium [42]. A corium representative of what could be expected during an accident in the EPR reactor is used for the VULCANO VE-U7 experiment and Journeau et al. [35] suggest that a constant corium density and a homogeneous composition can be assumed.

Corium rheology is complex and Ramacciotti et al. [54] suggest that corium behaves as a Newtonian fluid above liquidus temperature and as a Bingham fluid below liquidus temperature. Spindler and Veteau [61] conducted numerical simulations of solidifying corium spreading using both a Newtonian fluid model and a Bingham fluid model. The results obtained using the two models are similar and, given the uncertainty on the corium yield stress, which is a parameter of the Bingham fluid model, the use of a Newtonian model for corium spreading simulations is recommended. In the present contribution, corium is assumed to be a non-reactive Newtonian fluid and the spreading flow is assumed incompressible.

With the compressive VOF method used in the present work to discretise the corium/gas interface (see chapter 3), an explicit representation of the corium free surface, which is required to calculate heat loss by radiation, is not available. Heat transfer through radiation is, thus, not considered in the present work. This simplification may result in an under-prediction of corium cooling at the free-surface, and, an over-prediction of corium spreading.

In the work of Farmer et al. [24], corium spreading is investigated, taking into account corium

oxidation and concrete ablation. It is reported that the time scale of spreading is shorter than the time scale of corium oxidation and concrete ablation, therefore, the author suggest that concrete ablation and corium oxidation can be neglected for corium spreading studies. In the present contribution, the substrate material is not included in the simulated system, and, its effect on spreading is accounted for by prescribing boundary conditions. Phenomena such as ablation, fracture of the substrate and gas release from the substrate decomposition into the corium are not considered.

The physical properties of corium used in the present work correspond to the values presented in Journeau et al. [35]. The relevant properties are discussed in this section and the interested reader is referred to Journeau et al. [33] for a detailed description of the methodology used to obtain the corium properties.

The solidification process is captured using an apparent viscosity model. With the apparent viscosity model, the solidifying fluid is characterised by a solid fraction, and modelled as a liquid with a viscosity that increases with the solid fraction. Ramacciotti et al. [54] proposed a methodology to calculate the viscosity of semi-solid mixtures as a function of the liquid viscosity and the solid fraction of the mixture, given as

$$\mu_{\text{Mixture}} = \mu_{\text{liq}} e^{2.5 C_\mu \phi}, \tag{6.1}$$

where μ_{liq} is the liquid viscosity, ϕ is the volume fraction of the solid phase and C_μ is an empirical coefficient between 4 and 8. Journeau et al. [35] suggests that this viscosity law with a parameter C_μ between 6 and 6.3 provides a satisfactory estimate of the effective viscosity of solidifying corium to simulate corium spreading flows. In Journeau et al. [35], it is suggested that, given the large uncertainty around the viscosity estimates, the use of a simplified version of the viscosity law, where the corium viscosity is expressed directly as an exponential function of the corium temperature, is acceptable. The corium viscosity is then given as

$$\mu_{\text{Corium}} = A e^{-B(T - T_{\text{Liquidus}})}, \tag{6.2}$$

where T_{Liquidus} is the liquidus temperature of corium and the model coefficients A and B have values $A = 0.0206$ Pa.s and $B = 0.0287$ K^{-1}. Using an apparent viscosity to model the solidification of corium, there is no explicit representation of the corium crust, which is reported to form at the interface with the substrate and at the corium free surface [57]. The effect of crust mechanics and crust fracture are, thus, not considered in the work presented in this chapter.

In Journeau et al. [35], the estimated corium inlet temperature and corium liquidus temperature (T_{Liquidus}) are respectively 2450±80 K and 2623±50 K, which correspond to, using equation 6.2, a corium initial viscosity of 2.95 Pa.s. In Journeau et al. [35], it is reported that the CORFLOW simulations used a slightly modified version of the viscosity law described by equation 6.2 to have a vertical asymptote at 1750 K and a corium inlet temperature of 2438 K. The reported initial corium viscosity used for the CORFLOW simulations is 35 Pa.s. For the LAVA simulations, a corium inlet temperature of 2450 K and a corium initial viscosity of 22 Pa.s are reported.

In the present work, a corium inlet temperature of 2450 K is assumed for all the simulations, but three different values of T_{Liquidus} are considered to account for the uncertainty of the initial corium viscosity. The corium liquidus temperatures considered in the present work are $T_{\text{Liquidus}} = 2623$ K, $T_{\text{Liquidus}} = 2692$ K and $T_{\text{Liquidus}} = 2706$ K, corresponding to, using equation 6.2, initial corium viscosity values of 2.9 Pa.s, 22 Pa.s and 35 Pa.s respectively.

Given the exponential relation between corium viscosity and temperature, described in equation 6.2, very high viscosity values can be reached, which affect adversely the convergence of the simulations. Therefore, viscosity values are limited to $\mu_{\text{Corium}} = 1000$ Pa.s in the simulations.

A common practice used to improve the convergence of numerical solvers, when simulating multiphase flows using interface capturing methods, is convolution (also called smoothing) of fluid properties ([14] [20] [53]). In the present work, the smoothing method used in Lafaurie et al. [37], referred to as Laplacian smoothing, is applied to smooth the viscosity field. Following the suggestion of chapter 5 for simulations involving fluids with temperature-dependent properties, the smoothing operation is applied directly to the viscosity field.

In Journeau et al. [35], the fluid properties of corium are given with an uncertainty range. In

the present work, simulations are run using the reference values as well as the lower and upper bound of the uncertainty range to study the influence of the uncertainty on the spreading predictions. Table 6.1, summarises the reference corium properties used to run simulations, including their uncertainty range, where available.

Table 6.1: Reference properties of corium and properties of Nitrogen at room temperature

	Corium	**Gas**
Density [kg.m^{-3}]	5100.0 ± 300	1.251
Viscosity [Pa.s]	T dependent	$1.76 \cdot 10^{-5}$
Specific Heat Capacity [J.kg^{-1}.K^{-1}]	500.0	1040
Thermal conductivity [W.(m^{-1}.K^{-1}]	3.0 ± 0.5	$25.83 \cdot 10^{-3}$

The specific heat capacity of corium depends on the corium temperature, however, in the present work, a constant value is assumed for the specific heat capacity of corium corresponding to the specific heat capacity of solidified corium (500 J.kg^{-1}.K^{-1}). The effect of this simplification on the spreading calculation is investigated by running simulations with different specific heat capacity values.

The system considered for the CFD simulations is a two-phase flow of immiscible fluids (corium and gas). The corium phase has temperature-dependent viscosity and spreads under the effect of gravity. The gas properties, given in table 6.1, correspond to the properties of Nitrogen at room temperature. The flow of immiscible fluids can be affected by surface tension. The capillary number, which represents the relative effect of viscous force relative to surface tension force, is used to determine if surface tension significantly affects the flow of interest. It is defined as

$$Ca = \frac{\mu \cdot u}{\sigma}, \tag{6.3}$$

where μ is the dynamic viscosity, u is the flow characteristic velocity and σ is the surface tension coefficient. For corium spreading flows the capillary number is much greater than one due to the high viscosity, which means that the surface tension force is small compared to viscous

forces and it is thus neglected.

6.3 Numerical methods

The governing equations of two-phase flows and the numerical methods used in this thesis are described in details in chapter 3. In this section, the numerical methods used for corium simulations are briefly discussed.

The finite volume numerical framework used in this thesis is based on the framework developed by Denner and van Wachem [21]. A linear system with momentum and continuity equations is constructed and solved in a fully coupled manner. The compressive VOF method, developed by Denner and van Wachem [22], is used to capture the evolution of the interface between corium and gas with time. Based on the results of chapter 5, to minimise the numerical diffusion of temperature in the vicinity of the interface, the energy and VOF equations are solved separately from the momentum/continuity system of equation. The VOF and energy equations are solved in the same time loop, using a time step smaller than the time step used to solve the flow equations (momentum and continuity equations). Figure 6.2 illustrates the procedure used in this chapter to solve the governing equations.

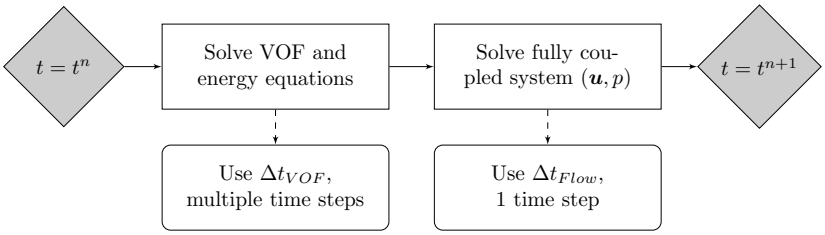


Figure 6.2: Solution procedure to calculate variables $(\boldsymbol{u}, p, T, C)$ at time $t = t^{n+1}$ from time $t = t^n$ using an approach where the VOF and energy equation are solved in the same time loop and the flow equations are solved in a separate time loop.

Following the compressive VOF method developed by Denner and van Wachem [22] to model the dynamics of the interface, the Crank Nicolson scheme is used for the time discretisation of the VOF equation and the CICSAM scheme [68] is used to discretise the advection terms in the VOF equations. To maintain interface sharpness, following the recommendations of Gopala

and van Wachem [29], a time step corresponding to a CFL number lower than 0.01 is used to solve the VOF equation.

As discussed in chapter 5, the simulation of the flow of two immiscible fluids with temperature-dependent fluid properties may be affected by numerical diffusion of temperature in the vicinity of the interface. Based on the findings of chapter 5, the CICSAM scheme, which is designed for the discretisation of sharp interfaces, is applied for the discretisation of the advection term in the energy equation. The Crank Nicolson scheme is used for the time discretisation of the energy equation and it is solved using a time step corresponding to a CFL number lower than 0.01.

