Modeling and Order Reduction for Hydraulics Simulation in Managed Pressure Drilling

Mohammad Hossein Abbasi

Modeling and Order Reduction for Hydraulics Simulation in Managed Pressure Drilling

Mohammad Hossein Abbasi

The work described in this thesis was carried out at the Eindhoven University of Technology in the "HYDRA" project, which has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 675731.

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-5154-5

Printed by: www.prijsprinter.nl, The Netherlands

Cover design by Mohammad Hossein Abbasi, based on pictures from www.freepik.com

Modeling and Order Reduction for Hydraulics Simulation in Managed Pressure Drilling

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus prof.dr.ir. F.P.T. Baaijens, voor een
commissie aangewezen door het College voor
Promoties, in het openbaar te verdedigen
op dinsdag 10 November 2020 om 13:30 uur

door

Mohammad Hossein Abbasi

geboren te Eilam, Iran

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter:	prof.dr. J.J. Lukkien
1^{e} promotor:	prof.dr. W.H.A. Schilders
2^{e} promotor:	prof.dr.ir. N. van de Wouw
copromotor:	dr. L. Iapichino
leden:	prof.dr. G. Rozza (SISSA International School for Advanced Studies, Italy)
	prof.dr.ir. B. Koren
	prof.dr. B. Maschke (Université de Lyon, France)
	dr. G.O. Kaasa (Kelda Drilling Controls, Norway)

Het onderzoek dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Contents

List of Abbreviations

BHA ..... Bottom-Hole Assembly

BHP ..... Bottom-Hole Pressure

BVP ..... Boundary Value Problem

CFL ..... CourantFriedrichsLewy

CTRS ..... Constant Terminal Rate Solution

DAE ..... Differential Algebraic Equation

DEIM ..... Discrete Empirical Interpolation Method

DFM ..... Drift Flux Model

EIM ..... Empirical Interpolation Method

FV ..... Finite Volume

FVM ..... Finite Volume Method

IADC ..... International Association of Drilling Contractors

ISS ..... Input-to-State Stability

LTI ..... Linear Time Invariant

MPD ..... Managed Pressure Drilling

MOR ..... Model Order Reduction

NRV ..... Non-Return Valve

ODE ..... Ordinary Differential Equation

PDAE ..... Partial Differential Algebraic Equation

PDE ..... Partial Differential Equations

pH ..... port-Hamiltonian

PI ..... Productivity Index

POD ..... Proper Orthogonal Decomposition

RB ..... Reduced Basis

RCD ..... Rotating Control Device

TFM ..... Two Fluid Model

UBD ..... Under Balanced Drilling

Chapter 1

Introduction

This thesis presents the research activities carried out in the HYDRA project in the framework of Marie Skodowska-Curie Innovative Training Networks (ITN-EID). HYDRA has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 675731. The main goal of this thesis is to develop different Model Order Reduction (MOR) techniques with specific objectives. These MOR methods lead to fit-for-purpose models to support drilling operations. Specifically, in this thesis, the effort is directed towards developing MOR techniques for hydraulics simulation for Managed Pressure Drilling (MPD) applications. In this chapter, the research activities are motivated and different steps to reach to the final goal of the project are explained. These steps can be summarized as: i) developing a new hydraulics model for MPD together with new numerical techniques, ii) developing efficient model order reduction techniques with error estimates, and iii) introducing port-Hamiltonian (pH) formulations of the hydraulics models to preserve key properties after reduction. Finally, the outline of the thesis concludes this chapter.

1.1 Motivation

To this date, human need for energy has been ever increasing. Among all energy resources, fossil energy has been resolving most of this need. Obtaining abundant fossil energy, especially through burning gas and oil, requires drilling into reservoirs (a formation of rock in which oil and gas has been trapped). This process consists in drilling a hole into the ground until a targeted reservoir is reached. The hole is created by rotating a cutting device, called drill bit, attached to the end of a heavy thick-walled pipe called the drillstring. During the drilling operation, a drilling fluid, usually known as mud, is pumped through the drillstring, exits through the drill bit nozzles with high velocity and pressure,

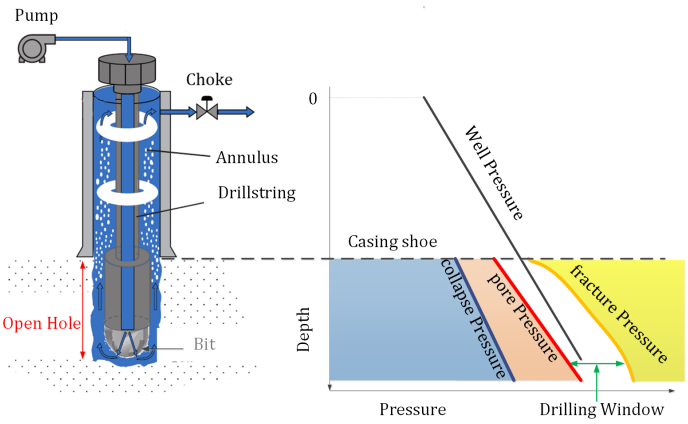

Figure 1.1: Schematic representation of a drilling rig (left) and different pressure zones (right).

cools and lubricates the bit and the drillstring, and carries drilled cuttings to the surface. The main function of the drilling mud is to build a barrier between the reservoir and the wellbore by providing a hydrostatic pressure, added to the frictional pressure drop in the annular space between the drillstring and wellbore. The section of the well that is exposed to the reservoir is called "open hole" (see Figure 1.1). Controlling pressure in the open hole is a major challenge in drilling.

Fulfilling the ever-increasing demand for fossil energy requires investigation and exploitation of difficult-to-drill reservoirs. Drilling to reach oil and gas in such reservoirs meets many challenges, regarding safety, environmental and economical risks. In particular, pressure control in such wells is highly critical as explained below. To control the pressure in the open hole (called downhole pressure), three pressure constraints, as denoted in Figure 1.1, should be considered:

I. *Collapse pressure*: A pressure profile constraint for the downhole pressure below which the well collapses, which should be avoided at all times;

II. *Pore pressure*: This is the pressure of the fluid trapped in the reservoir. When the downhole pressure is less than the pore pressure (i.e., for an underbalanced well), the fluid from the reservoir flows into the wellbore and, if not controlled properly, such scenario might lead to an explosion at the surface in case of a gas reservoir, such as the one happened in Deepwater Horizon drilling rig in Gulf of Mexico, 2010;

III. *Fracture pressure*: If the downhole pressure exceeds the fracture pressure,

the expensive mud penetrates into the formation and decreases the productivity of the reservoir.

Difficult-to-drill reservoirs are formations for which the pore and fracture pressures are very close to each other. In a safe drilling scenario, the downhole pressure profile has to be driven between the path of the pore and fracture pressure (called drilling window, see Figure 1.1), which is highly challenging if this window is narrow. A drilling technique called Managed Pressure Drilling (MPD) with improved well control capabilities has been recently developed to keep the downhole pressure under control. MPD differs from conventional drilling techniques by installing a Rotating Control Device (RCD) at the top of the well to create a seal around the drillstring, which, together with a back-pressure choke, enables manipulating the downhole pressure indirectly. This is sometimes also coupled with a dedicated back-pressure pump to enable control even when the main pump is shut off. In MPD, the slope of the pressure profile can be modified by either changing the pump flow rate (which is not used in practice) or by changing the mud density. More importantly, the starting point of the profile can be shifted by manipulating the opening of a valve installed at the top of the well, called a choke manifold. This control input affects the downhole pressure in a matter of seconds; i.e., much faster than non-MPD control inputs such as changing the mud density. This feature of MPD is also often used to handle uncertainties in the assumed physical parameters of the reservoir that enables the driller (or the automatic controller) to better and more quickly react to the scenario that is actually encountered.

To control the downhole pressure accurately in MPD, an accurate virtual drilling simulator is required, in which the hydraulics (pressure and flow evolution of the drilling mud in the drill-string and the annulus) in the well is modeled and can be simulated. This enables 1) virtual drilling scenario testing and 2) the use of model-based control techniques for MPD. To construct models for MPD that are both highly predictive/accurate and of reduced complexity, we face three challenges in constructing an MPD model for such a purpose: 1) Developing an accurate model and simulation platform for hydraulics in MPD, 2) Improving the computational efficiency of the simulation platform through application of Model Order Reduction (MOR) techniques, 3) Developing a mathematical framework to introduce special structure in the MPD model. This structure then can be used to preserve key properties of the system dynamics while reducing the dimension of the system model.

1.2 Modeling for Managed Pressure Drilling

In support of MPD application, advanced tools for virtual drilling scenario testing are needed, especially during the drilling operations to evaluate the effects of a potential action without risks. To model the flow inside the drillstring, which

contains single-phase flow in MPD, isothermal Euler equations are usually employed [115]. To analyze the flow inside the annulus, hyperbolic sets of Partial Differential Equations (PDEs) such as the Drift Flux Model (DFM) and the Two-Fluid Model (TFM) are employed [5], [69]. A critical issue in MPD modeling is the discontinuous cross-sectional area of the wellbore and the drillstring. The mathematical model should be able to capture the physics induced by this feature. The models introduced in [103], [151] take this aspect into account accurately. However, solving these models numerically is challenging due to the discontinuous features of the solution. Another challenge in modeling MPD is to connect the two models in the drillstring and annulus correctly, which has not been dealt with extensively in the literature. Especially at the interconnection of the drillstring and annulus, a switching dynamics of a Non-Return Valve (NRV) is present, which allows the mud to flow only from the drillstring to the annulus, not vice versa. This further challenges the modeling of MPD operations.

After constructing a reliable model for MPD on the infinite-dimensional level, an accurate numerical technique should be employed to discretize the dynamical model and obtain a finite-dimensional model representation as a basis for simulation studies. Highly nonlinear schemes [50], [69] are employed to discretize the PDEs over space and time and to predict the evolution of pressure and velocity. There are many techniques in the literature to numerically solve the isothermal Euler equations and the DFM in case of constant cross-sectional area [69], [83]. However, presence of area discontinuity in MPD model leads to discontinuities in the state solution of the coupled system. This introduces new challenges in the numerical technique to solve MPD. The method in [103] captures this effect for the Euler equations; however, such a method for the DFM is not available in the literature and the DFM with discontinuous cross-sectional area should be tackled by introducing a new numerical technique. Another challenge of the numerical scheme is the correct implementation of the boundary conditions together with the switching dynamics of the NRV. While many studies use the extrapolation method to compute necessary variables at the boundaries of a coupled system [5], the method in [69], [71] introduces transformed version of Euler equations and the DFM along their characteristics. These new PDEs are then discretized to correctly enforce the boundary conditions. However, combining these equations at the interconnection of the isothermal Euler equations and the DFM while taking the NRV into account is still an open challenge.

Although the MPD model at the infinite-dimensional level takes into account the effect of source terms, such as friction and gravity, correctly, the main challenge emerges at the finite-dimensional level after numerical discretization. Significant contributions of the source terms may lead to a drift of the numerical solution from the actual solution. For Euler equations in case of zero flow with only gravity, many numerical approaches, which are called well-balanced, have been proposed to satisfy the steady-state solution [100]. In case of non-zero flow and again with only gravity, few studies have been carried out [46]. Euler equa-

tions with both gravity and friction and also DFM lack a well-balanced solver in the literature.

In safety-critical applications such as drilling for oil and gas, a model has to be validated against field data. For Euler equations and DFM separately, many studies have been carried out to validate these models only against experimental data [5], [37]. There is a gap in the literature to validate an MPD model in case of single- and multi-phase flows against data directly collected from a real-life well.

The discretization yields a system of state-space models with high number of equations and unknowns. Simulating this system requires high demands concerning hardware and computation times. Moreover, the solution of the system has to be computed several times for different parameter configurations (so-called multi-query scenario). Let us assume that the different parameters, such as pump discharge, can be represented by a multi-dimensional parameter μ and the corresponding solution of the system by $U^{\mathcal{N}}(\mu)$. In the multi-query setting, the long computational time of these simulations in the forward analysis (see Figure 1.2 top) renders the well planning longer and increases the non-productive time of the drilling. On the other hand, the controller in MPD usually requires a reference pressure profile for the choke $p_{c\text{ref}}$. This reference pressure can be generated via solving an inverse problem by knowing the reference pressure profile for the downhole pressure $p_{dh\text{ref}}$ together with the measure choke pressure p_c, the downhole pressure p_{dh}, and the pump pressure p_p. This information can be fed into the inverse problem solver as an input (see Figure 1.2 bottom). In real-time setting, if the simulations in the inverse analysis are expensive to run, the computation time exceeds the sampling time of the sensors and this might lead to instability of the closed-loop system. Moreover, the state-space model obtained after the discretization is of high dimension, rendering model-based controller design infeasible. This can be seen as another motivation for model order reduction.

To summarize, the main challenges that are tackled within Part I are: i) proposing a numerical technique for the DFM with discontinuous cross-section, ii) developing an MPD model by interconnecting appropriate models of single- and multi-phase flow models via MPD equipment, iii) validating the MPD model under single- and multi-phase flow against real field-data, and finally, iv) developing a numerical method to predict the accurate steady-state solutions in the presence of significant contributions of source terms.

1.3 Model Order Reduction

To control the downhole pressure in the desired region, via a model-based controller, a computationally fast and trustworthy simulator or a low-complexity model for the hydraulics in the well is required. Therefore, accelerating the simulator is vital.

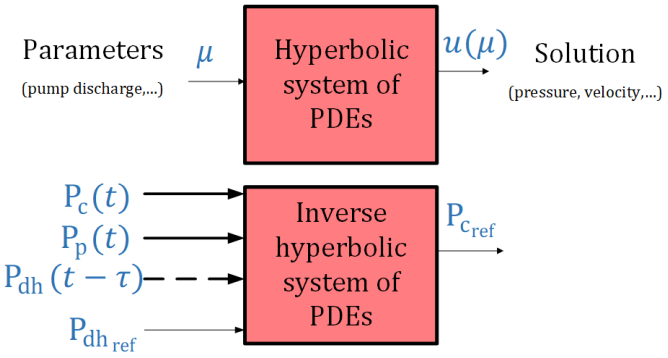

Figure 1.2: Time-consuming simulations for MPD, forward analysis (top), inverse analysis (bottom).

To attain faster than real-time simulations and develop models to be used in multi-query setting, the model complexity and the dimension of the system of equations should be reduced. The reduced-order model generated by this reduction should be accurate enough to capture the most important aspects of the physics involved in the original model. In the MPD context, the most important feature to be preserved is the distributed nature of the pressure dynamics. In this thesis, MOR approaches are used to develop a fit-for-purpose model for simulation of single- and two-phase flow in drilling.

Discretized models for MPD yield parametric models, to be simulated from scratch for any change in the parameters. These parameters can address geometrical features of the well or the mud properties. Therefore, a method should be used that exploits the parameter dependency of the model to gain maximum efficiency. In this thesis, we are mainly concerned with a class of model reduction techniques for parametric PDEs, so-called Reduced Basis (RB) methods. These methods yield low-dimensional parametric models typically leading to fast and accurate numerical simulations, suitable for multi-query and real-time scenarios. However, this acceleration in the simulation comes with a price: errors in approximating the solutions. For the RB method, two aspects are pivotal: first, basis generation to project the full-order space onto a lower-order space, and second, a certification or an estimation of the error induced by the reduction procedure.

Let us assume that the parameters μ introduced in the last section lie in a set $\mathcal{D}$, i.e., $\mu \in \mathcal{D} \subset \mathbb{R}^b$, where b denotes the number of varying parameters. The parametric solution $U^{\mathcal{N}}(\mu)$ originates from a standard, high-dimensional solution space of dimension $\mathcal{N}$ (e.g., originating from finite-volume discretization). Frequently, the output quantity $y^{\mathcal{N}}(\mu)$ of the model is more of interest than the full state solution. Particularly for MPD, this quantity of interest is the

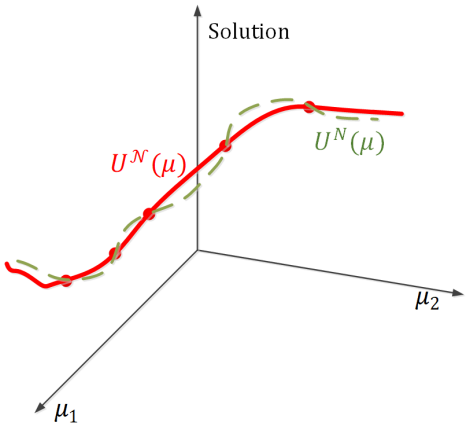

Figure 1.3: Solution manifold and the RB approximation for a case with two varying parameters $\mu = \{\mu_1, \mu_2\}$.

pressure profile in the open hole section. To circumvent the computation of the high-dimensional $U^{\mathcal{N}}(\mu)$, the RB method provides a low-dimensional, and hence rapidly computable, approximation for the parametric solution, called $U^N(\mu)$. This solution originates from solving a system of N equations. This paves the way for the fast approximation of the output $y^N(\mu)$. The main assumption behind the RB method is the fact that the solution manifold, i.e., the set of parametric solutions of the system, often can be well approximated by a low-dimensional subspace. In RB-methods, one popular way is the construction of this subspace from the snapshots of the full-order model for a suitable set of parameter values $\mu^i, i \in \{1, \cdots, k\}$. The most popular method to extract the basis functions for this subspace is Proper Orthogonal Decomposition (POD) [117]. However, POD requires the solution for all parameter values in the discrete parameter domain, rendering its application limited. An alternative is to approximate the space of solutions, which requires a careful choice of the parameters μ^i by the greedy algorithm [65]. After the construction of the space, the reduced model is obtained, e.g., by Galerkin projection [31], and provides an approximation $U^N(\mu)$ of the solution and an approximation $y^N(\mu)$ of the output quantity of interest. See Figure 1.3 for an illustration of the RB-approximation scenario.

The computational procedure in RB methods is decomposed in an offline and online phase as shown in Figure 1.4. During the offline phase, performed only once, the basis functions for the low-dimensional space are generated and further auxiliary operators are precomputed, mostly based on the affine decomposable nature of the system [84]. These basis functions are obtained based on the solution of the full-order model with respect to parameter settings $\mu^i, i \in \{1, \cdots, k\}$.

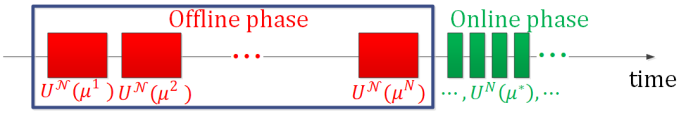

Figure 1.4: Offline and online phases in the RB approximation.

The offline phase is typically expensive as it entails several simulations of the full-order model (red blocks in Figure 1.4). Then, in the online phase, for any new varying parameter μ^*, the approximate output is provided rapidly (shown by green blocks in Figure 1.4). The computational complexity of the online phase should not depend on the dimension of the high-dimensional space. Hence, the computational complexity of the online phase will reduce depending on the dimension of the low-order model. This dimension should be chosen in a trade-off between the required speedup in the simulation and the desired accuracy of the RB solution. Notably, the computational efficiency of RB pays off in the online phase, where the solution for many new parameters is demanded. This is generally required in the optimization procedure and multi-query settings. If the solution of the system is only needed few times, it is suggested to directly use the full-order model solution [84].

The RB method is well developed for linear systems with Dirichlet boundary conditions [89]. For nonlinear systems, RB is usually coupled with the Empirical Interpolation Method (EIM) [59]. However, inclusion of nonlinear state-dependent boundary conditions in RB is challenging and, in MPD, nonlinear state-dependent boundary conditions play a crucial rule in the overall dynamics.

In addition to the reduction strategy, rapidly computable, effective and rigorous error bounds and estimates are necessary to quantify the error in the state or output due to the order reduction [89]. The advantage of such error bounds and estimates are twofold. First, without having access to a reliable error bound and estimate, to run the greedy algorithm, the high-fidelity solution for all members of the discretized parameter domain is required to compute the actual error induced by reduction. This is time consuming and also memory demanding. By introducing a good error bound or estimate, instead of finding the actual error, a bound or an estimate of this value is provided cheaply to choose the parameters in the greedy algorithm. This accelerates the offline phase and also reduces the memory demand to save the solution for all members of the discretized parameter domain [84]. Second, the solution obtained in the online phase is certified and the error loss is approximated or bounded [182]. Such certificates have a key role in 1) making a well-informed tradeoff between model complexity and model accuracy and 2) providing confidence on the trustworthiness of the reduced-order model for the user. Error bounds and estimates in RB method are tailored for specific systems [84], [182]. Beyond these systems, error bounds

and estimates either are typically conservative or exponentially increases over time. Error estimates for general nonlinear systems are missing in the literature.

To summarize, Part II tackles i) tailoring the RB method for systems with nonlinearly state-dependent boundary conditions, ii) developing error estimates for systems with local and distributed nonlinearities, and finally, iii) approximating MPD-related hydraulics by the RB method while taking into account the area discontinuity as a varying parameter.

1.4 Port-Hamiltonian Systems

It has been observed that the RB methods do not preserve stability in the multiphase flow scenario of interest. Therefore, MOR techniques which preserve key properties of the system such as stability are necessary, which is currently lacking in the RB context.

Many complex systems are modeled by a network of different subsystems, each having their underlying mathematical model representations. Energy-based modeling of each of these subsystems can yield a port-Hamiltonian (pH) representation, providing a modular framework for multi-physics and interconnected systems. PH systems have recently received a lot of attention for modeling physical phenomena governed by nonlinear PDEs and ordinary differential equations [64], [153]. In general, a pH system is a system of the following form:

$$\partial_t z = (\mathcal{J}(z) - \mathcal{R}(z))\, \delta_z \mathcal{H}(z), \tag{1.1}$$

where t represents time, $z := z(t, x)$ is the state of the system, x is the spatial coordinate, $\mathcal{J}(z)$ is a formally skew-adjoint operator describing the interconnection of different parts of the system, $\mathcal{R}(z)$ is a positive semi-definite matrix describing the power loss in the system, $\mathcal{H}(z)$ is a scalar-valued functional describing the energy of the system and δ denote the variational derivative. The operator $\mathcal{J}(z)$ usually contains differential operators such as ∂_x. System (1.1) requires a suitable set of boundary conditions to allow non-zero energy flow through the boundary and guarantee power preservation of the system [64]. Moreover, structure-preserving methods for discretization and model order reduction of infinite-dimensional pH systems can preserve certain original system-theoretic properties such as stability and passivity [64]. In addition, a pH realization offers a suitable description of the components for the modeling, analysis and controller design [64] while taking the infinite-dimensional nature of the dynamics into account. Generally, the Hamiltonian functional $\mathcal{H}$ defined in pH framework represents a good candidate for the Lyapunov function, rendering the physics-based control design and the stability proof more tangible [119]. A key property of pH systems is that, if the interconnection of the pH subsystems is performed in a power-preserving manner [42], [64], it preserves the pH structure of the coupled (aggregated) system. The key point in aggregating different pH subsystems is the identification of the interconnection structure and casting this

interconnection into a power-preserving structure. Isothermal Euler equations are proved to have a pH structure [181]; however, investigation of the pH structure in DFM and TFM is missing in the literature. Moreover, interconnection of the isothermal Euler equations and DFM or TFM is lacking in the literature.

1.5 Objectives and Contributions

From the overview of the state of the art presented in the previous sections and the related open challenges, the following research objectives are pursued in this thesis:

1. Modeling, numerical techniques for simulation and model validation for MPD:
    - Developing a model accurately representing all dynamics involved in MPD,
    - Proposing a numerical solver to take into account the effect of area discontinuity in pipes,
    - Developing a numerical solver for the MPD model,
    - Validation of the MPD model against the field data,
    - Proposing a numerical technique to approximate the steady-state solutions more accurately.
2. Model order reduction of the MPD model together with error estimates:
    - Developing model order reduction techniques for the MPD model,
    - Proposing error estimates to approximate the error induced by reduction.
3. Investigating pH structure for MPD model:
    - Constructing an MPD model with pH structure.

Due to the different objectives of this thesis, it has been divided into three parts with different contributions, which are detailed below.

In Part I, the focus lies on the modeling, numerical simulation and validation for MPD. To this end, a numerical solver for the DFM with discontinuous cross section is developed. This numerical solver can be applied to any Godunov-type scheme to account for the discontinuous features in the geometry of drillstring and annulus. To further extend our numerical solver to simulate fluid flows for MPD, characteristics boundary conditions of the drillstring and annulus are coupled at the location of drilling bit. At this location, a dynamical NRV is designed, which blocks the drilling mud flow from annulus to drillstring. To validate the results of the numerical solver, simulation results are compared to

the data gathered from real drilling wells in case of single-phase flow, liquid influx and gas influx. To accommodate the liquid influx effect, a new reservoir model is developed and coupled with the MPD dynamics. Finally, to take into account the effect of significant source terms, a new upgrade procedure for first-order Godunov-type schemes has been introduced. This upgrade is tailored for the isothermal Euler equations and the DFM in the presence of laminar friction and gravity.

Model order reduction and developing error estimates are the themes of Part II. To be able to capture the nonlinear and state-dependent boundary conditions involved in MPD, a new RB ansatz together with an interconnection between the boundary dynamics and internal dynamics has been proposed. This method exactly enforces the boundary conditions at the ROM level without generating non-physical spikes in the solution at the boundaries, which the classical ansatz does. For the first time, the developed RB method is applied to the MPD with single-phase flow while considering the location and numbers of area discontinuities as a varying parameter. The effect of this varying parameter is taken into account by enriching the RB functions with local basis functions. This local enrichment truly captures the physics induced by the discontinuous features. In case of single-phase flow, the internal dynamics is linear and the only nonlinearities appear at the boundaries. Therefore, the interconnection of these dynamics can be formulated as a Lur'e-type system. To reduce the dimensionality, the linear (internal) system is reduced while the finite and low dimension of boundary dynamics is not changed. Exploiting the linear structure of the internal dynamics and also the ℓ_2-gain notion, we have proposed a new error estimate for linear systems coupled with systems with local nonlinearities. To extend the error estimate to systems with distributed nonlinearities, a new perspective on the coupling between RB and EIM through Lur'e-type systems is proposed. However, for the existence of this error estimate, a small-gain condition should be satisfied, which significantly restricts the range of state values in the simulations. This motivated the introduction of a hierarchical empirical error estimate which does not suffer from such restrictions.

Part III is dedicated to pH system modelling for MPD. RB methods do not preserve stability through reduction [89]. For the case of DFM, the application of RB and EIM leads to an unstable system. To circumvent this issue, for the first time, a pH framework is developed for the TFM and DFM. Then, the pH formalism of isothermal Euler equations and the TFM are coupled together via MPD equipment models to construct an aggregated MPD model. To preserve power through the pH MPD model, a condition, based on the velocity of the drilling mud before and after the drilling bit, is derived to render the drilling bit model dissipative. The platform proposed in this thesis can be used as a stepping stone for a controller design based on energy methods for MPD.

1.6 Organization of the thesis

The thesis has been organized in three parts. In the first part, MPD modeling is explained and the challenges for its numerical approximation are elaborated. In the second part, the MOR techniques are developed together with error estimates and the novel RB method is applied to MPD models. Finally, in the last part, pH formulations of the models used in MPD are developed and are connected together to build an aggregated pH model for MPD.

Particularly, MPD can be characterized by interconnection of subsystems governed by a single-phase flow in the drillstring and a two-phase flow in the annulus, and, mathematical models governed by nonlinear ordinary differential equations or static equations [128]. A single-phase flow is usually modelled by the isothermal Euler equations, which obey a pH formalism [181]. For two-phase flow modelling, the Two-Fluid Model (TFM) and the Drift Flux Model (DFM) are typically employed [5]. We show that the TFM and a DFM without slippage between the two phases can also be cast in the pH formalism [15]. Drilling with MPD is composed of single- and two-phase flow pH realizations, which can be interconnected via MPD equipment (bit) in a power-preserving manner to form an aggregated pH system. We show that the drilling bit model connecting the drillstring and the annulus is conditionally power-preserving.

1.6.1 Part I: Modeling and Validation of Multi-phase Flow

Hydraulics for MPD consists of a single-phase flow in the drillstring and a potentially multi-phase flow model in the annulus, connected through nonlinear equations of the bit. Boundary conditions of such systems are described by the MPD equipment. However, the MPD model cannot be numerically solved by classical numerical methods due to, i) discontinuous well geometry, ii) essential contribution of frictional and gravitational source terms, iii) nonlinear state-dependent boundary conditions containing switching dynamics. In Part I, a model and its corresponding numerical approach are presented to deal with these challenges. Moreover, the model and its numerical solution are validated against field data obtained from real-world drilling wells in this part. Therefore, Part I consists of:

- **Chapter 2: A Godunov-type Scheme for the Drift Flux Model with Variable Cross Section**
  - In this chapter, a numerical solver is proposed to take into account the discontinuous geometry of the pipelines for multi-phase flow. This method enforces some algebraic relations at the location of the area discontinuity, which resolves the non-physical spikes generated by classical solvers at these locations.

- **Chapter 3: A Well-Balanced Godunov-Type Scheme for the Isothermal Euler Equations and the Drift Flux Model with Laminar Friction and Gravitation**
  - A new method is developed here to tackle the significant effect of source terms in both single- and multi-phase flow models. Laminar friction and gravity source terms are included in the isothermal Euler equation and the DFM. The proposed method enforces algebraic constraints from the physical steady-steady solutions, which brings about significant accuracy increase in steady-state prediction compared to classical solvers. The method presented here can be applied to any Godunov-type schemes.
- **Chapter 4: Modelling and numerical implementation of managed pressure drilling systems for the assessment of pressure control systems**
  - The methodology developed in Chapter 2 is extended to simulate hydraulics in MPD. Dealing with nonlinear boundary conditions is another main focus of this chapter. The switching dynamics for the NRV is proposed here. Moreover, validation of the MPD model against the field data in case of single-phase flow is carried out in this chapter.
- **Chapter 5: Model Validation for Multi-phase Flow**
  - This chapter completes the validation of the model proposed in the previous chapter. In the current chapter, the MPD model in case of multi-phase flow is validated against the field data. This is carried out by exploiting data gathered from liquid-liquid flow and liquid-gas flow. In case of liquid-liquid flow, a novel reservoir model is proposed which accurately predicts the field data.

1.6.2 Part II: Model Order Reduction

The numerical simulation of hydraulics in MPD is computationally expensive to be run in real time and in a multi-query setting. Therefore, the model is not suitable for the optimization of drilling plans or simulation-based controller design. To enable these features, MOR of the parameterized model for MPD is beneficial. In this part, the RB method tailored for MPD in case of single-phase flow is explained in three chapters:

- **Chapter 6: Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions**
  - In this chapter, the RB method is utilized for the reduction of single-phase flow models. To deal with the highly nonlinear, state-dependent

boundary conditions employed in MPD, a modified version of the RB method has been proposed. To quantify the accuracy loss due to the reduction, a new error estimate is developed based on a Lur'e type model formulation for the discretized model, which can be efficiently extended to other systems of such a form. Results show that the new RB ansatz does not generate non-physical results at the boundaries and the error estimate approximates the actual error with a high accuracy.

- **Chapter 7: Error estimates for model order reduction of Burgers' equation**
  - The existence condition of the error estimate developed in the previous chapter is not always satisfied when highly nonlinear terms are present in the system of equations. To enlarge the existence condition for such error estimate for nonlinear systems, a loop transformation has been conducted in this chapter. Moreover, to generalize the proposed approach to highly nonlinear systems and also for systems with distributed nonlinearities, a new hierarchical error estimate based on two ROM solutions with two different levels of accuracy has been proposed based on the available simulation data in the RB method.
- **Chapter 8: Reduced Basis Method for Managed Pressure Drilling Based on a Model with Local Nonlinearities**
  - In this chapter, a new RB method has been developed to capture discontinuous features of the drilling well geometry for single-phase scenarios. This is done by the enrichment of the RB space by basis functions with local support. Simulations confirm that this new method captures the physics in MPD in the presence of discontinuous well geometry.

1.6.3 Part III: Port-Hamiltonian Systems

The RB method works well for the case of single-phase flow as the system is almost linear except at the boundaries. For the multi-phase flow model, applying RB method to the model obtained after discretization typically generates an unstable reduced-order system, possibly induced by the high complexity of multi-phase flow models. One way to circumvent this issue is to formulate the multi-phase flow model in a pH formalism and apply structure-preserving MOR techniques to the obtained model, thereby avoiding such issues related to unstable reduced-order models. In this part, we set up a framework to generate such pH-based model formulations in the two chapters mentioned below.

- **Chapter 9: Port-Hamiltonian Formulation of Two-phase Flow Models**

  - In this chapter, pH model formulations are proposed for multi-phase flow models, the TFM and DFM, used in MPD modeling.

- **Chapter 10: Power-Preserving Interconnection of Single- and Two-Phase Flow Models for Managed Pressure Drilling**

  - The pH models, developed in the previous chapter, are interconnected in a power-preserving manner, yielding an aggregated pH system model for MPD. To render the aggregated system power-preserving, a condition is derived on the drilling bit model.

1.7 Publications

The content of this thesis is mostly built upon the publications of the author during the 4 years of the PhD study. These publications are listed below.

Remark 1.1. *It might be the case that other authors of the following papers include the same paper as part of their thesis.*

1.7.1 First-Authored papers

The contributions of the candidate have been published in the following journal papers and conference papers with peer review as the first author.

Journal papers

- **M.H. Abbasi**, S. Naderi Lordejani, N. Velmurugan, C. Berg, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***A Godunov-type Scheme for the Drift Flux Model with Variable Cross Section***, *Journal of Petroleum Science and Engineering*, Volume 179, August 2019, Pages 796-813.

- **M.H. Abbasi**, S. Naderi Lordejani, C. Berg, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***An Approximate Well-Balanced Godunov-Type Scheme for the Isothermal Euler Equations and the Drift Flux Model with Laminar Friction and Gravitation***, *International Journal for Numerical Methods in Fluids*, in press, 2020.

- **M.H. Abbasi**, L. Iapichino, B. Besselink, W.H.A. Schilders, N. van de Wouw, ***Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions***, *Computer Methods in Applied Mechanics and Engineering*, Volume 360, 1 March 2020, 112688.

- **M.H. Abbasi**, S. Naderi Lordejani, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***Reduced Basis Method for Managed Pressure***

Drilling Based on a Model with Local Nonlinearities, *International Journal for Numerical Methods in Engineering*, in press, 2020.

- **M.H. Abbasi**, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***A Data-based Stability-preserving Model Order Reduction Method for Hyperbolic Partial Differential Equations***, Submitted to *SIAM Journal on Control and Optimization*, 2020.

- **M.H. Abbasi**, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***A Non-Intrusive Stable Model Order Reduction Method for Drift Flux Model***, In preparation.

Conference papers

- **M.H. Abbasi**, L. Iapichino, B. Besselink, W.H.A. Schilders, N. van de Wouw, ***Error estimates for model order reduction of Burgers' equation***, *IFAC World Congress*, Berlin, Germany, July 2020.

- **M.H. Abbasi**, H. Bansal, H. Zwart, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***Power-Preserving Interconnection of Single- and Two-Phase Flow Models for Managed Pressure Drilling***, *American Control Conference*, Denver, U.S.A., July 2020, Pages 3097-3102.

1.7.2 Co-Authored papers

In addition, the candidate has contributed as co-author to the following journal papers and conference papers with peer review. The specific contribution of the author to these publications are explained in detail.

Journal papers

- S. Naderi Lordejani, B. Besselink, **M.H. Abbasi**, G. -O. Kaasa, W.H.A. Schilders, N. van de Wouw, ***Control-oriented modelling for managed pressure drilling automation using model order reduction***, *IEEE Transactions on Control Systems Technology*, in press, 2020,

  - Modeling, numerical implementation and simulations corresponding to the full-order model are performed in collaboration with S. Naderi Lordejani. The model order reduction development and the corresponding simulations are solely the contribution of S. Naderi Lordejani.

- S. Naderi Lordejani, **M.H. Abbasi**, N. Velmurugan, C. Berg, J.A. Stakvik, B. Besselink, L. Iapichino, F. Di Meglio, W.H.A. Schilders, N. van de Wouw, ***Modelling and numerical implementation of managed***

pressure drilling systems for the assessment of pressure control systems, *SPE Drilling & Completion*, in press, 2020,

  - Modeling, numerical implementation and simulations are performed in collaboration with S. Naderi Lordejani and N. Velmurugan. The validation of the single-phase flow model against the field data is solely the contribution of S. Naderi Lordejani.

- H. Bansal, P. Schulze, **M.H. Abbasi**, H. Zwart, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***Port-Hamiltonian Formulation of Two-phase Flow Models***, Submitted to *Systems & Control Letters*, 2020,

  - Derivation of the Hamiltonian functional, rewriting the two-fluid model in terms of only four primitive variables and the numerical simulations are the contribution of the author.

Conference papers

- S. Naderi Lordejani, B. Besselink, **M.H. Abbasi**, G. -O. Kaasa, W.H.A. Schilders, N. van de Wouw, ***Model order reduction for managed pressure drilling systems based on a model with local nonlinearities***, *IFAC-PapersOnLine*, Volume 51 (8), January 2018, Pages 50-55,

  - Modeling, numerical implementation and simulations corresponding to the full-order model are performed in collaboration with S. Naderi Lordejani. The model order reduction development and the corresponding simulations are solely the contribution of S. Naderi Lordejani.

Part I

Modeling and Validation of Multi-phase Flow

This part concerns the modelling of MPD-related hydraulics, to be used for model order reduction in the subsequent parts of the thesis. Here, a new technique to capture the effect of the discontinuous cross-sectional area of a pipe on a multi-phase flow is discussed and compared to existing techniques in the literature. Apparent from the results, other techniques in literature generate non-physical results at the location of area discontinuity while the proposed technique captures the physics induced by this feature accurately. The order of accuracy of the scheme is also investigated.

Next, the correct approximation of steady-state solution in the presence of powerful source terms is studied. Steady-state solutions play a crucial role in the decision-making process of industrial systems and thus should be computed accurately. We propose an upgrade of first-order Godunov-type schemes to predict the steady-state solution significantly more accurately than the original Godunov-type scheme for single- and two-phase flow models. The upgrade is applied to Rusanov scheme and tested over illustrative test cases of zero and non-zero mass flow rates.

To enable MPD simulations and interconnect the drillstring and annulus internal dynamics, state-dependent and nonlinear boundary conditions involved in MPD are investigated. U-tube modeling of MPD is carried out by connecting two pipes with discontinuous cross-sectional areas, representing drillstring and annulus, via these boundary conditions. The aggregated MPD model successfully simulates MPD-relevant scenarios such as kick circulation. Moreover, its accurate prediction is validated against field data for single-phase flow scenarios.

