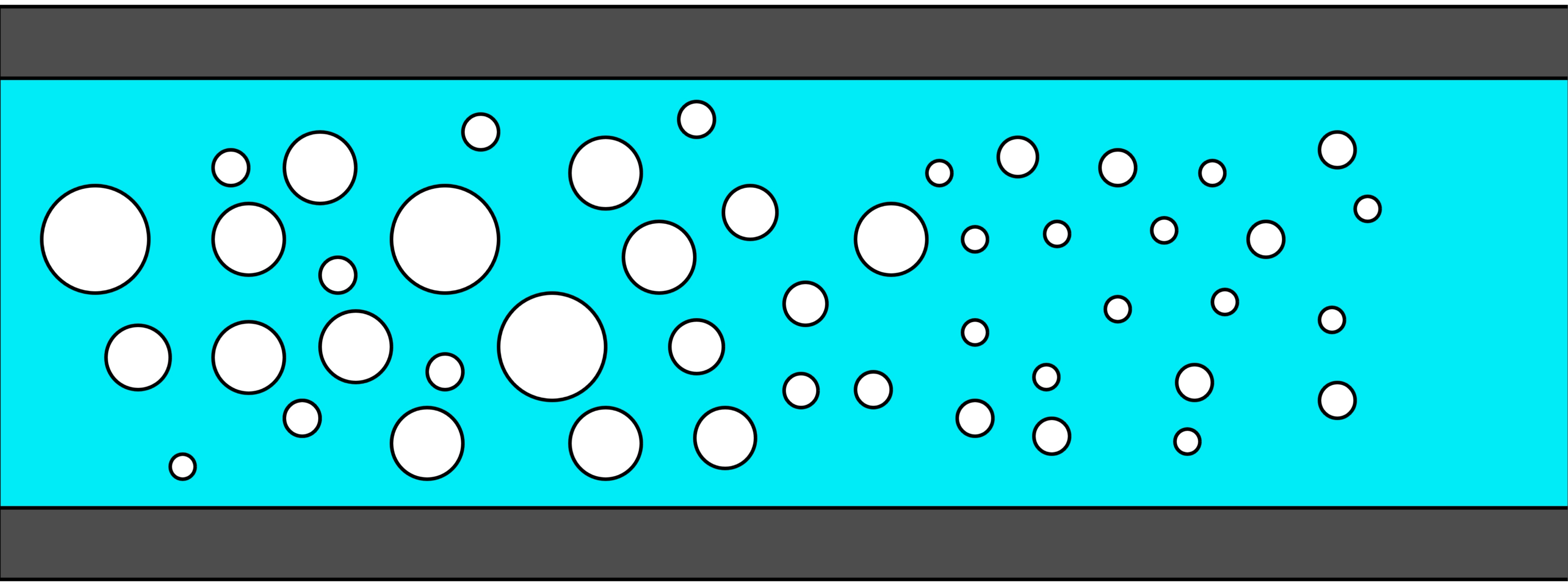
Modelling Carbon Dioxide Depressurization with Different Two-Phase Models

Master of Science Thesis
Aerospace Engineering

Cristian Bogdan Andrei Cotovanu

Delft University of Technology



Delft University of Technology

Modelling Carbon Dioxide Depressurization with Different Two-Phase Models

by

Cristian Bogdan Andrei Cotovanu

In partial fulfilment of the requirements for the degree of

Master of Science
in Aerodynamics

at the Delft University of Technology
to be defended publicly on the 10th of February, 2025

Student number:	4644840	
Project duration:	May 1, 2024 – January 26, 2025	
Thesis committee:	Prof. dr. ir. R. A. W. M. Henkes,	Shell/TU Delft, supervisor
	Dr. ir. F. F. J. Schrijer,	TU Delft, supervisor
	Dr. ir. B. W. van Oudheusden,	TU Delft, chairman
	Dr. ir. M. I. Gerritsma,	TU Delft, examiner

TU Delft
Delft
University of
Technology

Preface

Working towards my master thesis may have been the most challenging attempt that I have ever done. Many people have helped me along the road, and to all of them I would like to express my utmost gratitude.

Firstly, I would like to say a very big thank you to my supervisor from Shell, Prof.dr.ir Ruud Henkes, for all the support and encouragement he gave me during the entire research at Shell. You helped me enormously, putting a lot of time and effort into me including, but not limited to, assisting me with programming. Without your guidance and willingness to impart your knowledge, this master thesis would not have been possible.

Many thanks also to my supervisor from TU Delft, Dr.ir. Ferry Schrijer, who helped me during the nine months of my research with his vast experience. You helped me a lot with discussions and constructive comments, and inspired me to exceed my own limits.

My thanks also go out to MSc Pardeep Kumar. You were immensely helpful in explaining a variety of topics, and I always loved the discussions we had together. Many thanks for giving me your thermodynamics code, and I wish you the best of luck with your PhD project.

I thank all the colleagues from Shell for their sharing their experience and knowledge to me. They taught me how to be a good colleague, and excel in a thriving and a prestigious company.

A big thanks goes to all my friends, who supported me along the way. Although it would be impossible to list every name, some of those that come to mind include Maximilian, Thomas, Eltjo, Adriaan, Chris, Marco, Jostijn, Max, my old roommates from Huize Lamaaz and of course my friends from Stoep.

Lastly, I would also like to say a heartful thank you to my family for always believing in me and encouraging to follow my dreams. To my Mum and Dad who raised me with a love of science and supported me in all my pursuits, and to my brother, who I could always rely on and would push me to the best version of myself.

Cristian Bogdan Andrei Cotovanu
Delft, December 2024

Abstract

Carbon capture and storage (CCS) has emerged as a cornerstone technology for mitigating climate change, as highlighted by the Intergovernmental Panel on Climate Change (IPCC) and International Energy Agency (IEA). A key component of CCS involves the safe and efficient transport of CO_2 between capture sites and storage facilities, often over long distances. Pipelines can be an attractive mode of transport, with CO_2 typically maintained in a liquid or supercritical state to optimize density and viscosity. However, rapid depressurization events, whether due to intentional releases or accidental pipeline ruptures, lead to complex two-phase flow dynamics. Accurately modelling these phenomena is critical to designing pipelines that are both safe and cost-effective.

This thesis investigates various one-dimensional, unsteady, compressible two-phase flow models to simulate CO_2 depressurization scenarios in pipelines. All models are implemented using a finite volume method and discretization is done with the HLLC approximate Riemann solver, enabling the resolution of shocks and discontinuities. Thermodynamic properties are computed using equations of state (EOS), with a focus on the Span-Wagner (SW) EOS, to capture the CO_2 unique phase behaviour under high-pressure conditions. The models are validated against existing experimental data from SINTEF's depressurization facility, which provided high-resolution measurements of pressure and temperature during rapid phase transitions.

Three different models are initially looked at: $DF3$, $DF4$, and $TF5$. Key findings demonstrate that while the $DF3$ model provides accurate predictions of pressure variations over time, it underestimates the initial pressure drop by up to 20 $[bar]$. In contrast, the $DF4$ model, through the manual adjustment of the mass transfer term (Γ) using the relaxation parameter θ, more accurately captures the initial transient behaviour, aligning closely with experimental data. Both models ultimately converge to approximately the same state after 20 $[ms]$, with no more than 4 $[bar]$ difference between the models.

Temperature predictions, however, pose a greater challenge. The $DF3$ model exhibits a spurious downward temperature spike immediately following pipeline rupture, while the $DF4$ model predicts an even greater initial temperature drop, neither of which align with experimental observations. The delayed cooling effect observed in experiments is not captured by either model, highlighting limitations in the energy equation and the need for additional source terms to account for temperature drop delays.

Limitations of the models include challenges in simulating temperature variations near the critical point and an inability to accurately model delayed temperature effects for both the $DF3$ and $DF4$ models. Recommendations for future work include developing a generalized mass transfer model, incorporating implicit numerical schemes for stability near critical conditions. Furthermore, the $TF5$ model shows promise for improving temperature predictions over longer timescales.

Contents

Nomenclature

Latin letters

A	Area	m^2
a	Specific Helmholtz free energy	J
c	Speed of sound	$\mathrm{m\ s^{-1}}$
$\tilde{C}_p$	Component heat capacity at constant pressure	$\mathrm{J\ K^{-1}\ m^{-3}}$
c_p	Specific heat capacity at constant pressure	$\mathrm{J\ K^{-1}\ kg^{-1}}$
c_v	Specific heat capacity at constant volume	$\mathrm{J\ K^{-1}\ kg^{-1}}$
E	Total energy	$\mathrm{J\ m^{-3}}$
e	Specific internal energy	$\mathrm{J\ kg^{-1}}$
$\mathscr{F}$	Friction force	$\mathrm{N\ m^{-3}}$
$\mathbf{F}$	Flux vector	-
F	Force	N
f	Friction factor	-
f	Function	-
G	Gibbs free energy	J
g	Gravitational acceleration	$\mathrm{m\ s^{-2}}$
h	Specific enthalpy	$\mathrm{J\ kg^{-1}}$
M	Mass	kg
$\dot{m}$	Mass-flow rate	$\mathrm{kg\ s^{-1}}$
N	Total number of control volumes	-
$\boldsymbol{n}$	Normal unit outward-facing on the boundary	-
p	Pressure	Pa
s	Specific entropy	$\mathrm{J\ kg^{-1}}$
S	Speed of wave	$\mathrm{m\ s^{-1}}$
$\mathbf{S}$	Source term vector	-
T	Temperature	K
$\mathbf{U}$	State vector	-
V	Volume	m^3
v	Velocity	$\mathrm{m\ s^{-1}}$
x	Spatial location	m
Y	Mass fraction	$\mathrm{m^3\ m^{-3}}$
y	Spatial location	m
z	Spatial location	m

Greek letters		
α	Volume fraction	$m^3\ m^{-3}$
$\boldsymbol{\Lambda}$	Eigenvalue matrix	-
λ	Eigenvalue	-
δ	Inverse reduced density	-
ξ	Relaxation coefficient	-
ε	Pipe mean roughness	m
Γ	Phase-transfer rate	$kg\ m^{-3}\ s^{-1}$
γ	Grüneisen parameter	$Pa\ m^3\ J^{-1}$
ρ	Density	$kg\ m^{-3}$
δ	Angle with the horizontal axis	rad
μ	Chemical potential	$J\ kg^{-1}$
η	Dynamic viscosity	$N\ s\ m^{-2}$
ψ	Conserved variable	-
σ	Surface tension	$N\ m^{-1}$
∇	Divergent	-
τ	Inverse reduced temperature	-

Subscripts	
c	Critical condition
E	Energy equation source term
g	Gas/vapour phase
i	Spatial discretization
int	Interfacial
K	Left or Right state
k	Gas or Liquid phase
l	Liquid phase
M	Momentum equation source term
sat	Saturation
t	Time derivative

Superscripts	
$HLLC$	Reference state for HLLC solver
n	Time-step discretization
$*$	Reference state for HLLC solver

Dimensionless numbers	
CFL	Courant-Friedrichs Lewy number
Fr	Froude number
Re	Reynolds number
We	Weber number

Abbreviations	
BC	Boundary condition
BN	Baer-Nunziato (model)
CCS	CO_2 capture and storage
CFD	Computational fluid dynamics
$DF3$	Three-equation drift-flux (model)
$DF4$	Four-equation drift-flux (model)
EOS	Equation of state
FAD	Forward automatic differentiation
FVM	Finite volume method
HEM	Homogeneous equilibrium model
HLLC	Harten-Lax-van Leer-Contact
IEA	International Energy Agency
IC	Initial condition
IPCC	Intergovernmental Panel on Climate Change
ODE	Ordinary differential equation
PDE	Partial differential equation
PR	Peng-Robinson
SW	Span-Wagner
$TF5$	Five-equation Two-phase (model)

1
Introduction

Carbon capture and storage (CCS) is increasingly recognized as a viable solution to mitigate the climate crisis, as highlighted by the IPCC [1]. According to the IEA, achieving net-zero emissions by 2050 requires the annual capture and storage of several gigatonnes of CO_2 [2]. One critical aspect of CCS is the safe and efficient transport of CO_2 between capture plants and storage facilities, which are often separated by large distances. This transport is typically conducted via pipelines, ships, or other methods. To optimize transport efficiency, CO_2 is maintained in a liquid phase at supercritical pressures, ensuring high density and low viscosity [3].

However, rapid depressurization events, such as intentional releases (e.g., opening pressure relief valves or injecting CO_2 into empty gas fields) or accidental scenarios (e.g., pipe fractures), can lead to significant changes in pressure, temperature, sound speed and mass flow. During such events, the CO_2 fluid undergoes a phase transformation into a two-phase regime, introducing complex flow dynamics and thermodynamic behaviour. Accurate modelling of these processes is essential, particularly given the potential for cost savings by repurposing existing pipelines for CO_2 transport.

Despite substantial prior research on two-fluid modelling, simulating CO_2 with high accuracy remains a challenge, due to its unique phase behaviour and thermodynamic properties. Consequently, numerical techniques for modelling CO_2 flow in pipelines are actively being developed by the energy sector, including Shell. In this master thesis, various models are constructed and compared, with validation against experimental data from SINTEF [4] [5] [6].

1.1. Problem description

As previously mentioned, CO_2 flow can transition into a two-phase regime, significantly increasing the complexity of simulations. Accurate modelling is essential for designing pipelines that are both cost-efficient and safe for CO_2 transportation. The materials used in pipeline construction are highly sensitive to temperature variations. If simulations predict lower temperatures than those actually experienced by the fluid, the pipeline will be over-designed, resulting in unnecessary additional costs. Conversely, if simulations overestimate the temperature, the pipeline will be under-designed and more susceptible to rupture. This presents a significant hazard, as ruptures can propagate along the pipeline in a phenomenon known as running-ductile fracture. A running-ductile fracture can cause extensive damage to large portions of the pipeline, as illustrated in Figure 1.1. The energy released during such an event is substantial, especially given that the pressure in pipelines transporting CO_2 can reach up to 160 bar [3]. Such failures not only pose serious safety risks but also result in considerable economic and environmental consequences. Accurate simulations are therefore critical to ensure that pipelines are designed to withstand these extreme conditions while minimising costs.

The challenge with simulating two-phase CO_2 flow lies in accounting for non-equilibrium effects. The most accessible method for simulating CO_2 flow is by employing the one-dimensional unsteady Euler flow equations and assuming bulk quantities across the cross-section of the pipeline. However, this approach overlooks critical non-equilibrium effects, including pressure undershoots during rapid depressurisation, delays in temperature changes, and differing velocities for the gas and liquid phases in stratified flow. These effects are crucial for accurately capturing the dynamics of two-phase CO_2 flow and ensuring reliable simulation results.

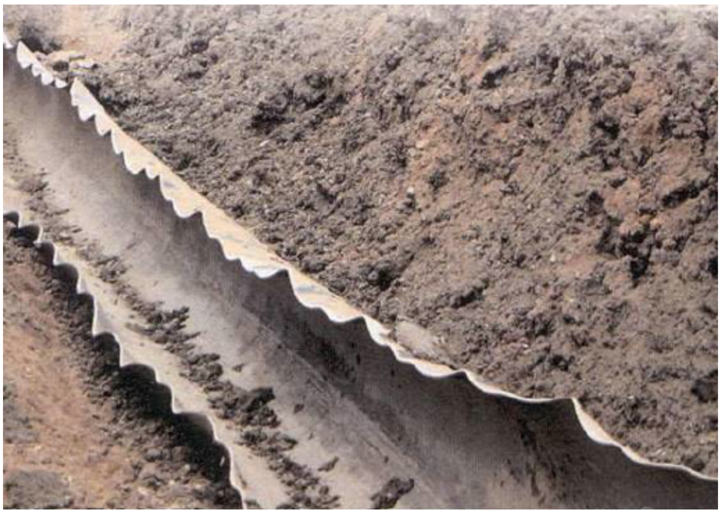


Figure 1.1: Simulation of a pipeline ductile fracture. In the simulation, there is an initial rupture and this gives a snowball effect that propagates the rupture over a large part of the pipeline. Source: Istre [7]

1.2. Present contribution

This study aims to simulate two-phase CO_2 flow in a pipeline using various two-phase flow models and to compare these models to identify their shortcomings. The main research question is:

- *What are the differences and shortcomings of various flow models in simulation of CO_2 flow in a pipeline, and how can CO_2 flow be predicted more accurately?*

Several models were implemented and tested in this work. The simplest models, the HEM and $DF3$ model, which are explained in subsection 3.1.1, serves as a foundational approach to understanding and developing the simulation framework. This model was compared to the solution by OLGA, a state-of-the-art simulation tool for 1D multiphase flow, in which also the HEM and $DF3$ models are used.

Furthermore, the $DF4$ model, explained in subsection 3.1.3, was discretized with the HLLC method and evaluated. Here, the results are compared to the $DF3$ model, whereby the difference in mass transfer is evaluated and the physical meaning behind it is discussed. Additionally, different methods for calculating the mass transfer are incorporated, with a discussion on what the implementation of the different mass transfer correlations mean. Lastly, the $TF5$ model, explained in subsection 3.1.5, is discretized and used. It proved time-wise to be too optimistic to simulate the $TF5$ model with CO_2 in this study. Nevertheless, the model's theoretical framework provides valuable insights into its potential applications.

1.3. Thesis structure

This thesis is divided into three main parts: *Theory* (chapters 2, 3, 4 and 5), *Implementation* (chapters 6 and 7), and *Results* (chapters 8, 9 and 10).

Chapter 1 introduces the motivation and context for this research.

Chapter 2 reviews existing research, derives the physical modelling of CO_2 flow using the Euler equations, identifies gaps in the literature, and formulates the research questions and sub-questions.

In chapter 3, it is explained how the different flow models have been established, what additional source terms are needed and how these models are simulated with a Riemann setup. Lastly, a discussion is given on the mixture speed of sound of the models.

The thermodynamics of CO_2 are described in chapter 4. It is briefly discussed what the Span-Wagner and Peng-Robinson equation of states are. In addition, an explanation is given on how all primitive variables are calculated.

In chapter 5, the numerics and discretization of the flow models are discussed. This means using the FVM (Finite Volume Method) and the HLLC (Harten, Lax, van Leer, Contact) approximate Riemann solver. In addition, it is shortly discussed how ODEs (Ordinary Differential Equations) are discretized for the thermodynamics.

Chapter 6 provides the details of implementing the numerics into the code. Moreover, difficulties are explained and how to handle these in the code.

For an adequate implementation, the verification is crucial and this is addressed in chapter 7.

In chapter 8, all the results and outcomes of the models are presented.

In chapter 9, a discussion is presented of the observed physical implementations. Furthermore, the research questions are answered.

Lastly, chapter 10 presents the conclusions of this research, and gives further recommendations for future studies.

I

THEORY

2
Literature Review

In this chapter, a literature review on the background of two-phase flow simulation for pure CO_2 fluid is presented. The aim is to establish a foundation for the remainder of this study by identifying research gaps and, ultimately, by justifying the research question introduced in chapter 1.

This chapter begins with an overview of two-phase flow in section 2.1, explaining the basic concepts and importance of accurately modelling liquid and gas phases. In addition, in section 2.1, it is discussed what the various two-phase models commonly found in the literature are, and their relevance to CO_2 transport. Particular attention is given to the assumptions that underpin each model and how they influence accuracy and computational complexity. It is then examined how these models can be simplified through assumptions regarding pressure, velocity, temperature, and chemical potential.

The discussion then turns to the choice of the EOS (Equation Of State). In section 2.2, the emphasis is placed on the choice of an appropriate EOS for two-phase flow simulation of pure CO_2, outlining how an unsuitable EOS can undermine flow predictions. Several commonly used EOSs, namely Peng-Robinson, Stiffened Gas, and Span-Wagner, are compared demonstrating the trade-offs between computational efficiency and physical accuracy. In addition, section 2.3 addresses numerical discretisation, focusing on hyperbolicity and the selection of a reliable Riemann solver, namely the HLLC scheme, to manage shocks effectively.

Following this, the chapter highlights a key gap in the literature: although multiple authors have derived and simplified two-phase flow equations, especially leading to the so-called $TF5$ model, there appears to be no documented application of the $TF5$ model for pure CO_2 flow. The potential benefits of this more advanced model remain largely unexplored in the context of rapid depressurization of CO_2 flow.

Finally, the essential experimental framework is introduced in section 2.5. This involves describing the CO_2 depressurisation laboratory facility at ECCSEL in Norway, the test conditions, and the instrumentation used for validating two-phase flow models.

2.1. General two-phase flow model

As stated earlier, CO_2 fluid can exist in a state of single-phase flow or two-phase flow. In single-phase flow, all the CO_2 is in a gaseous, liquid, or supercritical state. If one considers a cross-section of the pipeline, the assumption is that the fluid in that cross-section is homogenous, with uniform properties across the phase.

In a two-phase flow regime, however, a pipeline cross-section can simultaneously contain liquid and gaseous phases. Still, it is assumed that the properties within each phase remain uniform. This means that in each cross-section, there is a property for the liquid phase, and a property for the gaseous phase (such as liquid and gaseous density). In Figure 2.1, a cross-sectional area of a pipeline is illustrated: three distinct two-phase flow regimes are seen. In these examples, the blue area represents the liquid phase, and the white area represents the gaseous phase. Note that these are only three of many possible flow regimes.

The proportion of each phase within a cross-section is expressed as the volume fraction, denoted as α. For instance, in the rightmost cross-section of Figure 2.1, the liquid occupies approximately one-third of the total area, while the gas occupies two-thirds. This corresponds to $\alpha_g = \frac{2}{3}$ and $\alpha_l = \frac{1}{3}$. The sum of the volume fractions always need to add up to one, see Equation 2.1. These volume fractions form the foundation for constructing one-dimensional two-phase flow models.

$$\alpha_g + \alpha_l = 1. \tag{2.1}$$

The most general and comprehensive model is the Baer-Nunziato model [8], $BN7$ model, named after the two authors who developed the model. This model is based on the one-dimensional unsteady compressible Euler equations for fluid dynamics and consists of seven equations: two mass conservation equations (one for each phase), two momentum equations (one for each phase), two energy equations (one for each phase), and a compaction law that governs the phase interaction. The compaction law describes how the volume fraction of one phase changes in response to pressure variations, enabling the model to account for interfacial dynamics. Each phase has its own set of properties, including pressure, velocity, temperature, and chemical potential (p_g, p_l, v_g, v_l, T_g, T_l, μ_g, μ_l). Although this model was originally developed for a solid-gas mixture for detonation waves in granular explosives, it is also possible to use it for liquid-gas mixtures [8] [9] [10].

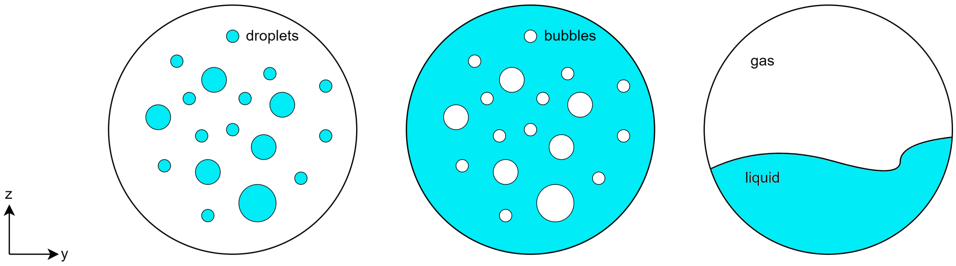


Figure 2.1: Cross sectional area of a pipeline with three different two-phase flow regimes. The blue area indicates where the flow is in a liquid phase, and the white area indicates where the flow is in a gaseous phase.

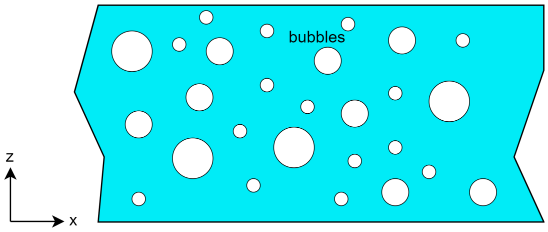


Figure 2.2: Cross sectional area of a pipeline in the longitudinal direction. The flow is in a two-phase regime with liquid carrying gas bubbles.

Nevertheless, the $BN7$ model has not yet been applied to simulate CO_2 two-phase flow. This is primarily due to its complexity and the numerical challenges it presents, as its hyperbolicity is not guaranteed. As a result, oscillations can arise in the solution, as discussed by Dallet [11] and Andrianov & Warnecke [12]. Additionally, no validation of the model for a gas-liquid scenario has been documented. The only validation of the $BN7$ model appears in the original paper by Baer & Nunziato [8], where it was tested against experimental data for solid-gas mixtures. The results were in good agreement with the experimental data, showcasing its potential for capturing interphase dynamics. Despite its robustness, the complexity of the $BN7$ model has motivated researchers to derive simplified models. For instance, Lund et al. [13], Martinez et al. [14], Morin et al. [15], and Saurel et al. [16] have built upon the $BN7$ framework to create less computationally demanding models.

2.1.1. Two-phase Euler equations

This derivation section explains how the $BN7$ model is developed. The derivation done below is based on the book of Toro, chapter 1 [17].

Imagine a control volume, where a fluid element has a differential volume dV. It is known that there are three fundamental physical principles: the conservation of mass, momentum and energy. Then integrated over a three-dimensional domain, it becomes:

$$\int_{\mathscr{V}} \rho_t + \nabla\cdot(\rho v) dV = \frac{\partial}{\partial t}\int_{\mathscr{V}} \rho dV + \int_{\mathscr{V}} \nabla\cdot(\rho v) dV = 0, \tag{2.2}$$

$$\int_{\mathscr{V}} (\rho v)_t + \nabla\cdot(\rho v^2 + p) dV = \frac{\partial}{\partial t}\int_{\mathscr{V}} (\rho v) dV + \int_{\mathscr{V}} \nabla\cdot(\rho v^2) dV + \int_{\mathscr{V}} \nabla\cdot p dV = 0, \tag{2.3}$$

$$\int_{\mathscr{V}} E_t + \nabla\cdot(Ev + pv) dV = \frac{\partial}{\partial t}\int_{\mathscr{V}} E dV + \int_{\mathscr{V}} \nabla\cdot(Ev) dV + \int_{\mathscr{V}} \nabla\cdot(pv) dV = 0. \tag{2.4}$$

Now, the Divergence Theorem states that for a vector field ($\mathbf{F}$) defined on a region ($\mathscr{V}$) with a closed surface ($\mathscr{A}$) (which is the boundary of ($\mathscr{V}$)), the following relationship holds:

$$\iint_{\mathscr{A}} \mathbf{F}\cdot \boldsymbol{n} dA = \iiint_{\mathscr{V}} (\nabla\cdot\mathbf{F})\, dV, \tag{2.5}$$

where the left hand side is the surface integral of **F** over the closed surface A, representing the total flux of **F** through A. The right hand side is the volume integral of the divergence of **F** over the volume V, representing the total divergence within V. The set of equations becomes:

$$\frac{\partial}{\partial t}\int_{\mathscr{V}} \rho dV + \int_{\mathscr{A}} (\rho v)\cdot \boldsymbol{n} dA = 0, \tag{2.6}$$

$$\frac{\partial}{\partial t}\int_{\mathscr{V}} (\rho v) dV + \int_{\mathscr{A}} (\rho v^2)\cdot \boldsymbol{n} dA + \int_{\mathscr{A}} p\cdot \boldsymbol{n} dA = 0, \tag{2.7}$$

$$\frac{\partial}{\partial t}\int_{\mathscr{V}} E dV + \int_{\mathscr{A}} (Ev)\cdot \boldsymbol{n} dA + \int_{\mathscr{A}} (pv)\cdot \boldsymbol{n} dA = 0. \tag{2.8}$$

Equations (2.6)–(2.8) are known as the bulk equations, as they contain bulk quantities of both phases. The volume part of the equations state the total quantity (mass, momentum and energy), and the surface part of the equations state the flux through the volume (mass flux, momentum flux and energy flux). In addition, the momentum and energy equations have an additional component that is produced by the pressure. This is the pressure force exerted on the volume for the momentum, and the rate of work done by the pressure force on the volume for the energy. It is also possible to write a separate mass, momentum and energy equation for each fluid. This is the following set:

$$\frac{\partial}{\partial t}\int_{V_{k(t)}} \rho_k dV_k + \int_{A_{k(t)}} \Big(\rho_k(v - v_{A_k})\Big)\cdot \boldsymbol{n} dA_k = \int_{\mathscr{V}} \Gamma_k dV, \tag{2.9}$$

$$\frac{\partial}{\partial t}\int_{V_{k(t)}} (\rho_k v) dV_k + \int_{A_{k(t)}} \Big(\rho_k v(v - v_{A_k})\Big)\cdot \boldsymbol{n} dA_k + \int_{A_{k(t)}} p_k\cdot \boldsymbol{n} dA_k = \int_{\mathscr{V}} S_{M,k} dV, \tag{2.10}$$

$$\frac{\partial}{\partial t}\int_{V_{k(t)}} E_k dV_k + \int_{A_{k(t)}} \Big(E_k(v - v_{A_k})\Big)\cdot \boldsymbol{n} dA_k + \int_{A_{k(t)}} (p_k(v - v_{A_k})\cdot \boldsymbol{n} dA_k = \int_{\mathscr{V}} S_{E,k} dV. \tag{2.11}$$

In these equations, the subscript k indicates the phase, where $k = l$ denotes the liquid phase and $k = g$ denotes the gaseous phase. Thus, equations Equation 2.9 through Equation 2.11 comprise two distinct sets: one for the liquid phase (l) and one for the gaseous phase (g). In Equation 2.9 to Equation 2.11, the challenge arises from the time-dependent changes in volume ($V_k(t)$) and surface area ($A_k(t)$). This is attributed to the moving interface between the phases, which progresses with a velocity denoted as v_{A_k}. This issue can be addressed by introducing the volume fraction, as defined in Equation 2.12.

$$\alpha_k = \frac{V_k}{V_l + V_g} = \frac{V_k}{V}. \tag{2.12}$$

By employing this variable, the integration can once again be carried out from $V_k(t)$ to over the constant volume V. v_{A_k} becomes zero again, as the integration is now done over a fixed domain. For the flux terms, the divergence theorem is applied to express them over the volumes, resulting in the following form:

$$\frac{\partial}{\partial t}\int_V (\alpha_k\rho_k)dV + \int_V \nabla(\alpha_k\rho_k v_k)dV = \int_V \Gamma_k dV, \tag{2.13}$$

$$\frac{\partial}{\partial t}\int_V (\alpha_k\rho_k v_k)dV + \int_V \nabla(\alpha_k\rho_k v_k^2)dV + \int_V \nabla(\alpha_k p_k)dV = \int_V S_M dV, \tag{2.14}$$

$$\frac{\partial}{\partial t}\int_V (\alpha_k E_k)dV + \int_V \nabla(\alpha_k E_k v_k)dV + \int_V \nabla(\alpha_k p_k v_k)dV = \int_V S_E dV, \tag{2.15}$$

$$\frac{\partial}{\partial t}\int_V \alpha_g dV + v_{int}\int_V \nabla\alpha_g dV = \int_V \xi_p(p_g - p_l)dV. \tag{2.16}$$

In addition to separate mass, momentum, and energy equations, the model also includes a topology equation describing the evolution of the volume, Equation 2.16. This is the result of changing from the unknown moving domain $V_k(t)$, into the known fixed domain V. Since the liquid volume fraction is simply one minus the gas volume fraction, only one topology equation is required. Finally, the system of equations are converted into a one-dimensional differential form, facilitating numerical discretization and simulation. The resulting seven equations are the following:

$$\frac{\partial(\alpha_g\rho_g)}{\partial t} + \frac{\partial(\alpha_g\rho_g v_g)}{\partial x} = \Gamma, \tag{2.17}$$

$$\frac{\partial(\alpha_l\rho_l)}{\partial t} + \frac{\partial(\alpha_l\rho_l v_l)}{\partial x} = -\Gamma, \tag{2.18}$$

$$\frac{\partial(\alpha_g\rho_g v_g)}{\partial t} + \frac{\partial(\alpha_g\rho_g v_g^2)}{\partial x} + \frac{\partial(\alpha_g p_g)}{\partial x} = \mathbf{S}_M, \tag{2.19}$$

$$\frac{\partial(\alpha_l\rho_l v_l)}{\partial t} + \frac{\partial(\alpha_l\rho_l v_l^2)}{\partial x} + \frac{\partial(\alpha_l p_l)}{\partial x} = -\mathbf{S}_M, \tag{2.20}$$

$$\frac{\partial(\alpha_g E_g)}{\partial t} + \frac{\partial(\alpha_g E_g v_g)}{\partial x} + \frac{\partial(\alpha_g p_g v_g)}{\partial x} = \mathbf{S}_E, \tag{2.21}$$

$$\frac{\partial(\alpha_l E_l)}{\partial t} + \frac{\partial(\alpha_l E_l v_l)}{\partial x} + \frac{\partial(\alpha_l p_l v_l)}{\partial x} = -\mathbf{S}_E, \tag{2.22}$$

$$\frac{\partial\alpha_g}{\partial t} + v_{int}\frac{\partial\alpha_g}{\partial x} = \xi(p_g - p_l). \tag{2.23}$$

These set of equations is the $BN7$ model. It has a separate mass, momentum and energy equation for each phase and as mentioned before, an additional topology equation, being Equation 2.23. By introducing simplifying assumptions, the model can be reduced to less equations:

1. **Equal Pressure Assumption**:
   Assuming equal pressures for both phases, i.e. ($p_g = p_l = p$), the topology equation, Equation 2.23, is no longer required. This reduction results in the two-phase 6 model ($TF6$), with six equations spanning Equation 2.17 to Equation 2.22.

2. **Equal Velocity Assumption**:
   A second assumption is to equal the velocities for both phases, i.e. ($v_g = v_l = v$). With this assumption, one of the momentum equations can be replaced by a bulk momentum equation, obtained by summing Equation 2.19 and Equation 2.20. The resulting model is the two-phase 5 model ($TF5$), which includes separate mass and energy conservation equations for each phase, Equation 2.17, Equation 2.18, Equation 2.21 and Equation 2.22, alongside the bulk momentum equation, Equation 2.24.

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2)}{\partial x} + \frac{\partial(p)}{\partial x} = 0. \tag{2.24}$$

3. **Equal Temperature Assumption**:
   Furthermore, assuming equal temperatures for both phases ($T_g = T_l = T$), the energy equations can be combined into a bulk energy equation:

$$\frac{\partial E}{\partial t} + \frac{\partial(Ev)}{\partial x} + \frac{\partial(pv)}{\partial x} = 0. \tag{2.25}$$

This yields the drift-flux 4 model ($DF4$), which includes separate mass conservation equations for each phase, Equation 2.17 and Equation 2.18, a bulk momentum equation, Equation 2.24, and a bulk energy equation,Equation 2.25.

4. **Equal Chemical Potential Assumption**:
   Lastly, by assuming equal chemical potentials for both phases ($\mu_g = \mu_l = \mu$), the mass transfer terms in Equation 2.17 and Equation 2.18 are combined, resulting in a bulk mass equation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0. \tag{2.26}$$

Combining assumptions 1, 2, 3 and 4 gives the homogeneous equilibrium model (HEM), consisting of three bulk equations: Equation 2.24, Equation 2.25, and Equation 2.26. If source terms, such as friction, are included in any equation, the model becomes the drift-flux 3 model ($DF3$). Figure 2.3 shows the models with their respectively assumptions in a flowchart.

The path taken to arrive at the HEM model is not unique. An example is given in Figure 2.4. The $DF4-a$ model consists of a bulk mass conservation equation, two distinct momentum conservation equations and a bulk energy conservation equation. The reason for taking the path of Figure 2.3 will be examined in subsection 2.1.2.

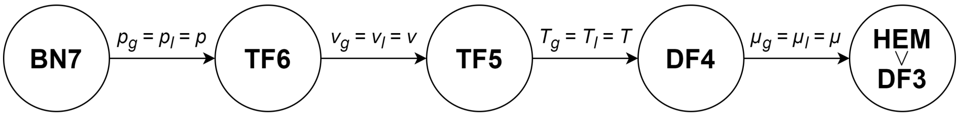


Figure 2.3: Flowchart of the various models and how to obtain these models by assuming different parameters to be identical.

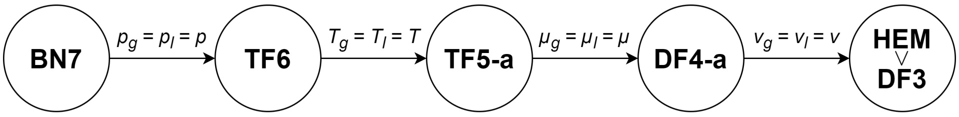


Figure 2.4: Flowchart of the various models and how to obtain these models by assuming different parameters to be identical.

2.1.2. Simplifications

As discussed above, various assumptions can be made to simplify the original $BN7$ model. The key question is which assumptions are valid. A first reasonable assumption is the equality of pressures between the phases. This assumption is based on the fact that pressure adjustments between two phases occur over a much shorter timescale compared to other properties. Kapila et al. [18] argue that, in comparison to temperature variations, the timescale for pressure changes is approximately 100 times smaller. This suggests that a single pressure for both phases is a plausible assumption, and that both phases reach the same pressure almost instantaneously. As Kapila et al. [18] state: "This is one reason why standard two-phase fluid models assume pressure equilibrium."

Further supporting this, Guo et al. [19] conducted CO_2 depressurization experiments in a horizontal pipeline. These experiments, which included both dense-phase and gaseous-phase starting conditions, demonstrated that the pressure was uniform across the top and bottom of the pipeline. In contrast, temperature differences were observed, with a larger temperature difference near the rupture location, which diminished farther from it. This suggests that the relaxation time for temperature changes is longer than for pressure.

A second reasonable assumption is that the velocities of both phases are in equilibrium. Hammer et al. [20] modelled CO_2 flow in a vertical pipeline and validated their model with experiments. They assumed identical properties for both phases: pressure, velocity, temperature, and chemical potential, i.e., $p_g = p_l = p$, $v_g = v_l = v$, $T_g = T_l = T$, and $\mu_g = \mu_l = \mu$. By including friction as a source term, this model corresponds to the $DF3$ model, rather than the HEM model. Hammer et al. also simulated the experiments where the assumption of identical velocities was relaxed, leading to the $DF4-a$ model shown in Figure 2.4. Three different

$DF4 - a$ models were used, which differed by their choice of source terms for the friction in the momentum equations. Their conclusion was that the $DF3$ model provided the most accurate results. As Hammer et al. stated: "The experimental results indicate that the flow is close to no-slip, within the experimental uncertainty." It is important to note that these results were obtained for a vertical pipeline.

Again, the same paper of Guo et al. [19] is used to argue that the equality in velocity is reasonable. In the gaseous-phase tests, the flow underwent depressurization, and upon reaching saturation conditions, part of the flow transitioned into liquid droplets, as shown in the leftmost cross-section of Figure 2.1. In the dense-phase tests, depressurization led to the formation of gas bubbles, as illustrated in the middle of Figure 2.1. In both scenarios, the CO_2 fluid phases were well-mixed within the pipeline cross-section. When only droplets or bubbles are present, the two phases flow together, and the velocities remain the same. This phenomenon is illustrated in Figure 2.2. Here, the fluid velocity will be in the x-direction. The bubbles are surrounded by the liquid, and therefore, are transported with the liquid in the same direction. When the bubbles are small, they travel with the same velocity as the liquid, being a no-slip condition. This principle applies to both gas bubbles in liquid and liquid droplets in gas, providing a strong rationale for assuming that both phases share the same velocity in a well-mixed flow.

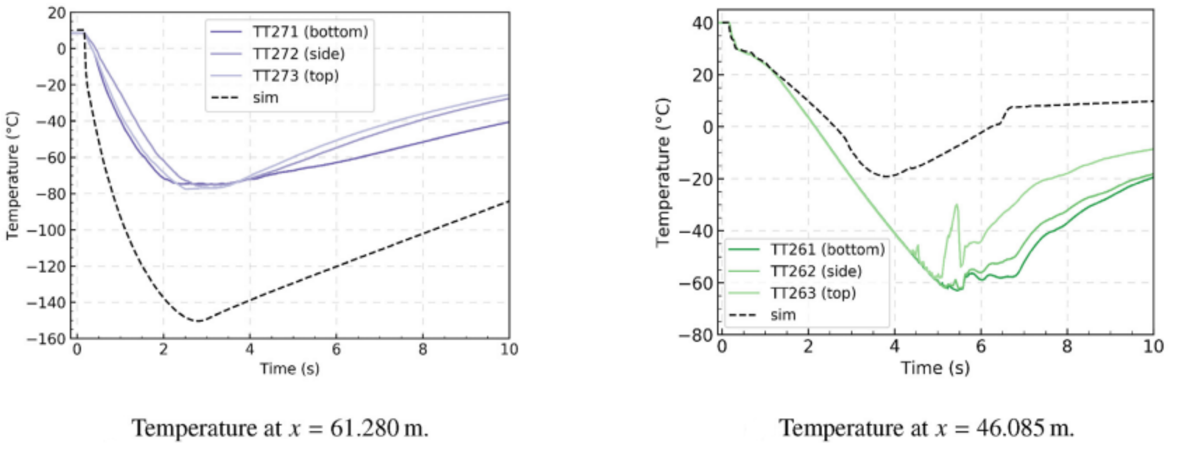
Under low flow conditions, a clear phase separation can occur, as seen in the rightmost cross-section in Figure 2.1. Here the top portion of the pipeline contains the gas phase, while the bottom portion contains the liquid phase. This is a so-called stratified flow regime. This phase separation implies that different velocities for each phase may exist, resulting in frictional interactions. This was not the situation in the tests of Guo et al.[19].

2.1.3. Flow models discrepancies

Building on the research and theories presented previously, there is substantial evidence to justify assuming that both phases share the same pressure ($p_g = p_l = p$) and velocity ($v_l = v_g = v$) in two-phase CO_2 flow in rapid depressurization situations. However, whether the chemical potential (μ) and temperature (T) can also be taken as identical for both phases remains an open question.

At the SINTEF research centre in Norway, Munkejord et al. [21] and Log et al. [22] [23] [24] carried out multiple depressurization experiments starting with CO_2 at high pressure. The initial conditions for these tests are listed in Table 2.2 and shown in Figure 2.7, which presents a pressure–temperature diagram. During these experiments, Munkejord et al. [21] observed that the CO_2 pressure dropped below the local saturation pressure before entering the two-phase regime—a phenomenon known as “pressure undershoot.” Under strictly chemical-equilibrium conditions, one would expect the pressure to reach the saturation line, transition to two-phase flow, and then follow that line in both pressure and temperature. The observed pressure undershoot indicates a degree of non-equilibrium behaviour during depressurization; an illustrative example is provided in Figure 2.6 (not based on any specific simulation or experiment). Log et al. [22] [23] [24] showed that abandoning the assumption of chemical equilibrium—by using a $DF4$ model instead yielded simulations that more closely matched the experimental data, especially during the first 100 $[ms]$ of the tests.

In addition to pressure measurements, the SINTEF experiments also recorded pipeline temperatures at several locations, including the bottom, top, and sides of the cross-section, in order to identify if different temperatures occur. Munkejord et al. [21] compared these results with simulations using the $DF3$ model, which assumes identical properties for both phases. Although this approach produced accurate results for some scenarios, certain cases showed significant discrepancies. Two such examples are given in Figure 2.5. In Figure 2.5a, the simulated temperature is notably lower than the measurements, leading to an overly conservative pipeline design (and unnecessary cost). Conversely, Figure 2.5b shows a scenario in which the simulation predicts a higher temperature than observed, posing an increased risk of pipeline failure. Furthermore, the measured data indicate that temperatures at the top and bottom of the pipeline can diverge, suggesting that assuming a single temperature for both phases is not always valid.



(a) Temperature result of the experiment and simulation of test 11 at a distance of 61.280 [m]. Source: Munkejord et al. [21].

(b) Temperature result of the experiment and simulation of test 6 at a distance of 46.085 [m]. Source: Munkejord et al. [21].

Figure 2.5: Temperature results of the $DF3$ model compared with experimental data. Source: Munkejord et al. [21].

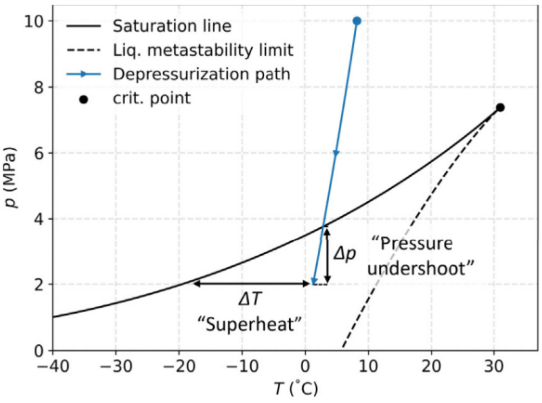


Figure 2.6: Pressure undershoot of a possible depressurization path. Source: Log et al. [22].

2.2. Equation of state

While choosing an appropriate two-phase flow model, two additional factors must be considered. The first is the choice of the equation of state (EOS). An EOS relates the state variables of a fluid, and an unsuitable EOS can lead to inaccurate predictions of the fluid properties. Although the ideal gas law is the most familiar and straightforward EOS, it often fails under high-pressure conditions and during phase transitions.

A review of the literature shows that three EOSs are most commonly employed for simulating CO_2:

- **Peng-Robinson (PR) EOS**, introduced by Peng & Robinson [25]. Originally developed for natural gas systems, it can be adapted for CO_2 by modifying standard parameters. Its algebraic cubic form makes it relatively easy to solve.
- **Stiffened Gas (SG) EOS**, formulated by Harlow & Amsden [26] and adapted for CO_2 by adjusting constants, as demonstrated by Lund et al. [13], [27]. This equation is also algebraic and is computationally efficient.
- **Span-Wagner (SW) EOS**, proposed by Span and Wagner [28] specifically for CO_2. It is considered the most accurate choice for pure CO_2 applications.

Munkejord et al. [29] reviewed the SW EOS, reporting uncertainties of 0.05% for density, 0.006% for vapour pressure, and about 1.0% for the speed of sound. Böttcher et al. [30] likewise compared the SW, SG, and PR EOS, concluding that the SW EOS provided superior performance across a wide range of conditions, a result corroborated by other studies from Hammer et al. [31], and Zhou et al. [32]. Despite its accuracy, the SW EOS can be computationally intensive—often requiring ten to twenty times more computational resources than the SG or PR EOS, as stated by Böttcher et al. [30]. Nevertheless, if computational resources permit moderate solution times, the SW EOS remains the most recommended option.

2.3. Discretization and hyperbolicity of the $TF5$ model

Discretization is a crucial step in evaluating the feasibility and convergence of two-phase flow models, as it involves examining various model attributes. One such attribute is the hyperbolicity of the partial differential equations (PDEs): a hyperbolic system ensures a unique solution for a given set of initial conditions. For the $TF5$ model, its hyperbolicity has been established by Martinez et al. [14], Munkejord et al. [33], Saurel et al. [16], and Kreeft et al. [34].

Another major consideration in discretization is selecting an appropriate Riemann solver to compute fluxes in regions featuring shocks. The Harten–Lax–van Leer contact (HLLC) scheme, introduced by Toro [17] [35], is a widely recognised and validated solver. Although it was originally formulated for the HEM model, the HLLC method has since been applied successfully to other models, including the $DF4$ model investigated by Log et al. [22] [23] [24]. Moreover, Saurel et al. [36] and Pelanti & Shyue [37] employed HLLC for the $TF6$ model. In addition, de Böck et al. [38] used it for a different five-equation two-phase flow model. Collectively, these findings provide strong evidence that it is both feasible and reliable to discretize the $TF5$ model using the HLLC scheme as the approximate Riemann solver for flux calculations.

2.4. Literature gap

Given the aforementioned considerations, it is reasonable to assume that pressure and velocity (no-slip condition) are identical for both phases at rapid depressurization cases. By contrast, there are compelling arguments against assuming the same temperature and chemical potential for each phase. A review of the literature shows that several authors have provided derivations of the $TF5$ model, where only pressure and velocity are set equal. This derivation has been presented by Martinez et al. [14], Saurel et al. [16], Kapila et al. [18], Munkejord et al. [33], and Kreeft & Koren [34]. However, Martinez et al. [14], Kapila et al. [18] and Munkejord et al. [33], did not implement these models in any numerical simulations. Although Kreeft & Koren [34] performed simulations, their study addressed a two-fluid system without mass transfer, and Saurel et al. [16] focused on a single fluid with two phases of dodecane rather than CO_2. Consequently, there is no evidence in the literature that the $TF5$ model has been applied to CO_2. This represents a clear gap, suggesting that implementing the $TF5$ model for CO_2 could enhance simulation accuracy.

2.4.1. Objective

As outlined in section 1.2, this research aims to investigate existing flow models for CO_2 two-phase flow, assess their predictive capabilities in greater detail, and implement a new flow model to improve accuracy in simulating CO_2 flows. Several sub-questions have been formulated to fulfil this objective, as shown in Table 2.1.

Table 2.1: Overview of sub-questions and whether the sub-question has already been answered in the literature study, or will be answered in this master thesis.

Question	Method
1. What flow models have been used for simulating CO_2 flow?	
1.1 How accurate where the HEM and the $DF3$ models?	
1.1.1 What where the shortcomings of the $DF3$ model?	Literature study & master thesis
1.2 How accurate was the $DF4$ model?	
1.2.1 What where the shortcomings of the $DF4$ model?	Literature study & master thesis
1.3 Why did the $DF4$ model better predict in comparison to the $DF3$ model?	Master thesis
1.4 Which equation of state is used and recommended?	Literature study & master thesis
2. What flow models can be used for simulating CO_2 flow more accurately?	
2.1 Has the $TF5$ model been used for predicting CO_2 flow?	Literature study
2.2 Can the $TF5$ model be used for predicting CO_2 flow?	
2.2.1 Has the $TF5$ model been used by other researchers?	Literature study
2.2.2 Can the HLLC approximate Riemann solver be used for flux calculation?	Master thesis
2.3 Does the $TF5$ model predict CO_2 flow more accurately?	Master thesis

Although the $TF5$ model was successfully discretised and verified against several test cases, time constraints prevented its application to the CO_2-specific experiments. Consequently, further work is required to fully evaluate the model's potential for improving CO_2 flow simulations.