In the work presented in this chapter, the Backward Euler scheme is applied for the time discretisation of the continuity and momentum equations and the upwind differencing scheme is applied for the discretisation of the advection terms in the continuity and momentum equations. A time step corresponding to a CFL number lower than 1 is used to solve the momentum and continuity equations.

6.3.1 Simulation domain

Compared to the experimental set-up, illustrated in figure 6.1, the geometry used for the simulations does not include the retention cup and only one channel of the spreading area is considered (see figure 6.3). The channel considered has a width of 95 mm at the inlet and 206 mm at the outlet. The length of the channel is 650 mm and its height is 100 mm. Corium is injected into the channel through the entire width of the inlet and up to a height of 65 mm corresponding the post-mortem height of the spread.

A mesh, illustrated in figures 6.4 and 6.5, with 600 000 cells is chosen as a reference and simulations, with different mesh refinements, are compared to ensure that the results are not sensitive to the mesh size.

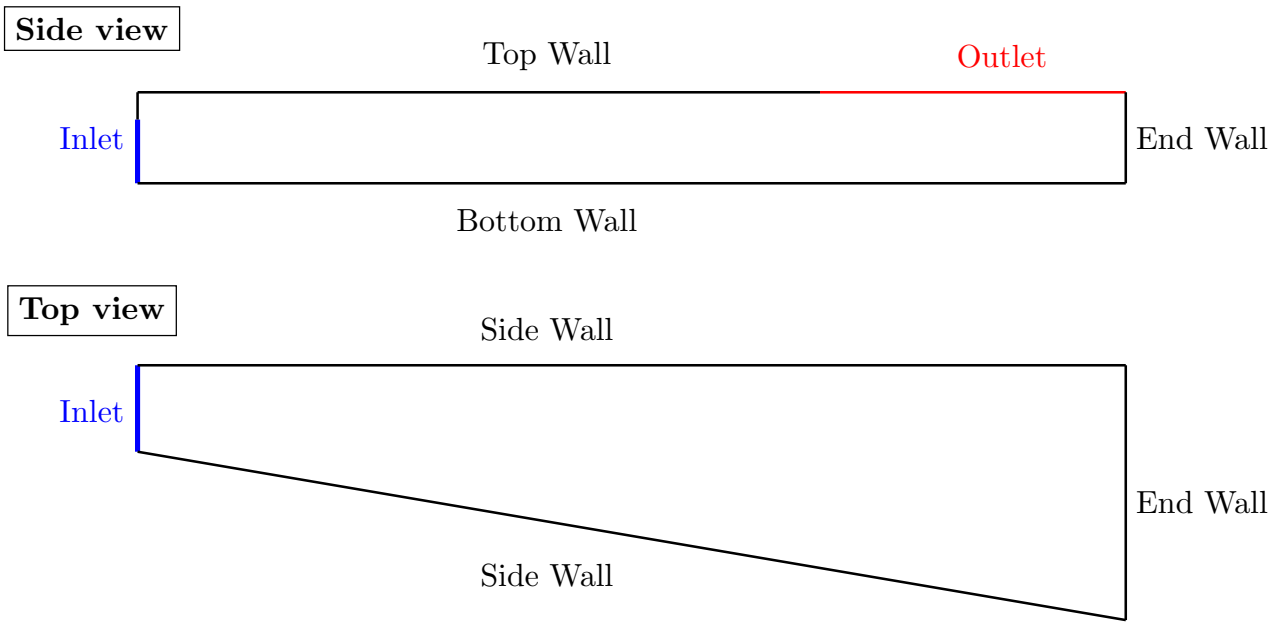


Figure 6.3: Geometry used to run numerical simulations of the VULCANO VE-U7 corium spreading experiment. Compared to the experiment, the retention cup, where corium is poured, is not represented in the numerical model and only one of the two spreading channels is considered.

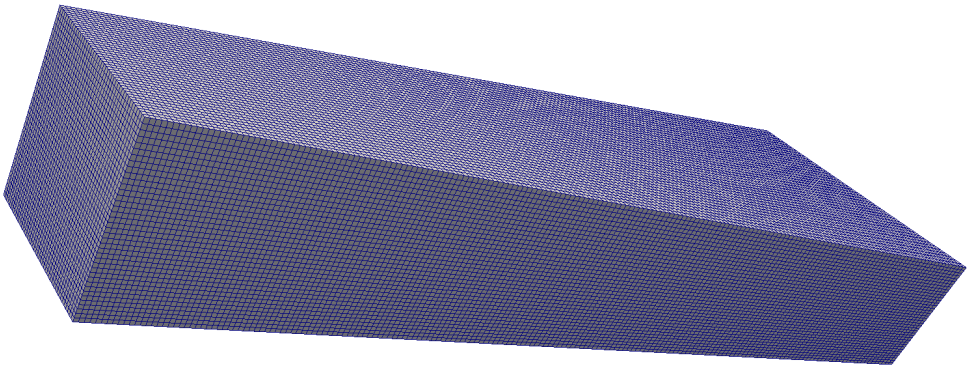


Figure 6.4: 3D view of a mesh used to run numerical simulations of the VULCANO VE-U7 experiment.

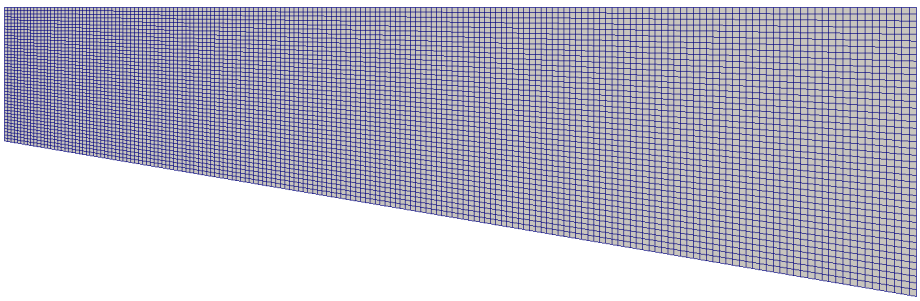


Figure 6.5: Top view of the 180 000 cells mesh used to run numerical simulations of the VULCANO VE-U7 experiment.

6.4 Initial and boundary conditions

Initially, the channel is completely filled with gas at a temperature of 350 K and corium, at a temperature of 2450 K, is continually injected through the inlet. The corium inlet velocity is time dependent. The inlet velocity value is calculated based on the corium mass values, in the channels, measured during the experiment as

$$u_{\text{Inlet}} = \frac{1}{\rho_{\text{Corium}} \cdot A_{\text{Inlet}}} \frac{m^{j+1}_{\text{Corium}} - m^{j}_{\text{Corium}}}{t^{j+1} - t^{j}}, \tag{6.4}$$

where the superscript j refers to measurements performed during the experiment (larger values of j correspond to later measurements), m^{j}_{Corium} is the mass of corium measured in the spreading channel for measurement j, t^{j} is the time of measurement j, ρ_{Corium} is the density of corium and A_{Inlet} is the area of the corium inlet (95 mm x 65 mm).

The top boundary is split in two parts, as illustrated in figure 6.3. The upstream part is an adiabatic wall with no-slip boundary condition and the pressure at the wall is extrapolated from the neighbouring cells pressure. The downstream part is an outlet with zero gradient boundary condition for the velocity and temperature, and the pressure is set to a fixed pressure.

A zero gradient pressure boundary condition is applied on the side walls and the pressure at the bottom wall is extrapolated from the value at neighbouring cells. The temperature at the bottom and side walls is set to a constant value of 350 K and a no-slip velocity boundary condition is prescribed except at the cells containing the contact line (line where the corium phase, the gas phase and the solid boundary meet). At the contact line, the Navier-slip boundary condition [41] is applied. The Navier-slip boundary condition is commonly used in CFD simulations involving a contact line because the no-slip boundary condition conflicts with the displacement of the contact line. The Navier-slip boundary condition imposes a slip velocity at the wall, which is characterised by a model parameter called slip length. A slip length value of 2 cm is prescribed in the present work. The slip length value is chosen so that the spreading of corium at the contact line is not hindered.

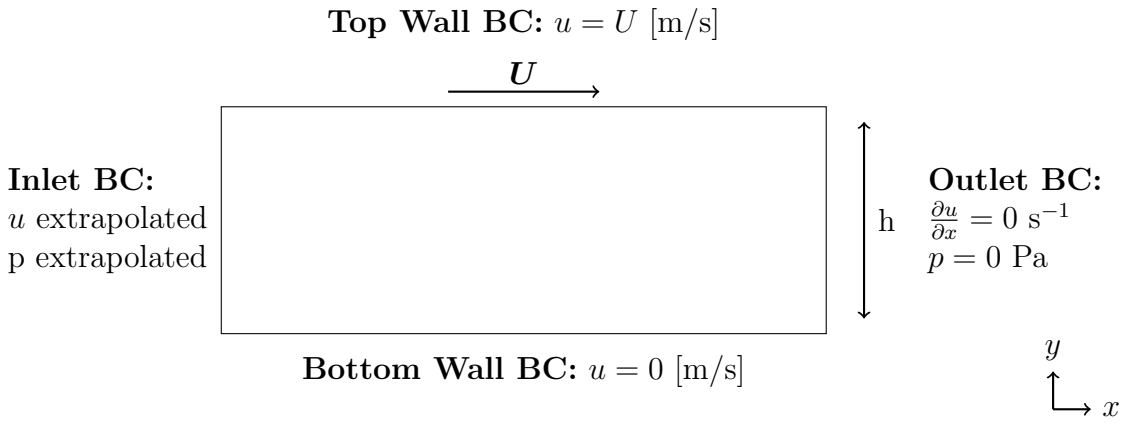


Figure 6.6: Schematic illustration of a two dimensional Couette flow including the applied boundary conditions (BC). Initially the fluid is at rest. A constant velocity U is then apply at the top boundary which drives the flow motion.