To enable the simulation of liquid influx, the MPD model is extended to simulate liquid-liquid flow as well. A new reservoir model for the case of liquid influx is also introduced and coupled to the MPD. The coupled model is then validated against liquid and gas influx scenarios. Results show that the MPD model is capable of reproducing the field data accurately through adjusting the uncertain parameters in the model.

Chapter 2

A Godunov-type Scheme for the Drift Flux Model with Variable Cross Section

This paper presents a modification of a classical Godunov-type scheme for the numerical simulation of a two-phase flow in a pipe with a piecewise constant cross-sectional area. This type of flow can occur in wellbores during drilling for oil and gas as well as after well completion. Contrary to classical finite-volume schemes, the numerical scheme proposed in this paper captures the steady-state solution of the system without generating non-physical discontinuities in the numerical solution close to the locations of discontinuities in the cross-section. Moreover, the proposed scheme can be extended to problems with piecewise continuous cross-sectional area. This extension is achieved by discretization of the area along the spatial domain and converting the piecewise continuous area into a piecewise constant area. The proposed scheme reduces to the classical scheme when the cross-sectional area is constant along the spatial domain. For the purpose of computational efficiency, the modification to the classical scheme is only applied at the locations of area variation and the numerical solver reduces to the classical scheme where the cross-sectional area is constant. It is also shown that the proposed scheme can be effectively used to simulate two-phase flows arising from the perturbation of the steady-state solution. The effectiveness of the proposed scheme is shown through illustrative numerical simulations. Finally, it should be noted that the proposed scheme retains the same order of accuracy as the underlying classical scheme.

This chapter is based on "M.H. Abbasi, S.Naderi Lordejani, N.Velmurugan, C.Berg, L.Iapichino, W.H.A.Schilders, N.van de Wouw, ***A Godunov-type Scheme for the Drift Flux Model with Variable Cross Section***, *Journal of Petroleum Science and Engineering*, Volume 179, August 2019, Pages 796-813".

2.1 Introduction

Reliable models and accurate numerical solutions for single- and two-phase flows are necessary for many industrial applications, such as drilling for oil and gas and flow in fuel bundles and pipelines [86], [120], [132]. Modeling of the transient behavior of the flow dynamics in these industrial systems plays a crucial role in the design, decision making and control of such systems. For the simulation of two-phase flows, the one-dimensional Drift Flux Model (DFM), which is constituted by a set of first-order nonlinear hyperbolic partial differential equations, has gained attention [69], [72] due to its balance between predictive capabilities and simplicity. Compared to the two-fluid model, the DFM is favorable from a numerical simulation perspective [127]. In addition, the DFM remains hyperbolic over a wider region of the variables and it is also more accurate than the two-fluid model in homogeneous two-phase regimes [127].

In many industrial applications, the computational domain, typically a pipe, frequently has a variable cross-sectional area along its length. In particular, a drilling well experiences discontinuities in the cross-sectional area [93] as schematically illustrated in Figure 2.1. These area discontinuities affect the resonance frequency of the wave propagation effects inside the system, especially the rapid pressure dynamics. If this phenomenon is not considered, the model may lose its predictive capacity. Moreover, in the scope of controller design for Managed Pressure Drilling (MPD), the system performance may seriously deteriorate when such effects are not appropriately represented in the model. Hence, a model that accounts for such phenomena is required.

Two-phase flow in a pipe with variable cross-sectional area increases the complexity of the governing model and, subsequently, its numerical solution as a non-conservative term is added to the governing equations [57]. This means that after adding this term, all derivatives over the spatial variable cannot be gathered into a single differential term.

Classical finite-volume schemes are suitable for numerical simulation of conservative hyperbolic PDEs, such as the DFM with a constant cross-sectional area [67], [72], [172]. However, these classical methods cannot be effectively used to solve non-conservative PDE models, such as the DFM with variable cross-sectional area [103]. A common approach to incorporate the area variation is to treat the non-conservative term as a source term [169]. This treatment leads to non-physical and numerical spikes in the numerical solution and, subsequently, this approach cannot be reliably used [103].

Addressing the issue of the presence of non-conservative terms in mathematical models of various systems in the scope of numerical implementation is an active research area. Different methods have been developed for the simulation of the behavior of a single-phase flow in a pipe with a variable cross-sectional area. Instead of treating the non-conservative term as an additional source term, a modification to the Rusanov scheme [171] has been proposed in [49] to capture

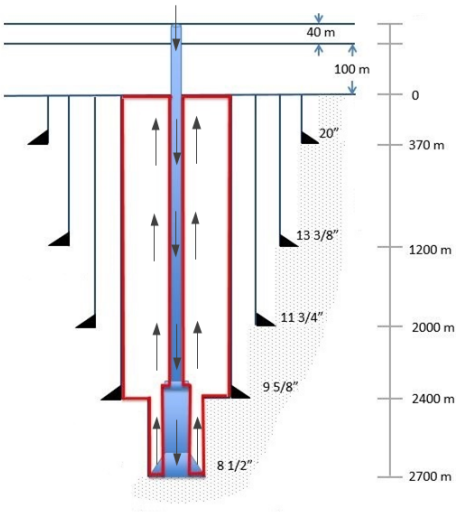

Figure 2.1: The discontinuous cross-section of area in a real drilling well with diameters shown in inch and depths shown in meter (as common in the drilling community). The path of the fluid is marked by black arrows. The data is extracted from [93].

the steady-state solution of the Euler equations. However, this method is not well-balanced in the presence of non-zero flow in the system, i.e., the numerical solution does not preserve the steady-state solution for non-zero flow scenarios. A model-based modification of the input arguments of the finite-volume scheme has also been introduced in [57], [88], [103]. All the mentioned works deal with the variable cross-sectional area in single-phase flow systems while two-phase flows frequently occur in many realistic industrial applications [37], [86], [120], [132].

To the best of the authors' knowledge, the effect of non-conservative terms in two-phase flow models has been studied only to a rather limited extent for conservative shock capturing schemes. As an example, in [167], the non-conservativeness in the two-fluid model originating from the state variables is considered. However, the non-conservativeness originating from the variable cross-sectional area is not discussed. Therefore, this paper focuses on developing a reliable numerical approach for the DFM capturing the effects of variations in the cross-sectional area by introducing a model-based scheme, inspired by [103].

The results of this paper can be used to simulate the flow of gas and liquid mixtures in pipelines. In particular, this kind of flow is common in the up-stream, midstream and downstream sector of the oil and gas industry. As in the upstream sector, for any drilling well, the understanding of the flow (and

pressure) dynamics in the drill-string and annulus is essential. Herein, the drill-string consists of a series of drill pipes and the bottom hole assembly (assembly of heavy weight drill pipes and mud motors). Together these system components represent a drastically varying flow path, both inside the drill-string and the annulus (refer to Figure 2.1). In drilling operations for oil and gas, multi-phase fluid flow arises in several cases, such as gas influx into the annulus or during under-balanced drilling where the gas is also present in the drill-string in addition to the annulus. Moreover, the proposed method can be used to validate hydraulics models in a drilling well with the DFM, as an extension to [37]. In many studies for the DFM validation in a drilling well, such as e.g. the one in [37], the effect of area discontinuity has been ignored. However, dealing with the field data, the effects of these discontinuities should be taken into account and the developments in this paper support this. The work of this paper can then be used to support the operational design of MPD-based operations and controller design for MPD. In the midstream operations, multi-zone completion designs, using a liner or an open-hole in combination with tubings, pose a multi-phase flow scenario in a pipe with potentially variable cross-sectional area during the production phase. In the downstream sector, the refining and separation of the natural gas and crude oil can be simulated using the results of this paper.

Contributions of this paper are provided in two areas. First, this paper reviews the current techniques to deal with non-conservative terms within single-phase flow. These techniques, which are not applicable for the DFM, are adapted for the DFM. Secondly, new approaches within the context of the DFM are introduced to deal with the non-conservative term induced by the variable cross-sectional area. Since the goal of this paper is to evaluate the merit of the scheme in capturing the effects of area variation, the effect of source terms such as friction and gravity has not been considered. The evaluation of the scheme in the absence of the source term is a common practice in the finite-volume community [103], [105], [114]. In addition, considering source terms raises the issue of the well-balancedness [38], which is beyond the scope of this paper. Incorporating these additional source terms is the subject of future works.

The structure of this paper is as follows. In Section 2.2, an overview of the model is given together with a comparison between the conservative and non-conservative DFM and the corresponding eigenvalue analysis is performed. Various methodologies to deal with the non-conservative term are proposed in Section 2.3. In Section 2.4, the effectiveness of the different variable cross-section treatments in preserving the steady-state solution is evaluated through numerical tests and, subsequently, transient simulations are performed. Finally, conclusions and recommendations for future works are presented in Section 2.5.

2.2 Two-Phase Flow Model

In this section, the DFM in case of constant and variable cross-sectional area is presented. Eigenvalue analysis is performed for the non-conservative system with variable cross-sectional area. Since the dynamics originating from the perturbation to the steady states are of numerical interest in this study, the steady-state solution is also presented in this section. Next, the necessary boundary conditions for both the transient and the steady-state model are defined.

2.2.1 Introduction to the Drift Flux Model

The Drift Flux Model is widely used to describe the behavior of two-phase flow systems [5], [69], [94], [185]. It consists of two mass balance equations, one for each phase, and one combined momentum balance equation for the mixture of the phases. The governing equations for one-dimensional systems are given by

$$\frac{\partial(\alpha_l \rho_l)}{\partial t} + \frac{\partial(\alpha_l \rho_l u_l)}{\partial x} = 0, \tag{2.1a}$$

$$\frac{\partial(\alpha_g \rho_g)}{\partial t} + \frac{\partial(\alpha_g \rho_g u_g)}{\partial x} = 0, \tag{2.1b}$$

$$\frac{\partial(\alpha_l \rho_l u_l + \alpha_g \rho_g u_g)}{\partial t} + \frac{\partial(\alpha_l \rho_l u_l^2 + \alpha_g \rho_g u_g^2 + p)}{\partial x} = 0, \tag{2.1c}$$

where $\alpha(t,x), \rho(t,x), u(t,x)$ and $p(t,x)$ are, respectively, the volume fraction, density, velocity and pressure, which are functions of time t and the one-dimensional spatial coordinate x. The subscripts l and g denote the liquid and gas, respectively. Noteworthy, the DFM is based on the assumption of mechanical equilibrium between the two phases, i.e., the pressure of the gas and the pressure of the liquid are equal. The DFM as in (2.1) contains seven variables while it is expressed in only three equations. Thus, four other equations, called closure relationships, are required to, potentially, uniquely solve the system of equations. The most widely used closure relationships are listed below [69], [91]:

$$\alpha_l + \alpha_g - 1 = 0, \tag{2.2}$$

$$u_g - (K u_{mix} + S) = 0, \tag{2.3}$$

$$p - \rho_g c_g^2 = 0, \tag{2.4}$$

$$p - \big((\rho_l - \rho_0) c_l^2 + p_0\big) = 0. \tag{2.5}$$

Equation (2.2) implies that every section of the pipe is filled up with a mixture of the liquid and gas. The slip law (2.3), showing a static relation between the velocity of the gas and the liquid, compensates for the fact that only one momentum balance is included in the DFM. Here, K and S are, respectively, the distribution coefficient and the drift velocity of the gas relative to the liquid defined according to the flow regime [30], [142] and $u_{mix} = \alpha_l u_l + \alpha_g u_g$ is the

velocity of the mixture. Equations (2.4) and (2.5) represent the equation of state (EOS) for the gas and liquid phases, respectively. c_l and c_g are the speeds of sound in the liquid and gas phases, which can be assumed to be constants or be functions of other variables, such as pressure. Finally, ρ_0 and p_0 are, respectively, the reference values for density and pressure around which the EOS for the liquid has been linearized.

Model (2.1) is based on the assumption that the cross-section is constant all along the spatial domain. When this does not hold, the DFM in (2.1) should be modified in order to take into account the cross-sectional variations. By including a variable cross-sectional area along the computational domain, $A = A(x)$, as used in [151], the system (2.1) changes to

$$\frac{\partial(\alpha_l\rho_l A)}{\partial t} + \frac{\partial(\alpha_l\rho_l u_l A)}{\partial x} = 0, \tag{2.6a}$$

$$\frac{\partial(\alpha_g\rho_g A)}{\partial t} + \frac{\partial(\alpha_g\rho_g u_g A)}{\partial x} = 0, \tag{2.6b}$$

$$\frac{\partial\left((\alpha_l\rho_l u_l + \alpha_g\rho_g u_g)A\right)}{\partial t} + \frac{\partial\left((\alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2 + p)A\right)}{\partial x} = p\frac{\partial A}{\partial x}, \tag{2.6c}$$

$$\frac{\partial A}{\partial t} = 0. \tag{2.6d}$$

Equation (2.6d) is trivial and it is only added to enable the eigenvalue analysis presented later. Notably, system (2.1) can be written in the conservative form while system (2.6) cannot be written in such a form due to the presence of the term $p\partial A/\partial x$ (in (2.6c), all terms concerning spatial derivatives of the state variables cannot be gathered in one single term and therefore the system (2.6) is non-conservative). Therefore, the area variation included in (2.6) significantly affects the solution, in particular the wave reflection pattern inside the domain of a pipe with a discontinuous area. For this type of cross-sectional characteristics, the term $p\partial A/\partial x$ becomes an impulsive force per unit length acting on a infinitesimally small spatial interval. This means that when a fluid particle passes a location with discontinuity in the area, it experiences an excessively large force (an impulsive force) for an infinitesimally short period of time that leads to a finite change in the momentum of the particle. The effect of such impulsive forces can not be captured appropriately by the classical finite-volume schemes. Therefore, other methods are needed to solve this system of equations.

In the following section, an eigenvalue analysis of the system (2.6) is carried out to explain the effect of a variable cross-sectional area on the solution.

2.2.2 Eigenvalue Analysis of the non-conservative DFM

To perform the eigenvalue analysis, $Q := [\alpha_g \quad u_l \quad p \quad A]^T$ is defined as the state variable vector and by inserting the closure laws into the PDEs (2.6), the system

in (2.6) can be rewritten in a compact form as follows:

$$J_1(Q)\frac{\partial Q}{\partial t} + J_2(Q)\frac{\partial Q}{\partial x} = 0, \tag{2.7}$$

where the concise representation of J_1 and J_2 (by partially inserting the closure laws) are as below

$$J_1(Q) = \begin{bmatrix} -\rho_l & 0 & 0 \\ \rho_g & 0 & 0 \\ \rho_g u_g - \rho_l u_l + \alpha_g \rho_g \dfrac{K\left((K-1)u_l + S\right)}{(1-K\alpha_g)^2} & \alpha_l \rho_l + \alpha_g \rho_g \dfrac{K\alpha_l}{(1-K\alpha_g)} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \dfrac{\alpha_l}{c_l^2} & 0 \\ \dfrac{\alpha_g}{c_g^2} & 0 \\ \dfrac{\alpha_l u_l}{c_l^2} + \dfrac{\alpha_g u_g}{c_g^2} & 0 \\ 0 & 1 \end{bmatrix},$$

$$J_2(Q) = \begin{bmatrix} -\rho_l u_l & \alpha_l \rho_l & \dfrac{\alpha_l u_l}{c_l^2} & \dfrac{\alpha_l \rho_l u_l}{A} \\ \rho_g \dfrac{u_g - K\alpha_g u_l}{(1-K\alpha_g)} & \alpha_g \rho_g \dfrac{K\alpha_l}{(1-K\alpha_g)} & \dfrac{\alpha_g u_g}{c_g^2} & \dfrac{\alpha_g \rho_g u_g}{A} \\ a_{31} & a_{32} & \dfrac{\alpha_l u_l^2}{c_l^2} + \dfrac{\alpha_g u_g^2}{c_g^2} + 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.8}$$

where

$$a_{31} = \rho_g u_g^2 - \rho_l u_l^2 + 2\alpha_g \rho_g u_g \frac{K\left((K-1)u_l + S\right)}{(1-K\alpha_g)^2},$$
$$a_{32} = 2\alpha_l \rho_l u_l + 2\alpha_g \rho_g u_g \frac{K\alpha_l}{(1-K\alpha_g)}.$$

It should be noted that although the area is a known variable, it is considered to be one of the states of the system to facilitate the analyses and to enable writing system (2.6) in the format of (2.7). This is the main reason for keeping the last trivial equation in system (2.6). For the DFM, the eigenvalues of the matrix $J_1^{-1}J_2$ have the following form [69]:

$$\lambda_1 = u_l + \omega, \quad \lambda_2 = u_g, \quad \lambda_3 = u_l - \omega, \quad \lambda_4 = 0, \tag{2.9}$$

where ω is the speed of sound in the mixture of the gas and the liquid [69]. In case of no-slip, i.e., $K = 1,\ S = 0$, after computing the eigenvalues and casting the results into the format of (2.9), the speed of sound in the mixture can be analytically written as follows known as Wood or Wallis speed of sound [56], [178]:

$$\omega = c_l c_g \sqrt{\frac{\rho_l \rho_g}{\bar{\rho}(\alpha_g \rho_l c_l^2 + \alpha_l \rho_g c_g^2)}}, \tag{2.10}$$

with $\bar{\rho} = \alpha_l \rho_l + \alpha_g \rho_g$.

In the case when slip occurs between the two phases, computing the analytical speed of sound in the mixture is complex, if not impossible. Thus, simplified surrogates for the speed of sound have been suggested; for instance the surrogate in [69] is introduced for cases with $\alpha_g \rho_g \ll \alpha_l \rho_l$ and $0 < \alpha_g < 1$ as below:

$$\omega \simeq \sqrt{\frac{p}{\alpha_g \rho_l (1 - K\alpha_g)}}. \tag{2.11}$$

Notably, although the gas and the liquid phase play a symmetric role in (2.6), the contribution of the phases are non-symmetric in the eigenvalues (2.9) due to the closure laws (2.2)-(2.5). For a detailed analysis of the eigenvalue problem of the conservative DFM, the reader may refer to [58]. The eigenvalue λ_4 shows that there is a stationary wave in the computational domain that becomes visible when the cross-sectional area is discontinuous and $\partial A/\partial x$ becomes closer to the impulse function.

Remark 2.1. *Equation* (2.11) *becomes ill-posed when* α_g *tends to zero or* $K\alpha_g \to 1$. *When* $K > 1$, *the singular point even occurs for* $\alpha_g < 1$. *In these cases, other surrogate formulations should be used [94], which has not been studied in this paper.*

Remark 2.2. *In some special cases, the speed of sound* ω *becomes very low, even less than* c_g, *in the presence of both phases. However, in the special application of drilling for oil and gas, phase velocities are still lower than* ω *and the flow is subsonic. Henceforth, we only consider subsonic flows.*

In general, dynamics of the problems studied in this paper are the perturbation dynamics with respect to the steady-state solution. Thus, finding the steady-state solution is the first step towards solving this type of problems.

2.2.3 Steady-state solution

For obtaining the analytical steady-state solution of system (2.6), the partial derivatives of the variables with respect to time is set to zero and the following system of equations should be solved:

$$\frac{\partial(\alpha_l \rho_l u_l A)}{\partial x} = 0, \tag{2.12a}$$

$$\frac{\partial(\alpha_g \rho_g u_g A)}{\partial x} = 0, \tag{2.12b}$$

$$\frac{\partial\Big((\alpha_l \rho_l u_l^2 + \alpha_g \rho_g u_g^2 + p)A\Big)}{\partial x} = p\frac{\partial A}{\partial x}. \tag{2.12c}$$

By embedding (2.12a) and (2.12b) into (2.12c), simplified governing equations are obtained as in (2.13)

$$\frac{\partial(\alpha_l \rho_l u_l A)}{\partial x} = 0, \tag{2.13a}$$

$$\frac{\partial(\alpha_g \rho_g u_g A)}{\partial x} = 0, \tag{2.13b}$$

$$\alpha_l \rho_l u_l A\frac{\partial u_l}{\partial x} + \alpha_g \rho_g u_g A\frac{\partial u_g}{\partial x} + A\frac{\partial p}{\partial x} = 0. \tag{2.13c}$$

After inserting the closure laws and considering the slip law (2.3), the steady-state solution of the new state variable $W := [u_l \quad u_g \quad \alpha_g \quad p]^T$ is governed by the system of equations:

$$M(W,x)\frac{\partial W}{\partial x} = E(W,x), \tag{2.14}$$

where the concise representation of M and E (by partially inserting the closure laws) are as below

$$M(W,x) = \begin{bmatrix} -K(1-\alpha_g) & 1-K\alpha_g & K(u_l-u_g) & 0 \\ (1-\alpha_g)(\frac{p-p_0}{c_l^2}+\rho_0)A & 0 & -(\frac{p-p_0}{c_l^2}+\rho_0)u_l A & \frac{(1-\alpha_g)u_l A}{c_l^2} \\ 0 & \alpha_g\frac{p}{c_g^2}A & \frac{p}{c_g^2}u_g A & \alpha_g\frac{u_g A}{c_g^2} \\ (1-\alpha_g)(\frac{p-p_0}{c_l^2}+\rho_0)u_l A & \alpha_g\frac{p}{c_g^2}u_g A & 0 & A \end{bmatrix},$$

$$E(W,x) = \begin{bmatrix} 0 \\ -(1-\alpha_g)(\frac{p-p_0}{c_l^2}+\rho_0)u_l\frac{\partial A}{\partial x} \\ -\alpha_g\frac{p}{c_g^2}u_g\frac{\partial A}{\partial x} \\ -\Big((1-\alpha_g)(\frac{p-p_0}{c_l^2}+\rho_0)u_l^2 + \alpha_g\frac{p}{c_g^2}u_g^2\Big)\frac{\partial A}{\partial x} \end{bmatrix}. \tag{2.15}$$

Notably, the term $\partial A/\partial x$ represents an impulsive term at the discontinuities of $A(x)$, which leads to discontinuities in the steady-state solution of W. To solve this system for discontinuous function $A(x)$, left- or right-continuity of the function should be specified to be able to define the area at any location.

The initial condition for the simulations in the presence of area variation is the solution of (2.14)-(2.15) unless otherwise mentioned. If a system starts from its unique steady-state solution, the numerical solution should remain on the same solution afterwards. Therefore, a significant discrepancy between the numerical simulation and the steady-state initial condition reveals the poor performance of the scheme, which may be hard to diagnose in dynamical simulations. Thus, this test is a powerful measure for assessing the necessary performance of a scheme, i.e., predicting the correct steady-state solution.

Equations (2.14)-(2.15) represent a two-point boundary value problem due to the boundary conditions specified at both ends, which are detailed in the next section.

2.2.4 Boundary conditions

As three PDEs are involved in the system of (2.14), three physical boundary conditions have to be specified. For subsonic flow, it is typical to set a specific mass flow rate of the liquid and the gas at the left boundary (at $x = 0$) and a pressure at the right boundary (at $x = L$) [69], [71]. Henceforth, the mass flow rate of the liquid and gas are, respectively, denoted by $\dot{m}_l$ and $\dot{m}_g$ (i.e., $\dot{m}_l := \alpha_l \rho_l u_l A$ and $\dot{m}_g := \alpha_g \rho_g u_g A$), and the pressure at the right boundary is denoted by p_R.

However, for finding the numerical solution of system (2.6), all conservative variables should be prescribed at the boundaries. Since the number of conservative variables at each boundary is more than the number of physical boundary conditions, additional conditions at the boundary are required to find the unique solution for the boundary variables. For instance, at the right boundary, only pressure is prescribed and other variables should be obtained by some compatibility equations. By following the approach described in [71], characteristic boundary conditions are combined with the physical boundary conditions in order to fulfill all the necessary conditions at the boundaries. The characteristic boundary equations can be found in [71]. Under the assumption of constant area only at the boundaries, the characteristic boundary equation corresponding to the pressure wave propagating in the downstream direction $\lambda_1 = u_l + \omega$ reads as:

$$\frac{\mathrm{d}}{\mathrm{d}t}p + \rho_l \omega (u_g - u_l)\frac{\mathrm{d}}{\mathrm{d}t}\alpha_g - \rho_l \alpha_l (u_g - u_l - \omega)\frac{\mathrm{d}}{\mathrm{d}t}u_l = 0, \qquad \text{with} \quad \frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + (u_l + \omega)\frac{\partial}{\partial x}. \tag{2.16}$$

Similarly, for the gas volume wave $\lambda_2 = u_g$, we have:

$$\frac{\mathrm{d}}{\mathrm{d}t}p + \frac{p}{\alpha_g(1-K\alpha_g)}\frac{\mathrm{d}}{\mathrm{d}t}\alpha_g = 0, \quad \text{with} \quad \frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + u_g\frac{\partial}{\partial x}. \tag{2.17}$$

Finally, for the pressure wave propagating in the upstream direction $\lambda_3 = u_l - \omega$, we have:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}p - \rho_l\omega(u_g-u_l)\frac{\mathrm{d}}{\mathrm{d}t}\alpha_g - \rho_l\alpha_l(u_g-u_l+\omega)\frac{\mathrm{d}}{\mathrm{d}t}u_l = 0,\\ \text{with} \quad \frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + (u_l-\omega)\frac{\partial}{\partial x}.\end{aligned} \tag{2.18}$$

The discrete version of equations (2.16) and (2.17) are solved at the right boundary and discrete version of equation (2.18) is solved at the left boundary.

Remark 2.3. *Before going through the numerical solvers, it should be noted that, in this paper, the function $A(x)$ is piecewise continuous and it is discretized over the spatial domain. After such discretization, $A(x)$ becomes piecewise constant as it is constant within each grid cell and the discontinuities occur only at the interfaces. Then, in the case of discontinuous area, wherever the computation of $\partial A/\partial x$ is required, for instance in the steady-state calculations, the spatial derivative of $A(x)$ is approximated by the finite difference method.*

Remark 2.4. *Some existing methods to deal with the variable cross-section are based on adapting the Rusanov scheme [105]. For the sake of a fair comparison, we also consider the Rusanov scheme as our numerical scheme. However, the method introduced in this paper is a universal modification that can be used along with any numerical scheme such as the AUSMV scheme [69].*

2.3 Numerical solvers for the DFM with piecewise continuous cross-section

In this section, different approaches are presented to deal with the non-conservative term in the DFM; some of which are proposed in this paper and some are extensions of existing methods for (2.6). The effects of variable cross-sectional area in the DFM and Euler equations are similar; a stationary wave is added to the existing waves in both cases. Therefore, the strategies introduced in different works such as [57], [72], [103], [169] to deal with non-conservative terms in the Euler equations with area variation are the main source of inspiration for this work.

For the sake of completeness, some terminologies widely used in the context of finite-volume method are introduced beforehand. For a general first-order PDE of the form

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\Big(f(u)\Big) = S(u,t,x), \text{ with } t \in \mathrm{I} = [0,\mathrm{T}],\ x \in \Omega = [0,\mathrm{L}], \tag{2.19}$$

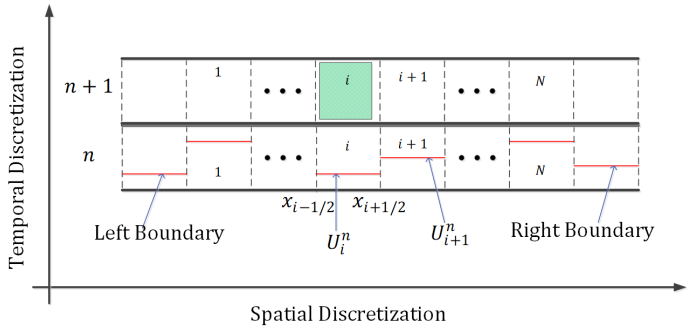

Figure 2.2: An illustration of the finite-volume spatial and temporal discretization; green block: the desired solution at the next time step, red lines: constant solutions within each grid cell at the current time step.

u is called the conservative variable, f is called the mathematical flux function and S is the source term. Furthermore, primitive variables with physical interpretation are defined, denoted by v. For instance, for the system (2.1), the conservative variables are $u = [\alpha_l\rho_l \quad \alpha_g\rho_g \quad \alpha_l\rho_l u_l + \alpha_g\rho_g u_g]^T$ and primitive variables are any combination of three independent variables, such as $v = [\alpha_g \quad u_l \quad p]^T$. Finite-volume discretization is commonly employed to solve such PDEs by a discretization of the spatial computational domain Ω and the temporal computational domain I, as shown in Figure 2.2. Assume that we are interested in the solution at the i-th spatial grid cell at the time step $n+1$ (the green block in Figure 2.2). First-order Godunov-type schemes numerically solve (2.19) by

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F(U_i^n, U_{i+1}^n) - F(U_{i-1}^n, U_i^n)\right) + \Delta t\ S(U_i^n, t^n, x_i), \qquad (2.20)$$

where U_i^n is the spatial average of the conservative variable u over i-th cell at the time instant $t^n = n\Delta t$, schematically shown at different grid cells by red lines in Figure 2.2. Similarly, V_i^n is the spatial average of the primitive variables v, which will be used later. Also, Δt and Δx refer to temporal and spatial discretization step sizes, respectively.

The numerical flux function F is a scheme-dependent function of the conservative variables. The classical Rusanov scheme [171] for the system (2.6) employs a flux function as below:

$$F(U_i^n, U_{i+1}^n) = \frac{f(U_{i+1}^n) + f(U_i^n)}{2} - \lambda_{i+1/2}^n (U_{i+1}^n - U_i^n), \qquad (2.21)$$

with

$$f = [\alpha_l\rho_l u_l A, \quad \alpha_g\rho_g u_g A, \quad (\alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2 + p)A]^T, \qquad (2.22a)$$

$$U = [\alpha_l \rho_l A, \quad \alpha_g \rho_g A, \quad (\alpha_l \rho_l u_l + \alpha_g \rho_g u_g) A]^T, \tag{2.22b}$$

and

$$\lambda^n_{i+1/2} = \frac{1}{2} \max(\{|u^n_l| + \omega^n, |u^n_g|\}_{i+1}, \{|u^n_l| + \omega^n, |u^n_g|\}_i), \tag{2.23}$$

where the operator "max"gives the maximum value of its arguments, and $\lambda^n_{i+1/2}$ is the half of maximum local eigenvalue of (2.9) computed at the left-hand and right-hand side of the interface $x_{i+1/2}$ at the time instant t^n. Below, various methods for dealing with the non-conservative term are presented.

2.3.1 First approach: source term approximation

The first approach involves considering the non-conservative term in (2.6), $p\partial A/\partial x$, as a source term. In other words in (2.20), the term $S(U^n_i, t^n, x_i)$ is taken equal to an approximation of $(p\partial A/\partial x)^n_i$, with the conservative variable vector and mathematical flux function as in (2.22). When the area is discontinuous, this source term approaches an impulsive force. Different approaches such as the one in [169] have tried to approximate this term in different ways.

2.3.2 Second approach: modified Rusanov scheme

The second approach is adopted from [49] for dealing with the non-conservative Euler equations by modifying the Rusanov scheme. Clain and Rochette in [49] adapted this scheme to enforce the numerical solution to be steady-state-preserving for zero flow for the system of the non-conservative Euler equations. The extension of the approach in [49] to the DFM is as below:

$$U^{n+1}_i = U^n_i - \frac{\Delta t}{\Delta x} \left(\left(F(U^n_i, U^n_{i+1}) + G^{-,n}_{i+1/2} \right) - \left(F(U^n_{i-1}, U^n_i) + G^{+,n}_{i-1/2} \right) \right), \tag{2.24}$$

with

$$\begin{aligned} F(U^n_i, U^n_{i+1}) = \frac{f(U^n_{i+1}) + f(U^n_i)}{2} - \lambda^n_{i+1/2} A_{i+1/2} \times \\ \begin{bmatrix} (\alpha_l \rho_l)^n_{i+1} - (\alpha_l \rho_l)^n_i \\ (\alpha_g \rho_g)^n_{i+1} - (\alpha_g \rho_g)^n_i \\ (\alpha_l \rho_l u_l + \alpha_g \rho_g u_g)^n_{i+1} - (\alpha_l \rho_l u_l + \alpha_g \rho_g u_g)^n_i \end{bmatrix}, \\ A_{i+1/2} = \max(A_{i+1}, A_i), \end{aligned} \tag{2.25}$$

where A_i is the cross-sectional area at the center of i-th grid cell (it is assumed that the area is constant within each grid cell and only changes at the interfaces)

and

$$G^{-,n}_{i+1/2} := -\frac{A_{i+1} - A_i}{2} \begin{bmatrix} 0 \\ 0 \\ p^n_i \end{bmatrix}, G^{+,n}_{i-1/2} := \frac{A_i - A_{i-1}}{2} \begin{bmatrix} 0 \\ 0 \\ p^n_i \end{bmatrix}. \tag{2.26}$$

In other words, the third entry of the following vector which appears in (2.24),

$$\frac{-1}{\Delta x}(G^{-,n}_{i+1/2} - G^{+,n}_{i-1/2}) = \frac{A_{i+1} - A_{i-1}}{2\Delta x} \begin{bmatrix} 0 \\ 0 \\ p^n_i \end{bmatrix}$$

approximates the term $p\partial A/\partial x$. Clain and Rochette in [49] prove that the modified scheme is well-balanced in the case of zero flow.

2.3.3 Third approach: novel, modified Rusanov scheme

The third method is inspired by the idea behind the second approach, by applying further modifications to the scheme in (2.25). The new modification, proposed here, is motivated by the need to capture the steady-state solution with a higher accuracy. In the second approach, when evaluated at the steady-state solution at the presence of flow inside the domain, we obtain $U^{n+1}_i \neq U^n_i$, and therefore, the numerical solution deviates from the actual steady-state solution. This approach is motivated by enforcing the last entry of U^n_i, which is $\Big((\alpha_l\rho_l u_l + \alpha_g\rho_g u_g)A\Big)^n_i$, to be well-balanced. By using the fact that the mass flow rates of both phases are constant at every location during a steady-state solution, the new modification of the scheme is introduced by the flux function as given below:

$$F(U^n_i, U^n_{i+1})_{i+1/2} = \frac{f(U^n_{i+1}) + f(U^n_i)}{2} - \lambda^n_{i+1/2} A_{i+1/2} \times \begin{bmatrix} (\alpha_l\rho_l)^n_{i+1} - (\alpha_l\rho_l)^n_i \\ (\alpha_g\rho_g)^n_{i+1} - (\alpha_g\rho_g)^n_i \\ \dfrac{(\alpha_l\rho_l u_l A + \alpha_g\rho_g u_g A)^n_{i+1}}{A_{i+1/2}} - \dfrac{(\alpha_l\rho_l u_l A + \alpha_g\rho_g u_g A)^n_i}{A_{i+1/2}} \end{bmatrix}, \tag{2.27}$$

with the same f and U mentioned in the previous section. In this case, while starting from the steady-state solution, the third entry of the flux function F computed by (2.27), i.e., $(\alpha_l\rho_l u_l A + \alpha_g\rho_g u_g A)^n_{i+1} - (\alpha_l\rho_l u_l A + \alpha_g\rho_g u_g A)^n_i$, becomes zero; therefore, at least the deviation of the third entry of conservative variable at the first time-step is zero and this modification outperforms the second approach. Similar to the second approach, this scheme reverts to the original Rusanov scheme when there is no change in the cross-sectional area.

2.3.4 Fourth approach: model-based modification

The next novel method, proposed here, is obtained through modifying the conservative variables before calculating the flux functions that allows one to treat the non-conservative term in an indirect way, regardless of the type of the Godunov scheme. The underlying idea is inspired by [57], [103] and consists of defining the solution as follows:

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F(U_i^n, U_{i+1}^{*,n}) - F(U_{i-1}^{*,n}, U_i^n)\right), \tag{2.28}$$

where U_{i+1}^* and U_{i-1}^* are the modified conservative variables, which are henceforth called *starred* conservative variables. All conservative variables U and the mathematical flux function f (which will be used in the computation of F) in this approach are defined according to (2.1), not (2.6), as the effect of area is included in the starred variables, meaning that

$$\begin{aligned} U &= [\alpha_l\rho_l, \quad \alpha_g\rho_g, \quad \alpha_l\rho_l u_l + \alpha_g\rho_g u_g]^T, \\ f &= [\alpha_l\rho_l u_l, \quad \alpha_g\rho_g u_g, \quad \alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2 + p]^T. \end{aligned}$$

First of all, an explanation to this approach is provided, and then, the computational steps for computing U_{i+1}^* and U_{i-1}^* at the i-th spatial grid cell is established to obtain the numerical solution by (2.28).

As mentioned in the eigenvalue analysis in Section 2.2.2, a stationary time-independent wave (corresponding to λ_4 in (2.9)) lies in the system that should be captured by (2.28). To this end, the method proposed in this section is composed of two steps. In the first step, the effect of the stationary wave generated by the non-conservative source term is captured. As the stationary waves are time-independent, steady-state model is employed to reconstruct this type of waves. In the second step, the effect of the non-stationary waves is taken into account. The latter waves are governed by (2.1) as the effect of the stationary wave is already considered in the first step.

This approach is visualized in Figure 2.3, where the first step mentioned above is denoted by ① and the second step is denoted by ②. Assume that U_i^{n+1} is required (the green block in Figure 2.3). This grid cell is surrounded by two interfaces at $x_{i+1/2}$ and $x_{i-1/2}$. The conservative variables at the right-hand and left-hand side of the interface $i+1/2$ are shown by U_{i+1} and U_i, respectively, in Figure 2.3. Similarly, the left- and right-hand side values for the interface $i-1/2$ can be defined. In this approach, the conservative and primitive variables affecting the solution in the $i-$th cell, U_{i+1} and U_{i-1}, are modified such that these variables contain the effect of the stationary waves at the inlet and outlet of the cell. The starred values are denoted by U_{i+1}^* and U_{i-1}^* in Figure 2.3 at level ①. Now, as the system has a constant area only over the i-th grid cell and its neighboring cells, classical finite-volume discretization can be applied on (2.1) at level ① to obtain the solution at the i-th spatial grid cell. It should be

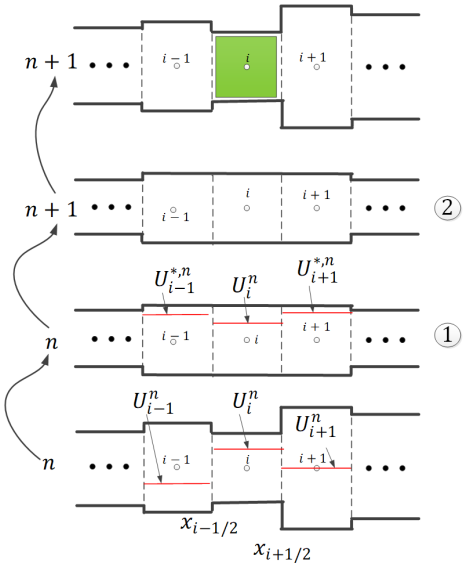

Figure 2.3: Finite-volume modification for the fourth approach, model-based modification.

noted that the area is assumed to be constant over only these three grid-cells while the area of other grid cells do not affect the solution of the i-th grid cell at the current time step $n+1$. Therefore, the solution obtained by (2.28) contains both the effect of the stationary and non-stationary waves.