2.5. Experiments

In order to validate the different flow models, experimental data are essential. ECCSEL has a depressurization lab facility especially made for CO_2 and CO_2-rich mixtures. The facility is located at the roof of the Thermal Engineering Laboratories at the campus of the Norwegian University of Science and Technology (NTNU) in Trondheim. In the facility's experimental setup, a 61.67 $[m]$ horizontal pipeline is filled with CO_2 gas, and liquefied by two stages of compression and cooling, until the desired initial conditions are reached. The procedure of the experiment, and the properties, dimensions and sensors of the pipeline are given in Appendix E.

2.5.1. Test cases

In total, eight full-bore CO_2 pipe depressurization tests were conducted by Log et al. [24] and Munkejord et al. [21]. These tests all begin at a high initial pressure, in order for the CO_2 to be in the dense phase or liquid phase, before transitioning into the two-phase region. The complete set of experiments was conducted with initial temperatures approximately ranging from 5[°C] to 40[°C], and with an initial pressure of around 120 $[bar]$. Table 2.2 provides the initial conditions for the eight pure CO_2 tests, and Figure 2.7 illustrates their locations on a pressure–temperature diagram.

Table 2.2: Experimental conditions of the CO_2 depressurization tests, listed in the order of descending initial temperature. * done by Log et al. [24] ; † done by Munkejord et al. [21]. All conditions had an ambient pressure of 1 $[bar]$.

Test no.	Pressure avg. $[bar]$	Temperature avg. [°C]	Ambient Temperature [°C]
6^*	104.0	40.0	8.0
$24^\dagger$	115.6	35.8	10.4
$23^\dagger$	121.9	31.5	15.3
8^*	122.2	24.6	10.1
4^*	125.4	21.1	26.8
$22^\dagger$	124.8	14.9	14.4
$19^\dagger$	124.7	10.2	18.3
$25^\dagger$	122.7	4.6	-8.5

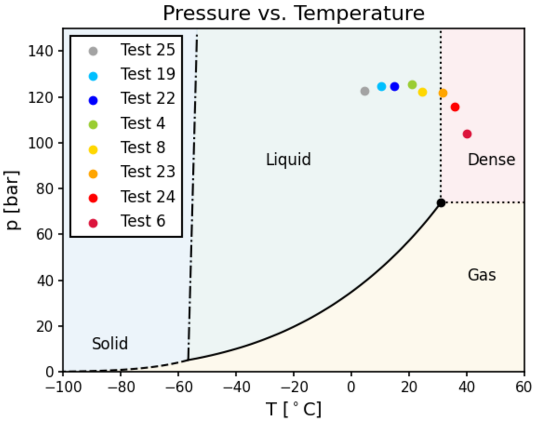


Figure 2.7: Initial conditions of the test-cases on the PT diagram of CO_2.

2.6. Chapter Summary

- **Two-Phase Flow Fundamentals**
  The chapter begins by explaining the transition from single-phase to two-phase flow for CO_2, highlighting the importance of accurately capturing liquid-gas interactions under high-pressure conditions.

- **Model Derivations and Simplifications**
  The Baer–Nunziato ($BN7$) model and its derived forms ($TF6$, $TF5$, $DF4$, and $DF3$) are introduced. Each model's simplifying assumptions—equalising pressures, velocities, temperatures, or chemical potentials—are discussed, illustrating the trade-off between complexity and computational practicality.

- **Equation of State (EOS) Considerations**
  The discussion covers key equations of state, including the Peng–Robinson (PR), Stiffened Gas (SG), and Span–Wagner (SW) formulations. While the SW EOS provides the highest accuracy for pure CO_2, it also involves significantly higher computational costs.

- **Numerical Discretisation**
  The importance of hyperbolicity in ensuring unique solutions to the flow equations is emphasised. The HLLC approximate Riemann solver is introduced as a robust method for handling shocks, having been applied successfully to models such as $TF6$, $DF4$, and $DF3$. This background supports the feasibility of employing $TF5$ with the same solver.

- **Literature Gap**
  A critical gap is identified: although the $TF5$ model is well-documented theoretically, its application to CO_2 is absent from the literature. Existing research primarily focuses on simpler models (e.g., $DF3$, $DF4$) or applies $TF5$ to other fluids without incorporating mass transfer. This suggests an opportunity for improved accuracy by implementing and testing $TF5$ for CO_2.

- **Experimental Context**
  The chapter concludes by outlining the ECCSEL depressurisation facility, describing the high-pressure tests undertaken on CO_2. These experiments, aimed at replicating dense and liquid-phase conditions, serve as reference points for validating both established and novel flow models.

By synthesising theoretical analysis, model derivations, and experimental findings, this chapter lays the groundwork for deeper investigation into CO_2 two-phase flow simulation and the potential application of the $TF5$ model.

3 Multiphase Flow

This chapter begins by defining what two-phase flow modelling is and which assumptions are used. In addition, the different flow models are examined in section 3.1. Here, more information is given about the flow models, and what the pros and cons are. section 3.2 introduces the Riemann setup, followed by an examination of the source terms in section 3.3. Finally, section 3.4 presents a discussion of the speed of sound in these flow models.

3.1. Two-phase flow modelling

Two-phase flow is the simplest form of multiphase flow, consisting of two distinct phases, which may be solid–liquid, solid–gas, or gas–liquid. In the context of pure CO_2 transport, two-phase gas–liquid flow typically arises during transient conditions and steady-state transport. Furthermore, because CO_2 is most commonly transported in a liquid or dense phase, and rapid depressurisation can cause it to transition to vapour, the gas–liquid regime is the most relevant for this research. The phase diagram in Figure 3.1 illustrates the conditions under which CO_2 is typically transported, as given by Han et al. [39].

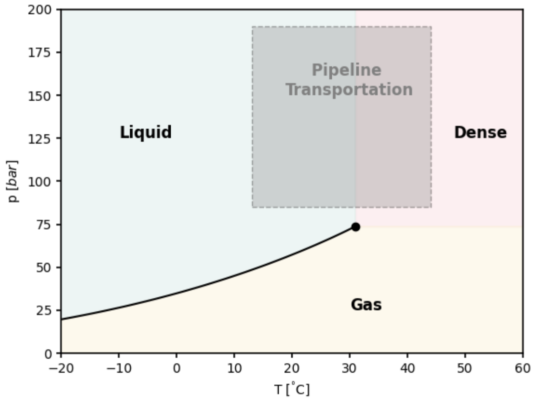


Figure 3.1: Phase diagram for CO_2, and the relevant operating range for CO_2 pipeline transportation [39].

As mentioned in subsection 2.1.1, the multiphase flow in this research is derived from the unsteady, compressible Euler equations. In these equations, viscosity and heat conduction are neglected, although they can be incorporated as source terms if required. The Euler equations are particularly suitable for flows with high Reynolds numbers, where inertial forces dominate over viscous forces. Conversely, they may be less appropriate in scenarios involving strong shocks or significant vorticity, as the simplification to the potential equation is no longer valid.

In the context of pipeline flow, especially during depressurisation, the flow often reaches high speeds and thus exhibits high Reynolds numbers. In addition, as the working fluid transitions from liquid to gas, the

Reynolds number increases further. Furthermore, as will be more thoroughly explained in subsection 3.2.3, only rarefaction waves (rather than shocks) occur in the pipeline, making the Euler equations a reasonable choice for modelling depressurisation test cases.

The following general assumptions, applicable to all models discussed, are used:

- The flow is modelled as one-dimensional, and thus the flow is assumed to be mixed in the cross-sectional direction.
- The viscous term, τ_{xx}, is neglected. However, wall-friction will be incorporated in the model.
- Turbulence are captured through the applied wall friction correlation.
- The flow is assumed to be adiabatic, indicating no heat transfer through the pipeline walls or to the ambient environment. However, heat transfer may occur within the fluid inside the pipeline.

3.1.1. The Homogeneous Equilibrium Model (*HEM*)

The Homogeneous Equilibrium Model (*HEM*) is widely used for modelling flow in pipelines, leveraging the Euler flow equations for accuracy under certain conditions. The *HEM* employs bulk conservation equations for mass, momentum, and energy, which, in differential form, are given as follows:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} = 0, \tag{3.1}$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2 + p)}{\partial x} = 0, \tag{3.2}$$

$$\frac{\partial E}{\partial t} + \frac{\partial\big((E+p)v\big)}{\partial x} = 0. \tag{3.3}$$

$$\tag{3.4}$$

The model assumes that both phases are in complete equilibrium for all thermodynamic and flow parameters, meaning that the pressure, velocity, temperature and chemical potential are assumed to be the same for both phases ($p_g = p_l = p,\ v_g = v_l = v,\ T_g = T_l = T,\ \mu_g = \mu_l = \mu$). This assumption simplifies the problem by treating the flow as a single-phase system with averaged properties, which is computationally efficient but limits the ability to capture non-equilibrium effects.

The eigenvalues of the Euler equations, as derived by Munkejord et al. [29], are all real and linearly independent, confirming that the system is hyperbolic. These eigenvalues are seen in Equation 3.5. Hyperbolicity is critical for ensuring well-posedness, as it guarantees that the solution evolves predictably over time. Additionally, the equations are written in a conservative form, meaning that all conserved variables are explicitly included in the flux terms. This ensures that physical quantities such as mass, momentum, and energy are conserved over time, even in the presence of discontinuities such as shock waves. Further details about the mixture speed of sound, an important characteristic for modelling compressible flows, is provided in section 3.4.

$$\mathbf{\Lambda} = \begin{bmatrix} v - c_{\text{HEM}} \\ v \\ v + c_{\text{HEM}} \end{bmatrix} \tag{3.5}$$

Pros and cons

The primary advantage of the *HEM* model lies in its simplicity and computational efficiency, making it suitable for simulations where rapid solutions are needed. By assuming complete equilibrium, the model avoids the complexities of tracking separate phase properties, leading to faster convergence and easier implementation.

However, the HEM model has significant limitations. The assumption of equilibrium for all parameters (pressure, velocity, temperature, and chemical potential) means it cannot capture non-equilibrium phenomena such as phase separation, velocity slip, or temperature gradients between phases. These effects, which are critical in certain scenarios like rapid depressurisation or highly dynamic flows, cannot be accurately resolved using the HEM model.

3.1.2. The Drift Flux Three Model ($DF3$)

The HEM model serves as the foundation for inviscid fluid dynamics. However, to better capture the physics of the problem, source terms can be incorporated. In the $DF3$ model, two additional source terms are included: wall friction ($\mathscr{F}$) and gravity ($\rho g \cdot sin(\delta)$). Here, the angle δ is defined with respect to the horizontal axis (positive if upward inclined). The modified momentum equation with the added source terms is shown in Equation 3.6. Further details about the source terms will be given in section 3.3.

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} &= 0, \\
\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2 + p)}{\partial x} &= \rho g \cdot sin(\delta) - \mathscr{F}, \\
\frac{\partial E}{\partial t} + \frac{\partial\big((E+p)v\big)}{\partial x} &= 0.
\end{aligned} \tag{3.6}$$

A heat transfer term representing the exchange of energy with the wall and ambient environment was also considered for the energy equation. However, due to the short simulation times typical of the scenarios analysed, this term was found to have an insignificant effect on the results and was therefore omitted.

The addition of these source terms does not alter the eigenvalues or speed of sound of the system. Consequently, the $DF3$ model retains the hyperbolic properties of the HEM model, making it a computationally efficient extension that improves the realism of the physical representation without significantly increasing complexity.

Pros and cons

The $DF3$ model offers advantages over the simpler HEM model without giving drawbacks. By incorporating source terms such as wall friction and gravity, it captures additional physical effects that are important in real-world pipeline flow scenarios. The model retains the computational efficiency and simplicity of the HEM model, as the addition of source terms does not alter the hyperbolicity or eigenstructure of the equations. This makes it well-suited for scenarios where the primary goal is to balance computational cost with improved physical accuracy.

The only additional downside is that the computation effort will slightly increase. Besides, it still does not account for non-equilibrium effects such as phase separation, velocity slip, or temperature gradients between phases, as it assumes complete equilibrium for all thermodynamic and flow parameters. Additionally, while the model includes gravitational and frictional effects, the exclusion of heat transfer terms may limit its applicability in cases where heat exchange plays a significant role over longer timescales.

3.1.3. The Drift Flux Four Model ($DF4$)

The Drift Flux Four Model ($DF4$) introduces a second equation for mass transfer, ensuring that each phase has its own mass conservation equation. The governing equations for this model are:

$$\begin{aligned}
\frac{\partial(\alpha_g\rho_g)}{\partial t} + \frac{\partial(\alpha_g\rho_g v)}{\partial x} &= \Gamma, \\
\frac{\partial(\alpha_l\rho_l)}{\partial t} + \frac{\partial(\alpha_l\rho_l v)}{\partial x} &= -\Gamma, \\
\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2 + p)}{\partial x} &= \rho g \cdot sin(\delta) - \mathscr{F}, \\
\frac{\partial E}{\partial t} + \frac{\partial\big((E+p)v\big)}{\partial x} &= 0.
\end{aligned} \tag{3.7}$$

In this model, the pressure, temperature, and velocity are still assumed to be in equilibrium. However, the chemical potential is no longer assumed to be equal. This adjustment allows the mass transfer between phases to be controlled dynamically through the source term Γ, aligning better with the physical processes. Further details regarding Γ and its implications are discussed in Section 3.3.3.

The eigenvalues for the $DF4$ model, derived by Martinez et al. [14], are shown in Equation 3.8:

$$\mathbf{\Lambda} = \begin{bmatrix} v - c_{\text{DF4}} \\ v \\ v \\ v + c_{\text{DF4}} \end{bmatrix} \tag{3.8}$$

Similar to the HEM model, the eigenvalues are all real and linearly independent, confirming that the system is hyperbolic. The additional eigenvalue $\lambda_3 = v$, which is not present in the HEM model, is associated with the propagation of the discontinuity in the mass fraction of the phases at the flow speed.

Pros and cons

Similar to the HEM and $DF3$ models, the $DF4$ model is conservative, with all fluxes explicitly defined within the differential equations. This ensures compatibility with the same numerical methods and schemes. Moreover, the inclusion of a second mass conservation equation introduces the flexibility to dynamically adjust the mass transfer between phases, enhancing the model's ability to capture non-equilibrium effects.

However, this flexibility also presents a challenge. Determining an accurate and physically consistent mass transfer relation can be complex and case-dependent. Additionally, the inclusion of the mass transfer source term can increase the model's stiffness, particularly over longer simulation time-scales, potentially requiring smaller time steps and more computational effort.

3.1.4. Alternative formulation

An alternative approach to the $DF4$ model involves replacing the separate mass transfer equations with one equation for a specific phase and another for the total mass transfer. Mathematically, this formulation is equivalent to the original $DF4$ equations and will produce identical results. However, from a programming perspective, this approach can simplify implementation and debugging. Additionally, it allows for an easy transition back to the $DF3$ model by disabling the phase mass transfer equation. It should be noted that transitioning back to the $DF3$ model impacts the calculation of primitive variables, as discussed in section 4.4. The governing equations for this alternative are as follows:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} &= 0, \\
\frac{\partial(\alpha_k\rho_k)}{\partial t} + \frac{\partial(\alpha_k\rho_k v)}{\partial x} &= \Gamma_k, \\
\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2 + p)}{\partial x} &= \rho g \cdot sin(\delta) - \mathscr{F}, \\
\frac{\partial E}{\partial t} + \frac{\partial\big((E+p)v\big)}{\partial x} &= 0.
\end{aligned} \tag{3.9}$$

3.1.5. The Two Phase Model $(TF5)$

The Two Phase 5 $(TF5)$ model extends the capabilities of the $DF4$ model by introducing an additional energy equation, resulting in a total of five governing equations. The complete set of equations is as follows:

$$
\begin{aligned}
\frac{\partial\left(\rho_g\alpha_g\right)}{\partial t}+\frac{\partial\left(\rho_g\alpha_g v\right)}{\partial x} &= \Gamma,\\
\frac{\partial\left(\rho_l\alpha_l\right)}{\partial t}+\frac{\partial\left(\rho_l\alpha_l v\right)}{\partial x} &= -\Gamma,\\
\frac{\partial(\rho v)}{\partial t}+\frac{\partial(\rho v^2)}{\partial x}+\frac{\partial p}{\partial x} &= \rho g\cdot sin(\delta)-\mathscr{F},\\
\frac{\partial E}{\partial t}+\frac{\partial(Ev)}{\partial x}+\frac{\partial(pv)}{\partial x} &= 0,\\
\frac{\partial E_g}{\partial t}+\frac{\partial(E_g v)}{\partial x}+\frac{\partial(\alpha_g p v)}{\partial x} &= \mathbf{S}_E.
\end{aligned}
\tag{3.10}
$$

In this model, only the pressure and velocity are assumed to be uniform for both phases, while the chemical potential and temperature are not. This allows the $TF5$ model to better capture non-equilibrium effects, such as phase-specific temperature gradients and chemical disequilibrium.

The eigenvalues for the $TF5$ model, shown in Equation 3.11, confirm that all eigenvalues are real and linearly independent, making the system hyperbolic. The eigenvalues are derived in Appendix C. The additional eigenvalues, $\lambda_{3,4}=v$, compared to the HEM model, are associated with the discontinuities in temperature and mass fraction of the phases, both of which propagate at the flow speed.

$$
\mathbf{\Lambda}=\begin{bmatrix} v-c_{\mathrm{TF5}}\\ v\\ v\\ v\\ v+c_{\mathrm{TF5}}\end{bmatrix}
\tag{3.11}
$$

In subsection 3.3.4, the source term $\mathbf{S}_E$ will be examined in detail. There it will be showed that the variables are not all contained in the derivatives, and the set of equations is non-conservative. This can give multiple problems, especially when dealing with shocks and discontinuities. In section 5.6, an explanation will be given on how to deal with these problems.

Pros and cons

The $TF5$ model offers significant advantages compared to the other models, as it can capture more non-equilibrium effects, including temperature differences between the phases. Additionally, the inclusion of a separate energy equation for the gas phase allows the model to potentially capture temperature delay effects, which can be crucial in accurately modelling transient flow behaviour.

However, the $TF5$ model also presents notable challenges. The inclusion of derivatives with outside variables in the source term makes the system non-conservative, complicating its numerical implementation. Non-conservative systems require special care to ensure that physical conservation laws are respected. This complexity often necessitates advanced numerical techniques, which can increase computational cost and implementation difficulty. Furthermore, the additional equations and source terms may introduce stiffness into the system.

3.1.6. Alternative Two Phase Model $(TF5-2)$

An alternative model where five equations are used is seen in Equation 3.12. The difference here is that a second energy equation is replaced by a topology equation. Nevertheless, the same set of eigenvalues is obtained as for the $TF5$ model, Equation 3.11. This model is used in order to verify the numerics, as it also contains derivatives where not all variables are contained in the derivatives, thus making it non-conservative. However, it will not be used in simulating CO_2 in the pipelines. Note that Γ, $\mathscr{F}$ and the gravity terms are neglected, as for the verification, these source terms were not important.

$$\begin{aligned}
\frac{\partial\left(\rho_g\alpha_g\right)}{\partial t}+\frac{\partial\left(\rho_g\alpha_g v\right)}{\partial x}&=0,\\
\frac{\partial\left(\rho_l\alpha_l\right)}{\partial t}+\frac{\partial\left(\rho_l\alpha_l v\right)}{\partial x}&=0,\\
\frac{\partial}{\partial t}(\rho v)+\frac{\partial}{\partial x}(\rho v^2)+\frac{\partial p}{\partial x}&=0,\\
\frac{\partial E}{\partial t}+\frac{\partial (E v)}{\partial x}+\frac{\partial (p v)}{\partial x}&=0,\\
\frac{\partial \alpha_g}{\partial t}+\frac{\partial \alpha_g v}{\partial x}+(\phi-\alpha_g)\frac{\partial v}{\partial x}&=0,\\
\phi=\alpha_g\alpha_l\frac{\rho_g c_g^2-\rho_l c_l^2}{\alpha_l\rho_g c_g^2+\alpha_g\rho_l c_g^l}&.
\end{aligned} \tag{3.12}$$

3.2. Riemann setup

The Riemann problem is a fundamental concept in the study of hyperbolic partial differential equations (PDEs). It provides a simplified framework for understanding wave propagation and the behaviour of solutions to these equations. The dynamics of conserved variables and the relationships between other wave types, including shock waves, contact discontinuities, and rarefaction waves can be learned by looking at the Riemann problem. This section will explore the characteristics and Riemann invariants, followed by a detailed discussion of the Riemann problem and its implications for flow models.

3.2.1. Characteristics and Riemann invariants

The hyperbolic nature of the models allows them to be rewritten in their characteristic form. By examining the characteristic properties, we can gain insight into the transport of conserved variables. The models are rewritten by using the chain rule:

$$\mathbf{U}_t+\mathbf{J}\mathbf{U}_x=0, \tag{3.13}$$

$$\mathbf{J}=\mathbf{R}\boldsymbol{\Lambda}\mathbf{R}^{-1}, \tag{3.14}$$

where **J** is the Jacobian matrix. The model can then be redefined by the characteristic vector:

$$\mathbf{w}=\mathbf{R}^{-1}\mathbf{U} \tag{3.15}$$

$$\mathbf{w}_t+\boldsymbol{\Lambda}\mathbf{w}_x=0. \tag{3.16}$$

This means that the conserved variables, **U**, are reformulated into linearly independent characteristics variables, **w**. The reason for doing this is that the velocities of the independent variables are solely given by the eigenvalues, λ. The travel path of the information can be plotted in a space-time diagram. An example is sketched in Figure 3.2. Each line indicates the trajectory along which information about each characteristic property travels. The slope of these lines is given by $\frac{1}{\lambda}$. There are four separate regions in the time-space diagram. In region *a*, **w** is equal to $\mathbf{w}_L$, as all information comes from the most left state and no mixing happens. The same applies for region *d*, where **w** is equal to $\mathbf{w}_R$. To the contrary, regions *b* and *c* do have information coming from the left and right state, meaning that the primitive variables can be calculated as a combined function of $\mathbf{w}_L$ and $\mathbf{w}_R$.

3.2.2. Riemann problem

The Riemann problem is initially a shock tube, which is separated in the middle by a diaphragm. It is is crucial for understanding the behaviour of solutions to hyperbolic PDEs. The left side and right side both have different starting conditions, and at t = 0, the diaphragm is removed. The setup is written in Equation 3.17.

$$\begin{aligned}
&\mathbf{U}_t+\mathbf{F(U)}_x=0,\\
&\mathbf{U}(x,0)=\begin{cases}\mathbf{U}_L & \text{if } x<0,\\ \mathbf{U}_R & \text{if } x>0.\end{cases}
\end{aligned} \tag{3.17}$$

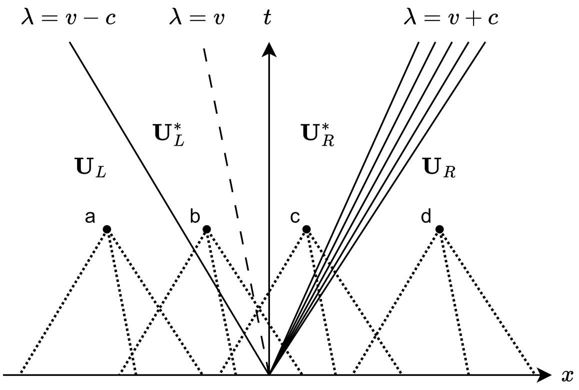


Figure 3.2: A space-time diagram with the propagations of the characteristics. To the left are the shock wave & contact discontinuity, and to the right is the rarefaction wave. Assuming a sufficiently short time scale, the variables 'u' and 'c' can be considered approximately constant. This approximation results in straight characteristic lines.

The result will be that multiple waves will appear. The number of waves depends on the system of flow equations, and how many eigenvalues it has. There are three wave type: a contact discontinuity, a shock wave or a rarefaction wave.

A shock wave is known as a type of propagating disturbance that moves with the local speed of sound in the medium. Here, all variables (p, v, ρ, e), are discontinuous. It is characterized by an abrupt change in the variables. The velocity of a shock wave is calculated with the Rankine-Hugoniot (RH) condition, which is:

$$\mathbf{F}(\mathbf{U})_L - \mathbf{F}(\mathbf{U})_R = S(\mathbf{U}_L - \mathbf{U}_R). \tag{3.18}$$

Here, S is the speed of the wave. In section 5.2, the RH condition is used to derive the different states in the HLLC approximate Riemann solver.

A contact discontinuity is characterized by a surface in the flow field across which there is a discontinuity in the fluid properties. It reflects the contact of two fluids in different states. While in a shock wave, all variables are discontinuous, at a contact discontinuity, the pressure, p, and velocity, v, are not discontinuous. The density, ρ, and internal energy, e, are discontinuous, and thus the entropy, s, is also discontinuous. Again, the speed of the wave is calculated with Equation 3.18.

Lastly, there is the rarefaction wave. A rarefaction wave is a type of wave that causes a decrease in density and pressure in the medium. In rarefaction waves, the variables, such as density ρ, velocity v, internal energy e, and pressure p change gradually and continuously. This means that the entropy is constant along the flow lines.

3.2.3. Practical application

Now applying the Riemann problem to the flow models, it is seen that every model has three distinct eigenvalues, namely $v - c$, v and $v + c$. These eigenvalues correspond to the speed of the three waves. The middle wave with speed v will always be a contact discontinuity, while the two other waves can be rarefaction or shock waves, depending on the initial condition. With these details, the solution of numerical simulations can be found, and the waves will tell how different stages are connected with each other. In a more practical sense, when a pipeline ruptures, the conditions in the pipeline can be seen as one state, while the ambient conditions are a second state. Because the initial conditions in the pipeline often have much higher pressure than outside, there will only be a rarefaction wave moving inside the pipeline, while there will be a shock wave and contact discontinuity travelling outside of the pipeline.

3.3. Source Term

In order to describe the physics more accurately, multiple source terms can be included. The source terms that are used are the Γ term for the mass transfer but also for the delay in condensation/evaporation, the friction of the fluid with the pipeline denoted as $\mathscr{F}$ and a gravity term (which is zero for horizontal test cases). In addition, the source term for the energy equation of the gas is derived at the end of this section.

3.3.1. Friction (Friedel method)

The friction is calculated with the Friedel correlation [40], in the simulations made by Log et al. [23]. The friction is calculated as follows, where the top equation is for single phase flow and the the bottom for two-phase flow.

$$\mathscr{F} = \begin{cases} f_k \frac{\rho v|\rho v|}{2\rho_k d_\mathrm{i}}, \\ f_\ell \frac{\rho v|\rho v|}{2\rho_\ell d_\mathrm{i}} \Phi. \end{cases} \tag{3.19}$$

$f_k = f(Re_k)$ is the Darcy friction factor, d_i is the inner diameter of the pipeline and Φ is an empirical correlation, which is used to account for two-phase flow, and it depends on various properties of both phases.

The Darcy friction factor is a parameter that depends on whether the flow is laminar or turbulent. It is calculated based on how high the Reynolds number is. In the Reynolds number, the viscosity is calculated with the method of Laesecke & Muzny [41]. For single phase flow, it is calculated as follows:

$$\boldsymbol{f_k} = \begin{cases} \frac{64}{Re}, & \text{if } Re < 1055, \\ \left(0.86859 \cdot ln\left[\frac{Re}{1.964 \cdot ln(Re) - 3.8215}\right]\right)^{-2}, & \text{if } Re > 1055, \end{cases} \tag{3.20}$$

$$Re = \frac{|\rho \cdot v| \cdot d_i}{\eta}. \tag{3.21}$$

For the two-phase flow, the Reynolds number, Re, and viscosity, η[1], are based on the volume fraction of each phase:

$$\boldsymbol{f_k} = \begin{cases} \frac{64}{Re}, & \text{if } Re < 1055, \\ \left(0.86859 \cdot ln\left[\frac{Re}{1.964 \cdot ln(Re) - 3.8215}\right]\right)^{-2} \cdot \Phi, & \text{if } Re > 1055, \end{cases} \tag{3.22}$$

$$Re = \alpha_g Re_g + \alpha_l Re_l, \quad Re_k = \frac{|\rho_k \cdot v| \cdot d_i}{\eta_k}, \tag{3.23}$$

$$\eta = \alpha_g \eta_g + \alpha_l \eta_l, \quad \eta_k = f(\rho_k, T). \tag{3.24}$$

Φ is calculated the following way. Here A, B and C are arbitrary letters to indicate functions. σ is the surface tension, and for all calculations, it is set to $53 \cdot 10^{-3} [N/m]$. A better way is to calculate the surface tension with the method of Mulero et al. [42], as recommended by NIST [43].

$$\Phi^2 = A + \frac{3.24 \cdot B \cdot C}{Fr_h^{0.045} We_l^{0.035}}, \tag{3.25}$$

$$A = (1-Y)^2 + Y^2 \frac{\rho_l f_g}{\rho_g f_l}, \tag{3.26}$$

$$B = Y^{0.78}(1-Y)^{0.224}, \tag{3.27}$$

$$C = \left(\frac{\rho_l}{\rho_g}\right)^{0.91} \left(\frac{\eta_g}{\eta_l}\right)^{0.19} \left(1 - \frac{\eta_g}{\eta_l}\right)^{0.7}, \tag{3.28}$$

$$f_l = \frac{0.316}{Re_l^{0.25}}, \quad f_g = \frac{0.316}{Re_g^{0.25}}, \tag{3.29}$$

$$Fr_h = \frac{(\rho \cdot v)^2}{9.81 \cdot d_i \cdot \rho}, \quad We_l = \frac{(\rho \cdot v)^2 d_i}{\sigma \cdot \rho}. \tag{3.30}$$

Please note, because Φ is multiplied in the friction factor, and in the final friction equation, it is the same as having Φ^2, More details on the expressions can be found in [40] and [44].

[1] In literature, viscosity can also be defined as μ

3.3.2. Friction (Churchill method)

Another method to calculate friction is the Churchill method [45]. This method is easier to implement and, as will be shown later, provides similar results as the previous method. Below is shown how the friction is calculated with the Churchill method. What is clear is that it is much less complex to calculate and to implement. In addition, there is only a small difference between single phase, two phase and two phase for the $TF5$ model. The only difference is how the mixture viscosity is calculated.

$$\mathscr{F} = f_{churchill} \frac{2\rho v|v|}{d_\mathrm{i}}, \tag{3.31}$$

$$f_{churchill} = 2 \cdot (Term_1 + Term_2)^{\frac{1}{12}}, \tag{3.32}$$

$$Term_1 = \left(\frac{8}{Re}\right)^{12}, \tag{3.33}$$

$$Term_2 = \left(\frac{1}{A+B}\right)^{1.5}, \tag{3.34}$$

$$A = \left[-2.457 \cdot ln\left(\left(\frac{7}{Re}\right)^{0.9} + 0.27\frac{\varepsilon}{d_i}\right)\right]^{16}, \tag{3.35}$$

$$B = \left(\frac{37530}{Re}\right)^{16}. \tag{3.36}$$

In this method, the Reynolds number is always calculated with the mixture density and mixture viscosity. ε is the pipe mean roughness.

$$Re = \frac{|\rho \cdot v| \cdot d_i}{\eta}, \tag{3.37}$$

$$\eta = \begin{cases} \eta(\rho, T), & \text{if Single Phase ,} \\ \alpha_g \eta_g(\rho_g, T) + \alpha_l \eta_l(\rho_l, T), & \text{if Two-Phase,} \\ \alpha_g \eta_g(\rho_g, T_g) + \alpha_l \eta_l(\rho_l, T_g). & \text{if Two-Phase with } TF5. \end{cases} \tag{3.38}$$

3.3.3. Mass transfer for the $DF4$ model

The mass transfer between the gas and liquid allows for incorporating non-equilibrium effects, and gives the ability of superheating of the liquid phase or subcooling of the gas phase. A possible formulation for Γ is:

$$\Gamma = \rho \frac{Y_{g,sat} - Y_g}{\theta}, \tag{3.39}$$

$$Y_g = \frac{\rho_g \alpha_g}{\rho}. \tag{3.40}$$

Here, θ is a relaxation time related to the delay in the phase change process. A larger θ provides a higher phase change process, allowing for more superheating of the liquid phase. In the testcases, θ is a tuning parameter. Y_g is the mass fraction of the gaseous phase.

The saturation mass fraction of the gaseous phase, $Y_{g,sat}$, can be computed using three distinct methods:

1. **Bilicki & Kestin Method** Based on the work of Bilicki & Kestin [46], the saturated mass fraction is given by:

$$Y_{g,\mathrm{sat}} = \frac{\frac{1}{\rho} - \frac{1}{\rho_{l,\mathrm{sat}}}}{\frac{1}{\rho_{g,\mathrm{sat}}} - \frac{1}{\rho_{l,\mathrm{sat}}}}. \tag{3.41}$$

In the original paper, the saturated liquid and vapour densities ($\rho_{l,sat}$ and $\rho_{g,sat}$) were functions of pressure. In this research, however, they are functions of temperature.

2. **Reduced UV Flash for *DF3*** This method leverages saturation densities as functions of temperature and is based on a reduced UV flash calculation developed by Kumar et al. [47]. More is explained in subsection 4.4.3. It is defined as:

$$Y_{g,sat} = \frac{\rho_{g,sat}\alpha_{g,sat}}{\rho}, \tag{3.42}$$

$$\alpha_{g,sat} = \frac{\rho - \rho_{l,sat}}{\rho_{g,sat} - \rho_{l,sat}}. \tag{3.43}$$

$$\tag{3.44}$$

3. **UV Flash of *DF3*** Used by Log et al. [22], this method calculates the gaseous density and volume fraction using the UV flash from the $DF3$ model, as explained in subsection 4.4.2:

$$Y_{g,\text{sat}} = \frac{\rho_{g,\text{HEM}}\alpha_{g,\text{HEM}}}{\rho}. \tag{3.45}$$

The first two methods yield identical results, making them reliable and computationally efficient. The third method, while slightly different, is computationally more expensive and prone to numerical errors. As such, it is recommended to use one of the first two methods. In this research, the 'Reduced UV Flash for $DF3$' is used.

The mass transfer term has been used more frequently for other substances [46]. Other mass transfer terms are applied in literature. However, for these transfer terms, other equilibrium assumptions have to be made, resulting in a not working term for the used models in this research.

- $\Gamma = K(\mu_l - \mu_g)$,
- $\Gamma = C \cdot \alpha_l \rho_l \frac{T - T_{sat}}{T_{sat}}$,
- $\Gamma = C \cdot \alpha_l \rho_l \frac{p - p_{sat}}{p_{sat}}$.

3.3.4. Source term energy equation

The derivation of the energy source term is based on Kreeft & Koren [34]. It should be noted that it has been independently done by Munkejord et al. [33] for a six equation model (see Appendix B).

The energy exchange is possible due to mechanical work, thermodynamic work, and heat exchange with the phases. In differential form, the rate of energy exchange is given to be:

$$\frac{\mathrm{d}w}{\mathrm{d}t} = \mathbf{F} \cdot \frac{\mathrm{d}r}{\mathrm{d}t} = \mathbf{F} \cdot v. \tag{3.46}$$

In here, $\mathbf{F}$ is the force exerted by the liquid phase on the gaseous phase. We denote $\mathbf{F}$ as $\mathbf{S}_M$, being the mechanical work done. Next, there is the thermodynamics work. This happens when a fluid changes volume, and is given in the time-derivative as:

$$\frac{\mathrm{d}w}{\mathrm{d}t} = -p \cdot \frac{\mathrm{d}V}{\mathrm{d}t}. \tag{3.47}$$

We denote this as $\mathbf{S}_T$. Lastly, there is the energy exchange by heat exchange of the mass transfer. We we call this $\mathbf{S}_H$. The total source term is:

$$\mathbf{S}_E = \mathbf{S}_M \cdot v + \mathbf{S}_T + \mathbf{S}_H. \tag{3.48}$$

To find $\mathbf{S}_M$, the assumption of a bulk velocity is applied. This means that the velocity in the momentum equation of the liquid phase is the same as the velocity in the momentum equation of the gaseous phase, Equation 3.49. Now, we fill in the mass conservation equation of the gaseous phase with no mass transfer, Equation 3.50, and we get Equation 3.51.

$$\frac{\partial(\alpha_g\rho_g v_g)}{\partial t}+\frac{\partial(\alpha_g\rho_g v_g^2)}{\partial x}+\frac{\partial(\alpha_g p_g)}{\partial x}=\mathbf{S}_M, \tag{3.49}$$

$$\frac{\partial(\alpha_g\rho_g)}{\partial t}+\frac{\partial(\alpha_g\rho_g v_g)}{\partial x}=0, \tag{3.50}$$

$$\frac{\partial v}{\partial t}+v\frac{\partial v}{\partial x}+\frac{1}{\alpha_g\rho_g}\frac{\partial\alpha_g p}{\partial x}=\frac{1}{\alpha_g\rho_g}\mathbf{S}_M. \tag{3.51}$$

When doing this for the bulk momentum and bulk mass conversation equations, respectively Equation 2.7 and Equation 2.6, the following is found:

$$\frac{\partial v}{\partial t}+v\frac{\partial v}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}=0. \tag{3.52}$$

Note that here there are no source terms. Because the velocity v should be the same for both phases, the term $\partial v/\partial t+v\partial v/\partial x$ can be substituted in both equations, resulting in:

$$-\frac{1}{\rho}\frac{\partial p}{\partial x}=\frac{1}{\alpha_g\rho_g}\mathbf{S}_M-\frac{1}{\alpha_g\rho_g}\frac{\partial(\alpha_g p)}{\partial x}, \tag{3.53}$$

$$\mathbf{S}_M=\frac{\partial(\alpha_g p)}{\partial x}-\frac{\alpha_g\rho_g}{\rho}\frac{\partial p}{\partial x}. \tag{3.54}$$

Note that $\alpha_g\rho_g/\rho$ is in fact the mass transfer. In addition, the $\partial(\alpha_g p)/\partial x$ will be split. This becomes:

$$\mathbf{S}_M=\alpha_g\frac{\partial p}{\partial x}+p\frac{\partial\alpha_g}{\partial x}-Y_g\frac{\partial p}{\partial x}=p\frac{\partial\alpha_g}{\partial x}+(\alpha_g-Y_g)\frac{\partial p}{\partial x}. \tag{3.55}$$

$$\tag{3.56}$$

Now for the thermodynamic source term, the assumption is made that the model is isentropic in a continuous domain. This means that the entropy is constant, and thus:

$$\mathrm{d}s=0\Rightarrow\frac{\partial s}{\partial t}+v\frac{\partial s}{\partial x}=0. \tag{3.57}$$

$$\tag{3.58}$$

In addition, it is known that the speed of sound is Equation 3.59, and so it can be substituted:

$$c_k^2=\left(\frac{\mathrm{d}p}{\mathrm{d}\rho_k}\right)_{s_k}, \tag{3.59}$$

$$\mathrm{d}s_k\propto\mathrm{d}p-c_k^2\mathrm{d}\rho_k, \tag{3.60}$$

$$\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}-c_k^2\left(\frac{\partial\rho_k}{\partial t}+v\frac{\partial\rho_k}{\partial x}\right)=0. \tag{3.61}$$

Now substituting the phase mass conservation equations, Equation 2.17 and Equation 2.18, in Equation 3.61, it becomes the following:

$$\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}+\rho_g c_g^2\left[\frac{1}{\alpha_g}\left(\frac{\partial\alpha_g}{\partial t}+v\frac{\partial\alpha_g}{\partial x}\right)+\frac{\partial v}{\partial x}\right]=0, \tag{3.62}$$

$$\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}+\rho_l c_l^2\left[\frac{-1}{\alpha_l}\left(\frac{\partial\alpha_l}{\partial t}+v\frac{\partial\alpha_l}{\partial x}\right)+\frac{\partial v}{\partial x}\right]=0. \tag{3.63}$$

Both equations can be filled in for $\partial p / \partial t + v \partial p / \partial x$, which will result in:

$$\frac{\partial \alpha_g}{\partial t} + v\frac{\partial \alpha_g}{\partial x} + \varphi\frac{\partial v}{\partial x} = 0, \tag{3.64}$$

$$\tag{3.65}$$

with

$$\varphi = \alpha_g \alpha_l \frac{\rho_g c_g^2 - \rho_l c_l^2}{\alpha_l \rho_g c_g^2 + \alpha_g \rho_l c_l^2}.$$

Now because the pressures are equal to each other, the following holds:

$$\left(\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x}\right)(e_g, \rho_g) = \left(\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x}\right)(e_l, \rho_l) \tag{3.66}$$

This can be rewritten as:

$$\mathbf{S}_T = -p\left(\frac{\partial \alpha_g}{\partial t} + v\frac{\partial \alpha_g}{\partial x}\right). \tag{3.67}$$

Because it is very hard to model an additional source term in the time derivative, we want to rewrite this term into only spatial derivatives. This is done by filling in Equation 3.64 in Equation 3.67. The term becomes:

$$\mathbf{S}_T = p\left(\varphi\frac{\partial v}{\partial x}\right), \tag{3.68}$$

$$\tag{3.69}$$

with

$$\varphi = \alpha_g \alpha_l \frac{\rho_g c_g^2 - \rho_l c_l^2}{\alpha_l \rho_g c_g^2 + \alpha_g \rho_l c_l^2}. \tag{3.70}$$

The term $\mathbf{S}_H$ is based on the flow equations derived by Collier and Thome in chapter 2.2 [48]. The $\mathbf{S}_H$ term is:

$$\mathbf{S}_H = \Gamma\left(h_g + \frac{v_g^2}{2}\right). \tag{3.71}$$

Adding all parts up, the total source term becomes:

$$\mathbf{S}_E = \left(p\frac{\partial \alpha_g}{\partial x} + (\alpha_g - Y_g)\frac{\partial p}{\partial x}\right) \cdot v + p\left(\varphi\frac{\partial v}{\partial x}\right) + \Gamma\left(h_g + \frac{v_g^2}{2}\right). \tag{3.72}$$

3.4. Speed of Sound

A sound wave is an oscillatory motion in a compressible fluid with small amplitude. It alternately compresses and decompresses the fluid at each location. Since the perturbations of the density, pressure and velocity are small, the derivation of the speed of sound is typically conducted using a first-order approximation. This approach leads to the general expression for the speed of sound, given by:

$$c^2(T, \rho) = \left(\frac{\partial p}{\partial \rho}\right)_s. \tag{3.73}$$

The full derivation is given in Appendix A.

Table 3.1: Parameters for mixture speed of sounds for CO_2. The parameters are found with the Span Wagner EOS at the saturation line.

Quantity	Liquid (300 $[K]$)	Gas (300 $[K]$)	Liquid (280 $[K]$)	Gas (280 $[K]$)
Pressure $[bar]$	67.13	67.13	41.61	41.61
Temperature $[K]$	300.0	300.0	280.0	280.0
Density $[kg/m^3]$	679.24	268.58	121.74	883.58
Sound speed $[m/s]$	245.67	185.33	207.72	471.54
Isobaric heat capacity $[J/(kg \cdot K)]$	8697.9	11921.0	2276.9	2814.1
Entropy $[m^2/(s^2 \cdot K)]$	1275.9	1621.5	1805.0	1059.8
Grüneisen parameter $[-]$	0.396	0.287	0.315	0.738

From the general speed of sound, the mixture speed of sound of every model can be derived. This becomes Equation 3.75 for the HEM & $DF3$ models, Equation 3.76 for the $DF4$ model and Equation 3.77 for the $TF5$ & $TF5-2$ models. The derivation for the $DF3$ mixture speed of sound is derived by Saurel et al. [16], for the $DF4$ model by Flåtten et al. [49] and for the $TF5$ model by Wood & Lindsay [50]. In addition, Morin et al. [15] showed that the hierarchy of the models follow the sub-characteristics condition, provided that the phase velocities are equal to each other. Accordingly, models that permit greater non-equilibrium variables get an equal or higher speed of sound:

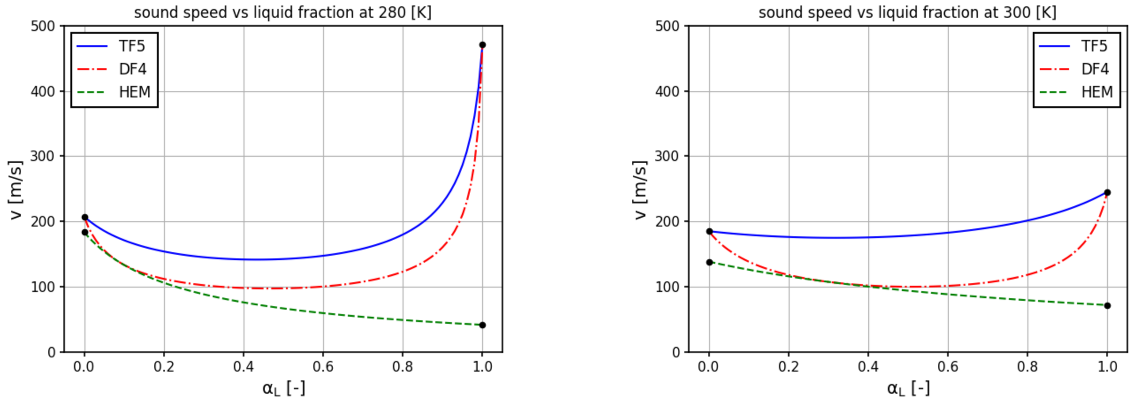
$$c_{\text{HEM}} \leq c_{\text{DF4}} \leq c_{\text{TF5}} \leq c_{\text{TF6}}, \tag{3.74}$$

$$c_{\text{HEM}} = \left[\rho\left(\frac{\alpha_g}{\rho_g c_g^2} + \frac{\alpha_\ell}{\rho_\ell c_\ell^2} + T\tilde{C}_{p,g}\left(\frac{\frac{1}{\rho_\ell} - \frac{1}{\rho_g}}{h_g - h_\ell} + \frac{\gamma_g}{\rho_g c_g^2}\right)^2 + T\tilde{C}_{p,\ell}\left(\frac{\frac{1}{\rho_\ell} - \frac{1}{\rho_g}}{h_g - h_\ell} - \frac{\gamma_\ell}{\rho_\ell c_\ell^2}\right)^2\right)\right]^{-1/2}, \tag{3.75}$$

$$c_{\text{DF4}} = \left[\rho\left(\frac{\alpha_g}{\rho_g c_g^2} + \frac{\alpha_\ell}{\rho_\ell c_\ell^2} + T\frac{\tilde{C}_{p,g}\tilde{C}_{p,l}}{\tilde{C}_{p,g} + \tilde{C}_{p,l}}\left(\frac{\gamma_g}{\rho_g c_g^2} - \frac{\gamma_l}{\rho_l c_l^2}\right)^2\right)\right]^{-1/2}, \tag{3.76}$$

$$c_{TF5} = \left[\rho\left(\frac{\alpha_g}{\rho_g c_g^2} + \frac{\alpha_l}{\rho_l c_l^2}\right)\right]^{-1/2}, \tag{3.77}$$

The three mixture speed of sounds are plotted in Figure 3.3. This is for a temperature of 6.85 °C or 280 $[K]$ and a temperature of 26.85 °C or 300 $[K]$. All the other properties are found at the saturation line with the SW equation of state, and given in Table 3.1. These are found with the NIST database [43]. As can be seen, the speed of sound of the HEM model is discontinuous, that is, c_{HEM} $(\alpha_l = 1) \neq c_l$ and c_{HEM} $(\alpha_g = 1) \neq c_g$. The reason for this is that the usual linear theory of sound does not work. Landau & Lifshitz [51] described this by stating that in addition to only the compressions and decompressions, the waves are accompanied by a change between a one-phase and a two-phase system. In literature, it is regarded as unphysical that the sound wave is discontinuous. In experiments by Brennen et al. [52] in Figure 9.2, it has been observed that the mixture speed of sound corresponds more with the 'shapes' of the $DF4$ and $TF5$ models. Especially for lower temperatures, the speed of sound of the liquid phase is much higher than the mixture speed of sound, and it quickly falls. This is observed in Figure 3.3a for the $DF4$ and $TF5$ models. An explanation for this phenomenon is not found in the literature. A possible explanation could be that in the multiphase regime, the mixture behaves as a sponge, and it is harder for the discontinuity to progress. It is very important to model the speed of sound correctly, as it determines the speed at which pressure waves travel through it. In a depressurization case, it will determine how rapid the pressure drop will propagate through the pipeline.



(a) Mixture speed of sound for the *HEM*, *DF*4 and *TF*5 models for CO_2 at T =280 [K]. The mixture speed of sound is the same for the *HEM* & *DF*3 models. Note that it is discontinuous at the boundaries for the *HEM* mixture speed of sound

(b) Mixture speed of sound for the *HEM*, *DF*4 and *TF*5 models for CO_2 at T =300 [K]. The mixture speed of sound is the same for the *HEM* & *DF*3 models. Note that it is discontinuous at the boundaries for the *HEM* mixture speed of sound

Figure 3.3: Mixture speed of sounds for CO_2, applied with different models and two temperatures.

Two-phase parameters

The following parameters are commonly employed to mathematically describe the two-phase flow (with the assumption that the flow is 1D):

Volume fraction:

$$\alpha_k = \frac{V_k}{V_l + V_g}, \quad \alpha_g + \alpha_l = 1. \tag{3.78}$$

Mixture density:

$$\rho = \rho_g \alpha_g + \rho_l \alpha_l. \tag{3.79}$$

The mass fraction:

$$Y_k = \frac{\rho_k \alpha_k}{\rho}. \tag{3.80}$$

The mass flux:

$$\dot{m} = \rho \cdot v. \tag{3.81}$$

Mixture specific internal energy:

$$\rho e = \rho_g \alpha_g e_g + \rho_l \alpha_l e_l, \quad e = Y_g e_g + Y_l e_l. \tag{3.82}$$

The total energy:

$$E = \rho(e + \frac{1}{2}v^2) = E_l + E_g. \tag{3.83}$$

The component energy:

$$E_k = \rho_k \alpha_k (e_k + \frac{1}{2}v^2). \tag{3.84}$$

The component heat capacity:

$$\tilde{C}_{p,k} = \alpha_k \rho_k c_{p,k}. \tag{3.85}$$

Mixture viscosity

$$\mu = \alpha_g \mu_g + \alpha_l \mu_l.^{2} \tag{3.86}$$

[2] Some authors use the mass fraction instead of volume fraction, see mixture specific internal energy.