6.5 Results

6.5.1 Simulation of a Couette Flow

Considering the complexity of spreading flows and the large uncertainties associated with experiments using prototypical Corium, the simulation code is first validated using a simpler flow problem. In this section, a two dimensional Couette flow problem is considered for which analytical solutions are available ([1], [10]). The flow of a Newtonian fluid between two parallel plates separated by a distance h is simulated, as illustrated in figure 6.6. The fluid is initially at rest and flow motion is then driven by the movement of the top plate in the x direction with uniform velocity U. The analytical solution for the flow velocity along the y-axis is given by:

$$u(y,t) = U\frac{y}{h} - \frac{2U}{\pi}\sum_{n=1}^{\infty}\frac{1}{n}e^{-(n\pi/h)^2\nu t}sin\left(n\pi(1-\frac{y}{h})\right) \tag{6.5}$$

Figure 6.7 compares the simulation and analytical results for different values of $\nu t/h$. Initially, the fluid is at rest, then, for small values of $\nu t/h$ (i.e. early stage of the flow development) there is fluid motion close to the top boundary but the fluid is at rest in the lower regions. For larger values of $\nu t/h$, the flow velocity converges towards the fully developed flow velocity given by $U\frac{y}{h}$. The simulation results presented in figure 6.7 are in very good agreement with

the analytical solution.

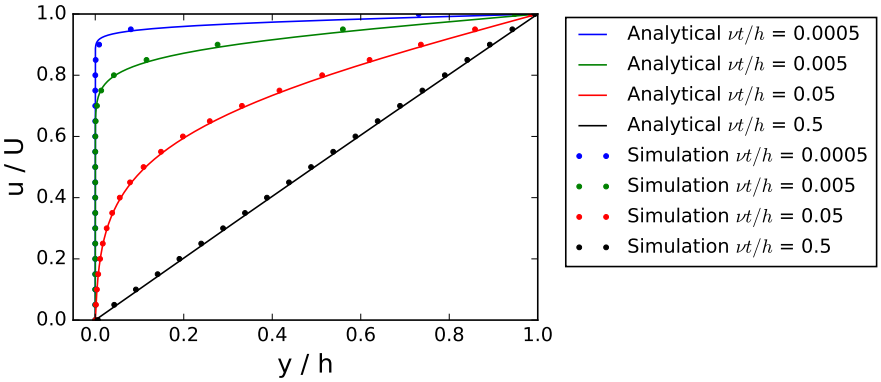


Figure 6.7: Analytical and simulation results for a two-dimensional Couette flow problem. The chart presents the adimensional velocity u/U as a function of the adimensioanl height y/h where U is the velocity applied at the top boundary and h is the spacing between the plates.

6.5.2 VE-U7 simulation results

In this section numerical simulations of the VULCANO VE-U7 experiment are discussed. The main quantitative result is the spreading length, which can be compared to the experimental results (see section 6.5.3).

Firstly, a grid refinement study is performed to investigate the refinement required to reach a converged corium spreading length. Simulations with several mesh refinements are run for four seconds. The computational cost of corium spreading simulations on fine grids is high and simulations may take several weeks to run on a high performance computing cluster using 128 CPUs. To limit the computational cost, the grid refinement is not performed uniformly throughout the grid but rather focuses on the lower cells that contain corium and the interface region. The mesh resolutions used range from 100 000 cells to 750 000 cells.

In figure 6.8, the spreading length predicted for simulations on a coarse mesh is bigger compared to simulations on finer meshes. Indeed, on coarse meshes, the volume occupied by interfacial cells, for which errors in the evaluation of the temperature-dependent corium viscosity arise, is big compared to simulations on finer meshes. Thus, on coarse meshes, increased viscosity at interfacial cells, as a result of numerical errors, is more likely to affect the flow and limit

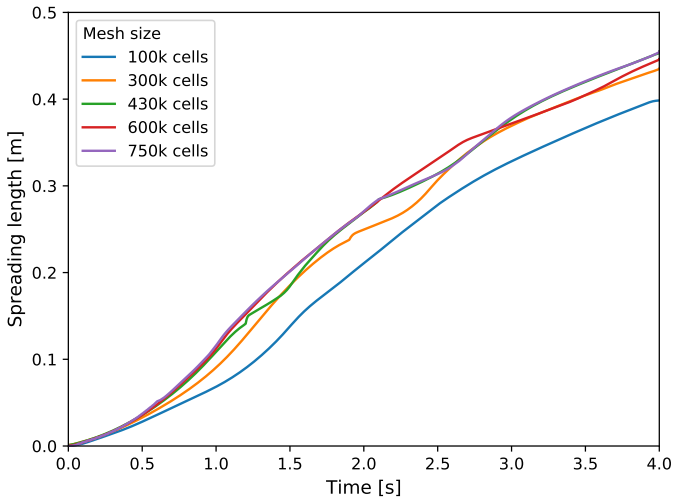


Figure 6.8: Spreading length as a function of time for different mesh refinement levels.

the spreading distance. The spreading distance does not change significantly when the mesh is refined above 600 000 cells, thus, a mesh size of 600 000 is used for the simulations presented in the remainder of this chapter.

In chapter 5, it is shown that, in a VOF framework, when simulating a fluid with temperature-dependent properties, the advection scheme used for the discretisation of the energy equation affects the predicted spreading length. Figure 6.9 presents the spreading length as a function of time obtained using the CICSAM scheme and the Upwind scheme for the discretisation of the energy equation.

In figure 6.9, the predicted spreading length using the Upwind scheme for the discretisation of the energy equation is significantly smaller than predictions using a CICSAM scheme. When the Upwind scheme is used, numerical diffusion of temperature occurs and the bulk of the corium phase is cooled down very quickly. Because of its strong temperature-dependency, corium viscosity increases to very high values and the corium spreading is hindered.

Based on the results presented in chapter 5, the CICSAM scheme is more suitable to simulate corium spreading flows as it minimizes the numerical diffusion of temperature. In the remainder of the present work, the CICSAM scheme is applied for the discretisation of the energy equation.

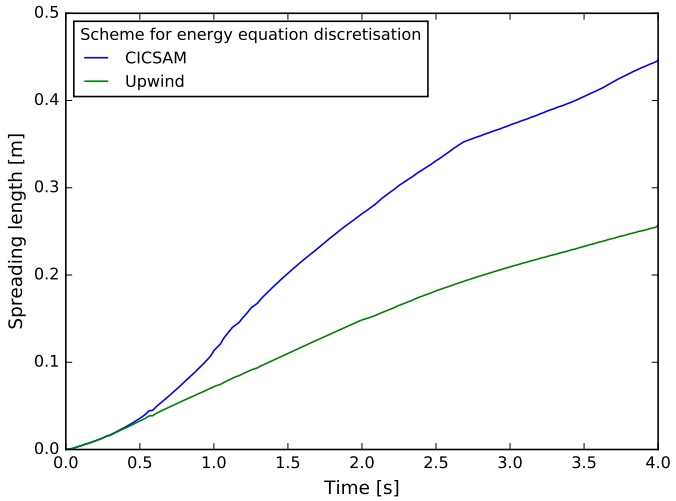


Figure 6.9: Spreading length as a function of time for different mesh refinement levels using the CICSAM scheme and the Upwind scheme for the discretisation of the energy equation.

As discussed in section 6.2, the determination of the viscosity of corium is challenging and different models for the relationship between corium viscosity and temperature are found in the literature. Figure 6.10 presents the spreading length obtained for simulation using equation 6.2 to model the temperature-dependency of corium viscosity and using the corium liquidus temperature values 2623 K, 2692 K and 2706 K, corresponding respectively to initial corium viscosity values of 2.9 Pa.s, 22 Pa.s and 35 Pa.s.

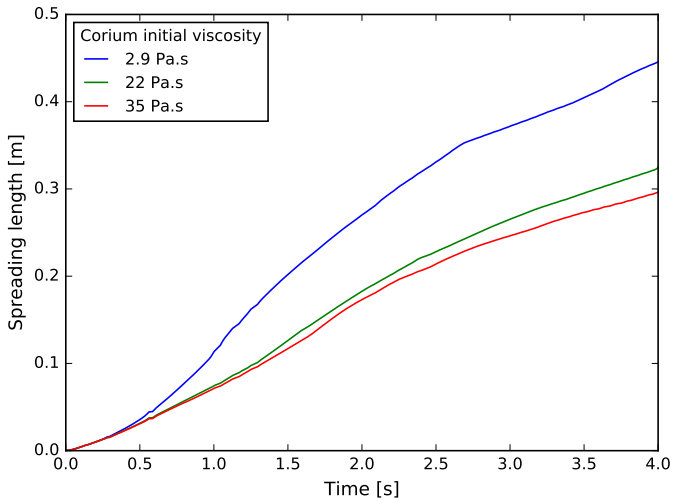


Figure 6.10: Spreading distance as a function of time for simulations using different values for the initial viscosity of corium.

In figure 6.10, there are significant discrepancies in the spreading length predicted using differ-

ent values for the initial corium viscosity. It should be noted that the corium viscosity model given in equation 6.2 depends on the corium temperature and the corium liquidus temperature, which are both uncertain parameters. The viscosity of corium is thus uncertain and the predicted spreading distance is sensitive to corium viscosity. In the remainder of this section, the simulations are run using an initial corium viscosity of 2.9 Pa.s.