After providing the intuition how this method works, the framework for finding the starred values is established here. As the stationary waves are time-independent, the steady-state solution of the non-conservative system (2.6) is exploited to find algebraic constraints that capture the stationary waves. Recalling steady-state equations in (2.13), we have

$$\begin{aligned} \frac{\partial(\alpha_l \rho_l u_l A)}{\partial x} &= 0, \\ \frac{\partial(\alpha_g \rho_g u_g A)}{\partial x} &= 0, \\ A(\alpha_l \rho_l u_l \frac{\partial u_l}{\partial x} + \alpha_g \rho_g u_g \frac{\partial u_g}{\partial x} + \frac{\partial p}{\partial x}) &= 0. \end{aligned}$$

To find an algebraic relation over the area discontinuity, (2.13) should be integrated over the interface. However, (2.13c) is difficult to be integrated analytically over the spatial domain and needs more investigation. This differential equation can be simplified, as shown in the following Lemma 2.5.

Lemma 2.5. *The equation* (2.13c) *is equivalent to the following equation if both phases are present in the system:*

$$\begin{aligned}\bar{m}_g(u_g\frac{\partial u_g}{\partial x}+\frac{1}{\rho_g}\frac{\partial p}{\partial x})+\bar{m}_l(u_l\frac{\partial u_l}{\partial x}+\frac{1}{\rho_l}\frac{\partial p}{\partial x})+\\ (u_l-u_g)(\bar{m}_g\alpha_l\frac{\partial u_g}{\partial x}-\bar{m}_l\alpha_g\frac{\partial u_l}{\partial x})=0,\end{aligned} \tag{2.29}$$

where $\bar{m}_l = A\alpha_l\rho_l u_l$ *and* $\bar{m}_g = A\alpha_g\rho_g u_g$.

Proof. From (2.13c), we have:

$$\bar{m}_g\frac{\partial u_g}{\partial x}+\bar{m}_l\frac{\partial u_l}{\partial x}+A\frac{\partial p}{\partial x}=0.$$

As variables $\alpha_l u_l$ and $\alpha_g u_g$ are not constantly zero in general due to the presence of both phases, we can multiply the above equation by these two variables:

$$\begin{aligned}\alpha_l u_l(\bar{m}_g\frac{\partial u_g}{\partial x}+\bar{m}_l\frac{\partial u_l}{\partial x}+A\frac{\partial p}{\partial x})=0,\\ \alpha_g u_g(\bar{m}_g\frac{\partial u_g}{\partial x}+\bar{m}_l\frac{\partial u_l}{\partial x}+A\frac{\partial p}{\partial x})=0.\end{aligned}$$

Summation of the above equations and using the identity $\alpha_g+\alpha_l=1$ leads to:

$$\begin{aligned}\bar{m}_g\alpha_l u_l\frac{\partial u_g}{\partial x}+\bar{m}_l(1-\alpha_g)u_l\frac{\partial u_l}{\partial x}+A\alpha_l u_l\frac{\partial p}{\partial x}+\bar{m}_g(1-\alpha_l)u_g\frac{\partial u_g}{\partial x}+\bar{m}_l\alpha_g u_g\frac{\partial u_l}{\partial x}+\\ A\alpha_g u_g\frac{\partial p}{\partial x}=0.\end{aligned}$$

Rewriting the above relation leads to the claimed equation (2.29). □

The relation in (2.29) cannot be simplified further unless additional assumptions are made. In the following sections, we analyze the DFM in two categories.

2.3.4.1 DFM without slip

Assume that there is no slip between the two phases, i.e., $K=1$ and $S=0$ in (2.3) and subsequently

$$\hat{u} := u_l = u_g. \tag{2.30}$$

In this case, as $u_l - u_g = 0$ in (2.29), system (2.13) changes to:

$$\begin{aligned}\frac{\partial(\alpha_l\rho_l\hat{u}A)}{\partial x}=0,\\ \frac{\partial(\alpha_g\rho_g\hat{u}A)}{\partial x}=0,\end{aligned}$$

$$A\alpha_l\rho_l\hat{u}(\hat{u}\frac{\partial\hat{u}}{\partial x}+\frac{1}{\rho_l}\frac{\partial p}{\partial x})+A\alpha_g\rho_g\hat{u}(\hat{u}\frac{\partial\hat{u}}{\partial x}+\frac{1}{\rho_g}\frac{\partial p}{\partial x})=0.$$

As $\partial p/\partial x = c_l^2\partial\rho_l/\partial x = c_g^2\partial\rho_g/\partial x$, the above equation is integrable over x. Therefore, over the interface, the following set of functions are set to be constant:

$$\alpha_l\rho_l\hat{u}A = \text{constant}, \tag{2.31a}$$
$$\alpha_g\rho_g\hat{u}A = \text{constant}, \tag{2.31b}$$
$$\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2}+c_l^2\ln\rho_l)+\alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2}+c_g^2\ln\rho_g) = \text{constant}. \tag{2.31c}$$

Equations (2.31a)-(2.31b) represent the mass flow continuity at the interface. Equation (2.31c) governs the rate of energy exchange at the interface. In other words, the term $\hat{u}^2/2$ is the kinetic energy per unit mass of each phase. Also, $c_{l,g}^2\ln\rho_{l,g}$ is the potential energy of the compressible liquid and gas per unit mass. Therefore, at the interface, the mass and energy continuity should be preserved.

For instance, for finding $U_{M_1}^*$ from U_{M_1}, we should solve:

$$(\alpha_l\rho_l\hat{u})_{M_1}^* A_{M_2} = (\alpha_l\rho_l\hat{u}A)_{M_1}, \tag{2.32a}$$
$$(\alpha_g\rho_g\hat{u})_{M_1}^* A_{M_2} = (\alpha_g\rho_g\hat{u}A)_{M_1}, \tag{2.32b}$$
$$\left(\alpha_l\rho_l\hat{u}(\frac{\hat{u}^2}{2}+c_l^2\ln\rho_l)+\alpha_g\rho_g\hat{u}(\frac{\hat{u}^2}{2}+c_g^2\ln\rho_g)\right)_{M_1}^* A_{M_2} = \left(\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2}+c_l^2\ln\rho_l)+\alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2}+c_g^2\ln\rho_g)\right)_{M_1}, \tag{2.32c}$$

where M_1 and M_2 refer to two neighboring cells, see Algorithm 1 for more details.

However, when the slip law is not discarded, finding an algebraic relation becomes hard as the third term $(u_l-u_g)(\bar{m}_g\alpha_l\partial u_g/\partial x-\bar{m}_l\alpha_g\partial u_l/\partial x)$ in (2.29) is not negligible. In order to extend the applicability of the fourth approach for cases with slip, extra physical assumptions should be made, which is dealt with below.

2.3.4.2 DFM with slip

In this case, the analytical integration of the momentum equation (2.13c) is challenging. In [18], the authors claim that the DFM cannot be endowed with an entropy pair unless restrictive assumptions such as no-slip are made. This fact that the integration of the momentum equation is challenging might be related to the lack of an entropy inequality. Alternatively, here, we try to impose some physical assumptions to be able to integrate the momentum equation.

It is assumed that at the interfaces of area variation, the ratio of volume and mass composition of the mixture do not change (in other words, still mass

continuity is maintained). This assumption is valid when one of the phases is dominant in volumetric sense in the pipe or the area varies smoothly. Otherwise, in situations where both phases occupy the space rather equally and the area variation is sudden, this assumption is less accurate. These two assumptions result in constant volumetric fraction and mass fraction of each phase at the interface, respectively. Meaning that at each interface, the following conditions hold:

$$\frac{\alpha_l}{\alpha_g+\alpha_l} = \text{constant} \Longrightarrow \alpha_{l,g} = \text{constant}, \tag{2.33a}$$

$$\frac{\alpha_l\rho_l}{\alpha_l\rho_l+\alpha_g\rho_g} = \text{constant} \Longrightarrow \frac{\rho_l}{\rho_g} = \text{constant}. \tag{2.33b}$$

Then (2.13c) is rewritten as below:

$$A\rho_g(\alpha_l\frac{\rho_l}{\rho_g}u_l\frac{\partial u_l}{\partial x}+\alpha_g u_g\frac{\partial u_g}{\partial x}+\frac{1}{\rho_g}\frac{\partial p}{\partial x}) = 0 \overset{A\rho_g\neq 0}{\Longrightarrow}$$

$$\alpha_l\frac{\rho_l}{\rho_g}\frac{u_l^2}{2}+\alpha_g\frac{u_g^2}{2}+c_g^2\ln\rho_g = \text{constant}.$$

Finally, the set of algebraic constraints under the set of assumptions mentioned in (2.33) for $\rho_g > 0$ is defined as:

$$\alpha_l\rho_l u_l A = \text{constant}, \tag{2.34a}$$

$$\alpha_g\rho_g u_g A = \text{constant}, \tag{2.34b}$$

$$\alpha_l\frac{\rho_l}{\rho_g}\frac{u_l^2}{2}+\alpha_g\frac{u_g^2}{2}+c_g^2\ln\rho_g = \text{constant}. \tag{2.34c}$$

Still the mass continuity exactly helds. The kinetic energy of both phases and the potential energy of the gaseous phase can still be detected in (2.34c). This is due to the assumption that we consider that one of the phases is dominant in space. Therefore, the potential energy due to the expansion of the dominant phase, here gas, is only reflected in the algebraic relation. Moreover, due to the assumption of the prevalence of one phase, the area has been disappeared from (2.34c). We emphasize again if both phases are rather equally present in the pipe and the area variation is sudden, the relation (2.34c) is not accurate.

Similar to the previous set of assumptions, for finding $U_{M_1}^*$ from U_{M_1}, the following set of algebraic equations should be solved:

$$(\alpha_l\rho_l u_l)_{M_1}^* A_{M_2} = (\alpha_l\rho_l u_l A)_{M_1}, \tag{2.35a}$$

$$(\alpha_g\rho_g u_g)_{M_1}^* A_{M_2} = (\alpha_g\rho_g u_g A)_{M_1}, \tag{2.35b}$$

$$(\alpha_l\frac{\rho_l}{\rho_g}\frac{u_l^2}{2}+\alpha_g\frac{u_g^2}{2}+c_g^2\ln\rho_g)_{M_1}^* = (\alpha_l\frac{\rho_l}{\rho_g}\frac{u_l^2}{2}+\alpha_g\frac{u_g^2}{2}+c_g^2\ln\rho_g)_{M_1}, \tag{2.35c}$$

Algorithm 1: Fourth approach: model-based modification

Input: $U_{i-1}^n, U_i^n, U_{i+1}^n, A_{i-1}, A_i, A_{i+1}$

Output: $U_i^{n+1} = \left([\alpha_l\rho_l \quad \alpha_g\rho_g \quad \alpha_l\rho_l u_l + \alpha_g\rho_g u_g]^T\right)_i^{n+1}$

Compute primitive variables, $V_{i-1}^n, V_i^n, V_{i+1}^n$, from conservative variables, $U_{i-1}^n, U_i^n, U_{i+1}^n$ [169],
if $A_{i+1} \neq A_i$ **then**
  Solve (2.32) or (2.35), based on the assumption taken in Section 2.3.4, with $M_1 = i+1$ and $M_2 = i$, obtain $V_{i+1}^{*,n}$ and then $U_{i+1}^{*,n}$
else
  $U_{i+1}^{*,n} = U_{i+1}^n$
end if
if $A_i \neq A_{i-1}$ **then**
  Solve (2.32) or (2.35), based on the assumption taken in Section 2.3.4, with $M_1 = i-1$ and $M_2 = i$, and obtain $V_{i-1}^{*,n}$ and then $U_{i-1}^{*,n}$,
else
  $U_{i-1}^{*,n} = U_{i-1}^n$
end if
Compute U_i^{n+1} via (2.28).

where M_1 and M_2 refer to the neighboring cells, see Algorithm 1 for more details.

All steps involved in the fourth approach are summarized in Algorithm 1. Now, a justification on the performance of this approach is presented in the following claims.

Theorem 2.6. *For any given U_i^0 in $x_i \in [0, L]$ that satisfies* (2.13)*, then $U_i^n = U_i^0 \ \forall \ n \in \mathbb{N}$, if no-slip condition* (2.30) *is assumed to obtain the solution using Algorithm 1.*

Proof. Starting from a steady-state solution U_i^0, according to (2.13) and using (2.31), we have:

$$(\alpha_l\rho_l\hat{u}A)_{M_2} = (\alpha_l\rho_l\hat{u}A)_{M_1}, \tag{2.36a}$$

$$(\alpha_g\rho_g\hat{u}A)_{M_2} = (\alpha_g\rho_g\hat{u}A)_{M_1}, \tag{2.36b}$$

$$\begin{aligned}&\left(\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)_{M_2} = \\ &\left(\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)_{M_1}.\end{aligned} \tag{2.36c}$$

Comparing equations (2.36) with (2.32) reveals that $U_{M_1}^* = U_{M_2}$. Using this

property in (2.28), by replacing M_1 and M_2 as described in Algorithm 1 yields

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F(U_i^n, U_i^n) - F(U_i^n, U_i^n)\right) = U_i^n. \tag{2.37}$$

Solving (2.37) recursively yields $U_i^n = U_i^0$. Thus, the proposed scheme captures the steady-state solution exactly. For the case with slip and assumptions (2.33), the scheme captures the steady-state solution only approximately. □

Theorem 2.7. *In case of constant cross-sectional area, i.e., $\partial A/\partial x = 0$, the solution U_i^n obtained from* (2.28) *is equal to that obtained from* (2.20) *if no-slip condition* (2.30) *is assumed.*

Proof. For the case of constant area over two neighboring cells and assuming no-slip condition, (2.32) reduces to

$$\begin{aligned}
&(\alpha_l\rho_l\hat{u})^*_{M_1} = (\alpha_l\rho_l\hat{u})_{M_1},\\
&(\alpha_g\rho_g\hat{u})^*_{M_1} = (\alpha_g\rho_g\hat{u})_{M_1},\\
&\left(\alpha_l\rho_l\hat{u}(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)^*_{M_1} =\\
&\left(\alpha_l\rho_l\hat{u}(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)_{M_1},
\end{aligned}$$

where a candidate solution would be $U^*_{M_1} = U_{M_1}$. Then, by following Algorithm 1, (2.28) reduces to

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F(U_i^n, U_{i+1}^n) - F(U_{i-1}^n, U_i^n)\right).$$

This is similar to (2.20) with zero source terms (since area is constant and $\partial A/\partial x = 0$) and the modified scheme reverts back to the classical scheme. This feature is reflected in the Algorithm 1 to accelerate the numerical solution. Again we should mention that for the case with slip and assumptions (2.33), the scheme recovers the classical scheme only approximately when the area is constant. □

Remark 2.8. *After finding the starred values through solving the algebraic constraints, the numerical solution of (2.6) can be computed. If the starred-values are not found, one cannot compute the numerical solution by this method and other methods should be followed. Since these constraints exploit the steady-state equation (2.13), there is the possibility that these constraints are not satisfied either when starting at initial conditions far from the steady-state solution or when abrupt perturbations occur inside the domain. Notably, the algebraic constraints (2.32) and (2.35) may also have multiple solutions. Although the assumption of starting close to the steady-state solution is restrictive, there are many applications in the industry for which the analysis of perturbations with respect to*

the steady-state solution is important. In addition, studying the perturbations of steady-state solution of different systems has been the subject of many other studies [159], [161]. Moreover, this work can be the first step towards solving the non-conservative DFM by understanding the restrictions of the current approach. For a more in-depth discussion on features of the solution of the algebraic constraints, refer to Appendix A at the end of this paper, where the procedure to choose the feasible solution in case of multiple solutions is also explained.

In the following section, numerical results associated to each approach are discussed.

2.4 Numerical Results

Numerical results in this section are divided into five categories. First, in order to find the best scheme to solve system (2.6), the methods introduced in Section 2.3 have tested their ability to preserve the steady-state solution of system (2.6) with a piecewise constant cross-sectional area. Second, the most accurate approach, in the sense of capturing the steady-state solution, is used to compute the solution of (2.6) by imposing no variation in cross-sectional area in order to compare its performance with the solution obtained from the classical scheme of (2.21) applied to (2.1). Third, the selected approach has been tested for a transient flow modeling in a horizontal pipe with piecewise constant cross-sectional area. Fourth, the method is tested on a piecewise continuous cross-sectional area, both for capturing the steady-state solution and performing transient simulation. Finally, an error convergence study is performed.

The values of the parameters involved in system (2.6) are summarized in Table 2.1. The steady-state solution of (2.14) is acquired by the `bvp4c` solver of `MATLAB`. This solver approximates the solution to (2.13) in an iterative way while considering the boundary values at both ends of the computational domain [156].

Remark 2.9. *In Sections 2.4.1 and 2.4.2 for the fourth approach, lines* $2, 4-7, 9-11$ *of Algorithm 1 are ignored to test the performance of the new model-based modified scheme to automatically recover the classical scheme in locations where the area is constant. After becoming assure of the performance at constant area locations, the entire Algorithm 1 is used in other sections.*

Remark 2.10. *As mentioned before, to make fair comparisons between this approach and the second and the third approaches introduced in previous sections,*

Table 2.1: Test case parameters.

ρ_0	p_0	$\dot{m}_l$	$\dot{m}_g$	c_g	c_l
1000 kg/m^3	1 bar	0.3 kg/s	0.003 kg/s	316 m/s	1000 m/s

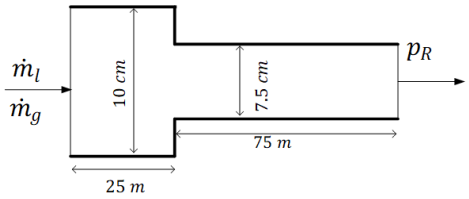

Figure 2.4: Configuration of the computational domain.

we use the Rusanov scheme as the case study. However, the modification introduced in the fourth approach can be applied to other finite-volume schemes in a similar manner; only the numerical flux function F in (2.21) should be changed according to the finite-volume scheme.

Remark 2.11. *The Rusanov scheme is subject to the CFL condition*

$$\Delta t = CFL \frac{\Delta x}{\max(|\lambda_1|, |\lambda_2|, |\lambda_3|, |\lambda_4|)}, \tag{2.38}$$

where $\lambda_i, i \in \{1, 2, 3, 4\}$ are given by (2.9). For all simulations in this section, we estimate $\max(|\lambda_1|, |\lambda_2|, |\lambda_3|, |\lambda_4|) \approx c_l$ and set $CFL = 1$. Then, according to the chosen Δx, the temporal discretization Δt is specified.

Remark 2.12. *To the best of authors' knowledge, no Riemann solution for the DFM with variable cross-sectional area has been published. Before doing any transient simulations, the performance of the scheme at the steady state is evaluated. For transients, the performance of the scheme is assessed against the classical Rusanov scheme for the case of constant cross sectional area along the pipe. For a reference solution for piecewise constant area, refer to Appendix B. For a general piecewise continuous cross-sectional area, no reference solution exists for transient simulations.*

2.4.1 Preservation of the steady-state solution

This section is dedicated to check the steady-state preservation of the numerical approaches proposed in Section 2.3. Since for the set of algebraic relations (2.31) in the fourth approach, no slip between the phases is considered, i.e., $K = 1$ and $S = 0$, we apply the same condition in this section to perform a fair comparison between different approaches. Figure 2.4 shows the computational domain for this case study that is a horizontal pipe with one discontinuity in diameter along its length. Moreover, the time horizon is 1 s with discretization steps $\Delta x = 1$ m. The pressure at the right boundary is also $p_R(t) = 1$ bar. It should be noted as the simulation is stopped at $t = 1$ s, there are still some transient effects in the solution.

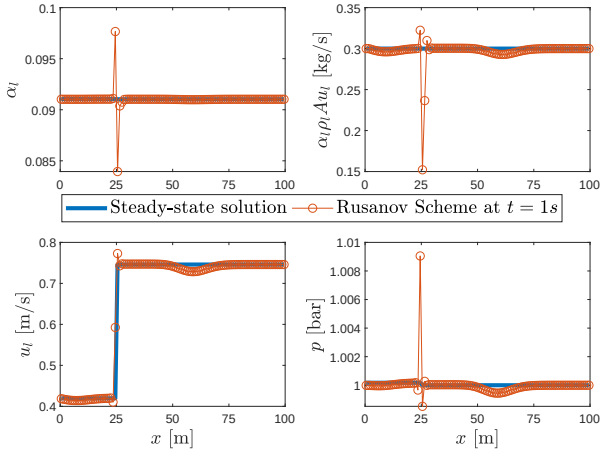

Figure 2.5: Liquid phase variables of system (2.6) by using original Rusanov scheme on variable area and the first approach: source term approximation.

2.4.1.1 First approach: source term approximation

The numerical solution of (2.6) obtained by the first approach mentioned in Section 2.3.1 in comparison with solution of system (2.13) is shown in Figure 2.5. Clearly, this kind of non-conservative remedy is afflicted by non-physical peaks at the locations of the discontinuities in the cross-sectional area where the impulsive force lies. As this method fails to capture the steady-state solution, it is not selected for further investigation by transient simulations.

2.4.1.2 Second approach: modified Rusanov scheme

After modifying the Rusanov scheme as explained in Section 2.3.2, the solution of this test cases is shown in Figure 2.6. Clearly, this modification suffers from non-physical jumps at the locations of the area discontinuities. This should not be a surprise since this modification is proved in [49] to be well-balanced only in the case of zero flow inside the domain. However, in the presence of flow, the performance of this scheme is not necessarily satisfactory in the sense that it is incapable of preserving the steady-state solution, similar to the presented results. Therefore, this method is also not studied further in this paper.

2.4.1.3 Third approach: novel, modified Rusanov scheme

The results for the third type of modification is shown in Figure 2.7. Apparently, the results of the new modified Rusanov are better than the first and

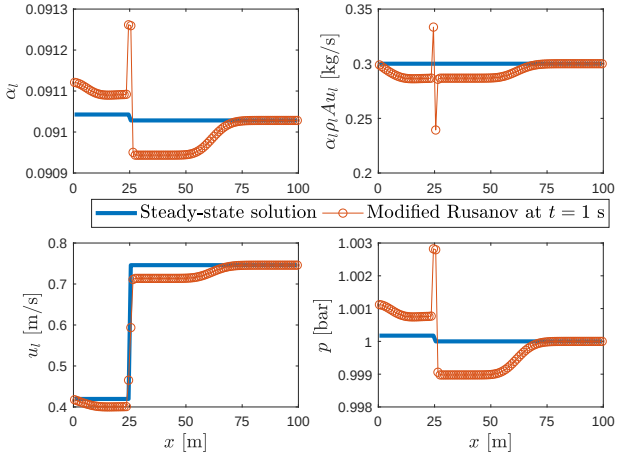

Figure 2.6: Liquid phase variables of system (2.6) by using Rusanov scheme on variable area and the second approach: modified Rusanov scheme in Section 2.3.2.

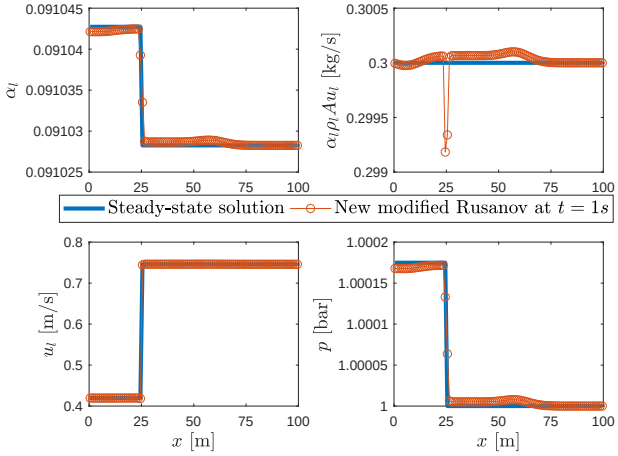

Figure 2.7: Liquid phase variables of system (2.6) by using Rusanov scheme on variable area and the third approach: novel, modified Rusanov scheme in Section 2.3.3.

second approach in preserving the steady-state solution except for relatively small discontinuity jumps in the mass flow rate. These jumps are related to the

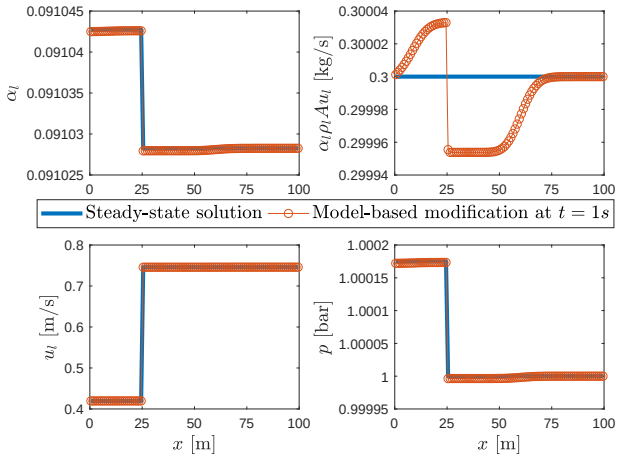

Figure 2.8: Liquid phase variables of system (2.6) by using Rusanov scheme on variable area and the fourth approach, model-based modification, together with (2.31).

approximation of the integral $\int_{x_{i-1/2}}^{x_{i+1/2}} p\, \partial A/\partial x\, dx$ by $p_i(A_{i+1} - A_{i-1})/2$. This numerical deficiency pollutes the numerical solutions. Another weakness of this approach is that by increasing the spatial rate of cross-section variation in the discretized sense at a certain location (i.e., higher jumps in the cross section and a larger impulsive force), the solution deviates from the actual steady-state solution even further. Therefore, this method is also not selected for further investigation.

2.4.1.4 Fourth approach: model-based modification

The numerical results obtained by using the fourth approach with the set of algebraic constraints in (2.31) are demonstrated in Figure 2.8. The numerical results show a significant accuracy in the preservation of the steady-state solution of the PDEs (2.6). Compared to the previous results, pressure and mass flow rate are preserved with significantly higher accuracy. The small deviation from steady-state is due to the error in solving the algebraic relations (2.31).

The simulation results for the set of algebraic constraints in (2.34) are depicted in Figure 2.9. This set of assumptions also performs well in capturing the steady-state solution. As obvious from the top-left side of Figure 2.9, the gas phase is dominant here and α_l, and subsequently α_g, change negligibly over the interface. In addition, due to the very small change of pressure apparent from bottom-right side of Figure 2.9, the assumption of constant ρ_l/ρ_g over the

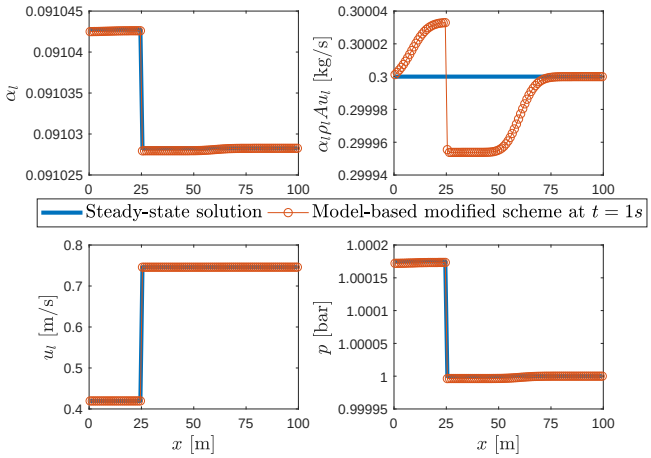

Figure 2.9: Liquid phase variables of system (2.6) by using Rusanov scheme on variable area and the fourth approach, model-based modification, together with (2.34).

interface is also valid.

Comparing all the results, the last method outperforms the other methods in capturing the steady-state solution. Therefore, the proposed modification in Section 2.3.4 together with the set of algebraic constraints (2.31) and (2.34) has been used for transient simulations. In these simulations, if the no-slip condition is imposed, the set of algebraic constraints (2.31) is used; otherwise, the algebraic constraints at the location of area variation are governed by (2.34). Since this approach uses a model-based modification, it does not suffer from any non-physical discontinuities in the solution of the state variables.

2.4.2 Comparisons in case of constant area

In this section, the performance of the proposed scheme to accurately characterize the dynamic behavior of the system in the case of constant area is analyzed. In order to do so, the result of the classical Rusanov scheme (2.21) applied to (2.1) is compared with the result of the model-based modified Rusanov scheme (2.31) or (2.34) applied to (2.6) in case of constant cross-sectional area. It should be noted again that the full Algorithm 1 is not implemented here as mentioned in Remark 2.9.

For the case of constant cross-sectional area, various benchmark tests for the DFM have been introduced. One well-known benchmark test is the DFM shock-tube problem, where the parameters of the simulation are taken from [69].

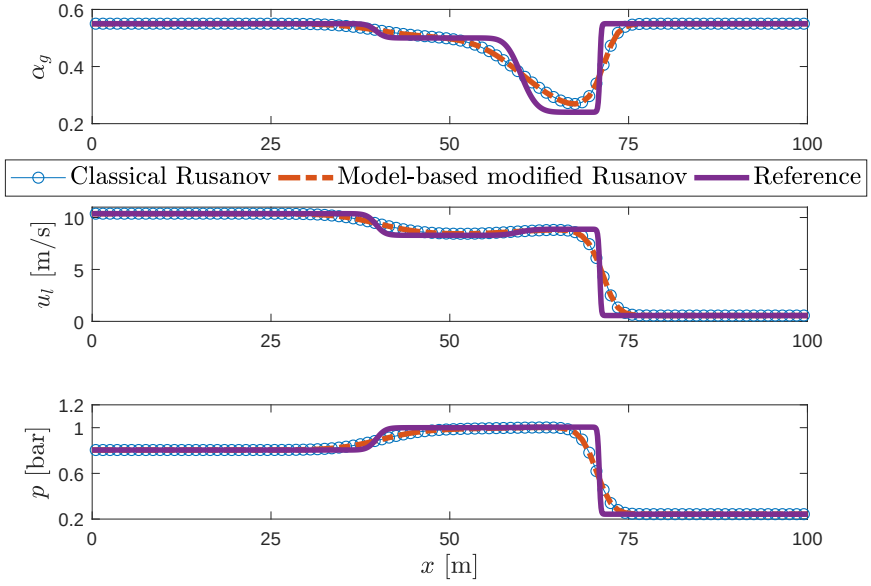

Figure 2.10: Performance of model-based modified and classical Rusanov scheme for shock tube test, constant area.

A horizontal pipe with the constant diameter of 0.1 m and length of 100 m is divided into two sections at the middle length of the pipe. For the left half of the pipe, the initial data are as below:

$$\alpha_g = 0.55,\ u_l = 10.37 \text{ m/s},\ p = 80450 \text{ Pa}.$$

For the right half of the pipe, we set

$$\alpha_g = 0.55,\ u_l = 0.561 \text{ m/s},\ p = 24282 \text{ Pa}.$$

In addition, discretization steps are $\Delta x = 1$ m, and $K = 1.07$ and $S = 0.216$.

First, we apply the modified Rusanov scheme (2.28) along with (2.34) on system (2.6) and compare the results with the original Rusanov scheme (2.21) applied to system (2.1). Both solutions should correspond exactly with each other, as seen in Figure 2.10. As area is constant, the assumptions made for the DFM with slip as in (2.33) are valid. Reference solution is obtained on a fine discretization of $\Delta x = 0.1$ m with the classical Rusanov scheme.

Thus, in order to reduce the computational cost, the algebraic relations of (2.31) and (2.34) are enforced only at the locations where the cross-section varies.

At the other locations, we set $U^*_{i+1} = U_{i+1}$ and $U^*_{i-1} = U_{i-1}$ as already noted in Algorithm 1.

2.4.3 Wave reflection in the presence of piecewise constant cross section

In this section, transients near the steady-state solution for a flow inside a pipe with piecewise constant cross-section are analyzed. At the location of discontinuous cross-section, any pressure wave is partially reflected back. Therefore, presence of the discontinuous cross-section dramatically affects the frequency response and the natural frequency of the system, which highly depends on the location and number of area variations. This kind of behavior has to be captured by the numerical simulation. In this section, the wave reflection behavior in the model-based modified Rusanov scheme together with (2.31) in both cases of discontinuous and constant cross-section of area are compared as another criterion for assessing the performance of the scheme.

The wave propagation can be excited by variation of a mass flow at the left boundary or a pressure change at the right boundary. In this simulation, the pressure at the right boundary $p_R(t)$ is increased from 1 bar to 1.2 bar at $t = 1$ s to initiate a propagating pressure wave inside the domain. Other simulation parameters are shown in Table 2.1 with $K = 1$ and $S = 0$ with the pipe shown in Figure 2.4. The comparison of pressure wave propagation and reflection between two boundaries and area discontinuities can be observed in Figure 2.11 and 2.12.

The reference solution is obtained as explained in Appendix B. In Figure 2.11, the reference solution is obtained on a coarse with $\Delta x = 1$ m (the same spatial resolution as the solution with the modified scheme) and a fine grid with $\Delta x = 0.1$ m. Result of the modified scheme and the reference solution are very close to each other. It can be observed that the pressure front generated at the location of discontinuous area (in this case at $x = 25$ m which is shown by dashed black line in Figure 2.12) opposes the incoming pressure waves and this front partially reflects back the pressure waves as obvious from the top part of Figure 2.12 at around $t = 3$ s and similarly at the other time instants. This feature, which changes the wave propagation pattern inside the domain, has been well predicted by the scheme. In case of constant area, no pressure reflection occurs until the pressure wave reaches the other boundary.

Remark 2.13. *This example resembles a choke plugging scenario within managed pressure drilling operations. When the mud gets stuck in the choke installed at the top of the annulus, the choke pressure experiences a sudden increase. Due to this pressure increase, a pressure wave propagates inside the annulus, which can be analyzed similar to the example in this section by adding relevant friction and gravity source terms into the governing equations.*

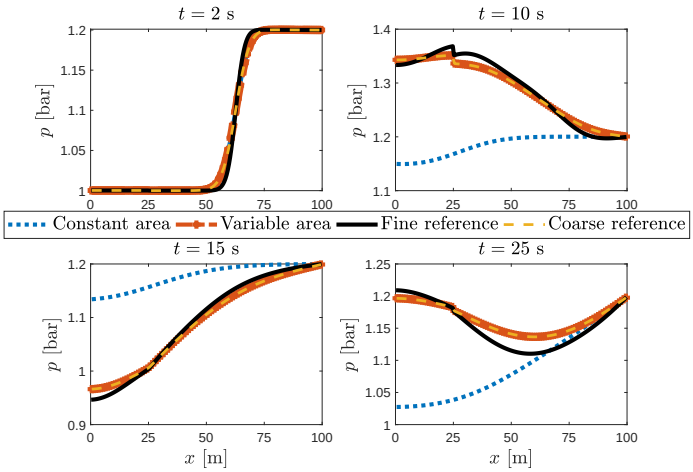

Figure 2.11: Effect of discontinuous cross section in pressure reflection (the variable cross-section is defined in Figure 2.4 for the top case and the constants cross section is with diameter of 7.5 cm for the bottom case).

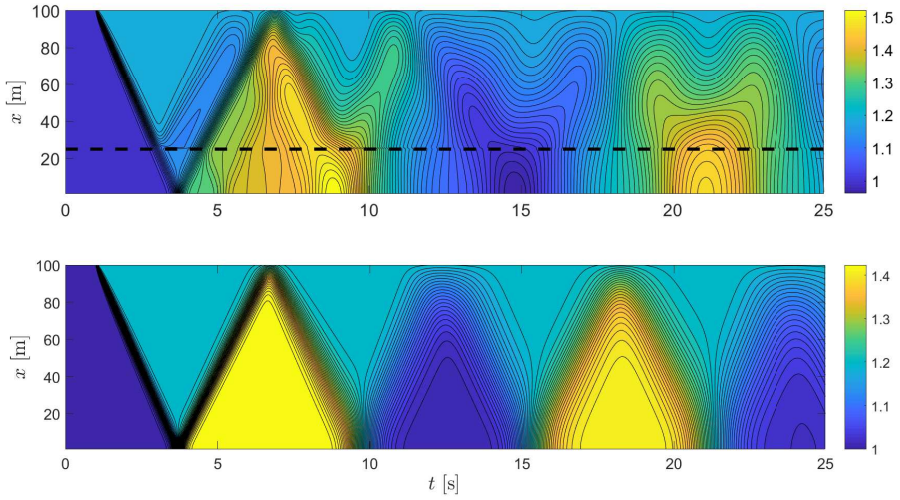

Figure 2.12: Space-time graph for pressure [bar] wave reflection pattern; top:variable cross section as in Figure 2.4, bottom: constant cross section with diameter of 7.5 cm.

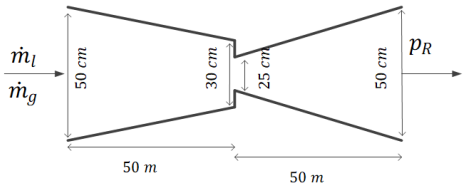

Figure 2.13: Dimension of the pipe with piecewise continuous cross-section.

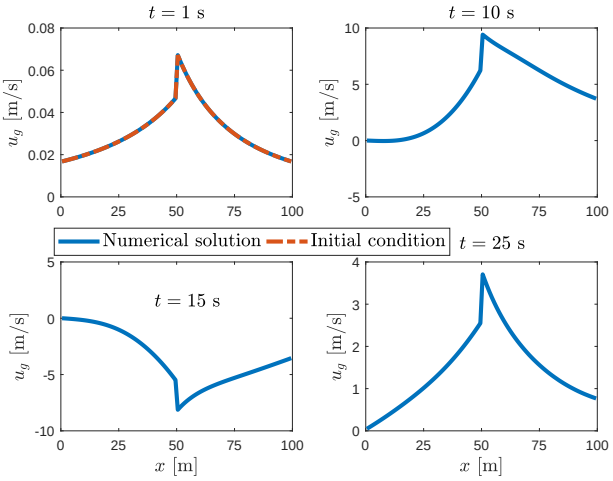

Figure 2.14: Gas velocity at different time instants.