Summary

This chapter delves into the mathematical and physical principles of multiphase flow models, focusing on two-phase gas/liquid flow models, applied for depressurization of CO_2. The foundational framework for these models is derived from the unsteady compressible Euler equations, which are well-suited for high Reynolds number flows typical of pipeline transport.

A hierarchy of flow models is presented, ranging from the simplest, the Homogeneous Equilibrium Model (HEM), to more complex models like the Two-Phase Five Model ($TF5$). Each model incrementally relaxes equilibrium assumptions, adding layers of complexity to better capture the non-equilibrium effects present in real-world scenarios. While the HEM assumes equilibrium for all variables, the $DF3$ introduces source terms for friction and gravity. The $DF4$ extends this further by including a second mass conservation equation to account for non-equilibrium chemical potential effects, and the $TF5$ incorporates separate energy equations for each phase, enabling the modelling of temperature disequilibrium.

The chapter also examines the Riemann problem, a crucial concept in solving hyperbolic partial differential equations (PDEs) for fluid flow and wave propagation. By analysing the characteristic properties and eigenvalues of the models, insights are gained into wave propagation, including shock waves, contact discontinuities, and rarefaction waves. This analysis provides a foundation for understanding wave dynamics within the flow and demonstrates its practical applications.

Source terms are explored in detail to enhance the physical accuracy of the models. These include friction, which is calculated using both the Friedel and Churchill methods, mass transfer terms for phase transitions, and energy source terms for capturing mechanical and thermodynamic work. The derivation of these terms, particularly for the $TF5$ model, highlights the interplay between physical realism and computational feasibility.

Finally, the mixture speed of sound is analysed for all models, revealing significant differences in how they capture this critical parameter. The HEM model shows discontinuities in the speed of sound at phase boundaries, which are considered unphysical. In contrast, the $DF4$ and $TF5$ models exhibit smoother transitions, aligning more closely with experimental observations. The chapter emphasizes the importance of accurately modelling the speed of sound, as it directly impacts the propagation of pressure waves in depressurization scenarios.

4

Thermodynamics

In order to solve the flow models described in chapter 3, an equation of state (EOS) is needed. The EOS, relates different state variables of a fluid. The most widely used equation of state is the functional relationship between pressure, temperature, and density. In section 4.1, the different fundamental properties are stated and derived. section 4.2 explains the Span Wagner (SW) EOS and section 4.3, explains the Peng Robinson (PR) EOS. Lastly, the procedure of finding the primitive variables is described in section 4.4

4.1. Equation of state

State variables are intrinsic properties that depend exclusively on the equilibrium state of a fluid, irrespective of the process undertaken to achieve that state. Density, (ρ), temperature (T), pressure (p), specific internal energy (e), and specific entropy (s) are a few examples of thermodynamic state variables. In addition, a combination of either of the previously stated state variables are also state variables. Several important ones are:

The specific enthalpy:

$$h = e + \frac{p}{\rho}, \tag{4.1}$$

The specific Gibbs free energy:

$$G = h - Ts, \tag{4.2}$$

The specific Helmholtz free energy:

$$a = e - Ts. \tag{4.3}$$

Just two of the thermodynamic state variables are necessary to fully characterise the thermodynamic state of a fluid. It is possible to derive all additional thermodynamic state variables from the fundamental equation. The SW EOS is widely recognized as the most reliable EOS for modelling for pure CO_2. It is derived from the Helmholtz free energy, $a = a(T, \rho)$, from which all other properties can be derived. The following definitions are used:

Pressure:

$$p(T, \rho) = \rho^2 \left(\frac{\partial a}{\partial \rho}\right)_T, \tag{4.4}$$

Specific entropy:

$$s(T, \rho) = -\left(\frac{\partial a}{\partial T}\right)_\rho, \tag{4.5}$$

Specific internal energy:

$$e(T, \rho) = a - T\left(\frac{\partial a}{\partial T},\right)_\rho \tag{4.6}$$

Specific enthalpy:

$$h(T,\rho) = a - T\left(\frac{\partial a}{\partial T}\right)_\rho + \rho\left(\frac{\partial a}{\partial \rho}\right)_T, \tag{4.7}$$

Specific isochoric heat capacity:

$$c_v(T,\rho) = \left(\frac{\partial e}{\partial T}\right)_\rho, \tag{4.8}$$

Specific isobaric heat capacity:

$$c_p(T,\rho) = \left(\frac{\partial h}{\partial T}\right)_\rho, \tag{4.9}$$

Speed of sound:

$$c^2(T,\rho) = \left(\frac{\partial p}{\partial \rho}\right)_s, \tag{4.10}$$

Grüneisen parameter:

$$\gamma(T,\rho) = \frac{1}{\rho}\left(\frac{\partial p}{\partial e}\right)_\rho = \frac{1}{\rho c_v}\left(\frac{\partial p}{\partial T}\right)_\rho. \tag{4.11}$$

4.2. Span-Wagner EOS

As mentioned in section 2.2, the Span Wagen (SW) EOS is the most reliable EOS for pure CO_2. It is seen as the reference EOS. It is derived using experimental data to create a Helmholtz free energy function for CO_2. It is reliable from 217 $[K]$ (This is the triple point temperature) to 1100 $[K]$ and up to 800 $[MPa]$. Munkejord et al. [29] reviewed the SW EOS, and concluded that the density uncertainty is 0.05%. The uncertainty in vapour pressure is even lower with 0.006% and the speed of sound uncertainty is 1.0%. Nevertheless, it is very complicated with 51 terms, especially when also calculating the differentials. The full equation can be seen in Equation 4.12. The equation is a function of τ and δ. Here, τ is the inverse reduced temperature, T_c/T, and δ is the reduced density, ρ/ρ_c. T_c and ρ_c are constants, respectively 304.1282 $[K]$ and 467.6 $[kg/m^3]$. This means that every other parameter can be calculated with the temperature and density at every stage. All the constants and formulas can be found in the original paper of Span & Wagner [28].

It is derived by expressing the Helmholtz free energy into a part for the ideal-gas behaviour, ϕ^0, and a part that takes into account the fluid behaviour of CO_2, ϕ^r. It should be noted that ϕ^r is determined empirically. It is determined with 150 experimental data points. Experiments done after inventing the relationship have shown that it is accurate from 217 to 1100 $[K°]$ and up to 800 $[MPa]$.

$$\begin{gathered}
\frac{a(\rho,T)}{RT} = \phi(\tau,\delta) = \phi^0(\delta,\tau) + \phi^r(\delta,\tau),\\
\phi^0(\tau,\delta) = \ln(\delta) + a_1^0 + a_2^0\tau + a_3^0\ln(\tau) + \sum_{i=4}^{8} a_i^0 \ln[1-\exp(-\tau\theta_i^0)],\\
\phi^r(\tau,\delta) = \sum_{i=1}^{7} n_i\delta^{d_i}\tau^{t_i} + \sum_{i=8}^{34} n_i\delta^{d_i}\tau^{t_i}\exp\left(-\delta^{c_i}\right) +\\
\sum_{i=35}^{39} n_i\delta^{d_i}\tau^{t_i}\exp\left(-\alpha_i\left(\delta-\varepsilon_i\right)^2 - \beta_i\left(\tau-\gamma_i\right)^2\right) + \sum_{i=40}^{42} n_i\Delta^{b_i}\delta\exp\left(-C_i(\delta-1)^2 - D_i(\tau-1)^2\right).
\end{gathered} \tag{4.12}$$

4.3. Peng-Robinson EOS

A more simplified EOS is the PR EOS. This is a standard cubic EOS, used to predict the behaviour of real gases. It was developed by Peng and Robinson in 1976 to improve the accuracy of phase behaviour predictions [25]. The model can be tuned accordingly to a substance or a mixture of substances. It allows for phase transitions between liquid and gas, and is a reliable EOS below the critical point. The PR EOS is given in Equation 4.13 to Equation 4.16, and in Table 4.1 are the tuned values corresponding to CO_2.

$$
\begin{aligned}
p(T, V_m) =& \frac{RT}{V_m - b} - \frac{a_{PR}}{V_m^2 + 2bV_m - b^2}, && (4.13)\\
a_{PR} =& 0.45723553 \cdot R^2 \frac{T_c^2}{p_c} \left[1 + \kappa \left(1 - \sqrt{\frac{T}{T_c}}\right)\right]^2, && (4.14)\\
b =& 0.07779607 \cdot R \frac{T_c}{p_c}, && (4.15)\\
V_m =& \frac{M}{\rho}. && (4.16)
\end{aligned}
$$

Table 4.1: CO_2 values for the PR EOS

Quantity	Value
$P_c\,[Pa]$	7.3773e6
$T_c\,[K]$	304.1282
$R\,[J/(K \cdot mol)]$	8.314
$M\,[g/mol]$	44.095
$\kappa\,[-]$	0.706477

4.4. Determining primitive variables

In order to calculate all primitive variables, it is needed to know if the fluid is in single phase or two-phase conditions. To distinguishing single phase from two-phase flow, the following approach is applied. For the temperature of the previous step, T, the saturation gaseous and liquid densities, ρ_g and ρ_l, are computed. If $\rho_g < \rho < \rho_l$, then the cell is in two-phase flow, otherwise it is single phase flow, as seen below:

$$
\boldsymbol{Phase} = \begin{cases} \text{if } \rho_g \leq \rho \leq \rho_l, & \text{Two-phase ,} \\ \text{else,} & \text{Single phase .} \end{cases} \tag{4.17}
$$

With every time-step, the state vector **U** is established. This means that always, the (mixture) density, ρ, the momentum, ρv, and the total energy, E are known. The velocity is simply the momentum divided by the (mixture) density. From the total energy, the (mixture) specific internal energy, e, can be found with Equation 4.18.

$$
E = \rho(e + \frac{1}{2}v^2). \tag{4.18}
$$

For the $DF4$ and $TF5$ models, the state vector contains $\alpha_g \rho_g$ and $\alpha_l \rho_l$. With these variables known, the (mixture) density is simply found with Equation 4.19.

$$
\rho = \alpha_g \rho_g + \alpha_l \rho_l. \tag{4.19}
$$

4.4.1. Single phase regime

For single phase flow, the density, velocity, and specific internal energy are given at each time step. Using an optimization algorithm, the temperature can be determined from Equation 4.20. The variables denoted with a tilde are known, leaving the temperature as the only unknown. Once the temperature and density are determined, all other primitive variables (e.g., pressure, speed of sound, etc.) can be computed. This approach is applicable to all three models ($DF3$, $DF4$, $TF5$).

$$
e(\tilde{\rho}, T) = \tilde{e}. \tag{4.20}
$$

4.4.2. Two-phase regime HEM & $DF3$ model

When the flow is in multiphase, the "behaviour" of the flow becomes more complex. The gas phase and the liquid phase are both present and are in equilibrium in the two-phase zone. After every time step, the mixture density is known. If the density is between the saturated gas and liquid density, then it is concluded that the fluid is in two phase. Now the unknowns are $\rho_g, \rho_l, \alpha_g, T$. In order to calculate these unknowns, four independent equations are needed. These four equations are given in Equation 4.21 to Equation 4.24. Since the specific energy and density are used to compute the other parameters, this phenomenon is referred to as UV flash [1].

The first equation describes the mixture density. Here the density is known from the mass transfer equation, from Equation 3.6. The second equation is the mixture internal energy. Again, the density is known and the total energy is known from Equation 3.6. The specific internal energy is than calculated with Equation 4.18. Thirdly, the pressures of both states are equal to each other. Therefore, the pressure of the liquid is the same as the pressure of the gas, however, the pressure is unknown. Finally, it is assumed that the states are in the lowest energy state, so the Gibbs free energy of both phases are the same. It is observed that there are four unknowns ($\alpha_g, \rho_g, \rho_l, T$) and four equations. All known variables are indicated with a tilde. In section 6.3 is explained how the set of equations is solved.

$$\alpha_g \rho_g + (1-\alpha_g)\rho_l = \tilde{\rho}, \tag{4.21}$$

$$\alpha_g \rho_g e_g(\rho_g, T) + (1-\alpha_g)\rho_l e_l(\rho_l, T) = \tilde{\rho}\tilde{e}, \tag{4.22}$$

$$p_g(\rho_g, T) - p_l(\rho_l, T) = 0, \tag{4.23}$$

$$G_g(\rho_g, T) - G_l(\rho_l, T) = 0. \tag{4.24}$$

4.4.3. Two-phase regime HEM & $DF3$ model, second method

Another method for calculating the primitive variables is through a reduced UV flash. This method has been suggested by Kumar et al. [47]. Because of the assumption that the phases are in chemical equilibrium, the phases are on the saturation line and the densities can easily be calculated with the temperature. This reduces the number of unknowns only to the temperature T, as ρ_l and ρ_g are both functions of T.

$$\tilde{\rho} = \rho_g \alpha_g + \rho_l(1-\alpha_g) \quad \Rightarrow \quad \alpha_g = \frac{\rho_l(T) - \tilde{\rho}}{\rho_l(T) - \rho_g(T)}. \tag{4.25}$$

Substituting the expression for α_g into the mixture internal energy equation, the only unknown becomes T. This becomes the following equation, and can easily be solved with an optimization routine:

$$\alpha_g \rho_g(T) e_g(\rho_g(T), T) + (1-\alpha_g)\rho_l(T) e_l(\rho_l(T), T) = (\tilde{\rho}\tilde{e}). \tag{4.26}$$

4.4.4. Two-phase regime $DF4$ model

With the $DF4$ model, as mentioned before, the assumption of chemical equilibrium is not valid anymore. This results in the Gibbs free energy being not the same for both phases. On the contrary, because of the two mass equations, it is possible to describe an algebraic equation for both densities. Looking at Equation 4.27 to Equation 4.30, the first equation states that $\alpha_g \rho_g$ is equal to the first entry of the state vector. While it is not known what the individual values are, it is known at every time-step what the product of those variables is through the state vector. The same principle is applied for the second equation. The third and fourth equation are the same as for the $DF3$ model. Again, all known variables are indicated with a tilde. The four unknowns, which are identical to the problem with the $HEM/DF3$ model, are then solved, with the method being explained in section 6.3.

$$\alpha_g \cdot \rho_g = (\alpha_{\tilde{g}} \rho_g), \tag{4.27}$$

$$(1-\alpha_g) \cdot \rho_l = (\alpha_{\tilde{l}} \rho_l), \tag{4.28}$$

$$\alpha_g \rho_g e_g(\rho_g, T) + (1-\alpha_g)\rho_l e_l(\rho_l, T) = \tilde{\rho}\tilde{e}, \tag{4.29}$$

$$p_g(\rho_g, T) - p_l(\rho_l, T) = 0. \tag{4.30}$$

[1] In literature, U stands for internal energy, and V for specific volume

4.4.5. Two-phase regime $TF5$ model

In the $TF5$ model, an additional unknown is introduced, resulting in different temperatures for both phases. Due to the presence of two energy equations, the total energy and total energy for the gaseous phase are calculated at each time step. With this information, the specific energies can be calculated with Equation 4.18 and Equation 4.31.

$$E_g = \alpha_g \rho_g (e_g + \frac{1}{2} v^2). \tag{4.31}$$

Consequently, there are five unknowns and five corresponding equations, allowing the system to be solved. The system of equations is presented in Equation 4.32 to Equation 4.36. The first two equations are analogous to those in the $DF4$ model, as the mass transfer equations remain unchanged. Additionally, the pressure is assumed to be equal in both phases. The third and fourth equations arise from the additional energy equation. The five unknowns are ρ_g, ρ_l, T_g, T_l, and α_g. Again, all known variables are indicated with a tilde. The solution methodology for this set of equations is detailed in section 6.3.

$$\begin{aligned}
\alpha_g \cdot \rho_g &= (\widetilde{\alpha_g \rho_g}), &(4.32)\\
(1-\alpha_g) \cdot \rho_l &= (\widetilde{\alpha_l \rho_l}), &(4.33)\\
e_g(\rho_g, T_g) &= \frac{\tilde{E_g}}{(\widetilde{\alpha_g \rho_g})} - \frac{1}{2}\tilde{v}^2, &(4.34)\\
\alpha_g \rho_g e_g(\rho_g, T_g) + (1-\alpha_g)\rho_l e_l(\rho_l, T_g) &= \tilde{\rho}\tilde{e}, &(4.35)\\
p_g(\rho_g, T_g) - p_l(\rho_l, T_l) &= 0. &(4.36)
\end{aligned}$$

4.4.6. Phase diagram

In Figure 4.1, the phase diagram of CO_2 is plotted for the density and internal energy. For each location in the diagram of the saturation space, it is possible to find the volume fractions of each specific condition. While this diagram is specifically for the case where the Gibbs free energy is the lowest (the HEM and $DF3$ case), the same principle applies to all three UV flash methods.

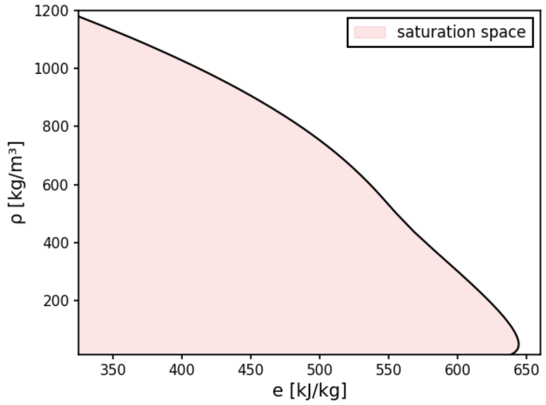


Figure 4.1: Phase diagram for density vs specific internal energy for CO_2. In the saturation space, CO_2 is in the two-phase regime.

4.5. Summary

This chapter provides a comprehensive overview of the thermodynamic framework necessary for modelling two-phase CO_2 flows. It begins with a discussion on the fundamental thermodynamic properties and their interrelations. The chapter details the Span-Wagner (SW) EOS, noted for its exceptional accuracy in modelling pure CO_2 across a wide range of conditions, and the Peng-Robinson (PR) EOS, a simpler cubic model used for phase behaviour predictions.

The determination of primitive variables is explored, with distinct methodologies outlined for single-phase and multi-phase regimes. The chapter discusses the UV flash approach for the three flow models, and its reduced variant, which can simplify the calculations under specific assumptions.

5

Numerical Method

In order to solve the transport equations, numerical methods are used to determine solutions for PDEs. This chapter delves into the techniques and algorithms used to approximate solutions for the flow equations. It starts with the explanation of the FVM in section 5.1, and the HLLC method in section 5.2 for each model. Then, the stability requirement is explained in section 5.3 and a short description is given for how the derivative of ODEs are computed in section 5.4. Furthermore, in section 5.5, the method is described for calculating the mass transfer from the HEM and $DF3$ models. Lastly, in section 5.6, the discretization of the source term in the $TF5$ & $TF5-2$ models is evaluated.

5.1. Finite Volume Method

The Finite Volume Method (FVM) is a numerical technique used to approximate solutions for differential equations, especially those related to conservation laws, such as the Euler flow equations with the conservation of mass, momentum, and energy. FVM is widely used in computational fluid dynamics, heat transfer, and related fields because of its ability to ensure conservation properties at the discrete level. The key characteristic that differentiates FVM from other methods, such as the finite difference or finite element methods, is that it inherently conserves quantities.

The FVM is based on dividing the computational domain into small, finite control volumes (or cells) and solving the integral form of the governing equations over these volumes. Assuming one-dimensional flow, the control volumes are defined as intervals along the x-axis. Each control volume is assigned a property value, Ψ^n, at its midpoint, representing the average value, ψ, within the volume at a given time step, n. The method description below is mostly based on LeVeque, Chapter 4 [53]. Below is the derivation for one-dimensional:

$$\Psi_i^n = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \psi(x,t_n)\,dx, \tag{5.1}$$

$$\int_{\Delta x_i} \frac{d}{dt}\psi + \int_{\Delta x_i} \frac{d}{dx}\mathbf{F}(\psi)\,dx_i = \int_{\Delta x_i} \mathbf{S}(\psi)\,dx_i, \tag{5.2}$$

$$\frac{d}{dt}\int_{\Delta x_i} \psi\,dx_i + \int_{\Delta x_i} \frac{d}{dx}\mathbf{F}(\psi)\,dx_i = \int_{\Delta x_i} \mathbf{S}(\psi)\,dx_i, \tag{5.3}$$

$$\frac{d}{dt}\int_{\Delta x_i} \psi\,dx_i + \Big[\mathbf{F}(\psi)\Big]_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \int_{\Delta x_i} \mathbf{S}(\psi)\,dx_i. \tag{5.4}$$

The first term (time derivative) is discretized with a first-order forward Euler method. This becomes:

$$\frac{d}{dt}\int_{\Delta x_i} \psi\,dx_i = \frac{\psi_i^{n+1}-\psi_i^n}{\Delta t}\Delta x. \tag{5.5}$$

The flux term is expressed as:

$$\Big[\mathbf{F}(\psi)\Big]_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \mathbf{F}(\psi)_{i+\frac{1}{2}} - \mathbf{F}(\psi)_{i-\frac{1}{2}}, \tag{5.6}$$

and the source term is discretized as:

$$\int_{\Delta x_i} \mathbf{S}(\psi)\, dx_i = \mathbf{S}_i(\psi)\Delta x. \tag{5.7}$$

The discretized equation becomes then:

$$\frac{\psi_i^{n+1} - \psi_i^n}{\Delta t}\Delta x + \mathbf{F}(\psi)^n_{i+\frac{1}{2}} - \mathbf{F}(\psi)^n_{i-\frac{1}{2}} = \mathbf{S}_i(\psi)^n \Delta x. \tag{5.8}$$

Rearranging the known terms to the right hand side and the unknowns to the left hand side, it finally becomes:

$$\frac{\psi_i^{n+1} - \psi_i^n}{\Delta t} + \frac{\mathbf{F}(\psi)^n_{i+\frac{1}{2}} - \mathbf{F}(\psi)^n_{i-\frac{1}{2}}}{\Delta x} = \mathbf{S}_i(\psi)^n, \tag{5.9}$$

$$\psi_i^{n+1} = \psi_i^n - \frac{\Delta t}{\Delta x}\left(\mathbf{F}(\psi)^n_{i+\frac{1}{2}} - \mathbf{F}(\psi)^n_{i-\frac{1}{2}}\right) + \mathbf{S}_i(\psi)^n \Delta t. \tag{5.10}$$

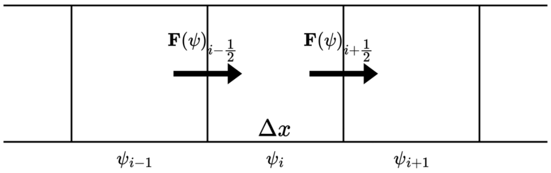

$\mathbf{F}(\psi)|_{i-\frac{1}{2}}$
$\mathbf{F}(\psi)|_{i+\frac{1}{2}}$
Δx
ψ_{i-1}
ψ_i
ψ_{i+1}

Figure 5.1: Illustration of the finite volume method as given in Equation 5.10. The value ψ_i changes with the flux. The source term, S_i, is neglected in the figure.

The flux **F** depends on ψ. However, at the cell boundaries, it is not known what ψ is. Therefore, the determination of the fluxes $\mathbf{F}_{i-\frac{1}{2}}$ and $\mathbf{F}_{i+\frac{1}{2}}$ presents a hurdle when employing the approach provided in Equation 5.10. A possible method to determine the fluxes is using the upwind differencing scheme or central differencing scheme. The problem with these methods is that they cannot handle discontinuous situations correctly. A different method used for handling the discontinuous situations is the HLLC method, known as an approximate Riemann solver.

5.2. HLLC method

The HLLC (Harten, Lax, van Leer, Contact) method is a numerical approach for approximating solutions to the Riemann problem in computational fluid dynamics (CFD), particularly when dealing with systems of hyperbolic conservation laws. It is an extension of the original HLL (Harten-Lax-van Leer) method, where an additional middle wave speed is included. It is known for the good accuracy, giving exact resolutions on shock and contact waves, while still being an easy solver to implement. As stated by Batten et al. [54] 'We further demonstrate that the resulting scheme is positively conservative. This property, which cannot be guaranteed by any linearized approximate Riemann solver, forces the numerical method to preserve initially positive pressures and densities.'. The method is based on the description done by Toro, Chapter 10, [17].

5.2.1. Overview of the HLLC method

The main concept is to consider a wave configuration in which four constant states are separated by three waves for the solution. It is assumed that the middle wave is a contact discontinuity, and the two outer waves are shock waves. In reality, this can be shock waves, rarefaction waves or one of both. Application of the integral form of the conservation laws yields a closed-form approximate formula for the flux, assuming that the wave speeds are provided by some algorithm. This is seen in Figure 5.2. The full derivation is given in Appendix D. The state vector is:

$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right)\cdot \begin{pmatrix} \rho_K \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^*\rho_K + \frac{p_K}{(S_K - v_K)}\right) \end{pmatrix}, \tag{5.11}$$

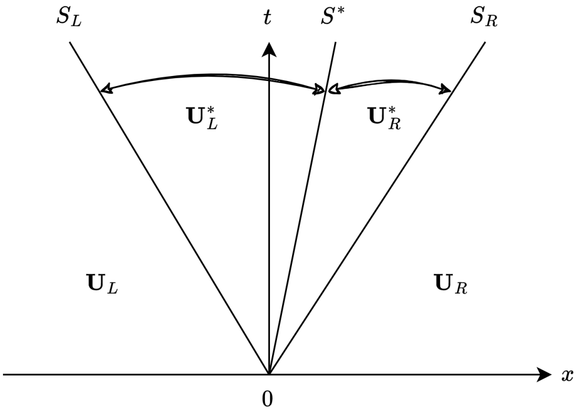


Figure 5.2: HLLC regions defined by the wave speed S_L, S^*, S_R.

with S_L and S_R being:

$$S_L = \min(u_L - c_L, u_R - c_R), \tag{5.12}$$

$$S_R = \max(u_L + c_L, u_R + c_R). \tag{5.13}$$

5.2.2. HLLC for the $DF4$ model

The $DF4$ model has obviously a different U and thus a different reference state is needed. Since the momentum and energy equations are not modified, the corresponding components of the state vector will remain identical. For the mass transfer, the derivation for the gas equation is given in Appendix D, which is similar for the liquid equation. The state vector then becomes:

$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right) \cdot \begin{pmatrix} \alpha_{g,K}\rho_{g,K} \\ \alpha_{l,K}\rho_{l,K} \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^*\rho_K + \frac{p_K}{(S_K - v_K)}\right) \end{pmatrix}. \tag{5.14}$$

5.2.3. HLLC for the $TF5$ model

For the $TF5$ model, the first four equations in the state vector identical as Equation 5.14. The fifth term accounts for the energy balance specific to the gaseous phase. The derivation is done in Appendix D. It should be noted that an additional assumption is made, that is $\alpha_g^* = \alpha_g$. The final state vector then becomes:

$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right) \cdot \begin{pmatrix} \alpha_{g,K}\rho_{g,K} \\ \alpha_{l,K}\rho_{l,K} \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^*\rho_K + \frac{p_K}{(S_K - v_K)}\right) \\ E_{g,K} + (S^* - v_K)\left(S^*(\alpha_{g,K}\rho_g)_K + \frac{\alpha_{g,K}p_K}{(S_K - v_K)}\right) \end{pmatrix}. \tag{5.15}$$

5.3. Stability requirement

One crucial aspect of the numerical approach is its numerical stability. The maximum allowable time step, Δt, is constrained by the stability requirement, which is typically determined using the Courant-Friedrichs-Lewy (CFL) condition. Named after its inventors [55], the CFL number is defined as:

$$CFL = \frac{|\lambda|_{max}\Delta t}{\Delta x}. \tag{5.16}$$

Here, $|\lambda|_{max}$ represents the maximum eigenvalue in the simulation, Δt is the time step, and Δx is the spatial resolution. It effectively means that the eigenvalues are computed for every cell in the domain, and the largest eigenvalue is used to enforce the CFL condition. In order to ensure stability in an explicit method, the CFL condition must be satisfied, which requires that the CFL number is less than one. This requirement implies that the time step, Δt, is chosen such that a wave cannot travel further than one grid cell during a single time step. While the time step may vary depending on the conditions in the simulation, it often remains constant in practice. This is because the maximum eigenvalue usually occurs when the fluid is in the liquid phase, as explained in section 3.4. Physically, the CFL condition ensures that wave propagation remains resolved within the grid, avoiding instabilities that could arise if waves travel distances greater than the grid resolution during a single time step.

5.4. Differentials EOS

As seen in chapter 4, the calculation for different primitive variables consists of calculating differentials. Algebraic or symbolic manipulation is impossible to do because of the complexity. Another option could be to use numerical differentiation. However, the drawback hereby is that round-off errors are introduced. A third method, yet less known method, is by using forward or backward automatic differentiation. Hereby, no round-off errors are made and the results are correct up to machine precision. In this project, forward automatic differentiation is used, instead of backward.

Forward Automatic Differentiation (FAD) is a method for computing the derivative of a function by applying the chain rule at each elementary operation in the function's computational graph. It is based on the concept of dual numbers, where each variable is extended to a dual form , $x_\epsilon = x + \epsilon\dot{x}$, with ϵ being infinitely small and $\epsilon^2 = 0$. As a result, both the value of the function and its derivative are propagated simultaneously. At each step of the computation, the real part gives the function value, while the infinitesimal part tracks the derivative. For instance, if a function involves operations like addition, multiplication, or transcendental functions (e.g., sine or exponentials), FAD computes the derivative of each operation by leveraging the chain rule and combining the derivatives of the inputs. This yields exact derivatives up to machine precision for each intermediate step, effectively providing the final derivative of the overall function without explicit symbolic manipulation. For further reading, the book by Iri & Kubota [56] and article by Fang et al. [57] are recommended.

5.5. Calculating mass transfer for the HEM & $DF3$ model

In the HEM and $DF3$ models, the bulk mass conservation equation, as given in Equation 3.1, is equal to zero as the total mass in the system needs to be conserved. However, since it represents the bulk equation, it sums the contributions of both phases, as shown in Equation 5.17. While these models incorporate mass transfer between phases, the exact details of the transfer are not explicitly specified. If the phase densities, phase volume fractions, and velocity, $\rho_g, \alpha_g, \rho_l, \alpha_l, v$, are known for each cell and at every time step, the mass transfer rate Γ can be calculated in reverse.

$$\begin{array}{c} \dfrac{\partial(\rho_g\alpha_g)}{\partial t} + \dfrac{\partial(\rho_g\alpha_g v)}{\partial x} = \Gamma \\ \dfrac{\partial(\rho_l\alpha_l)}{\partial t} + \dfrac{\partial(\rho_l\alpha_l v)}{\partial x} = -\Gamma \\ \hline \dfrac{\partial\rho}{\partial t} + \dfrac{\partial(\rho v)}{\partial x} = 0. \end{array} + \tag{5.17}$$

To calculate Γ, the derivatives in the equations must be discretized using the same numerical approach as in the original flow equations. The time derivative is discretized using the forward Euler method, while the spatial derivative is discretized using the HLLC method. The state vector for the flux computation includes the top two components of Equation 5.14. The final discretization is given in Equation 5.18. The same approach can be applied to compute $-\Gamma$ for the liquid phase.

$$\frac{(\rho_g\alpha_g)_i{}^{n+1} - (\rho_g\alpha_g)_i{}^{n}}{\Delta t} + \frac{\mathbf{F}(\rho_g\alpha_g v)^n_{i+\frac{1}{2}} - \mathbf{F}(\rho_g\alpha_g v)^n_{i-\frac{1}{2}}}{\Delta x} = \Gamma. \tag{5.18}$$

5.6. Discretization non-conservative models ($TF5$ & $TF5-2$)

In order to simulate the non-conservative models, special treatment is required for terms involving derivatives outside the flux. These terms, which involve a constant multiplied by a differential, are handled as source terms using a method proposed by Johnsen & Colonius [58] and later by de Böck et al. [38]. All these terms are put to the right side of the equation and seen as source terms. This is than added up to the equation, the same way as in Equation 5.10. The derivatives are computed using a HLLC-like method, ensuring consistency and reliability based on established practices.

5.6.1. Discretization for the $TF5-2$ model

The method proposed by Johnsen & Colonius [58] and by de Böck et al. [38] has only be applied for the $TF5-2$ model, and thus, only the velocity had to be discretized in the HLLC method. When doing the same steps from Equation 5.19, but now with the velocity instead of the density, it gives Equation 5.20. The steps are done in Appendix D. Note that as mentioned before, $v_{L*} = v_{R*} = S^*$.

$$F_K^* = F_K + S_K(U_K^* - U_K), \tag{5.19}$$

$$v(x,t)^{HLLC} = \begin{cases} v_L, & \text{if } \frac{x}{t} < S_L, \\ \left(\frac{S_L - v_L}{S_L - S^*}\right) S^*, & \text{if } S_L \leq \frac{x}{t} < S_C, \\ \left(\frac{S_R - v_R}{S_R - S^*}\right) S^*, & \text{if } S_C \leq \frac{x}{t} < S_R, \\ v_R, & \text{if } \frac{x}{t} \geq S_R. \end{cases} \tag{5.20}$$

The discretized source term is then Equation 5.21:

$$S(\psi) = (\alpha_i - \phi_i)\frac{1}{\Delta x}\left(v_{i+\frac{1}{2}}^{HLLC} - v_{i-\frac{1}{2}}^{HLLC}\right), \tag{5.21}$$

where:

$$\phi = \alpha_g \alpha_l \frac{\rho_g c_g^2 - \rho_l c_l^2}{\alpha_l \rho_g c_g^2 + \alpha_g \rho_l c_g^l}. \tag{5.22}$$

5.6.2. Discretization for the $TF5$ model

For the $TF5$ model, in addition to the velocity, the gaseous volume fraction, α_g, and the pressure, p, must also be discretized. This is again done with the method of HLLC, and thus Equation 5.19. When discretizing the volume fraction, Equation 5.23 is obtained. α can be the gas or liquid volume fraction, but for the $TF5$ model, it is the gas volume fraction α_g.

$$\alpha(x,t)^{HLLC} = \begin{cases} \alpha_L, & \text{if } \frac{x}{t} < S_L, \\ \left(\frac{S_L - v_L}{S_L - S^*}\right) \alpha_L, & \text{if } S_L \leq \frac{x}{t} < S_C, \\ \left(\frac{S_R - v_R}{S_R - S^*}\right) \alpha_R, & \text{if } S_C \leq \frac{x}{t} < S_R, \\ \alpha_R, & \text{if } \frac{x}{t} \geq S_R, \end{cases} \tag{5.23}$$

The pressure is also discretized with the HLLC method. However, for p_L^* and p_R^*, Equation 5.24 is used (The derivation is done in Appendix D). The discretized pressure is then Equation 5.25. The final source term for the $TF5$ model is Equation 5.27. Note that here $\alpha = \alpha_g$.

$$p_K^* = p_K + \rho_K(S_K - u_K)(S^* - u_K), \tag{5.24}$$

$$p(x,t)^{HLLC} = \begin{cases} p_L, & \text{if } \frac{x}{t} < S_L, \\ (S_L - v_L)(S^* - v_L)\rho_L + p_L, & \text{if } S_L \leq \frac{x}{t} < S_C, \\ (S_R - v_R)(S^* - v_R)\rho_R + p_R, & \text{if } S_C \leq \frac{x}{t} < S_R, \\ p_R, & \text{if } \frac{x}{t} \geq S_R. \end{cases} \tag{5.25}$$

The final source term for the $TF5$ model is Equation 5.27:

$$S(\psi) = v_i p_i \frac{1}{\Delta x}\left(\alpha^{HLLC}_{i+\frac{1}{2}} - \alpha^{HLLC}_{i-\frac{1}{2}}\right) + p_i \phi_i \frac{1}{\Delta x}\left(v^{HLLC}_{i+\frac{1}{2}} - v^{HLLC}_{i-\frac{1}{2}}\right) + (\alpha_i Y_i) v_i \frac{1}{\Delta x}\left(p^{HLLC}_{i+\frac{1}{2}} - p^{HLLC}_{i-\frac{1}{2}}\right). \tag{5.26}$$

Note that here $\alpha = \alpha_g$. ϕ is:

$$\phi = \alpha_g \alpha_l \frac{\rho_g c_g^2 - \rho_l c_l^2}{\alpha_l \rho_g c_g^2 + \alpha_g \rho_l c_g^l}. \tag{5.27}$$

5.7. Summary

This chapter discusses the numerical methods used to solve transport equations, focusing on techniques and algorithms for approximating solutions to PDEs. It begins with an explanation of the Finite Volume Method (FVM), a numerical technique for solving differential equations, in particularly conservation laws. The FVM divides the computational domain into small control volumes and solves the integral form of the governing equations over these volumes. The chapter also introduces the HLLC (Harten, Lax, van Leer, Contact) method, an approximate Riemann solver used in CFD for systems of hyperbolic conservation laws. The HLLC method considers a wave configuration with four constant states separated by three waves, providing accurate solutions for shock and contact waves. The method is applied to the $DF3$, $DF4$ and $TF5$ flow models, with specific state vectors for each.

The chapter further explains the stability requirement using the CFL number to ensure numerical stability, where the time-step Δt must satisfy the CFL condition. It also covers FAD, a method for computing derivatives by applying the chain rule at each elementary operation. Additionally, the chapter describes the calculation of mass transfer in the HEM and $DF3$ models using reverse computation with the conservation equations. Lastly, the discretization of the non-conservative terms in the $TF5$ model is discussed, as special treatment is needed for these terms.

II

MODEL IMPLEMENTATION

6

Implementation in Compute Code

To carry out numerical simulations for the two-fluid model, a custom JULIA code was developed. While the SW and PR EOS were previously implemented by P. Kumar, the code for this project integrates the discretized models discussed earlier. However, problems and non-trivial solutions will be discussed in this chapter. It starts with the flow solver in section 6.1, then the EOS and thermodynamics in section 6.2. In addition, the method of solving the UV flashes is explained in section 6.3 and the implementation of the BCs is described in section 6.4. Finally, the different checks and balances are explained for each model, with their non-trivial solutions.

6.1. Flow solver

The discretization of the two-fluid models is implemented using the state variables in the conservative form. To compute the primitive variables, an iterative approach is used.

Initialization

Firstly, the initial condition (IC) is given in pressure and temperature. Using this information, the density is computed via an optimization solver, as it is known whether the condition is in liquid or gaseous phase. All the other state variables are subsequently calculated as functions of density and temperature.

Primitive variable calculation

With the state vector, all primitive variables need to be calculated. The UV flash method is applied, for each specific model, as explained in section 6.3. Before the calculation of the primitive variables, it is checked if the system is in single-phase or two-phase conditions. This determination is made using the mixture density at the new time-step and the temperature from the previous time-step. If the mixture density lies between the saturation density of the liquid and gas, the system is in two-phase. Otherwise, it is in single-phase. This process is outlined in Equation 4.17. If in single-phase, Equation 4.20 is solved. When convergence has not been reached, the solver calculates the two-phase equations, depending on which fluid dynamics model is used.

State vector update

With the primitive variables computed, the flux is calculated using the HLLC approximate Riemann solver. The fluxes are then used to update the state vector. With the intermediate state vector, intermediate primitive variables are again calculated using the UV flash method, allowing the source terms to be computed accurately. The source terms are then added to the state vector, and the simulation time is incremented. This loop repeats itself until the final time is reached. An overview of the algorithm is given in section 6.6.

6.2. Thermodynamics

The SW EOS & PR EOS were both programmed in the JULIA programming language by P. Kumar. All the derivates in the thermodynamics are computed with the FAD method, as explained in section 5.4. This is very easily done in JULIA with the *ForwardDiff* package. The Grüneisen parameter is calculated with the derivative of p over T, as the EOS are both a function of T:

$$\gamma(T,\rho) = \frac{1}{\rho c_v}\left(\frac{\partial p}{\partial T}\right)_\rho . \tag{6.1}$$

Another option is to use the *Thermopack* library in python [59]. The PR EOS is built in, and all the derivates are easily obtained from the package. While, the SW EOS is not built in, the GERG-2008 EOS is, and this EOS simplifies back to the SW EOS if only CO_2 is used as a component. The EOS is obtained with the following line:

```
eos = multiparam("CO2", eos="GERG2008")
```

6.3. Solving UV flash

In section 4.4, the different methods were mentioned to solve the primitive variables with a set of equations. The method used, is the *Trust Method.* Instead of searching for the minimum along a line (as in line search methods), the trust region method defines a "trust region" around the current solution. This region is a subset of the variable space where the model is considered to be a good approximation of the objective function. Within this trust region, a model quadratically approximates the objective function. The goal is to minimize this model rather than the original function directly. The advantage is that the method is generally more robust than line search methods. Compared to the Newton-Rapson method, the initial guess is less important and can be more reliable, especially in the beginning of the simulation where there are big gradients.

The way this has been implemented is by using the *NLsolve* library in JULIA. Here, it is important to rewrite the set of equations into a mixed complementarity problem. In Equation 6.2, it is done for the $DF4$ model, but the same principle applies to the other UV flashes. The code is given in Listing 6.1. It is found that lowering the xtol and ftol settings does not change the end result, and therefore 1e-4 is chosen. In addition, the method was also changed to ':newton'. For higher CFL values, it did not converge, but for lower CFL values, it gave the same result as using the 'trust region' method.

$$\mathbf{f} = \begin{pmatrix} \alpha_g\rho_g - (\alpha_g\rho_g)^{\mathrm{n+1}} \\ (1-\alpha_g)\rho_l - (\alpha_l\rho_l)^{\mathrm{n+1}} \\ \alpha_g\rho_g e_g(\rho_g,T) + (1-\alpha_g)\rho_l e_l(\rho_l,T) - (\rho e)^{\mathrm{n+1}} \\ p_g(\rho_g,T) - p_l(\rho_l,T) \end{pmatrix} = 0. \tag{6.2}$$

```
using NLsolve

function create_2phase_flash_4eq(rho, e, rhoG_alphaG, rhoL_alphaL)

    function flashProblem(x)

        T, rho_G, rho_L, alpha_G = x

        F_1 = alpha_G * rho_G - rhoG_alphaG
        F_2 = (1 - alpha_G) * rho_L - rhoL_alphaL
        F_3 = alpha_G * rho_G * InternalEnergy(rho_G, T) +
              (1.0 - alpha_G) * rho_L * InternalEnergy(rho_L, T) - rho * e
        F_4 = pressure(rho_G, T) - pressure(rho_L, T)

        return [F_1, F_2, F_3, F_4]
    end

    flashProblem
end
```

```

flashFunc = create_2phase_flash_4eq(rho, e, rhoG_alphaG, rhoL_alphaL)
u_guess = [T, rho_G, rho_L, alpha_G]     ### Values from last time step
result = nlsolve(flashFunc, u_guess; xtol = 1e-4, ftol = 1e-4,
    method = :trust_region, autodiff= :forward, show_trace = false)
T, rho_G, rho_L, alpha_G = result.zero
```

Listing 6.1: UV flash for *DF*4 model in JULIA

6.4. Boundary Conditions

Two boundary conditions (BCs) were implemented for the simulations: the reflective boundary condition and the outflow boundary condition.

The reflective BC assumes an impermeable wall where the fluid velocity at the wall is zero. The state vector of the ghost cell is the same as the state vector of the previous cell. The only difference is that the momentum is inverted. It is shown in Equation 6.3.

$$\begin{cases} \boldsymbol{U}_{i+1}^{continuity} & = \boldsymbol{U}_{i}^{continuity}, \\ \boldsymbol{U}_{i+1}^{momentum} & = -\boldsymbol{U}_{i}^{momentum}, \\ \boldsymbol{U}_{i+1}^{energy} & = \boldsymbol{U}_{i}^{energy}. \end{cases} \tag{6.3}$$

The outflow BC assumes that the state vector of the ghost cell is identical to that of the adjacent cell. This assumption is based on the observation that, in the test cases, the pipeline is ruptured, allowing CO_2 fluid to escape. Consequently, a rarefaction wave propagates within the pipeline. Given that this wave does not constitute a discontinuity and considering the proximity of the cells, it is reasonable to assume that the properties within the ghost cell remain largely unchanged. The exception to this is the pressure, which is set to atmospheric pressure. The implementation of this BC is seen in Listing 6.2. Note that only the relevant code necessary to understand the BC is provided.

```
nx = 402 ### number of cells + ghost cells

rho_1, v_1, E_1, p_1, ... = compute_primitive_vars_3eq(U[:, 1])
rho_2, v_2, E_2, p_2, ... = compute_primitive_vars_3eq(U[:, 2])

prim_1 = [rho_1, v_1, E_1, p_1, ...]
prim_2 = [rho_2, v_2, E_2, p_2, ...]

for i = 2:(nx - 1)
    rho_3, v_3, E_3, p_3, ... = compute_primitive_vars_3eq(U[:, i+1]

    if i == nx-1
        p_3 = p_amb
    end

    prim_3 = [rho_3, v_3, E_3, p_3, ...]

    F_l = HLLC(prim_1, prim_2)
    F_r = HLLC(prim_2, prim_3)

    prim_1 = prim_2
    prim_2 = prim_3
end
```

Listing 6.2: Outflow boundary condition in JULIA

6.5. Checks & Differences between models

The HEM & $DF3$, $DF4$ and $TF5$ models all had there own problems, and so different programming tricks were used in order to make the models work and get results without (or with as little as possible) numerical fluctuations.

6.5.1. HEM & $DF3$ model

The first model, comprising the HEM and $DF3$ models, is extensively utilized in the literature due to its relative simplicity and reliable predictive capability. However, the experimental test cases present significant challenges, particularly due to the pressure difference exceeding a factor of 100. Notably, during the initial stages when the system transitions to a two-phase state, convergence becomes more difficult. This difficulty arises from the initial guesses for the four unknowns in the thermodynamics algorithm (see subsection 4.4.2).

Initially, the mixture density of CO_2 was employed as the initial guess for both the liquid and gaseous densities. This approach was successful for test cases with the highest initial temperatures in the pipeline. However, the three tests with the lowest temperatures failed to converge. It was subsequently determined that using the initial pipeline density, ρ^0, as the initial guess for the liquid density was effective. For the initial guess of the gas density, the density of CO_2 was calculated based on ambient pressure and temperature. This method proved successful for all experimental test cases.

```
rho_L_guess = rho_0
rho_G_guess = rho_amb
```

Another problem was that the solution of the thermodynamics was not always physical. This is shown in Figure 6.1. From this graph, it was observed that a solution is found for the temperature of 200 $[K]$. However, as mentioned before, the SW EOS is only reliable from 217 $[K]$ up to 1100 $[K]$. Therefore, if a solution is found outside of the domain, the converged solution can not be seen as reliable and correct. The simple approach is then to disregard the solution, assume that the four unknowns are exactly the same as the previous time-step, and try again for the next time-step. This proved to work correctly, and the right solution could be found for the next step.

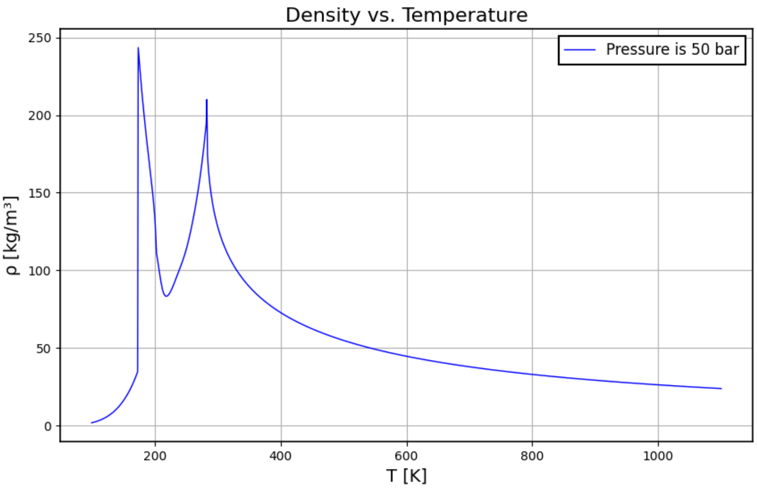


Figure 6.1: Density vs Temperature for the SW EOS at 50 bar pressure. It is shown that for a density, multiple solutions for the temperature exist, what can lead to complications.

6.5.2. $DF4$ model

In the $DF4$ model, an initial guess is similarly required. Initially, the same value, ρ^0, was used for both the liquid and gaseous density guesses. However, this approach resulted in identical densities for both phases, such as a liquid and gaseous density of 900 $[kg/m^3]$. This outcome was inconsistent with the thermodynamics of the previous model, which required the phase densities to match the mixture density, indicating a single-phase flow. To address this issue, the initial values for the two-phase flow from the $DF3$ model were used as the guess values for each cell. Subsequently, the values from the preceding time step were found to be adequate for the initial guess.

Despite the fact that the correct results have been obtained using the Friedel friction model in the literature, it will be demonstrated in section 8.2 that the Churchill method of computing the friction has a negligible influence on the overall results. Subsequently, the Churchill friction method was employed, yielding accurate results. Therefore, it is recommended to use the Churchill friction method due to its ease of implementation.

6.5.3. $TF5$ model

It was observed that α_g, the gas volume fraction, could approach or exceed 1 due to numerical dissipation, which is physically unrealistic and could cause the simulation to fail. To prevent this, a conditional check was implemented to round α_g to one if it was close to this value within a predefined tolerance. The threshold for this adjustment was set to 1×10^{-4}, although a lower threshold is recommended for longer simulations. In JULIA, this adjustment can be implemented as follows:

```
    if isapprox(alpha_g, 0, atol = 1e-4)
        alpha_g = 0
        ...
    else isapprox(alpha_g, 1, atol = 1e-4)
        alpha_g = 1
        ...
    end
```

6.6. Algorithm

An outline of the numerical algorithm for the unsteady compressible two-flow model is provided in this section. The overall algorithm is identical for the $DF3$, $DF4$ and $TF5$ models.