As explained in section 6.2, a constant value for the corium specific heat capacity is used for the simulations. To test the influence of the specific heat capacity value on the spreading length, simulations with $c_P = 500$ J.kg^{-1}.K^{-1}, $c_P = 750$ J.kg^{-1}.K^{-1} and $c_P = 1000$ J.kg^{-1}.K^{-1} are run and the results are shown in figure 6.11. The predicted spreading length is larger when a large specific heat capacity of corium is used, however, compared to corium viscosity, the effect of specific heat capacity on spreading is not significant. The use of a constant specific heat capacity rather than a temperature-dependent one is thus acceptable. To be conservative, the specific heat capacity which leads to the shortest spreading length is used for the simulations presented in the present study ($c_P = 500$ J.kg^{-1}.K^{-1} which corresponds to the specific heat capacity of solidified corium). If the corium does not spread enough, it is more difficult to cool it down efficiently, therefore, the situation with the shortest spreading length is the most conservative.

The corium properties used in the present work are given in Journeau et al. [35] with an uncertainty range. To investigate the effect of the uncertainties in fluid properties on spreading predictions, simulations are run with the reference value of the properties as well as the upper and lower bounds of the uncertainty range. Figure 6.12 presents the spreading length as a function of time obtained using different corium density values (graph on the left) and different thermal diffusion coefficient values (graph on the right).

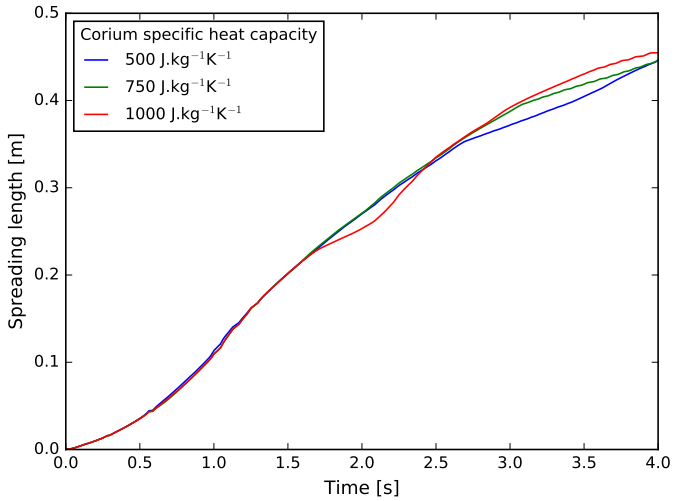


Figure 6.11: Spreading distance as a function of time for simulations using different values for the specific heat capacity values of corium.

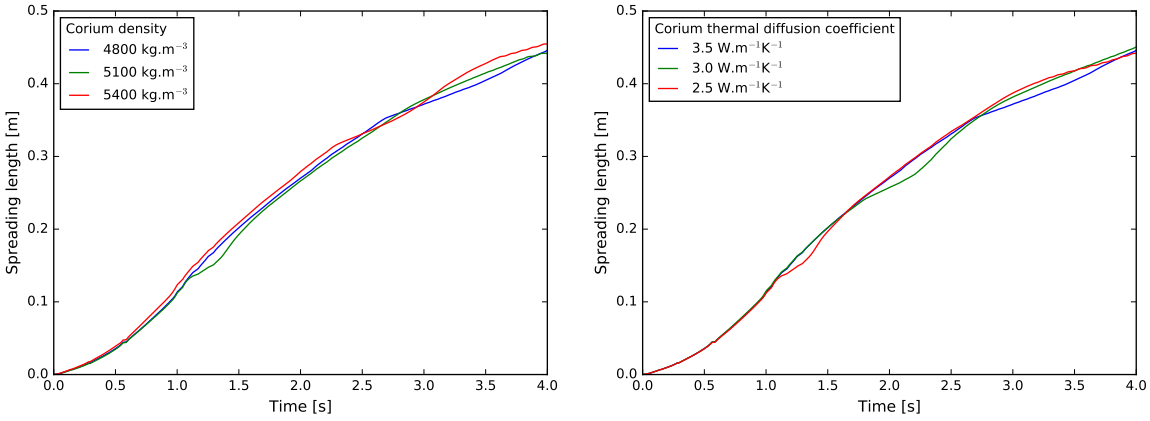


Figure 6.12: Spreading distance as a function of time for simulations using different corium density (left) and different corium thermal diffusion coefficient (right).

The simulations are not very sensitive to the thermal diffusion coefficient of corium. The thermal coefficient of corium is very high, therefore, changes in temperature quickly diffuse throughout the corium phase, even when the thermal coefficient value is reduced to 2.5 W.m^{-1}K^{-1}. When the corium density is increased, corium is cooled down more slowly, therefore, the temperature-dependent viscosity increases slowly. The predicted spreading length is thus bigger when corium density is high. However, the effect of density on spreading is small compared to the effect of viscosity.

6.5.3 Comparison with the VULCANO VE-U7 experimental results

The numerical simulation of the VULCANO VE-U7 is run for seven seconds using two different initial corium viscosity values, 2.9 Pa.s and 35 Pa.s, and the results are compared with experimental data. For this simulation, the generalised CICSAM scheme is used for the discretisation of the energy equation, the corium density is set to 5100 kg.m^{-3}, the corium thermal diffusion coefficient is set to 3.0 W.m^{-1}K^{-1} and the specific heat capacity is set to 500 J.kg^{-1}K^{-1}. The domain is represented by a 600 000 cells mesh and the applied flow time step is 1×10^{-4} s. The simulation takes two weeks to run on a high performance computing cluster using 128 CPUs.

The mass of corium in the spreading channel as a function of time is shown in figure 6.15. The predicted corium mass, which depends on the corium velocity specified at the inlet as defined in section 6.4, is in very good agreement with the experimental results. The results presented in figure 6.15 show that the boundary condition applied are correctly implemented and that mass conservation is ensured during the simulation.

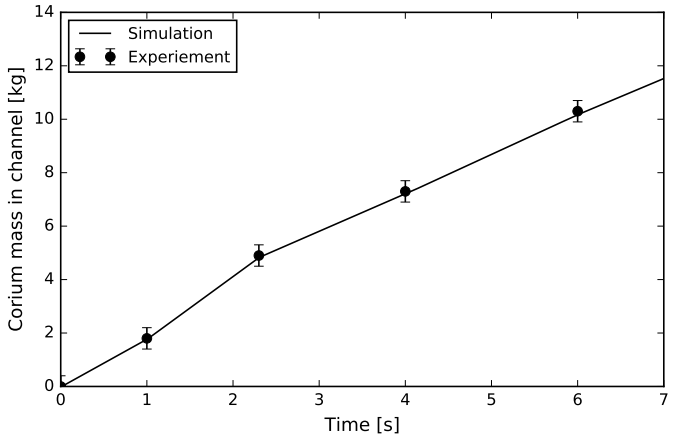


Figure 6.13: Mass of corium in a spreading channel as a function of time.

The corium temperature at the end of the simulation is presented in figure 6.14. The temperature in the corium bulk remains close to the initial corium temperature of 2450 K but the corium layer near the corium/air interface is colder. Corium cooling near the interface is due to the heat exchange with the surrounding air and it causes the corium to solidify. Corium cooling also happens at the bottom layer due to the heat exchange with the substrate. The so-

lidification observed at the interface and at the bottom layer is expected to limit the spreading of corium.

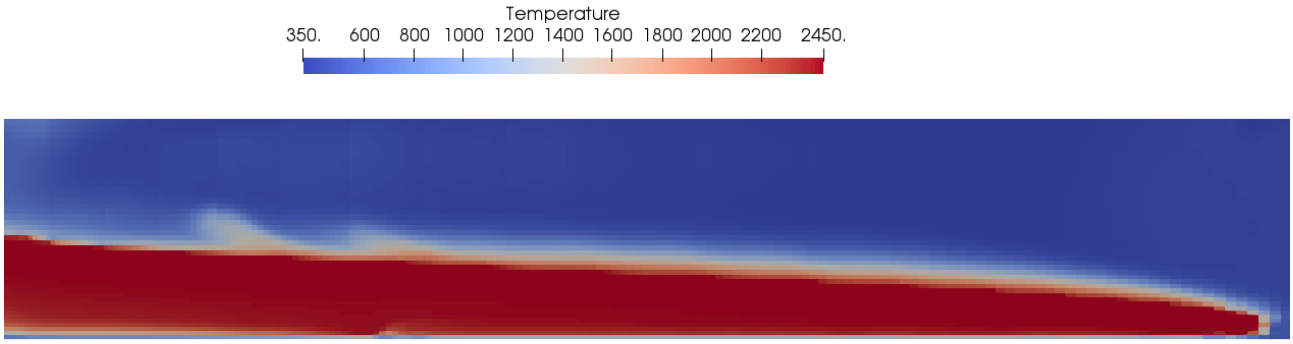


Figure 6.14: Corium temperature at the end of the simulation on a slice of the mesh normal to the y-axis.

Figure 6.15 presents the height of the corium layer as a function of the x-distance at the end of the simulation as well as the experimental data for the post-mortem corium heights on the two channels. Close to the Corium inlet, experimental and simulation results for the corium height are similar, but, further from the inlet, the corium height predicted by the simulation drops from 0.065 m to about 0.04 m. Overall, the predicted corium height is smaller than the experimental results and the simulated corium is more spread than in the experimental results. During the experiment, a rigid crust forms at the spreading front which prevents the front propagation and favours a thicker corium layer. With the simulation, corium solidification is modelled using an apparent viscosity model which does not accurately represent the behaviour of a rigid crust. The difference in corium layer height could be explained by the simplifications introduced by the corium solidification model. Indeed, real corium properties, such as density, heat capacity or thermal conductivity, vary significantly when corium solidifies while the simulation performed assumes constant properties except for the viscosity. The uncertainty on corium properties and the modelling assumptions could also explain the differences between simulation and experiment observed in figure 6.15.