2.4.4 Piecewise continuous cross section

Here, a two-phase flow inside a converging-diverging nozzle is investigated. The dimensions of the pipe are shown in Figure 2.13. The simulation parameters are the same as in Section 2.4.3. Figure 2.14 depicts the gas velocity at different time instants.

The top-left plot in Figure 2.14 shows that the steady-state solution is captured well by the numerical scheme. Then, by increasing the pressure at the right boundary, the opposition force at the right boundary lowers the velocity inside the domain. This test case shows the capability of the proposed scheme both in capturing the steady-state solution and in the transient simulation of two-phase flow inside a pipe with piecewise continuous cross section.

2.4.5 Error convergence analysis

To accurately study the error convergence of the scheme, a test case should be introduced such that other aspects of the numerical solution does not affect the accuracy. For instance, $\partial A/\partial x$ and the mixture speed of sound ω should be analytically known. Therefore, a pipe with continuous cross section is selected as shown in Figure 2.15, which defines $\partial A/\partial x$ analytically. To know the analytical speed of sound, no slip condition is set, i.e., $K = 1$ and $S = 0$; then, ω is governed by (2.10). Other parameters are taken from Table 2.1. Boundary conditions do not change ($p_R(t) = 1$ bar) and the system remains on its steady-state.

In order to study convergence properties of the scheme, the number of grid-cells (N) are varied, the problem is solved with the model-based modified Rusanov scheme at other values for Δx (i.e., other number of grid-cells) and the solution (here the gas velocity) is compared with the reference solution at the last time instant, $t = 1$ s. The error is defined as the relative difference between numerical gas velocity at the last time instant and the initial gas velocity at all locations, as in (2.39),

$$e = \frac{\|u_g(x,1) - u_g(x,0)\|_{L_r}}{\|u_g(x,0)\|_{L_r}}, \tag{2.39}$$

where $\|\cdot\|_{L_r}$ is the L_r norm of its argument over the spatial domain. The values for the error indicator (2.39) with increasing the number of the cells are reported in Table 2.2 and in Figure 2.16 for $r = 2, \infty$. The absolute value of the slope of each line is also depicted in this figure, confirming that the proposed scheme does not affect order of accuracy of the underlying scheme (it is well-known that the classical Rusanov scheme is first order accurate [105]).

The small discrepancy between the order of accuracy at different level of the number of grid cells is due to the inaccuracy in solving the nonlinear algebraic constraints (2.31).

All discussions in this paper focused on first-order schemes; the extension to higher-order scheme can be done, which is beyond the scope of this paper. This will be the topic of future works.

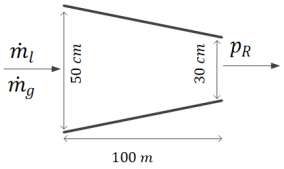

Figure 2.15: Dimension of the pipe with continuous cross-section to study the error convergence.

Table 2.2: Relative error convergence for the model-based, modified Rusanov scheme.

Number of cells	$r = 2$	$r = \infty$
50	0.012	0.0289
100	0.0061	0.0143
200	0.0031	0.0070
400	0.0015	0.0033
800	0.00076	0.0016

Generally, this approach of dealing with variable cross-sectional area is problematic when the system (2.6) starts far from steady-state solution or when it is used for simulating abrupt dynamics. In this setting, an analysis has been presented in Appendix A.

2.5 Conclusions

This paper studied numerical solvers for the non-conservative Drift Flux Model in the presence of variable cross-sectional area. Different numerical approaches have been proposed and compared to the existing approaches in the sense of accurate preservation of the steady-state solution. It has been shown that one of the new proposed schemes, the model-based modified scheme, indeed captures

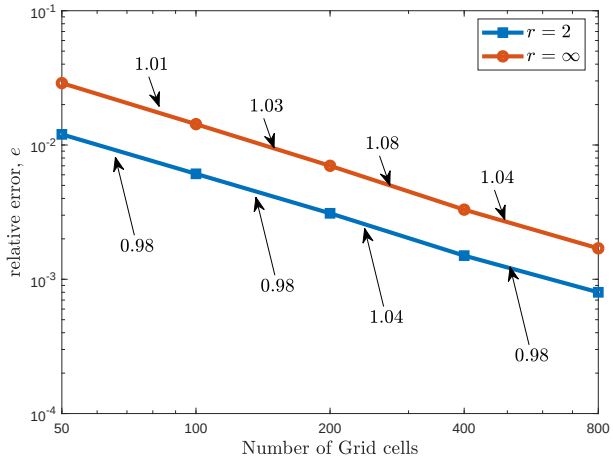

Figure 2.16: Relative error convergence by increasing the number of the grid cells with the model-based modified Rusanov scheme together with (2.31).

the physical steady-state solution with an acceptable accuracy. The model-based modified scheme can be applied to piecewise continuous cross-sectional areas as well. This modified scheme reduces to the classical scheme in case of constant area and it is also shown that the modified scheme enables simulation of the wave reflection in case of discontinuous cross-sectional area. It has been numerically proved that the proposed modification retains the first order of accuracy of the underlying scheme. Based on the performance of the proposed scheme, it can be used for simulation of industrial applications such as the hydraulics of two-phase flow occurring in drilling for oil and gas in a well with discontinuous cross-sectional area.

Chapter 3

An Approximate Well-Balanced Upgrade of Godunov-Type Schemes for the Isothermal Euler Equations and the Drift Flux Model with Laminar Friction and Gravitation

In this paper, approximate well-balanced finite-volume schemes are developed for the isothermal Euler equations and the drift flux model, widely used for the simulation of single- and two-phase flows. The proposed schemes, which are extensions of classical schemes, effectively enforce the well-balanced property to obtain a higher accuracy compared to classical schemes for both the isothermal Euler equations and the drift flux model in case of non-zero flow in the presences of both laminar friction and gravitation. The approximate well-balanced property also holds for the case of zero flow for the isothermal Euler equations. This is achieved by defining a relevant average of the source terms which exploits the steady-state solution of the system of equations. The new extended schemes reduce to the original classical scheme in the absence of source terms in the system of equations. The superiority of the proposed well-balanced schemes to classical schemes, in terms of accuracy and computational effort, is illustrated by means of numerical test cases with smooth steady-state solutions. Furthermore, the new schemes are shown numerically to be approximately first-order accurate.

This chapter is based on "M.H. Abbasi, S.Naderi Lordejani, C.Berg, L.Iapichino, W.H.A.Schilders, N.van de Wouw, ***An Approximate Well-Balanced Upgrade of Godunov-Type Schemes for the Isothermal Euler Equations and the Drift Flux Model with Laminar Friction and Gravitation***, *International Journal for Numerical Methods in Fluids*, in press, 2020."

3.1 Introduction

Numerical simulation of single- and two-phase flows has attracted the attention of researchers in the past few decades. This interest is invoked by many associated industrial applications, such as flow dynamics in petroleum refineries, distillation units, boilers of petrochemical plants and refineries, pipelines for long-distance transportation of gas and liquid [120] as well as in drilling systems [132]. Accurate prediction of the steady-state solution of such systems is crucial in the decision-making process; for instance, pipelines are usually designed to operate in the steady condition [48]. Moreover, mathematical models are used for leak detection in pipelines by comparing the measurements with the numerical steady-state solution [102]. This highly relies on the accuracy of the numerical solution.

In addition, transient multi-phase flows commonly occur in pipelines when changes in operational conditions, such as inlet and outlet flow rates, and set-point pressures, are induced. These changes are usually exerted to reach a new steady condition in the system. All these points dictate that a reliable simulator should predict not only transients accurately but also the steady-state solution.

Many numerical methods have been used for solving the equations governing the physics of a phenomenon. When there is no effect from external sources, these methods are often highly accurate in predicting steady-state behaviors of the system. However, many realistic industrial systems, such as, e.g., managed pressure drilling systems, are inevitably affected by external sources such as friction and gravitation [128]. It has been observed that classical finite-volume schemes do not preserve the analytical (or the trustworthy numerical) smooth (i.e., continuous and differentiable) steady-state solution of systems in the presence of such source terms [25], [38], [133]. To resolve this issue, much effort has been put into deriving schemes capable of preserving the analytical steady-state solution. Such schemes are called well-balanced schemes [124], [125].

The isothermal Euler equations (henceforth called Euler equations) [115] and the Drift Flux Model (DFM) [71] have been widely used for modeling single and two-phase flows in pipelines. The accuracy of Euler equations and the DFM for single-phase and two-phase flows in drilling scenarios has been verified by comparing it to real-life field data [5], [128]. Thus, these models are trustworthy for simulation of single-phase and two-phase flows in pipelines and in drilling. Developing well-balanced schemes for Euler equations and the DFM is necessary to approximate the correct steady behavior of the flow in pipelines. To capture the analytical steady-state solution of Euler equations, a few studies have been carried out for special cases [21], [39], [46], [100]. The developed schemes in these studies are all well-balanced only with respect to gravity, and not friction. Moreover, the proposed solution in [21], [39], [100] relies on the analytical steady-state solution, , which is, in general, not available or is computationally expensive to be computed at each time step. Furthermore, schemes in [21], [39], [100] are

only well-balanced in case of zero flow (stationary steady-state solutions), which restricts the applicability of these schemes. Stationary steady-state solution of Euler equations is, however, much easier to capture compared to moving (non-zero flow) steady-state solutions as the mass conservation law is automatically satisfied in the stationary situation. Moreover, the analytical stationary steady-state solution replicates a trivial hydrostatic solution, which might not be the most important scenario in drilling for oil and gas and also the transportation of liquid and gas through pipelines. The solution in [46], although applicable for moving steady-states, is based on neglecting the diffusive part of Rusanov scheme [105] when the system is in its steady state (Rusanov scheme then converts to a centered scheme when their algorithm detects that the system has reached steady condition). Nonetheless, the diffusive part of this scheme is essential for the stability of the solution during transients. Moreover, the mechanism that detects the solution is now steady and the diffusive part should be neglected is not well explained. Therefore, in this study, a different method is proposed to solve Euler equations in a well-balanced manner in a general scenario (zero and non-zero flow) in the presence of both laminar friction and gravitation with an accuracy much higher than classical schemes; however, our method does not capture steady-state solution exactly and therefore we recover the actual steady-state solution *approximately*. This leads to an *approximate* well-balanced scheme. Moreover, we also propose a scheme for the case of two-phase flows governed by the DFM. To the best of authors' knowledge, no study has been performed on capturing the smooth steady-state solution of multi-phase flows, especially the DFM.

Contrary to Euler equations and the DFM, the shallow water equations have attracted many researchers for developing a well-balanced scheme [14], [24], [39], [124], [125]. These efforts were pioneered by developing a well-balanced scheme for a lake at rest [14], [39], further extended to non-zero velocity with topography [125] and friction source terms [124]. In case of non-zero velocity, the schemes become *approximately* well-balanced as many assumptions have to be made. Moreover, the convergence rate of the scheme also reduces due to these simplifying assumptions [124], [125]. Fortunately, Euler equations [46], the DFM [69] and the shallow water equations [125] share many common features; for example, all three systems are hyperbolic (for the DFM, it is hyperbolic over a wide region of the state variables [183]) and these systems are typically discretized by finite-volume techniques. Therefore, we propose to generalize the ideas proposed for the shallow water equations and extend these techniques to Euler equations and the DFM. Compared with the studies on shallow water equations, apart from the application for other classes of PDEs, this study differs in another crucial aspect. In [124], [125], the intermediate values in the approximate Riemann solver are found to force the system to be approximately well-balanced. Here, we modify an existing scheme to upgrade it approximately well-balanced. All in all, this research aims to develop a scheme- and model-dependent framework for

increasing the accuracy of numerically obtained steady-state solutions of Euler equations and the DFM. Nonetheless, this approach can be extended straightforwardly to other numerical schemes and also to other systems of hyperbolic PDEs.

This paper is organized as follows. In Section 3.2, Euler equations and the DFM together with their steady-state solutions are introduced. In Section 3.3, a methodology to upgrade a finite-volume scheme to become well-balanced is discussed. Moreover, the application of this methodology to an advection equation, Euler equations and DFM is elaborated (the *exact* well-balanced property of the scheme is proved for the advection equation). In Sections 3.4 and 3.5, the proposed schemes are supported by illustrative numerical test cases for Euler equations and the DFM, respectively. In Section 3.6, the error convergence of the schemes is studied. Finally, Section 3.7 concludes the paper.

3.2 Single- and two-phase flow models

In this section, Euler equations and the DFM, together with the corresponding steady-state solutions, are introduced.

3.2.1 Isothermal Euler equations

Single-phase flow inside a pipe can be modeled by the isothermal Euler equations [57]. This system of equations is as follows:

$$\frac{\partial w}{\partial t} + \frac{\partial}{\partial x}\Big(f(w)\Big) = \begin{pmatrix} 0 \\ q(w,x) \end{pmatrix}, \quad t \in [0,T], \quad x \in (0,L), \tag{3.1}$$

where $w, q(w,x)$ and $f(w)$ are, respectively, the vector of conservative variables, the source terms and the mathematical flux function defined as

$$w = \begin{pmatrix} \rho \\ \rho u \end{pmatrix}, \quad f(w) = \begin{pmatrix} \rho u \\ \rho u^2 + p \end{pmatrix}, \quad q(w,x) = F(w,x) + G(w,x), \tag{3.2}$$

with $\rho(t,x)$, $u(t,x)$ and $p(t,x)$ denoting density, velocity and pressure of the fluid, respectively. The temporal and spatial variables are denoted by t and x while T and L are the final time of the simulation and the length of the spatial domain (i.e., the length of the pipe). Moreover, $F(w,x)$ is the laminar friction and $G(w,x)$ is the gravitational source term. In this paper, we consider

$$F(w,x) = -32\frac{\mu}{d^2}u, \tag{3.3a}$$

$$G(w,x) = -\rho g \sin\theta, \tag{3.3b}$$

where μ, d, g and θ are, respectively, the viscosity of the fluid, the hydraulic diameter of the pipe, the gravitational acceleration and the pipe inclination

with respect to the horizontal plane. System (3.1) is completed by the Equation Of State (EOS) of the fluid as follows:

$$p = (\rho - \rho_0)c^2 + p_0, \tag{3.4}$$

where ρ_0 and p_0 are the reference density and pressure for defining the EOS of the fluid and c is the constant speed of sound in the medium occupied by the fluid. As the dynamics for single-phase flow are now uniquely described, we can proceed to find a numerical steady-state solution of system (3.1).

3.2.1.1 Steady-state solution of the isothermal Euler equations

The equations describing the steady-state solution are obtained by using $\partial w/\partial t = 0$ in (3.1) and substituting p from (3.4) into (3.1), which read as

$$\frac{\mathrm{d}m}{\mathrm{d}x} = 0, \tag{3.5a}$$

$$\frac{\mathrm{d}\left(\dfrac{m^2}{\rho} + c^2\rho\right)}{\mathrm{d}x} = q, \tag{3.5b}$$

where $m = \rho u$ is the momentum. For subsonic flows (which is the common case for drilling applications and transport of gas and liquid), the two boundary conditions to be specified to solve system (3.5) lead to a two-point Boundary Value Problem (BVP), which is hard to be solved analytically, especially because both friction and gravity are present in q. However, this BVP can be solved numerically by the `bvp4c` solver of `MATLAB`. This solver approximates the solution to (3.5) in an iterative way while considering the boundary values at both ends of the computational domain [156]. This numerical solution later serves as a reference solution to evaluate the accuracy of the proposed scheme in predicting the steady-state solution.

However, due to the nature of the BVP, the steady-state solution is computationally expensive and thus the approach from [21], [39], [100] is not applicable here to develop a well-balanced solver for Euler equations. Moreover, since friction and gravity are present simultaneously and we aim to maintain all characteristics of the scheme (not neglecting the diffusive part), the methodology from [46] cannot be employed either. Thus, we try to find a more general approach to develop a well-balanced numerical solver in Section 3.3. Next, we analyze the steady-state solution in case of only laminar friction and only gravitational force, which later supports our arguments in Section 3.3.2.2.

Lemma 3.1. *The isothermal Euler equations* (3.1) *with* (3.4) *for the case of only friction source term* (3.3a) *has (a unique or two) steady-state solutions for*

a reference point x_0 *with* $\rho(x_0) = \hat{\rho}_0$ *and* $m(x_0) = m_0$ *if*

$$-m_0^2 \ln \frac{m_0}{c} + m_0^2 \ln \hat{\rho}_0 + \frac{m_0^2}{2} - \hat{\rho}_0 \frac{c^2}{2} + 32 \frac{\mu}{d^2} m_0 (x - x_0) \leq 0. \tag{3.6}$$

Proof. In the case of only laminar friction as the source term, (3.5) reads as

$$\frac{\mathrm{d}m}{\mathrm{d}x} = 0, \tag{3.7a}$$

$$\frac{\mathrm{d}\left(\dfrac{m^2}{\rho} + c^2 \rho\right)}{\mathrm{d}x} = -32 \frac{\mu}{d^2} \frac{m}{\rho}, \tag{3.7b}$$

Since the momentum is constant along the spatial domain in the steady state, we consider $m = m_0$ and rewrite (3.7b) as follows:

$$m_0^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2 \frac{\mathrm{d}\rho}{\mathrm{d}x} = -32 \frac{\mu}{d^2} \frac{m_0}{\rho} \rightarrow -m_0^2 \frac{\mathrm{d}\ln\rho}{\mathrm{d}x} + \frac{c^2}{2} \frac{\mathrm{d}\rho^2}{\mathrm{d}x} = -32 \frac{\mu}{d^2} m_0. \tag{3.8}$$

Consider $x_0 \in \mathbb{R}$ as a reference point with $\rho(x_0) = \hat{\rho}_0$ and integrating (3.8) over (x_0, x) and denoting $\rho(x) = \rho$, we have:

$$\zeta(\rho; x, x_0, m_0, \hat{\rho}_0) := -m_0^2 \left(\ln \rho - \ln \hat{\rho}_0\right) + \frac{c^2}{2}\left(\rho^2 - \hat{\rho}_0^2\right) + 32 \frac{\mu}{d^2} m_0 (x - x_0) = 0. \tag{3.9}$$

The minimum of ζ occurs at

$$\frac{\mathrm{d}\zeta}{\mathrm{d}\rho} = -\frac{m_0^2}{\rho} + c^2 \rho = 0 \rightarrow \rho_c = \frac{m_0}{c}. \tag{3.10}$$

Function ζ is strictly increasing on $\rho \in (\rho_c, \infty)$ and decreasing on $\rho \in (0, \rho_c)$. Therefore, function ζ admits a unique minimum for $\rho = \rho_c$. As $\zeta \rightarrow \infty$ when $\rho \rightarrow 0$ and $\rho \rightarrow \infty$, to have any steady solution, $\zeta(\rho_c) \leq 0$ must hold. This leads to the condition (3.6). □

Lemma 3.2. *The isothermal Euler equations* (3.1) *with* (3.4) *for the case of only gravity source term* (3.3b) *has (a unique or two) steady-state solutions for a reference point* x_0 *with* $\rho(x_0) = \hat{\rho}_0$ *and* $m(x_0) = m_0$ *if*

$$\frac{c^2}{2} - \frac{m_0^2}{2\rho_0^2} + c^2 \left(\ln \frac{m_0}{c} - \ln \rho_0\right) + g sin\theta (x - x_0) \leq 0. \tag{3.11}$$

Proof. Similar to Lemma 3.1, we have

$$m_0^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2 \frac{\mathrm{d}\rho}{\mathrm{d}x} = -\rho g \sin\theta \rightarrow \frac{m_0^2}{2} \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho^2}\right) + c^2 \frac{\mathrm{d}\ln\rho}{\mathrm{d}x} = -g \sin\theta. \tag{3.12}$$

Consider $x_0 \in \mathbb{R}$ as a reference point with $\rho(x_0) = \hat{\rho}_0$ and integrating above over (x_0, x) and showing $\rho(x) = \rho$, we have:

$$\zeta(\rho; x, x_0, m_0, \hat{\rho}_0) := \frac{m_0^2}{2}\left(\frac{1}{\rho^2} - \frac{1}{\hat{\rho}_0^2}\right) + c^2\left(\ln\rho - \ln\hat{\rho}_0\right) + g\sin\theta(x - x_0) = 0. \tag{3.13}$$

The minimum of ζ occurs at

$$\frac{\mathrm{d}\zeta}{\mathrm{d}\rho} = -\frac{m_0^2}{\rho^3} + \frac{c^2}{\rho} = 0 \rightarrow \rho_c = \frac{m_0}{c}. \tag{3.14}$$

Function ζ is strictly increasing on $\rho \in (\rho_c, \infty)$ and decreasing on $\rho \in (0, \rho_c)$. Therefore, function ζ admits a unique minimum for $\rho = \rho_c$. As $\zeta \rightarrow \infty$ when $\rho \rightarrow 0$ and $\rho \rightarrow \infty$, to have any steady solution, $\zeta(\rho_c) \leq 0$ must hold. This gives the condition (3.11). □

3.2.2 The drift flux model

The DFM is described by the following system of partial differential equations (PDEs) [127], which describes a two-phase flow inside a pipe:

$$\frac{\partial w}{\partial t} + \frac{\partial}{\partial x}\Big(f(w)\Big) = \begin{pmatrix} 0 \\ 0 \\ q(w,x) \end{pmatrix}, \quad t \in [0,T], \quad x \in (0,L), \tag{3.15}$$

with

$$\begin{aligned} w &= \begin{pmatrix} \alpha_l\rho_l \\ \alpha_g\rho_g \\ \alpha_l\rho_l u_l + \alpha_g\rho_g u_g \end{pmatrix}, \quad f(w) = \begin{pmatrix} \alpha_l\rho_l u_l \\ \alpha_g\rho_g u_g \\ \alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2 + p \end{pmatrix}, \\ q(w,x) &= F(w,x) + G(w,x), \end{aligned} \tag{3.16}$$

where the subscripts l and g denote the liquid and gaseous phase, respectively, and $\alpha_l(t,x)$ and $\alpha_g(t,x)$ denote the volume fraction of each phase. Frictional and gravitational terms are given by

$$F(w,x) = -32\frac{\mu_{\mathrm{mix}}}{d^2}u_{\mathrm{mix}}, \tag{3.17a}$$

$$G(w,x) = -\rho_{\mathrm{mix}} g\sin\theta \tag{3.17b}$$

with $\mu_{\mathrm{mix}} = \alpha_l\mu_l + \alpha_g\mu_g$ the mixture viscosity, $u_{\mathrm{mix}} = \alpha_l u_l + \alpha_g u_g$ the mixture velocity ($\alpha_l u_l$ and $\alpha_g u_g$ are the superficial velocities of each phase), and $\rho_{\mathrm{mix}} = \alpha_l\rho_l + \alpha_g\rho_g$ the mixture density of the gas and liquid. The DFM is completed by closure relations, as listed below [69], [91]:

$$\alpha_l + \alpha_g - 1 = 0, \tag{3.18a}$$

$$p - \left(\left(\rho_l - \rho_0\right) c_l^2 + p_0\right) = 0, \tag{3.18b}$$

$$p - \rho_g c_g^2 = 0, \tag{3.18c}$$

$$u_g - \left(K u_{\text{mix}} + S\right) = 0, \tag{3.18d}$$

where K and S are two constant parameters depending on the flow regime [30], [142]. Moreover, c_l and c_g are the constant speed of sound in the liquid and gas medium, respectively. Now, the steady-state solution of system (3.15) can be computed numerically.

3.2.2.1 Steady-state solution of the DFM

The steady-state solution of the DFM can be obtained by solving the following system of equations, which is obtained by setting $\partial w/\partial t = 0$ in (3.15):

$$\frac{\mathrm{d}m_l}{\mathrm{d}x} = 0, \tag{3.19a}$$

$$\frac{\mathrm{d}m_g}{\mathrm{d}x} = 0, \tag{3.19b}$$

$$\frac{\mathrm{d}(m_l u_l + m_g u_g + p)}{\mathrm{d}x} = q, \tag{3.19c}$$

where $m_i = \alpha_i \rho_i u_i, i \in \{l, g\}$, is the momentum of phase i. By embedding the closure relations (3.18) in (3.19), this system, which also leads to a two-point BVP due to the boundary conditions for subsonic flow, can be solved numerically. This numerical steady-state solution will be used as the reference solution to assess the well-balanced property of the numerical solver to be proposed. Similar to the discussion for Euler equations in Section 3.2.1.1, the approaches proposed in [100] and [46] cannot be employed for solving system (3.15) and new approaches should be developed. In the following section, the novel well-balanced schemes are introduced.

Remark 3.3. *Notably, other friction functions rather than laminar friction can also be studied; however, more complicated frictional source terms complicates the analysis presented in this paper, if not impossible.*

Remark 3.4. *The analysis in this paper holds true if the parameters present in the source terms such as θ, hydraulic diameter and viscosity vary smoothly along the spatial domain. In this case, the steady-state solution of Euler equations and the DFM will also be smooth, which will be used in the proofs presented in Sections 3.3.2.2 and 3.3.2.3. For the sake of simplicity, these parameters are assumed to be constant in this study.*

Remark 3.5. *Analysis of the steady-state solution of the DFM in case of only friction and only gravity, similar to what we did for Euler equations, is complicated, if not impossible.*

3.3 Well-balanced finite-volume scheme

Classical cell-centered finite-volume schemes are reliable for solving systems in the absence of source terms [69], [105], [115]. When source terms appear in the governing equations, these numerical methods are no longer well-balanced; i.e., the steady-state solutions predicted by the finite-volume solvers differ from the analytical steady-state solution. As a consequence, the solution of the finite-volume method deviates from the analytical steady-state solution even when initialized on the steady-state solution. This deviation can be made smaller by increasing the number of grid cells to a computationally unfeasible large number (much more than the required spatial resolution). Here, we propose a method that can achieve a significantly higher accuracy in predicting the steady-state solution with relatively low number of grid cells. As the method we are suggesting is scheme-dependent, in this section, first Rusanov scheme [105] is introduced as a reference classical scheme, and then a modification of the scheme is proposed, which is able to compute an accurate approximation of the analytical steady-state solution. Reasons for choosing Rusanov scheme are its simple formulation compared to other numerical schemes and the independence of its diffusivity properties on the Courant-Friedrichs-Lewy (CFL) number [105]. These features yield less complicated nonlinearities. However, the methodology introduced in this paper can be applied straightforwardly to other Godunov-type schemes as well.

3.3.1 Rusanov scheme

Let Δt and Δx refer to the temporal and spatial discretization intervals over time and space, respectively. The spatial discretization consists of cells spatially located between two interfaces $(x_{i-1/2}, x_{i+1/2})$ with the length of Δx centered at $x_i = x_{i-1/2} + \Delta x/2$. Time discretization is performed using a forward Euler integration method. Finally, first-order Gudonov-type schemes are used to numerically solve systems (3.1) and (3.15) by

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(W_i^n, W_{i+1}^n) - \mathcal{F}(W_{i-1}^n, W_i^n)\right) + \Delta t\; q\left(W_i^n, x_i\right), \quad (3.20)$$

where W_i^n and $q\left(W_i^n, x_i\right)$ are approximate averages of the conservative variables and the source terms within the i-th cell at the time instant $t^n := n\Delta t$, respectively. Variables W_i^n and W_{i+1}^n are approximations of the conservative variables at the left and right sides of the interface $x_{i+1/2}$ at the time instant t^n, respectively. Moreover, $\mathcal{F}(\cdot,\cdot)$ is the scheme-specific numerical flux function. Various numerical flux functions have been introduced in the literature [58], [69], [105], [111], [115], [158]. As a case study, the flux function for Rusanov scheme is defined as follows

$$\mathcal{F}(W_i^n, W_{i+1}^n) = \frac{f(W_i^n) + f(W_{i+1}^n)}{2} - \lambda^n(W_{i+1}^n - W_i^n), \quad (3.21)$$

where $\lambda^n := \lambda(W_{i+1}^n, W_i^n)$ is half of the absolute value of the largest eigenvalue of the Jacobian matrix of the system of equations (3.1) or (3.15) (the Jacobian matrix is $\partial\left(f(w)\right)/\partial w$ in these equations). This state-dependent eigenvalue is is calculated locally at the left and right side of each interfaces. For instance, for system (3.1),

$$\lambda(W_i^n, W_{i+1}^n) = \frac{1}{2}\max(c + |u_i^n|, c + |u_{i+1}^n|), \tag{3.22}$$

and for system (3.15),

$$\lambda(W_i^n, W_{i+1}^n) = \frac{1}{2}\max(\omega_i^n + |u_{l_i}{}^n|, |u_{g_i}{}^n|, \omega_{i+1}^n + |u_{l_{i+1}}^n|, |u_{g_{i+1}}{}^n|), \tag{3.23}$$

with ω_i^n denoting the speed of sound in the mixture of liquid and gas [69]. In the case of no-slip, i.e., $K = 1,\ S = 0$, the speed of sound in the mixture, ω_i^n, can be analytically written as follows, known as the Wood or Wallis speed of sound [56], [178]:

$$\omega_i^n = c_l c_g \sqrt{\frac{\rho_{l_i}{}^n \rho_{g_i}{}^n}{\rho_{\mathrm{mix}_i}{}^n(\alpha_{g_i}{}^n \rho_{l_i}{}^n c_l^2 + \alpha_{l_i}{}^n \rho_{g_i}{}^n c_g^2)}}. \tag{3.24}$$

When slip occurs between the two phases, computing the analytical sound velocity, due to the effect of slippage between the two phases and its effect on the wave propagation speed, is mathematically involved, if not impossible. Thus, simplified surrogates have been suggested in [69] for cases with $\alpha_g \rho_g \ll \alpha_l \rho_l$ and $0 < \alpha_g < 1$, such as

$$\omega_i^n \simeq \sqrt{\frac{p_i^n}{\alpha_{g_i}{}^n \rho_{l_i}{}^n (1 - K\alpha_{g_i}{}^n)}}. \tag{3.25}$$

These surrogates are not exact and may lead to inaccurate solutions. For this reason, only the case of no-slip is considered in this paper. The reader is referred to [58], [69] for a detailed analysis of the speed of sound in the mixture of the gas and liquid.

Remark 3.6. *Equation* (3.25) *becomes ill-posed when* α_g *tends to zero or* $K\alpha_g \to 1$. *When* $K > 1$, *the singular point even occurs for* $\alpha_g < 1$. *In these cases, other surrogate formulations should be used [94], which have not been studied in this paper.*

Remark 3.7. *In some special cases, the speed of sound* ω_i^n *becomes very low, even less than* c_g, *in the presence of both phases. However, in the relevant application domain of managed pressure drilling operations for oil and gas, phase velocities are still lower than the speed of sound in the mixture and the flow is subsonic. Henceforth, we only consider subsonic flows.*

Remark 3.8. *Physical boundary conditions are coupled with the so-called characteristic-based boundary conditions to obtain the primitive variables at the boundaries. For more information, refer to [71]. Noteworthy, the method in this paper is not dependent on the type of the boundary conditions and can be easily applied to other boundary conditions such as periodic boundary conditions.*

After this concise introduction to Rusanov scheme, a novel modification to this scheme is proposed next to upgrade the solver to be approximately well-balanced.

3.3.2 Modified Rusanov scheme

In the presence of source terms, the numerical solutions obtained by classical finite-volume schemes may drift from the actual solution depending on the contribution of the source terms to the solution. Resolving the issue of generating non-physical steady-state solutions thus requires further adjustments of the scheme by considering the effects of the source terms, leading to the definition of a well-balanced scheme. By definition, a well-balanced scheme preserves the actual steady-state equation [38]. In the following, we introduce a framework for an approximately well-balanced scheme and provide the motivation for choosing such a framework. This framework differs from the one in (3.20) in one crucial aspect: the effects of the source term $q\left(W_i^n, x_i\right)$ are incorporated in the input arguments of the numerical flux function $\mathcal{F}(\cdot,\cdot)$. The proposed modification is inspired by the work in [124]. The applicability of that work is, however, limited to basic shallow water equations. This paper extends the introduced framework in that paper to more advanced and generic models, such as the isothermal Euler equations and DFM.

The proposed structure for the well-balanced scheme: The proposed well-balanced solver of the PDEs (3.1) and (3.15) has the following structure

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(W_i^n, W_{i+1}^{*,n}) - \mathcal{F}(W_{i-1}^{*,n}, W_i^n)\right), \tag{3.26}$$

where $\mathcal{F}(\cdot,\cdot)$ can be any numerical flux function and specifically for this study it is defined in (3.21) as in Rusanov scheme. Moreover, $W_{i\pm1}^{*,n}$, henceforth called the intermediate variable, satisfies a consistency condition defined in [87], [124]. In order to define $W_{i\pm1}^{*,n}$ uniquely, complementary equations alongside the consistency condition, which are source- and model-dependent, are defined for an advection equation, for Euler equations (3.1) and for the DFM (3.15) in Sections 3.3.2.1, 3.3.2.2 and 3.3.2.3, respectively. Specifically, the computational steps for the numerical simulation of Euler equations and the DFM are summarized in Algorithm 2 and 3. In the following, we provide a motivation for the proposed structure in (3.26).

Motivation. Comparing the newly proposed scheme in (3.26) with the classical one in (3.20) reveals that the effect of the source terms in (3.26) are hidden in the intermediate variables while in (3.20) the source terms are simply integrated over a grid cell. The latter leads to an erroneous steady-state prediction, which originates from the underlying nature of classical schemes that approximate the source terms with a simple average. The problem in predicting a wrong steady-state solution is not resolved by using a different time integrator. To resolve this issue, we embed the effect of the source terms in the intermediate variables in (3.26) while the intermediate variables satisfy some algebraic constraints derived from steady-state equations. This aspect is the main difference between this study and other studies in literature [124], [125], where the intermediate variables in the approximate Riemann solver have been defined such that the scheme becomes approximately well-balanced. Instead, we modify an existing scheme to make it approximately well-balanced.

Consistency. When manipulating the scheme to contain the effect of source terms, some properties of the scheme should remain intact. Most importantly, the modified scheme should still simulate the transients accurately. Also, when reaching the steady-state solution, the analytical steady-state solution should be approximated as accurately as possible. Besides, the scheme should always be consistent, i.e., $\mathcal{F}(W_i^n, W_i^n) = f(W_i^n)$, which is the case for the original Rusanov scheme in (3.21). This is also the case for the proposed well-balanced scheme as we are not changing the definition of the flux function itself, but only modify its input arguments. Another consistency condition, defined in [87], [124], states that the average of the conservative variables obtained by the scheme should be equal to the average over the same cell of the exact solution of the Riemann problem over a length of Δx. If we focus on the spatial interval of (x_i, x_{i+1}), this consistency condition imposes the following equality:

$$\int_{x_i}^{x_{i+1/2}} W_i^{n+1} \mathrm{d}x + \int_{x_{i+1/2}}^{x_{i+1}} W_{i+1}^{n+1} \mathrm{d}x = \int_{x_i}^{x_{i+1}} W_{\mathcal{R}}\left(W_i^n, W_{i+1}^n\right) \mathrm{d}x, \tag{3.27}$$

where $W_{\mathcal{R}}\left(W_i^n, W_{i+1}^n\right)$ gives the conservative variables at the time instant t^{n+1} obtained from the exact solution of the Riemann problem at the interface $x_{i+1/2}$, which is dependent on the solutions at the neighboring cell of this interface at time instant t^n. Moreover, W_i^{n+1} is obtained from (3.20) together with (3.21). To compute the right-hand side of (3.27), the exact solution of the Riemann problem should be defined over the spatial domain of (x_i, x_{i+1}) and in the temporal domain $(t^n, t^n + \Delta t)$. It should be noted that the exact solution varies continuously over the spatial and temporal coordinate. For obtaining the above integral for the exact solution of the Riemann problem, one can use the following

equivalent equation [87]:

$$\frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} W_{\mathcal{R}}\left(W_i^n, W_{i+1}^n\right)\mathrm{d}x = \frac{1}{2}(W_i^n + W_{i+1}^n) - \frac{\Delta t}{\Delta x}\left(f(W_{i+1}^n) - f(W_i^n)\right) + \frac{1}{\Delta x}\int_{x_i}^{x_{i+1}}\int_{t^n}^{t^n+\Delta t} q\left(W_{\mathcal{R}}\left(W_i^n, W_{i+1}^n\right), x\right)\,\mathrm{d}t\,\mathrm{d}x, \tag{3.28}$$

where $f(\cdot)$ is the mathematical flux function as in (3.1) or (3.15). The integral of the numerical solution on the left-hand side of (3.27) depends on the order of the accuracy of the scheme, i.e., how the solution changes within a grid cell (for first-order accurate schemes, the solution is constant within a grid cell). We proceed with the computation below by considering first-order accurate schemes. This changes the left side of (3.27) to:

$$\frac{1}{\Delta x}\left(\int_{x_i}^{x_{i+1/2}} W_i^{n+1}\mathrm{d}x + \int_{x_{i+1/2}}^{x_{i+1}} W_{i+1}^{n+1}\mathrm{d}x\right) = \frac{1}{2}(W_i^{n+1} + W_{i+1}^{n+1}). \tag{3.29}$$

By embedding (3.20) into (3.29), we obtain:

$$\frac{1}{\Delta x}\left(\int_{x_i}^{x_{i+1/2}} W_i^{n+1}\mathrm{d}x + \int_{x_{i+1/2}}^{x_{i+1}} W_{i+1}^{n+1}\mathrm{d}x\right) = \frac{1}{2}(W_i^n + W_{i+1}^n) - \frac{\Delta t}{2\Delta x}\left(\mathcal{F}(W_{i+1}^n, W_{i+2}^n) - \cancel{\mathcal{F}(W_i^n, W_{i+1}^n)} + \cancel{\mathcal{F}(W_i^n, W_{i+1}^n)} - \mathcal{F}(W_{i-1}^n, W_i^n)\right) + \frac{\Delta t}{2}\left(q(W_i^n, x_i) + q(W_{i+1}^n, x_{i+1})\right). \tag{3.30}$$

Now, we propose to accommodate the effects of the source terms q into those input arguments of the numerical flux functions that are not in the neighborhood of the interface $x_{i+1/2}$ (in this case, W_{i+2} and W_{i-1} and change their subscripts to enable the solution locally at each interface). This leads to the definition of the intermediate variables, $W_{i+1}^{*,n}$ and $W_i^{*,n}$, that encompass the effect of the source terms. Embedding these into (3.30) leads to

$$\frac{1}{\Delta x}\left(\int_{x_i}^{x_{i+1/2}} W_i^{n+1}\mathrm{d}x + \int_{x_{i+1/2}}^{x_{i+1}} W_{i+1}^{n+1}\mathrm{d}x\right) = \frac{1}{2}(W_i^n + W_{i+1}^n) - \frac{\Delta t}{2\Delta x}\left(\mathcal{F}(W_{i+1}^n, W_{i+1}^{*,n}) - \mathcal{F}(W_i^{*,n}, W_i^n)\right). \tag{3.31}$$

Finally, the consistency equations at each interface are obtained by equating (3.31) and (3.28):

$$2\left(f(W_{i+1}^n) - f(W_i^n)\right) - \frac{2}{\Delta t}\int_{x_i}^{x_{i+1}}\int_{t^n}^{t^n+\Delta t} q\left(W_{\mathcal{R}}\left(W_i^n, W_{i+1}^n\right), x\right) \mathrm{d}x\ \mathrm{d}t = \mathcal{F}(W_{i+1}^n, W_{i+1}^{*,n}) - \mathcal{F}(W_i^{*,n}, W_i^n). \tag{3.32}$$

Equation (3.32) is the consistency equation that we consider for the proposed scheme (3.26).