- Set ICs: p^0, v^0 & T^0. Calculate ρ^0 & e^0 and determine the state vector, $\mathbf{U}^0$. In addition, calculate c^0 for the stability requirement.

while $t < t_{final}$

- Calculate Δt with the CFL number

for i **in** $2 : N - 1$

- Calculate primitive variables of $i-1$, i & $i+1$.
  - If in single-phase,
    - ◇ With the state vector $\mathbf{U}$, calculate the density ρ, velocity, v, and specific internal energy, e,
    - ◇ Calculate T with the Newton-Rapson method,
    - ◇ Use T and ρ to calculate all other primitive variables.
  - If in multiphase,
    - ◇ With the state vector $\mathbf{U}$, calculate the density ρ, velocity, v, and specific internal energy, e,
    - ◇ Use the respectively UV flash to calculate ρ_g, ρ_l α_g and T or T_g & T_l, according to the flow model,
    - ◇ Use T or T_g and ρ_g to calculate all other primitive variables.
- Calculate fluxes using HLLC method.
- Update new state vector $\mathbf{U}^{\frac{1}{2}}$ with $\mathbf{F}$.
- Store maximum wavespeed.
- If $DF3$ or $DF4$ model, calculate primitive variables i,

```
        – Calculate source term(s).
        – Update final state vector **U** with **S**.
    • If $TF5$ model, calculate primitive variables $i-1$, $i$ & $i+1$.
        – Calculate source terms.
        – Update final state vector **U** with **S**.
    • Update $\mathbf{U}_1$ & $\mathbf{U}_N$ with their respectively BCs.
    • Write primitive variables to file.
end
    • Post process
```

6.7. Summary

This chapter presents the implementation of numerical simulations for the two-fluid model using the JULIA programming language. The implementation begins with the flow solver, which discretizes the two-fluid models based on state variables in their conservative form. Initial conditions are specified in terms of pressure and temperature, from which density is derived using an optimization solver. Primitive variables are computed using the UV flash method, which determines whether the system is in a single-phase or two-phase regime. The fluxes are calculated with the HLLC method, and the state vector is iteratively updated until the simulation reaches the final time.

The chapter also details the implementation of the SW and PR EOS in JULIA. Forward automatic differentiation (FAD) is employed to efficiently compute thermodynamic derivatives, ensuring high precision. The method for solving UV flashes is explained, utilizing the Trust optimization method implemented with the *NLsolve* library in JULIA to ensure robustness and accuracy.

Boundary conditions are discussed next, covering reflective and outflow boundary conditions. Reflective conditions assume impermeable walls with inverted momentum, while outflow conditions allow for fluid escape with pressure set to ambient levels. The specific implementation details for these conditions are provided.

Finally, the chapter addresses the challenges encountered and solutions devised for the HEM, $DF3$, $DF4$, and $TF5$ models. These include adjustments to initial guesses for thermodynamic variables, handling unphysical solutions, and numerical stabilization techniques. The chapter concludes with an outline of the overall algorithm used for the program, emphasizing its adaptability across different models.

7
Verification

Verification plays a critical role in ensuring the accuracy and reliability of numerical models. This chapter is dedicated to assessing the implemented models by comparing their results with theoretical expectations and established benchmarks. It begins with the verification of single phase in section 7.1. In section 7.2 two-phase flow scenarios are examined, for two different test cases. Additionally, in section 7.3 further explores the convergence behaviour of the mesh and time-step to validate the stability and precision of the numerical schemes. Finally, a two fluid test for the $TF5$ model is presented to evaluate its fidelity in section 7.4.

Table 7.1: Initial values for different verification problems

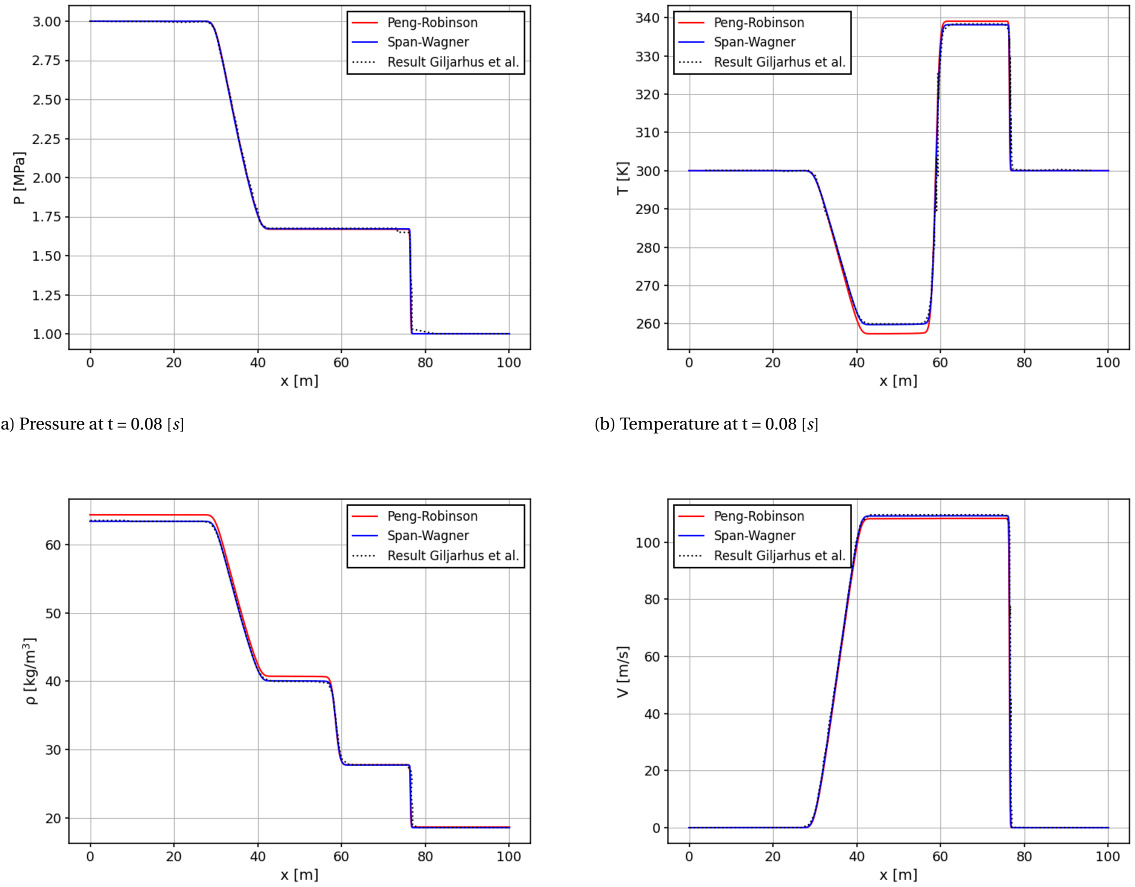
Test case	Pressure left $[bar]$	Pressure right $[bar]$	Temperature $[K]$	End time $[s]$
Giljarhus	30	10	300	0.08
Hammer 1	100	30	250	0.2
Hammer 2	30	1	300	0.06

7.1. Single phase

For a correct evaluation of the output from the model simulation and to ensure reliable outcomes, a proper verification of the results is needed. The verification begins with single phase flow, applying the SW EOS and the PR EOS. A test case inspired by Giljarhus et al. [60] involves a 100 $[m]$ long thermally insulated horizontal pipeline closed at both ends. Initially, the left half of the pipe is set at 30 $[bar]$, and the right half at 10 $[bar]$, with a uniform temperature of 300 $[K]$. The initial conditions are summarised in Table 7.1. Although the reflective BC has been imposed on both ends, the BC does not influence the results as the waves did not reached an end of the pipeline at the final time. At time zero, the diaphragm between the two sections is removed; this is the set-up of the Riemann problem. The CO_2 is only in the gaseous form, and thus the flow is in single phase. With the simulation results of Giljarhus et al. [60] it is possible to verify the working of the SW EOS.

In this test case, the HEM model was used. The test is run for 0.08 $[s]$, with a CFL condition set to 0.9 and 1000 cells, consistent with the original paper [60]. In Figure 7.1, the results are displayed. The outputs—pressure, temperature, density, and velocity—align perfectly with the results presented by Giljarhus et al., confirming that the solver, the implementation of the SW EOS, and the optimization algorithm are accurate.

Seven distinct regions can be identified in Figure 7.1. From 0 to 20 $[m]$, the wave has not yet propagated, and the conditions remain identical to the initial state. Subsequently, a rarefaction wave emerges, resulting in a gradient decline in pressure, temperature, and density. This wave is identified as a rarefaction wave, as it invariably forms in the region with higher pressure. In the third region, a region with uniform flow is observed, representing an intermediate state between waves. In the next region, a contact discontinuity wave is observed. Here, the density exhibit a steep decline and the temperature rises. However, the pressure remains constant. As explained in subsection 3.2.1, a contact discontinuity affects only the density and temperature, not the pressure. In the fifth region, again a region with uniform flow is observed. The sixth region features



(a) Pressure at t = 0.08 [s]

(b) Temperature at t = 0.08 [s]

(c) Density at t = 0.08 [s]

(d) Velocity at t = 0.08 [s]

Figure 7.1: Results of the Giljarhus simulation with HEM, CFL = 0.9, 1000 cells.

the third and final wave, a shock wave, characterized by a steep drop in pressure, density, and temperature, indicating a discontinuity. This shock wave always occurs in the low-pressure region of the Riemann problem. In the final region, the initial condition of the low-pressure region remains unchanged, as the wave has not yet reached this area.

7.2. Multi-phase

The next step is to get the verification of the computational model for a multi-phase problem. The verification process for multiphase flow is inherently more complex due to the need to solve a system of four equations to solve, see Equation 4.21 to Equation 4.24. A simulation has been done by Hammer et al. [31], where the initial conditions where 100 [bar] inside the pipeline and 1 [bar] outside. This simulation is referred to as 'Hammer 1'. At 0.2 [s], the pressure at the pipeline outlet decreased to 30 [bar]. Therefore, to avoid BC effects, the length of the pipeline was doubled, setting the second half to 30 [bar], effectively creating a Riemann problem. The results for the pressure is seen in Figure 7.2. This configuration ensured that the waves did not reach the boundaries within the simulation time, maintaining the integrity of the results. The starting values are also given in Table 7.1.

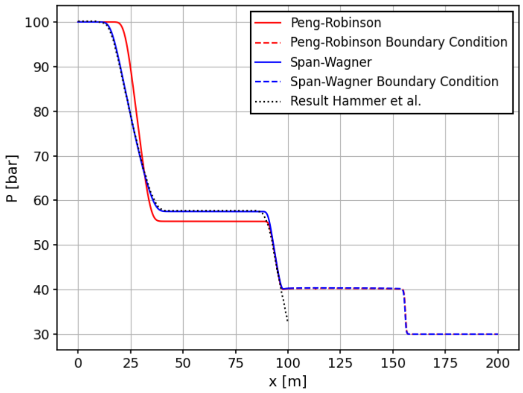


Figure 7.2: Pressure at t = 0.08 [s] with the extended BC shown for the Hammer 1 simulation. The simulation is done with HEM, CFL = 0.5 and 2000 cells.

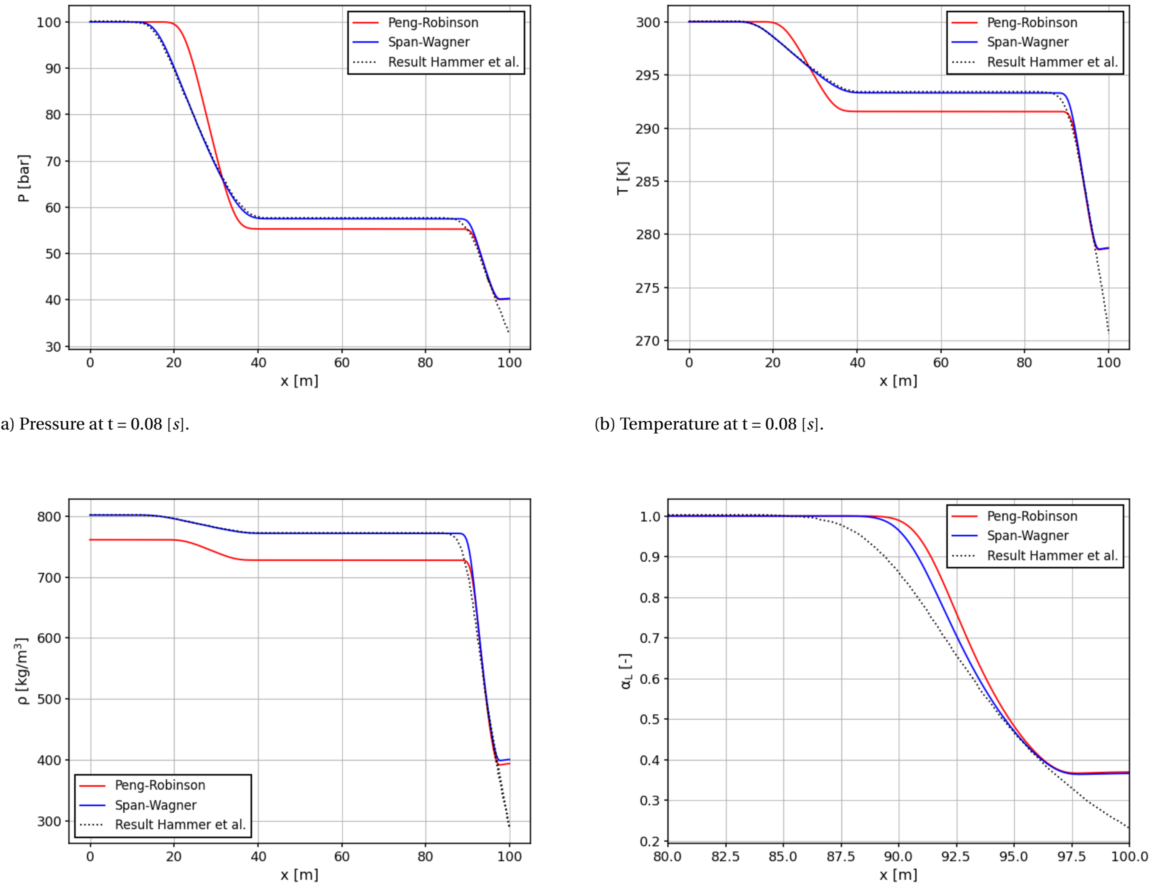
For this test case, the HEM model was used, the CFL condition was set to 0.5, and 2000 cells were used, ensuring 1000 cells in the initially pressurized pipeline, as in Hammer et al.'s original study. The test is run for 0.2 [s]. In Figure 7.3, the results are shown for the pressure, temperature, density and liquid holdup. The observed behaviour aligns closely with the reference results, confirming that the multi-phase flow model is implemented correctly. Minor discrepancies at the boundary are attributed to differences in boundary condition setups.

In Figure 7.3, four distinct regions can be identified. The first region corresponds to the initial state, where conditions remain unchanged. Following this is a rarefaction wave, leading to a decrease in pressure, temperature, and density as it propagates from right to left. In the SW EOS model, this wave travels slightly faster, reaching further along the pipeline after 0.2 [s]. The third region features a plateau between two waves, where the CO_2 remains in a single-phase state at the boiling pressure. This pressure is higher in the SW EOS model, indicating that the PR EOS may not fully capture the two-phase boiling dynamics correctly. Finally, the fourth region represents a slower-moving rarefaction wave, during which boiling occurs, transitioning the CO_2 from a single-phase state to a two-phase system.
A second simulation is done, and compared with the results by Hammer et al. [31]. This problem is referred to as 'Hammer 2'. The initial conditions are 30 [bar] at the left half and 1 [bar] at the right half of the pipeline. The temperature of the whole pipeline was set to 300 [K]. The starting values are also given in Table 7.1.

Again, the HEM model is used. The CFL condition was set to 0.5, in accordance with the paper. 1000 cells were used. Hammer et al. used 4000 cells. However, as will be explained in section 7.3, convergence is still met with 1000 cells, reducing computational effort.

The result of the simulation is displayed in Figure 7.4. The simulation closely resemble Hammer et al.'s findings, with pressure, density, velocity, and temperature profiles following similar trends. A notable difference is the presence of a small hump in the pressure profile, which Hammer et al. attributed to the formation of dry CO_2 ice. This phenomenon, occurring due to low temperature and pressure, results in latent heat release as CO_2 transitions to a solid phase. Since the current model does not account for solid CO_2 formation, the temperature drops further than in Hammer et al.'s study. The results of Hammer et al. are obtained with the use of a graph reading tool[1]. The consequence is that slight differences can occur. This could explain the pressure difference at the beginning of the pipeline. Despite these differences, the overall agreement is strong, verifying the model's performance under two-phase conditions.

[1] https://plotdigitizer.com/app is used for obtaining the results from the graphs



(a) Pressure at t = 0.08 [s].

(b) Temperature at t = 0.08 [s].

(c) Density at t = 0.08 [s].

(d) Liquid holdup at t = 0.08 [s]. Note that the distance is from 80 [m].

Figure 7.3: Results of the Hammer 1 simulation with HEM, CFL = 0.5, 2000 cells.

7.3. Mesh & time-step convergence

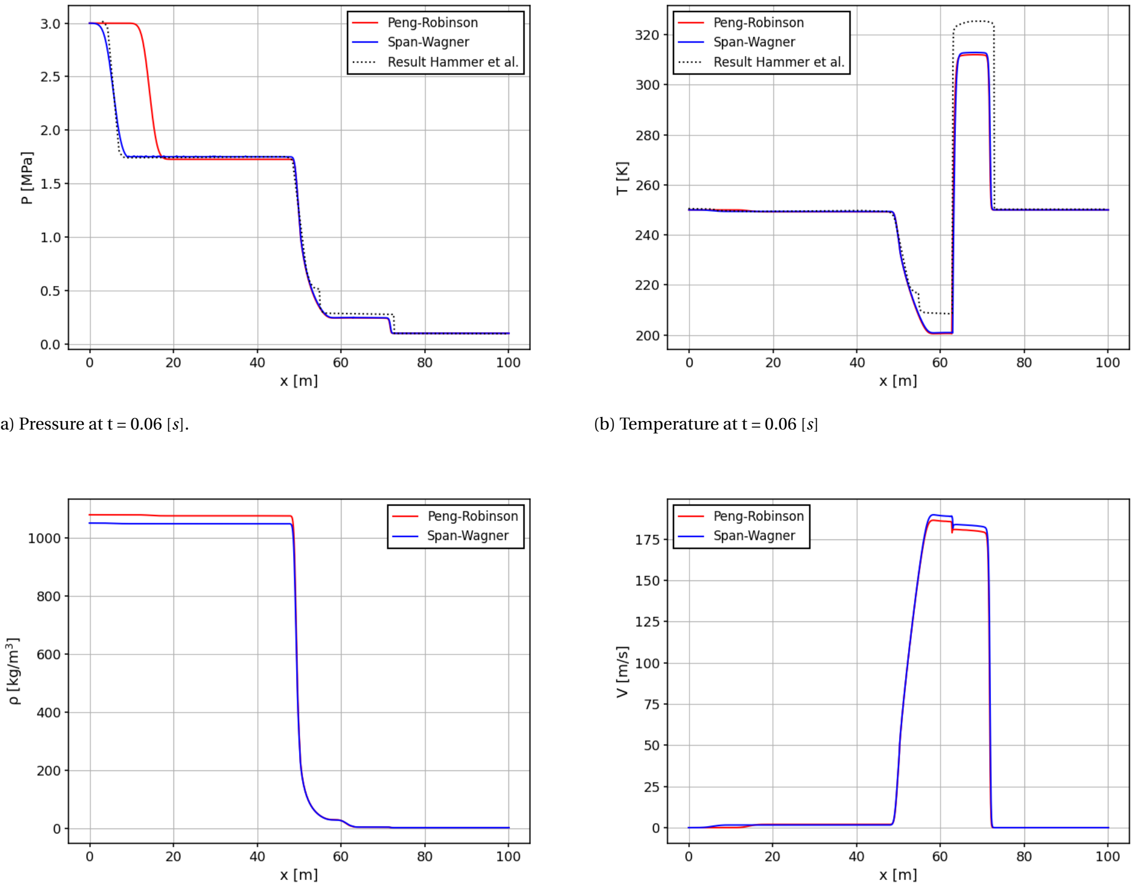
Convergence studies are critical for verifying the correctness and robustness of numerical simulations. For an explicit Euler method, the numerical solution must converge with increasing mesh density and decreasing time step size. This section examines the convergence behaviour of the Hammer 2 test case by varying the mesh size and CFL number. The objective is to identify the optimal configuration that balances accuracy and computational efficiency.

First, a mesh convergence study was conducted using grids with 200, 400, 800, and 1600 cells. The results for pressure and temperature at t = 0.06 [s] are displayed in Figures 7.5a and 7.5b.

The pressure results in Figure 7.5a show that the 200-cell grid deviates significantly from the finer grids, particularly at the rarefaction wave and the shock wave near 70 [m]. This behaviour is expected, as a coarser grid smooths out sharp discontinuities, whereas finer grids capture sharper transitions.
The temperature results in Figure 7.5b further highlight the inadequacy of the 200-cell grid. The coarse grid fails to capture the full flow dynamics, evident in the underestimation of the top plateau and the inaccurate representation of the contact discontinuity at 65 [m]. With finer grids, the temperature rise near 65 65 [m] becomes sharper, indicating improved resolution.

The maximum temperature differences across grid sizes were as follows:

- 1600 cells: Maximum temperature 313.4 [K].
- 800 cells: Maximum temperature 312.6 [K] with a 0.2 % relative difference (absolute difference of 0.8 [K]).



(a) Pressure at t = 0.06 [s].

(b) Temperature at t = 0.06 [s]

(c) Density at t = 0.06 [s]. The cross denotes a slight decrease in density.

(d) Velocity at t = 0.06 [s]

Figure 7.4: Results of the Hammer 2 simulation with HEM, CFL = 0.5, 1000 cells.

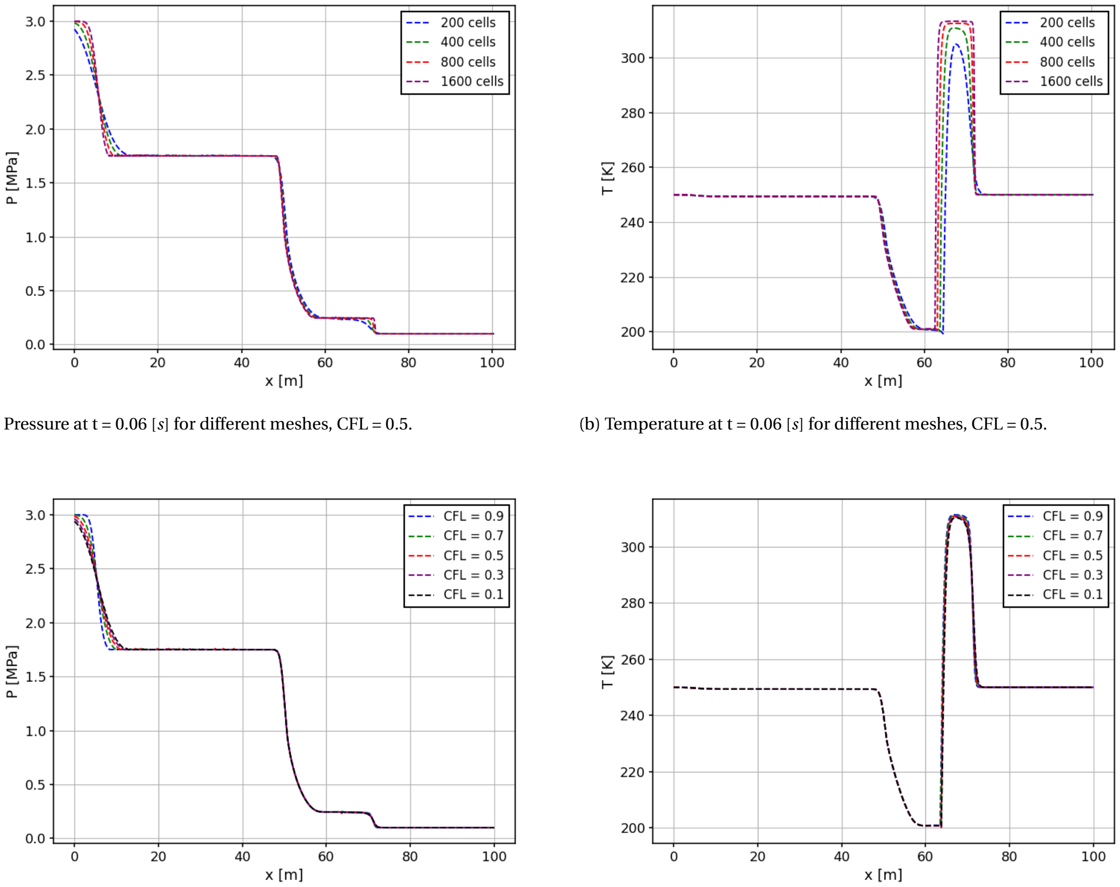
- 400 cells: Maximum temperature 311.1 [K] with a 0.7 % relative difference (absolute difference of 2.3 [K]).
- 200 cells: Maximum temperature 304.0 [K] with a 3.1 % relative difference (absolute difference of 9.4 [K]).

Based on these results, a 400-cell grid is deemed optimal, offering a good trade-off between accuracy and computational cost while maintaining a relative difference below 1%.
Time-step convergence was evaluated by varying the CFL number for a 400-cell grid. The CFL numbers tested were 0.9, 0.7, 0.5, 0.3, and 0.1. Figures 7.5c and 7.5d illustrate the pressure and temperature results, respectively. The results show negligible differences between simulations with different CFL numbers, except for the pressure drop at the beginning of the rarefaction wave. This discrepancy can be mitigated by refining the mesh. Consequently, the highest possible CFL number is used to minimize simulation time without compromising accuracy.

7.4. $TF5$ model

Table 7.2: Initial values for high pressure, high density Sod problem

Fluid	Density [kg/m^3]	Velocity [m/s]	Pressure [Pa]	Liquid Holdup [−]	Gamma [−]
Water	10.0	0.0	10.0	1.0	1.4
Air	0.125	0.0	0.1	0.0	1.6



(a) Pressure at t = 0.06 [*s*] for different meshes, CFL = 0.5.

(b) Temperature at t = 0.06 [*s*] for different meshes, CFL = 0.5.

(c) Pressure at t = 0.06 [*s*] for different CFL numbers, 400 cells.

(d) Temperature at t = 0.06 [*s*] for different CFL numbers, 400 cells.

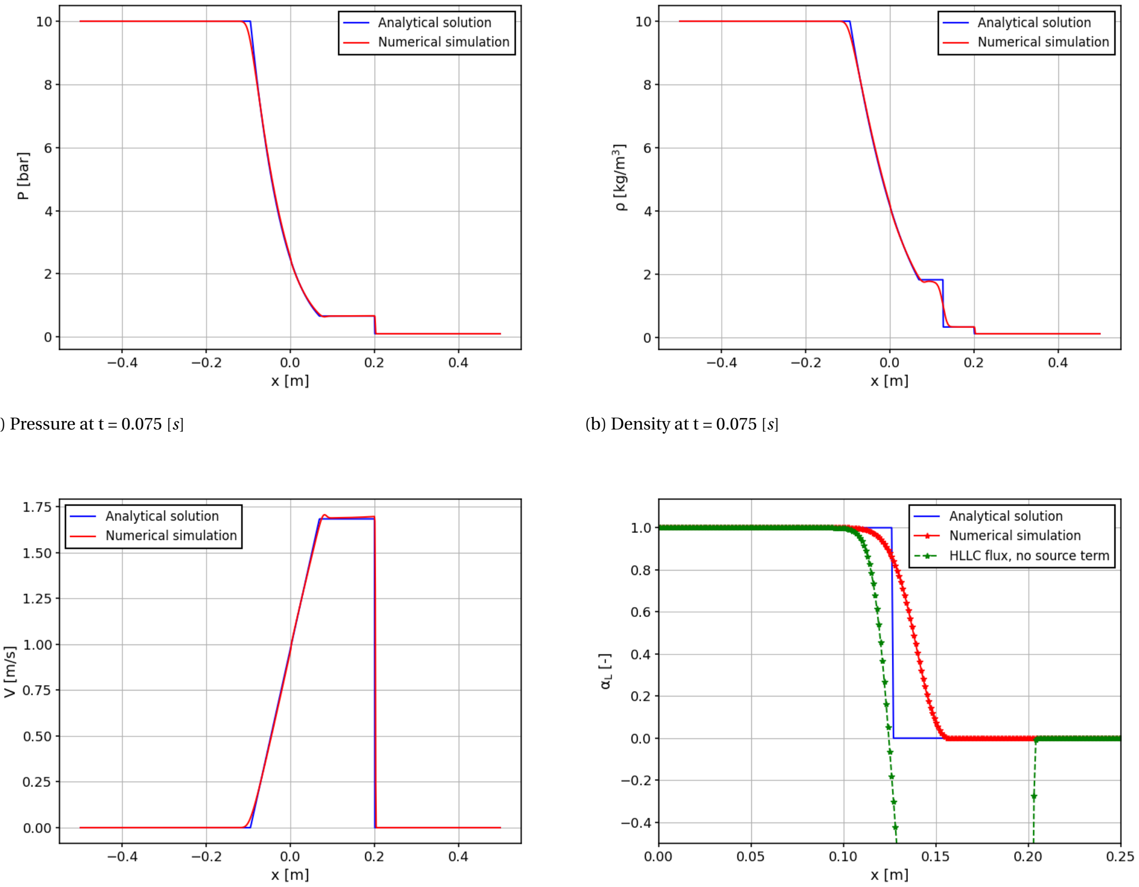
Figure 7.5: Results of the Hammer 2 simulation with *HEM*, for different mesh sizes and CFL numbers.

The $TF5$ model differs significantly from the $DF3$ and $DF4$ models due to its non-conservative formulation, which introduces additional numerical complexities. To verify the numerical implementation, a water vapour-air shock tube problem, commonly referred to as the Sod shock tube problem [61], was used. This setup involves a diaphragm initially separating two regions of differing states in the pipeline, with the left state filled with water vapour, and the right state filled with air. At t=0 [*s*], the diaphragm is removed, initiating the simulation. The initial conditions are outlined in Table 7.2. For simplicity, the ideal gas law was used instead of the PR or SW EOS. This test case consists therefore out of 2 non-mixing gases, so there is no phase change and mass transfer between the fluids.

The verification began with the $TF5-2$ model, as literature has established methods for handling fluxes and source terms using the HLLC solver and HLLC-based approaches, respectively, as detailed in Section 5.6. Once the $TF5-2$ model was validated, the same discretization approach was extended to the $TF5$ model.

The results of the $TF5$ model are shown in Figure 7.6, comparing numerical outcomes to the analytical solution. The pressure, density, and velocity align well with the analytical solution, demonstrating the accuracy of the numerical implementation.

Notably, the liquid volume fraction graph highlights the importance of incorporating the source term. Without the source term, the liquid holdup fraction overshoots significantly, leading to inaccurate results. With the source term included, the simulation accurately reproduces the expected behaviour. This confirms that the source term has been implemented correctly.



(a) Pressure at t = 0.075 [*s*]

(b) Density at t = 0.075 [*s*]

(c) Velocity at t = 0.075 [*s*]

(d) Liquid volume fraction at t = 0.075 [*s*]

Figure 7.6: The Sod shock problem for a water vapour-air combination. The initial conditions are given in Table 7.1. A CFL of 0.9 and 1000 cells have been used.

7.5. Summary

This chapter verified the numerical implementation of the models through a series of test cases. For single phase verification, the Giljarhus test case demonstrated the accuracy of the SW equation of state and the numerical solver, reproducing expected pressure, temperature, density, and velocity profiles.

The multi-phase verification used two Hammer test cases. The results matched the behaviour observed in the literature, verifying the implementation of the multi-phase flow model. Minor deviations in pressure and temperature, attributed to solid CO_2 formation, were noted but did not detract from the overall accuracy.

Mesh and time-step convergence studies for the Hammer 2 test case confirmed the reliability of the numerical solution. A grid of 400 cells and a high CFL number were found to provide an optimal balance between computational efficiency and accuracy.

Finally, the $TF5$ model, with its non-conservative formulation, was verified using the Sod shock tube problem. The results showed strong agreement with the analytical solution, verifying the flux and source term implementations with mass transfer.

In conclusion, the chapter established confidence in the numerical implementation of all models and their ability to accurately simulate single phase and multi-phase flows under various conditions.

III

RESULTS & DISCUSSION

8 Results

This chapter presents the results obtained from the different models for various test cases. Additionally, it examines the differences between these models. The chapter begins by analysing the flow development over time, describing how pressure and temperature evolve in the pipeline during depressurization events. Subsequently, the pressure and temperature results along the pipeline for the $DF3$ model are discussed in section 8.2. These results are also compared with those obtained from OLGA, a state-of-the-art multiphase flow simulator. Furthermore, the results for the $DF4$ model are presented and compared with those of the $DF3$ model in section 8.3. In section 8.4, the pressure variation over time is analysed, with a discussion of its implications. Finally, the temperature evolution over time is detailed in section 8.5.

8.1. Flow development

As mentioned in section 2.5, pipeline depressurization tests have been conducted by Munkejord et al. [21] and Log et al. [22] [23] [24], to quantify the effects of such events. In the tests, the pipeline is initially pressurized to a target pressure. Subsequently, a rupture disk is opened at the left side of the pipeline at t = 0 [s]. The high pressure CO_2 fluid is exposed to the ambient conditions. The starting conditions are shown in Figure 8.1

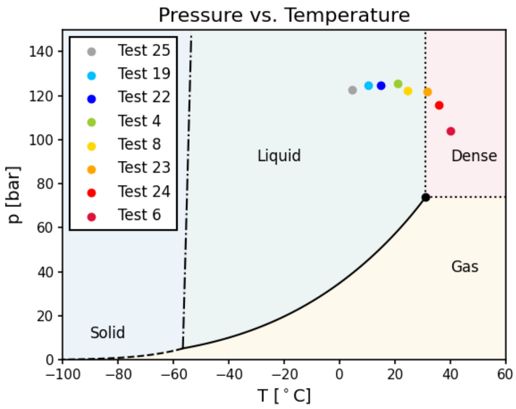


Figure 8.1: Initial conditions of the test-cases on the PT diagram of CO_2. The saturation line and all the regions are calculated with the SW EOS.

Following the rupture, high-pressure CO_2 begins to escape the pipeline. The significant pressure difference between the pipeline and ambient accelerates the flow to Mach 1, reaching the critical outflow speed. A sketch is given in Figure 8.2, where it is seen that the CO_2 escapes at the left end of the pipeline, where the rupture disk is opened. The rapid drop in pressure also causes a significant temperature drop, primarily due to adiabatic expansion as the gas expands rapidly without heat exchange. However, as found in the literature and explained in subsection 2.1.2, the temperature drop is expected to occur on a longer timescale compared to the pressure drop.

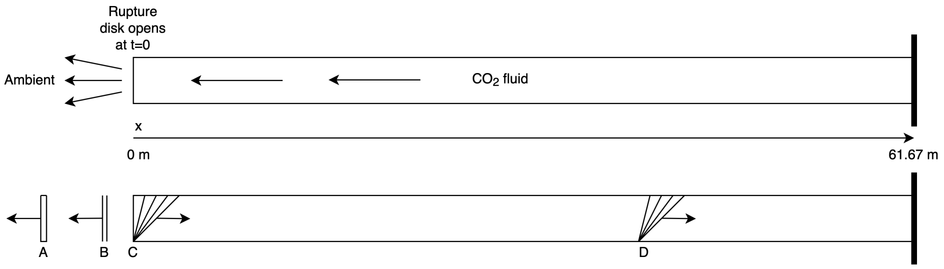


Figure 8.2: Sketch of the flow development in the pipeline. The CO_2 fluid in the pipeline moves from right to left, into the ambient. A represents the shock wave in the ambient, B is the contact discontinuity, likewise in the ambient, C is the slow moving two-phase rarefaction wave and D is the fast moving single phase rarefaction wave, both moving in the opposite direction of the CO_2 fluid.

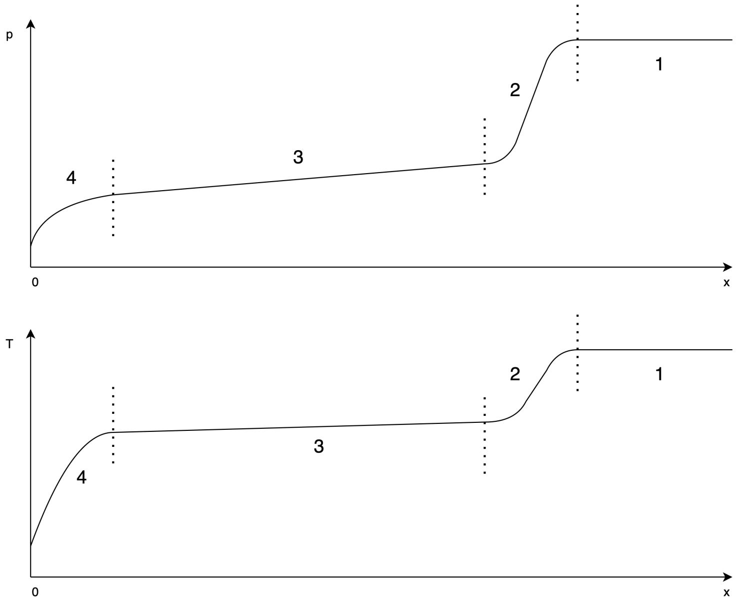


Figure 8.3: Sketch of an expected time-frame in the pipeline. Region 1 is where the IC conditions still are present, region 2 is where the fast moving single phase rarefaction wave drops the pressure and temperature, region 3 is where the plateau for pressure and temperature sits as the flow is chocked, and region 4 is where the slow moving two-phase rarefaction wave drops the temperature and pressure further, as the flow transits into a two-phase regime.

To simulate this scenario, a Riemann problem setup, as described in section 3.2, can be employed. The rupture disk acts as a diaphragm separating two distinct regions: the pipeline conditions on the right and ambient conditions on the left. Due to the higher pressure on the right side, the fluid always flows from right to left. Consequently, the simulation only requires one boundary condition cell for the ambient state, as no information propagates back into the pipeline.

Upon opening the pipeline, three distinct waves are observed:

- **Shock Wave:** This wave propagates into the low-pressure ambient region (Wave 'A' in Figure 8.2). Since it does not affect conditions inside the pipeline, it can be ignored for this analysis.
- **Contact Discontinuity:** This wave travels with the fluid's velocity (Wave 'B' in Figure 8.2). As the fluid exits the pipeline, the contact discontinuity wave also propagates into the ambient region, rendering it irrelevant for the pipeline's internal conditions.
- **Rarefaction Waves:** This wave travels into the high-pressure region within the pipeline and plays a significant role in the results. Two rarefaction waves are produced, a fast moving single phase rarefaction wave (Wave 'D' in Figure 8.2), and a slow moving two-phase rarefaction wave (Wave 'C' in Figure 8.2).

In Figure 8.3, a sketch of the pressure and temperature at a time-frame are sketched. The sketch is divided into four regions. In region 1, nothing has yet happened and the pressure and temperature are still at the

Table 8.1: Speed of sound for the initial conditions. While the speed of sound is different for different conditions, the highest speed of sound is always when the fluid is in a liquid phase.

Test no.	Pressure avg. [bar]	Temperature avg. [°C]	Speed of sound [m/s]
25	122.7	4.6	635.9
19	124.7	10.2	598.0
22	124.8	14.9	563.4
4	125.4	21.1	516.7
8	122.2	24.6	484.0
23	121.9	31.5	428.6
24	115.6	35.8	373.0
6	104.0	40.0	289.6

ICs, as the fast moving rarefaction wave did not yet reached the pipeline end. In region 2, the fast moving rarefaction wave causes a drop in pressure and temperature within the pipeline. These quantities decrease until reaching a stable plateau, which is seen in region 3. This plateau is at the pressure where boiling of liquid CO_2 into two-phase CO_2 begins. The flow is choked and at this point, the mass flow rate is maximized. Further reductions in downstream pressure do not increase the escape rate of CO_2. If friction would not be included, this plateau would be horizontal. Once the slow moving rarefaction wave appears, seen in region 4, the pressure inside the pipeline falls below this plateau, CO_2 transitions into a two-phase regime, resulting in further reductions in pressure and temperature as liquid and vapour phases coexist.

8.2. *DF*3 model

Figure 8.4 to Figure 8.7 shows the simulation results for the experiments of tests 25, 19, 22 and 6 using the *DF*3 model. The results are given for the pressure and temperature at a time of 0.1 [s]. All simulations are done with the friction calculated by the Friedel method, the friction calculated by the Churchill method, and without any friction. The latter is essentially the *HEM* model. The effects of gravity are set to zero, because it is a horizontal pipeline setup. All the simulations are done with a CFL number of 0.95, and with 400 cells. Table 8.1 can be used to check which test number corresponds to which initial condition. These simulation tests were also carried out by Log et al. [24] [22] [23].

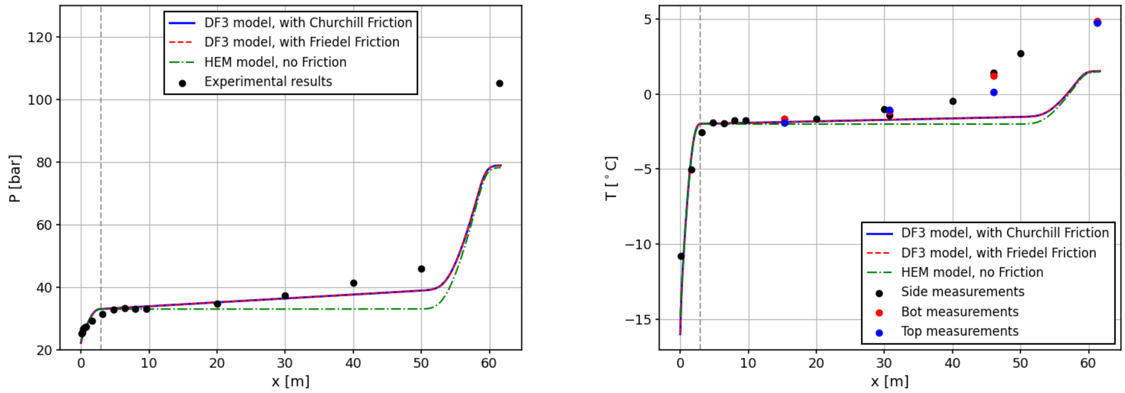
8.2.1. Pressure results

The pressure results for test 25, 19, 22 and 6 are presented in Figure 8.4 to Figure 8.7. The other results are found in Appendix F. The analysis indicates that the difference between the two friction methods is negligible. Therefore, it is recommended to use the Churchill method, as detailed in subsection 3.3.2, due to its significantly simpler implementation compared to the Friedel method.

Additionally, the results of the *DF*3 model with friction align more closely with the experimental data than those of the *HEM* model. This discrepancy is particularly evident in simulations with lower initial temperatures. The plateau in the *DF*3 model is declining from right to left, unlike the horizontal plateau observed in the *HEM* model. This phenomenon is attributed to friction, which causes momentum loss to the environment, resulting in a pressure decrease.

Furthermore, it is observed that the decompression wave propagates faster when the initial conditions have lower temperatures. As shown in Table 8.1, the speed of sound at the initial condition for test 25 is more than twice that for test 6. This is clearly illustrated in Figure 8.4a and Figure 8.7a. In test 6, the pressure did not change beyond 30 meters up to a time of 0.1 [s], whereas in test 25, the pressure wave had already reached the end of the pipeline within the same time-frame. This is expected, as the speed of sound of test 25 is 635.9 [m/s], so after 0.1 [s], the decompression wave would have travelled 63.6 [m], which is more than the length of the pipeline (61.67 [m]).

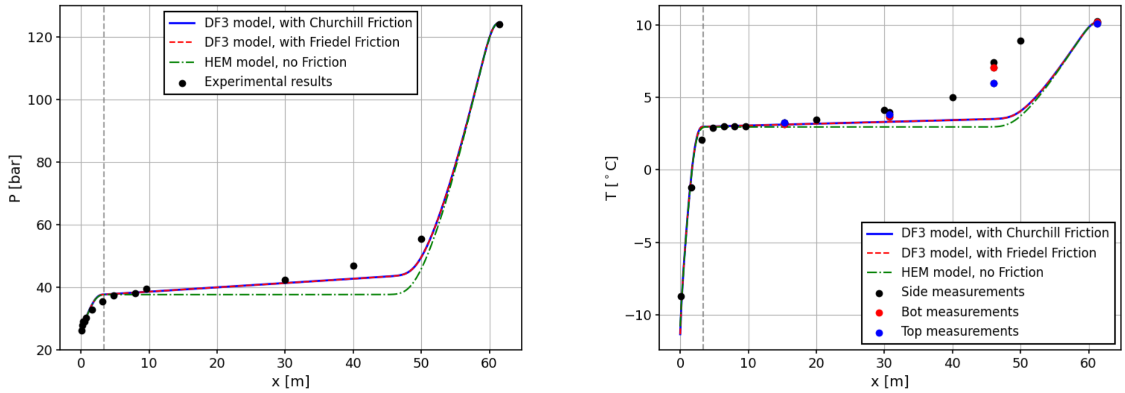
The only mismatch with the experimental result occurs in test 25, shown in Figure 8.4a. A clear explanation for this discrepancy has not been identified. However, experimental data indicates that at 0.1053 [s], the pressure at the end of the pipeline was 80.67 [bar], which aligns with the simulation results. This implies that only 5.3 [ms] were required for the pressure to drop by 20 [bar]. Given this small time-scale, the discrepancy could potentially be attributed to a measurement error. Since this issue does not appear in any of the other tests, it is reasonable to conclude that the mismatch is not caused by a physical phenomenon.



(a) Pressure of test 25 at t = 0.1 [s]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

(b) Temperature of test 25 at t = 0.1 [s]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

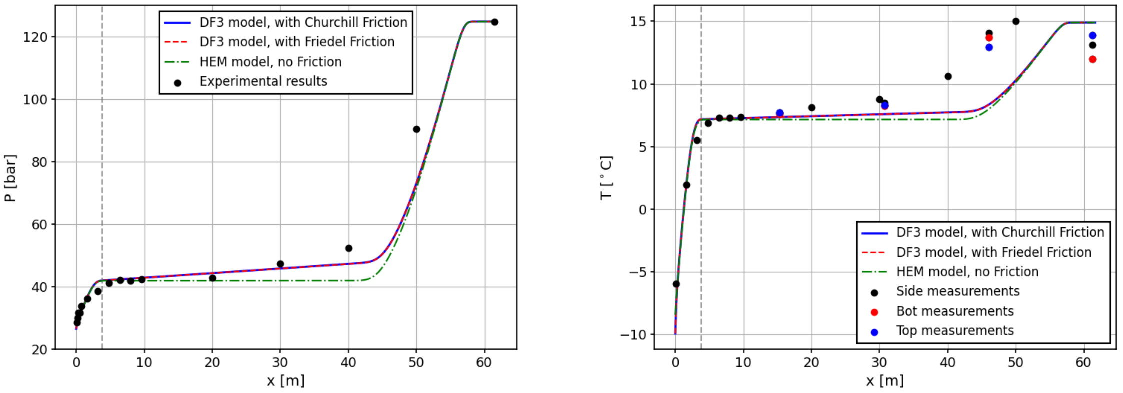
Figure 8.4: Pressure and temperature results of test 25 t = 0.1[s] for the $DF3$ model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 19 at t = 0.1 [s]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

(b) Temperature of test 19 at t = 0.1 [s]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

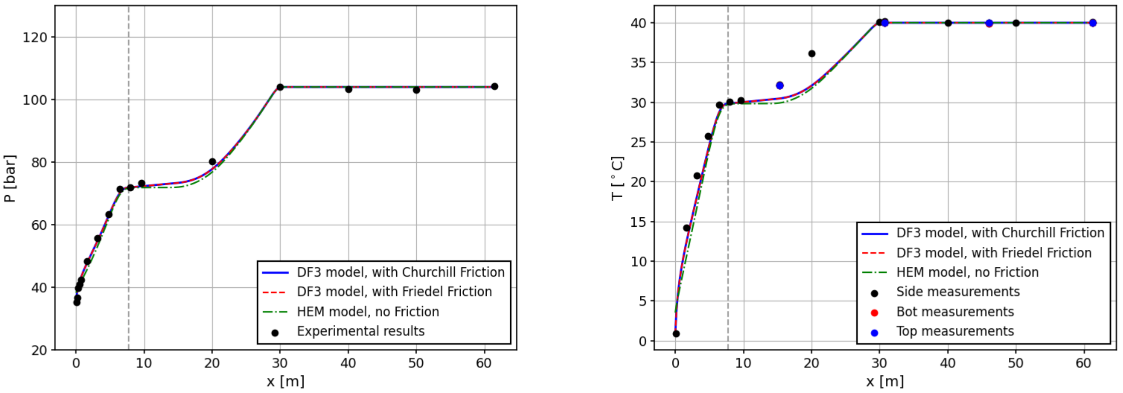
Figure 8.5: Pressure and temperature results of test 19 t = 0.1[s] for the $DF3$ model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 22 at t = 0.1 [*s*]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

(b) Temperature of test 22 at t = 0.1 [*s*]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

Figure 8.6: Pressure and temperature results of test 22 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 6 at t = 0.1 [*s*]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

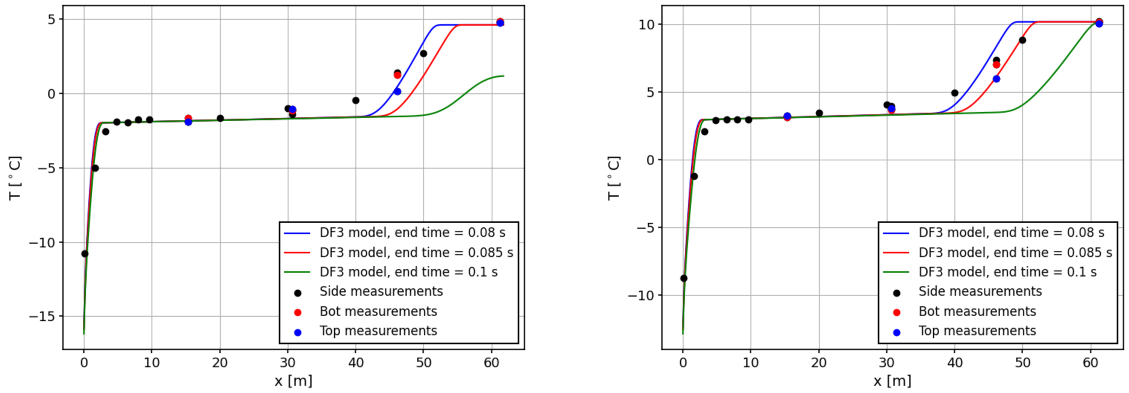
(b) Temperature of test 6 at t = 0.1 [*s*]. The flow left from the vertical gray line is in the two-phase region, right it is in single phase.