Figure 6.16 shows the spreading lengths obtained in the experiment on the concrete and ceramic channels and the spreading lengths predicted by the simulations.

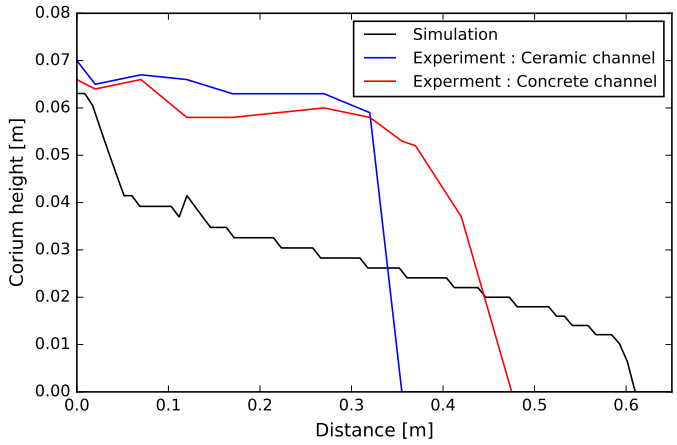


Figure 6.15: Height of the corium layer as a function of the x-distance.

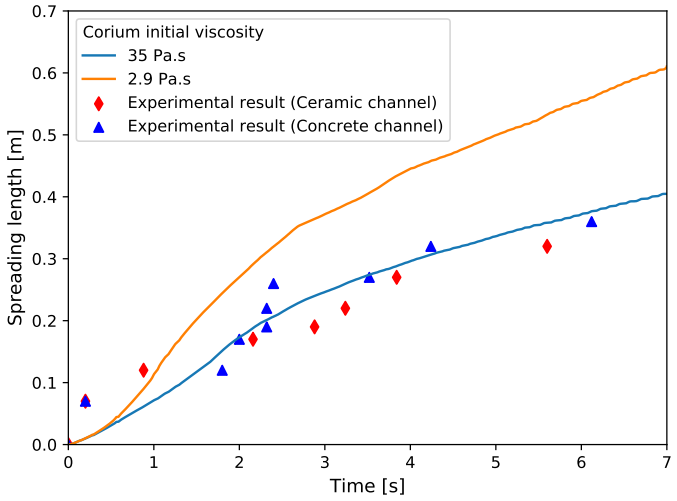


Figure 6.16: Spreading distance as a function of time predicted by the numerical simulation and experimental results.

There is a good match between the experimental results and the simulation using an initial corium viscosity of 35 Pa.s. In that case, the discrepancy between the simulation and the experiment is less than ten percent. When the initial viscosity of corium is 2.9 Pa.s (corresponding to a corium inlet temperature of 2450 K and a corium liquidus temperature of 2623 K), the simulation over-predicts the spreading distance compared to the experimental results. The over-prediction of the spreading length could be explained by the simplifications and the assumptions made, such as the neglected heat transfer by radiation at the corium free surface or the use of an effective viscosity model to capture corium solidification, which does not explicitly

take into account corium crust formation and crust fracture.

6.6 Estimation of heat loss by radiation

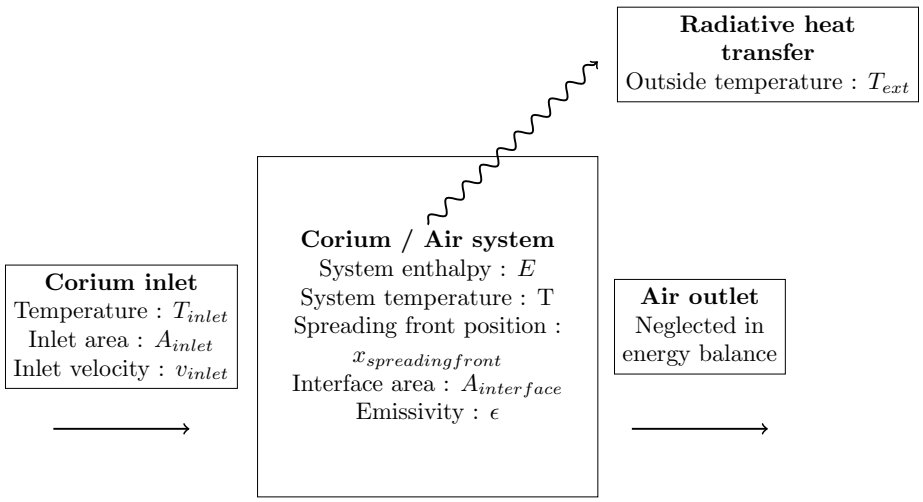


Figure 6.17: Illustration of the energy balance in the system considered for radiation heat loss estimation.

In the simulation presented in section 6.5, heat transfer by radiation is not considered. In the present section, a study is performed to estimate the impact of heat transfer by radiation on the evolution of corium bulk temperature during spreading. The system considered for the study, illustrated in figure 6.17, is described by a single temperature value (zero-dimension analysis). The change of energy in the system is given by:

$$\Delta E = E_{in} - E_{out} - E_{radiation}, \tag{6.6}$$

Where E is the enthalpy in the system, E_{in} is the enthalpy of the fluid entering the control volume, E_{out} is the enthalpy of the fluid leaving the control volume and $E_{radiation}$ is the heat loss by radiation. Considering that the enthalpy of the air phase is small compared to the enthalpy of the corium phase, air is neglected in the energy balance. Given this simplification, corium temperature is calculated as follows

$$E = E_{corium} + E_{air} \approx E_{corium}, \tag{6.7}$$

$$T_{corium} = \frac{E_{corium}}{\rho_{corium} c_{p,corium} V_{corium}} \approx \frac{E}{\rho_{corium} c_{p,corium} V_{corium}}, \tag{6.8}$$

where V_{corium} is the volume occupied by the corium phase. Assuming the fluid entering the control volume is pure corium and assuming that its temperature is constant, the enthalpy of the fluid entering the system is :

$$E_{in} = \rho_{corium} c_{p,corium} A_{inlet} v_{inlet}, \tag{6.9}$$

where A_{inlet} is the area of the inlet and v_{inlet} is the inlet velocity taken from the experimental results. The fluid leaving the control volume is assumed to be pure air and its enthalpy is thus neglected.

$$E_{out} \approx 0. \tag{6.10}$$

The heat loss by radiation is described using the StefanBoltzmann law as

$$E_{radiation} = \epsilon \sigma_B A_{interface} (T_{corium} - T_{ext})^4, \tag{6.11}$$

where $\epsilon = 0.8$ is the emissivity of corium given in the experimental results, σ is the Boltzmann constant, $A_{interface}$ is the corium/air interface area and T_{ext} is the temperature outside the system which is assumed to have a constant value of 500 K. To compute the interface area, a simple model is used where the corium height is assumed to be constant from the inlet to the spreading front. The interface area is thus computed as

$$A_{interface} = A_{top} + A_{front}, \tag{6.12}$$

where A_{top} and A_{front} are respectively the top surface and the front surface of the corium volume. Using the notation from figure 6.18, A_{top} and A_{front} are calculated in accordance to the geometry of the VULCANO experimental setup as follows

$$A_{top} = w_{inlet} x_{coriumfront} + 0.5(w_{coriumfront} - w_{inlet}) x_{coriumfront} \tag{6.13}$$

and A_{front} is given by

$$A_{front} = h_{corium} w_{coriumfront}. \tag{6.14}$$

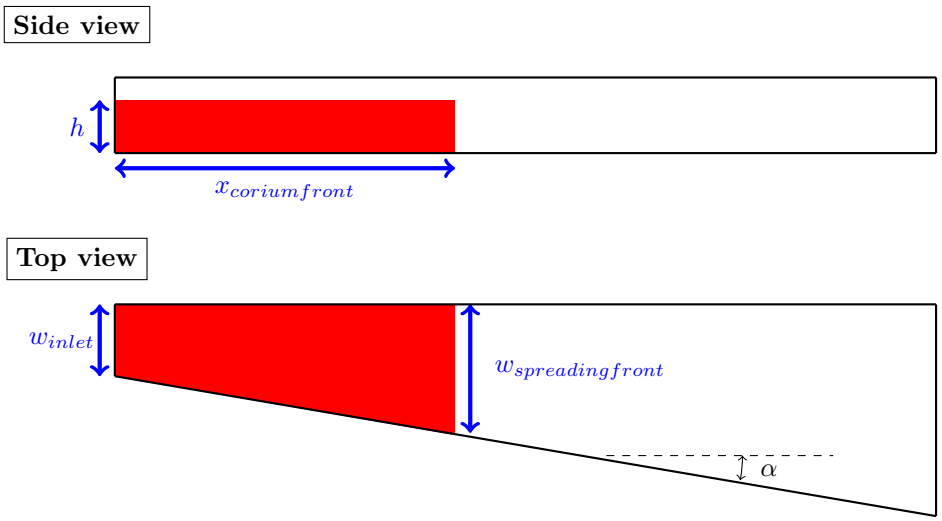


Figure 6.18: Illustration of the VULCANO VE-U7 experimental setup. The corium volume is simplified assuming a constant height from the inlet up to the corium front. The variables presented in blue are used to compute the corium/air interface area.