Remark 3.9. *The intermediate variables, $W_{i+1}^{*,n}$ and $W_i^{*,n}$ introduced in* (3.31)*, contain the effect of the source terms, $q(W_i^n, x_i)$ and $q(W_{i+1}^n, x_{i+1})$, and the effect conservative variables, W_{i+2}^n and W_{i-1}^n. These variables are computed based on the governing equations of the physical phenomenon and also the source terms, which will be explained in Sections 3.3.2.1, 3.3.2.2 and 3.3.2.3 for each case study.*

Remark 3.10. *In this study, we focus only on first-order schemes. To upgrade the scheme to second-order, MUSCL approaches (see for example [26]) can be followed, which will be the topic of future studies.*

The main question is how to approximate the integral in (3.32) which contains the exact solution to the Riemann problem. This can be resolved by exploiting the governing PDEs that encompass the exact solution, which will be explained in Sections 3.3.2.2 and 3.3.2.3. Moreover, (3.32) contains many unknowns and the equation is not uniquely solvable in isolation. All the complementary equations, necessary to obtain the intermediate variables such as $W_i^{*,n}$ uniquely, will be introduced later for each system of equations in Sections 3.3.2.2 and 3.3.2.3.

The unknown terms in (3.32) are the intermediate variables and the average of the source term (the integral term in (3.32)). The average of source terms should be determined such that the numerical steady-state solution approximately recovers the analytical one. Since we do not have the exact solution to the Riemann problem, we can exploit the original PDEs to find the average of the source terms. To this end, this average will be approximated by exploiting the algebraic relations originated from the system of equations (3.1) and (3.15), which will be clarified in Sections 3.3.2.2 and 3.3.2.3. Henceforth, the integral $1/\Delta t\ 1/\Delta x \int_{x_i}^{x_{i+1}}\int_{t^n}^{t^n+\Delta t} q\Big(W_{\mathcal{R}}(W_i^n, W_{i+1}^n), x\Big)\ \mathrm{d}x\ \mathrm{d}t$ is denoted by $\bar{Q}$. It should be noted that this treatment of the scheme is dependent on the nature of the source terms, which is explained later in the aforementioned sections. The intermediate variables should be obtained by considering two essential properties that are

required to be satisfied by the well-balanced solver: consistency with the actual system and the well-balanced property.

The consistency with the actual system of equations is ensured by satisfying algebraic relations originating from the steady-state model. These algebraic relations are obtained in the next section for the isothermal Euler equations and the DFM. For the well-balanced property, from (3.26), we deduce that the solution is stationary, $W_i^{n+1} = W_i^n$, if the solution is assumed constant within each grid cell (the underlying scheme is first-order accurate) and $W_{i+1}^{*,n} = W_i^n$ and $W_i^{*,n} = W_{i+1}^n$. Therefore, we seek $W_{i+1}^{*,n} = W_i^n$ and $W_{i-1}^{*,n} = W_i^n$ as soon as W_i^n and W_{i+1}^n define a steady state. Then, the right-hand side of (3.32) becomes zero and the left-hand side resembles the steady-state model at the discrete level. Here, the pair (W_i, W_{i+1}) is said to define a steady state if the equations (3.5) and (3.19) are satisfied at the discrete level. Such intermediate states will enforce the well-balancedness of our scheme. All these properties will help to define the intermediate variables in Sections 3.3.2.2 and 3.3.2.3.

The idea proposed in [100] consists of modifying the effect of the source term by knowing the difference between the numerical steady-state solution and the analytical one. This means that finding the averaged contribution of the source terms requires finding the analytical steady-state solution, which is however challenging and expensive due to the BVP structure of the steady-state problem. Instead, one can use algebraic relations that are valid during the steady state without prior knowledge of the steady-state solution itself. Now, the methodology introduced in [124], [125] is employed and modified for any general scheme and applied to Euler equations and the DFM. But first, in the following, we prove that the proposed scheme leads to an *exact* well-balanced solution for an advection equation.

Remark 3.11. *In this study, we investigate laminar friction characterizations. In general, the approach in this paper is applicable to turbulent friction functions as well. However, with turbulent friction, the analysis is highly demanding and the numerical solution is generally hard to obtain, even in the case of a classical numerical solver. As a result, the well-balanced solution would be even more complex.*

3.3.2.1 An advection equation with a source term

As Euler equations and DFM are coupled and the corresponding source terms are sometimes nonlinear with respect to the conservative variables, the analytical assessment of the performance of the proposed scheme on these equations is cumbersome if not impossible. Therefore, we provide the assessment for a simple, though relevant, test case, a scalar PDE governing an advection phenomenon as below:

$$\frac{\partial w}{\partial t} + \frac{\partial w}{\partial x} = q(w), \tag{3.33}$$

where we set $q(w) = w$ for simplicity of the assessment.

Theorem 3.12. *Consider the advection equation in* (3.33)*. The numerical solution of this system for* $q(w) = w$ *obtained by the solver* (3.26) *with numerical flux function in* (3.21) *and the intermediate variables satisfying* (3.32) *and* $\Delta t/\Delta x < 1$ *leads to zero error in approximating the steady-state solution while the solver* (3.20) *together with* (3.21) *yields an erroneous steady-state solution.*

Proof. Applying the Rusanov scheme (3.21) yields ($\lambda_i^n = 1/2$)

$$\mathcal{F}(W_i^n, W_{i+1}^n) = W_i^n. \tag{3.34}$$

The consistency condition (3.32) adapted for this advection equation at any interface is obtained as follows

$$2\bigg(f(W_{i+1}^n) - f(W_i^n)\bigg) - 2\bar{Q}\Delta x = \mathcal{F}(W_{i+1}^n, W_{i+1}^{*,n}) - \mathcal{F}(W_i^{*,n}, W_i^n). \tag{3.35}$$

Embedding (3.34) into the above equation gives,

$$[W] - \bar{Q}\Delta x = \frac{1}{2}(W_{i+1}^n - W_i^{*,n}). \tag{3.36}$$

Now, we have to exploit (3.33) in the steady-state condition to compute $\bar{Q}$, as below. In steady state, it holds that

$$W_{i+1}^n - W_i^n = \bar{Q}\Delta x. \tag{3.37}$$

Moreover, as we know the source term, for the steady-state solution it holds that

$$\frac{\mathrm{d}w}{\mathrm{d}x} = w \Rightarrow [\ln W] = \Delta x \Rightarrow W_{i+1} = W_i e^{\Delta x}. \tag{3.38}$$

Therefore, we can find the expression for $\bar{Q}\Delta x = W_i(e^{\Delta x} - 1)$. Using the Taylor expansion of $e^{\Delta x}$, it can be verified that $\bar{Q}\Delta x$ is consistent with $W_i\Delta x$. Then, at each interface, (3.36) yields

$$W_i^{*,n} = -W_{i+1}^n + 2W_i^n e^{\Delta x}. \tag{3.39}$$

Due to the specific form of the advection equation and the Rusanov scheme, the calculation of $W_{i+1}^{*,n}$ is not required. The proposed well-balanced solver (3.26) is repeated here:

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(W_i^n, W_{i+1}^{*,n}) - \mathcal{F}(W_{i-1}^{*,n}, W_i^n)\right). \tag{3.40}$$

When reaching the exact steady-state profile, the relation $W_i^{n+1} = W_i^n$ should hold. We define, $\varepsilon = W_i^{n+1} - W_i^n$ and we compute ε for the advection equation

with the classical way of dealing with source term as in (3.20) and the proposed well-balanced way as in (3.26).

For the well-balanced solver (3.26) with the Rusanov scheme (3.21), it holds that

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(W_i^n, W_{i+1}^{*,n}) - \mathcal{F}(W_{i-1}^{*,n}, W_i^n)\right) = -\frac{\Delta t}{2\Delta x}\left(W_i^n - W_{i-1}^{*,n}\right). \tag{3.41}$$

By incorporating (3.39), we obtain

$$W_i^{n+1} = \left(1 - \frac{\Delta t}{\Delta x}\right) W_i^n + \frac{\Delta t}{\Delta x} W_{i-1}^n e^{\Delta x} \rightarrow \varepsilon^n = -\frac{\Delta t}{\Delta x}\left(W_i^n - W_{i-1}^n e^{\Delta x}\right). \tag{3.42}$$

Due to the stability of the scheme and the positive numerical diffusion coefficient (because of the CFL condition $\Delta t/\Delta x < 1$) [54], as $n \rightarrow \infty$, $\epsilon \rightarrow 0$. Therefore $W_i^n = W_{i-1}^n$, which corresponds to the actual steady-state solution (3.38). This shows that the new proposed scheme will lead to zero error at the actual steady-state solution and if the solver starts from the analytical steady-state solution, it remains there. Also, it can be inferred that the only solution of the well-balanced scheme that yields $W_i^{n+1} = W_i^n$ is the analytical steady-state solution. Next, we prove that the classical scheme does not preserve the solution. This is proved by contradiction.

Recalling the classical solver (3.20) with Rusanov scheme (3.21) as below:

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}\left(W_i^n, W_{i+1}^n\right) - \mathcal{F}\left(W_{i-1}^n, W_i^n\right)\right) + \Delta t q(W_i^n), \tag{3.43}$$

we obtain

$$\varepsilon^n = -\frac{\Delta t}{\Delta x}(W_i^n - W_{i-1}^n) + \Delta t W_i^n. \tag{3.44}$$

Then, if the solver starts from the analytical steady-state solution, i.e. $W_i = W_0 e^{i\Delta x}$, we obtain,

$$\varepsilon^n = W_0 \Delta t e^{i\Delta x}\left(1 - \frac{1}{\Delta x}\left(1 - e^{-\Delta x}\right)\right) = W_0 e^{i\Delta x}\Delta t\left(\frac{\Delta x}{2!} + \mathcal{O}\left(\Delta x^2\right)\right) \neq 0, \tag{3.45}$$

Clearly, the right-hand side of (3.45) is non-zero, meaning that $W_i = W_0 e^{i\Delta x}$ is not the steady-state solution of (3.43). This contradiction completes the proof. The error in steady state approximation tends to zero only by making the spatial and temporal grid size (Δx and Δt) smaller. □

Supported by Theorem 3.12, we project that the proposed scheme also leads to better results for coupled equations such as Euler equations and DFM.

3.3.2.2 Modified scheme for the isothermal Euler equations

To compute the average source terms $\bar{Q}$ that satisfy steady-state equations (3.5), we exploit the discrete version of the steady-state equations. To this end, we integrate (3.5) at each interface over (x_i, x_{i+1}),

$$m_L = m_R =: m_0, \tag{3.46a}$$

$$m_0^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2 \frac{\mathrm{d}\rho}{\mathrm{d}x} = q \Longrightarrow m_0^2 \left[\frac{1}{\rho}\right] + c^2 \left[\rho\right] = \bar{Q}\Delta x, \tag{3.46b}$$

where $[\cdot] = (\cdot)_R - (\cdot)_L$ denotes the difference of variables between the right (subscript R) and left (subscript L) side of the interface $x_{i+1/2}$. Considering (3.46b), there are two unknowns, m_0 and $\bar{Q}$. In this paper, $\bar{Q} = \bar{F} + \bar{G}$ where $\bar{F}$ and $\bar{G}$ are the average of frictional and gravitational source terms. One more equation is thus required to solve this equation. The steady-state solution associated to the full source term Q does not admit an algebraic expression in the presence of both friction and gravity and therefore the source terms should be decomposed into individual source terms. So, instead of finding $\bar{Q}$ such that (3.46b) is satisfied, we find $\bar{F}$ and $\bar{G}$ satisfying other equations with similar structure to (3.46b), and then set $\bar{Q} = \bar{F} + \bar{G}$. It should be noted the $\bar{Q}$ found in this way might not satisfy (3.46b) exactly an therefore it leads to some errors in the steady-state solution. We first explain this step for the friction-related terms and then for the gravity contribution to the source term. To find consistent (relevant) source terms $\bar{F}$ and $\bar{G}$ even in transient case, we exploit the discrete steady-states equations in case of only friction and only gravity, respectively. These source terms entail defining new parameters in the source terms, which converge to the corresponding steady values when the system is reaching the steady condition.

Frictional source terms

Considering only laminar friction (3.3a) in (3.5b), to define an average friction source term that is consistent with the steady-state equations, we set:

$$m_0^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2 \frac{\mathrm{d}\rho}{\mathrm{d}x} = -32\frac{\mu}{d^2}\frac{\bar{m}}{\rho} \Longrightarrow m_0^2 \rho \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2 \rho \frac{\mathrm{d}\rho}{\mathrm{d}x} = -32\frac{\mu}{d^2}\bar{m}, \tag{3.47}$$

where $\bar{m}$ can be interpreted as an average of the momentum over the left and right side of the interface and it should be defined such that it converges to m_0 as the solution reaches the steady state. To this end, the following description

for $\bar{m}$ is assumed,

$$\bar{m} = \begin{cases} 0, & \text{if} \quad m_L = m_R = 0 \\ \dfrac{2|m_L m_R|}{|m_L|+|m_R|}\text{sign}(m_L + m_R), & \text{otherwise} \end{cases}, \qquad \text{with} \;\; \text{sign}(a) = \begin{cases} 1, & \text{if } a > 0, \\ 0, & \text{if } a = 0, \\ -1, & \text{if } a < 0. \end{cases} \tag{3.48}$$

This average indeed ensures that when the system reaches its steady state, $m_L = m_R$, then $\bar{m} = m_L = m_R = m_0$. By integrating (3.47) over the interval of (x_i, x_{i+1}), we obtain:

$$-m_0^2[\ln \rho] + \frac{c^2}{2}[\rho^2] = -32\frac{\mu}{d^2}\bar{m}\Delta x. \tag{3.49}$$

When reaching steady-state, $\bar{m} = m_0$ and (3.49) becomes a discrete version of (3.8) and the same conditions for the availability of the solution holds as in Lemma 3.1 (with $x_0 = x_L$, $x = x_R$, $\hat{\rho}_0 = \rho_L$ and $\rho = \rho_R$). Finally solving (3.49) gives m_0, and by substituting this m_0 into (3.46b) in case of only laminar friction as the source term, we obtain

$$m_0^2\left[\frac{1}{\rho}\right] + c^2[\rho] = \bar{F}\Delta x. \tag{3.50}$$

Now, by substituting m_0 from (3.49) into (3.50), we obtain

$$\bar{F}\Delta x = \frac{\frac{c^2}{2}[\rho^2] + 32\frac{\mu}{d^2}\bar{m}\Delta x}{[\ln \rho]}\left[\frac{1}{\rho}\right] + c^2[\rho]. \tag{3.51}$$

This source term should be equivalent to (consistent with) the actual friction source term, which is stated below.

Proposition 1. *Under the assumption of smooth (steady and transient) solution, $\bar{F}$ obtained in* (3.51) *is consistent with the actual friction defined in* (3.3a).

Proof. For smooth solutions, there exists ρ_i and ρ_j with $|\rho_i - \rho_L| < |\rho_R - \rho_L|$ and $|\rho_j - \rho_L| < |\rho_R - \rho_L|$ such that

$$[\ln \rho] = \frac{\rho_R - \rho_L}{\rho_i}, \quad \left[\frac{1}{\rho}\right] = -\frac{\rho_R - \rho_L}{\rho_j^2}. \tag{3.52}$$

Substitution of (3.52) in (3.51) yields:

$$\begin{aligned} \bar{F}\Delta x = &-\left(\frac{c^2}{2}\left(\rho_R^2 - \rho_L^2\right) + 32\frac{\mu}{d^2}\bar{m}\Delta x\right)\frac{\rho_i}{\rho_j^2} + c^2\left(\rho_R - \rho_L\right) = \\ &c^2\left(\rho_R - \rho_L\right)\left(1 - \frac{\rho_R + \rho_L}{2}\frac{\rho_i}{\rho_j^2}\right) - 32\frac{\mu}{d^2}\bar{m}\Delta x\frac{\rho_i}{\rho_j^2}. \end{aligned} \tag{3.53}$$

For smooths solutions, by increasing the number of the cells, the values of $\rho_R, \rho_L, \rho_i, \rho_j$ all converge to a single value, let us say ρ. Then, (3.53) changes to:

$$\bar{F}\Delta x = -32\frac{\mu}{d^2}\frac{\bar{m}}{\rho}\Delta x. \tag{3.54}$$

In smooth solutions, it holds that $\bar{m}/\rho = u$. This proves that under the assumption of smooth solutions, $\bar{F}$, obtained from (3.51), indeed approximates (3.3a). □

Gravitational source terms

For the case of only gravitational source term (3.3b), (3.5b) changes to:

$$m_0^2\frac{\mathrm{d}}{\mathrm{d}x}(\frac{1}{\rho}) + c^2\frac{\mathrm{d}}{\mathrm{d}x}(\rho) = -\rho g\sin\theta \Longrightarrow m_0^2\frac{1}{\rho}\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho}\right) + c^2\frac{1}{\rho}\frac{\mathrm{d}}{\mathrm{d}x}(\rho) = -g\sin\theta. \tag{3.55}$$

By integrating (3.55) over the interval of (x_i, x_{i+1}), we obtain

$$\frac{m_0^2}{2}\left[\frac{1}{\rho^2}\right] + c^2[\ln\rho] = -g\sin\theta\Delta x. \tag{3.56}$$

Notably, (3.56) is a discrete version of (3.12) and the same conditions for the availability of the solution holds as in Lemma 3.2 (with $x_0 = x_L,\ x = x_R,\ \hat{\rho}_0 = \rho_L$ and $\rho = \rho_R$). After obtaining m_0 from the above equation, $\bar{G}$ can also be computed from (3.46b) as follows:

$$m_0^2\left[\frac{1}{\rho}\right] + c^2[\rho] = \bar{G}\Delta x. \tag{3.57}$$

Finally, we obtain the following expression for $\bar{G}$:

$$\bar{G}\Delta x = -2\frac{g\sin\theta\Delta x + c^2[\ln\rho]}{\left[\frac{1}{\rho^2}\right]}\left[\frac{1}{\rho}\right] + c^2[\rho]. \tag{3.58}$$

Now, we study the equivalence of this source term with the actual gravitational source term.

Proposition 2. *Under the assumption of smooth (steady and transient) solution, $\bar{G}$ in* (3.58) *is equivalent to* (3.3b).

Proof. Substituting $[1/\rho]/[1/\rho^2] = \rho_R\rho_L/(\rho_R + \rho_L)$ to (3.58) leads to

$$\bar{G}\Delta x = -2\frac{\rho_R\rho_L}{\rho_R + \rho_L}g\sin\theta\Delta x + c^2\left(\rho_R - \rho_L\right)\left(1 - 2\frac{\rho_R\rho_L}{\rho_R + \rho_L}\left[\ln\rho\right]\right). \tag{3.59}$$

Increasing the number of the cells for smooth solutions yields $\rho_L, \rho_R \to \rho$ and therefore (3.59) changes to:

$$\bar{G}\Delta x = -\rho g \sin\theta \Delta x. \tag{3.60}$$

This proves that under the assumption of smooth solution, the term $\bar{G}$ indeed approximates (3.3b). □

Remark 3.13. *We note that using Propositions 1-2 allows for the recovery of the intermediate states for the gravity only if the friction source term vanishes. Similarly, if the gravity source term vanishes, we recover the intermediate states for friction only. As a consequence, the computational step yields intermediate states that are well-balanced for the individual source terms of gravity or friction. Let us recall that the steady-state relation for* (3.5b) *with both frictional and gravitational source terms cannot be written in the form of an algebraic relation. Therefore, we can only preserve the steady state solution* ***approximately*** *when both source terms are present. See [124] where a similar approach is used for the shallow-water equations.*

Remark 3.14. *Note that the expressions of the averaged source terms* (3.51) *and* (3.58) *have been obtained by considering W_L and W_R satisfying steady-state models. Since these expressions only depend on the left and right states and it has been proved in Propositions 1 and 2 that the averaged source terms are equivalent to the actual source terms, it is relevant to extend the usage of the averaged source terms to the case where these states do not define a steady state, and actually use the well-balanced expressions* (3.51) *and* (3.58) *for all W_L and W_R.*

The average source term is then specified as $\bar{Q} = \bar{F} + \bar{G}$. After obtaining the average source term, the modified scheme can be completed by calculating the intermediate values $m_L^*, m_R^*, \rho_L^*, \rho_R^*$ needed in the modification of the scheme. To this end, by rewriting the consistency conditions (3.32) for the modified Rusanov scheme (3.26) and for (3.1), we have:

$$2[m] = \left(\frac{m_R^* + m_R}{2} - \frac{\lambda_R}{2}(\rho_R^* - \rho_R)\right) - \left(\frac{m_L + m_L^*}{2} - \frac{\lambda_L}{2}(\rho_L - \rho_L^*)\right), \tag{3.61a}$$

$$2\left[\frac{m^2}{\rho} + c^2\rho\right] - 2\bar{Q}\Delta x = \left(\frac{\left(\frac{m_R^{*2}}{\rho_R^*} + c^2\rho_R^*\right) + \left(\frac{m_R^2}{\rho_R} + c^2\rho_R\right)}{2} - \frac{\lambda_R}{2}(m_R^* - m_R)\right) - \left(\frac{\left(\frac{m_L^2}{\rho_L} + c^2\rho_L\right) + \left(\frac{m_L^{*2}}{\rho_L^*} + c^2\rho_L^*\right)}{2} - \frac{\lambda_L}{2}(m_L - m_L^*)\right). \tag{3.61b}$$

As we are focusing on subsonic scenarios, λ_L and λ_R are dominantly governed by the sound velocity apparent from (3.22) as $u \ll c$, not by the state values, we can consider $\lambda := \lambda_L = \lambda_R$. Although this assumption leads to some errors in the end, it will significantly simplify the calculations and the nonlinear equations can be solved more easily by Newton-based methods. Due to the specific form of the steady-state solution (constant momentum over the spatial domain) and after [87], we set $m^* := m_R^* = m_L^*$ (this choice is also used in case of transients and helps to satisfy the steady equations more easily). Therefore, the system of algebraic relations in (3.61) simplifies to

$$\rho_R^* + \rho_L^* = \rho_R + \rho_L - \frac{3}{\lambda}[m], \tag{3.62a}$$

$$\frac{3}{2}\left[\frac{m^2}{\rho} + c^2\rho\right] - 2\bar{Q}\Delta x = \frac{m^{*2}}{2}\left(\frac{1}{\rho_R^*} - \frac{1}{\rho_L^*}\right) - \lambda m^* + \frac{1}{2}c^2(\rho_R^* - \rho_L^*) + \frac{\lambda}{2}(m_R + m_L). \tag{3.62b}$$

Still one more equation is needed to be able to compute the variables uniquely. This last equation should be defined such that when we are on the steady-state profile, the intermediate variables satisfy the steady-state equation (and also should be usable during transients). To do so, we suggest an equation which can be used both at the steady-states and transients [124]. This is carried out by adapting the steady equation (3.46b) as follows:

$$\begin{aligned} &\bar{m}^2\left[\frac{1}{\rho}\right] + c^2[\rho] = \bar{Q}\Delta x \Longrightarrow \bar{m}^2\left(\frac{1}{\rho_R} - \frac{1}{\rho_L}\right) + c^2(\rho_R - \rho_L) = \bar{Q}\Delta x \Longrightarrow \\ &\left(c^2 - \frac{\bar{m}^2}{\rho_R\rho_L}\right)(\rho_R - \rho_L) = \bar{Q}\Delta x. \end{aligned} \tag{3.63}$$

The intermediate values should also satisfy the last relation in the above equation [124], [125], meaning that:

$$\left(c^2 - \frac{\bar{m}^2}{\rho_R\rho_L}\right)(\rho_R^* - \rho_L^*) = \bar{Q}\Delta x. \tag{3.64}$$

Now, equations (3.62) and (3.64) form a complete system of equations from which the intermediate values can be computed. Algorithm 1 summarizes the procedure for computing the intermediate variables W_R^*, W_L^*, i.e., the variables that are required in the numerical scheme (3.26).

Remark 3.15. *Due to the high density of the liquid in test cases of this paper, a negative intermediate density was not observed in our realistic industrial test cases. Positivity preserving techniques [35] can be used in case of encountering such problems.*

Now, we continue with the DFM and modify the Rusanov scheme correspondingly.

Algorithm 2: WB-Euler: well-balanced scheme for the Euler equations

Input: W_L, W_R at each interface
Output: W_L^*, W_R^* at the same interface
1 Calculate $\bar{m}$ from (3.48),
2 Compute $\bar{F}$ and $\bar{G}$ from (3.51) and (3.58), respectively,
3 Set $\bar{Q} = \bar{F} + \bar{G}$,
4 Solve (3.62) and (3.64) simultaneously and obtain ρ_L^*, ρ_R^* and m^*,
5 Set $W_L^* = [\rho_L^*,\ m^*]^T$ and $W_R^* = [\rho_R^*,\ m^*]^T$.

3.3.2.3 Modified scheme for the DFM

Integrating system (3.19) at each interface over the interval (x_i, x_{i+1}), we have:

$$(m_l)_L = (m_l)_R = m_{l0}, \tag{3.65a}$$

$$(m_g)_L = (m_g)_R = m_{g0}, \tag{3.65b}$$

$$m_{l0}^2 \left[\frac{1}{\alpha_l \rho_l}\right] + m_{g0}^2 \left[\frac{1}{\alpha_g \rho_g}\right] + [p] = \bar{Q}\Delta x. \tag{3.65c}$$

Three unknowns, m_{l0}, m_{g0} and $\bar{Q}$, are present in (3.65c). Two more source-specific equations are thus needed to be coupled with (3.65c) to uniquely find these unknowns. Since the additional equations are source-dependent, friction and gravity are treated separately. Similar arguments as put forward for the isothermal Euler equations also hold here. So, instead of finding $\bar{Q}$ such that (3.65c) is satisfied, we find $\bar{F}$ and $\bar{G}$ satisfying other conditions, and then set $\bar{Q} = \bar{F} + \bar{G}$. It should be noted the $\bar{Q}$ found in this way might not satisfy (3.65c) exactly. We will first do the decomposition for the friction-related terms and then for the gravity contribution to the source term. Due to the complicated nature of the DFM, the following assumption is made only to attain the average source terms.

Assumption 3.16. *[10] Only to obtain the average source terms, it is assumed that at interfaces of the discrete DFM as in* (3.19)*, the volume and mass composition of the mixture do not change.*

This assumption is approximately valid when smooth solutions are considered; otherwise, this assumption is less accurate. Assumption 3.16 results in constant volumetric fraction and mass fraction of each phase at the interface, respectively. This implies that at each interface, the following conditions hold:

$$\frac{\alpha_l}{\alpha_g + \alpha_l} = \text{constant} \xrightarrow{\alpha_l + \alpha_g = 1} \alpha_{l,g} = \text{constant}, \tag{3.66a}$$

$$\frac{\alpha_l\rho_l}{\alpha_l\rho_l+\alpha_g\rho_g}=\text{constant}\xrightarrow{\alpha_{l,g}=\text{constant}}\frac{\rho_l}{\rho_g}=\text{constant}. \tag{3.66b}$$

This assumption may restrict the applicability of the method to systems with smooth solutions; however, the analysis of the applicability is beyond the scope of this study. Again we note that Assumption 3.16 is only used for obtaining the average source terms and not applied into the DFM (3.15) itself.

Frictional source terms

Considering only friction, from (3.19c), we have:

$$m_{l_0}^2\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\alpha_l\rho_l}\right)+m_{g_0}^2\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\alpha_g\rho_g}\right)+\frac{\mathrm{d}p}{\mathrm{d}x}=-32\frac{(\alpha_l\mu_l+\alpha_g\mu_g)}{d^2}(\alpha_l u_l+\alpha_g u_g). \tag{3.67}$$

A similar approach to (3.47) cannot be followed here to find similar algebraic relations. Therefore, simplifying Assumption 3.16 (equation (3.66)) is considered. Following [124], rearranging (3.67) and having the constant variables of (3.66) in mind, we have:

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\rho_g}\left(\frac{m_{l_0}^2}{\alpha_l\dfrac{\rho_l}{\rho_g}}+\frac{m_{g_0}^2}{\alpha_g}+p\rho_g\right)\right)=-32\frac{(\bar{\alpha_l}\mu_l+\bar{\alpha_g}\mu_g)}{d^2}\frac{1}{\rho_g}\left(\frac{\bar{m_l}}{\overline{\left(\dfrac{\rho_l}{\rho_g}\right)}}+\bar{m}_g\right), \tag{3.68}$$

where $\bar{(\cdot)}$ represents an average of the variable over the left and right side of the interface, which is defined similar to (3.48). After some steps of straightforward computations and integration over each interface, we obtain:

$$-A\left[\ln\rho_g\right]+\frac{c_g^2}{2}\left[\rho_g^2\right]=B, \tag{3.69}$$

where

$$A=\frac{m_{l_0}^2}{\alpha_l\dfrac{\rho_l}{\rho_g}}+\frac{m_{g_0}^2}{\alpha_g},\quad B=-32\frac{(\bar{\alpha_l}\mu_l+\bar{\alpha_g}\mu_g)}{d^2}\left(\frac{\bar{m_l}}{\overline{\left(\dfrac{\rho_l}{\rho_g}\right)}}+\bar{m}_g\right). \tag{3.70}$$

From (3.70), the term B can be computed. By substituting the value of B in (3.69), the value of A can be computed. Moreover, for computing $\bar{F}$ in (3.65c) in the case of only friction using:

$$m_{l_0}^2\left[\frac{1}{\alpha_l\rho_l}\right]+m_{g_0}^2\left[\frac{1}{\alpha_g\rho_g}\right]+[p]=\bar{F}\Delta x,$$

the values of m_{l0} and m_{g0} are required. By knowing the value of A, taking into account the structure of A specified in (3.70) and assumptions (3.66), $\bar{F}$ is obtained as follows:

$$\bar{F}\Delta x = \frac{A}{(\rho_g)_R} - \frac{A}{(\rho_g)_L} + [p]. \tag{3.71}$$

Finally, we obtain

$$\bar{F}\Delta x = \left[\frac{1}{\rho_g}\right]\left(\frac{c_g^2}{2}\left[\rho_g^2\right] - B\right)\frac{1}{[\ln \rho_g]} + [p]. \tag{3.72}$$

Proposition 3. *Under the assumption of smooth (steady and transient) solutions, $\bar{F}$ obtained in* (3.72) *is approximately equivalent to the actual friction* (3.17a)*.*

Proof. For smooth solution, there exists $(\rho_g)_i$ with $|(\rho_g)_i - (\rho_g)_L| < |(\rho_g)_R - (\rho_g)_L|$ such that

$$[\ln \rho_g] = \frac{(\rho_g)_R - (\rho_g)_L}{(\rho_g)_i}. \tag{3.73}$$

Substituting (3.73) and EOS (3.18c) to (3.72) and carrying out some straightforward simplification leads to:

$$\begin{aligned}\bar{F}\Delta x =& -\frac{c_g^2}{2}\left((\rho_g)_R^2 - (\rho_g)_L^2\right)\frac{(\rho_g)_R - (\rho_g)_L}{(\rho_g)_R(\rho_g)_L}\frac{1}{[\ln \rho_g]} + c_g^2((\rho_g)_R - (\rho_g)_L) + \\ & B\frac{(\rho_g)_R - (\rho_g)_L}{(\rho_g)_R(\rho_g)_L}\frac{(\rho_g)_i}{(\rho_g)_R - (\rho_g)_L} \\ =& c_g^2((\rho_g)_R - (\rho_g)_L)\left(1 - \frac{\left((\rho_g)_R^2 - (\rho_g)_L^2\right)\left((\rho_g)_R - (\rho_g)_L\right)}{2(\rho_g)_R(\rho_g)_L\left[\ln \rho_g\right]}\right) + \\ & B\frac{(\rho_g)_i}{(\rho_g)_R(\rho_g)_L}.\end{aligned} \tag{3.74}$$

By increasing the number of grid cells, $(\rho_g)_R, (\rho_g)_L, (\rho_g)_i \to \rho_g$. Using this and also substituting the expression of B from (3.70) leads to:

$$\bar{F}\Delta x = \frac{1}{\rho_g}B = -32\frac{(\bar{\alpha}_l\mu_l + \bar{\alpha}_g\mu_g)}{d^2}\Delta x\left(\frac{\bar{m}_l}{\rho_g\overline{\left(\frac{\rho_l}{\rho_g}\right)}} + \frac{\bar{m}_g}{\rho_g}\right) \approx -32\frac{\mu_{\text{mix}}}{d^2}u_{\text{mix}}\Delta x. \tag{3.75}$$

□

After obtaining the friction contribution, a similar treatment is applied for finding the contribution from the gravitational term.

Gravitational source terms

If gravity is the only source term, from (3.19c), we have:

$$m_{l0}^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\alpha_l \rho_l}\right) + m_{g0}^2 \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{1}{\alpha_g \rho_g}\right) + \frac{\mathrm{d}p}{\mathrm{d}x} = -(\alpha_l \rho_l + \alpha_g \rho_g) g \sin\theta. \tag{3.76}$$

Similar to the friction source term and using Assumption 3.16 (equation (3.66)), we obtain:

$$\frac{A}{2}\left[\frac{1}{\rho_g^2}\right] + c_g^2 \left[\ln \rho_g\right] = B, \tag{3.77}$$

where

$$A = \frac{m_{l0}^2}{\alpha_l \dfrac{\rho_l}{\rho_g}} + \frac{m_{g0}^2}{\alpha_g}, \; B = -\left(\bar{\alpha}_l \overline{\left(\frac{\rho_l}{\rho_g}\right)} + \bar{\alpha}_g\right) g \sin\theta \Delta x. \tag{3.78}$$

All terms including $\bar{(\cdot)}$ are defined similar to the average momentum in Euler equations by the structure introduced in (3.48). Finally, by computing A and knowing that $m_{l0}^2 \left[\frac{1}{\alpha_l \rho_l}\right] + m_{g0}^2 \left[\frac{1}{\alpha_g \rho_g}\right] + [p] = \bar{G}\Delta x$, we obtain:

$$\bar{G}\Delta x = \frac{A}{(\rho_g)_R} - \frac{A}{(\rho_g)_L} + [p]. \tag{3.79}$$

Finally, we obtain:

$$\bar{G}\Delta x = \frac{2}{\left[\dfrac{1}{\rho_g^2}\right]} \left(B - c_g^2 \left[\ln \rho_g\right]\right) \left[\frac{1}{\rho_g}\right] + [p]. \tag{3.80}$$

Proposition 4. *Under the assumption of smooth (steady and transient) solution, $\bar{G}$ in* (3.80) *is approximately equivalent to the actual gravitation source term* (3.17b).