Figure 8.7: Pressure and temperature results of test 6 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.

8.2.2. Temperature results

The temperature results for test 25, 19, 22 and 6 are presented in Figure 8.4 to Figure 8.7. The other results are found in Appendix F. The simulation results for temperature do not align well with the experimental data. In all test cases, the temperature wave appears to rise earlier in the simulations than observed experimentally. Interestingly, reducing the simulation runtime to 0.085 [*s*] or 0.08 [*s*] results in better agreement between the simulations and experimental data. This is demonstrated for tests 25, 19, 22 and 6 in Figure 8.8 and Figure 8.9. The other tests are shown in Appendix F.

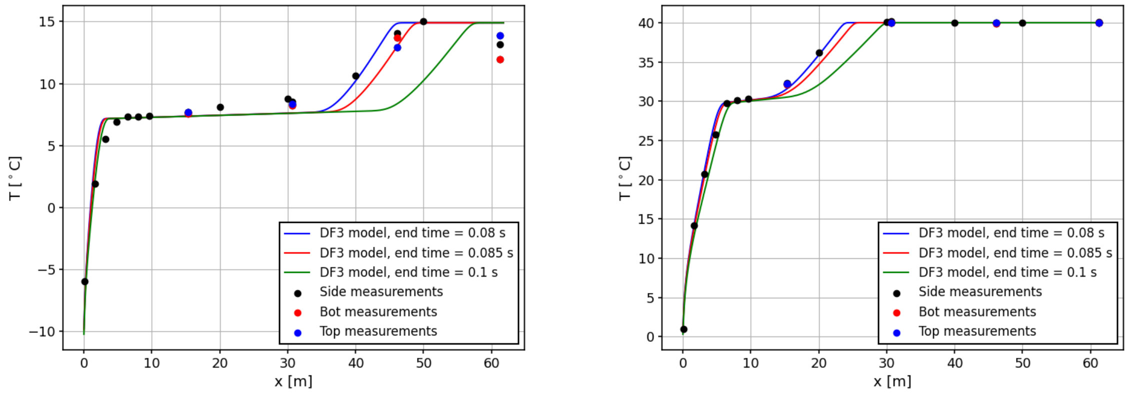
One possible reason for this delay is that the temperature requires more time to respond to pressure changes. As supported by findings in the literature, and mentioned in subsection 2.1.2, temperature typically evolves over a slower time-scale than pressure. However, in the *HEM* & *DF*3 model, this delay is not captured, and the changes happen with the same velocity as the decompression wave. A second reason could be that there is no heat exchange term with the wall and the ambient. However, due to the short simulation time, this effect is likely minimal. Besides, introducing a heat exchange source term would likely result in a declining temperature plateau, analogous to the impact of friction on pressure. A third yet less satisfied reason for this mismatch could be simply that the temperature sensors have a delay in the measuring. These temperature



(a) Different temperature simulations of test 25.

(b) Different temperature simulations of test 19.

Figure 8.8: Different temperature simulations for test 25 and 19, with a CFL number of 0.95 and 400 cells.



(a) Different temperature simulations of test 22.

(b) Different temperature simulations of test 6.

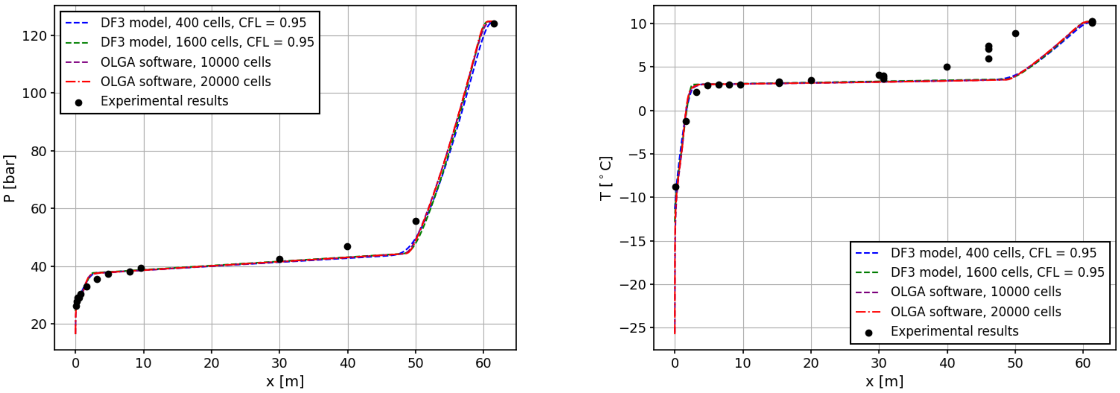
Figure 8.9: Different temperature simulations for test 22 and 6, with a CFL number of 0.95 and 400 cells.

results have not been published by Log et al. [24] and Munkejord et al. [21] for this short time-span, and can therefore not be compared.

In addition the temperature delay, discrepancies arise in measurements at the same location but from different positions within the cross-sectional area of the pipeline. These locations, which have multiple sensors, are situated at the end of the pipeline. Especially in test 25, indicated in Figure 8.4b, and test 19, shown in Figure 8.5b, the temperatures measured at 46 $[m]$ are for three positions in the pipe showing a difference of 1.5 $[°C]$. The dotted grey vertical line in the figures delineates the boundary between two-phase flow (left of the line) and single-phase flow (right of the line). Since these discrepancies occur in the single phase region, they cannot be attributed to two-phase flow. AA plausible explanation could be thermal stratification, where hotter fluid rises, resulting in higher temperatures measured at the top. This is seen in test 22, in Figure 8.6b, at the end of the pipeline. But here, the flow is not disturbed yet. When looking at the temperatures of the disturbed flow, for example in test 19, in Figure 8.5b and test 22, in Figure 8.6b, the side measurement device shows the highest temperature. Again, a clear physical explanation is elusive. Slight measurement errors may also contribute to these discrepancies.

8.2.3. $DF3$ model comparison with OLGA

OLGA is an industry-standard tool used for modelling transient flow behaviour in pipelines and well-bores. It is a numerical tool for simulating one-dimensional multiphase flow and it is used by production, process, and



(a) Pressure of test 19 at t = 0.1 [*s*]

(b) Temperature of test 19 at t = 0.1 [*s*]

Figure 8.10: Pressure and temperature results of test 19 at t = 0.1 [*s*] for the *DF*3 model & OLGA software for a different number of cells.

flow assurance engineers to ensure efficient and safe operations in the oil and gas industry. Version 2024.2 of OLGA [62] was used for these simulations, with the settings configured for no-slip between the phases and adiabatic flow while enabling friction with the pipeline. The results are displayed in Figure 8.10 for test 19 and in Figure 8.11 for test 6. Test 19 and test 6 have been chosen, representing one scenario with an initially high CO_2 temperature in the pipeline and another with an initially low CO_2 temperature.

In test 19, the simulation results of the *DF*3 model perfectly match those obtained using OLGA. Although OLGA used a significantly higher number of grid cells (20,000 vs 1600) due to its advanced optimization capabilities and faster computational speed, this did not yield any noticeable advantage. Both simulations converged to the same solution, confirming the robustness and accuracy of the *DF*3 model.

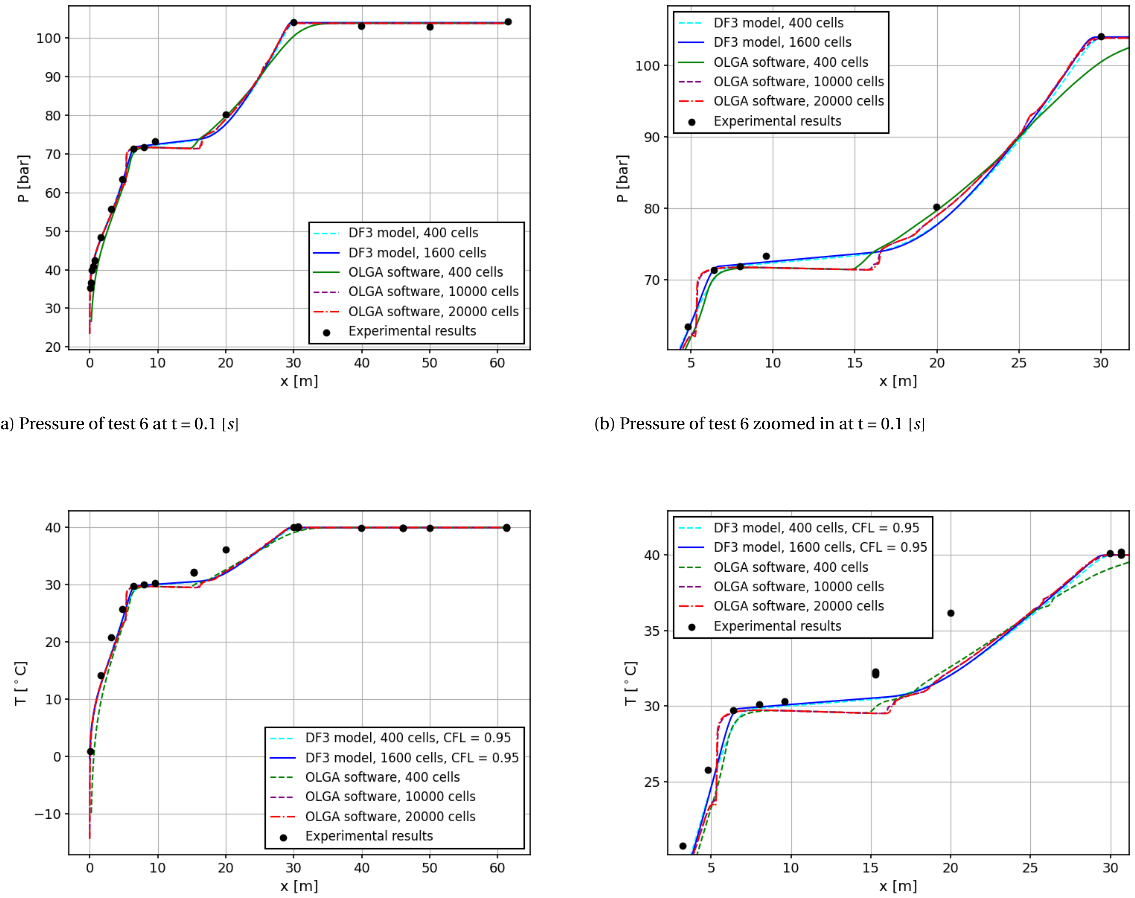
However, in test 6, discrepancies between the *DF*3 model and OLGA simulations were observed. While the overall flow dynamics are similar, notable differences persisted even with increased grid resolution. The first difference, is that while friction is enabled in the OLGA software, it seems that the results are not as expected. With the *DF*3 model, the pressure plateau is slightly increasing from left to right, which is expected. For the OLGA software simulation, the plateau is not increasing, and with closer inspection in the zoomed in pictures, it is even decreasing.

This difference is not due to an issue with friction modelling in OLGA but rather stems from its treatment of conditions near the critical point during depressurization. The reason for this phenomenon with the OLGA software is that with the depressurization of test 6, the conditions come very close to the critical point, and the software has a different method of handling situations close to the critical point condition. The difference is clearly shown in Figure 8.12a. The liquid volume fraction increases from 0 to 8 [*m*], yet decreases after 8 [*m*] and finally increases again at 15 [*m*]. This decrease in liquid volume is non-physical.

Further analysis, as shown in the depressurization paths at x = 0.08 [*m*], Figure 8.12, highlights how both simulations navigate near the critical point. Here it is observed how the OLGA software tries to avoid the critical point, leading to the observed discrepancies. Although these differences do not significantly impact pipeline design outcomes, they are noteworthy. In the author's opinion, there is potential for OLGA to improve its handling of conditions near the critical point to better align with physical expectations.

8.2.4. Span Wagner vs Peng Robinson EOS

The SW EOS is widely regarded as the reference equation of state for modeling pure CO_2 fluid due to its high accuracy. However, it is computationally more expensive compared to the PR EOS. If the PR EOS can provide comparable results with significantly lower computational costs, it may serve as a valuable alternative in scenarios where runtime is a critical factor.



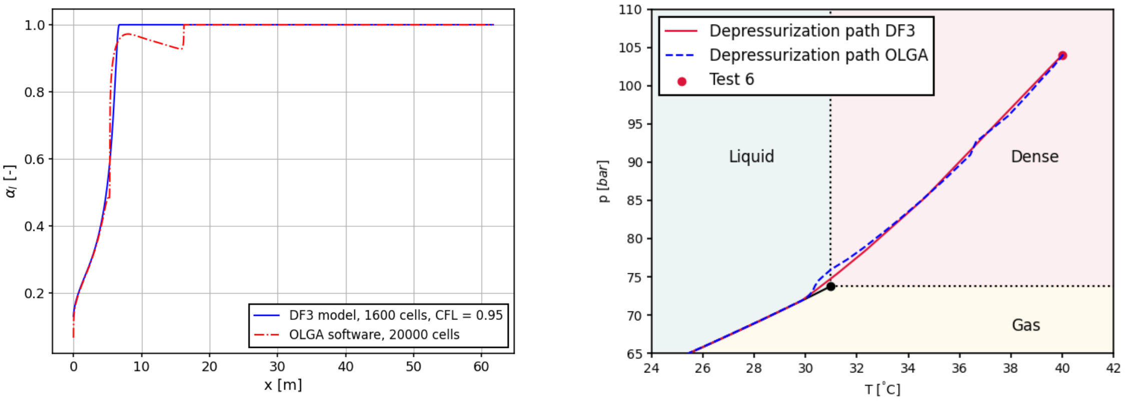
(a) Pressure of test 6 at t = 0.1 [*s*]

(b) Pressure of test 6 zoomed in at t = 0.1 [*s*]

(c) Temperature of test 6 at t = 0.1 [*s*]

(d) Temperature of test 6 zoomed in at t = 0.1 [*s*]

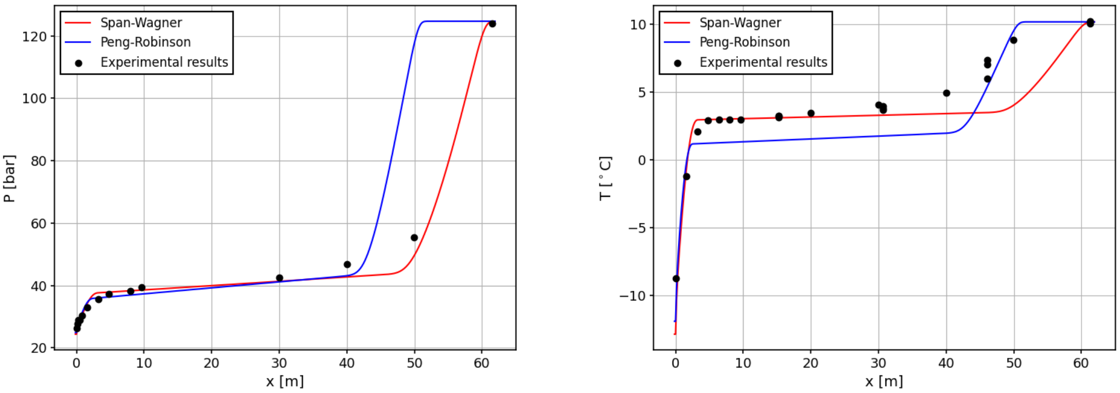
Figure 8.11: Pressure & temperature results of test 6 at t = 0.1 [*s*] for the *DF*3 model & OLGA software for a different amount of cells.



(a) Liquid volume fraction of test 6 at t = 0.1 [*s*] for the *DF*3 model & OLGA software

(b) Pressure vs Temperature graph. The depressurization of test 6 with the *DF*3 model & the OLGA software are shown.

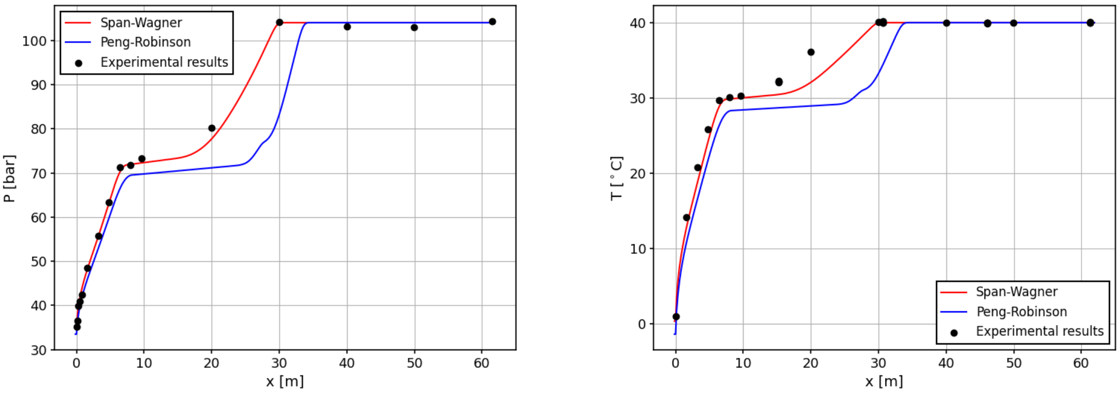
Figure 8.12: A comparison of the OLGA software and the *DF*3 model simulation for test 6.



(a) Pressure of test 19 at t = 0.1 [*s*].

(b) Temperature of test 19 at t = 0.1 [*s*]

Figure 8.13: Pressure and temperature results of test 19 at t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 6 at t = 0.1 [*s*].

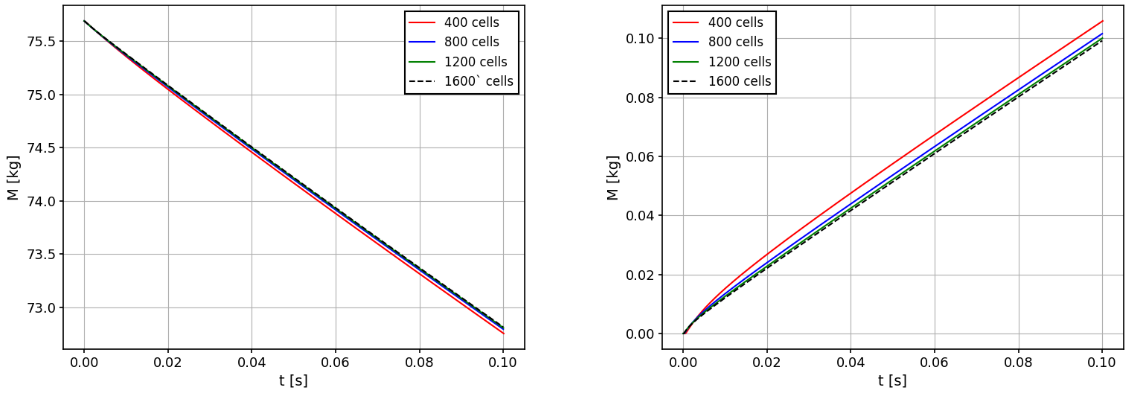
(b) Temperature of test 6 at t = 0.1 [*s*]

Figure 8.14: Pressure and temperature results of test 6 at t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.

The pressure and temperature results for test 19 and test 6 are presented in Figure 8.13 and Figure 8.14. In test 19, the pressure predictions from the PR EOS closely align with those from the SW EOS. However, the decompression wave in the PR simulation lags behind that of the SW simulation. This delay results in a pressure drop at 50 [*m*], while experimental data indicates that the decompression wave should have propagated further along the pipeline.

Interestingly, the slower decompression wave in the PR simulation appears to result in a more accurate prediction of the temperature drop. However, this agreement seems to be coincidental rather than a reflection of the underlying physics captured by the PR EOS. Besides, while the PR EOS predicts the magnitude of the temperature drop more closely, it fails to accurately capture the temperature distribution following the drop, as shown in the range from 5 [*m*] to 45 [*m*] in Figure 8.13b.

In test 6, the results obtained using the PR EOS demonstrate noticeable discrepancies compared to the SW EOS. Both the predicted pressure and temperature plateaus are underestimated. Additionally, the pressure and temperature profiles exhibit a distinct kink during the drop. This behaviour is attributed to the inherent limitations of the PR EOS when modelling fluid behaviour near or across the critical point. Such conditions arise in test 6, as will be further elaborated in the subsequent section.



(a) Total mass in the pipeline until 0.1 $[s]$ for test 19.

(b) Vapour mass in the pipeline until 0.1 $[s]$ for test 19.

Figure 8.15: The mass in the pipeline until 0.1 $[s]$ for test 19. The simulations are done with the $DF3$ model and a CFL of 0.95.

8.2.5. $DF3$ total mass & mass transfer

In the state vector of the $DF3$ model, one of the variables is the density. With the density known, the area of the pipeline and the cell size, the mass in the whole pipeline can be calculated with Equation 8.1.

$$M = A\sum_{i=1}^{N}(\rho_i \Delta x_i). \tag{8.1}$$

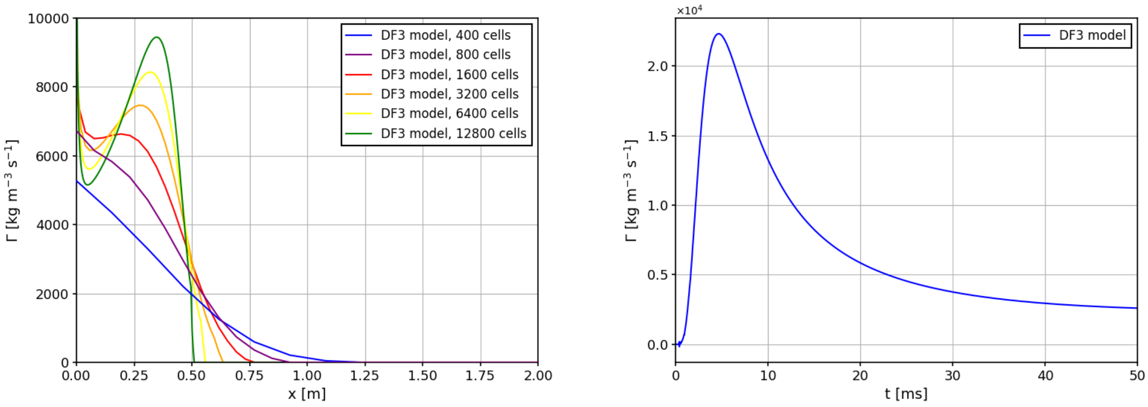
The results are shown in Figure 8.15 for test 19. The mass calculation seems to be converged as increasing the number of cells gives less difference in the results. The difference between 400 cells and 1600 cells is 0.05 $[kg]$ after 0.1 $[s]$. Compared to the total mass 72.8 $[kg]$, the difference is less than 0.1 %. What is observed from these graphs is that the total mass decreases by 3 $[kg]$, while the vapour mass only increases by 0.1 $[kg]$. It means that most of the mass has left the pipeline.

Additionally, the mass transfer term for the $DF3$ model, as explained in section 5.5, has been calculated for test 19. The result for test 19 is shown in Figure 8.16. The results for the other tests are given in Appendix F, but due to the same behaviour, only test 19 is shown and discussed. A notable observation is that the mass transfer does not easily converge upon mesh refinement. This is first seen in Figure 8.16a. Here, the mass transfer rate is plotted against the distance. The simulations are done with a CFL of 0.1, to ensure stability. The simulation with 12,800 cells was performed at a higher CFL of 0.8, to achieve a reasonable computation time. Despite the higher CFL, the same behaviour is observed: with increasing cell numbers, the mass transfer exhibits a more pronounced spike and a larger parabola shape when plotted over the pipeline's distance.

When examining the depressurization paths in Figure 8.17, all tests depressurize to the saturation line, transitioning into a two-phase flow. This behaviour is expected with the $DF3$ model, as it enforces phase equilibrium conditions. The spikes in the mass transfer likely arises because the $DF3$ model enforces phase equilibrium conditions, requiring the mass transfer to adjust rapidly to maintain consistency with the saturation state during depressurization.

In Figure 8.16b, the mass transfer is plotted over time at 0.08 $[m]$ from the pipeline outlet. The results again show a distinct spike, suggesting that this feature propagates through the pipeline. A definitive reason for this spike is not found, and more research needs to be done. Especially, as later will be shown in subsection 8.3.3, this phenomena is not seen in the $DF4$ model.

Efforts were made to determine the mass transfer using simulations conducted with the OLGA software for verification purposes. However, due to the lack of understanding the precise flux calculation methodology employed by the software, this proved to be unfeasible, rendering the results unreliable.



(a) Mass transfer for test 19 at 0.02 [*s*]. The simulation is done with a CFL of 0.02. Only for the simulation with 12800 cells has a CFL of 0.8 been used.

(b) Mass transfer for test 19 at 0.08 [*m*] until 0.05 [*s*]. The simulation is done with 4000 cells and a CFL of 0.95.

Figure 8.16: The mass transfer rate in the pipeline for test 19. The simulations are done with the *DF*3 model.

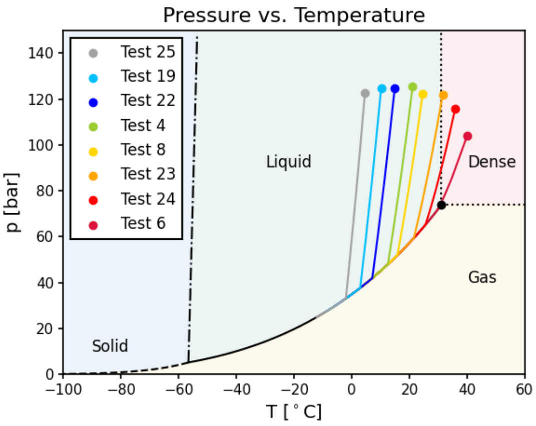


Figure 8.17: Depressurization paths of the CO_2 fluid modelled by the *DF*3 model up to a time of 0.02 [*s*]. The simulations have been done with a CFL number of 0.95 and 4000 cells.

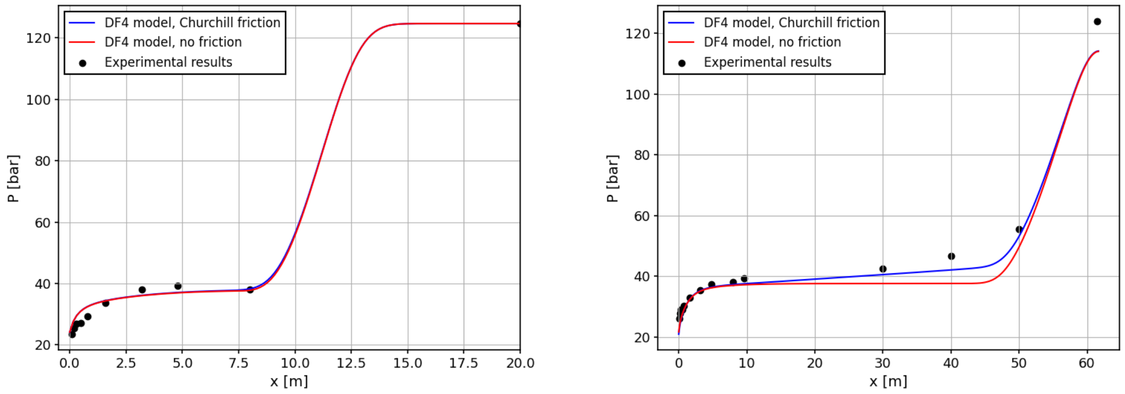
8.3. *DF*4 model

As explained more thoroughly in subsection 3.1.3, the *DF*4 model consists of four flow PDEs, with two mass conservation equations, one bulk momentum conservation equation and one energy conservation equation. In Figure 8.20 to Figure 8.22, the simulations of multiple experiments, specifically tests 25, 19, and 22, are presented using the *DF*3 and *DF*4 models. For simulations that cross closer to the critical point, the results were inaccurate. According to Log et al. [22], this inaccuracy was due to the use of an explicit method of calculating the primitive variables. Log et al. recommended using an implicit method for other test cases. More will be explained in **??**. However, due to time constraints, it was not feasible to implement the implicit method.

The pressure results are provided for t = 0.02 [*s*] and for t = 0.1 [*s*]. All simulations were conducted with friction calculated using the Churchill method. The effects of gravity are absent, as the setup involved a horizontal pipeline. All simulations were performed with a CFL number of 0.5 and 400 cells. Table 8.1 shows the initial conditions for each test number and the value of θ used in calculating Γ, as detailed in subsection 3.3.3.

Table 8.2: The initial conditions with θ defined for the tests that were successful. The value for θ is taken from Log et al. [23].

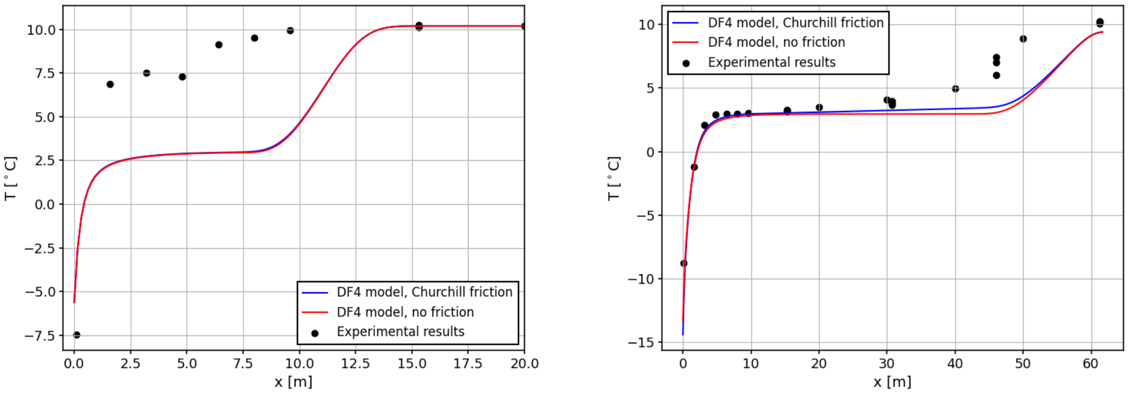
Test no.	Pressure avg. $[bar]$	Temperature avg. [°C]	θ $[ms]$
22	124.8	14.9	2.0
19	124.7	10.2	2.5
25	122.7	4.6	2.8



(a) Pressure of test 19 at t = 0.02 $[s]$.

(b) Pressure of test 19 at t = 0.1 $[s]$.

Figure 8.18: Pressure results of test 19 t = 0.02 $[s]$ & t = 0.1 $[s]$ for the $DF4$ model with Churchill friction. Both simulations have been done with a CFL number of 0.5, and with 400 cells.



(a) Temperature of test 19 at t = 0.02 $[s]$.

(b) Temperature of test 19 at t = 0.1 $[s]$.

Figure 8.19: Temperature results of test 19 at t = 0.02 $[s]$ & t = 0.1 $[s]$ for the $DF4$ model with Churchill friction. Both simulations have been done with a CFL number of 0.5, and with 400 cells.

8.3.1. $DF4$ with Churchill friction

In Figure 8.18, the simulations are shown with the $DF4$ model with and without the (Churchill) friction. The results are only shown for test 19, as all the other tests showed the same behaviour. The simulations are done with 400 cells, and a CFL of 0.95. θ = 2.5e-3 $[s]$ in this simulation, as shown in Table 8.2. The simulations are done until 0.02 $[s]$ and 0.1 $[s]$. Note that for the results on 0.02 $[s]$, the pipeline is shown until 20 $[m]$.
As previously observed with the $DF3$ model, friction results in momentum loss and a decrease in the pressure plateau along the length of the pipeline. At the early stage of 0.02 $[s]$, however, friction has minimal influence on mass transfer. This is evident from the near-identical pressure distributions with and without friction, indicating that the effect of friction on the mass transfer rate is negligible in the initial phase.

In addition to the pressure, temperature results for the $DF4$ model are also shown in Figure 8.19, comparing simulations with and without friction. The data reveal that friction has little to no effect on the temperature distribution. This result is consistent with physical expectations, as temperature variations in this context are predominantly governed by heat transfer mechanisms, while friction primarily impacts momentum. Although friction dissipates energy in the form of heat, the amount of heat generated is relatively small compared to the system's overall thermal energy. Consequently, the effect of friction on the temperature is minimal and does not significantly alter the thermal behaviour of the flow.

8.3.2. $DF4$ vs $DF3$

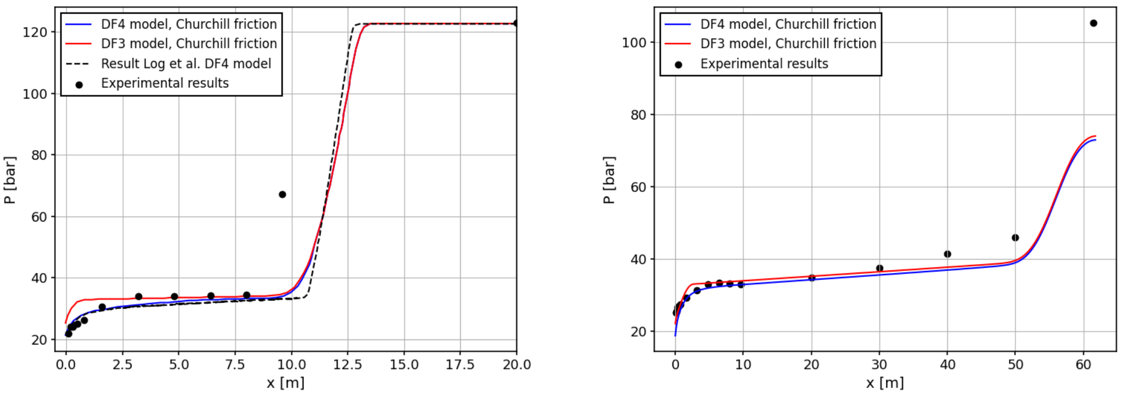
In order to find the differences and benefits of the $DF4$ model, it is compared with the $DF3$ model. This is done for the pressure and temperature. These are shown from Figure 8.20 till Figure 8.25. This is for a final simulation of 0.02 $[s]$ and 0.1 $[s]$.

Pressure results

A key observation is that the present simulation aligns closely with the results obtained by Log et al. [22]. This is seen in the simulation up to a time of 0.02 $[s]$, Figure 8.20a, Figure 8.21a and Figure 8.22a. The simulations match almost perfectly until the point where the pressure begins to drop from the initial value. This divergence can be attributed to Log et al.'s use of 2500 cells, resulting in a more converged solution. Additionally, the differences in the phase transfer region, from 0 to 7.5 $[m]$ in all three testcases, may be due to the method Log et al. employed to derive their results. Their results were obtained using a graph digitization tool[1], which, while generally accurate, can introduce minor errors.

The most notable differences between the $DF3$ and $DF4$ models are observed at the beginning of the pipeline, where the flow transitions into the two-phase regime. In the provided figures, the marker indicates the boundary between the two-phase and single-phase regions. To the left of the marker lies the two-phase flow, while the region to the right represents the single-phase flow. It is evident that the two-phase region extends further along the pipeline in the $DF4$ model compared to the $DF3$ model. This difference is attributed to the delayed mass transfer in the $DF4$ model, resulting in a prolonged two-phase regime. However, as simulations progress to 0.1 $[s]$, the results of the $DF4$ model converge toward those of the $DF3$ model, diminishing the observed benefits of the former.

A limitation of the $DF4$ model is that its mass transfer is governed by the relaxation constant θ, which was set at a specific value in this work. While this can be validated against experimental results, determining an accurate value for θ is challenging in the absence of experimental data, making it difficult to quantify the correct mass transfer for novel scenarios.

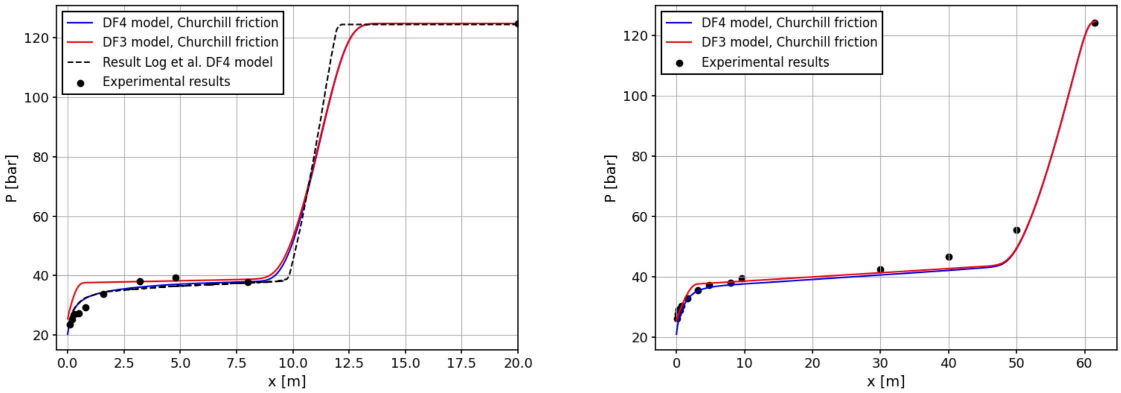


(a) Pressure of test 25 at t = 0.02 $[s]$.

(b) Pressure of test 25 at t = 0.01 $[s]$.

Figure 8.20: Pressure results of test 25 at t = 0.02$[s]$ & t = 0.1$[s]$ for the $DF4$ model compared with the $DF3$ model. All four simulations were done with a CFL of 0.5 and 400 cells.

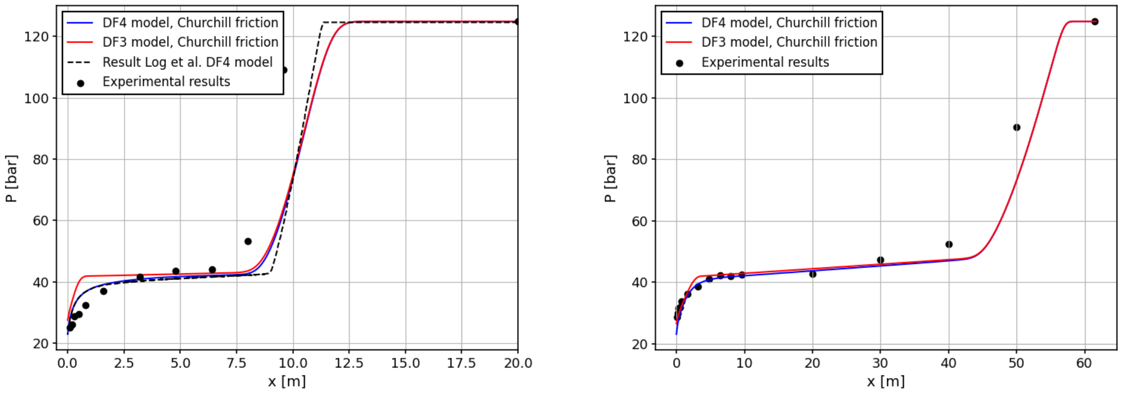
[1] https://plotdigitizer.com/app



(a) Pressure of test 19 at t = 0.02 [s].

(b) Pressure of test 19 at t = 0.01 [s].

Figure 8.21: Pressure results of test 19 at t = 0.02[s] & t = 0.1[s] for the $DF4$ model compared with the $DF3$ model. All four simulations were done with a CFL of 0.5 and 400 cells.



(a) Pressure of test 22 at t = 0.02 [s].

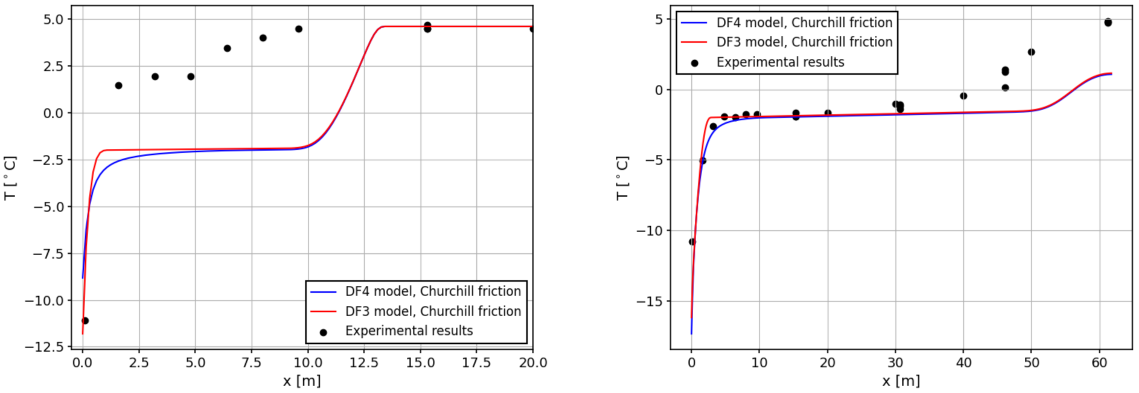
(b) Pressure of test 22 at t = 0.01 [s].

Figure 8.22: Pressure results of test 22 at t = 0.02[s] & t = 0.1[s] for the $DF4$ model compared with the $DF3$ model. All four simulations were done with a CFL of 0.5 and 400 cells.

Temperature results

When examining temperature simulations, the differences between the $DF3$ and $DF4$ models are minimal. Figure 8.23 through Figure 8.25 illustrate that, for simulations up to 0.1 [s], the temperature results of both models largely overlap throughout most of the pipeline. The only notable difference appears at the beginning of the pipeline, where the transition from liquid to two-phase flow occurs. In the $DF4$ simulation, this transition is more gradual, aligning more closely with experimental observations, as seen in Figure 8.23b, Figure 8.24b, and Figure 8.25b. However, this difference is minor in magnitude (less than 0.5 [] for the three test cases) and occurs over a short timescale, making it unlikely to have significant practical implications.

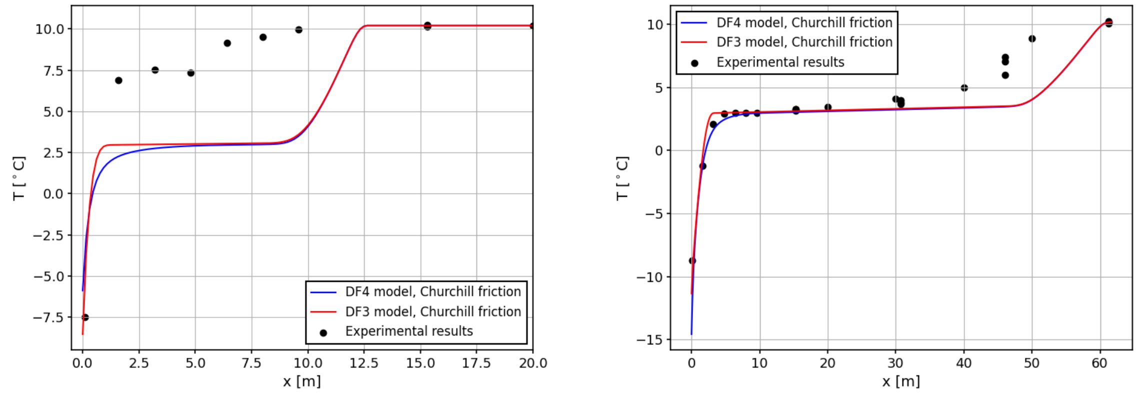
For simulations up to 0.02 [s], the temperature results do not align with experimental data. A steep temperature drop coincides with the location of a steep pressure drop, a phenomenon not observed in the temperature measurements. The reason for this mismatch is due to the inability of both models to accurately capture the slower temperature adaptation timescale, as discussed in subsection 2.1.2. When comparing the $DF3$ and $DF4$ models, similar behaviour is observed, with the temperature profiles of the two models nearly overlapping. The only noticeable difference occurs at the beginning of the pipeline, where the delayed mass transfer in the $DF4$ model results in a slightly different temperature response.



(a) Temperature of test 25 at t = 0.02 [*s*].

(b) Temperature of test 25 at t = 0.1 [*s*].

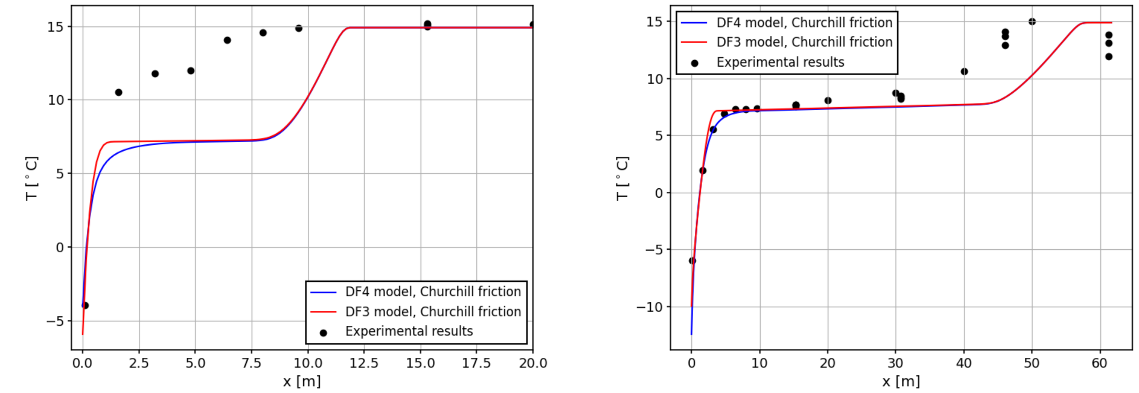
Figure 8.23: Temperature results of test 25 at t = 0.02 [*s*] & t = 0.1 [*s*] for the *DF*4 model compared with the *DF*3 model. The simulations until 0.02 [*s*] were done with a CFL of 0.95 and 400 cells. The simulations until 0.1 [*s*] were done with a CFL of 0.5 and 400 cells.



(a) Temperature of test 19 at t = 0.02 [*s*].

(b) Temperature of test 19 at t = 0.1 [*s*].

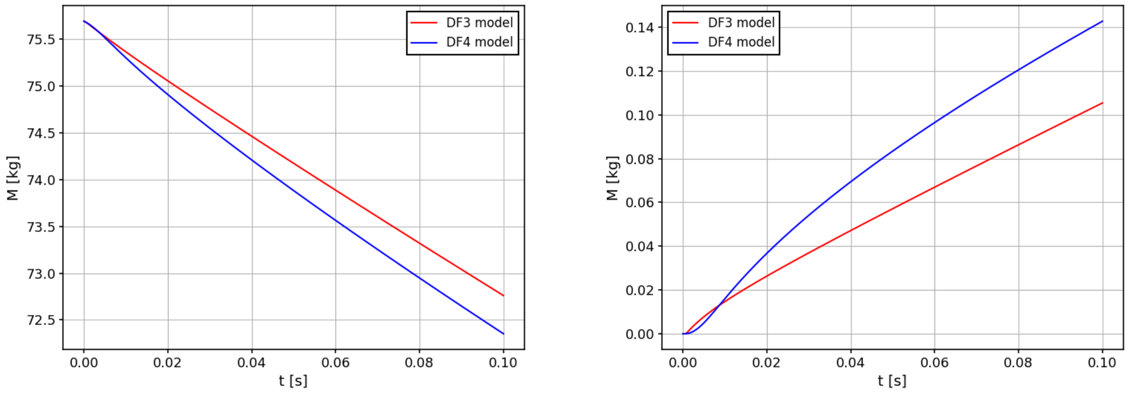
Figure 8.24: Temperature results of test 19 at t = 0.02 [*s*] & t = 0.1 [*s*] for the *DF*4 model compared with the *DF*3 model. All four simulations were done with a CFL of 0.95 and 400 cells.



(a) Temperature of test 22 at t = 0.02 [*s*].

(b) Temperature of test 22 at t = 0.1 [*s*].

Figure 8.25: Temperature results of test 22 at t = 0.02 [*s*] & t = 0.1 [*s*] for the *DF*4 model compared with the *DF*3 model. All four simulations were done with a CFL of 0.95 and 400 cells.



(a) Total mass in the pipeline up to a time of 0.1 [s] for test 19.

(b) Vapour mass in the pipeline up to a time of 0.1 [s] for test 19.

Figure 8.26: The mass in the pipeline up to a time of 0.02 [s] for test 19. For both the $DF3$ and $DF4$ models, the simulation is done with a CFL of 0.95 and 400 cells.

8.3.3. Total mass & mass transfer

A comparison of the total mass within the pipeline between the $DF3$ and $DF4$ models reveals that the $DF4$ model exhibits a faster decrease in total mass, as shown in Figure 8.26a. This indicates a higher rate of CO_2 expulsion from the pipeline for the $DF4$ model, primarily attributed to its higher mixture speed of sound. The increased speed of sound leads to a higher critical velocity at the pipeline's exit, which governs the maximum mass flow rate under the choking condition. Consequently, the $DF4$ model allows more mass to escape the pipeline within the same time-frame.

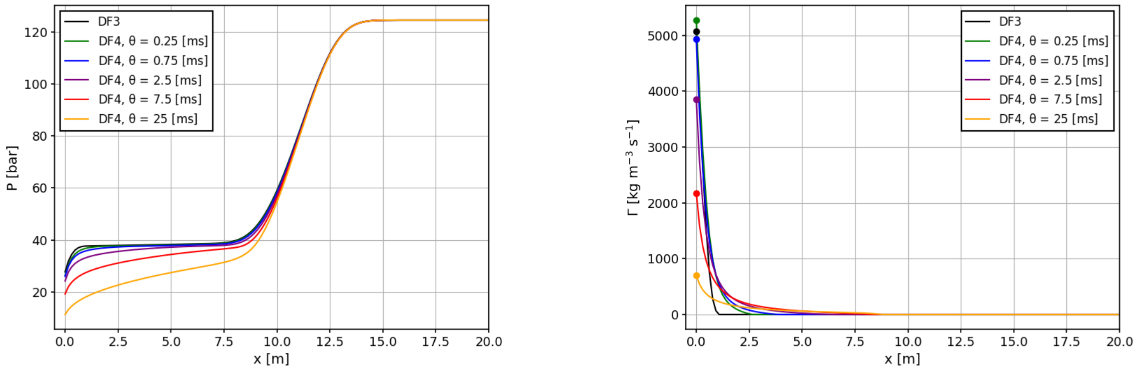
Additionally, Figure 8.26b displays the vapour mass within the pipeline. Although the $DF4$ model generally predicts a faster mass transfer rate, an exception is observed during the first 10 [ms], where the $DF3$ model exhibits a higher vapour mass. This behaviour is further reflected in the mass transfer results shown in Figure 8.27.