The corium front position $x_{corium_{front}}$ as a function of time is taken from the experimental results. A constant value of 0.065 m is used for the height of the corium layer h corresponding to the post-mortem height of the spread. The width of the corium inlet w_{inlet} is fixed (0.095 m) whereas the width of the corium front $w_{coriumfront}$ is time dependent. $w_{corium_{front}}$ is computed from the corium front position $x_{coriumfront}$ as

$$w_{coriumfront} = tan(\alpha) x_{coriumfront}, \tag{6.15}$$

where α is the angle between the side wall and the x-axes (see figure 6.18). The model described above is then discretised using a forward Euler scheme

$$\frac{E^{n+1} - E^n}{\Delta t} = f(E^n), \tag{6.16}$$

and it is implemented in a stand-alone code written in the python language. The computed corium temperature as a function of time is presented in figure 6.19. A time step value of $1 \cdot 10^{-2}$ s ensures a converged value of the corium temperature after seven seconds of simulation. The final corium temperature computed using the zero dimensional analysis model is 2432 K which means that the heat loss by radiation is responsible for a 68 K decrease in temperature. With the zero-dimensional model, corium is described with a single temperature value but in real life, a temperature profile in the corium layer is expected. Corium temperature close to the interface should be lower than the mean corium temperature, therefore, the decrease in temperature due to heat losses by radiation at the corium/air interface is expected to be higher than 68 K. Given the high temperature dependency of corium viscosity, the impact of heat losses by radiation on corium spreading is probably relevant and should be studied in future work on corium spreading.

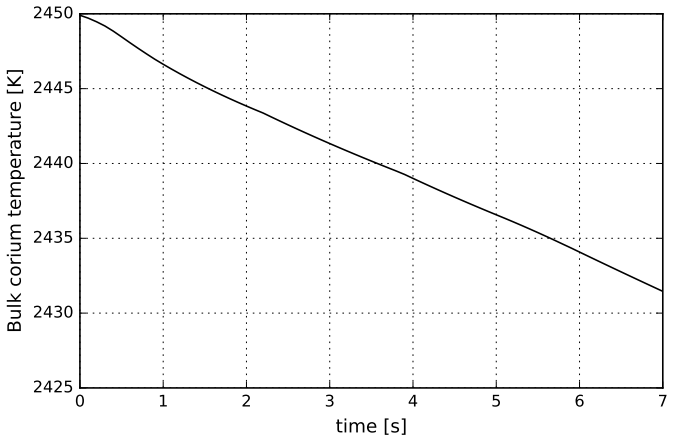


Figure 6.19: Analysis of the decrease in corium bulk temperature as a function of time using a zero-dimensional model.

6.7 Conclusions

In the present work, CFD simulations of the VULCANO VE-U7 corium spreading experiment are presented using a simplified corium model. Corium is assumed to be a Newtonian non-reactive fluid with constant properties except for the viscosity which is temperature dependent. An effective viscosity model, where the viscosity of corium depends on the temperature, is used to capture the corium solidification. Phenomena, such as ablation and fracture of the substrate, heat transfer by radiation and crust mechanics are neglected in the CFD simulations.

After validating the simulation code performances on a Couette flow problem, a study of the parameters that affect the spreading predictions is conducted. In the present work, it is shown that a mesh size of 600 000 cells is required to reach a converged value of the spreading length. The predicted spreading length is also sensitive to the choice of advection scheme for the discretisation of the energy equation. Specific heat capacity and thermal diffusion of corium did not significantly affect the simulation results. The determination of the viscosity of corium is challenging and the viscosity models available contain uncertainties. When different initial values of the corium viscosity are used, there are significant discrepancies in the predicted spreading length.

The height of the corium layer predicted by the simulation code is compared to the post-mortem corium height measured after the experiment. The simulated corium is more spread compared to the experimental results. The differences in post-mortem corium shape could be explained by the model used to capture corium solidification which does not accurately represents the behaviour of a rigid crust. Predicted spreading lengths obtained using two different values for the uncertain initial corium viscosity are compared to experimental results. Using an initial corium viscosity of 35 Pa.s, the discrepancy between experiment and simulations in terms of spreading length is below ten percent. The simulation using an initial corium viscosity of 2.9 Pa.s, however, over-predict the spreading length. These results highlight the effect of uncertainties in corium fluid properties on spreading simulations.

A study of the impact of heat transfer by radiation on corium bulk temperature during spreading

is conducted and the corium temperature drop computed is larger than 60K. Considering the strong temperature dependency of corium viscosity, the large temperature drop due to heat transfer by radiation is expected to impact corium spreading and heat transfer by radiation should therefore be considered in future corium spreading simulations.

The present study shows that a fine numerical grid is required to obtain converged results using a VOF method to capture the evolution of the corium/air interface. The computational cost of such simulations is thus high and simulation of larger scale systems, such as a full core-catcher geometry, may be unpractical.

Thesis conclusions

Corium is a very complex fluid to study due to the many phenomena – such as radioactivity, chemical reactions, phase change – which affect its behaviour. Experiments involving corium are costly and the extreme conditions of corium flows (e.g. temperature above 2000 ^{o}C), limit the number and type of measurements that can be performed during experiments. Numerical simulations offer the opportunity to investigate large-scale systems and hazardous environments. However, simulation results may lack representativeness due to assumptions and simplifications in the modelling and due to errors in the numerical solutions of the equations governing the fluid flow. In this thesis, the use of a general purpose three-dimensional Computational Fluid Dynamics (CFD) code with the Volume-of-Fluid methods to simulate corium spreading is investigated. Numerical errors associated with such simulations are identified and their impact on corium spreading predictions is assessed. Numerical methods to limit such errors are proposed and simulation results are compared to a corium spreading experiment.

In the scientific literature, simulations of corium flows have been performed using codes specifically developed for corium simulations (LAVA [2, 58], THEMA [61]) as well as general purpose CFD code (CORFLOW [77, 78]). Unlike the present work, the codes MELTSPREAD, THEMA and LAVA solve governing equations integrated over the vertical axis, which has the advantage of numerical simplicity and low computational cost [59]. The corium spreading simulations presented in this thesis are fully three-dimensional and they are performed on a fine computational mesh. The vertical velocity profile is, thus, computed rather than assumed and the simulations are applicable to a large range of flow configurations.

The three-dimensional CFD code CORFLOW, which has also been used for corium simulations,

considers a single fluid, corium, and prescribes boundary conditions at the free surface of corium. In contrast, the present work considers a two phase corium/air system and uses the VOF method to capture the evolution of the corium free surface (i.e. the corium/air interface). With the VOF method, the displacement of the interface is computed by solving a transport equation and it is not necessary to prescribe boundary conditions at the interface. Compared to corium simulations performed using a commercially available CFD code [79], the simulations presented in this thesis are performed using a CFD code, developed for research purposes, that allows for the investigation of a large range of numerical methods.

Most of the past corium simulations in the literature have been performed between 1995 and 2005. Thanks to the large increase in computational power and the development of efficient tool for parallel computations observed in the last decades, the simulations performed in the present work use much finer numerical meshes compared to simulations in the literature. For their simulations, Ye et al. [79] report a mesh size of 100 000 cells, LAVA simulations of the VE-U7 experiment used a mesh with 1900 cells and Wittmaack [78] reports a maximum mesh size of 10 000 cells for CORFLOW simulations. In contrast, when simulating corium spreading flows, the present thesis recommends a mesh size of 600 000 cells to yield a converged spreading length.

Numerical simulations of corium flows are notoriously difficult due to the complexity of the flow, which involves phenomena such as phase change, chemical reactions or non-Newtonian behaviour, and due to the uncertainty in the physical properties of corium (composition, viscosity, density, etc). In the present work, the implementation of a Bingham fluid model to model the non-Newtonian behaviour of corium, is presented in section 4. Numerical simulations using Bingham fluid models are available in the scientific literature, but, the implementation in a three-dimensional CFD code is rarely described in details. This thesis provides a detailed descriptions of the implementation of several regularized Bingham models in a numerical framework based on the finite volume methods. The proposed implementation could be used as a basis for future works using Bingham fluid models to compare different possible implementations.

A lid-driven cavity test case is used to test the different Bingham fluid models implemented. It is shown that all the models considered ([11, 43, 45]) yield similar results and the choice of a model is, therefore, not critical to make Bingham fluid flow predictions. The simulation results on the lid-driven cavity test case are compared to results available in the scientific literature ([12, 73]) for different values of the yield stress, which characterizes a Bingham fluid. Some discrepancies are observed for high values of the yield stress but, for most simulations, the implementation of a regularized Bingham fluid model proposed in this thesis leads to predictions similar to the results from the literature. It is observed that the non-linearity of the Bingham fluid models increase the computational cost of the simulations. For corium flow simulations, the sensitivity of the flow predictions to the use of a Bingham model needs to be assessed. If the Bingham fluid model does not impact corium spreading considerably, assuming a Newtonian fluid behaviour may limit the computational cost of simulations and facilitate the convergence of the numerical solvers.

The main goal of this thesis is to develop numerical methods to simulate two-phase flow with temperature dependent properties, such as the flow of solidifying nuclear corium in a core-catcher. To achieve this goal, a detailed investigation of the numerical errors affecting corium simulations is performed. In particular, the present work focuses on simulations using the Volume-of-Fluid method to capture the dynamics of the interface between air and corium. Interface capturing methods, such as VOF methods, can be implemented using a fixed Eulerian mesh as opposed to moving mesh methods which require to recompute the mesh at each time step of the simulation. Among methods based on a fixed Eulerian fluid mesh, VOF methods are generally simpler to implement compared to front-tracking methods which require the adaption of the surface mesh representing the interface ([51, 66]). Lastly, the implementation of VOF methods in a parallel, three dimensional CFD code is relatively simple.