Proof. For smooth solution, there exists $(\rho_g)_i$ with $|(\rho_g)_i - (\rho_g)_L| < |(\rho_g)_R - (\rho_g)_L|$ such that

$$\left[\ln \rho_g\right] = \frac{(\rho_g)_R - (\rho_g)_L}{(\rho_g)_i}. \tag{3.81}$$

Substituting (3.81) and EOS (3.18c) to (3.80) and carrying out some straightforward simplification leads to:

$$\begin{aligned}\bar{G}\Delta x &= 2B\frac{(\rho_g)_R(\rho_g)_L}{(\rho_g)_R+(\rho_g)_L} - 2c_g^2\frac{(\rho_g)_R(\rho_g)_L}{(\rho_g)_R+(\rho_g)_L}\frac{(\rho_g)_R-(\rho_g)_L}{(\rho_g)_i} + c_g^2\left((\rho_g)_R-(\rho_g)_L\right)\\ &= 2B\frac{(\rho_g)_R(\rho_g)_L}{(\rho_g)_R+(\rho_g)_L} + c_g^2\left((\rho_g)_R-(\rho_g)_L\right)\left(1-\frac{2(\rho_g)_R(\rho_g)_L}{(\rho_g)_i\left((\rho_g)_R+(\rho_g)_L\right)}\right).\end{aligned} \tag{3.82}$$

By increasing the number of grid cells, $(\rho_g)_R, (\rho_g)_L, (\rho_g)_i \to \rho_g$. By using this and also substituting the expression of B from (3.78) leads to:

$$\bar{G}\Delta x = B\rho_g = -\left(\bar{\alpha}_l\rho_g\overline{\left(\frac{\rho_l}{\rho_g}\right)} + \rho_g\bar{\alpha}_g\right)g\sin\theta \approx -\rho_{mix}g\sin\theta. \tag{3.83}$$

This proves that $\bar{G}$ is an approximation of (3.17b). □

Remark 3.17. *Assumption 3.16 is adopted only to find expressions for the average friction and gravity source terms. These are not applied to the DFM* (3.15) *and, from now on, these assumptions are only hidden in $\bar{F}$ and $\bar{G}$ and do not affect the structure of the governing equations.*

The average source term is now specified as $\bar{Q} = \bar{F} + \bar{G}$. Following the same steps introduced in the previous section for Euler equations, we can solve the following equations, consisting of three consistency equations, one algebraic steady-state condition and eight closure laws. Following the same line of reasoning for Euler equations, after imposing $\lambda := \lambda_L = \lambda_R$, $m_i^* := (m_i)_R^* = (m_i)_L^*$ with $i \in \{l, g\}$, we have

$$(\alpha_l\rho_l)_R^* + (\alpha_l\rho_l)_L^* = (\alpha_l\rho_l)_R + (\alpha_l\rho_l)_L - 3\frac{[m_l]}{\lambda}, \tag{3.84a}$$

$$(\alpha_g\rho_g)_R^* + (\alpha_g\rho_g)_L^* = (\alpha_g\rho_g)_R + (\alpha_g\rho_g)_L - 3\frac{[m_g]}{\lambda}, \tag{3.84b}$$

$$\begin{aligned}&\frac{1}{2}\left(m_l^{*2}\left(\frac{1}{(\alpha_l\rho_l)_R^*}-\frac{1}{(\alpha_l\rho_l)_L^*}\right)+m_g^{*2}\left(\frac{1}{(\alpha_g\rho_g)_R^*}-\frac{1}{(\alpha_g\rho_g)_L^*}\right)+(p_R^*-p_L^*)\right)\\ &-\lambda(m_l^*+m_g^*)+\frac{\lambda}{2}\left((m_l+m_g)_R+(m_l+m_g)_L\right)=\\ &\frac{3}{2}\left[\frac{m_l^2}{\alpha_l\rho_l}+\frac{m_g^2}{\alpha_g\rho_g}+p\right]-2\bar{Q}\Delta x,\end{aligned} \tag{3.84c}$$

$$\begin{aligned}&\frac{-\bar{m}_l^2}{(\alpha_l\rho_l)_R(\alpha_l\rho_l)_L}\left((\alpha_l\rho_l)_R^*-(\alpha_l\rho_l)_L^*\right)-\frac{-\bar{m}_g^2}{(\alpha_g\rho_g)_R(\alpha_g\rho_g)_L}\left((\alpha_g\rho_g)_R^*-(\alpha_g\rho_g)_L^*\right)+\\ &(p_R^*-p_L^*)=\bar{Q}\Delta x,\end{aligned} \tag{3.84d}$$

$$(\alpha_l + \alpha_g)^*_R = 1, \tag{3.84e}$$
$$(\alpha_l + \alpha_g)^*_L = 1, \tag{3.84f}$$
$$p^*_R = \left((\rho_l)^*_R - \rho_0\right) c_l^2 + p_0, \tag{3.84g}$$
$$p^*_L = \left((\rho_l)^*_L - \rho_0\right) c_l^2 + p_0, \tag{3.84h}$$
$$p^*_R = (\rho_g)^*_R c_g^2, \tag{3.84i}$$
$$p^*_L = (\rho_g)^*_L c_g^2, \tag{3.84j}$$
$$\frac{m^*_g}{(\alpha_g\rho_g)^*_L} = \frac{K\dfrac{m^*_l}{(\rho_l)^*_L} + S}{1 - K(\alpha_g)^*_L}, \tag{3.84k}$$
$$\frac{m^*_g}{(\alpha_g\rho_g)^*_R} = \frac{K\dfrac{m^*_l}{(\rho_l)^*_R} + S}{1 - K(\alpha_g)^*_R}. \tag{3.84l}$$

The short-hand notation, for instance, $(\alpha_l\rho_l)^*_R = (\alpha_l)^*_R(\rho_l)^*_R$ is used to compact the equations. These twelve equations can be solved simultaneously to compute the intermediate primitive variables, $(\alpha_l)^*_R$, $(\alpha_g)^*_R$, $(\alpha_l)^*_L$, $(\alpha_g)^*_L$, $(\rho_l)^*_R$, $(\rho_g)^*_R$, $(\rho_l)^*_L$, $(\rho_g)^*_L$, p^*_R, p^*_L, m^*_l, m^*_g. The entire procedure is summarized in Algorithm 2. By running Algorithm 2 for both (W_L, W_R) at $x_{i+1/2}$ and at $x_{i-1/2}$, the variables that should be substituted into (3.26) are computed.

Now, all the required components for implementing the modified scheme (3.26) are available and numerical simulations can be obtained.

Remark 3.18. *Before going through the numerical examples, it should be noted that the numerical steady-state solution is calculated point-wise at the center of the cells while finite-volume solution is the average of solutions over a cell.*

Algorithm 3: WB-DFM: Well-balanced scheme for the DFM

Input: W_L, W_R at each interface
Output: W^*_L, W^*_R at the interface

1 Compute primitive variables from conservative variables at the left and right side of the interface,
2 Similar to (3.48), calculate $\overline{\left(\dfrac{\rho_l}{\rho_g}\right)}, \bar{\alpha}_l, \bar{\alpha}_g, \bar{m}_l$ and $\bar{m}_g$
3 Compute $\bar{F}$ and $\bar{G}$ via (3.72) and (3.80), respectively,
4 Set $\bar{Q} = \bar{F} + \bar{G}$,
5 Solve the system (3.84a)-(3.84l) simultaneously to obtain the intermediate primitive variables,
6 Set $W^*_L = \left[(\alpha_l\rho_l)^*_L,\ (\alpha_g\rho_g)^*_L,\ m^*_l + m^*_g\right]^T$ and $W^*_R = \left[(\alpha_l\rho_l)^*_R,\ (\alpha_g\rho_g)^*_R,\ m^*_l + m^*_g\right]^T$.

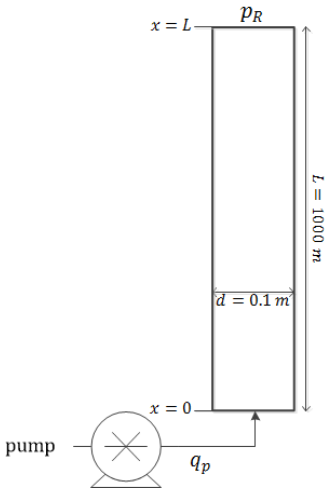

Figure 3.1: Configuration of the computational domain.

However, point-wise values are very close to the average values in the test cases of this paper as the solution within each grid cell can be approximated by a line.

Remark 3.19. *The numerical steady-state reference solution is obtained at the finite-volume centers, as is the case for the numerical finite-volume solutions.*

3.4 Numerical results for single-phase flow

In this section, numerical results of a single-phase flow inside a pipe are shown. First, preservation of the steady-state solution is considered and then, a transient simulation from an initial steady-state to another steady-state is carried out. The values of the parameters involved in system (3.1) are listed in Table 3.1 (q_p and p_R are the volumetric flow rate of the pump at the left boundary and pressure at the right boundary, respectively). Figure 3.1 shows the computational domain, which is a vertical pipe with a constant cross-sectional area with the left boundary at the bottom and the right boundary at the top of the pipe. For all simulations in this section, we set $L = 1000$ m and $\Delta x = 10$ m.

Table 3.1: Parameters for the test case of the single-phase flow.

Parameter	ρ_0	p_0	θ	p_R	c	g	μ
Value	1000 kg/m^3	1 bar	90°	1 bar	1000 m/s	9.81 m/s^2	0.5 Pa.s

Remark 3.20. *The Rusanov scheme is subject to the CFL condition*

$$\Delta t = CFL \frac{\Delta x}{\max(|2\lambda_i^n|, i = 1, \cdots, N, n = 1, \cdots, N_t)}, \tag{3.85}$$

where λ_i^n *is given by* (3.22) *and* N *is the number of grid cells and* N_t *is the number of time-steps. For all simulations in this section, we estimate* $\max(|2\lambda_i^n|) = c + 10$ *and* $CFL = 0.99$*. Then, according to the chosen* Δx*,* Δt *is specified by* (3.85)*.*

3.4.1 Preservation of the steady-state solution

In this section, the performance of the well-balanced scheme in terms of the accuracy of the steady-state profile is evaluated in case of zero and non-zero flow. The initial condition in the test cases of this section coincides with the steady-state solution. Results are also compared against the classical Rusanov scheme. Moreover, we set $T = 10$ s. After this time instant, the solution varies negligibly over time (the spatial 2-norm of the solution varies less than 0.1% relatively over time), indicating that the solution has reached its steady state.

3.4.1.1 Zero flow

Results of simulating a steady system with zero flow are shown in Figures 3.2 and 3.3. The differences between the steady-state velocity and the numerical velocity of both the well-balanced and the classical schemes at the right boundary are depicted over time in Figure 3.4. This example bears relevance with the connection scenario commonly performed in the drilling context, which is shutting down the pump, waiting for the drilling liquid to become stagnant, and adding a new stand of pipe to the current configuration before resuming drilling ahead.

In the left plot of Figure 3.2, the classical scheme gives a linear change of mass flow over the spatial domain in the steady state. This is in contrast with the physical steady-state solution, i.e., the mass flow rate should be constant over the spatial length. The slope of the line in that plot is significantly smaller for the well-balanced scheme, close to zero as the physical governing equations show. However, the pressure in the well-balanced scheme deviates slightly from the steady-state solution, in an extent comparable to the classical scheme. As shown in Figure 3.4, the error for approximating the velocity at the right boundary in the well-balanced scheme decreases as time evolves. Nonetheless, this error remains unchanged for the classical scheme.

Most notably, the classical scheme implies that zero mass enters from the left boundary and approximately 0.004 kg per second exits the pipe from the right boundary (note that in Figure 3.2, momentum, ρu, at the right boundary is around 0.5 kg/m^{2}s and $\dot{m} = \rho u A = 0.5 \times \frac{\pi}{4} d^2 = 0.004$ kg/s), meaning that

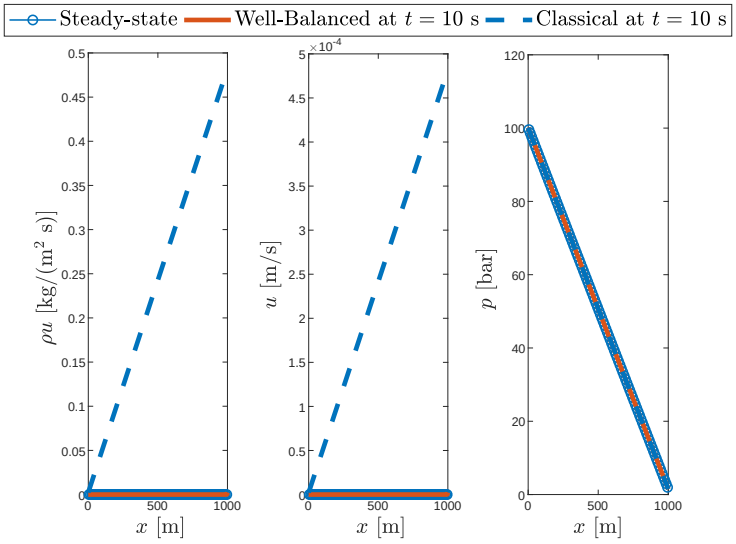

Figure 3.2: Comparison of solutions of the well-balanced and the classical scheme in capturing steady state with zero flow at the left boundary, Euler equations.

the mass inside the pipe decreases 0.004 kg per second, contrary to the physical governing condition of the test case.

3.4.1.2 Non-zero flow

For simulation scenario with non-zero flow, we set $q_p = 2000$ l/min. Simulating such a system leads to the results shown in Figures 3.5 and 3.6, where the states and the error in approximating the correct steady-state solution are depicted, respectively. Evolution of the error in approximating the velocity at the right boundary for both schemes is shown in Figure 3.7 over time. This flow scenario is usually present in the pipeline networks while the flow is pumped from one location to another under a constant volumetric flow rate. This can also be observed in drilling with single-phase flow when the rate of penetration is too low.

Analyzing the results of Figures 3.5 and 3.6 reveals that, in case of non-zero flow, the well-balanced scheme always outperforms the classical scheme, such that, its error is considerably lower than that of the classical one. The most important feature of the well-balanced scheme is the preservation of the mass flow rate (see the plots related to ρu in Figures 3.5 and 3.6). Figure 3.7 shows that the well-balanced scheme always remains closer to the analytical steady-state solution, which further confirms the accuracy of this scheme.

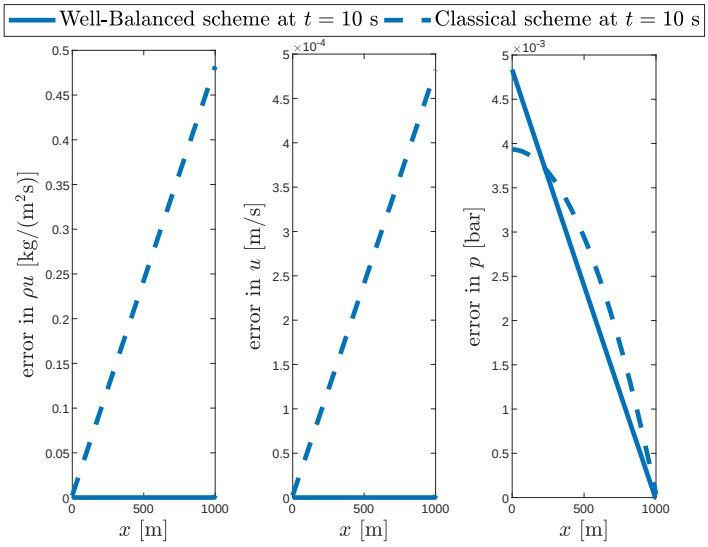

Figure 3.3: Comparison of difference between the analytical and numerical steady-state solution of the well-balanced and the classical scheme in capturing steady state with zero flow at the left boundary, Euler equations.

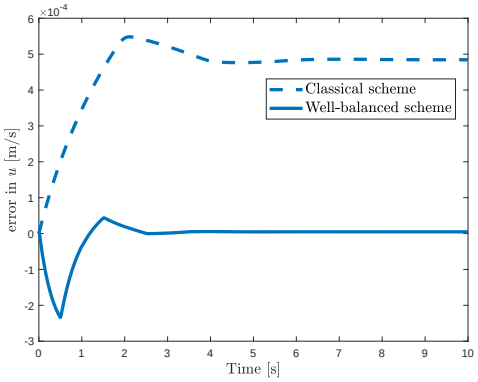

Figure 3.4: Comparison of the evolution of the error in the steady-state velocity prediction for the well-balanced and classical scheme with zero flow at the left boundary, Euler equations.

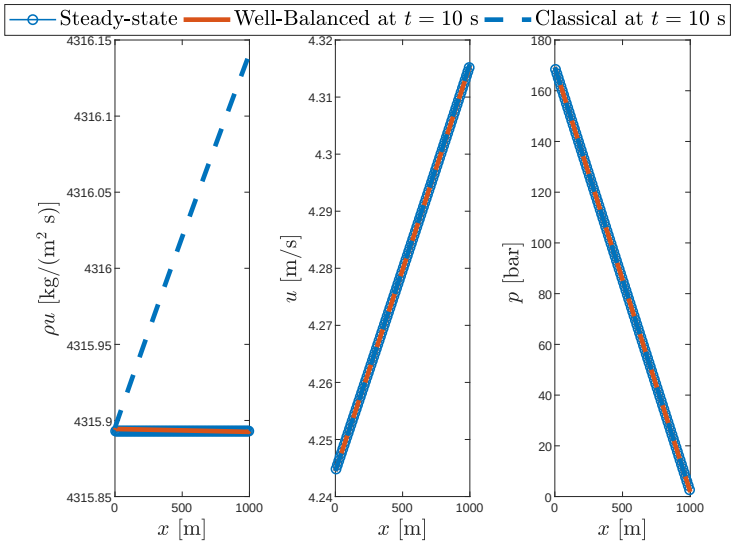

Figure 3.5: Comparison of solutions of the well-balanced and the classical scheme in capturing steady state with non-zero flow at the left boundary, Euler equations.

3.4.2 Transient simulation scenario

By observing the results of the previous section, it can be inferred that the well-balanced scheme preserves the steady-state solution with a higher accuracy compared to the classical scheme. In this section, we numerically verify that the well-balanced scheme also approximates the correct steady-state solution with a better accuracy during a transient going from one steady-state to another one by changing the inputs of the system. To this end, in the previous test case at $t = 10$ s, the right boundary pressure changes from $p_R = 1$ bar to $p_R = 10$ bar and the pump flow rate changes from $q_p = 2000$ l/min to $q_p = 4000$ l/min and the dynamics is simulated until $T = 100$ s. After this time instant, the state variables of the system do not vary with time, i.e., the system has reached its steady state. This example shows a set-point change in the pipeline networks or in a drilling operation.

Figures 3.8 and 3.9 show the initial condition and the final steady-state solution and the numerical steady-state solution of both schemes and the associated error, respectively. Results show that the well-balanced scheme converges to the steady states with an error significantly smaller than the classical scheme. The approximation error of the velocity at the right boundary is depicted in Figure 3.10.

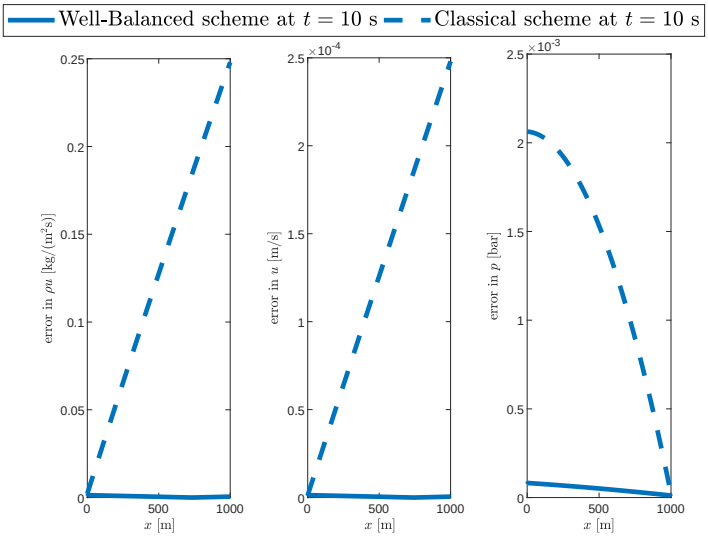

Figure 3.6: Comparison of difference between the analytical and numerical steady-state solution of the well-balanced and the classical scheme in capturing steady state with non-zero flow at the left boundary, Euler equations.

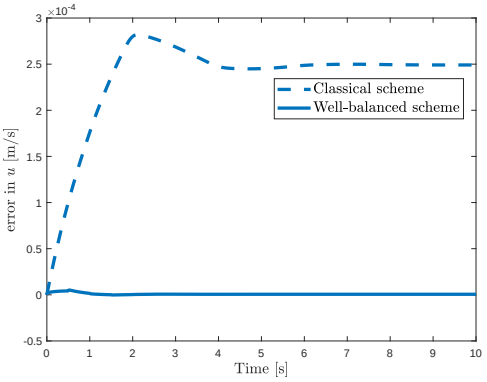

Figure 3.7: Comparison of the evolution of the error in the steady-state velocity prediction for the well-balanced and classical scheme with non-zero flow at the left boundary, Euler equations.

The well-balanced scheme captures the new steady-state solution with a lower error. The small deviation of the well-balanced solver from the steady-state

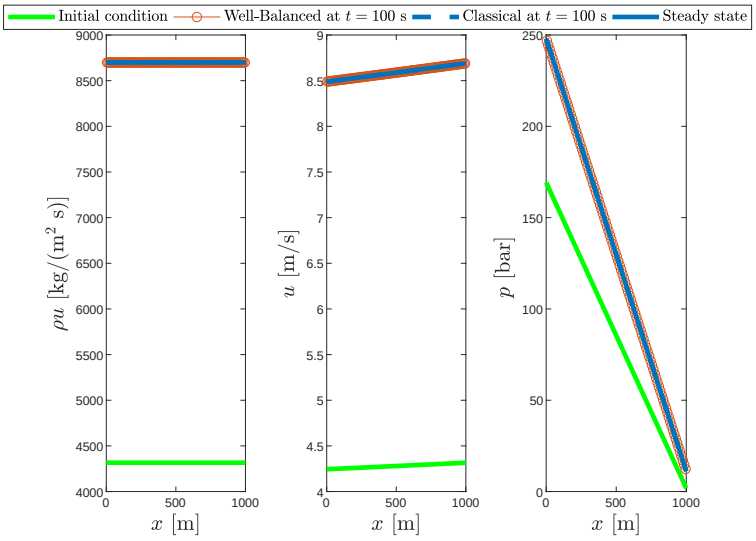

Figure 3.8: Comparison of solutions of the well-balanced and the classical scheme in capturing new steady state, Euler equations.

solution is due to the simplifying assumption made in Assumption 3.16 and also the simplification mentioned before (3.62). The difference of the accuracy of both solvers is more striking if we highlight the source contributions by increasing the nominal density of the liquid (ρ_0) or viscosity of the fluid. The same procedure is carried out for the DFM in the next section.

3.5 Numerical Results for two-phase flow

In this section, numerical results of a two-phase flow inside a pipe, as illustrated in Figure 3.11, are shown. Similar to Section 3.4, preservation of the steady-state solution and then a transient simulation are provided. The values of the parameters involved in system (3.15) are summarized in Table 3.2. It should be mentioned that $\dot{m}_{l,g} = \alpha_{l,g}\rho_{l,g}u_{l,g}A$ and p_R are the mass flow rate of the liquid and the gas at the left boundary and pressure at the right boundary, respectively, and A is the cross-sectional area of the pipe.

Figure 3.11 shows the computational domain which is a vertical pipe with a constant cross-sectional area. For all simulations in this section, we set $L = 1000$ m and $\Delta x = 10$ m. For the temporal discretization, we use the CFL definition (3.85) by estimating $\max(|2\lambda_i^n|) = c_l$ and imposing CFL $= 0.99$.

As it has been illustrated in [69], the speed of sound in the mixture appeared in (3.23) affects the solution significantly. As we want to focus on the perfor-

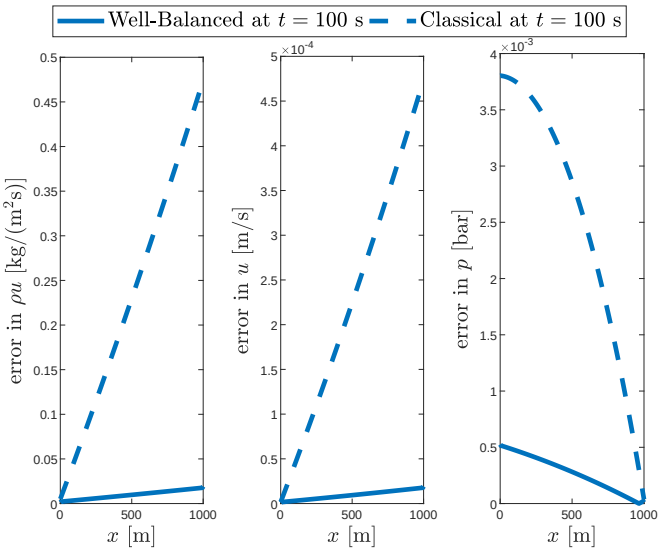

Figure 3.9: Comparison of errors between analytical and numerical steady-state solution of the well-balanced and the classical scheme in capturing new steady state, Euler equations.

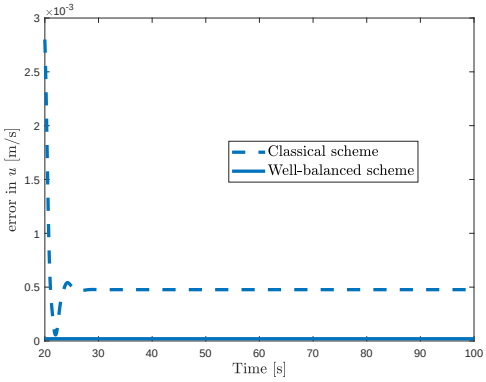

Figure 3.10: Comparison of the evolution of the error in velocity prediction for the well-balanced and classical scheme in capturing new steady state, Euler equations.

mance of the well-balanced scheme compared to the classical scheme, we do not want any other errors rather than source-related errors to affect the numerical

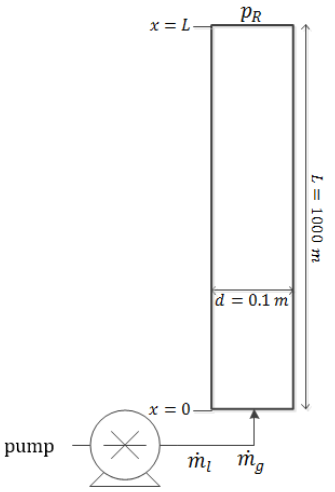

Figure 3.11: Configuration of the computational domain for the DFM.

solution. As the analytical speed of sound is available only in case of $K = 1$ and $S = 0$, these parameters are used for simulations. The effect of choosing the speed of sound model can be seen by simulating the system with different values for K and S [30], [142]. Moreover, in this section, only the values related to the liquid properties are reported in the figures since these values can represent the entire dynamics. The gas-related properties can be easily obtained from the liquid-related values by using the closure laws (3.18).

3.5.1 Preservation of the steady state

For the case of zero flow in a two-phase system in a vertical pipe, due to the density difference of the two phases, the gaseous phase migrates up the pipe and the liquid goes down along the pipe in a non-horizontal pipe; then, we cannot have a mixture of the gas and liquid for such a pipe. Moreover, the slip law does not permit the separation of the gas and the liquid since the velocity of the gas and liquid have to be equal. Thus, the system does not have any solution for stationary case of a non-horizontal pipe containing two-phase flow governed by

Table 3.2: Test case parameters.

Parameter	ρ_0	p_0	$\dot{m}_l$	$\dot{m}_g$	p_R	K
Value	1000 kg/m^3	1 bar	0.3 kg/s	0.03 kg/s	1 bar	1
Parameter	c_g	c_l	g	μ_g	μ_l	S
Value	316 m/s	1000 m/s	9.81 m/s^2	5e-4 Pa.s	0.5 Pa.s	0

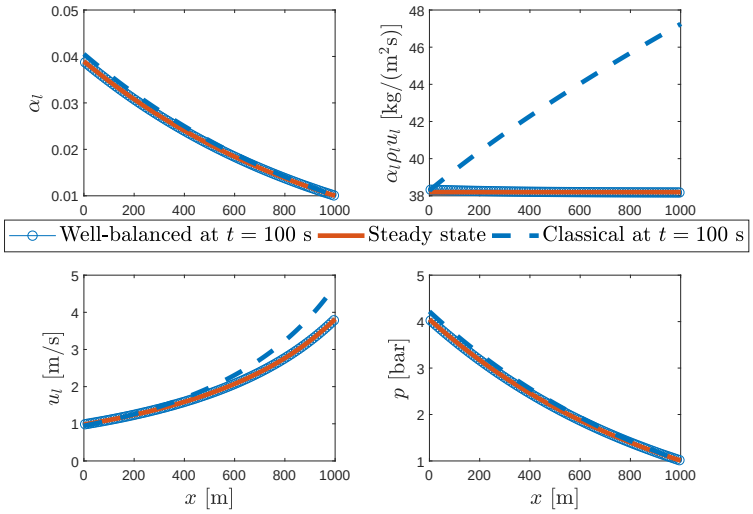

Figure 3.12: Comparison of solutions of the well-balanced and the classical scheme in capturing steady state with non-zero flow at the left boundary, the DFM.

the DFM without any slip. For the horizontal pipe, both the gravitational and the friction term are zero and no source term affects the solution. Hence, in this section, unlike the single-phase flow, only the case of nonzero flow is studied for assessing the ability of the scheme to retain the equilibrium profile of the system. Again, the initial condition for the test cases in this section is the steady-state solution of the system.

3.5.1.1 Nonzero flow for a vertical pipe

Parameter values are mentioned in Table 3.2; besides that, T is set to 100 s. Comparison of the steady-state solution and the numerical solutions obtained from the well-balanced and the classical solvers are depicted in Figure 3.12 and the difference between these solutions and the steady-state solution is shown in Figure 3.13. Also, the time evolution of the error in approximating the velocity at the right boundary is shown in Figure 3.14. This scenario can occur in the two-phase pipelines as well as the flow inside a well-bore during an under-balanced drilling operation [2].

It can be observed that the well-balanced scheme performs better than the classical scheme in terms of remaining on the steady-state profile when starting from the steady initial condition. The effectiveness of the well-balanced scheme

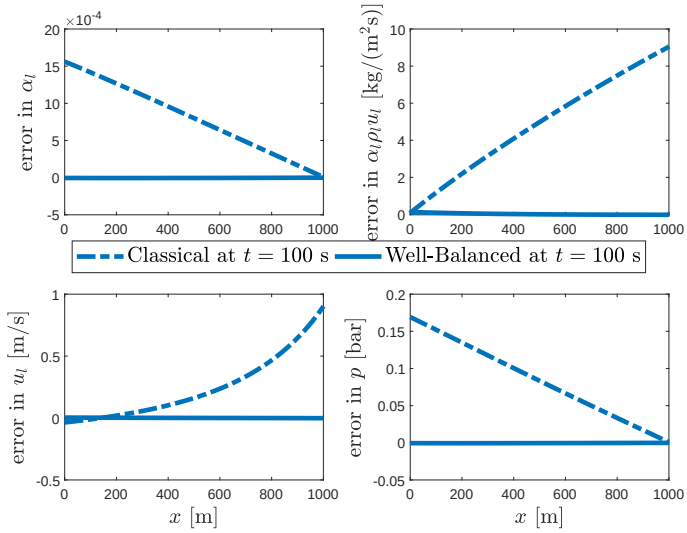

Figure 3.13: Comparison of errors between the analytical and numerical steady-state solution of the well-balanced and the classical scheme in capturing steady state with non-zero flow at the left boundary, the DFM.

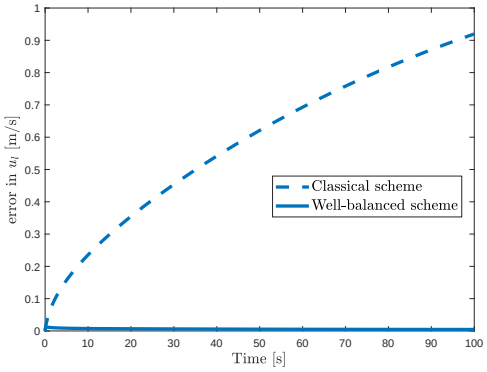

Figure 3.14: Comparison of the evolution of the error in the velocity prediction for the well-balanced and classical scheme in capturing steady state with non-zero flow at the left boundary, the DFM.

can be better observed in the plots related to mass flow rate of the liquid ($\alpha_l\rho_l u_l$ in Figures 3.12 and 3.13). Apparent from these figures, the classical scheme

generates a non-physical steady-state solution. Pressure and velocity predicted by the classical scheme deviate from the actual values significantly. Again, in the steady state, the classical scheme predicts that 0.3 kg of the liquid enters the pipe per second and around 0.37 kg of the liquid exits the pipe, which is in contrast with the physical governing equations. The results of the well-balanced scheme and the actual steady-state solution correspond to each other with a high accuracy.

Now, the scheme can be tested under transient simulations by going from one steady initial condition to another steady-state solution.

3.5.2 Transient simulation scenario

Analogous to the case of single-phase flow, boundary conditions are changed to excite the transients of the systems. At $t = 1$ s, the values of the mass flow rates are doubled instantly and the right boundary pressure changes to $p_R = 2$ bar slowly over 10 s. This set-point change can also happen in the two-phase pipelines and also as a control action to harness the gas migration in a drilling well.

The results acquired at time instant $T = 500$ s are shown in Figures 3.15, 3.16 and 3.17. These figures reveal that the well-balanced scheme approximates the steady-state solution of the system with a higher accuracy compared to the classical scheme. As the system approaches to the steady condition, the errors in all state variables also tend to zero over time.

Results presented in Sections 3.4 and 3.5 confirm that the well-balanced schemes yield more accurate results in both preserving and approximating the steady-state solution compared to the classical scheme. Moreover, the modified schemes have also been tested in the case of no source terms (the flow with zero viscosity in a horizontal pipe), which has shown that it produces the same results as the classical scheme (these results are not included in the paper). Now, we can analyze the order of accuracy for the proposed scheme.

3.6 Order of accuracy for the modified scheme

In order to study the error convergence of the proposed scheme, a norm of the difference between the steady-state solution and the numerical steady-state solution generated by the proposed schemes is computed. The error-norm is velocity-based and it is defined as below:

$$e = \frac{\left(\sum_{i=1}^{N} |u(x_i, T) - u_{ss}(x_i)|^r \right)^{\frac{1}{r}}}{\left(\sum_{i=1}^{N} |u_{ss}(x_i)|^r \right)^{\frac{1}{r}}}, \tag{3.86}$$

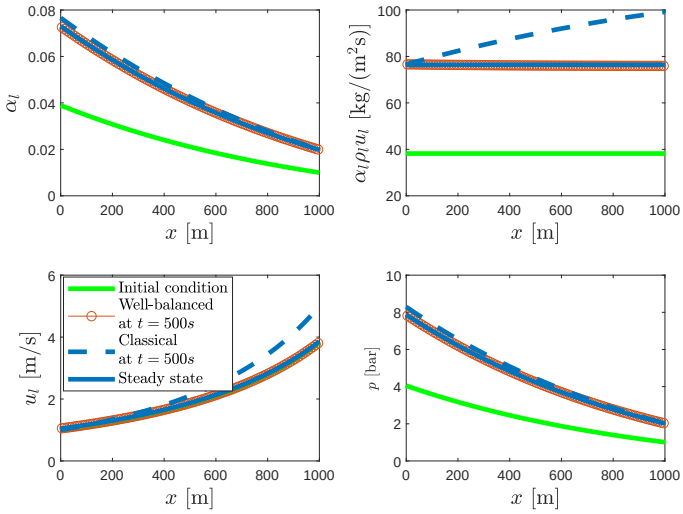

Figure 3.15: Comparison of solutions of the well-balanced and the classical scheme in capturing new steady state, the DFM.

where N is the number of the grid cells, T, as mentioned earlier, is the last time instant of the simulation and u_{ss} is the steady-state value of the velocity. In this study, we set $r = \{2, \infty\}$, meaning that we calculate the spatial $2-$norm and $\infty-$norm of the difference between the trustworthy and the numerical steady-state solution divided by the spatial $2-$norm or $\infty-$norm of the trustworthy steady-state solution. In this section, we denote e_2 for (3.86) calculated by $r = 2$ and e_∞ for (3.86) calculated by $r = \infty$. Simulations are performed for the single- and two-phase flow ($u := u_g = u_l$) and the number of grid cells is varied to analyze the dependency of (3.86) on the number of grid cells.

3.6.1 Single-phase flow

The parameter settings in Section 3.4.1.2 are used for this section. Table 3.3 illustrates the dependency of the steady-state error measure in (3.86) on the number of grid cells for both the classical and well-balanced (WB in Table 3.3) schemes. In addition, the CPU time for carrying out these simulations are also reported in this table. The first-order convergence of the classical Rusanov scheme is clear [105], as by doubling the number of the grid cells, the error is divided by two. Moreover, the error associated with the well-balanced scheme with 100 grid cells is much less than the error of the classical scheme with 1600 grid cells. By extrapolation, the classical scheme with $12,800$ grid cells yields

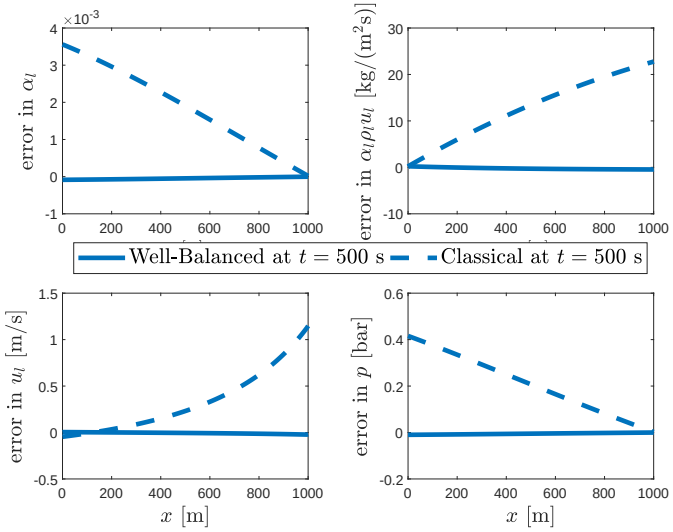

Figure 3.16: Comparison of difference between analytical and numerical steady-state solution of the well-balanced and the classical scheme in capturing new steady state, the DFM.

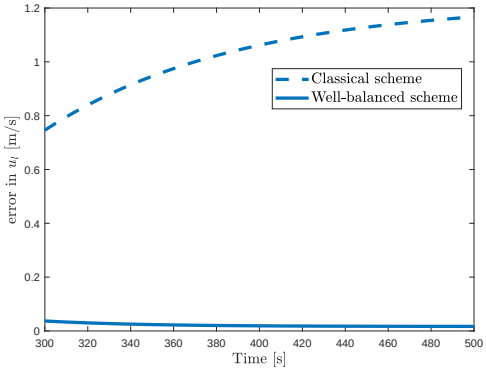

Figure 3.17: Comparison of error evolution for approximating velocity of the well-balanced and classical scheme in capturing new steady state, the DFM.

the same accuracy as the well-balanced scheme with 100 grid cells while the well-balanced scheme with 100 grid cells is less expensive than using classical scheme even with 400 grid cells. This clearly shows the superiority of the well-

balanced scheme over the classical scheme in correct steady-state calculation. The proposed scheme is close to first-order accurate. To be precise, its accuracy is of order of 0.82 on average (see Figure 3.18). The main reason for this lower convergence rate can be attributed to the assumption $\lambda := \lambda_L = \lambda_R$; it seems that by increasing the number of grid cells, the error related to this assumption does not reduce linearly and therefore decreases the overall convergence rate of the proposed scheme. Another reason can be the point-wise calculation of the reference solution, rather than the average solution over a grid-cell. Approximating the averaged source terms as the summation of the averaged frictional and gravitational source terms and also the fact mentioned in Remark 3.9 contribute to this behavior. The data shown in Table 3.3 together with the order of accuracy can be seen in Figure 3.18. It can be observed that by increasing the number of grid cells, the incremental order of accuracy (the order of accuracy in each step of doubling the number of grid cells) does not follow a specific trend. By going from 100 to 200 grid cells, the error drops even more than a first-order accurate scheme. Then the order of accuracy decreases and then increases.

3.6.2 Two-phase flow

The parameter settings in Section 3.5.1.1 are used for this section. The effect of increasing the number of the grid cells on capturing the steady-state solution of the DFM is reported in Table 3.4 together with the CPU time allocated for the simulations.

The first-order accuracy of the classical Rusanov scheme can easily be interpreted from Table 3.4. Moreover, the error associated with the well-balanced scheme with 100 grid cells is much less than the error of the classical scheme with 1600 grid cells. Similarly, by extrapolation, the classical scheme with $12,800$ grid cells generates the same accuracy as the well-balanced scheme with 100 grid cells while the well-balanced scheme with 100 grid cells is less expensive than

Table 3.3: Error convergence for the Euler equations together with the cpu time comparison (WB=well-balanced).