The spatial distribution of the mass transfer at 0.1 [s] is illustrated in Figure 8.27a. Results show convergence with increasing mesh resolution. Although the different mass transfer rate between 400 cells and 3200 cells is almost twice the rate, the overall behaviour is the same and with increasing cell density, the differences become smaller. In addition, the mass transfer rate is compared for the $DF3$ and $DF4$ models, shown in Figure 8.27b. There is a big spike for the mass transfer rate in the $DF3$ model. However after 0.02 [s], the mass transfer rate of the $DF3$ model drops below the rate of the $DF4$ model, and stays below up to a time of 0.05 [s]. This behaviour could be explained by looking at the depressurization paths.

In Figure 8.28, the depressurization paths for various test cases are presented, as modelled by the $DF3$ and $DF4$ models. The $DF3$ model demonstrates a drop in pressure and temperature until reaching the saturation line, after which the flow closely follows the saturation curve. In contrast, the $DF4$ model exhibits a drop in pressure below the saturation line, a phenomenon referred to as pressure undershoot, as explained in subsection 2.1.3. This pressure undershoot in the $DF4$ model allows for a more gradual relaxation of the mass transfer. Meanwhile, the $DF3$ model enforces an increase in mass transfer to ensure the flow adheres to the saturation line, resulting in distinct mass transfer behaviours between the two models.

8.3.4. Relaxation time

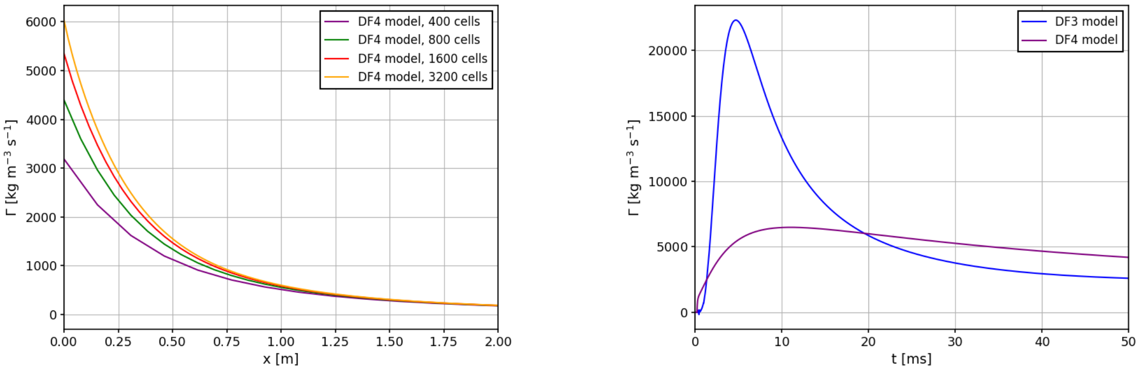
The relaxation parameter θ has a significant impact on the pressure results, as illustrated in Figure 8.29a. This figure depicts the pressure as a function of distance at t = 0.02 [s]for test 19. It can be observed that, as θ decreases, the $DF4$ model converges toward the $DF3$ model. This behaviour is expected, as the limit of $\theta \to 0$ is anticipated to yield results equivalent to the $DF3$ model. In Figure 8.29b, the mass transfer is presented for the same test. As expected, with larger values of θ, the mass transfer rate is lower. Interestingly, with larger values of θ, the mass transfer appears to be more diffused compared to the $DF3$ model. This suggests that the $DF3$ model imposes a constraint that drives the system towards a pressure plateau.



(a) Pressure of test 19 at t = 0.02 [s] with different θ.

(b) Mass transfer of test 19 at t = 0.02 [s] with different θ.

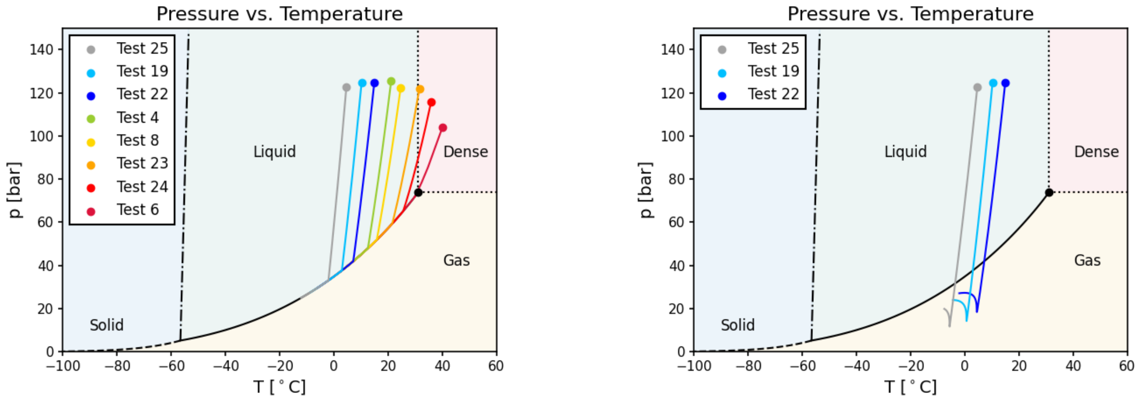
Figure 8.29: The pressure and mass transfer of test at t = 0.02 [s]. The simulation has been done with a CFL number of 0.5 and 400 cells.



(a) Mass transfer for test 19 using the $DF4$ model. The figure shows the result at 0.1 [s]. The simulation is done with a CFL of 0.1.

(b) Mass transfer for test 19 at 0.08 [m] up to a time of 0.02 [s]. The simulation is done with 4000 cells and a CFL of 0.95.

Figure 8.27: The mass transfer in the pipeline versus the distance and versus the time.



(a) Depressurization paths of the CO_2 fluid modelled by the $DF3$ model up to a time of 0.02 [s].

(b) Depressurization paths of the CO_2 fluid modelled by the $DF4$ model up to a time of 0.02 [s].

Figure 8.28: The depressurization paths of the different testcases modelled by the $DF3$ and $DF4$ models. The simulations have been done with a CFL number of 0.95 and 4000 cells.

8.4. Pressure over time

In Figure 8.31, test 19 and test 6 are shown with the pressure evolving over time. The remaining test results, which exhibit similar behaviour, are included in Appendix F. The results are shown up to a time of 0.02 [s], at a distance of 0.08 [m] from the open end of the pipeline.

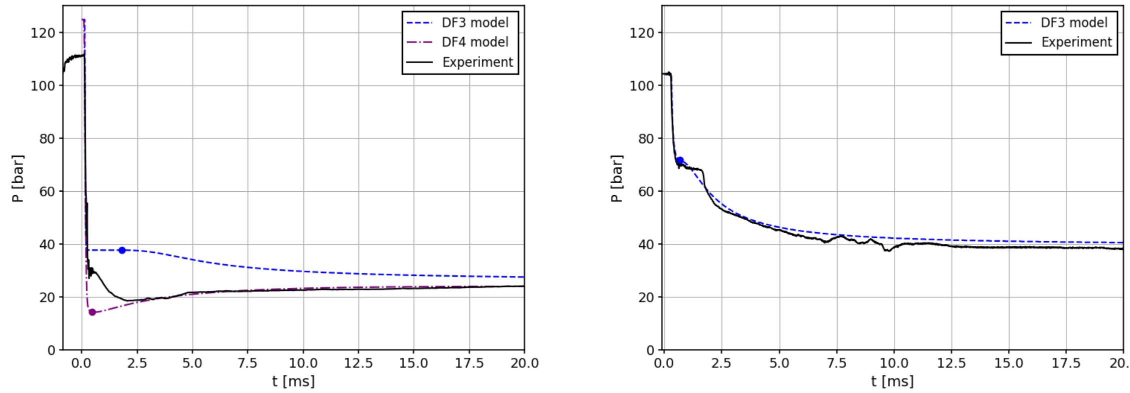
8.4.1. Simulation setup

These tests were performed with 4000 cells and simulated over a 20 [m] segment of the pipeline. This setup ensures an extremely dense discretization, guaranteeing converged results. Since the propagation wave would not reach beyond 20 [m] within the simulation time-frame, the shortened pipeline length was sufficient for accurate modelling. Additionally, the pressure jumps in the experimental data were aligned with the simulations. For example, in test 19, the experimental data was delayed by 0.00288 [s] to synchronize the timing of the pressure jump with the simulation. For the $DF4$ model, only the results of tests 25, 19, and 22 are included, as the solver encountered stability issues for other test cases.

8.4.2. Observations

The $DF3$ model consistently predicts a higher pressure plateau compared to the experimental results. This observation aligns with findings by Log et al. [22]. The discrepancy arises because the pressure difference between the pipeline conditions and the ambient was so significant that the fluid could not maintain its saturation state. In literature, this phenomenon is described as a metastable condition, where the fluid rapidly seeks to return to its stable state. The $DF4$ model captures this behaviour more effectively by delaying the mass transfer, resulting in the earlier mentioned pressure undershoot. For example, in test 19, shown in Figure 8.30a, the $DF4$ model closely aligns with the experimental data. However, also the $DF4$ model is not able to fully replicate the pressure evolution, with the used mass transfer model in this research.

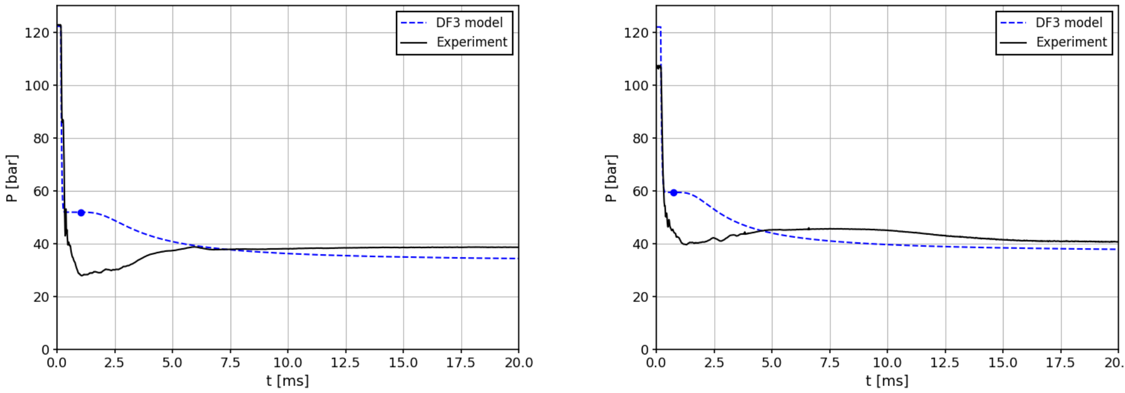
However, for test 6, shown in Figure 8.30b, the $DF3$ model aligns well with the experimental data, with no observed undershoot. This contrasts with other tests, such as tests 8 and 23, shown in Figure 8.31a and Figure 8.31b respectively, which share similar saturation pressures of around 40 [bar]. The absence of undershoot in test 6 may be attributed to its higher initial temperature, which results in a lower speed of sound. The slower pressure wave propagation allows the fluid to remain on its saturation curve.



(a) Pressure of test 19 at x = 0.08 [m]

(b) Pressure of test 6 at x = 0.08 [m]

Figure 8.30: Pressure evolution at x = 0.08 [m] for test 19 & test 6. Both simulations have been done with a CFL of 0.95 and 4000 cells.



(a) Pressure of test 8 at x = 0.08 $[m]$

(b) Pressure of test 23 at x = 0.08 $[m]$

Figure 8.31

8.5. Temperature over time

The temperature evolution over time for test 19 and test 6 is illustrated in Figure 8.32. Results for other tests, which exhibit similar behaviour, are provided in Appendix F. As for the pressures, the simulations were conducted up to 20 $[ms]$, at a location 0.08 $[m]$ from the open end of the pipeline, using 4000 cells over a 20 $[m]$ segment to ensure a high level of discretization.

8.5.1. Observations

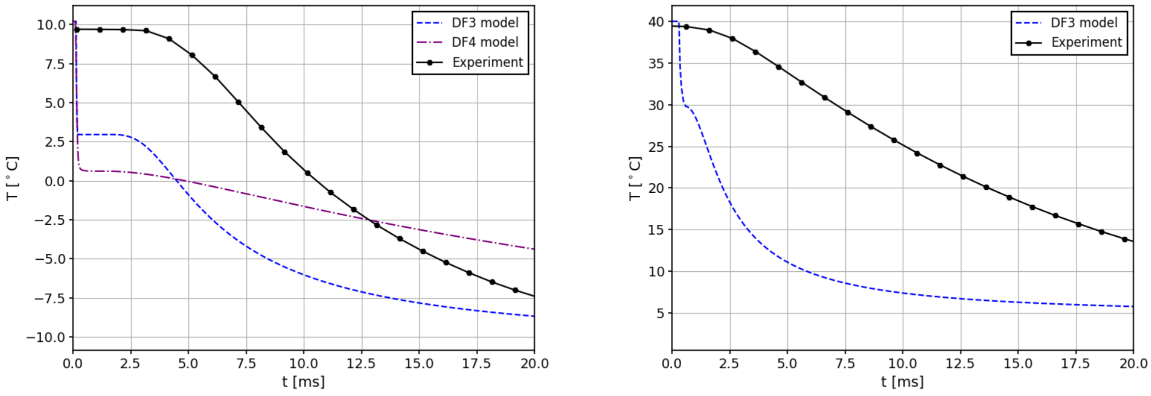
The differences here are much greater than in case of pressure. In the $DF3$ model, the temperature of the simulation spikes quickly downwards after the opening of the valve. Nevertheless, the results of the experiment display different behaviour. It takes time for the temperature to change and drop down to its equilibrium state.

In the $DF3$ model, the simulated temperature rapidly drops immediately after the valve opens, forming a sharp downward spike followed by a short plateau. However, experimental results indicate that the temperature takes longer to adjust, showing a more gradual decrease to the equilibrium state. This discrepancy was also noted in section 8.2, where the temperature profile at a given time did not align with the experimental data.

In section 8.2, it was also found that at a given time, the results of the simulation and the experiments did not match. A reason for the behaviour could be that it takes time for the measurement device to adjust to the temperature of the fluid. What is different in these scenarios, is that there is an immediate drop of temperature, and then a plateau is formed for a short time. This plateau is in the order of milliseconds, and so it could be that the measurement device can not catch this phenomenon. However, this is unlikely, as in test 25, shown in Figure F.13a, the plateau is there for 2.5 $[ms]$, and the results of the experiment show no change compared with the initial condition. The more likely explanation is that the $DF3$ model does not capture the physics behind it correctly, and a relaxation time is needed to implement into the model.

While at the start of the simulation the results do not match, both the experimental results and the simulation results tend to 'converge' to the same equilibrium condition. It is remarkable that this happens more quickly with the test conditions with a lower initial temperature (such as test 19). The reason could again be that these test conditions have a higher speed of sound. Yet, while the higher speed of sound tends to undershoot the pressure, for the temperature, it enables the fluid to more quickly converge to the equilibrium temperature, as the time-scale of temperature changes are slower than for pressure changes.

What is surprising is that the $DF4$ model has less accurate results, compared with the $DF3$ model. The higher spike downwards can be explained by the higher mass transfer. However, after the downwards spike, the temperature also tries to converge to the equilibrium state, but the path is unexpected. It tends to go slower to the equilibrium state than the $DF3$ model, and even than the experiment. While the $DF3$ model shows a big difference with the experiments, it is always with a lower temperature, and improvements can be made to correct this. For the $DF4$ model, there is not a physical reason why it behaves like this, and how it can be improved. This is very surprising, and the $DF4$ model can therefore not be used to model the temperature adequately.



(a) Temperature of test 19 at x = 0.08 $[m]$

(b) Temperature of test 6 at x = 0.08 $[m]$

Figure 8.32: Temperature results of test 19 and 6 at x = 0.08 $[m]$. The simulations have been done with 4000 cells and a CFL number of 0.95

8.6. Summary

This chapter presented the results of the depressurization tests simulated using the $DF3$ and $DF4$ models, alongside comparisons with experimental data and OLGA software simulations. The chapter highlighted the pressure, temperature, and mass transfer dynamics across various test cases, providing insights into the performance and limitations of the models under different conditions.

The $DF3$ model demonstrated good agreement with experimental pressure data, particularly for cases with lower initial temperatures, where the plateau pressures aligned well. However, discrepancies were observed in high-temperature tests, where the model failed to capture the undershoot in pressure attributed to metastable conditions. The $DF4$ model, with its enhanced mass transfer mechanism, addressed these undershoots effectively and aligned more closely with experimental data. Nonetheless, the $DF4$ model's reliance on predefined relaxation parameters like θ introduced complexities in cases lacking experimental validation.

Temperature results revealed significant challenges for both models. The $DF3$ model exhibited a rapid temperature drop immediately after depressurization, forming an unrealistic plateau inconsistent with experimental measurements. The $DF4$ model, despite its improved handling of pressure dynamics, performed worse in capturing the temperature evolution. The irregular transition towards equilibrium in the $DF4$ simulations lacked physical justification, raising concerns about its reliability for temperature modelling. Overall, neither model adequately captured the delayed temperature response observed in experiments, suggesting the need for additional mechanisms to simulate thermal relaxation processes.

Comparisons with OLGA software highlighted areas where the $DF3$ model performed well but also underscored limitations in its handling of near-critical point conditions. The OLGA simulations occasionally diverged from experimental trends, likely due to differences in how critical conditions were modelled. While the $DF3$ and $DF4$ models showed convergence in long-term results, their transient behaviours revealed contrasting strengths and weaknesses.

Finally, the analysis of mass transfer dynamics confirmed that the $DF4$ model predicted faster mass transfer rates than the $DF3$ model, particularly in the early stages of depressurization. However, this did not always translate to improved accuracy, as the underlying assumptions about mass transfer rates remained dependent on empirical parameters.

In conclusion, while the $DF3$ model proved robust for pressure predictions, it fell short in accurately modelling temperature dynamics. The $DF4$ model provided better alignment with experimental pressure data but struggled with temperature and introduced additional complexities. Both models require further refinement, particularly in representing thermal relaxation, to achieve comprehensive accuracy in pipeline depressurization scenarios.

9
Discussion

9.1. Pressure accuracy

The *DF*3 model demonstrated accurate predictions for pressures when the CO_2 fluid was in a dense-phase regime and depressurized to the saturation line. However, this implies that the CO_2 fluid should be at a high temperature (higher than 31 [°C]) . When transporting CO_2, this not always possible (such as in pipelines in deep cold water). For lower initial temperatures (lower than 31 [°C]), where the CO_2 fluid is in a liquid phase, the *DF*3 model fails to capture the pressure undershoot below the saturation line, as is mentioned by Log et al. [22] [23] [24]. This limitation is due to its forced adherence to the saturation line.

The *DF*4 model addresses this issue by incorporating a configurable mass transfer term, Γ, which allows it to model the pressure undershoot effectively. In this research, an accurate method for calculating the mass transfer is used for the specific test-cases. The method is developed by Log et al. [22] [23] [24] with the experimental work that was done, and the method could be validated. Nevertheless, the goal is to develop a method which is reliable without knowing beforehand what the result will be. Hereby, the mass transfer of the *DF*3 model can be used.

Looking at the timescale, these results are done in a matter of milliseconds, and after 20 [ms] the results of the experiments and the simulations are (almost) aligned. The biggest misalignment is of the *DF*3 model in test 22, shown in Figure 9.1. However, this difference at 20 [ms] is only 4 [bar]. This means the benefit of using a non-equilibrium mass transfer quickly vanishes, and the *DF*3 model can give accurate results for the pressure.

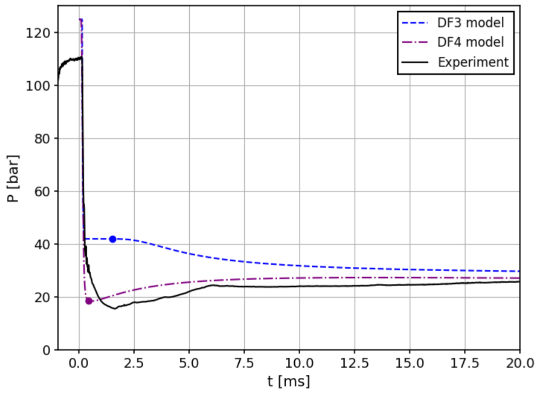


Figure 9.1: Pressure of test 22 at x = 0.08 [m]. Done with a CFL of 0.95 and 4000 cells.

9.2. Temperature accuracy

As mentioned in chapter 8, the accuracy of the temperature results is smaller compared to the pressure results. The temperature effects are not simulated correctly, and discrepancies occur. To start, the $DF3$ and $DF4$ models could both not correctly capture the delayed temperature effects. In literature, and described in subsection 2.1.2, it was found that the temperature time-scales are in a higher order compared to the pressure time-scales. In other words, it takes more time for the temperature to change with different conditions.

Taking a closer inspection in the $DF3$ and $DF4$ models, the reason for this discrepancy becomes more clear. In the energy equation, (Equation 9.1, repeated here for convenience reasons), the only changes are contributed by the flux and the pressure changes, which both travel with the velocity of the shock wave, contact discontinuity and rarefaction wave. It is therefore expected that in both models, the time-scales of the pressure and the temperature changes are identical.

$$\frac{\partial E}{\partial t} + \frac{\partial\big((E+p)v\big)}{\partial x} = 0 \tag{9.1}$$

Interestingly, the $DF4$ model predicts temperature less accurately than the $DF3$ model, even though the energy equations are identical. This suggests that the modified mass transfer in $DF4$ impacts the total energy evolution, leading to greater inaccuracies in temperature predictions.

To improve temperature modelling, additional source terms could be incorporated into the energy equation. Even without the $TF5$ model where two each phase has a separate energy equation, the bulk energy equation still can be improved as the discrepancies where observed in the single-phase. A delay mechanism for temperature, analogous to the delayed mass transfer in $DF4$, may offer a solution. Further investigation is required to explore whether similar approaches have been used for other substances and whether they can be adapted for CO_2.

9.3. Impact on pipeline design

With the design of pipelines, the question arises if this discrepancies have a noticeable effect on the strength and the material properties. It is of course important to understand what happens within the flow, and one can only be confident about it if a model is able to simulate the flow behaviour as what experimental results show.

However, when looking at the results, the discrepancies happen at very short time scales, and after, the simulations and the experiments quickly tend to go towards each other for the pressure. The biggest discrepancies happen in the first 20 $[ms]$ of the simulation. Here the pressure drop is greater than predicted with the $DF3$ model, with a possible pressure difference of 20 $[bar]$. However, the question arises what kind of impact this will have on the pipeline.

In addition, the temperature discrepancies are found because of the slower temperature changes. However, the temperature of the fluid in the simulation eventually drops to more or less the same temperature level that is found by the experimental results. his indicates that pipelines will still experience the predicted low temperatures. The problem is that it could be that the pipeline is over designed, as the temperature over time plots, such as Figure 8.32, show that the drop in temperature is more gradually instead of more instant as predicted by the $DF3$ model. With an instant drop, the stresses on the pipeline are calculated to be higher compared to a gradual drop.

9.4. Research questions

The research questions posed in subsection 2.4.1 are revisited here:

Research question 1.1.1: **What where the shortcomings of the $DF3$ model?**

- The $DF3$ model cannot simulate pressure undershoots due to its constraint to the saturation line. However, this discrepancy is most significant in the first 20 $[ms]$, after which the model predicts pressure within a 4 $[bar]$ margin.
- . The temperature predictions, while accurate in magnitude, are not aligned temporally. Furthermore, discrepancies at longer timescales (e.g., 2–10 $[s]$) may exceed 40 [°C], highlighting the need for improvements in long-term temperature modelling.

Research question 1.2.1: **What where the shortcomings of the $DF4$ model?**

- It was found that the model can more accurately the pressure behaviour, most noticeable in the first 20 $[ms]$. At a longer timespan, the advantages seem to vanish and both the $DF3$ and $DF4$ models approach the same pressure plateau.
- It was found in addition that the temperature is modelled less accurately compared to the $DF3$ model. Therefore, the conclusion can be made that while the $DF4$ predicts the physics of the pressure evolution better, it does not does not necessarily translate to better temperature predictions.

Research question 1.3: **Why did the $DF4$ model better predict in comparison to the $DF3$ model?**

- Comparing the mass transfer of the $DF3$ and $DF4$ models, it was found that the mass transfer of the $DF3$ spikes in the first 20 $[ms]$. By delaying the mass transfer in the $DF4$ model, it is possible to model the undershoot in pressure. With this difference, it could be possible what the differences are and if a general model for the difference can be invented. More research needs to be done on this problem.

Research question 1.4: **Which equation of state is used and recommended?** The SW and PR EOS were used in this research. As already found in literature, the SW EOS is the reference EOS for modelling pure CO_2 fluid, and this research has found the same conclusion.

Research question 2.2.2: **Can the HLLC approximate Riemann solver be used for flux calculation?** In section 7.4, the HLLC method is applied with the $TF5$ model, for a non-mixing two-fluid problem. It was hereby found that the HLLC method with the 1st order FVM and explicit Euler method in time is able to capture the flow dynamics, and that the $TF5$ model could be modelled hereby. Nevertheless, due to time constraints, it was not found to test the $TF5$ model on a problem with mass transfer, or a problem with CO_2 fluid. Therefore, more research needs to be done.

Research question 2.3: **Does the $TF5$ model predict CO_2 flow more accurately?** Unfortunately, no definite answer could be given in this research. However, it was found that the $DF3$ and $DF4$ models have their discrepancies with simulating the temperature, and therefore, there is reason to further investigate into the $TF5$ model.

9.5. Future work and recommendations

Building upon the findings of this study, several opportunities for future research and development are identified:

- The first possibility for improvement is to discretize the $DF4$ model in an implicit method. In this study, the primitive variables are calculated by an explicit method. However, close to the critical point, this method fails. Log et al. [24] suggested to use an implicit method for calculating the primitive variables, whereby the mass transfer term is incorporated into the calculation. This is shown in Equation 9.2. The same can be done for the calculation of the primitive variables in the $TF5$ model.

$$\mathbf{f} = \begin{pmatrix} \alpha_g\rho_g - \left((\alpha_g\rho_g)^{\mathrm{n+1}} + \Gamma^{t^n+\Delta t}\right) \\ (1-\alpha_g)\rho_l - \left((\alpha_l\rho_l)^{\mathrm{n+1}} + \Gamma^{t^n+\Delta t}\right) \\ \alpha_g\rho_g e_g(\rho_g, T) + (1-\alpha_g)\rho_l e_l(\rho_l, T) - (\rho e)^{\mathrm{n+1}} \\ p_g(\rho_g, T) - p_l(\rho_l, T) \end{pmatrix} = 0 \tag{9.2}$$

- The pressure undershoot suggest the need for more robust formulations of mass transfer terms, particularly under rapid depressurization scenarios. Future research could explore adaptive mass transfer models, whereby the mass transfer is developed based on the mass transfer of the $DF3$ model. Hereby, a general mass transfer model could be developed which differs based on the initial conditions of the CO_2 fluid.

- The temperature discrepancies may be solved by the use of adding source terms in the energy equation. Firstly, there is the possibility to add a heat exchange term with the ambient. However, this will not solve the delayed temperature effects. In order incorporate the delay, a same principle can be applied as for the mass transfer delay. Nevertheless, in the literature research, there has not been searched for a temperature delay equation, and if it has already been used for other substances. Therefore, it is first recommended if other methods have been used to model the temperature delay, and try to implement it for CO_2 fluid.

- Further development of the $TF5$ model, incorporating mass transfer and CO_2 thermodynamics, is expected to enhance its ability to accurately simulate the underlying physics of temperature variations. It is clearly seen that the $DF3$ and $DF4$ models do not accurately find the solution, and improving the modelling of the temperature is essential for the design of pipelines, as a low temperature can massively increase the brittleness of the material.

- Lastly, the models can be compared with experimental results from vertical pipelines. will demonstrate their viability, particularly as more CCS projects are being implemented, with the goal of underground CO_2 storage.

10

Conclusions and Recommendations

10.1. Conclusions

The main objective of this research was to investigate different multiphase flow models based on the Euler equations, with the special focus on differentiating between relaxing various flow parameters. In this research, three different multiphase models were implemented, with the first two models, $DF3$ and $DF4$ models, being tested for CO_2 simulations in a depressurizing pipeline. Based upon the work presented in this research, the following conclusions can be drawn:

- The friction impacts the pressure plateau, causing it to decrease along the length of the pipeline. However, it has a negligible effect on the phase transfer of the CO_2. In addition, two different methods of calculating the friction were used for the $DF3$ model, namely the Friedel method and Churchill method, and no significant differences was found in the simulations. It is therefore suggested to use the Churchill method as it is more easy for implementation.

- The $DF3$ model accurately captures the pressure variations overall. However, during the initial 20 $[ms]$, the model underestimates the pressure drop when starting in the liquid phase compared to the measured values in the pipeline. This difference is in certain scenarios 20 $[bar]$. Nevertheless, the $DF4$ model successfully captures this initial pressure drop with greater accuracy, aligning closely with the measured values in the pipeline. According to the research of Log et al. and the present research, it is done by arbitrarily decreasing the mass transfer, Γ, in the flow equations. While this was performed by changing the θ parameter by hand, it does prove that with a more sophisticated model for Γ, the pressure drop can be modelled more accurately compared to the $DF3$ model. It should be noted that both models did converge to the same equilibrium state after 20 $[ms]$, as found in the experimental results. The benefit of using the $DF4$ model is therefore mainly valuable at the start of the experiment.

- Contrary to the pressure, the temperature is less accurately simulated by the $DF3$ and $DF4$ models. The $DF3$ model predicts a downwards spike of the temperature immediately after the burst op the pipeline, however, this phenomenon is not seen in the experimental results. Afterwards, a plateau is formed until the temperature drops again, yet more smoothly now. However, over time, the $DF3$ model evolves in line with the experimental results, suggesting that both are progressing toward the same equilibrium state.

- The $DF4$ model cannot accurately capture the temperature changes in the beginning of the depressurization. As with the $DF3$ model, the $DF4$ model predicts an even greater spike downwards. Afterwards, the prediction of the temperature seems to also evolve to the equilibrium state, however, with a surprising path and the results do not seem to be trustworthy.

- The mass transfer in the $DF3$ model is initially higher than that of the $DF4$ model, indicating that the $DF3$ model compensates to remain on the saturation line. However, the results from the $DF3$ model highlight the distinctions between the two approaches, offering valuable insights that could serve as a foundation for developing a more versatile mass transfer model applicable to a wider range of scenarios.

10.2. Limitations of the Study

While this research provides valuable contributions, several limitations must be acknowledged:

- The $DF3$ and $DF4$ models fail to replicate the delayed temperature response observed experimentally. This highlights a limitation in the energy equation, which assumes that temperature changes occur on the same timescale as pressure changes—an oversimplification during rapid depressurization.
- The focus on horizontal pipelines two-phase CO_2 flow neglects the differences introduced by vertical pipelines. While the experimental data provided by Log et al. were invaluable, vertical pipeline conditions are likely to be encountered in real-world CCS applications and require further investigation.
- The explicit numerical schemes used to calculate the primitive variables in the $DF4$ model exhibit stability challenges near the critical point. Implementing implicit schemes could address these issues, improving reliability at the cost of increased computational time.

10.3. Recommendations

Building on the findings and limitations of this study, several avenues for future research are proposed:

- Incorporating a numerical implicit method for simulating the primitive variables in the $DF4$ and $TF5$ models. Although computationally intensive, this approach will enhance the reliability and stability of simulations, particularly near the critical point of CO_2.
- When the CO_2 is initially in the liquid phase, in a depressurization setting, the CO_2 fluid will undergo a pressure undershoot, where the pressure and temperature initially drop below the saturation line in the first 20 $[ms]$, and then converges back to the saturation line. This phenomena is captured by the $DF4$ model, by implementing a model for the mass transfer. However, this method of calculating the mass transfer is not applicable in all situations, as a pre-defined relaxation parameter, θ, is needed. A future research could look at the mass transfer of the $DF3$ and $DF4$ models, and by comparing the models' behaviours, it may be possible to design a mass transfer model that eliminates the need for a pre-defined relaxation parameter, θ, making it applicable to a wider range of scenarios.
- It was found that for short time-frames (until 0.1 $[s]$), the $DF3$ and $DF4$ models incorrectly predict the velocity of the cooling wave. A possible solution is to add a source term to the energy equation, and delay the cooling of the system. Because the phenomenon happens in the single-phase, the $TF5$ model will not provide additional benefits for this scenario.
- However, further development of the $TF5$ model can provide more accurate results for simulations of longer time-frames (2 $[s]$ till 10$[s]$), addressing discrepancies of up to 40 [°C] observed in the $DF3$ model. Improved temperature predictions are critical for pipeline design, as low temperatures significantly increase material brittleness.
- Future studies should extend these models to vertical pipelines, which are expected to play a significant role in CCS applications. The no-slip condition assumed for two-phase flow in horizontal pipelines may also hold in certain vertical scenarios, offering potential for high-accuracy simulations using the $DF3$, $DF4$, and $TF5$ models.

Bibliography

[1] IPCC, *Summary for policymakers.* IPCC, Aug. 2023, pp. 3–48. DOI: 10.1017/9781009157926.001. [Online]. Available: https://www.cambridge.org/core/books/climate-change-2022-mitigation-of-climate-change/summary-for-policymakers/ABC31CEA863CB6AD8FEB6911A872B321.

[2] S. Bouckaert, A. F. Pales, C. McGlade, *et al.*, "Net Zero by 2050: A Roadmap for the Global Energy Sector," *Net Zero Emmision*, Jan. 2021. [Online]. Available: https://www.iea.org/reports/net-zero-by-2050.

[3] S. Roussanaly, G. Bureau-Cauchois, and J. Husebye, "Costs benchmark of CO2 transport technologies for a group of various size industries," *International journal of greenhouse gas control*, vol. 12, pp. 341–350, Jan. 2013. DOI: 10.1016/j.ijggc.2012.05.008. [Online]. Available: https://doi.org/10.1016/j.ijggc.2012.05.008.

[4] A. M. Log, M. Hammer, and S. T. Munkejord, *A flashing flow model for the rapid depressurization of CO2 in a pipe accounting for bubble nucleation and growth – dataset*, version Version 1, Zenodo, Nov. 2023. DOI: 10.5281/zenodo.8164913. [Online]. Available: https://doi.org/10.5281/zenodo.8164913.

[5] S. T. Munkejord, A. Austegard, H. Deng, M. Hammer, H. G. J. Stang, and S. W. Løvseth, *Depressurization of CO2 in a pipe: High-resolution pressure and temperature data and comparison with model predictions – dataset*, Zenodo, Aug. 2020. DOI: 10.5281/zenodo.3928227. [Online]. Available: https://doi.org/10.5281/zenodo.3928227.

[6] A. M. Log, M. Hammer, H. Deng, A. Austegard, A. Hafner, and S. T. Munkejord, *Depressurization of CO2 in a pipe: Effect of initial state on non-equilibrium two-phase flow – dataset*, Zenodo, Oct. 2023. DOI: 10.5281/zenodo.7669536. [Online]. Available: https://doi.org/10.5281/zenodo.7669536.

[7] M. Istre, *Preventing fracture in CO2 transmission pipeline systems.* [Online]. Available: https://www.projectconsulting.com/pcs-insights/preventing-fracture-in-co2-transmission-system.

[8] M. Baer and J. Nunziato, "A two-phase mixture theory for the deflagration-to-detonation transition (ddt) in reactive granular materials," *International journal of multiphase flow*, vol. 12, no. 6, pp. 861–889, Nov. 1986. DOI: 10.1016/0301-9322(86)90033-9. [Online]. Available: https://doi.org/10.1016/0301-9322(86)90033-9.

[9] M. Hantke, S. Müller, and L. Grabowsky, "News on Baer–Nunziato-type model at pressure equilibrium," *Continuum mechanics and thermodynamics*, vol. 33, no. 3, pp. 767–788, Dec. 2020. DOI: 10.1007/s00161-020-00956-3. [Online]. Available: https://doi.org/10.1007/s00161-020-00956-3.

[10] H. B. Stewart and B. Wendroff, "Two-phase flow: Models and methods," *Journal of computational physics*, vol. 56, no. 3, pp. 363–409, Dec. 1984. DOI: 10.1016/0021-9991(84)90103-7. [Online]. Available: https://doi.org/10.1016/0021-9991(84)90103-7.

[11] S. Dallet, "A comparative study of numerical schemes for the Baer-Nunziato model," *International Journal on Finite Volumes*, vol. 13, pp. 1–37, Dec. 2016. [Online]. Available: https://hal.science/hal-01412148.

[12] N. Andrianov and G. Warnecke, "The Riemann problem for the Baer–Nunziato two-phase flow model," *Journal of computational physics*, vol. 195, no. 2, pp. 434–464, Apr. 2004. DOI: 10.1016/j.jcp.2003.10.006. [Online]. Available: https://doi.org/10.1016/j.jcp.2003.10.006.

[13] H. Lund, T. Flåtten, and S. T. Munkejord, "Depressurization of carbon dioxide in pipelines–Models and methods," *Energy procedia*, vol. 4, pp. 2984–2991, Jan. 2011. DOI: 10.1016/j.egypro.2011.02.208. [Online]. Available: https://doi.org/10.1016/j.egypro.2011.02.208.

[14] P. J. M. Ferrer, T. Flåtten, and S. T. Munkejord, “On the effect of temperature and velocity relaxation in two-phase flow models,” *ESAIM Mathematical Modelling and Numerical Analysis*, vol. 46, no. 2, pp. 411–442, Oct. 2011. DOI: 10.1051/m2an/2011039. [Online]. Available: https://www.cambridge.org/core/journals/esaim-mathematical-modelling-and-numerical-analysis/article/abs/on-the-effect-of-temperature-and-velocity-relaxation-in-twophase-flow-models/EAA9C85427D237DBDDD4DE810F653F52.

[15] A. Morin and T. Flåtten, “A two-fluid four-equation model with instantaneous thermodynamical equilibrium,” *ESAIM Mathematical Modelling and Numerical Analysis*, vol. 50, no. 4, pp. 1167–1192, Sep. 2015. DOI: 10.1051/m2an/2015074. [Online]. Available: https://doi.org/10.1051/m2an/2015074.

[16] R. Saurel, F. Petitpas, and R. Abgrall, “Modelling phase transition in metastable liquids: application to cavitating and flashing flows,” *Journal of Fluid Mechanics*, vol. 607, pp. 313–350, Jun. 2008. DOI: 10.1017/s0022112008002061. [Online]. Available: https://doi.org/10.1017/s0022112008002061.

[17] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, Jan. 2009. DOI: 10.1007/b79761. [Online]. Available: https://doi.org/10.1007/b79761.

[18] A. K. Kapila, R. Menikoff, J. B. Bdzil, S. F. Son, and D. S. Stewart, “Two-phase modeling of deflagration-to-detonation transition in granular materials: Reduced equations,” *Physics of fluids*, vol. 13, no. 10, pp. 3002–3024, Oct. 2001. DOI: 10.1063/1.1398042. [Online]. Available: https://doi.org/10.1063/1.1398042.

[19] X. Guo, X. Yan, J. Yu, *et al.*, “Pressure response and phase transition in supercritical CO2 releases from a large-scale pipeline,” *Applied Energy*, vol. 178, pp. 189–197, Jun. 2016. DOI: 10.1016/j.apenergy.2016.06.026. [Online]. Available: https://doi.org/10.1016/j.apenergy.2016.06.026.

[20] M. Hammer, H. Deng, L. Liu, M. Langsholt, and S. T. Munkejord, “Upward and downward two-phase flow of co2 in a pipe: Comparison between experimental data and model predictions,” *International Journal of Multiphase Flow*, vol. 138, p. 103 590, 2021, ISSN: 0301-9322. DOI: https://doi.org/10.1016/j.ijmultiphaseflow.2021.103590. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301932221000380.

[21] S. T. Munkejord, A. Austegard, H. Deng, M. Hammer, H. J. Stang, and S. W. Løvseth, “Depressurization of CO2 in a pipe: High-resolution pressure and temperature data and comparison with model predictions,” *Energy*, vol. 211, p. 118 560, Aug. 2020. DOI: 10.1016/j.energy.2020.118560. [Online]. Available: https://doi.org/10.1016/j.energy.2020.118560.

[22] A. Log, M. Hammer, H. Deng, A. Austegard, A. Hafner, and S. T. Munkejord, “Depressurization of co2 in a pipe: Effect of initial state on non-equilibrium two-phase flow,” *International Journal of Multiphase Flow*, vol. 170, Oct. 2023. DOI: 10.1016/j.ijmultiphaseflow.2023.104624.

[23] A. M. Log, M. Hammer, and S. T. Munkejord, “A flashing flow model for the rapid depressurization of CO2 in a pipe accounting for bubble nucleation and growth,” *International Journal of Multiphase Flow*, Nov. 2023. DOI: 10.1016/j.ijmultiphaseflow.2023.104666. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301932223002860.

[24] A. M. Log, M. Hammer, H. Deng, A. Austegard, A. Hafner, and S. T. Munkejord, “Depressurization of CO2 in a pipe: Effect of initial state on non-equilibrium two-phase flow,” Ph.D. dissertation, Norwegian University of Science and Technology, Sep. 2023, p. 104 624. DOI: 10.1016/j.ijmultiphaseflow.2023.104624. [Online]. Available: https://doi.org/10.1016/j.ijmultiphaseflow.2023.104624.

[25] D.-Y. Peng and D. B. Robinson, “A New Two-Constant Equation of State,” *Industrial Engineering Chemistry Fundamentals*, vol. 15, no. 1, pp. 59–64, Feb. 1976. DOI: 10.1021/i160057a011. [Online]. Available: https://doi.org/10.1021/i160057a011.

[26] F. H. Harlow and A. Amsden, “FLUID DYNAMICS. A LASL monograph.,” Tech. Rep., Jan. 1971. DOI: 10.2172/4762484. [Online]. Available: https://www.osti.gov/servlets/purl/4762484.

[27] H. Lund and P. Aursand, “Two-Phase Flow of CO2 with Phase Transfer,” *Energy procedia*, vol. 23, pp. 246–255, Jan. 2012. DOI: 10.1016/j.egypro.2012.06.034. [Online]. Available: https://doi.org/10.1016/j.egypro.2012.06.034.

[28] R. Span and W. Wagner, “A New Equation of State for Carbon Dioxide Covering the Fluid Region from the Triple-Point Temperature to 1100 K at Pressures up to 800 MPa,” *Journal of physical and chemical reference data*, vol. 25, no. 6, pp. 1509–1596, Nov. 1996. DOI: 10.1063/1.555991. [Online]. Available: https://doi.org/10.1063/1.555991.

[29] S. T. Munkejord, M. Hammer, and S. W. Løvseth, “CO2 transport: Data and models – A review,” *Applied energy*, vol. 169, pp. 499–523, May 2016. DOI: 10.1016/j.apenergy.2016.01.100. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261916300885.

[30] N. BÖTTCHER, J. TARON, O. KOLDITZ, R. LIEDL, and C.-H. PARK, “Comparison of equations of state for carbon dioxide for numerical simulations,” *Models – Repositories of Knowledge*, pp. 252–253, 2012. [Online]. Available: https://iahs.info/uploads/dms/15938.45-252-260-355-24-Boettcher.pdf.

[31] M. Hammer, Å. Ervik, and S. T. Munkejord, “Method Using a Density–Energy State Function with a Reference Equation of State for Fluid-Dynamics Simulation of Vapor–Liquid–Solid Carbon Dioxide,” *Industrial Engineering Chemistry Research*, vol. 52, no. 29, pp. 9965–9978, Jun. 2013. DOI: 10.1021/ie303516m. [Online]. Available: https://doi.org/10.1021/ie303516m.

[32] X. Zhou, K. Li, R. Tu, J. Yi, Q. Xie, and X. Jiang, “A modelling study of the multiphase leakage flow from pressurised CO 2 pipeline,” *Journal of hazardous materials*, vol. 306, pp. 286–294, Apr. 2016. DOI: 10.1016/j.jhazmat.2015.12.026. [Online]. Available: https://doi.org/10.1016/j.jhazmat.2015.12.026.

[33] S. T. Munkejord, S. Evje, and T. Flåtten, “A musta scheme for a nonconservative two-fluid model,” *SIAM J. Scientific Computing*, vol. 31, pp. 2587–2622, Jan. 2009. DOI: 10.1137/080719273.

[34] J. J. Kreeft and B. Koren, “A new formulation of Kapila’s five-equation model for compressible two-fluid flow, and its numerical treatment,” *Journal of Computational Physics*, vol. 229, no. 18, pp. 6220–6242, Apr. 2010. DOI: 10.1016/j.jcp.2010.04.025. [Online]. Available: https://doi.org/10.1016/j.jcp.2010.04.025.

[35] E. F. Toro, “The weighted average flux method applied to the Euler equations,” *Philosophical transactions. Physical sciences and engineering/Philosophical transactions-Royal Society of London. Physical sciences and engineering*, vol. 341, no. 1662, pp. 499–530, Dec. 1992. DOI: 10.1098/rsta.1992.0113. [Online]. Available: https://doi.org/10.1098/rsta.1992.0113.

[36] D. Furfaro and R. Saurel, “A simple hllc-type riemann solver for compressible non-equilibrium two-phase flows,” *Computers Fluids*, vol. 111, pp. 159–178, 2015, ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2015.01.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045793015000250.

[37] M. Pelanti and K.-M. Shyue, “A mixture-energy-consistent six-equation two-phase numerical model for fluids with interfaces, cavitation and evaporation waves,” *Journal of computational physics*, vol. 259, pp. 331–357, Feb. 2014. DOI: 10.1016/j.jcp.2013.12.003. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0021999113008024?via%3Dihub.

[38] R. De Böck, A. Tijsseling, and B. Koren, “A monotonicity-preserving higher-order accurate finite-volume method for Kapila’s two-fluid flow model,” *Computers Fluids*, vol. 193, p. 104 272, Aug. 2019. DOI: 10.1016/j.compfluid.2019.104272. [Online]. Available: https://doi.org/10.1016/j.compfluid.2019.104272.

[39] C. Han, U. Zahid, J. An, K. Kim, and C. Kim, “CO2 transport: design considerations and project outlook,” *Current Opinion in Chemical Engineering*, vol. 10, pp. 42–48, Sep. 2015. DOI: 10.1016/j.coche.2015.08.001. [Online]. Available: https://doi.org/10.1016/j.coche.2015.08.001.

[40] L. Friedel, “Improved friction pressure drop correlations for horizontal and vertical two-phase pipe flow,” *CiNii Research*, 1979. [Online]. Available: https://cir.nii.ac.jp/crid/1571417124944185600.

[41] A. Laesecke and C. D. Muzny, “Reference Correlation for the Viscosity of Carbon Dioxide,” *Journal of Physical and Chemical Reference Data*, vol. 46, no. 1, p. 013 107, Mar. 2017. DOI: 10.1063/1.4977429. [Online]. Available: https://doi.org/10.1063/1.4977429.

[42] A. Mulero, I. Cachadiña, and M. I. Parra, “Recommended Correlations for the Surface Tension of Common Fluids,” *Journal of Physical and Chemical Reference Data*, vol. 41, no. 4, Dec. 2012. DOI: 10.1063/1.4768782. [Online]. Available: https://doi.org/10.1063/1.4768782.

[43] U. S. of Commerce, *Thermophysical Properties of Fluid Systems*, 2024. [Online]. Available: https://webbook.nist.gov/chemistry/fluid/.

[44] F. Aakenes, "Frictional pressure-drop models for steady-state and transient two-phase flow of carbon dioxide," M.S. thesis, Norwegian University of Science and Technology, 2012. [Online]. Available: https://www.sintef.no/globalassets/project/co2-dynamics/publications/aakenes_master_friction-modelling-comparison.pdf.

[45] S. Churchill, "Friction-factor equation spans all fluid-flow regimes.," *Chemical Engineering*, vol. 84, no. 24, pp. 91–92, Jan. 1977. [Online]. Available: http://pascal-francis.inist.fr/vibad/index.php?action=getRecordDetail&idt=PASCAL7860099142.

[46] Z. Bilicki and J. Kestin, "Physical aspects of the relaxation model in two-phase flow," *Proceedings of the Royal Society of London A Mathematical and Physical Sciences*, vol. 428, no. 1875, pp. 379–397, Apr. 1990. DOI: 10.1098/rspa.1990.0040. [Online]. Available: https://doi.org/10.1098/rspa.1990.0040.

[47] P. Kumar, B. Sanderse, P. Rosen Esquivel, and R. Henkes, "A new temperature evolution equation that enforces thermodynamic vapour-liquid equilibrium in multiphase flows -application to co 2 modeling," CWI, Tech. Rep., Aug. 2024. DOI: 10.13140/RG.2.2.26156.48004.

[48] J. G. Collier and J. R. Thome, *Convective Boiling and Condensation*. Clarendon Press, May 1994. DOI: 10.1093/oso/9780198562825.001.0001. [Online]. Available: https://doi.org/10.1093/oso/9780198562825.001.0001.

[49] T. Flåtten, A. Morin, and S. T. Munkejord, "Wave propagation in multicomponent flow models," *SIAM Journal on Applied Mathematics*, vol. 70, no. 8, pp. 2861–2882, Jan. 2010. DOI: 10.1137/090777700. [Online]. Available: https://doi.org/10.1137/090777700.