Using the numerical framework described in chapter 3 to simulate a corium/air flow with temperature dependent properties, the occurrence of an artificial increase in corium viscosity in the vicinity of the interface has been observed. The numerical errors causing this artificial increase have been identified and the impact of the errors on spreading flow predictions has been assessed. Firstly, numerical diffusion of temperature is shown to cause an artificial cooling

of corium and, therefore, an increase of its temperature dependent viscosity. The CICSAM discretisation scheme, designed specifically to represent sharp discontinuities, has been adapted in the present work for the discretisation of the energy equation. This scheme proves to minimize numerical diffusion of temperature and, thus, limits the errors in viscosity predictions. The second source of numerical errors affecting corium simulation identified in this thesis is erroneous evaluation of corium viscosity in mesh cells that contain the interface. This error in viscosity evaluation is specific to the Volume-of-Fluid method for which a single temperature value is available in each mesh cell even if two fluids are present. In the present work, a new method to evaluate the temperature dependent viscosity of corium using the temperature in neighbour cells containing a high volume fraction of corium is proposed. This new methods proves to greatly improve the evaluation of corium viscosity in cells that contain the interface and, therefore, improve the accuracy of corium spreading predictions.

In chapter 6, the numerical framework investigated in this thesis is applied to a corium flow simulation. The VULCANO VE-U7 experiment, which investigates the spreading of prototypical corium on different substrate materials, is simulated. Corium properties are largely uncertain and different values have been used in the scientific literature to run corium spreading simulations. A discussion of the models and assumptions used for the simulations is, thus, presented. In particular, the initial temperature of corium and the law used to represent the temperature dependent viscosity of corium are uncertain and depending on the values considered, the initial viscosity of corium can vary by an order of magnitude (from 3 Pa.s to 35 Pa.s). A sensibility study is performed to test the influence of different fluid properties and numerical methods on spreading predictions. It is shown that the mesh resolution, the scheme used to discretise the energy equation and the initial viscosity of corium (which depends on the initial corium temperature) have a significant effect on spreading predictions.

The simulation results are compared with the experimental results. It is shown that using the corium viscosity calculated using the viscosity law given by Journeau et al. [35] in the paper describing the VULCANO VE-U7 experiment, the spreading distance is over-predicted by about 50%. On the other hands, when an initial corium viscosity of 35 Pa.s is used, (corresponding to the initial viscosity value reported by Wittmaack [77]), the predicted spreading distance is

in a good agreement with experimental data. The simulation results obtained in the present work are also comparable to the results from the literature obtained using the LAVA, THEMA and CORFLOW codes which suggests that simulations could be used to investigate large scale corium spreading configurations representative of a nuclear reactor conditions.

In the work presented in this thesis, corium cooling by radiation at the corium/air interface is neglected in the fluid dynamics simulations. The calculation of heat losses by radiation in a CFD simulation requires the knowledge of the interface area, however, with the VOF methods, an explicit representation of the interface is not directly available. To estimate the impact of neglecting corium heat losses due to radiation on corium spreading, a zero-dimensional model is constructed taking into account radiation heat losses and assuming a simplified shape for the corium air interface. A corium bulk temperature drop due to radiation heat loss of 68 K is calculated at the end of the spreading. Considering the strong temperature dependency of corium viscosity, the temperature drop due to radiation heat loss is expected to have an impact on spreading therefore it should be considered in future works. For that purpose, the radiation heat loss could be estimated prior to fluid dynamics simulations and prescribed explicitly as a source term in the energy equation.

As discussed in chapter 5, numerical errors can affect the evaluation of the temperature dependent viscosity of corium in a VOF framework. Numerical methods to limit such errors have been proposed in this thesis but future work could investigate other methods. For instance, with Lagrangian methods the mesh is adapted at each time step. The mesh is thus fitted to the interface position and there is always only one fluid in each mesh cell, with prevent errors associated with temperature dependent viscosity evaluation.

A fine mesh (600 000 cells) was used to run the simulations in chapter 6 (VULCANO) and the results obtained are comparable to simulations performed on coarse meshes (CORFLOW simulations used about 10 000 cells). Since that the geometry considered for reactor scale simulations is bigger than the VULANO experimental set-up, the computational cost of simulations using a fine mesh might be prohibitive. Future works could focus on numerical methods with a reduced computational cost. For instance, Smoothed Particle Hydrodynamics (SPH) methods

have been used to simulate large scale systems such as ocean waves or volcanic flows ([44], [52]). SPH methods are mesh free and SPH simulations generally have a lower computational cost than equivalent mesh-based simulations.

Bibliography

[1] Acheson, D.J. *Elementary fluid dynamics*. Oxford University Press, 1990.

[2] H.-J. Allelein, A. Breest, and C. Spengler. Simulation of core melt spreading with LAVA: Theoretical background and Status of Validation, 2000.

[3] H. Alsmeyer, T. Cron, J. Foit, G. Messemer, S. Schmidt-Stiefel, W. Häfner, and H. Kriscio. Test report of the melt spreading tests ECOKATS-V1 and ECOKATS-1. *SAM-ECOSTAR-D15/FZKA 7064*, 2004.

[4] H. Alsmeyer, T. Cron, G. Messemer, and W. Häfner. ECOKATS-2: A large scale experiment on melt spreading and subsequent cooling by top flooding, 2004.

[5] H. Alsmeyer, G. Albrecht, L. Meyer, W. Häfner, C. Journeau, M. Fischer, S. Hellman, M. Eddi, H.-J. Allelein, M. Bürger, B. Sehgal, M. Koch, Z. Alkan, J. Petrov, M. Gaune-Escard, E. Altstadt, and G. Bandini. Ex-vessel core melt stabilization research (ECOSTAR). *Nuclear Engineering and Design*, 235(2-4):271–284, February 2005.

[6] E. d. C. Andrade. XLI. *A theory of the viscosity of liquids.—Part I. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 17(112):497–511, February 1934.

[7] ANSYS Inc., Academic Research Mechanical. Fluent 6.3 User's Guide, 2006.

[8] R. Aris. *Vectors, Tensors, and the Basic Equations of Fluid Mechanics*. Dover books on engineering. Dover Publications, New York, dover ed edition, 1989.

[9] Autorité de Sûreté Nucléaire. Technical guidelines for the design and construction of the EPR. *http://www.french-nuclear-safety.fr/Inspections/Supervision-of-the-EPR-reactor/Resources/Technical-guidelines-for-the-design-and-construction-of-the-EPR.*

[10] Batchelor, G.K. *An introduction to fluid dynamics.* Cambridge University Press, Cambridge, 2000.

[11] M. Bercovier and M. Engelman. A finite-element method for incompressible non-Newtonian flows. *Journal of Computational Physics*, 36(3):313–326, July 1980.

[12] O. Botella and R. Peyret. Benchmark spectral results on the lid-driven cavity flow. *Computers & Fluids*, 27(4):421–433, May 1998.

[13] F. Bouteille, G. Azarian, D. Bittermann, J. Brauns, and J. Eyink. The EPR overall approach for severe accident mitigation. *Nuclear Engineering and Design*, 236(14-16): 1464–1470, August 2006.

[14] J. Brackbill, D. Kothe, and C. Zemach. Continuum Method for Modeling Surface Tension. *Journal of Computational Physics*, 100:335–354, 1992.

[15] G. Cognet, H. Alsmeyer, W. Tromm, D. Magallon, R. Wittmaack, B. Sehgal, W. Widmann, L. De Cecco, R. Ocelli, G. Azarian, D. Pineau, B. Spindler, G. Fieg, H. Werle, C. Journeau, M. Cranga, and G. Laffont. Corium spreading and coolability. *Nuclear Engineering and Design*, 209(1-3):127–138, November 2001.

[16] M. Darwish. Development and Testing of a Robust Free-Surface Finite Volume Method. Technical report, Faculty of Engineering and Architecture, Amercian University of Beirut, Beirut, Lebanon, 2003.

[17] G. De Vahl Davis. Natural convection of air in a square cavity: A bench mark numerical solution. *International Journal for Numerical Methods in Fluids*, 3(3):249–264, May 1983.

[18] E. J. Dean, R. Glowinski, and G. Guidoboni. On the numerical simulation of Bingham visco-plastic flow: Old and new results. *Journal of Non-Newtonian Fluid Mechanics*, 142 (1-3):36–62, March 2007.

[19] N. Deen, M. Van Sint Annaland, and J. Kuipers. Multi-scale modeling of dispersed gas–liquid two-phase flow. *Chemical Engineering Science*, 59(8-9):1853–1861, 2004.

[20] F. Denner and B. van Wachem. On the convolution of fluid properties and surface force for interface capturing methods. *International Journal of Multiphase Flow*, 54:61–64, 2013.

[21] F. Denner and B. van Wachem. Fully-coupled balanced-force VOF framework for arbitrary meshes with least-squares curvature evaluation from volume fractions. *Numerical Heat Transfer Part B: Fundamentals*, 65(3):218–255, 2014.

[22] F. Denner and B. van Wachem. Compressive VOF method with skewness correction to capture sharp interfaces on arbitrary meshes. *Journal of Computational Physics*, 279: 127–144, 2014.

[23] T. Dinh, M. Konovalikhin, and B. Sehgal. Core melt spreading on a reactor containment floor. *Progress in Nuclear Energy*, 36(4):405–468, January 2000.

[24] M. Farmer, J. Sienicki, C. Chu, and B. Spencer. The MELTSPREAD-1 computer code for the analysis of transient spreading and cooling of high temperature melts. *Report EPRI TR-103413*, 1993.