Cells	100	200	400	800	1600
e_2-classical	3.39e-5	1.69e-5	8.44e-6	4.22e-6	2.11e-6
e_2-WB	1.65e-7	7.05e-8	4.77e-8	2.8e-8	1.53e-8
Convergence rate	-	1.23	0.57	0.77	0.88
e_∞-classical	5.78e-5	2.89e-5	1.45e-5	7.21e-6	3.61e-6
e_∞-WB	3.19e-7	1.41e-7	9.21e-8	5.31e-8	2.87e-8
Convergence rate	-	1.2	0.62	0.80	0.89
cpu time classical	0.86 s	1.15 s	2.06 s	5.41 s	21.24 s
cpu time WB	1.29 s	2.07 s	5.89 s	15.36 s	52.36 s

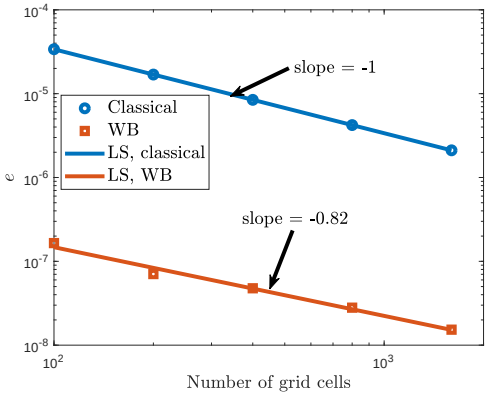

Figure 3.18: Error convergence comparison of the classical and the well-balanced scheme based on (3.86) with $r = 2$, WB: well-balanced scheme, LS: Least Square line passed through the points, Euler equations.

Table 3.4: Error convergence for the DFM together with the cpu time comparison (WB=well-balanced).

Cells	100	200	400	800	1600
e_2-classical	0.17	0.083	0.042	0.021	0.01
e_2-WB	7.65e-4	6e-4	4.2e-4	2.44e-4	1.3e-4
Convergence rate	-	0.35	0.52	0.79	0.91
e_∞-classical	0.24	0.12	0.06	0.03	0.015
e_∞-WB	12e-4	8.1e-4	5.6e-4	3.22e-4	1.7e-4
Convergence rate	-	0.57	0.54	0.81	0.93
cpu time classical	8.8 s	18 s	40 s	123.4 s	344.7 s
cpu time WB	272 s	978.2 s	3460.4 s	19,055 s	74,451 s

using classical scheme even with 1600 grid cells. Due to the nonlinearity of the equations in (3.84), the cpu time for the well-balanced scheme is higher than the classical scheme. This however can be alleviated remarkably by embedding the linear equations into the nonlinear ones among (3.84), so a smaller set of equations has to be solved at each time step. In general, the comparison of the error (3.86) shows the superiority of the well-balanced scheme over the classical scheme. However, this superiority comes at the expense of reduction in the error convergence rate; the proposed scheme is of order of 0.65 on average (see Figure 3.19). The main culprits for this reduction in the convergence rate are the assumption of $\lambda := \lambda_L = \lambda_R$ and the assumption (3.66). Approximating the averaged source terms as the summation of the averaged frictional and gravi-

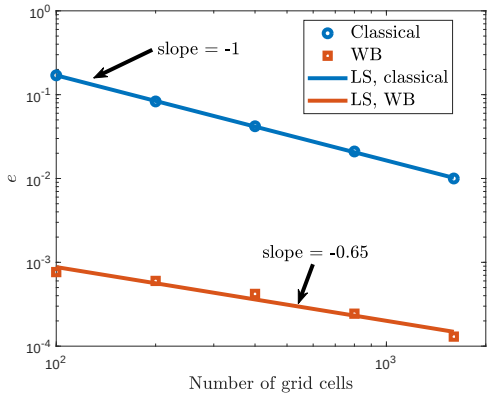

Figure 3.19: Error convergence comparison of the classical and the well-balanced scheme based on (3.86) with $r = 2$, WB: well-balanced scheme, LS: Least Square line passed through the points, the DFM.

tational source terms and also the fact mentioned in Remark 3.9 contribute to this behavior. The data shown in Table 3.4 together with the order of accuracy can be seen in Figure 3.19. As can be observed in this figure, by increasing the number of grid-cells, the incremental order of accuracy increases and becomes closer to 1. This can be attributed to the fact that by increasing the number of grid cells, the conservative and primitive variables vary even more smoothly from one grid cell to another and the set of assumptions (3.66) becomes more realistic. In the high number of grid cells, the error generated by the assumption $\lambda := \lambda_L = \lambda_R$ dominates the error. Similarly, point-wise calculation of the reference steady-state solution can contribute to this behavior.

3.7 Conclusion

In this paper, a novel extension of the Rusanov scheme has been proposed to improve the preservation of the steady-state solutions of Euler equations and the drift flux model. These schemes reduce to the original scheme when there is no source term. The proposed schemes capture the steady-state solution with significantly higher accuracy compared to the classical scheme in the presence of source terms. This is proved for an advection equation with a simple source term. Various test cases of zero and non-zero flow have been carried out and the improved performance of the well-balanced schemes has been shown numerically, both for single-phase and two-phase scenarios. The modification is model- and scheme-dependent, such that a similar approach can be followed for other systems of partial differential equations solved by different schemes.

The well-balanced schemes lead to physical results while this is not the case for the classical finite-volume schemes in the presence of source terms. This can be interpreted from the presented results, especially those for the mass flow rate. Both Euler equations and DFM imply that the mass flow rate along the spatial domain remains constant in the steady condition. This property is preserved with significantly higher accuracy in the proposed well-balanced schemes.

Chapter 4

Modelling and numerical implementation of managed pressure drilling systems for the assessment of pressure control systems

Automated Managed Pressure Drilling (MPD) is a method to enhance downhole pressure control performance and safety during drilling operations. It is becoming more common to use model-based simulation for the evaluation of pressure control systems designed for MPD automation before using those in the field. This demands a representative hydraulics simulation model which captures the relevant aspects of a drilling system. This paper presents such a model and, additionally, an approach to numerically implement that model for simulation studies. The complexity of this simulation model should be limited to, firstly, support effective numerical implementation and, secondly and most importantly, to allow for the analysis of the behaviour and performance of the automated pressure control systems during the controller design phase. To this end, aspects of a drilling system that can considerably affect the performance of the automated MPD system are captured in the model. This hydraulics model incorporates both the distributed and multi-phase flow nature of a drilling system. Moreover, it captures nonlinear boundary conditions at the inlet of the drillstring, at the drill bit and choke manifold, and also the variations in the cross-sectional area of

This chapter is based on "S. Naderi Lordejani, M.H. Abbasi, N. Velmurugan, C. Berg, J.A. Stakvik, B. Besselink, L. Iapichino, F. Di Meglio, W.H.A. Schilders, N. van de Wouw, ***Modelling and numerical implementation of managed pressure drilling systems for the assessment of pressure control systems***, *SPE Drilling & Completion*, in press, 2020". Modeling, numerical implementation and simulations are performed in a joint work with S. Naderi Lordejani. The validation of the single-phase flow model against the field data is solely the contribution of S. Naderi Lordejani.

the flow path. Model validations against field data from real-life MPD operations and simulations of industry-relevant scenarios indicate that these aspects are effectively captured in the model and preserved during the numerical implementation.

4.1 Introduction

Conventionally, the task of pressure control is accomplished by changing the mud density during drilling operations. However, this method of controlling the pressure is slow and inaccurate, and it lacks a means of compensating and responding to transient pressure fluctuations (e.g., occurring during pipe connection operations or drilling into high-pressure formations). Besides, this method cannot be used for the drilling of deep wells with narrow drilling windows because of its low accuracy. To overcome such drawbacks of conventional pressure control methods, managed pressure drilling (MPD) has been introduced. A main objective of MPD is to provide a means of fast, accurate and efficient control of the bottomhole pressure (BHP), as opposed to conventional methods. As we illustrate in Fig. 4.1, in MPD, the annulus is sealed off at the top with a rotating control device to direct the mud flow from the annulus to a choke valve with a variable opening (see, e.g., [75], [162]). This equipment, which is often accompanied by a back-pressure pump, pressurizes the fluid inside the wellbore by providing an active back pressure. The back pressure, and thus the BHP, can be controlled by manipulating the choke opening. In automated MPD systems, the task of manipulating the choke opening is primarily performed by an automatic pressure control system. This enhances safety and performance, and reduces drilling time and cost, see [76]. In particular, if equipped with advanced control systems, automated MPD can make it possible to handle many well-control events automatically without operator intervention and using conventional well-control methods [23].

A control system designed for automated MPD should pass some virtual and representative test scenarios on a simulation level before it can be used in the field. This is done because any failure in the drilling system, especially in the pressure control system, can have catastrophic consequences. Training new operators for drilling operations and well control incidents in a controlled environment and also well monitoring are other important reasons for performing model-based simulation studies in drilling, [40], [176]. However, simulations performed for training purposes often need to be well-supported by graphical interfaces, which is not the case when it comes to controller design. These simulations rely on a mathematical model of the drilling system dynamics, the complexity of which varies depending on the required purpose. In particular, the complexity of an MPD model developed for testing pressure controllers should be limited to facilitate the performance analysis and design of the control system by neglecting less important system aspects. Such a model, called the simula-

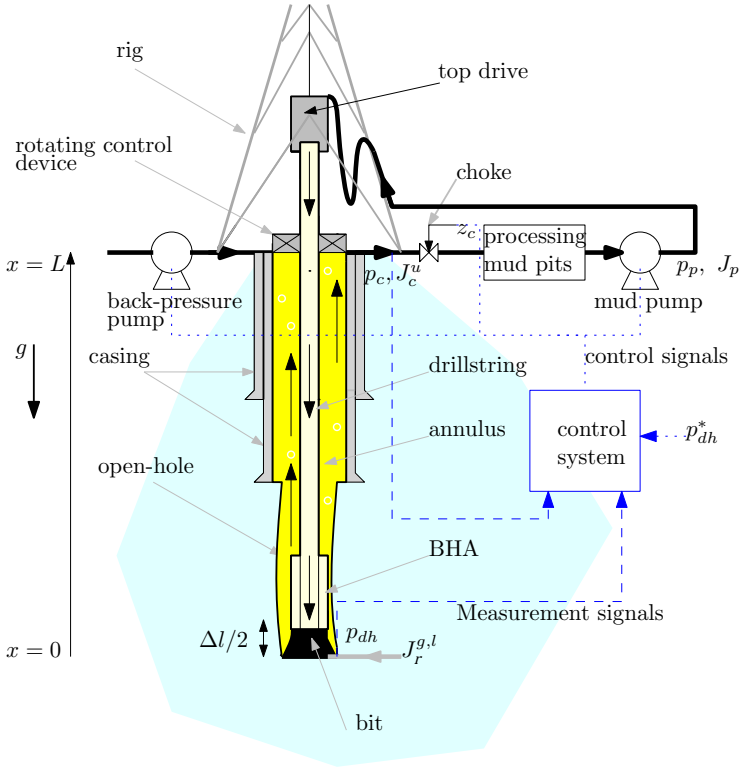

Figure 4.1: A simplified schematic diagram of a drilling system with MPD equipment.

tion model in this context, should only contain: 1) aspects of a drilling system considered in the controller designed and, 2) aspects that are not considered in the controller design but can have detrimental effects on the performance of the controller. Models used in the controller design are called the design models. A design model is often much simpler than a simulation model, as it usually contains only the mass transport dynamics, neglecting the distributed nature of drilling systems; see, e.g., [2], [60], [99], [134], [140]. Below are listed a number of the drilling aspects that can be detrimental to an automated MPD system.

- Pressure wave propagation: pressure controllers are usually developed based on simplified models in which pressure dynamics (i.e., wave propagation effects) are ignored partially [109] or totally [2], [12], [60], [99], [131], [140], [162]. The essential time scale associated to these dynamics can be in the range of tens of seconds, or even minutes in the case of gas influx into the wellbore. Not only can such dynamics deteriorate the control per-

formance, but they can also cause instability [141], if not accounted for during the controller design stage;

- Dynamics of the flow in the drillstring: in many cases, e.g., in [1], the dynamics of the flow in the drillstring (as opposed to the flow through the annulus), whether fast or slow, are ignored throughout the controller design stage. This part can, however, have a significant contribution to the system behaviour, e.g., by changing the location of the major resonance frequencies of the system or generating additional resonance frequencies. Thus, the closed-loop performance in practice can be worse than expected from simulation studies if system aspects imposed by the drillstring are ignored in the design model;

- Nonlinear behaviour of the drill bit: once the drillstring flow path is ignored in the design model, the drill bit is replaced by an independent source of flow. The flow through the bit is, however, nonlinearly dependent on the pressure drop over the bit. In particular, in the case of standpipe pressure control during gas influxes, this nonlinearity can be detrimental.

- The variable structure (i.e., switching nature) of the model: this variable structure is induced mainly by the presence of a non-return valve in the bottom-hole assembly. During operations such as pipe connection, the non-return valve usually remains closed, changing the system properties and behaviour;

- Lastly, variations in the cross-sectional area of the flow path: these variations, especially those in the annulus, may have significant contribution to frequency responses of a drilling system. Therefore, if not included in the design model, these aspects may compromise the control performance.

Other dynamical effects, such as temperature transients, evolve so slowly, see, e.g., Fig. 14 in [40], that these can be neglected in the simulation model. These effects may be modelled in terms of uncertainties in the system parameters. Effects such as well expansion can also be approximately lumped into the system parameters, see,e.g., [22].

In this paper, we rely on physical simplifications to derive a simulation model for MPD which is consistent with the modelling choices above. The simulation of the resulting model is based on a numerical discretization method. As these numerical methods are incapable of exactly preserving all the characteristics of the model, particular care should be taken in choosing, developing and using these numerical methods. Thus, we also provide a dedicated numerical approach for the simulation of this model.

The majority of existing hydraulics simulation models for drilling are single-phase models, often based on the (isothermal) Euler model, [108], [129], and two-phase models, which are often based on either the two-fluid model or the Drift-Flux Model (DFM), see, e.g., [4], [37], [107], [134], [157], [163], [170]. In MPD

modelling, it is key to specify the boundary conditions of the system carefully because exactly these boundary conditions differentiate MPD from conventional drilling methods and are means of applying control inputs. Moreover, one should consider the fact that the flow path in a drilling system, from the rig pump to drill bit all the way up to the choke, experiences discontinuities in its cross-sectional area. These discontinuities have a considerable contribution to the transient and steady-state behaviour of the flow and pressure along the flow path. These issues have not been addressed adequately in the literature. [107] showed, by means of simulations and experiments on test wells, the predictive capabilities of the DFM for drilling operations. Though providing an advanced numerical setting for the simulation of their model, they performed their studies in a conventional drilling setting, not MPD. They did also not consider variations in the cross-sectional area of the flow. The work by [179] is similar to [107], but the drilling model is solved with a different numerical scheme to improve the solution accuracy. [2] used the DFM as a basis for simulating the annulus, testing controllers and model complexity reduction for MPD. The issue of variations in the cross-sectional area was, however, not addressed. A similar hydraulics model was studied in [170], where only the annulus was considered in the model and an extrapolation method was used for solving the considered boundary conditions for simulations. [157] used the two-fluid model for the simulation of the flow in the annulus. [40] provided a survey on advances in drilling simulators, but no technical details were given on the underlying mathematical models. In this paper, we propose a model for two-phase MPD drilling operations based on the DFM. This model takes all the relevant aspects, mentioned above, into account.

4.1.1 Contributions.

The main contribution of this paper is a simulation platform suitable for evaluating controller performance for MPD operations, which includes both the physical model and the tool to implement it. In particular, a comprehensive formulation of a hydraulics model for MPD-relevant two-phase flow drilling scenarios is presented first. In this model, interactions between the different parts of a drilling system are formulated in terms of boundary conditions. The complexity of the model is limited to contain control relevant hydraulics aspects of a drilling system dynamics which can in some way be detrimental to the closed-loop performance of an automated MPD. The developed model allows for the simulation of many drilling scenarios ranging from making pipe connections, choke plugging and choke swapping, and bit nozzle plugging to liquid and gas influx scenarios. Then, we provide a numerical approach to support simulation tooling for fast scenario testing. In particular, we adapt a characteristics-based method to solve the nonlinear and boundary conditions, and also propose a dynamical model for the drill bit to circumvent numerical issues which appear at low pump flow rates. Since the effects of variations in the cross-sectional area of the flow path

can be significant both on transient and steady-state response of the system, we explicitly address these variations during the numerical implementations of the model. Finally, we validate the hydraulics model for relevant MPD scenarios of single-phase flow by comparing it with a set of field data obtained from commissioning tests of an automated MPD system.

4.1.2 Outline.

The next section is devoted to providing a short introduction to the DFM. Next, the mathematical modelling of the system is discussed. The steady-state solution of the model is discussed afterwards. After completing the modelling part, we present a numerical approach for the implementation of the model, which is later illustrated by means of a simulation study.

4.2 Drift-flux model

This section provides a short introduction to the DFM [74] as it is the cornerstone of the MPD model to be developed. Flow behaviour in a transmission line can, to some extent, be described by the DFM. Because of its relative simplicity yet favourable capabilities in capturing the pressure and mass transport dynamics of two-phase flows, the DFM is probably the most widely used model in literature on control and simulation of two-phase drilling scenarios [4]. Consisting of two mass conservation equations and one combined momentum conservation equation, the DFM reads as

$$\frac{\partial q}{\partial t} + \frac{\partial f(q)}{\partial x} = S, \tag{4.1}$$

with

$$\begin{aligned} q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} := \begin{bmatrix} \rho_l \alpha_l \\ \rho_g \alpha_g \\ \rho_l \alpha_l v_l + \rho_g \alpha_g v_g \end{bmatrix}, \quad f := \begin{bmatrix} \rho_l \alpha_l v_l \\ \rho_g \alpha_g v_g \\ \rho_l \alpha_l v_l^2 + \rho_g \alpha_g v_g^2 + p \end{bmatrix}, \\ S := \begin{bmatrix} \Gamma \\ -\Gamma \\ s(u,t,x) \end{bmatrix}. \end{aligned} \tag{4.2}$$

Here, $x \in (0, L)$ and $t > 0$ are the spatial and time variables, respectively, with L the length of the computational domain and the well in this case. The volume fraction, density, velocity and pressure are denoted by $\alpha = \alpha(t,x)$, $\rho = \rho(t,x)$, $v = v(t,x)$ and $p = p(t,x)$, respectively, where the subscript l denotes the liquid phase and g refers to the gas phase. The vector of primitive variables (individual variables which have a clear physical meaning) is indicated by $u = [\alpha_l, \rho_l, v_l, \alpha_g, \rho_g, v_g, p]^T$, while q represents the vector of conservative variables,

and $f(\cdot)$ is the flux function. The source term is represented by $s(u,t,x)$, and Γ_l and Γ_g model mass exchange between the phases, which are often assumed to be zero. The source term consisting of a gravitational and frictional term is given by

$$s(u,t,x) = \rho_m g \sin(\theta) - \frac{2\nu(u)\rho_m v_m |v_m|}{D}, \tag{4.3}$$

where g and $\theta(x)$ are the gravitational acceleration and the inclination of the transmission line with reference to the horizontal direction and D is hydraulic diameter. Moreover, $\rho_m = \alpha_l\rho_l + \alpha_g\rho_g$ and $v_m = \alpha_l v_l + \alpha_g v_g$ are the mixture density and velocity, respectively. Here, ν is the Fanning friction factor which is in general a challenging parameter to determine. It is well-known that drilling muds in general exhibit non-Newtonian behaviours [147]. There are a number of models describing these types of flows such as the Herschel-Bulkley, Bingham plastic and the Power Law model. Of these three, the three-parameter model of Herschel-Bulkley is the most accurate, as it includes the other two models as special cases. However, the respective equations are highly nonlinear and challenging to solve and, moreover, complex models with too many parameters are less useful from a control and estimation perspective. Here, we adapt the two-parameter Power Law model to trade off between complexity and accuracy. In this model, we define the Generalized Reynolds number as $\mathrm{Re} = \rho_m v_m D_{eff}/\mu_{m,app}$, where $D_{eff} = 4n_m D/(3n_m+1)$ is the effective diameter and

$$\mu_{m,app} = \mu_m^{n_m}\left(\frac{3n_m+1}{4n_m}\frac{8v_m}{D}\right)^{n_m-1}, \tag{4.4}$$

is the apparent mixture viscosity [147]. Here, $n_m = \alpha_l n_l + \alpha_g$, with n_l the liquid behaviour index, is the mixture behaviour index, and $\mu_m = \alpha_l m_l + \alpha_g \mu_g$, with μ_l and μ_g the liquid and gas viscosity, respectively, is the mixture viscosity. The Fanning friction factor for laminar flow, when $\mathrm{Re} < 3250 - 1150n_m$, is given by

$$\nu = \frac{16}{\mathrm{Re}}, \tag{4.5}$$

while for the turbulent flow, when $\mathrm{Re} > 4250 - 1150n_m$, ν is the solution to

$$\frac{1}{\sqrt{\nu}} + 4\log\left(\frac{0.27\varepsilon}{D_{eff}} + \frac{1.26^{n_m^{-1.2}}}{\left(\nu^{(1-\frac{n_m}{2})}\mathrm{Re}\right)^{n_m^{-0.75}}}\right) = 0, \tag{4.6}$$

where ε is the pipe roughness [147]. As can be seen, this equation is highly nonlinear and no exact explicit solution is currently available to it, thus we use an approximate solution, see Appendix C. For transition flow, when $\mathrm{Re} \in [3250, 4250] - 1150n_m$, we compute ν by a linear interpolation from (4.5) and (4.6). We stress again that one may use a different frictional model depending on the application and required accuracy, see [116] for a review of these models.

Remark 4.1. *The Power Law model is in general considered as complex when it comes to the design of controllers and estimators for MPD automation, given that it is far less common compared to simpler Newtonian models used for these purposes.*

Remark 4.2. *Using mixture parameters and variables μ_m, ρ_m and v_m is a common approach to extend liquid frictional models to two-phase flows. Likewise, we have used a mixture behaviour index n_m to be able to use the Power Law model for two-phase flow scenarios.*

The DFM is completed by four other equations to be, potentially, solvable uniquely. These, often known as the closure laws, can be expressed as follows:

$$p - p_0 - c_l^2 (\rho_l - \rho_0) = 0, \tag{4.7a}$$
$$p - c_g^2 \rho_g = 0, \tag{4.7b}$$
$$\alpha_l + \alpha_g - 1 = 0, \tag{4.7c}$$
$$v_g - v_l - \Phi(u) = 0, \tag{4.7d}$$

where p_0, ρ_0, c_l and c_g are the reference pressure, liquid reference density, sound velocity in the liquid and sound velocity in the gas, respectively. The first two equations are, respectively, known as the equations of state for the liquid and gas. An equation of state describes the state of matter in terms of physical variables such as temperature, density and pressure. Equations of state can be rather complex in general, but we here use linear variants approximating only the relation between pressure and density, as in (4.7a) and (4.7b). We note that these equations still capture the liquid and gas compressibility. Moreover, the volume balance between the phases is imposed by (4.7c), and the slip law (4.7d) describes the relative velocity between the two phases depending on the function $\Phi(\cdot)$. Here, the slip law is given by [96]

$$\Phi(u) = C_0(u) v_m + V_d(u) - v_l, \tag{4.8}$$

where the $C_0(\cdot)$, $V_d(\cdot)$ are the distribution parameter and drift velocity, respectively. Several descriptions, which are mostly obtained based on experiments and function fitting, for these parameters can be found in the literature. Depending on well conditions, mud properties and expected drilling scenarios, a particular description of these parameters can be selected to be used in the hydraulics model, assuming only bubbly and slug flows. For a review of a variety of descriptions for these parameters and their validity conditions, see [30].

It can be shown that the DFM admits three distinct eigenvalues in a wide region of the variable space, see [69]. These eigenvalues are $\lambda_1 = v_g$, $\lambda_2 = v_g - c_m$, $\lambda_3 = v_g + c_m$, with $c_m(u)$ the sound velocity in the mixture. Currently, no exact analytical expression is available for $c_m(u)$. Thus, we use an approximation of

$c_m(u)$ based on the local definition of the bulk moduli, as follows:

$$c_m(u) = \sqrt{\frac{\beta_m(u)}{\rho_m(u)}}, \tag{4.9}$$

where β_m is mixed bulk modulus, defined as $\beta_m = \frac{\beta_l \beta_g}{\alpha_l \beta_g + \alpha_g \beta_l}$, with $\beta_g = p$ and $\beta_l = c_l^2 \rho_l$ the bulk moduli of gas and liquid, respectively [1], [99].

4.3 MPD Description and Hydraulics Modelling

Consulting Fig. 4.1, an MPD system may be simply regarded as two equivalent hydraulic transmission lines (or simply pipes) which are connected through a drill bit in the middle and one of which ends up with a controllable choke valve. The exposed zone of the annulus (so-called open-hole section) is susceptible to gas and liquid influx from the surrounding formations that may potentially contain hydrocarbons. Therefore, to have a good description of the flow and pressure transients along the flow path, it is necessary to use a multi-phase flow model for the annulus. However, except in some specific drilling operations, such as operations performed in underbalanced drilling, it is quite reasonable to use a single-phase flow model to describe the flow in the drillstring.

4.3.1 Hydraulics modelling.

The DFM, as in (4.1), can be used only for the description of the flow lines with constant cross-sectional area, while in practice there are variations in the cross-sectional area of the flow path, due to changes in the diameter of pipes and open hole, that impact the flow behaviour. This urges the use of a modified version of the DFM that accounts for the variations in the cross-sectional area. The modified DFM for the annulus reads as follows [150]:

$$\frac{\partial (A_a q_a)}{\partial t} + \frac{\partial f_a(A_a q_a)}{\partial x} = A_a S_a + \frac{\partial A_a}{\partial x} P_a, \tag{4.10}$$

where $A_a(x)$ is the cross-sectional area of the annulus and we have denoted $P_a := [0, 0, p_a]^T$. In this context, a sub/superscript a refers to the annulus and a d to the drillstring. Compared to the model (4.1), changes in the cross-sectional area lead especially to the term $P_a \partial A_a / \partial x$. We should note that t, x, q_a, $f_a(\cdot)$ and S_a are the same as in (4.1) computed from the variables in the annulus. In the annulus, as illustrated in Fig. 4.1, $x = 0$ marks the well bottom and $x = L$ a point in the annulus that is in the same level as the choke. Moreover, $D_a = D_{in} - d_o$, with D_{in} the diameter of the annulus and d_o the outer diameter of the drillstring, and $\theta_a(x) = -\theta(L - x)$.

As will be explained in later sections, we need to switch between the primitive variables u_a and the conservative ones q_a to numerically solve the model under

development. Based on the closure laws in (4.7) and also the relation between the vectors q_a and u_a, as in (4.2), we derive the following relation for the pressure in terms of conservative variables:

$$p_a = \frac{-b + \sqrt{b^2 - 4d}}{2}, \tag{4.11}$$

where $b = \rho_0 c_l^2 - p_0 - q_{a,1} - c_g^2$ and $d = -(\rho_0 C_l^2 - p_0) c_g^2 q_{a,2}$. Next, given the pressure from (4.11), (4.7a) and (4.7b) can be used to compute ρ_l and ρ_g, respectively. Next, the definition of q_a leads to expressions for the volume fractions:

$$\alpha_l = \frac{q_{a,1}}{\rho_l}, \quad \alpha_g = \frac{q_{a,2}}{\rho_g}. \tag{4.12}$$

Next, we can compute v_l using

$$v_l = \frac{q_{a,3}\,(1 - C_0 \alpha_g) - q_{a,2} V_d}{q_{a,1} + C_0 \alpha_g q_{a,2}}. \tag{4.13}$$

Then from the slip law (4.7d), with (4.8), we obtain

$$v_g = \frac{C_0 \alpha_l v_l + V_d}{1 - C_0 \alpha_g}. \tag{4.14}$$

Remark 4.3. *Depending on the choice of $C_0(u)$ and $V_d(u)$, if these parameters are dependent on v_l and v_g, then* (4.13) *and* (4.14) *can become nonlinear equations with respect to v_l an v_g which need to be solved simultaneously using nonlinear solvers. However, these are often independent of v_l and v_g.*

As already mentioned, the main reason for using a two-phase model for the annulus is to enable modelling of a gas influx from the formation into the annulus. On the contrary, it is reasonable to use a single-phase model for describing the flow behaviour inside the drillstring. It is worth mentioning that there are certain drilling operations, such as operations in underbalanced drilling, where some rate of gas is intentionally injected into the drillstring. This gas injection in turn gives rise to a two-phase medium in the drillstring. The following isothermal Euler equation accounting for the variations in the cross-sectional area describes the flow behaviour in the drillstring:

$$\frac{\partial (A_d q_d)}{\partial t} + \frac{\partial f_d(A_d q_d)}{\partial x} = A_d S_d + \frac{\partial A_d}{\partial x} P_d, \tag{4.15}$$

where $f_d\,(A_d q_d) = [A_d \rho v, A_d \rho v^2 + A_d p_d]^T, S_d(u_d, t, x) = [0, s_d(u_d, t, x)]^T, P_d := [0, p_d]^T$. Moreover, $\rho = \rho(t, x), v = v(t, x)$, $p_d = p_d(t, x)$ are the mud density, velocity and pressure profiles along the drillstring. The vectors of primitive and conservative variables are indicated by $u_d = [\rho, v, p_d]^T$ and $q_d = [\rho, \rho v]^T$, respectively. Moreover, $A_d(x)$ is the cross-sectional area of the drillstring. For

the drillstring, $x = 0$ marks a point in the drillstring which is at the same level as the pump whereas $x = L$ marks its outlet at the bit. To avoid unnecessary notational complexities, we do not use a subscript to refer to the primitive variables in the drillstring, except for the pressure. As before, s_d is the source term acting on the flow in the drillstring, and the same model as in (4.3) is used to determine it, with the mixture variables and parameter reducing to liquid variables. Moreover, we have $D_d = d_{in}$, with d_{in} the inner diameter of the drillstring, and $\theta_d(x) = \theta(x)$. The equation of state considered for the liquid in the drillstring is the same as (4.7a), the one used in the annulus.

Remark 4.4. *It should be noted that (4.15) may be obtained from (4.10) by setting* $\alpha_g = 0$.

4.3.2 Boundary conditions

To potentially be able to solve (4.10) and (4.15) uniquely, one needs to specify a set of boundary conditions. In this regard, a careful observation of Fig. 4.1 reveals that the hydraulics behaviour of an MPD system is largely dictated by three main physical boundary conditions, which are the boundaries at the drillstring inlet, the bit together with the behaviour of the formations around the open-hole, and the choke valve, as the annulus outlet.

The boundary condition at the drillstring inlet is expressed in a general form as follows:

$$f_{ibc}\left(\rho\left(t,0\right),v\left(t,0\right),t\right)=0, \tag{4.16}$$

where $f_{ibc}(\cdot,\cdot,\cdot)$ is the boundary condition at the drillstring inlet, and it is determined depending on the ongoing drilling operation. For example, during normal operations, when the drillstring is connected to the mud pump, we can define

$$f_{ibc} := A_d\left(0\right)\rho\left(t,0\right)v\left(t,0\right)-J_p\left(t\right), \tag{4.17}$$

where $J_p(t)$ denotes the mass flow rate of the mud pumped into the drillstring. It should be noted that in practice we are often provided the pump strokes per minutes $n_p(t)$ rather than the mass flow rate. In that case, the mass flow rate can be computed using $J_p\left(\rho(t,0),t\right) = V_p n_p(t)\rho(t,0)$, where V_p is the volume that the pump sweeps per stroke. As another example, during a bleed-off operation, an operation to slowly release the trapped pressure within the drillstring before detaching it from the top-derive, a valve equation should be used to model this boundary condition. Next, at the bottom of the well, one can write three boundary equations, consisting of the bit equation that describes the liquid mass flow rate through the bit in terms of the pressure drop over the bit, the liquid mass balance equation between both sides of the bit and the gas balance between the formations and the annulus, respectively:

$$A_d(L)\rho(t,L)v(t,L)-c_dA_n\sqrt{2\rho(t,L)r\left(p_d(t,L)-p_a(t,0)\right)}=0, \tag{4.18}$$

$$A_d(L)\rho(t,L)v(t,L) + J_r^l(p_a(t,0),p_r) - A_a(0)\alpha_l(t,0)\rho_l(t,0)v_l(t,0) = 0, \quad (4.19)$$
$$J_r^g(p_a(t,0),p_r) - A_a(0)\alpha_g(t,0)\rho_g(t,0)v_g(t,0) = 0, \quad (4.20)$$

where the function $r(e) = \begin{cases} e, & \text{if } e > 0 \\ 0, & \text{if } e \leq 0 \end{cases}$ is used to model the non-return valve installed above the bit inside the Bottom-Hole Assembly (BHA), A_n is the effective area of the bit nozzles and c_d is the discharge coefficient. Also, $J_r^l(\cdot,\cdot)$ and $J_r^g(\cdot,\cdot)$ respectively represent the mass flow rates of the liquid and gas exchanged between the well-bore and the formations with a pressure p_r, known as the reservoir pressure. Here, we approximate these variables using a linear static reservoir model as follows (see, e.g., [3]):

$$J_r^i = \kappa_i r\left(p_r - p_a(t,0)\right), \quad i \in \{g,l\}, \quad (4.21)$$

where κ_i is the production index for the phase i. Coupling with a more intricate reservoir model is also possible and it can be substituted into (4.21). However, near well-bore reservoir modelling is out of the scope of this paper. At the top side of the annulus, the boundary condition is determined by the choke equation describing the mass flow rate of the mixture through the choke as a function of the pressure drop over the choke (see, e.g., [61])

$$\begin{aligned} 0 &= J_c(t,u_a(t,L)) - J_{bpp}(t) - J_c^u(u_a(t,L)), \\ J_c(t,u_a(t,L)) &= \sum_{i=1}^{n_c} k_{c,i}G_i(z_{c,i}(t))\sqrt{2\rho_m(t,L)r\left(p_a(t,L)-p_0\right)}, \end{aligned} \quad (4.22)$$

where $k_{c,i}$, $z_{c,i}$ and $G_i(\cdot)$ are the choke flow factor, the choke opening and the choke characteristic of the choke valve i, respectively. Here, n_c is the number of choke valves in the MPD set up and $J_{bpp}(t)$ is the mass flow rate from the back-pressure pump. Moreover, $J_c^u = A_a(L)\alpha_l(t,L)\rho_l(t,L)v_l(t,L) + A_a(L)\alpha_g(t,L)\rho_g(t,L)v_g(t,L)$ is the mass flow rate upstream the choke whereas J_c is that downstream the choke. Again, more accurate models of multi-phase flow through valve can be derived to replace (4.22).

The combination of Eqs. (4.7)-(4.22) constructs our MPD simulation model. Specifically, in this model, we have accounted for variations in the cross-sectional area of the flow path and also the nonlinear boundary conditions of an MPD system. Now the MPD model has been specified, and next the steady-state solution of the system can be found based on this model.

4.4 Steady-state solution of the model

Clearly, to be able to solve the MPD model derived in the previous section, an initial condition is required. As most of the drilling time is occupied by normal drilling operation, it is reasonable to start a simulation study from a

drilling ahead condition. In this case, the system shows a steady-state behaviour; therefore, all the derivatives with respect to the time variable t may be discarded from the Equations (4.10) and (4.15), resulting in the steady-state differential equations

$$\begin{aligned}\frac{df_a(A_a\bar{q}_a)}{dx} &= A_a\bar{S}_a + \frac{\partial A_a}{\partial x}\bar{P}_a,\\ \frac{df_d(A_d\bar{q}_d)}{dx} &= A_d\bar{S}_d + \frac{\partial A_d}{\partial x}\bar{P}_d.\end{aligned} \tag{4.23}$$

Now, using the closure laws (4.7) together with (4.23), we obtain

$$\begin{aligned}\frac{d\bar{y}_a(x)}{dx} &= M_a^{-1}(\bar{y}_a)H_a(\bar{y}_a, x),\\ \frac{d\bar{y}_d(x)}{dx} &= M_d^{-1}(\bar{y}_d)H_d(\bar{y}_d, x),\end{aligned} \tag{4.24}$$

where $\bar{y}_d(x) = [\bar{v}, \bar{\rho}]^T$, $\bar{y}_a(x) = [\bar{v}_l, \bar{v}_g, \bar{\alpha}_g, \bar{p}_a]^T$, with the a bar (i.e., $^-$) indicating the variables and vectors in the steady-state. Moreover,

$$\begin{aligned}M_a &= \begin{bmatrix} 1-\frac{\partial\Phi}{\partial\bar{v}_g} & -1-\frac{\partial\Phi}{\partial\bar{v}_l} & -\frac{\partial\Phi}{\partial\bar{\alpha}_g} & -\frac{\partial\Phi}{\partial\bar{p}_a}\\ (1-\bar{\alpha}_g)\,\bar{\rho}_l & 0 & -\bar{\rho}_l\bar{v}_l & \frac{v_l(1-\bar{\alpha}_g)}{c_l^2}\\ 0 & \bar{\alpha}_g\bar{\rho}_g & \bar{\rho}_g\bar{v}_g & \frac{\bar{\alpha}_g\bar{v}_g}{c_g^2}\\ (1-\bar{\alpha}_g)\,\bar{\rho}_l\bar{v}_l & \bar{\alpha}_g\bar{\rho}_g\bar{v}_g & 0 & 1\end{bmatrix}, \quad M_d = \begin{bmatrix}\bar{\rho} & \bar{v}\\ \bar{\rho}\bar{v} & c_l^2\end{bmatrix},\\ H_a &= \bar{S}_a - \frac{dA_a}{dx}\begin{bmatrix}0\\ \frac{(1-\bar{\alpha}_g)\bar{\rho}_l\bar{v}_l}{A_a}\\ \frac{\bar{\alpha}_g\bar{\rho}_g\bar{v}_g}{A_a}\\ 0\end{bmatrix}, \quad H_d = \bar{S}_d - \frac{1}{A_d}\frac{dA_d}{dx}\begin{bmatrix}\bar{\rho}\bar{v}\\ 0\end{bmatrix},\end{aligned} \tag{4.25}$$

with $\Phi(u)$ as in (4.8). The boundary conditions of the ordinary differential equation (4.24) are given by the physical boundary conditions (4.17), (4.18) and (4.22) in the steady-state conditions:

$$\begin{aligned}(A_d\bar{\rho}\bar{v})|_{x=0} - \bar{J}_p &= 0,\\ (A_d\bar{\rho}\bar{v})|_{x=L} - \left(A_a\,(1-\bar{\alpha}_g)\,\bar{\rho}_l\bar{v}_l\right)|_{x=0} + J_f^l\,(\bar{p}_a(0), \bar{p}_r) &= 0,\\ J_f^g\,(\bar{p}_a(0), \bar{p}_r) - (A_a\bar{\alpha}_g\bar{\rho}_g\bar{v}_g)|_{x=0} &= 0,\\ (A_d\bar{\rho}\bar{v})|_{x=L} - A_n c_d\sqrt{2\bar{\rho}(L)r\,(\bar{p}_d(L) - \bar{p}_a(0))} &= 0,\\ (A_a\bar{\alpha}_l\bar{\rho}_l\bar{v}_l + A_a\bar{\alpha}_g\bar{\rho}_g\bar{v}_g)|_{x=L} + \bar{J}_{bpp} -&\\ \sum_{i=1}^{n_c} k_{c,i}G_i(\bar{z}_{c,i})\sqrt{2\bar{\rho}_m(L)r\,(\bar{p}_a(L) - p_0)} &= 0.\end{aligned} \tag{4.26}$$

Eq. (4.24) together with (4.26) construct a two-point boundary value problem in terms of x as the independent variable.