[50] A. B. Wood and R. B. Lindsay, *A textbook of sound*. Jan. 1930. [Online]. Available: https://doi.org/10.1063/1.3059819.

[51] L. LANDAU and E. LIFSHITZ, "Chapter viii - sound," in *Fluid Mechanics (Second Edition)*, L. LANDAU and E. LIFSHITZ, Eds., Second Edition, Pergamon, 1987, pp. 251–312, ISBN: 978-0-08-033933-7. DOI: https://doi.org/10.1016/B978-0-08-033933-7.50016-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780080339337500167.

[52] C. E. Brennen, *Fundamentals of Multiphase Flow*. Cambridge University Press, Apr. 2005. DOI: 10.1017/cbo9780511807169. [Online]. Available: https://www.cambridge.org/core/books/fundamentals-of-multiphase-flow/FC7E6D7E54AC9D1C178EDF88D6A75FFF.

[53] R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems* (Cambridge Texts in Applied Mathematics). Cambridge University Press, 2002.

[54] P. Batten, N. Clarke, C. Lambert, and D. M. Causon, "On the Choice of Wavespeeds for the HLLC Riemann Solver," *SIAM Journal on Scientific Computing*, vol. 18, no. 6, pp. 1553–1570, Nov. 1997. DOI: 10.1137/s1064827593260140. [Online]. Available: https://doi.org/10.1137/s1064827593260140.

[55] R. Courant, K. Friedrichs, and H. Lewy, "Über die partiellen differenzengleichungen der mathematischen physik," *Mathematische Annalen*, vol. 100, no. 1, pp. 32–74, 1928.

[56] M. Iri and K. Kubota, "Automatic differentiation: Introduction, history and rounding error estimationautomatic differentiation: Introduction, history and rounding error estimation," in *Encyclopedia of Optimization*, C. A. Floudas and P. M. Pardalos, Eds. Boston, MA: Springer US, 2009, pp. 153–159, ISBN: 978-0-387-74759-0. DOI: 10.1007/978-0-387-74759-0_26. [Online]. Available: https://doi.org/10.1007/978-0-387-74759-0_26.

[57] Y.-H. Fang, H.-Z. Lin, J.-J. Liu, and C.-J. Lin, *A step-by-step introduction to the implementation of automatic differentiation*, 2024. arXiv: 2402.16020 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2402.16020.

[58] E. Johnsen and T. Colonius, "Implementation of WENO schemes in compressible multicomponent flow problems," *Journal of Computational Physics*, vol. 219, no. 2, pp. 715–732, Jun. 2006. DOI: 10.1016/j.jcp.2006.04.018. [Online]. Available: https://doi.org/10.1016/j.jcp.2006.04.018.

[59] Ø. Wilhelmsen, A. Aasen, G. Skaugen, *et al.*, "Thermodynamic modeling with equations of state: Present challenges with established methods," *Industrial & Engineering Chemistry Research*, vol. 56, no. 13, pp. 3503–3515, 2017. DOI: 10.1021/acs.iecr.7b00317. [Online]. Available: https://doi.org/10.1021/acs.iecr.7b00317.

[60] K. E. T. Giljarhus, S. T. Munkejord, and G. Skaugen, “Solution of the Span–Wagner Equation of State Using a Density–Energy State Function for Fluid-Dynamic Simulation of Carbon Dioxide,” *Industrial Engineering Chemistry Research*, vol. 51, no. 2, pp. 1006–1014, Nov. 2011. DOI: 10.1021/ie201748a. [Online]. Available: https://doi.org/10.1021/ie201748a.

[61] G. A. Sod, “A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws,” *Journal of Computational Physics*, vol. 27, no. 1, pp. 1–31, Apr. 1978. DOI: 10.1016/0021-9991(78)90023-2. [Online]. Available: https://doi.org/10.1016/0021-9991(78)90023-2.

[62] SLB, *Olga Dynamic Multiphase Flow Simulator*, Jul. 2024. [Online]. Available: https://www.slb.com/products-and-services/delivering-digital-at-scale/software/olga/olga-dynamic-multiphase-flow-simulator/.

IV

APPENDICES

A
Derivation speed of sound

Consider a pipeline with a constant cross-sectional area. The mass flow rate in the pipeline is:

$$\frac{dM}{dt} = \frac{d(\rho V)}{dt} = \frac{d(\rho Ax)}{dt} = \rho A\frac{dx}{dt} = \rho Av. \tag{A.1}$$

The mass flow in and out of the pipeline is naturally equal. However, when a sound wave propagates, the wave produces small perturbations in the density, pressure and velocity. The continuity equation states that the mass flow rate should be constant, so it becomes:

$$\rho Av = (\rho + \partial\rho)A(v + \partial v). \tag{A.2}$$

The area AA remains constant and can be cancelled. Additionally, because the perturbations are assumed to be small, the product $\partial\rho \cdot \partial v$ is negligible. This simplifies to:

$$\begin{aligned} \rho Av &= (\rho + \partial\rho)A(v + \partial v), && \text{(A.3)}\\ \rho v &= \rho v + \rho\partial v + v\partial\rho + \partial\rho\partial v, && \text{(A.4)}\\ 0 &= \rho\partial v + v\partial\rho, && \text{(A.5)}\\ -v &= \frac{\rho\partial v}{\partial\rho}. && \text{(A.6)} \end{aligned}$$

The force on the fluid is the sum of the pressure times the area. This is expressed as:

$$\begin{aligned} F = p\,dy\,dz - (p\ + \partial p)\,dy\,dz &= -\partial p\,dy\,dz, && \text{(A.7)}\\ m\frac{\partial v}{\partial t} &= -\partial p\,dy\,dz, && \text{(A.8)}\\ \frac{\partial v}{\partial t} = \frac{-\partial p\,dy\,dz}{m} &= \frac{-\partial p\,dy\,dz}{\rho\,dx\,dy\,dz} = \frac{-\partial p}{\rho\,dx}, && \text{(A.9)}\\ \partial v &= \frac{-\partial p}{\rho}\frac{1}{v}. && \text{(A.10)} \end{aligned}$$

Substituting Equation A.10 into Equation A.6, it becomes:

$$\begin{aligned} -v &= \frac{\rho}{\partial\rho}\frac{-\partial p}{\rho}\frac{1}{v}, && \text{(A.11)}\\ v^2 &= \frac{\partial p}{\partial\rho}. && \text{(A.12)} \end{aligned}$$

Thus, the speed of sound c is:

$$c^2 = \frac{\partial p}{\partial\rho}. \tag{A.13}$$

B

$TF6$ model derivation

This is the standard model for two phases with the assumption that the pressure is the same over the two phases.

$$\frac{\partial}{\partial t}\left(\rho_g\alpha_g\right)+\frac{\partial}{\partial x}\left(\rho_g\alpha_g v_g\right)=0 \tag{B.1}$$

$$\frac{\partial}{\partial t}\left(\rho_l\alpha_l\right)+\frac{\partial}{\partial x}\left(\rho_l\alpha_l v_l\right)=0 \tag{B.2}$$

$$\frac{\partial}{\partial t}\left(\rho_g\alpha_g v_g\right)+\frac{\partial}{\partial x}\left(\rho_g\alpha_g v_g^2\right)+\alpha_g\frac{\partial p}{\partial x}+\tau_i=\rho_g\alpha_g g_x \tag{B.3}$$

$$\frac{\partial}{\partial t}\left(\rho_l\alpha_l v_l\right)+\frac{\partial}{\partial x}\left(\rho_l\alpha_l v_l^2\right)+\alpha_l\frac{\partial p}{\partial x}-\tau_i=\rho_l\alpha_l g_x \tag{B.4}$$

$$\frac{\partial E_g}{\partial t}+\frac{\partial}{\partial x}\left(E_g v_g+\alpha_g v_g p\right)+p\frac{\partial\alpha_g}{\partial t}+v_\tau\tau_i=\rho_g\alpha_g v_g g_x \tag{B.5}$$

$$\frac{\partial E_l}{\partial t}+\frac{\partial}{\partial x}\left(E_l v_l+\alpha_l v_l p\right)+p\frac{\partial\alpha_l}{\partial t}-v_\tau\tau_i=\rho_l\alpha_l v_l g_x \tag{B.6}$$

Here v_τ and τ_k are given below to ensure hyperbolicity:

$$v_\tau=\frac{\alpha_l\gamma_g v_g+\alpha_g\gamma_l v_l}{\alpha_l\gamma_g+\alpha_g\gamma_l} \tag{B.7}$$

$$\tau_i=\Delta_i p\frac{\partial\alpha_g}{\partial x}+\mathscr{F}(v_g-v_l) \tag{B.8}$$

Now we want to write it with the material derivative equation, this is the following equation:

$$\frac{D\Phi_i}{Dt}=\frac{\partial\Phi_i}{\partial t}+v_i\frac{\partial\Phi_i}{\partial x} \tag{B.9}$$

Furthermore, the partial differential equations are decomposed with the product rule. For Equation B.1, it becomes:

$$\rho_i\frac{\partial\alpha_i}{\partial t}+\alpha_i\frac{\partial\rho_i}{\partial t}+v_i\alpha_i\frac{\partial\rho_i}{\partial x}+\rho_i\frac{\partial}{\partial x}(\alpha_i v_i)=0 \tag{B.10}$$

$$\alpha_i(\frac{\partial\rho_i}{\partial t}+v_i\frac{\partial\rho_i}{\partial x})=-\rho_i\frac{\partial\alpha_i}{\partial t}-\rho_i\frac{\partial}{\partial x}(\alpha_i v_i) \tag{B.11}$$

$$\alpha_i\frac{D\rho_i}{Dt}=-\rho_i\left(\frac{\partial\alpha_i}{\partial t}+\frac{\partial}{\partial x}(\alpha_i v_i)\right) \tag{B.12}$$

With the same process, we get the following equations for momentum transfer and energy:

$$\rho_i\alpha_i\frac{Dv_i}{Dt}+\alpha_i\frac{\partial p}{\partial x}-\tau_i=\rho_i\alpha_i g_x \tag{B.13}$$

$$\rho_i\alpha_i\frac{D}{Dt}(e_i+\frac{1}{2}v_i^2)+\frac{\partial(\alpha_i v_i p)}{\partial x}+p\frac{\alpha_i}{\partial t}-v_\tau\tau_i=\rho_i\alpha_i v_i g_x \tag{B.14}$$

Now we multiply Equation B.13 with v_i, and we subtract it from Equation B.14. This becomes:

$$\rho_i\alpha_i v_i\frac{Dv_i}{Dt}+\alpha_i v_i\frac{\partial p}{\partial x}-\tau_i v_i=\rho_i\alpha_i v_i g_x$$

$$\frac{\partial(\alpha_i v_i p)}{\partial x}=p\frac{\partial(\alpha_i v_i)}{\partial x}+\alpha_i v_i\frac{\partial p}{\partial x}$$

$$\frac{D}{Dt}(e_i+\frac{1}{2}v_i^2)=\frac{De_i}{Dt}+\frac{D}{Dt}(\frac{1}{2}v_i^2),\quad v_i\frac{Dv_i}{Dt}=\frac{D}{Dt}(\frac{1}{2}v_i^2)$$

$$\rho_i\alpha_i\frac{D}{Dt}(e_i+\frac{1}{2}v_i^2)-\rho_i\alpha_i v_i\frac{Dv_i}{Dt}+\frac{\partial(\partial\alpha_i v_i p)}{\partial x}+-\alpha_i v_i\frac{\partial p}{\partial x}+p\frac{\alpha_i}{\partial t}-v_\tau\tau_i+v_i\tau_i=\rho_i\alpha_i v_i g_x-\rho_i\alpha_i v_i g_x$$

$$\rho_i\alpha_i\frac{De_i}{Dt}=-p\frac{\partial(\alpha_k v_i)}{\partial x}-p\frac{\partial\alpha_i}{\partial t}-\tau_i(v_i-v_\tau) \tag{B.15}$$

however, it is hard to model $\partial\alpha_i/\partial t$. Therefore, we want to rewrite it. This are a lot of steps and so, we start at the beginning. First, the thermodynamic parameter is derived, and then an equation for $\partial\alpha_i/\partial t$. It is started with:

$$dp=\left(\frac{\partial p}{\partial\rho}\right)_s d\rho+\left(\frac{\partial p}{\partial s}\right)_\rho ds=c^2 d\rho+\left(\frac{\partial p}{\partial s}\right)_\rho ds \tag{B.16}$$

$$\left(\frac{\partial p}{\partial s}\right)_\rho=\left(\frac{\partial p}{\partial T}\right)_\rho\left(\frac{\partial T}{\partial s}\right)_\rho \tag{B.17}$$

$$ds=\frac{dq_{rev}}{T},\quad C_v=\frac{dq_{rev}}{dT},\quad\Rightarrow\quad\frac{ds}{dT}=\frac{C_v}{T} \tag{B.18}$$

$$\gamma\equiv\frac{1}{\rho C_v}\left(\frac{\partial p}{\partial T}\right)_\rho \tag{B.19}$$

$$dp=c^2 d\rho+\gamma\rho C_v\cdot\frac{T}{C_v}ds=c^2 d\rho+\gamma\rho T ds \tag{B.20}$$

Then we use the first law of thermodynamics:

$$Tds=de+pd\frac{1}{\rho} \tag{B.21}$$

Using reciprocal rule, we can rewrite it into:

$$g^{'}(x)=\frac{-f^{'}(x)}{f(x)^2}$$

$$d\frac{1}{\rho}=-\frac{1}{\rho^2}d\rho$$

Hereby, the first law can be rewritten into:

$$Tds=de-\frac{p}{\rho^2}d\rho \tag{B.22}$$

Then filling Equation B.22 into Equation B.20, it becomes:

$$dp=c^2 d\rho-\frac{\gamma p}{\rho}d\rho+\gamma\rho de \tag{B.23}$$

This can be rewritten by multiplying with α_i into:

$$\alpha_i\frac{Dp}{Dt}=(c_i^2-\frac{\gamma_i p}{\rho_i})\alpha_i\frac{D\rho_i}{Dt}+\gamma_i\rho_i\alpha_i\frac{De_i}{Dt} \tag{B.24}$$

Now we will use Equation B.12 and Equation B.15 and fill in for $\alpha_K\cdot D\rho_k/Dt$ and $\alpha_k\rho_k\cdot De_k/Dt$. This will lead to the following:

$$\alpha_i\frac{Dp}{Dt}=-\rho_i c_i^2\frac{\partial\alpha_i}{\partial t}-\rho_i c_i^2\frac{\partial(\alpha_i v_i)}{\partial x}+\gamma_i p\frac{\partial\alpha_i}{\partial t}+\gamma_i p\frac{\partial(\alpha_i v_i)}{\partial t}-\gamma_i p\frac{\partial\alpha_i}{\partial t}-\gamma_i p\frac{\partial(\alpha_i v_i)}{\partial t}-\gamma_i(v_i-v_\tau)\tau_i \tag{B.25}$$

$$\alpha_i\frac{Dp}{Dt}=-\rho_i c_i^2\frac{\partial\alpha_i}{\partial t}-\rho_i c_i^2\frac{\partial(\alpha_i v_i)}{\partial x}-\gamma_i(v_i-v_\tau)\tau_i \tag{B.26}$$

$$\frac{Dp}{Dt}=\frac{\partial p}{\partial t}+v_i\frac{\partial p}{\partial x} \tag{B.27}$$

Filling in $k = g$ and $k = l$, we get:

$$\alpha_g\left(\frac{\partial p}{\partial t} + v_g\frac{\partial p}{\partial x}\right) = -\rho_g c_g^2\frac{\partial \alpha_g}{\partial t} - \rho_g c_g^2\frac{\partial\left(\alpha_g v_g\right)}{\partial x} - \gamma_g\left(v_g - v_\tau\right)\tau_g \tag{B.28}$$

$$\alpha_l\left(\frac{\partial p}{\partial t} + v_l\frac{\partial p}{\partial x}\right) = -\rho_l c_l^2\frac{\partial \alpha_l}{\partial t} - \rho_l c_l^2\frac{\partial\left(\alpha_l v_l\right)}{\partial x} - \gamma_l\left(v_l - v_\tau\right)\tau_l \tag{B.29}$$

$$\frac{\partial \alpha_g}{\partial t} + \frac{\partial \alpha_l}{\partial t} = 0 \tag{B.30}$$

With Equation B.30, rewriting Equation B.28 and Equation B.29 into $\partial p/\partial t = ...$ and using the fact that $\tau_i = \tau_l = -\tau_g$, we get the following equation:

$$\begin{aligned}\sigma\frac{\partial \alpha_l}{\partial t} &= \rho_g\alpha_l c_g^2\frac{\partial}{\partial x}\left(\alpha_g v_g\right) - \rho_l\alpha_g c_l^2\frac{\partial}{\partial x}\left(\alpha_l v_l\right) + \alpha_g\alpha_l\left(v_g - v_l\right)\frac{\partial p}{\partial x}\\ &\qquad - \tau_i\left(\alpha_l\gamma_g\left(v_g - v_\tau\right) + \alpha_g\gamma_l\left(v_l - v_\tau\right)\right)\\ \sigma &= \alpha_g\rho_g c_l^2 + \alpha_l\rho_l c_g^2\end{aligned} \tag{B.31}$$

In addition, it is assumed the following formula for v_τ. Hereby, we can simplify Equation B.31.

$$\begin{aligned}\alpha_l\gamma_g\left(v_g - v_\tau\right) + \alpha_g\gamma_l(v_l - v_\tau) \quad &\Rightarrow \quad \alpha_l\gamma_g v_g + \alpha_g\gamma_l v_l - v_\tau(\alpha_g\gamma_l + \alpha_l\gamma_g)\\ \alpha_l\gamma_g v_g + \alpha_g\gamma_l v_l &- \alpha_l\gamma_g v_g - \alpha_g\gamma_l v_l = 0\end{aligned} \tag{B.32}$$

$$\sigma\frac{\partial \alpha_l}{\partial t} = \rho_g\alpha_l c_g^2\frac{\partial}{\partial x}\left(\alpha_g v_g\right) - \rho_l\alpha_g c_l^2\frac{\partial}{\partial x}\left(\alpha_l v_l\right) + \alpha_g\alpha_l\left(v_g - v_l\right)\frac{\partial p}{\partial x} \tag{B.33}$$

Now that we have an equation for $\partial\alpha_l/\partial t$, we can fill it in Equation B.6. This then becomes the final set of equations:

$$\eta = \frac{p}{\rho_l\alpha_g c_l^2 + \rho_g\alpha_l c_g^2} \tag{B.34}$$

$$\frac{\partial E_l}{\partial t} + \frac{\partial\left(E_l v_l\right)}{\partial x} + \frac{\partial\left(\alpha_l v_l p\right)}{\partial x} + \eta\rho_g\alpha_l c_g^2\frac{\partial(\alpha_g v_g)}{\partial x} - \eta\rho_l\alpha_g c_l^2\frac{\partial(\alpha_l v_l)}{\partial x} + \eta\alpha_g\alpha_l(v_g - v_l)\frac{\partial p}{\partial x} = \rho_l\alpha_l v_l g_x + v_\tau\tau_i \tag{B.35}$$

$$\frac{\partial\left(\alpha_l v_l p\right)}{\partial x} = p\frac{(\partial\alpha_l v_l)}{\partial x} + \alpha_l v_l\frac{\partial p}{\partial x} \tag{B.36}$$

$$-\eta\rho_l\alpha_g c_l^2\frac{\partial(\alpha_l v_l)}{\partial x} + p\frac{\partial(\alpha_l v_l)}{\partial x} = \eta\rho_g\alpha_l c_g^2\frac{\partial(\alpha_l v_l)}{\partial x} \tag{B.37}$$

$$\frac{\partial E_l}{\partial t} + \frac{\partial\left(E_l v_l\right)}{\partial x} + \left(\alpha_l v_l + \eta\alpha_g\alpha_l(v_g - v_l)\right)\frac{\partial p}{\partial x} + \eta\rho_g\alpha_l c_g^2\frac{\partial}{\partial x}(\alpha_g v_g + \alpha_l v_l) = \rho_l\alpha_l v_l g_x + v_\tau\tau_i \tag{B.38}$$

With the same logic, we can fill in $\partial\alpha_g/\partial t = -\partial\alpha_l/\partial t$ in Equation B.5, and this becomes:

$$\frac{\partial E_g}{\partial t} + \frac{\partial\left(E_g v_g\right)}{\partial x} + \left(\alpha_g v_g - \eta\alpha_g\alpha_l(v_g - v_l)\right)\frac{\partial p}{\partial x} + \eta\rho_l\alpha_g c_l^2\frac{\partial}{\partial x}(\alpha_g v_g + \alpha_l v_l) = \rho_g\alpha_g v_g g_x - v_\tau\tau_i \tag{B.39}$$

Now we can write the whole set of equations in the following mathematical form:

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} + \boldsymbol{B}(\boldsymbol{u})\frac{\partial \boldsymbol{w}(\boldsymbol{u})}{\partial x} = \boldsymbol{s}(\boldsymbol{u}) \tag{B.40}$$

$$(\boldsymbol{u}) = \begin{bmatrix} \rho_g\alpha_g \\ \rho_l\alpha_l \\ \rho_g\alpha_g v_g \\ \rho_l\alpha_l v_l \\ E_g \\ E_l \end{bmatrix}, \quad \boldsymbol{f}(\boldsymbol{u}) = \begin{bmatrix} \rho_g\alpha_g v_g \\ \rho_l\alpha_l v_l \\ \rho_g\alpha_g v_g^2 \\ \rho_l\alpha_l v_l^2 \\ E_g v_g \\ E_l v_l \end{bmatrix}, \quad \boldsymbol{s}(\boldsymbol{u}) = \begin{bmatrix} 0 \\ 0 \\ \rho_g\alpha_g g_x - \tau_i \\ \rho_l\alpha_l g_x + \tau_i \\ \rho_g\alpha_g v_g g_x - v_\tau\tau_i \\ \rho_l\alpha_l v_l g_x + v_\tau\tau_i \end{bmatrix} \tag{B.41}$$

$$\boldsymbol{B}(\boldsymbol{u}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \alpha_\mathrm{g} & 0 & 0 \\ \alpha_\ell & 0 & 0 \\ -\eta\alpha_\mathrm{g}\alpha_\ell\left(v_\mathrm{g} - v_\ell\right) & \eta\rho_\ell\alpha_\mathrm{g}c_\ell^2 & \eta\rho_\ell\alpha_\mathrm{g}c_\ell^2 \\ \eta\alpha_\mathrm{g}\alpha_\ell\left(v_\mathrm{g} - v_\ell\right) & \eta\rho_\mathrm{g}\alpha_\ell c_\mathrm{g}^2 & \eta\rho_\mathrm{g}\alpha_\ell c_\mathrm{g}^2 \end{bmatrix} \tag{B.42}$$

$$\boldsymbol{w}(\boldsymbol{u}) = \begin{bmatrix} p \\ \alpha_g v_g \\ \alpha_l v_l \end{bmatrix} \tag{B.43}$$

C

$TF5$ model derivation

The assumption is made that $\mathscr{F} \to \infty$ or in other words, the limit of stiff velocity relaxation. The results will be that $v_l = v_g = v$, and this makes $\tau_i = 0$ (Equation B.8). In addition, Equation B.3 and Equation B.4 should be replaced with their sum. This gives the following momentum transfer equation:

$$\frac{\partial}{\partial t}((\rho_g\alpha_g + \rho_l\alpha_l)v) + \frac{\partial}{\partial x}((\rho_g\alpha_g + \rho_l\alpha_l)v^2) + \frac{\partial p}{\partial x} = (\rho_g\alpha_g + \rho_l\alpha_l)g_x \tag{C.1}$$

When looking at Equation B.3 and Equation B.4, the limit must satisfy:

$$\lim_{\mathscr{F}\to\infty} \tau_\mathrm{i} = \left(\frac{\rho_\mathrm{g}\alpha_\mathrm{g}}{\rho_\mathrm{g}\alpha_\mathrm{g} + \rho_\ell\alpha_\ell} - \alpha_\mathrm{g}\right)\frac{\partial p}{\partial x} \tag{C.2}$$

Filling this in the energy equations, the total set of equations becomes then:

$$\begin{gathered}
\frac{\partial}{\partial t}\left(\rho_g\alpha_g\right) + \frac{\partial}{\partial x}\left(\rho_g\alpha_g v\right) = 0 \\
\frac{\partial}{\partial t}\left(\rho_l\alpha_l\right) + \frac{\partial}{\partial x}\left(\rho_l\alpha_l v\right) = 0 \\
\frac{\partial}{\partial t}((\rho_g\alpha_g + \rho_l\alpha_l)v) + \frac{\partial}{\partial x}((\rho_g\alpha_g + \rho_l\alpha_l)v^2) + \frac{\partial p}{\partial x} = (\rho_g\alpha_g + \rho_l\alpha_l)g_x \\
\frac{\partial E_g}{\partial t} + \frac{\partial}{\partial x}(E_g v) + v\frac{\rho_g\alpha_g}{\rho_g\alpha_g + \rho_l\alpha_l}\frac{\partial p}{\partial x} + \eta\rho_l\alpha_g c_l^2\frac{\partial v}{\partial x} = \rho_g\alpha_g v g_x \\
\frac{\partial E_l}{\partial t} + \frac{\partial}{\partial x}(E_l v) + v\frac{\rho_l\alpha_l}{\rho_g\alpha_g + \rho_l\alpha_l}\frac{\partial p}{\partial x} + \eta\rho_g\alpha_l c_g^2\frac{\partial v}{\partial x} = \rho_l\alpha_l v g_x \\
\eta = \frac{p}{\rho_l\alpha_g c_l^2 + \rho_g\alpha_l c_g^2}
\end{gathered} \tag{C.3}$$

Rewriting energy into entropy

We want to derive the energy equations into entropy equations as it can then be rewritten into the material derivative. It is started from the entropy equation and that is derived until the same energy equations as in Equation C.3.

$$\rho_i\alpha_i T_i\left(\frac{\partial s_i}{\partial t} + v\frac{\partial s_i}{\partial x}\right) = \sum_{j\neq i} H_{ij}(T_j - T_i), \qquad H_{ij} \geq 0, \quad H_{ij} - H_{ji} = 0 \tag{C.4}$$

Using Equation B.22 and filling in for ds, the following is obtained

$$\rho_i\alpha_i\left(\frac{\partial e_i}{\partial t} + v\frac{\partial e_i}{\partial x}\right) - \frac{p\alpha_i}{\rho_i}\left(\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x}\right) = \sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.5}$$

Now using Equation C.6 and rewritten into $\partial e_i = ...$, we get Equation C.7. Then, we can get rid of all the terms outside e_i, as the mass transfer terms are equal to 0

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t} = \rho_i\alpha_i\frac{\partial e_i}{\partial t} + e_i\frac{\partial \rho_i\alpha_i}{\partial t}, \quad \rightarrow \quad \rho_i\alpha_i\frac{\partial e_i}{\partial t} = \frac{\partial e_i\rho_i\alpha_i}{\partial t} - e_i\frac{\partial \rho_i\alpha_i}{\partial t} \tag{C.6}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t} - e_i\frac{\partial \rho_i\alpha_i}{\partial t} + \frac{\partial e_i\rho_i\alpha_i v}{\partial x} - e_i\frac{\partial \rho_i\alpha_i v}{\partial x} - \frac{p\alpha_i}{\rho_i}\left(\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x}\right) = \sum_{j\neq i} H_{ij}(T_j - T_i)) \tag{C.7}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t} + \frac{\partial e_i\rho_i\alpha_i v}{\partial x} - \frac{p\alpha_i}{\rho_i}\left(\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x}\right) = \sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.8}$$

The pressure term can be rewritten, using Equation C.7 again, in the following way:

$$\frac{\partial \rho_i\alpha_i}{\partial t} + \frac{\partial \rho_i\alpha_i v}{\partial x} = \alpha_i\frac{\partial \rho_i}{\partial t} + \rho_i\frac{\partial \alpha_i}{\partial t} + \alpha_i v\frac{\partial \rho_i}{\partial x} + \rho_i\frac{\partial \alpha_i v}{\partial x} = 0 \tag{C.9}$$

$$\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x} = -\frac{\rho_i}{\alpha_i}\left(\frac{\partial \alpha_i}{\partial t} + \frac{\partial \alpha_i v}{\partial x}\right) \tag{C.10}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t} + \frac{\partial e_i\rho_i\alpha_i v}{\partial x} + p\left(\frac{\partial \alpha_i}{\partial t} + \frac{\partial \alpha_i v}{\partial x}\right) = \sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.11}$$

We fill in Equation B.20 into Equation C.4 to get:

$$\frac{\partial p}{\partial t} = c_i^2\frac{\partial \rho_i}{\partial t} + \gamma_i\rho_i T_i\frac{\partial s_i}{\partial t} \tag{C.12}$$

$$\frac{\partial p}{\partial x} = c_i^2\frac{\partial \rho_i}{\partial x} + \gamma_i\rho_i T_i\frac{\partial s_i}{\partial x} \tag{C.13}$$

$$\frac{\partial s_i}{\partial t} + v\frac{\partial s_i}{\partial x} = \frac{1}{\gamma_i\rho_i T_i}\left(\frac{\partial p}{\partial t} + \frac{\partial p}{\partial x} - c_i^2\frac{\partial \rho_i}{\partial t} - c_i^2\frac{\partial \rho_i}{\partial x}\right) \tag{C.14}$$

$$\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x} = c_i^2\left(\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x}\right) + \frac{\gamma_i}{\alpha_i}\sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.15}$$

Then again with using Equation B.1, we fill in for $\partial\rho$,

$$\frac{\partial \rho_i\alpha_i}{\partial t} + \frac{\partial \rho_i\alpha_i v}{\partial x} = \alpha_i\frac{\partial \rho_i}{\partial t} + \rho_i\frac{\partial \alpha_i}{\partial t} + \alpha_i v\frac{\partial \rho_i}{\partial x} + \rho_i\frac{\partial \alpha_i v}{\partial x} = 0 \tag{C.16}$$

$$\frac{\partial \rho_i}{\partial t} + v\frac{\partial \rho_i}{\partial x} = -\frac{\rho_i}{\alpha_i}\left(\frac{\partial \alpha_i}{\partial t} + \frac{\partial \alpha_i v}{\partial x}\right) \tag{C.17}$$

$$\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x} = -c_i^2\left(\frac{\rho_i}{\alpha_i}\frac{\partial \alpha_i}{\partial t} + \frac{\rho_i}{\alpha_i}\frac{\partial \alpha_i v}{\partial x}\right) + \frac{\gamma_i}{\alpha_i}\sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.18}$$

$$\frac{\alpha_i}{\rho_i c_i^2}\left(\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x}\right) + \frac{\partial \alpha_i}{\partial t} + \frac{\partial \alpha_i v}{\partial x} = \frac{\gamma_i}{\rho_i c_i^2}\sum_{j\neq i} H_{ij}(T_j - T_i) \tag{C.19}$$

When summing both equation, all $\partial\alpha$ disappears as $\alpha_l + \alpha_g = 1$ so $\partial(\alpha_l + \alpha_g) = 0$. it becomes then:

$$\left(\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x}\right)\left(\frac{\alpha_g}{\rho_g c_g^2} + \frac{\alpha_l}{\rho_l c_l^2}\right) + \frac{\partial v}{\partial x} = \sum_{j\neq i} H_{ij}\left(\frac{\gamma_l}{\rho_l c_l^2} - \frac{\gamma_g}{\rho_g c_g^2}\right)(T_g - T_l) \tag{C.20}$$

This can be simplified by using the 'Wood' speed of sound [50]:

$$c^2 = \left(\rho\sum_i^N \frac{\alpha_i}{\rho_i c_i^2}\right)^{-1} \tag{C.21}$$

Equation C.20 becomes:

$$\left(\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}\right)+\rho c^2\frac{\partial v}{\partial x}=\rho c^2\sum_{j\neq i}H_{ij}\left(\frac{\gamma_l}{\rho_l c_l^2}-\frac{\gamma_g}{\rho_g c_g^2}\right)(T_g-T_l) \tag{C.22}$$

This is called the 'pressure evolution' equation. Now we want to obtain the internal energy evolution equation. This is done by substituting Equation C.19 into Equation C.11:

$$\frac{\partial \alpha_i}{\partial t}+\frac{\partial \alpha_i v}{\partial x}=\frac{\gamma_i}{\rho_i c_i^2}\sum_{j\neq i}H_{ij}(T_j-T_i)-\frac{\alpha_i}{\rho_i c_i^2}\left(\frac{\partial p}{\partial t}-v\frac{\partial p}{\partial x}\right) \tag{C.23}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t}+\frac{\partial e_i\rho_i\alpha_i v}{\partial x}+p\left(\frac{\gamma_i}{\rho_i c_i^2}\sum_{j\neq i}H_{ij}(T_j-T_i)-\frac{\alpha_i}{\rho_i c_i^2}\left(\frac{\partial p}{\partial t}-v\frac{\partial p}{\partial x}\right)\right)=H_{ij}(T_j-T_i) \tag{C.24}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t}+\frac{\partial e_i\rho_i\alpha_i v}{\partial x}-\frac{\alpha_i p}{\rho_i c_i^2}\left(\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}\right)=\left(1-\frac{\gamma_i p}{\rho_i c_i^2}\right)\sum_{j\neq i}H_{ij}(T_j-T_i) \tag{C.25}$$

Now, the final intern energy evolution equation can be obtained by filling in Equation C.22 into Equation C.25:

$$\left(\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}\right)=\rho c^2\sum_{j\neq i}H_{ij}\left(\frac{\gamma_l}{\rho_l c_l^2}-\frac{\gamma_g}{\rho_g c_g^2}\right)(T_g-T_l)-\rho c^2\frac{\partial v}{\partial x} \tag{C.26}$$

$$\frac{\partial e_i\rho_i\alpha_i}{\partial t}+\frac{\partial e_i\rho_i\alpha_i v}{\partial x}+\alpha_i p\frac{\rho c^2}{\rho_i c_i^2}\frac{\partial v}{\partial x}=\theta_i\sum_{j\neq i}H_{ij}(T_j-T_i)+\frac{\rho c^2}{\rho_i c^2}\alpha_i\sum_{k,j>k}H_{kj}(\theta_j-\theta_k)(T_j-T_k) \tag{C.27}$$

$$\theta_i=1-\frac{\gamma_i p}{\rho_i c_i^2}$$

Lastly, we want to rewrite it to an evolution equation for E, so that we proved how the energy evolution equation can be rewritten into an entropy evolution equation.

$$\frac{\partial}{\partial t}(\frac{1}{2}\rho_i\alpha_i v^2)+\frac{\partial}{\partial x}\left(\frac{1}{2}\rho_i\alpha_i v^3\right)+\frac{\rho_i\alpha_i v}{\rho}\frac{\partial p}{\partial x}=0 \tag{C.28}$$

We sum up Equation C.27 and Equation C.28 to obtain the final equation:

$$\frac{\partial}{\partial t}\left(\frac{1}{2}\rho_i\alpha_i v^2+e_i\rho_i\alpha_i\right)+\frac{\partial}{\partial x}\left(\frac{1}{2}\rho_i\alpha_i v^3+e_i\rho_i\alpha_i v\right)+\frac{\rho_i\alpha_i v}{\rho}\frac{\partial p}{\partial x}+\alpha_i p\frac{\rho c^2}{\rho_i c_i^2}\frac{\partial v}{\partial x}=\mathcal{Z} \tag{C.29}$$

$$\mathcal{Z}=\theta_i\sum_{j\neq i}H_{ij}(T_j-T_i)+\frac{\rho c^2}{\rho_i c^2}\alpha_i\sum_{k,j>k}H_{kj}(\theta_j-\theta_k)(T_j-T_k)$$

Filling in the formula for E, the final equation becomes:

$$E_i=\rho_i\alpha_i\left(e_i+\frac{1}{2}v^2\right) \tag{C.30}$$

$$\frac{\partial E}{\partial t}+\frac{\partial E v}{\partial x}+\frac{\rho_i\alpha_i v}{\rho}\frac{\partial p}{\partial x}+\alpha_i p\frac{\rho c^2}{\rho_i c_i^2}\frac{\partial v}{\partial x}=\mathcal{Z} \tag{C.31}$$

Which can be rewritten to:

$$\frac{\partial E_l}{\partial t}+\frac{\partial}{\partial x}(E_l v)+v\frac{\rho_l\alpha_l}{\rho_g\alpha_g+\rho_l\alpha_l}\frac{\partial p}{\partial x}+\eta\rho_g\alpha_l c_g^2\frac{\partial v}{\partial x}=\mathcal{Z} \tag{C.32}$$

Rewriting mass conservation into mass advection

We start with the mass conservation equation for both component. This are the top two equations of Equation C.3. This can be rewritten:

$$Y_i = \frac{\rho_i \alpha_i}{\rho} \tag{C.33}$$

$$\frac{\partial (Y_i \rho)}{\partial t} + \frac{\partial (Y_i \rho v)}{\partial x} = 0 \tag{C.34}$$

$$\rho \left(\frac{\partial Y_i}{\partial t} + v \frac{\partial Y_i}{\partial x} \right) + Y_i \left(\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} \right) = 0 \tag{C.35}$$

When summing up top two equations of Equation C.3, it becomes:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0 \tag{C.36}$$

$$\sum_{i=1}^{N} Y_i = 1 \tag{C.37}$$

This can be filled in Equation C.35 and becomes 0. Furthermore, because of Equation C.37, there is only 1 independent advection equation. The two mass conservation equations becomes then:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0 \tag{C.38}$$

$$\frac{\partial Y_i}{\partial t} + v \frac{\partial Y_i}{\partial x} = 0 \tag{C.39}$$

Eigenvalues

With the entropy derivation and Y, it can be rewritten in the following equations:

$$\begin{gathered}
\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho v) = 0 \\
\frac{\partial}{\partial t}(\rho v) + \frac{\partial}{\partial x}(\rho v^2) + \frac{\partial p}{\partial x} = 0 \\
\frac{\partial \boldsymbol{Y}}{\partial t} + v \frac{\partial \boldsymbol{Y}}{\partial x} = 0 \\
\rho_l \alpha_l T_l \left(\frac{\partial s_l}{\partial t} + v \frac{\partial s_l}{\partial t} \right) = 0 \\
\rho_g \alpha_g T_g \left(\frac{\partial s_g}{\partial t} + v \frac{\partial s_g}{\partial t} \right) = 0
\end{gathered} \tag{C.40}$$

It should be noted that all source terms are discarded as the eigenvalues are based indifferent of the source terms. In Equation C.40, the first and second equation can be rewritten to only a variable of v and p in the derivative, and so the material derivative can be deducted. We first start with the second equation:

$$\frac{Dp}{Dt} = \frac{\partial p}{\partial t} + v \frac{\partial p}{\partial x} = \frac{\partial p}{\partial \rho} \frac{\partial \rho}{\partial t} + v \frac{\partial p}{\partial \rho} \frac{\partial \rho}{\partial x} \tag{C.41}$$

Rewriting the continuity equation and filling it in Equation C.41

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho v) = 0, \quad \frac{\partial \rho}{\partial t} = -\rho \frac{\partial v}{\partial x} - v \frac{\partial \rho}{\partial x} \tag{C.42}$$

$$\frac{Dp}{Dt} = \frac{\partial p}{\partial \rho} \left(-\rho \frac{\partial v}{\partial x} - v \frac{\partial \rho}{\partial x} \right) + v \frac{\partial p}{\partial \rho} \frac{\partial \rho}{\partial x} \tag{C.43}$$

$$\frac{Dp}{Dt} = -v \frac{\partial p}{\partial \rho} \frac{\partial \rho}{\partial x} + v \frac{\partial p}{\partial \rho} \frac{\partial \rho}{\partial x} - \rho c^2 \frac{\partial v}{\partial x} \tag{C.44}$$

For the first equation in Equation C.40, the continuity equation is rewritten as in Equation C.42 and filled in the momentum equation.

$$v\frac{\partial \rho}{\partial t}+\rho\frac{\partial v}{\partial t}+2v\rho\frac{\partial v}{\partial x}+v^2\frac{\partial \rho}{\partial x}+\frac{\partial p}{\partial x}=0 \tag{C.45}$$

$$v\left(-\rho\frac{\partial v}{\partial x}-v\frac{\partial \rho}{\partial x}\right)+\rho\frac{\partial v}{\partial t}+2v\rho\frac{\partial v}{\partial x}+v^2\frac{\partial \rho}{\partial x}+\frac{\partial p}{\partial x}=0 \tag{C.46}$$

$$-v\rho\frac{\partial v}{\partial x}-v^2\frac{\partial \rho}{\partial x}+2v\rho\frac{\partial v}{\partial x}+v^2\frac{\partial \rho}{\partial x}+\rho\frac{\partial v}{\partial t}+\frac{\partial p}{\partial x}=0 \tag{C.47}$$

$$\frac{\partial v}{\partial t}+v\frac{\partial v}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}=0 \tag{C.48}$$

Now that the top two equations are rewritten, everything is stated in the material derivative and it is much easier to find the eigenvalues.

$$\begin{gathered}
\frac{\partial v}{\partial t}+v\frac{\partial v}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}=0\\
\frac{\partial p}{\partial t}+v\frac{\partial p}{\partial x}+\rho c^2\frac{\partial v}{\partial x}=0\\
\frac{\partial \boldsymbol{Y}}{\partial t}+v\frac{\partial \boldsymbol{Y}}{\partial x}=0\\
\rho_l\alpha_l T_l\left(\frac{\partial s_l}{\partial t}+v\frac{\partial s_l}{\partial t}\right)=0\\
\rho_g\alpha_g T_g\left(\frac{\partial s_g}{\partial t}+v\frac{\partial s_g}{\partial t}\right)=0
\end{gathered} \tag{C.49}$$

It can be seen that the third, fourth and fifth equation can be rewritten into the material derivative as in Equation B.9. The Jacobian matrix then becomes Equation C.51. The characteristic equation is then given by $(v-\lambda)^3\left((v-\lambda)^2-c^2\right)=0$. The final eigenvalues are stated in Equation C.54.

$$\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{A}(\boldsymbol{u})\frac{\partial \boldsymbol{u}}{\partial x}=0 \tag{C.50}$$

$$\boldsymbol{A}(\boldsymbol{u})=\begin{bmatrix} v & \frac{1}{\rho} & 0 & 0 & 0\\ \rho c^2 & v & 0 & 0 & 0\\ 0 & 0 & v & 0 & 0\\ 0 & 0 & 0 & v & 0\\ 0 & 0 & 0 & 0 & v \end{bmatrix} \tag{C.51}$$

$$\lambda_1(u)=v-c \tag{C.52}$$

$$\lambda_2(u)=\lambda_3(u)=\lambda_4(u)=v \tag{C.53}$$

$$\lambda_5(u)=v+c \tag{C.54}$$

Here c is the speed of sound defined in Equation C.21.