[25] J. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. Springer Verlag, Berlin Heidelberg New York, 3. edition, 2002.

[26] G. Fieg, F. Huber, H. Werle, and R. Wittmaack. Simulation Experiments on the spreading behavior of molten core melts, 1996.

[27] M. Fischer. The severe accident mitigation concept and the design measures for core melt retention of the European Pressurized Reactor (EPR). *Nuclear Engineering and Design*, 230(1-3):169–180, May 2004.

[28] R. Gingold and J. Monaghan. Smoothed particle hydrodynamics: Theory and application to non-sperical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389, 1977.

[29] V. Gopala and B. van Wachem. Volume of fluid methods for immiscible-fluid and free-surface flows. *Chemical Engineering Journal*, 141(1-3):204–221, July 2008.

[30] A. Harten. High Resolution Schemes for Hyperbolic Conservation Laws. *Journal of Computational Physics*, 49:357–393, August 1983.

[31] C. Hirt and B. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981.

[32] C. Journeau. L'étalement du corium : Hydrodynamique, rhéologie et solidification d'un bain d'oxydes à haute température, 2006.

[33] C. Journeau, E. Boccaccio, C. Brayer, G. Cognet, J.-F. Haquet, C. Jégou, P. Piluso, and J. Monerris. Ex-vessel corium spreading: Results from the VULCANO spreading tests. *Nuclear Engineering and Design*, 223(1):75–102, July 2003.

[34] C. Journeau, P. Piluso, and K. Frolov. Corium physical properties for severe accident R&D, 07 2004.

[35] C. Journeau, J.-F. Haquet, B. Spindler, C. Spengler, and J. Foit. The VULCANO VE-U7 Corium spreading benchmark. *Progress in Nuclear Energy*, 48(3):215–234, April 2006.

[36] I. Kataoka. Local instant formulation of two-phase flow. *International Journal of Multiphase Flow*, 12(5):745–758, September 1986.

[37] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti. Modelling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics*, 113:134–147, 1994.

[38] B. P. Leonard. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, 88(1): 17–74, 1991.

[39] B. Michel, B. Piar, F. Babik, J. Latché, G. Guillard, and C. De Pascale. Synthesis of the validation of the CROCO V1 spreading code. *Proc. OECD Workshop on Ex-Vessel Debris Coolability, Karlsruhe*, 1999.

[40] E. Mitsoulis and T. Zisis. Flow of Bingham plastics in a lid-driven square cavity. *Journal of Non-Newtonian Fluid Mechanics*, 101(1-3):173–180, November 2001.

[41] C. Navier. *Mémoire Sur Les Lois Du Mouvement Des Fluides*. Mem. Acad. R. Sci. Paris 6: 389-416, 1823.

[42] M. Nie. Temporary melt retention in the reactor pit of the European pressurized water reactor (EPR), May 2005.

[43] E. O'Donovan and R. Tanner. Numerical study of the Bingham squeeze film problem. *Journal of Non-Newtonian Fluid Mechanics*, 15(1):75–83, January 1984.

[44] P. Omidvar, H. Norouzi, and A. Zarghami. Smoothed particle hydrodynamics for water wave propagation in a channel. *International Journal of Modern Physics C*, 26:1550085, 01 2015.

[45] T. C. Papanastasiou. Flows of Materials with Yield. *Journal of Rheology*, 31(5):385–404, July 1987.

[46] S. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Company, 1980.

[47] C. S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977.

[48] B. Piar, B. Michel, F. Babik, J. Latché, G. Guillard, and J. Ruggieri. CROCO: A Computer Code for Corium Spreading. *Proc. 9th International Topical Meeting on Nuclear Thermal Hydraulics (NURETH-9), San Francisco CA*, 1999.

[49] P. Piluso, C. Journeau, G. Cognet, D. Magallon, and J. Seiler. Importance of prototypic-corium experiments for severe accident research. *Proc. International Conference on Nuclear Engineering (ICONE), Nice, France*, 2001.

[50] P. Piluso, C. Journeau, J. Monerris, and G. Cognet. Viscosity Measurements of Ceramic Oxides by Aerodynamic Levitation. *International Journal of Thermophysics*, 23(5):1229–1240, 2002.

[51] S. Popinet and S. Zaleski. A front-tracking algorithm for accurate representation of surface tension. *International Journal for Numerical Methods in Fluids*, 30:775 – 793, 1999.

[52] M. Prakash and P. W. Cleary. Three dimensional modelling of lava flow using smoothed particle hydrodynamics. *Applied Mathematical Modelling*, 35(6):3021 – 3035, 2011.

[53] M. Raessi, M. Bussmann, and J. Mostaghimi. A semi-implicit finite volume implementation of the CSF method for treating surface tension in interfacial flows. *International Journal for Numerical Methods in Fluids*, 59:1093–1110, 2009.

[54] M. Ramacciotti, C. Journeau, F. Sudreau, and G. Cognet. Viscosity models for corium melts. *Nuclear Engineering and Design*, 204(1-3):377–389, February 2001.

[55] C. M. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, November 1983.

[56] P. Roe. Characteristic-based schemes for the euler equations. *Annual Review of Fluid Mechanics*, 18:337–365, 1986.

[57] B. R. Sehgal, editor. *Nuclear Safety in Light Water Reactors: Severe Accident Phenomenology*. Elsevier/Academic Press, Amsterdam ; Boston, 1st ed edition, 2012.

[58] C. Spengler. Simulation der Ausbreitung von Kernschmelzen mit einem Binghamschen Fliessmodel unter Beruecksichtigung von Phasenuebergaengen, 2002.

[59] B. Spindler and J. Veteau. The simulation of melt spreading with THEMA code. *Nuclear Engineering and Design*, 236(4):415–424, February 2006.

[60] B. Spindler, J. Veteau, C. Brayer, M. Cranga, L. De Cecco, P. Montanelli, and D. Pineau. Assessment of THEMA code against spreading experiments. *Wiss. Ber. FZKA 6475*, pages 221–234, 2000.

[61] B. Spindler and J. Veteau. Status of the assessment of the spreading code Thema against the Corine experiments. *SARJ Mtg. (Mtg. Severe Accident Research held in Japan), Tokyo, Japan*, 1999.

[62] W. Steinwarz, A. Alemberti, W. Häfner, Z. Alkan, and M. Fischer. Investigations on the phenomenology of ex-vessel core melt behaviour (COMAS). *Nuclear Engineering and Design*, 209(1-3):139–146, November 2001.

[63] H. Suzuki, T. Matsumoto, I. Sasaki, T. Mitadera, M. Matsumoto, and T. Zama. Fundamental experiment and analysis for melt spreading on concrete floor. *Proc. 2nd ASME/JSME Nucl. Eng. Conf.*, pages 404–407, 1993.

[64] P. Sweby. High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. *SIAM Journal on Numerical Analysis*, 21(5):995–1011, 1984.

[65] W. Tromm, J. Foit, and D. Magallon. Dry and wet spreading experiments with prototypic material at the FARO facility and theoretical analysis. *Wiss. Ber. FZKA 6475*, pages 178–188, 2000.

[66] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y. Jan. A front-tracking method for the computations of multiphase flow. *Journal of Computational Physics*, 169(2):708–759, 2001.

[67] O. Ubbink. *Numerical Prediction of Two Fluid Systems with Sharp Interfaces*. PhD thesis, Imperial College London, 1997.

[68] O. Ubbink and R. Issa. A Method for Capturing Sharp Fluid Interfaces on Arbitrary Meshes. *Journal of Computational Physics*, 153:26–50, 1999.

[69] G. Urbain. Viscosity estimation of slags. *Steel Research*, 58(3):111–116, March 1987.

[70] B. van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined second-order scheme. *Journal of Computational Physics*, 14:361–370, 1974.

[71] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education, 2 edition, 2007.

[72] J. Veteau and R. Wittmaack. CORINE experiments and theoretical modelling. *in: G. Van Goetem, W. Balz, E. Della Loggia (Eds.), FISA 95 EU Research on severe accidents, Office Official Publ. Europ. Communities, Luxembourg*, pages 271– 285, 1996.

[73] D. Vola, L. Boscardin, and J. Latché. Laminar unsteady flows of Bingham fluids: A numerical strategy and some benchmark results. *Journal of Computational Physics*, 187 (2):441–456, May 2003.

[74] D. Vola, F. Babik, and J.-C. Latché. On a numerical strategy to compute gravity currents of non-Newtonian fluids. *Journal of Computational Physics*, 201(2):397–420, December 2004.

[75] V. R. Voller and C. R. Swaminathan. General source-based method for solidification phase change. *Numerical Heat Transfer, Part B: Fundamentals*, 19(2):175–189, 1991.

[76] V. Voller, A. Brent, and C. Prakash. The modelling of heat, mass and solute transport in solidification systems. *International Journal of Heat and Mass Transfer*, 32(9):1719–1731, September 1989.

[77] R. Wittmaack. Corflow: A Code for the Numerical Simulation of Free-Surface Flow. *Nuclear Technology*, 119(2):158–180, August 1997.

[78] R. Wittmaack. Simulation of Free-Surface Flows with Heat Transfer and Phase Transitions and Application to Corium Spreading in the EPR. *Nuclear Technology*, 137(3):194–212, March 2002.

[79] I.-S. Ye, J. A. Kim, C. Ryu, K. S. Ha, H. Y. Kim, and J. Song. Numerical investigation of the spreading and heat transfer characteristics of ex-vessel core melt. *Nuclear Engineering and Technology*, 45(1):21–28, 2013.