Speed of Sound

The derivation is based on the paper of Flatten et al. [49], section 3.1. Consider the differential:

$$\frac{d(\rho_l \alpha_l)}{\rho_l} + \frac{d(\rho_g \alpha_g)}{\rho_g} \tag{C.55}$$

It can be rewritten into:

$$\frac{d(\rho_l \alpha_l)}{\rho_l} + \frac{d(\rho_g \alpha_g)}{\rho_g} = \rho_l \frac{d(\alpha_l)}{\rho_l} + \rho_g \frac{d(\alpha_g)}{\rho_g} + \alpha_l \frac{d(\rho_l)}{\rho_l} + \alpha_g \frac{d(\rho_g)}{\rho_g} \tag{C.56}$$

$$= d(\alpha_l) + d(\alpha_g) + \alpha_l \frac{d(\rho_l)}{\rho_l} + \alpha_g \frac{d(\rho_g)}{\rho_g} \tag{C.57}$$

$$c_i^2 = \frac{dp}{d\rho_i} \tag{C.58}$$

$$= \alpha_l \frac{dp}{\rho_l c_l^2} + \alpha_g \frac{dp}{\rho_g c_g^2} + O(d\alpha_i) \tag{C.59}$$

Equation C.55 can also be written as:

$$\frac{d(\rho_l \alpha_l)}{\rho_l} + \frac{d(\rho_g \alpha_g)}{\rho_g} = \frac{d(\rho Y_l)}{\rho_l} + \frac{d(\rho Y_g)}{\rho_g} \tag{C.60}$$

$$= \frac{Y_l}{\rho_l} d(\rho) + \frac{Y_g}{\rho_g} d(\rho) + \frac{\rho}{\rho_l} d(Y_l) + \frac{\rho}{\rho_g} d(Y_g) \tag{C.61}$$

$$= \frac{Y_l}{\rho_l} d(\rho) + \frac{Y_g}{\rho_g} d(\rho) + O(dY_i) \tag{C.62}$$

combining both equations, we get:

$$\left(\frac{\alpha_l}{\rho_l c_l^2} + \frac{\alpha_g}{\rho_g c_g^2} \right) dp = \left(\frac{Y_l}{\rho_l} + \frac{Y_g}{\rho_g} \right) d\rho \tag{C.63}$$

$$\frac{dp}{d\rho} = c_{mix}^2 = \frac{\frac{Y_l}{\rho_l} + \frac{Y_g}{\rho_g}}{\frac{\alpha_l}{\rho_l c_l^2} + \frac{\alpha_g}{\rho_g c_g^2}} = \frac{\frac{1}{\rho}}{\frac{\alpha_l}{\rho_l c_l^2} + \frac{\alpha_g}{\rho_g c_g^2}} \tag{C.64}$$

$$c_{mix}^2 = \left[\rho \left(\frac{\alpha_g}{\rho_g c_g^2} + \frac{\alpha_l}{\rho_l c_l^2} \right) \right]^{-1} \tag{C.65}$$

D

HLLC scheme

The method is based on the description done by Toro, described in Chapter 10, [17]. It starts by looking first at the integral form of the conversation laws, as seen in Equation D.1

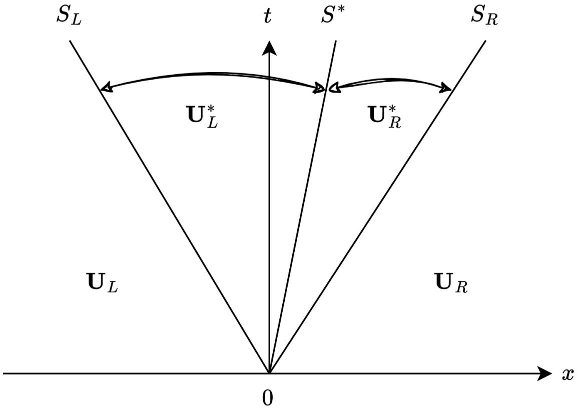


Figure D.1: HLLC regions defined by the wave speed S_L, S^*, S_R

$$
\begin{aligned}
&\boldsymbol{U}_t + \boldsymbol{F(U)}_x = 0,\\
&\boldsymbol{U}(x,0) = \begin{cases} U_L if x < 0, \\ U_R if x > 0 \end{cases}\\
&\int_{x_L}^{x_R} U(x,T)dx = \int_{x_L}^{x_R} U(x,0)dx + \int_0^T F(U(x_L,t)dt - \int_0^T F(U(x_R,t)dt
\end{aligned}
\tag{D.1}
$$

When evaluating, it gives:

$$
\int_{x_L}^{x_R} U(x,T)dx = x_R U_R - x_L U_L + T(F_L - F_R)
\tag{D.2}
$$

This is called the *consistency condition.* Now, the integral is split into three different integrals:

$$
\begin{aligned}
&\int_{x_L}^{x_R} U(x,T)dx = \int_{x_L}^{TS_L} U(x,T)dx + \int_{TS_L}^{TS_R} U(x,T)dx + \int_{TS_R}^{x_R} U(x,T)dx\\
&\int_{x_L}^{x_R} U(x,T)dx = \int_{TS_L}^{TS_R} U(x,T)dx + (TS_L - x_L)U_L + (x_R - TS_R)U_R
\end{aligned}
\tag{D.3}
$$

When substituting Equation D.1 into Equation D.3, the following relationship is obtained:

$$x_R U_R - x_L U_L + T(F_L - F_R) = \int_{TS_L}^{TS_R} U(x,T)dx + (TS_L - x_L)U_L + (x_R - TS_R)U_R \tag{D.4}$$

$$\int_{TS_L}^{TS_R} U(x,T)dx = T(S_R U_R - S_L U_L + F_L - F_R) \tag{D.5}$$

Now when dividing Equation D.5 by the length, $T(S_R - S_L)$, the integral average is given. This is because $T(S_R - S_L)$ is the width of the solution between the slowest and fastest signals at the time. If the speeds, S_L and S_R are known, the whole solution is known and thus the middle state is known.

$$\frac{1}{T(S_R - S_L)}\int_{TS_L}^{TS_R} U(x,T)dx = \frac{S_R U_R - S_L U_L + F_L - F_R}{S_R - S_L} \tag{D.6}$$

$$U^{HLL} = \frac{S_R U_R - S_L U_L + F_L - F_R}{S_R - S_L} \tag{D.7}$$

Now for the *HLLC* method, there are two middle states. This is simply the integral from the left wave till the middle wave and from the middle wave until the right wave:

$$\frac{1}{T(S_R - S_L)}\int_{TS_L}^{TS_R} U(x,T)dx = \frac{1}{T(S_R - S_L)}\int_{TS_L}^{TS^*} U(x,T)dx + \frac{1}{T(S_R - S_L)}\int_{TS^*}^{TS_R} U(x,T)dx \tag{D.8}$$

$$\tag{D.9}$$

So each state has it's own values. The integral averages are defined as follow:

$$\left.\begin{aligned} U_L^* &= \frac{1}{T(S^* - S_L)}\int_{TS_L}^{TS^*} U(x,T)dx \\ U_R^* &= \frac{1}{T(S_R - S^*)}\int_{TS^*}^{TS_R} U(x,T)dx \end{aligned}\right\} \tag{D.10}$$

note the difference between $1/[T(S^* - S_L)]$ and $1/[T(S_R - S_L)]$ for Equation D.8 and Equation D.10. The same for U_R^*. Now substituting Equation D.7, Equation D.8 and Equation D.10, the following is obtained:

$$\left(\frac{S^* - S_L}{S_R - S_L}\right)U_L^* + \left(\frac{S_R - S^*}{S_R - S_L}\right)U_R^* = U^{hll} \tag{D.11}$$

The Riemann solver with the matching flux algorithm is given as:

$$U(x,t) = \begin{cases} \boldsymbol{U}_L, & \text{if } \frac{x}{t} < S_L, \\ \boldsymbol{U}_L^*, & \text{if } S_L \le \frac{x}{t} < S_C, \\ \boldsymbol{U}_R^*, & \text{if } S_C \le \frac{x}{t} < S_R, \\ \boldsymbol{U}_R, & \text{if } \frac{x}{t} \ge S_R, \end{cases} \tag{D.12}$$

$$\boldsymbol{F}_{i+1/2} = \begin{cases} \boldsymbol{F}_L, & \text{if } 0 < S_L, \\ \boldsymbol{F}_L^*, & \text{if } S_L \le 0 < S_C, \\ \boldsymbol{F}_R^*, & \text{if } S_C \le 0 < S_R, \\ \boldsymbol{F}_R, & \text{if } 0 \ge S_R, \end{cases} \tag{D.13}$$

With these two systems, there are four unknowns. Namely $\boldsymbol{U}_L^*, \boldsymbol{U}_R^*, \boldsymbol{F}_L^*, \boldsymbol{F}^* RL$. Now applying the Rankine-Hugoniont Conditions three new equations can be stated. These are:

$$U_L \cdot S_L - F_L = U_L^* \cdot S_L - F_L^* \tag{D.14}$$

$$U_L^* \cdot S^* - F_L^* = U_R^* \cdot S^* - F_R^* \tag{D.15}$$

$$U_R^* \cdot S_R - F_R^* = U_R \cdot S_R - F_R \tag{D.16}$$

This gives three equations. For the fourth equation, it is needed to look specifically at the set of equations of the model, in this case for the DF_3 model. Equation D.14 and Equation D.16 will be used to define the fourth equation, which is an equation for solving the pressure. First, fill in the mass equation in Equation D.14, where the flux is Equation D.17.

$$F(U) = v \cdot U + p \cdot D, \quad D = [0,1,v]^T \tag{D.17}$$

$$\rho_L S_L - \rho_L v_L = \rho_L^* S_L - \rho_L^* v_L^* \tag{D.18}$$

$$\rho_L(S_L - v_L) = \rho_L^*(S_L - v_L^*) \tag{D.19}$$

Then, filling in the momentum equation in Equation D.14, the following is obtained:

$$\rho_L v_L S_L - \rho_L v_L^2 - p_L = \rho_L^* v_L^* S_L - \rho_L^*(v^2)_L^* - p_L^* \tag{D.20}$$

$$\rho_L v_L(S_L - v_L) - p_L = \rho_L^*(S_L - v_L^*) v_L^* - p_L^* \tag{D.21}$$

$$\rho_L v_L(S_L - v_L) - p_L = \rho_L(S_L - v_L)S^* - p_L^* \tag{D.22}$$

$$p_L^* = p_L + \rho_L(S_L - v_L)(S^* - v_L) \tag{D.23}$$

Note that the assumption can be made that $v_L^* = v_R^* = S^*$, so the middle wave. The same can be done for the right side, which gives:

$$p_R^* = p_R + \rho_R(S_R - v_R)(S^* - v_R) \tag{D.24}$$

Because the states U_R^* and U_L^* are separated by an contact discontinuity and not a shock, it can be said that the pressures on both side are the same. With this insight, it is possible to get the following formula for S^*, assuming $p_R^* = p_L^*$:

$$p_L + \rho_L(S_L - v_L)(S^* - v_L) = p_R + \rho_R(S_R - v_R)(S^* - v_R) \tag{D.25}$$

$$p_L + \rho_L S^*(S_L - v_L) - \rho_L v_L(S_L - v_L) = p_R + \rho_R S^*(S_R - v_R) - \rho_R v_R(S_R - v_R) \tag{D.26}$$

$$\rho_L S^*(S_L - v_L) - \rho_R S^*(S_R - v_R) = p_R - p_L + \rho_L v_L(S_L - v_L) - \rho_R v_R(S_R - v_R) \tag{D.27}$$

$$S^*\Big(\rho_L(S_L - v_L) - \rho_R(S_R - v_R)\Big) = p_R - p_L + \rho_L v_L(S_L - v_L) - \rho_R v_R(S_R - v_R) \tag{D.28}$$

$$S^* = \frac{p_R - p_L + \rho_L v_L(S_L - v_L) - \rho_R v_R(S_R - v_R)}{\rho_L(S_L - v_L) - \rho_R(S_R - v_R)} \tag{D.29}$$

Lastly, the state vector for U_K^{HLLC} needs to be derived. This is done by rewriting Equation D.14 and Equation D.16 into:

$$F_K^* = F_K + S_K(U_K^* - U_K) \tag{D.30}$$

Where K is L or R. Now the mass, momentum and energy equation are filled in. This becomes the following for the mass equation:

$$S_K(\rho_K^* - \rho_K) = \rho_K^* v_K^* - \rho_K v_K \tag{D.31}$$

$$\rho_K^*(S_K - v_K^*) = \rho_K(S_K - v_K) \tag{D.32}$$

$$\rho_K^* = \rho_K \frac{(S_K - v_K)}{(S_K - v_K^*}) \tag{D.33}$$

$$\rho_K^* = \rho_K \frac{(S_K - v_K)}{(S_K - S^*)} \tag{D.34}$$

Below is for the momentum equation (note that Equation D.24 is used):

$$S_K(\rho_K^* v_K^* - \rho_K v_K) = \rho_K^*(v^2)_K^* - \rho_K v_K^2 + p_K^* - p_K \tag{D.35}$$

$$\rho_K^* v_K^*(S_K - v_K^*) = \rho_K v_K(S_K - v_K) + p_K - p_K + \rho_K(S_K - v_K)(S^* - v_K) \tag{D.36}$$

$$\rho_K^* v_K^*(S_K - S^*) = \rho_K v_K(S_K - v_K) - \rho_K v_K(S_K - v_K) + \rho_K S^*(S_K - v_K^*) \tag{D.37}$$

$$\rho_K^* v_K^* = \rho_K S^* \frac{(S_K - v_K)}{(S_K - S^*)} \tag{D.38}$$

And lastly for the energy equation (again, Equation D.23 is used):

$$S_K(E_K^* - E_K) = E_K^* v_K^* - E_K v_K + p_K^* v_K^* - p_K v_K \tag{D.39}$$

$$E_K^*(S_K - v_K^*) = E_K(S_K - v_K) + p_K^* v_K^* - p_K v_K \tag{D.40}$$

$$E_K^*(S_K - v_K^*) = E_K(S_K - v_K) + \Big(p_K + \rho_K(S_K - v_K)(S^* - v_K)\Big) v_K^* - p_K v_K \tag{D.41}$$

$$E_K^*(S_K - v_K^*) = E_K(S_K - v_K) + \rho_K v_K^*(S_K - v_K)(S^* - v_K) + p_K(v_K^* - v_K) \tag{D.42}$$

$$E_K^* = E_K \frac{(S_K - v_K)}{(S_K - S^*)} + \rho_K S^*(S^* - v_K)\frac{(S_K - v_K)}{(S_K - S^*)} + p_k \frac{(S^* - v_k)}{(S_K - S^*)} \tag{D.43}$$

The state vector therefore is:

$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right) \cdot \begin{pmatrix} \rho_K \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^* \rho_K + \frac{p_K}{(S_K - v_K)}\right) \end{pmatrix} \tag{D.44}$$

With S_L and S_R being

$$S_L = \min(u_L - c_L, u_R - c_R), \tag{D.45}$$

$$S_R = \max(u_L + c_L, u_R + c_R) \tag{D.46}$$

$DF4$ model

The difference in the $DF4$ model is the two mass conservation equations. Again, to find the state vector, the derivation is started from Equation D.30. This becomes as follow:

$$S_K(\alpha_K^* \rho_K^* - \alpha_K \rho_K) = \alpha_K^* \rho_K^* v_K^* - \alpha_K \rho_K v_K \tag{D.47}$$

$$\alpha_K^* \rho_K^*(S_K - v_K^*) = \alpha_K \rho_K (S_K - v_K) \tag{D.48}$$

$$\alpha_K^* \rho_K^* = \alpha_K \rho_K \frac{(S_K - v_K)}{(S_K - S^*)} \tag{D.49}$$

The state vector then becomes:

$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right) \cdot \begin{pmatrix} \alpha_{g,K} \rho_{g,K} \\ \alpha_{l,K} \rho_{l,K} \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^* \rho_K + \frac{p_K}{(S_K - v_K)}\right) \end{pmatrix} \tag{D.50}$$

$TF5$ model

Compared with the $DF4$ model, the first four entries of the state vector of the $TF5$ model will be the same. Only an additional entry is added accounting for the extra energy equation. Because the extra energy equation is for the gaseous phase, the derivation will also be done for the gaseous phase, while it being identical for the liquid phase. The derivation starts again from Equation D.30, and the derivation is as follow:

$$S_K(E_{K,g}^* - E_{K,g}) = E_{K,g}^* v_K^* - E_{K,g} v_K + \alpha_{K,g}^* p_K^* v_K^* - \alpha_{K,g} p_K v_K \tag{D.51}$$

$$E_{K,g}^*(S_K - v_K^*) = E_{K,g}(S_K - v_K) + \alpha_{K,g} p_K^* v_K^* - \alpha_{K,g} p_K v_K \tag{D.52}$$

$$E_{K,g}^*(S_K - v_K^*) = E_{K,g}(S_K - v_K) + \Big(p_K + \rho_K(S_K - v_K)(S^* - v_K)\Big)\alpha_{K,g} v_K^* - \alpha_{K,g} p_K v_K \tag{D.53}$$

$$E_{K,g}^*(S_K - v_K^*) = E_{K,g}(S_K - v_K) + \alpha_{K,g} \rho_K v_K^*(S_K - v_K)(S^* - v_K) + \alpha_{K,g} p_K(v_K^* - v_K) \tag{D.54}$$

$$E_{K,g}^* = E_{K,g} \frac{(S_K - v_K)}{(S_K - S^*)} + \alpha_{K,g} \rho_K S^*(S^* - v_K)\frac{(S_K - v_K)}{(S_K - S^*)} + \alpha_{K,g} p_k \frac{(S^* - v_k)}{(S_K - S^*)} \tag{D.55}$$

Note that the assumption of $\alpha_{K,g}^* = \alpha_{K,g}$ is made. The state vector then becomes:

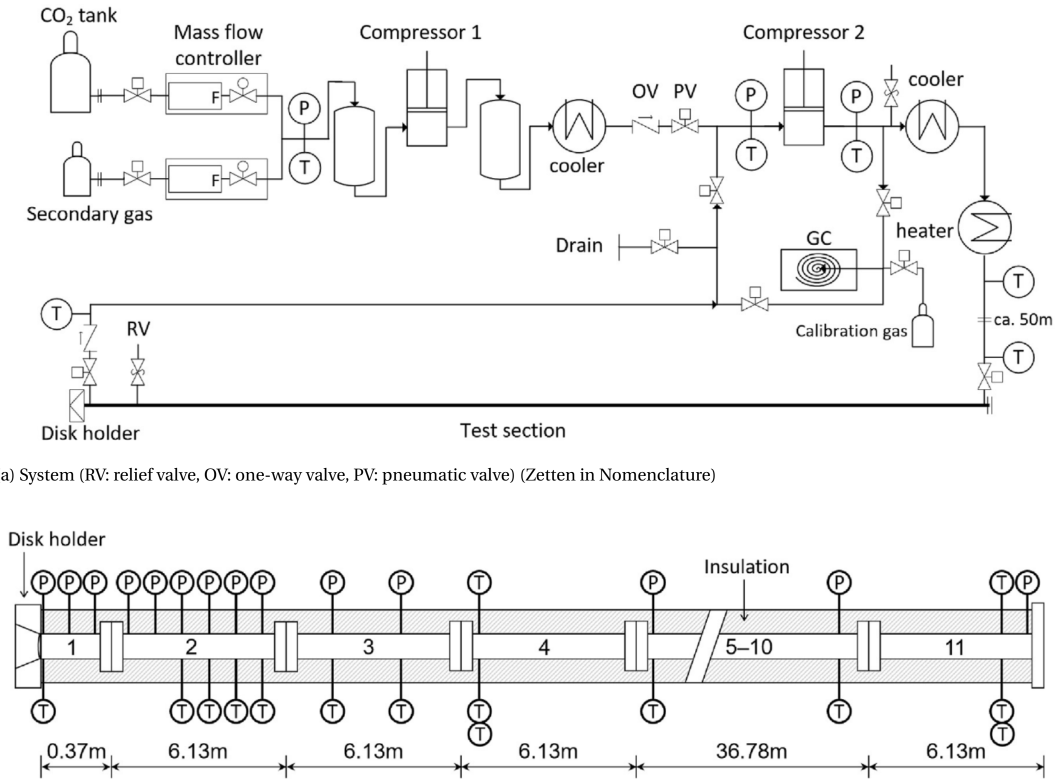
$$\boldsymbol{U}_K^{HLLC} = \left(\frac{S_K - v_K}{S_K - S^*}\right) \cdot \begin{pmatrix} \alpha_{g,K}\rho_{g,K} \\ \alpha_{l,K}\rho_{l,K} \\ \rho_K S^* \\ E_K + (S^* - v_K)\left(S^* \rho_K + \frac{p_K}{(S_K - v_K)}\right) \\ E_{g,K} + (S^* - v_K)\left(S^* (\alpha_{g,K}\rho_{g_K}) + \frac{\alpha_{g,K} p_K}{(S_K - v_K)}\right) \end{pmatrix} \tag{D.56}$$

E
Experiment data

16 fast-response pressure transducers and 23 type E thermocouples are flush-mounted to the test section's inner surface in order to record temperature, pressure and fluctuations during depressurisation. The Kulite CTL-190(M) kind of pressure sensors are used. The bandwidth of these high-frequency pressure sensors is up to 200 [kHz]. Eleven of the thermocouples are positioned axially, matching where a pressure sensor is located. The remaining 12 thermocouples are mounted at the top, bottom and side of the pipe at four positions. The thermocouples have a measurement speed of 1 [ms]. With this, it is possible to measure different temperatures in the cross-section. Hereby, it can be determined if the flow is stratified or dispersed. It has been found that the temperature readings have an uncertainty of ±0.22°C and the pressure measurements have an uncertainty of about 60 [kPa] with a 95% confidence interval, as noted by Munkejord et al. [21]. All locations of the sensors are indicated in Table E.2. A schematic overview, drawn by Austegard & Deng [24] [21] is seen in Figure E.1

At the rear end of the test section, the CO_2 or CO_2-mixture enters. For combination testing, circulation of the fluid is made possible by a return pipeline close to the open end of the test section, which guarantees a consistent temperature and composition. To measure the composition of the combination, a micro gas chromatograph is attached. The test portion and the second compressor are followed by pressure-relief valves that open at a pressure of 20 [MPa]. The system is emptied via the drain lines following every test. Before every test, a vacuum pump is set up on the same line to clear the system.

The following text is a citation by Log et al. [22] on the procedure of the experiments:
"The experimental procedure is as follows. First, the rupture disk is installed and the system is evacuated. Then the test section is filled with CO_2 and pressurized. When the pressure reaches about 70% of the desired value, the fluid is circulated to achieve a uniform temperature along the test section. The fluid temperature is controlled using heating elements wrapped around the test section. The pressure and temperature are then increased at a controlled rate by alternating filling and circulation of CO_2 until the disk ruptures. Upon disk rupture, the inlet valves at the closed end of the pipe, and outlet valve at the open end of the pipe are automatically closed to stop the circulation/filling. The heating cables are also automatically turned off at this point. The released CO_2 is vented through an exhaust pipe. The exhaust pipe is designed with a large enough flow area that it will not disturb the flow from the pipe."



(a) System (RV: relief valve, OV: one-way valve, PV: pneumatic valve) (Zetten in Nomenclature)

(b) Test section (dimensions are not to scale; pipe no. 5–10 and corresponding sensors are omitted).

Figure E.1: Schematic of the ECCSEL depressurization facility drawn by Austegard & Deng [21].

Table E.1: Specifications of the pipeline, obtained from Munkejord et al. [21].

Parameter	Value
Pipe steel density	8000 [kg/m^3]
Pipe steel thermal conductivity	15 [$W/(m\cdot K)$]
Pipe steel specific heat	500 [$J/(kg\cdot K)$]
Pipe inner diameter	40.8 [mm]
Pipe outer diameter	48.3 [mm]
Pipe length	61.668 [m] at 25 [°C]
Pipe mean roughness	0.3 [μm]
Insulation layer density	75 [kg/m^3]
Insulation layer thermal conductivity	0.032 [$W/(m\cdot K)$]
Insulation layer specific heat	840 [$J/(kg\cdot K)$]
Insulation thickness	60 [mm]

Table E.2: Locations of pressure and temperature sensors at 25 °C, obtained from Munkejord et al. [21].

Distance from open end [m]	Pressure sensor	Temperature sensor (side)	Temperature sensor (bottom, side, top)
0.080	PT201	TT201	
0.180	PT202		
0.280	PT203		
0.484	PT204		
0.800	PT205		
1.599	PT206	TT206	
3.198	PT207	TT207	
4.798	PT208	TT208	
6.397	PT209	TT209	
7.996	PT210	TT210	
9.595	PT211	TT211	
15.292			TT241, TT242, TT243
19.990	PT212	TT212	
29.986	PT213	TT213	
30.686			TT251, TT252, TT253
39.984	PT214	TT214	
46.085			TT261, TT262, TT263
49.982	PT215	TT215	
61.280			TT271, TT272, TT273
61.479	PT216		

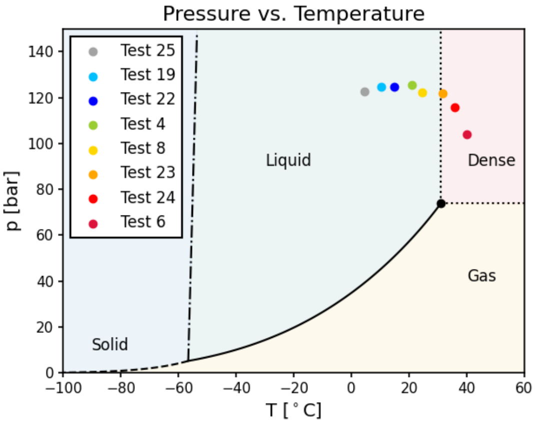
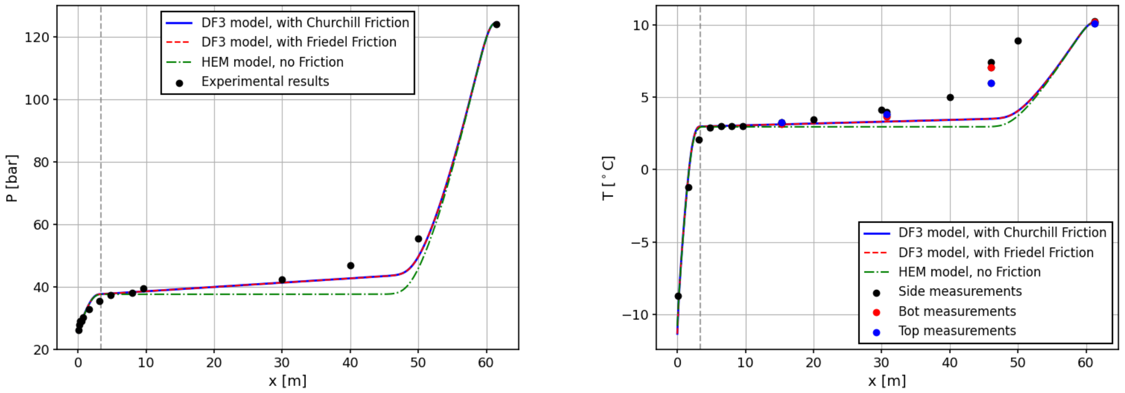


Figure E.2: Initial conditions of the test-cases on the PT diagram of CO_2.

F

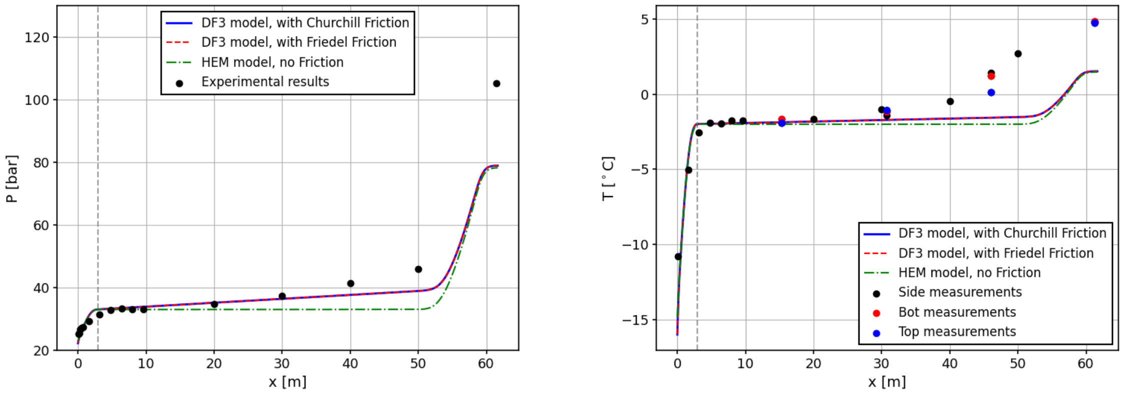
Additional figures with results



(a) Pressure of test 19 at t = 0.1 [*s*]

(b) Temperature of test 19 at t = 0.1 [*s*]

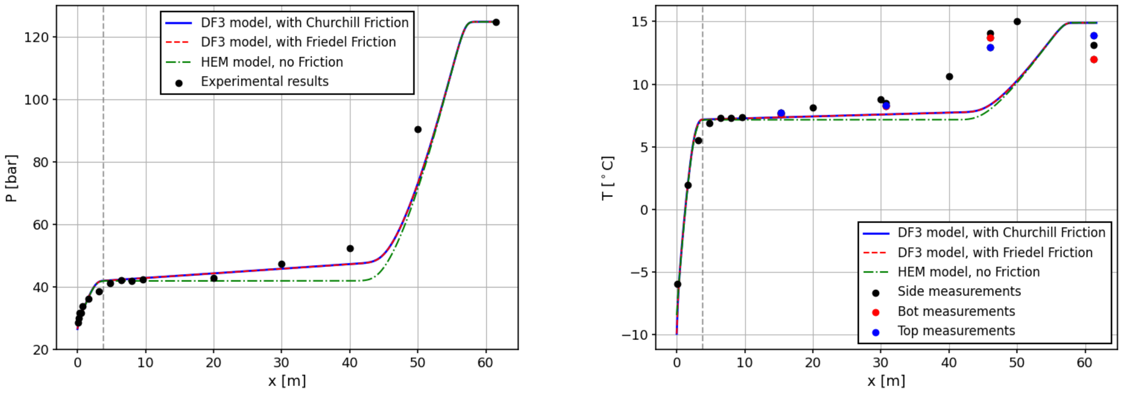
Figure F.2: Pressure & temperature results of test 19 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 25 at t = 0.1 [*s*]

(b) Temperature of test 25 at t = 0.1 [*s*]

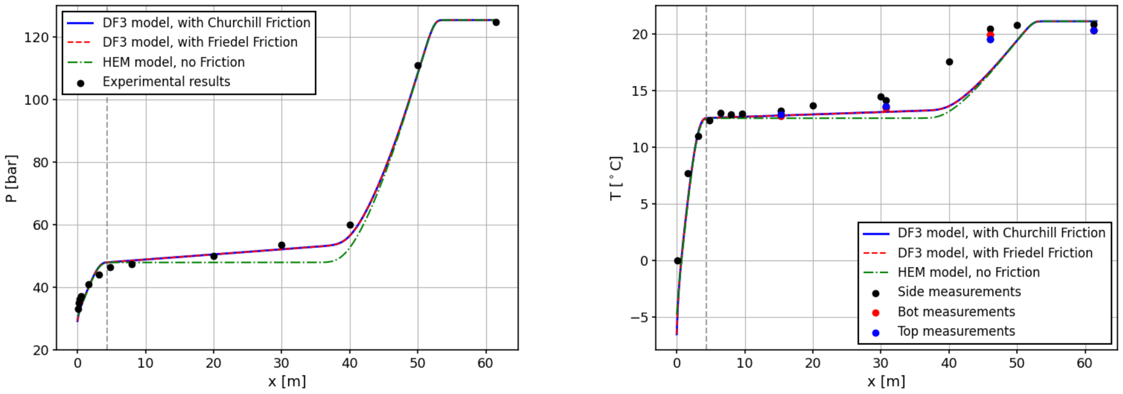
Figure F.1: Pressure & temperature results of test 25 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 22 at t = 0.1 [s]

(b) Temperature of test 22 at t = 0.1 [s]

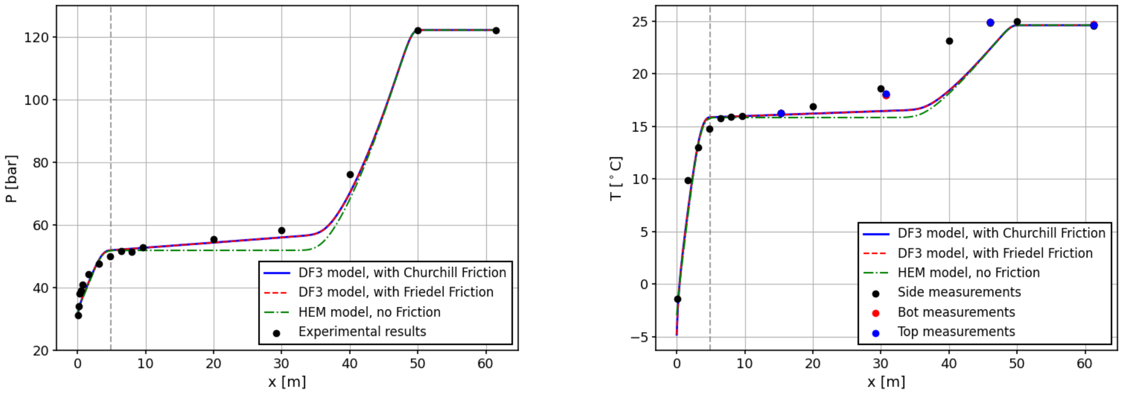
Figure F.3: Pressure & temperature results of test 22 t = 0.1[s] for the $DF3$ model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 4 at t = 0.1 [s]

(b) Temperature of test 4 at t = 0.1 [s]

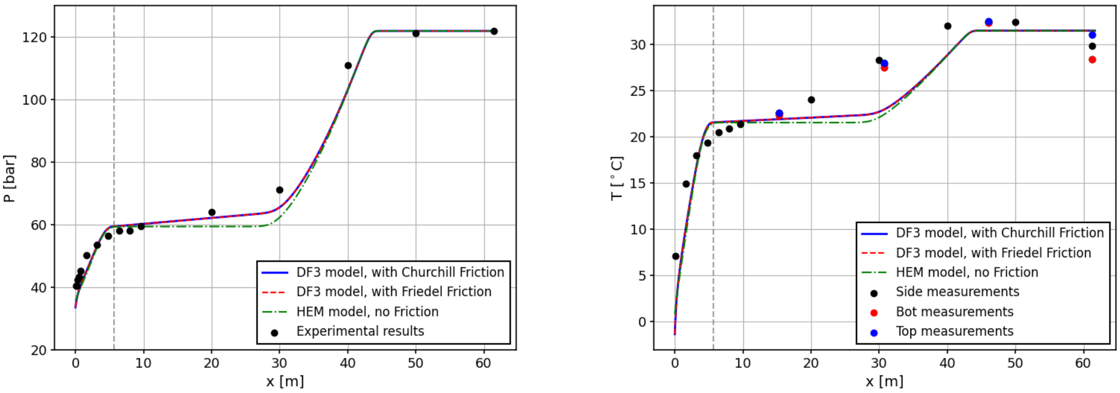
Figure F.4: Pressure & temperature results of test 4 t = 0.1[s] for the $DF3$ model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 8 at t = 0.1 [s]

(b) Temperature of test 8 at t = 0.1 [s]

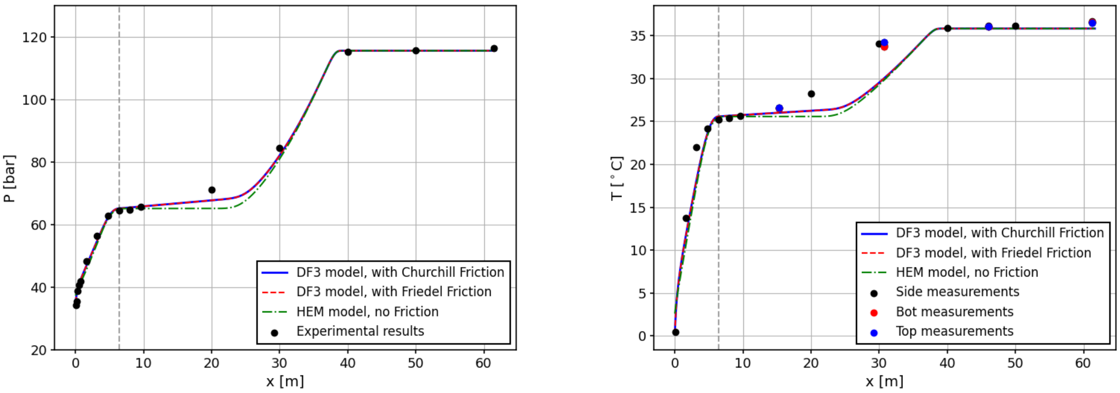
Figure F.5: Pressure & temperature results of test 8 t = 0.1[s] for the $DF3$ model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 23 at t = 0.1 [*s*]

(b) Temperature of test 23 at t = 0.1 [*s*]

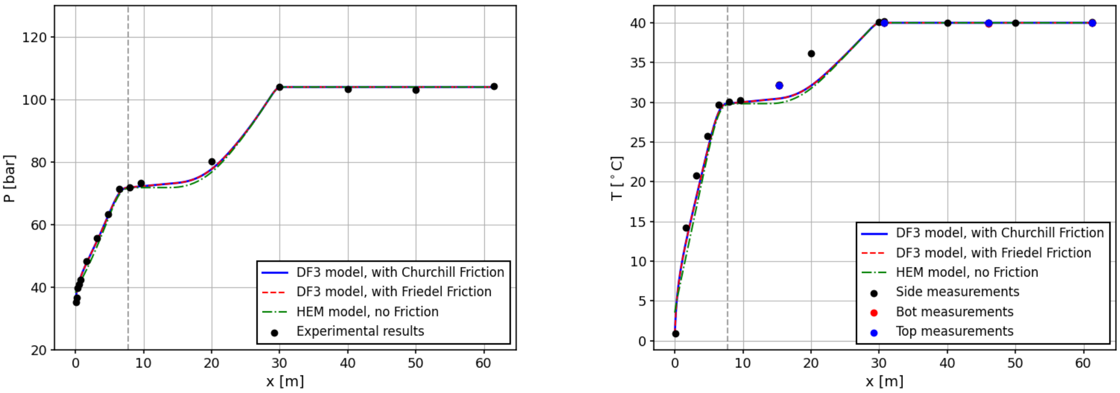
Figure F.6: Pressure & temperature results of test 23 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 24 at t = 0.1 [*s*]

(b) Temperature of test 24 at t = 0.1 [*s*]

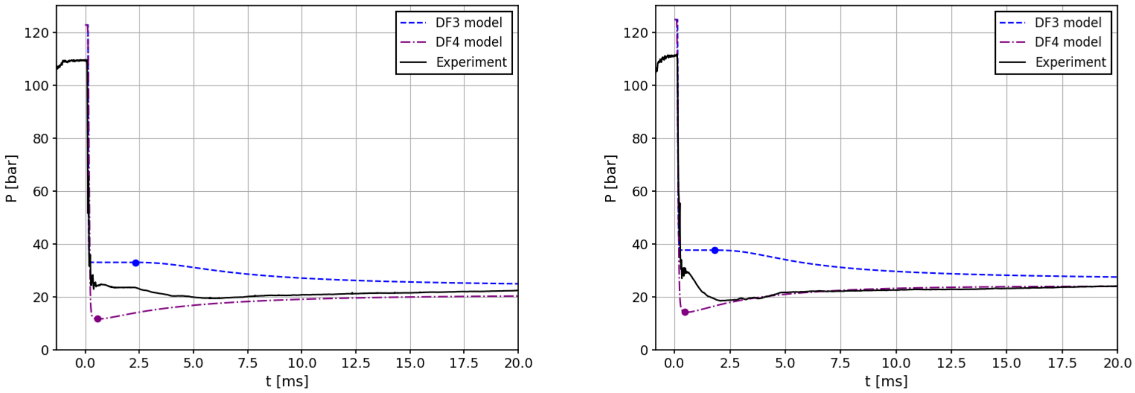
Figure F.7: Pressure & temperature results of test 24 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 6 at t = 0.1 [*s*]

(b) Temperature of test 6 at t = 0.1 [*s*]

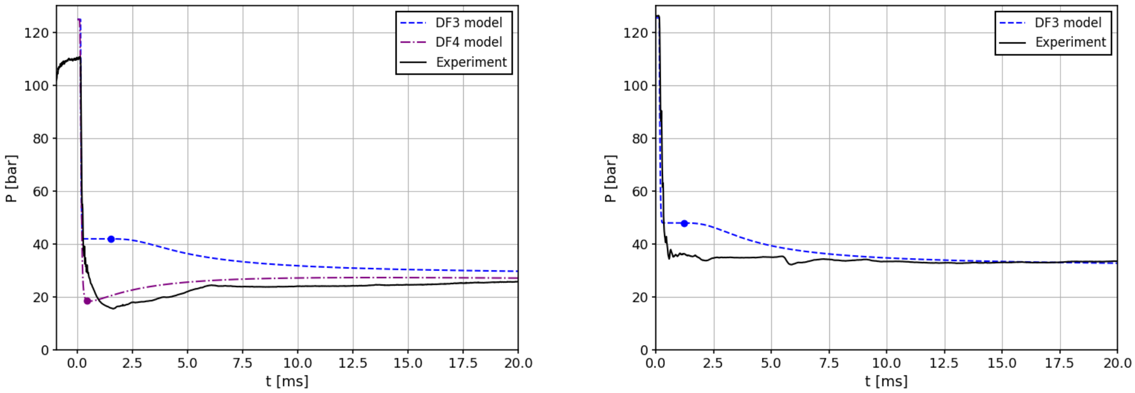
Figure F.8: Pressure & temperature results of test 6 t = 0.1[*s*] for the *DF*3 model. All simulations are done with a CFL of 0.95 and 400 cells.



(a) Pressure of test 25 at x = 0.08 $[m]$

(b) Pressure of test 19 at x = 0.08 $[m]$

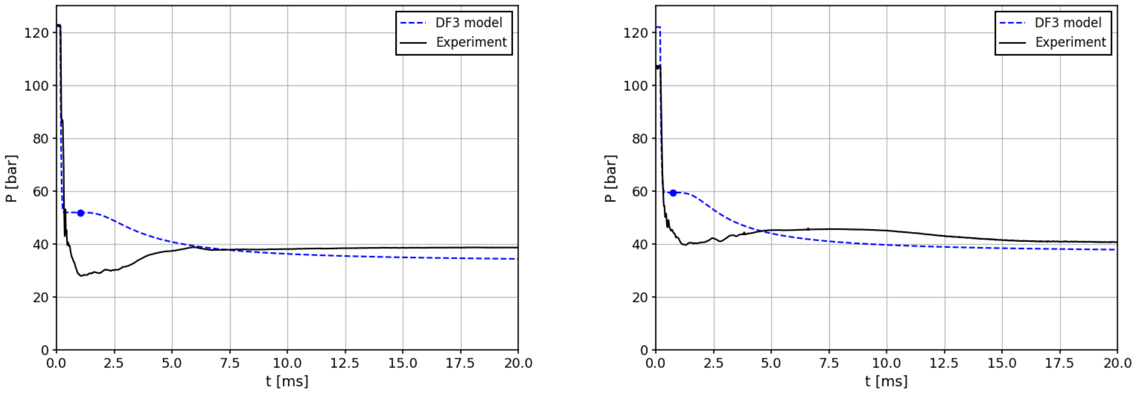
Figure F.9: Pressure evolution over time for test 25 and test 19, done at x = 0.08 $[m]$. A CFL of 0.5 and 4000 cells are used.



(a) Pressure of test 22 at x = 0.08 $[m]$

(b) Pressure of test 4 at x = 0.08 $[m]$

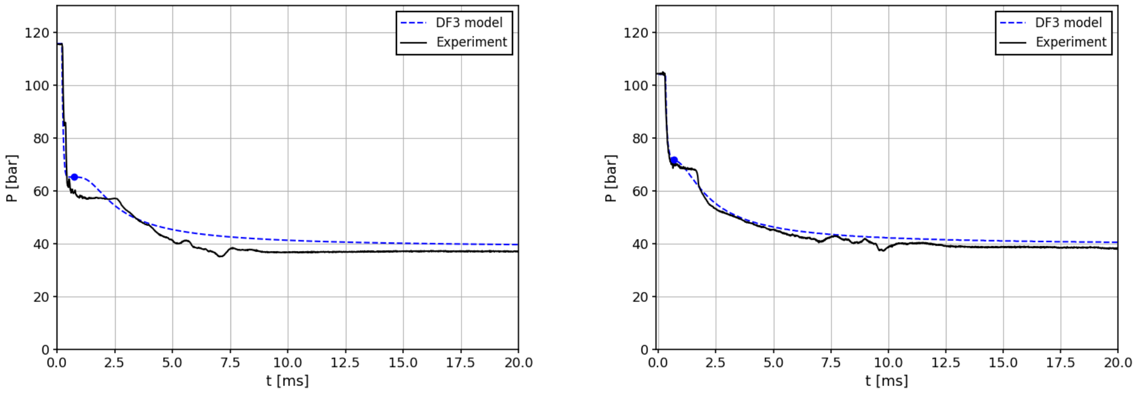
Figure F.10: Pressure evolution over time for test 22 and test 4, done at x = 0.08 $[m]$. A CFL of 0.5 and 4000 cells are used.



(a) Pressure of test 8 at x = 0.08 $[m]$

(b) Pressure of test 23 at x = 0.08 $[m]$

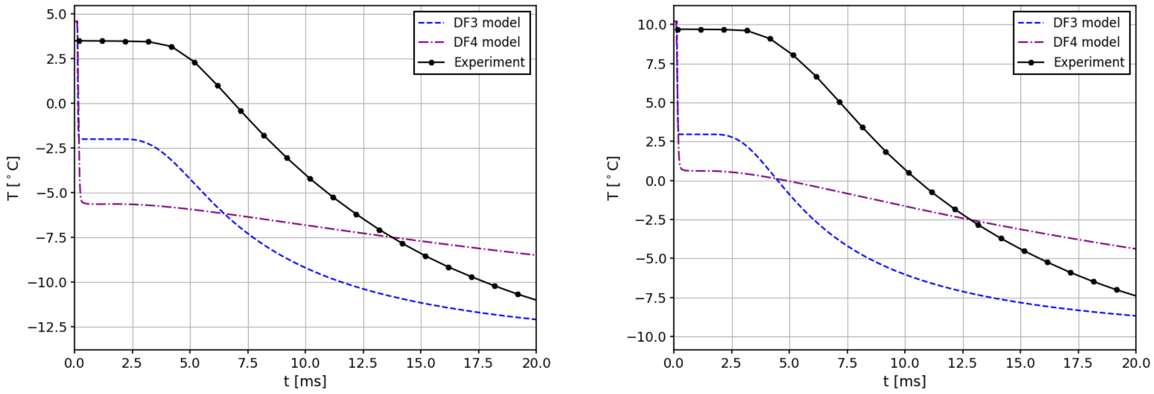
Figure F.11: Pressure evolution over time for test 8 and test 23, done at x = 0.08 $[m]$. A CFL of 0.5 and 4000 cells are used.



(a) Pressure of test 24 at x = 0.08 [*m*]

(b) Pressure of test 6 at x = 0.08 [*m*]

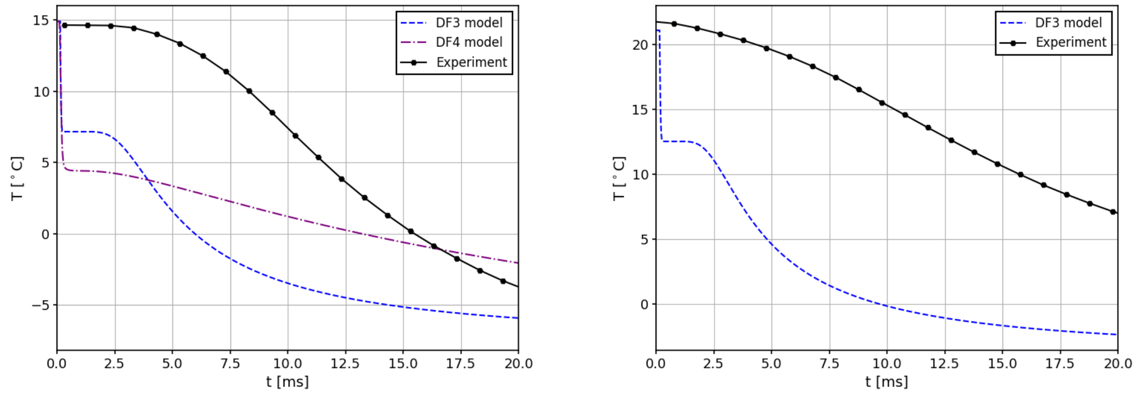
Figure F.12: Pressure evolution over time for test 24 and test 6, done at x = 0.08 [*m*]. A CFL of 0.5 and 4000 cells are used.



(a) Temperature of test 25 at x = 0.08 [*m*]

(b) Temperature of test 19 at x = 0.08 [*m*]

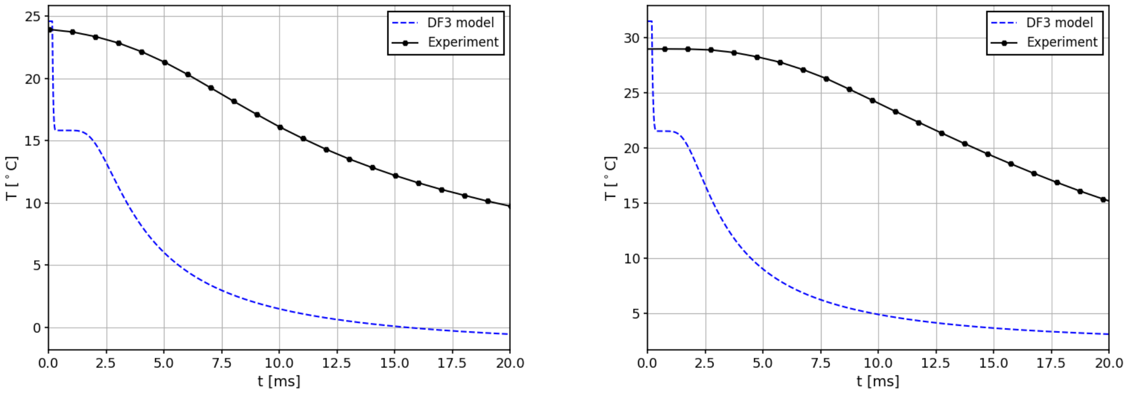
Figure F.13: Temperature evolution over time for test 25 and test 19, done at x = 0.08 [*m*]. A CFL of 0.5 and 4000 cells are used.



(a) Temperature of test 22 at x = 0.08 [*m*]

(b) Temperature of test 4 at x = 0.08 [*m*]

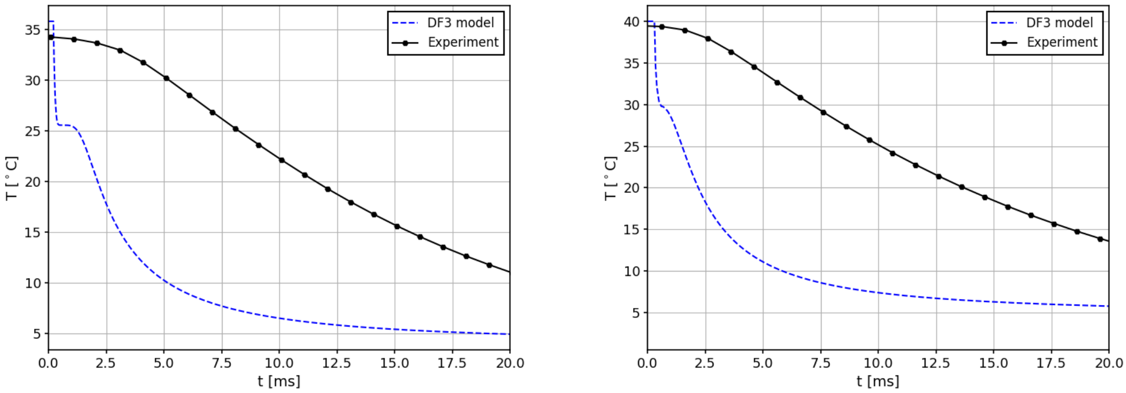
Figure F.14: Temperature evolution over time for test 22 and test 4, done at x = 0.08 [*m*]. A CFL of 0.5 and 4000 cells are used.



(a) Temperature of test 8 at x = 0.08 [m]

(b) Temperature of test 23 at x = 0.08 [m]

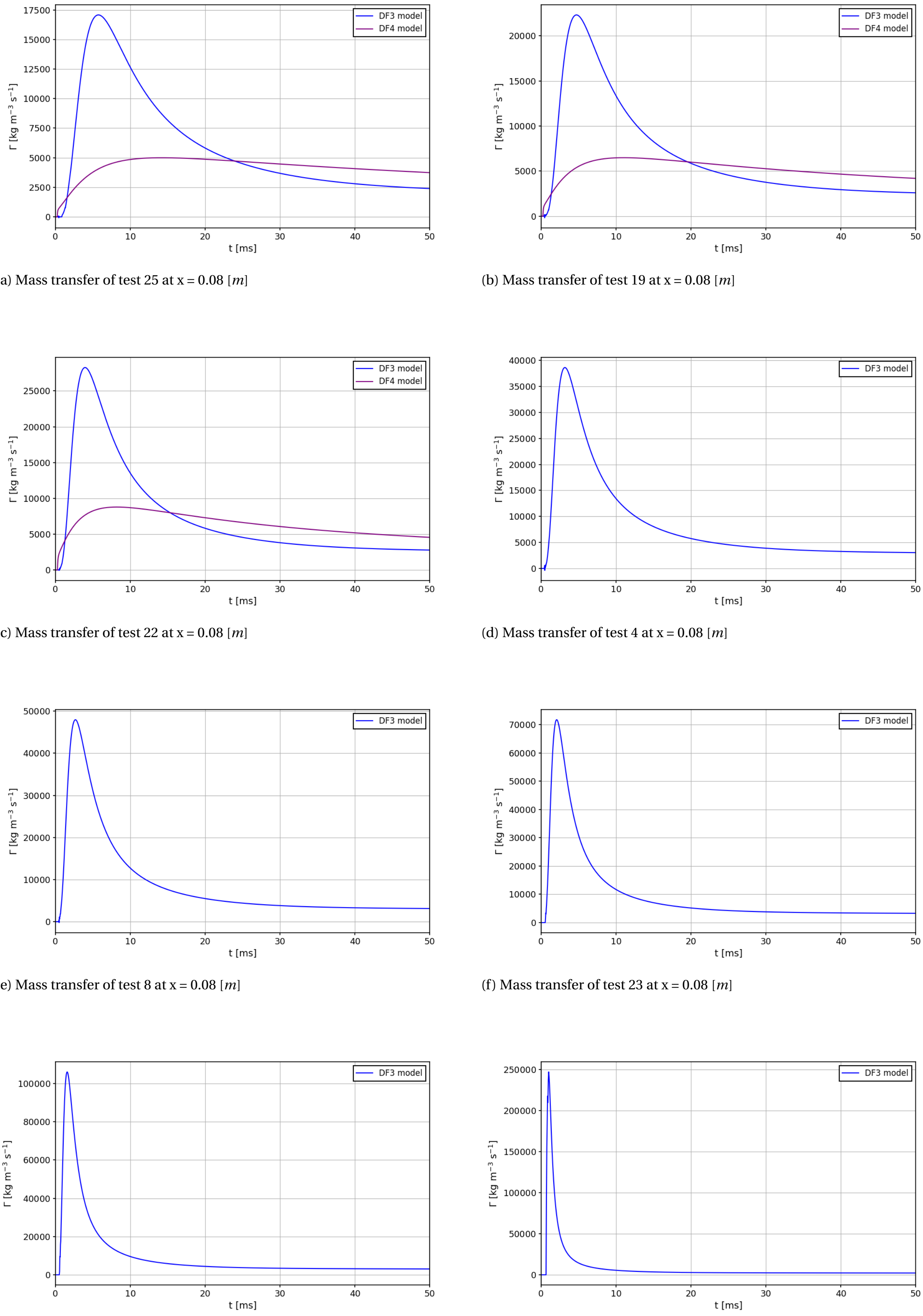
Figure F.15: Temperature evolution over time for test 8 and test 23, done at x = 0.08 [m]. A CFL of 0.5 and 4000 cells are used.



(a) Temperature of test 24 at x = 0.08 [m]

(b) Temperature of test 6 at x = 0.08 [m]

Figure F.16: Temperature evolution over time for test 24 and test 6, done at x = 0.08 [m]. A CFL of 0.5 and 4000 cells are used.



(a) Mass transfer of test 25 at x = 0.08 [m]

(b) Mass transfer of test 19 at x = 0.08 [m]

(c) Mass transfer of test 22 at x = 0.08 [m]

(d) Mass transfer of test 4 at x = 0.08 [m]

(e) Mass transfer of test 8 at x = 0.08 [m]

(f) Mass transfer of test 23 at x = 0.08 [m]

(g) Mass transfer of test 24 at x = 0.08 [m]

(h) Mass transfer of test 6 at x = 0.08 [m]

Figure F.17: mass transfer results of test all tests at x = 0.08 [m]. A CFL of 0.5 and 4000 cells are used.