Note that the presence of dA/dx, which contains impulses due to area discontinuities in $A(x)$, in the steady-state equations (4.24) causes discontinuities in the steady-state solution. After we have initialized our problem by solving (4.24), we start with the numerical implementation of the model in the sequel.

The developed model in this paper includes the system dynamics and aspects that are essential to the control performance. Considering only the control-relevant aspects of the system in the model keeps its complexity relatively low, such that it permits, for example, a semi-analytical assessment of its dynamical properties (an assessment that relies partially on theoretical analyses and partially on numerical analyses). For instance, semi-analytical analyses are used in [62, Chapter 1]. This type of assessment can be computationally expensive, if not impossible in the case of high-complexity models. Moreover, simulation studies performed during the controller design might need to be performed several times. Thus, it is important for these simulations to run fast. Moreover, the relatively low complexity of the model will allow designers and engineers to more easily identify the reason for or the source of problems in the case of poor simulation results.

4.5 Numerical implementation

The MPD model derived in the modelling section cannot be solved analytically, due to its complexity (e.g., infinite-dimensional nature and nonlinearities). To solve and then use this model for simulation purposes, we employ a numerical scheme based on a finite-volume method (FVM) discretization. As illustrated in Fig. 4.2, in FVM [113], the spatial domain of a hyperbolic partial-differential equation (PDE) is divided into a finite number of control volumes or cells. All the variables are assumed to have a predefined distribution in each control volume. As illustrated in the figure, the spatial domain is discretized into N cells denoted by $G_i = (x_{i-1/2}, x_{i+1/2})$, $i = \{1, 2, ..., N\}$, of length Δx, with $x_{i+1/2} = i\Delta x$ called the i^{th} cell interface and $x_i = (i - 1/2)\Delta x$ marking the middle point of that cell. The variable U_i^n (Q_i^n) is an approximation of the spatial average of the vector $u(n\Delta t, x)$ ($q(n\Delta t, x)$) over G_i and the approximate variables at right and left sides of each interface are indicated by U^- and U^+, respectively. Here, Δt is the time discretization step length. A finite volume Godunov-type method has the following general form [113]:

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}\left(F\left(U_{i+1/2}^{+*}, U_{i+1/2}^{-}\right) - F\left(U_{i-1/2}^{+}, U_{i-1/2}^{-*}\right)\right) + \Delta t S_i^n, \quad (4.27)$$

where $F(\cdot,\cdot)$, a conventional numerical flux function assuming a fixed cross-sectional area, is determined by the numerical scheme, and $S_i^n = S(U_i^n, n\Delta t, x_i)$ is the discretized source term. A starred variable U^*, yet to be computed, is an update of the variable U which accounts for variations in the cross-sectional area of the flow. Note that, since the same formula as in (4.27) is used for

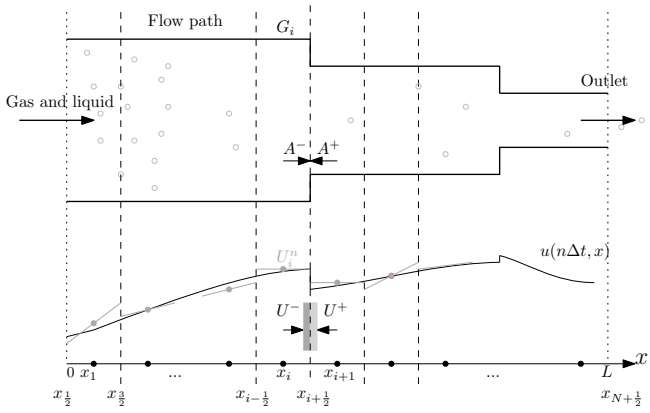

Figure 4.2: An illustration of a spatial discretization performed in FVM.

solving both of (4.10) and (4.15), the sub/superscripts a and d are omitted for readability. The time step size Δt is determined based on the Courant-Friedrichs-Lewy (CFL) condition [55]. In particular, the numerical implementation of the DFM (4.1) has been extensively studied in the last few decades (see, e.g., [71]). These studies mostly aim at developing accurate but computationally low-cost numerical schemes for computing the numerical flux function.

To treat the variations in cross-sectional area of the flow path of the MPD model, the method proposed by [103] for (4.15) and an extension of that method proposed by [10] for the modified DFM (4.10) are exploited here to compute the starred variables in (4.27). In this method, we use a coordinate transformation that gives the equivalent values of the variables in a cell if the geometry of that cell changes. In particular, when updating the variables in the cell G_i, we assume that the cells G_{i-1} and G_{i+1} have the same geometries as G_i and use this transformation to compute the equivalent of the variables in those cells, considering this change of geometry. In this way, we are still able to use a Godunov setting with numerical flux functions as in (4.27) to numerically solve our MPD simulation model. Following this approach, $U_d^{+*} = [\rho^{+*}, v^{+*}]^T$ of the interface $i + 1/2$ in the drillstring is obtained through a nonlinear coordinate transformation that is given in terms of the following system of equations:

$$\begin{aligned} \rho^{+*} v^{+*} A_d^- - \rho^+ v^+ A_d^+ &= 0, \\ \left(v^{+*}\right)^2 - \left(v^+\right)^2 + c_l^2 \ln\left(\frac{\rho^{+*}}{\rho^+}\right)^2 &= 0, \end{aligned} \tag{4.28}$$

where A^- and A^+ denote, as illustrated in Fig. 4.2, the cross-sectional area A immediately at the left and right side of an interface, respectively. Moreover, U_d^{-*} of the interface $i-1/2$ is obtained by replacing the superscripts $+$ and $-$ by

$-$ and $+$, respectively, in (4.28) and then solving the resulting equation. Next, U_a^{+*} for the DFM of the annulus is obtained by a coordinate transformation given by the following equations together with closure relations (4.7a)-(4.7d) and (4.8):

$$\begin{aligned}
\left(\alpha_l^{+*}\rho_l^{+*}v_l^{+*}\right)A_a^- - \left(\alpha_l^{+}\rho_l^{+}v_l^{+}\right)A_a^+ &= 0,\\
\left(\alpha_g^{+*}\rho_g^{+*}v_g^{+*}\right)A_a^- - \left(\alpha_g^{+}\rho_g^{+}v_g^{+}\right)A_a^+ &= 0,\\
\frac{\alpha_l^{+*}\rho_l^{+*}}{\rho_g^{+*}}\left(\frac{v_l^{+*}}{2}\right)^2 + \alpha_g^{+*}\left(\frac{v_g^{+*}}{2}\right)^2 - & \\
\frac{\alpha_l^{+}\rho_l^{+}}{\rho_g^{+}}\left(\frac{v_l^{+}}{2}\right)^2 - \alpha_g^{+}\left(\frac{v_g^{+}}{2}\right)^2 + c_g^2\ln\left(\frac{\rho_g^{+*}}{\rho_g^{+}}\right)^2 &= 0.
\end{aligned} \tag{4.29}$$

As before, U_a^{-*} of the interface $i-1/2$ is obtained by replacing $+$ and $-$ in (4.29) with $-$ and $+$, respectively.

Remark 4.5. *A close observation of* (4.28) *reveals that variations in the pressure profile $p_d(t,x)$ in the drillstring due to changes in the cross-sectional area at the location of these changes are not significant. This is mainly because the mud velocity v in the drillstring is far smaller than the sound velocity c_l. On the contrary, when there is gas inside the annulus, variations in the cross-sectional area can cause considerable variations in the pressure. Given this explanation, without losing much accuracy, we may assume that $\rho^{+*} = \rho^{+}$ for the drillstring and solve only the mass-balance equation of* (4.28)*, which is linear, for computing v^{+*}.*

After computing Q_i^{n+1} from (4.27), the vector of primitive variables are computed using (4.11)-(4.14). Then, if a first-order scheme is used, a uniform distribution is considered for the variables $u(n\Delta t, x)$ over a cell G_i, thus

$$\hat{u}(n\Delta t, x) = U_j^n, \quad x \in (x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}). \tag{4.30}$$

When a second-order scheme is used, this approximation is obtained by a linear interpolation as follows:

$$\hat{u}(n\Delta t, x) = U_j^n + (u_x)_j^n\,(x - x_i)\,, \quad x \in (x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}), \tag{4.31}$$

where $(u_x)_j^n$ is an approximation of the exact derivatives $\partial u(n\Delta t, x)/\partial x$ at $x = x_j$, computed using a flux limiter, see, e.g., [165].

Remark 4.6. *We reasonably assume that $A(x)$ is piecewise continuous, with a discontinuity occurring only at a cell interfaces such that $A(x)$ is constant in each cell G_i (i.e., for $x \in (x_{i-1/2}, x_{i+1/2})$, $i \in \{1, 2, ..., N\}$).*

In the next part of this section, we explain how to combine the implicit boundary conditions of the problem with the numerical scheme used for updating the internal domains of the model.

4.5.1 Boundary conditions treatment.

Expanding the scheme (4.27) for $i = 1$ and $i = N$ reveals explicit dependencies on U_0 and U_{N+1}, and also implicit dependencies on U_{-1}, U_{N+2} for the case of second-order schemes. These variables are required to incorporate the boundary conditions of the boundary value problem in the described scheme in the previous section. Extrapolation is a common method for determining the boundary variables. It is effective and can provide accurate results if there are no source terms in the model [144]. However, it lacks a sound theoretical support and can cause large spikes in the solution in the presence of source terms. Contrary to extrapolation, a method known as the characteristics-based method [51] offers more accurate and reliable solutions and it also has a more reliable theoretical foundation. This method involves breaking a two-point boundary value problem into two initial value problems and solving those separately at their respective boundaries. Now, by the use of a nonlinear coordinate transformation and approximations, the DFM with the closure laws (4.7a)-(4.7d) can be decomposed into its characteristic equations. In this form, two of the PDEs describe the propagation of the pressure waves, also called fast dynamics of the DFM, inside the domain and one PDE, called the slow dynamics, describes the migration of the gas phase. For the DFM (characterizing flow the annulus), these relations come in the following form, see [71]:

$$\alpha_g(1 - C_0\alpha_g)\frac{d_1 p_a}{d_1 t} + p_a \frac{d_1}{d_1 t}\alpha_g = 0, \tag{4.32}$$

$$\frac{d_2 p_a}{d_2 t} - \rho_l c_m \left(v_g - v_l\right) \frac{d_2 \alpha_g}{d_2 t} - \rho_l \alpha_l \left(v_g - v_l + c_m\right) \frac{d_2 v_l}{d_2 t} = \left(v_g - v_l + c_m\right) s_a, \tag{4.33}$$

$$\frac{d_3 p_a}{d_3 t} + \rho_l c_m \left(v_g - v_l\right) \frac{d_3 \alpha_g}{d_3 t} - \rho_l \alpha_l \left(v_g - v_l - c_m\right) \frac{d_3 v_l}{d_3 t} = \left(v_g - v_l - c_m\right) s_a, \tag{4.34}$$

where in this case we have defined $\frac{d_i}{d_i t} := \frac{\partial}{\partial t} + \lambda_{a,i}\frac{\partial}{\partial x}$, $i = 1, 2, 3$, which is a directional derivative along the vector $V = [1, \lambda_{a,i}]^T$, with $\lambda_{a,i}$ being an eigenvalue of the DFM of the annulus. Eqs (4.32)-(4.34) correspond to the gas volume wave travelling at a speed $\lambda_{a,1} = v_g$ downstream the annulus, the pressure waves propagating at $\lambda_{a,2} = v_l - c_m$ upstream the annulus and the pressure wave travelling at a speed of $\lambda_{a,3} = v_l + c_m$ towards the choke, respectively. The characteristic relations of the isothermal Euler equation describing the single-phase flow in the drillstring are given by

$$\frac{\partial p_d}{\partial t} - \lambda_{d,1}\frac{\partial p_d}{\partial x} - c_l\rho\frac{\partial v}{\partial t} - c_l\rho\lambda_{d,1}\frac{\partial v}{\partial x} = c_l s_d, \tag{4.35}$$

$$\frac{\partial p_d}{\partial t} + \lambda_{d,2}\frac{\partial p_d}{\partial x} + c_l\rho\frac{\partial v}{\partial t} + c_l\rho\lambda_{d,2}\frac{\partial v}{\partial x} = -c_l s_d, \tag{4.36}$$

where (4.35) corresponds to the pressure wave travelling upstream the flow with a velocity of $\lambda_{d,1} = v - c_l$, while (4.36) corresponds to the pressure wave travelling the opposite direction at a velocity of $\lambda_{d,2} = c_l + v$.

Remark 4.7. *Note that the characteristic relations* (4.32)-(4.34) *are obtained under several simplifying assumption such as an incompressible liquid phase,* $\partial C_0(u)/\partial u = 0$ *and* $\partial V_d(u)/\partial u = 0$*, and that* $\alpha_g \rho_g \ll \alpha_l \rho_l$ *holds, because otherwise the derivation of such relations is highly challenging, if not impossible. Naturally, these assumptions lead to some degree of inaccuracy in the computation of the boundary variables. Nonetheless, for small gas volume fractions* α_g *(less than 0.25) at the boundaries, these assumption are rather realistic, especially when managed pressure drilling operations are supported by high-performance kick detectors and pressure control systems that prevent large gas kicks. It is mentioned that the relations* (4.35) *and* (4.36) *are exact.*

In the remainder of this section, we propose a method for computing the boundary variables based on the characteristic relations (4.32)-(4.36) together with the physical boundary conditions introduced in the previous section.

We solve the drillstring inlet boundary condition in (4.16) together with the characteristic relation (4.35), forming an initial value problem as

$$\begin{aligned} f_{ibc}(\rho(t,0), v(t,0), t) &= 0, \\ \frac{\partial p_d}{\partial t} - \lambda_{d,1}\frac{\partial p_d}{\partial x} - c_l\rho\frac{\partial v}{\partial t} - c_l\rho\lambda_{d,1}\frac{\partial v}{\partial x} &= c_l s_d. \end{aligned} \tag{4.37}$$

This partial differential algebraic equation, if solved at $x = 0$, gives $u_d(t,0) = [v(t,0), \rho(t,0)]^T$. Finding the analytical solution of this nonlinear partial differential algebraic equation is however challenging. Therefore, we solve (4.37) numerically by performing a first-order Euler discretization (both spatial and temporal) on (4.37), yielding

$$\begin{aligned} f_{ibc}(\rho_0^n, v_0^n, n\Delta t) =& 0, \\ \frac{p_{d,0}^n - p_{d,0}^{n-1}}{\Delta t} - c_l\rho_1^{n-1}\frac{v_0^n - v_0^{n-1}}{\Delta t} =& -\left(\lambda_{d,1}\right)_1^{n-1}\frac{p_{d,0}^{n-1} - p_{d,1}^{n-1}}{\Delta x} - \\ & c_l\left(\rho\lambda_{d,1}\right)_1^{n-1}\frac{v_0^{n-1} - v_1^{n-1}}{\Delta x} + c_l s_d\left(U_{d,1}^{n-1}, (n-1)\Delta t, 0\right), \end{aligned} \tag{4.38}$$

where the variables ρ_0^n and v_0^n, the solutions of (4.38), are approximations of $\rho(n\Delta t, 0)$ and $v(n\Delta t, 0)$, respectively, and the notation $(\cdot)_j^i$ stands for the term inside the parentheses evaluated at the time instance i and the point x_j in the special domain. The resulting nonlinear algebraic equation (4.38), together with the equation of state (4.7a), can be solved with a proper zero-finder algorithm, such as a Newton solver. After this equation is solved, we can compute $U_{d,0}^n = [v_0^n, \rho_0^n]^T$ and $Q_{d,0}^n = [\rho_0^n, \rho_0^n v_0^n]^T$, as required in (4.27) for the drillstring. At the bottom of the well, we take a similar approach. At this boundary,

there are couplings between the boundary variables in the annulus and those in the drillstring, and nine unknown boundary variables in total with seven boundary equations and closure laws. These together with the characteristic equations (4.36) and (4.33) construct an initial value problem that is solved numerically at this boundary to compute the boundary vectors $U^n_{d,N+1}$ and $U^n_{a,0}$ (and $Q^n_{d,N+1}$ and $Q^n_{a,0}$). However, using a numerical method similar to the one in the previous section is computationally expensive, because it involves solving a nine-dimensional nonlinear algebraic equation. Moreover, it can cause numerical oscillations when the flow through the bit is close to zero, which is the case during, e.g., a pipe connection operation. Therefore, here we connect the two boundaries at both sides of the bit through an intermediate ordinary differential equation as follows (For the derivation, see Appendix D):

$$\dot{z}(t) \simeq \begin{cases} \xi\left(z(t), \Delta p_{dh}(t)\right), \text{for} & z(t) > 0, \\ \max(0, \xi\left(z(t), \Delta p_{dh}(t)\right)), \text{for} & z(t) = 0, \end{cases} \tag{4.39}$$

where $\Delta p_{dh}(t) = p(t, L - \Delta l/2) - p_a(t, \Delta l/2)$ and

$$\xi = \frac{2A_d(L)A_a(0)}{\Delta l(A_d(L) + A_a(0))}\left(\Delta p_{dh} - \frac{z^2}{2\rho(t, L - \Delta l/2)(A_a c_d)^2} + \frac{\bar{s}_a}{A_d(0)} + \frac{\bar{s}_d}{A_a(L)}\right). \tag{4.40}$$

The operator $\max(\cdot, \cdot)$ in (4.39) is used to account for the non-return valve installed in the drillstring to prevent a back-flow from the annulus into the drillstring and we take Δl is a parameter that determines the inertia of the dynamics of $z(t)$. Now, using the other characteristic relation in the drillstring and performing an Euler discretization over space and time, we can approximately compute the drillstring boundary variables at the bit by solving

$$\begin{aligned} z^n - A_d(L)\rho^n_{N+1}v^n_{N+1} =& 0, \\ \frac{p^n_{d,N+1} - p^{n-1}_{d,N+1}}{\Delta t} - c_l\rho^{n-1}_N\frac{v^n_{N+1} - v^{n-1}_{N+1}}{\Delta t} =& (\lambda_{d,2})^{n-1}_N\frac{p^{n-1}_{d,N} - p^{n-1}_{d,N+1}}{\Delta x} + \\ & c_l\left(\rho\lambda_{d,2}\right)^{n-1}_N\frac{v^{n-1}_N - v^{n-1}_{N+1}}{\Delta x} - c_l s_d(U^{n-1}_{d,N}, (n-1)\Delta t, L - \Delta x/2), \end{aligned} \tag{4.41}$$

where v^n_{N+1} and ρ^n_{N+1} are approximations of the boundary variables $v(t, L)$, $\rho(t, L)$, respectively, and z^n, an approximation of $z(n\Delta t)$, is obtained from the time discretization of (4.39) using an Euler method, i.e.,

$$\begin{aligned} \chi^j &= \chi^{j-1} + \begin{cases} \Delta t_z \xi\left(\chi^{j-1}, \Delta p^{n-1}_{dh}\right), \text{for} & \chi^{j-1} > 0, \\ \Delta t_z \max(0, \xi\left(\chi^{j-1}, \Delta p^{n-1}_{dh}\right)), \text{for} & \chi^{j-1} = 0, \end{cases} \quad j = \{1, ..., m\}. \\ z^n &:= \chi^m, \end{aligned} \tag{4.42}$$

Here, χ is an auxiliary variable such that $\chi^0 = z^{n-1}$ and Δt_z is the corresponding discretization time step length. Note that, to avoid numerical oscillations when the time step Δt is large, we can design Δt_z to be smaller than Δt. To this end, we set $\Delta t_z = \Delta t/m$, $m \in \mathbb{N}$, where m is chosen to be large enough.

Analogously, we can compute $U^n_{a,0}$, as an approximation of $u_a(n\Delta t, 0)$, by solving the following system of nonlinear algebraic equations as a result of spatial and temporal discretization of (4.33) using the Euler method with (4.19)-(4.20):

$$
\begin{aligned}
z^n + J^l_r(p^n_{a,0}, p_r) - A_a(0)\left(\alpha_l \rho_l v_l\right)^n_0 &= 0,\\
J^g_r(p^n_{a,0}, p_r) - A_a(0)\left(\alpha_g \rho_g v_g\right)^n_0 &= 0,\\
\frac{p^n_{a,0} - p^{n-1}_{a,0}}{\Delta t} - \left(\rho_l c_m \left(v_g - v_l\right)\right)^{n-1}_1 \frac{\alpha^n_{g,0} - \alpha^{n-1}_{g,0}}{\Delta t} -&\\
\left(\rho_l \alpha_l \left(v_g - v_l + c_m\right)\right)^{n-1}_1 \frac{v^n_{l,0} - v^{n-1}_{l,0}}{\Delta t} &= -\left(\lambda_{a,2}\right)^{n-1}_1 \frac{p^{n-1}_{a,1} - p^{n-1}_{a,0}}{\Delta x} +\\
&\left(\lambda_{a,2}\rho_l c_m \left(v_g - v_l\right)\right)^{n-1}_1 \frac{\alpha^{n-1}_{g,1} - \alpha^{n-1}_{g,0}}{\Delta x} +\\
&\left(\lambda_{a,2}\rho_l \alpha_l \left(v_g - v_l + c_m\right)\right)^{n-1}_1 \frac{v^{n-1}_{l,1} - v^{n-1}_{l,0}}{\Delta x} +\\
&\left(v_g - v_l + c_m\right)^{n-1}_1 s_a(U^{n-1}_{a,1}, (n-1)\Delta t, 0).
\end{aligned}
\tag{4.43}
$$

Note that the above equations need to be solved together with the closure laws to return (generally) a unique solution. After solving (4.41) and (4.43), we can compute $Q^n_{d,N+1}$ and $Q^n_{a,0}$. At the choke boundary, the initial value problem consists of the choke equation (4.22), and all closure laws of the DFM (4.7a)-(4.7d) together with the two characteristic equations (4.32) and (4.34). This problem is approximated in terms of a nonlinear algebraic equation, similar to (4.38), using a first-order Euler discretization over space and time domains. The solution of the resulting algebraic equation is then used to compute $U^n_{a,N+1}$ and $Q^n_{a,N+1}$.

Remark 4.8. *If a second-order scheme is used, in addition to the boundary vectors $U_{a,0}$ and $U_{a,N+1}$, the vectors $U_{a,-1}$ and $U_{a,N+2}$ also need to be determined. Although these variables are less crucial than the boundary variables for the accuracy of the MPD model , the way we compute these can have a significant impact on the solution. A common approach in this regard is to assume that $U_{a,-1} = U_{a,0}$ and $U_{a,N+2} = U_{a,N+1}$. However, one can use more advanced approaches such as the one proposed by [145], which comes at a higher computational expense.*

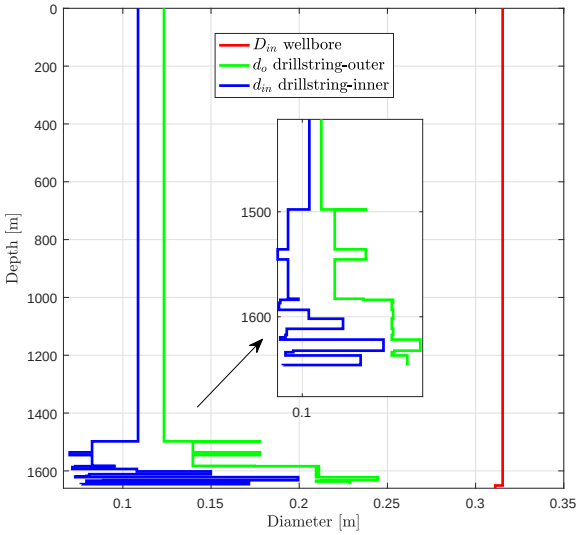

Figure 4.3: The diameter of the annulus and drillstring of the drilling well used in the model validation studies.

4.6 Field data comparisons and simulation studies of industry-relevant scenarios

To evaluate the predictive capabilities of the simulation model and the performance of the proposed numerical implementation of the model, simulations and model validations against experimental data are performed in this section.

4.6.1 Comparisons with field data

We have performed comparisons for single-phase flow scenarios between the hydraulics model presented in this paper and field data obtained during commissioning tests of an MPD operation on a real drilling well. These tests were performed after running casing and before resuming drilling ahead at the length of 1647 m to adjust the MPD control system. The geometries of the drillstring and wellbore are reported in Fig. 4.3. The other parameters used in the model are listed in Table 4.1. The considered measurements correspond to a time period when the drillstring was stationary. In this experiment, the mass flow of the mud pumped into the drillstring varies between low, medium and high values at different rates.

Table 4.1: Parameters used in the hydraulics model for model validation.

Parameter	Symbol	Value	Unit
Length of the well	l	1645	m
Average well inclination	$\bar{\theta}$	1.08	rad
Liquid bulk modulus	β_l	0.94×10^9	Pa
Reference pressure	p_0	10^5	Pa
Liquid density at p_0	ρ_0	1210	kg/m^3
Number of chokes	n_c	2	-
Choke flow factor	k_{c1}	0.0026	m^2
Choke flow factor	k_{c2}	0.0026	m^2
Liquid viscosity	μ_l	0.177	Pa.s
Liquid behaviour index	n_l	0.93	-
Bit nozzles area	A_n	5.69×10^{-4}	m^2
Bit discharge coefficient	c_d	0.8	-

In this paper, instead of identifying $G_i(z_{ci})$ as a function of $z_{c,i}$, we approximate it as a function of time, i.e., we compute the implicit choke characteristic $G(t)$ from the measurements and use it in our simulations. In particular, we use the following relation to approximately compute $G(t)$:

$$G(t) = \sum_{i=1}^{2} G_i(z_{c,i}) \simeq \frac{J_c(t)}{k_{c,1}\sqrt{2\rho_c(t) r (p_c(t) - p_0)}}, \tag{4.44}$$

where $p_c(t) = p_a(t, L)$ and $\rho_c(t) = \rho_a(t, L)$ are the measured pump pressure and flow density upstream the choke, and it is assumed that $k_{c,2} = k_{c,1}$. The choke flow $J_c(t)$ is also a measured variable in this equation. We note that this relation is directly obtained from the choke equation in (4.22). The pump flow rate together with the choke opening signals and implicit choke characteristic $G(t)$ are reported in Fig. 4.4.

In Fig. 4.5, the measured and simulated mass flow rates J_c are plotted in comparison to the measured pump flow rate J_p. We can observe a good match between these two signals. Since the entire length of the wellbore was cased throughout this scenario, we set $J_r^l = 0$. Next, we compare the measured and simulated pressure signals at the choke and pump. In Fig. 4.6, the left side, the modelled and measured choke pressures p_c are compared, where a good match is observed between the measurements and the model. The quality of this match is also an indication of the good accuracy of the implicit choke characteristic $G(t)$. Moreover, by comparing this figure to Fig. 4.5, reporting the flow rates, during periods when the pump flow is steady, we can observe that transients in the choke pressure p_c correspond to transients in the choke flow J_c. This is due the

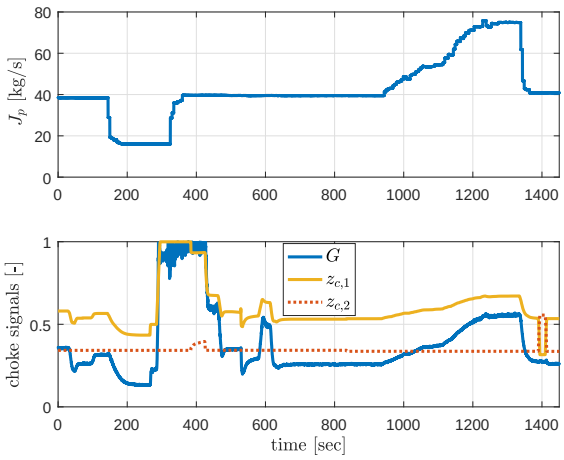

Figure 4.4: Field data: (top) the pump mass flow rate, (bottom) the choke opening signals $z_{c,1}$ and $z_{c,2}$ and the implicit choke characteristic $G(t)$.

compressibility of the mud, which is well captured by the hydraulics model. The model-based and measured pump pressures $p_p = p_a(t, L)$ are plotted in Fig. 4.6, on the right side. We can clearly see a good match between these two signals. However, there are some discrepancies between the two signals as well. These discrepancies are primarily due to imperfections in the Power Law model, used for computing the friction factor. We could expect a higher accuracy by using more advanced frictional models, such as the Herschel-Bulkley model, but at the expense of additional computational complexity and one additional parameter to identify.

A careful observation of the pump and choke pressure measurements reveals that there is a delay of about 4 seconds between transients in the pump pressure and those in the choke pressure, which is exactly due to the fact that pressure waves propagate at the limited speed of sound velocity. These delays are well captured by the model, which is another indication of the high predictive capability of the hydraulics model in terms of capturing fast transients and wave propagation effects. Moreover, in this figure, the high accuracy of the surface pressure control system of MPD can be well observed, when comparing the measured choke pressure to its reference.

4.6.2 Simulation studies

In this part, we present simulation studies. The geometries of the drillstring and the annulus considered in the simulations are reported in Fig. 4.7, and the

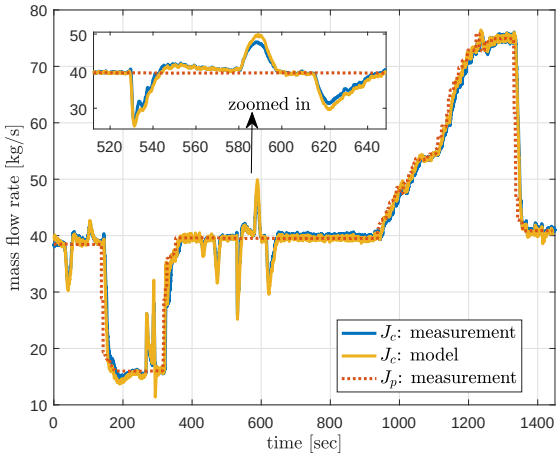

Figure 4.5: A comparison between the measured and modelled choke mass flow rates, together with the measured pump flow.

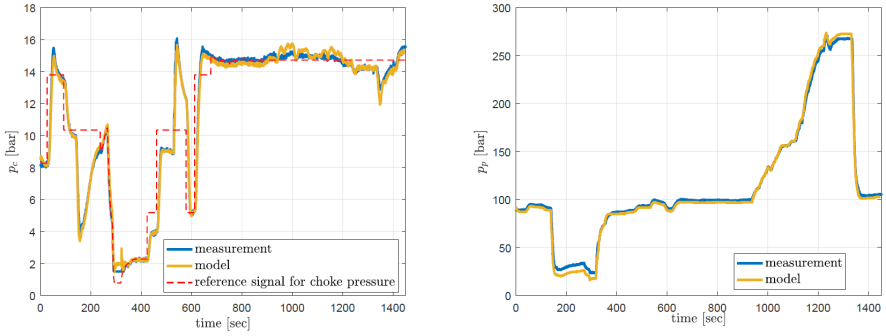

Figure 4.6: A comparison between (left) the measured and modelled choke pressures, together with the reference for the choke pressure as the reference for the MPD pressure control system, and (right) the measured and modelled pump pressures.

parameters are listed in Table 4.2. In this section we try, among other things, to convey the importance of having a simulation model the complexity of which is and should be kept relatively low by including pressure control-relevant aspects of an MPD system. This is achieved by demonstrating through simulations that theoretical analyses based on simple design models and simulation studies are not always sufficient for obtaining a comprehensive and reliable assessment of an

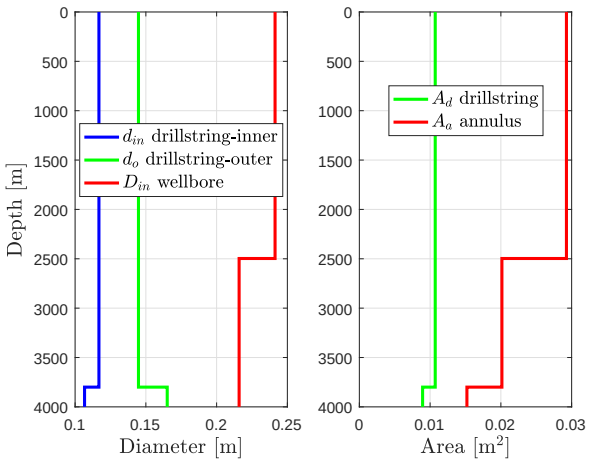

Figure 4.7: The diameter and cross-sectional area of the annulus and of the drillstring considered for the simulation case studies.

MPD pressure control system, and further types of assessments (semi-analytic assessments for instance) based on a simulation model might be needed. This type of assessment may itself become impossible if the simulation model includes irrelevant or less relevant aspects that can cause excessive complexity. However, these type of assessments are beyond the scope of this paper.

To compute the numerical flux functions $F_a(\cdot,\cdot)$ in (4.27) for the DFM of the annulus, a second-order flux-vector-splitting (FVS) scheme is used [69]. For the drillstring, a first-order upwind scheme [113, Chapter 4] is used to compute $F_d(\cdot,\cdot)$ in (4.15). To have a simpler numerical implementation and without losing much accuracy, we linearized the flux function $f_d(\cdot)$ in (4.15). After this linearization, which is obtained by considering $c_l \gg v(x)$, for all $x \in (0, L)$, the flux function in (4.15) reduces to $f_d^T(q_d) = [\rho v, c_l^2 \rho]$, which is now linear in terms of q_d. The maximum value that the time step Δt can take is determined by the Courant-Friedrichs-Lewy (CFL) condition, which is a necessary condition for the convergence of a numerical solution as in (4.27). Here, we use the following relation to compute the time step:

$$\Delta t = CFL\frac{\Delta x}{\bar{\lambda}}, \quad \bar{\lambda} = max\{|\lambda_{a,1}|, |\lambda_{a,2}|, |\lambda_{a,3}|, |\lambda_{d,1}|, |\lambda_{d,2}|\}, \tag{4.45}$$

where CFL is called the CFL number that should be less than one [55]. Note that while one can choose different time steps for the drillstring and the annulus in computing (4.27), here we choose both to be equal and assume that $\bar{\lambda} = c_l$. For the case when there is gas in the annulus, the time step (4.45) can be highly conservative (in the sense that it is chosen much smaller than the maximum value

Table 4.2: The simulation parameter values.

Parameter	Symbol	Value	Unit
Length of the well	l	4000	m
Well inclination	$\theta(x)$	$\pi/2$	rad
Liquid bulk modulus	β_l	1.1×10^{9}	Pa
Sound speed in gas	c_g	316	m/s
Reference pressure	p_0	10^5	Pa
Liquid density at p_0	ρ_0	1500	kg/m^3
Number of chokes	n_c	1	-
Choke flow factor	$k_{c,1}$	0.0025	m^2
Average velocity	V_d	0.5	m/s
Liquid viscosity	μ_l	0.04	Pa.s
Gas viscosity	μ_g	5×10^{-6}	Pa.s
Liquid behaviour index	n_l	0.95	-
Bit nozzles area	A_n	5.77×10^{-4}	m^2
Space discretization step length	Δx	12.5	m
Bit control volume length	Δl	1.5 Δx	m
Gas production index	κ_g	8×10^{-7}	kg/(Pa.s)
Bit discharge coefficient	c_d	0.8	-
Profile parameter	C_0	1.1	-
Number of discretization cells	N	320	-
CFL number	CFL	0.9	-
Discretization parameter	m	20	-

it can take) for computing (4.27) for the annulus, causing a diffusive solution for the annulus. This is because even for small values of α_g in a cell, the sound velocity in that cell can drop substantially.

We perform the simulations for three common and representative drilling scenarios as described below:

1. As the first case study, we run the implemented model for a **choke plugging** scenario, that is, a contingency where the choke effective area drops due to, for example, partial or complete blockage of the orifice by drilling cuttings. Here, we replicate such a scenario by a sudden decrease in the choke flow factor k_{c1} in (4.22) during drilling ahead.

2. Making a **pipe connection** is a common normal drilling operation that takes place around every two to ten hours, depending on the rate of penetration. A pipe connection operation entails halting drilling by slowly ramping down the pump flow to zero and, then, bleeding off the trapped pressure inside the drillstring by opening a bleed-off valve. Afterwards, the

top drive is detached from the drillstring, and a new stand of drillpipe is screwed onto the drillstring. After that, the rig pump flow is ramped up again, resuming the drilling.

3. Gas influx, or gas kick, and subsequent **gas migration** in the annulus is the third scenario that will be studied. A kick usually happens when the reservoir pressure exceeds the well pressure, which can occur for reasons, such as drilling into a high-pressure zone, a pressure drop during a pipe connection or swab and surge effects. We simulate such a scenario by increasing the reservoir pressure to replicate running into an unexpected high-pressure zone.

4.6.2.1 Control system:

To maintain the downhole pressure during these scenarios, a simple pressure control system is used. We design this control system based on a simple lumped-parameters model in [99]. This model consists of three ordinary differential equations and in this model wave propagations effect is compromised in exchange for simplicity. Although this model is derived based on a single-phase flow assumption, it partially accounts for two-phase scenarios through the parameters related to the bulk moduli. A comparison between this design model and the simulation model is provided in Table 4.3.

Table 4.3: Conditions of the simulation model versus the design model.

Model condition	Simulation model	Design model
Complexity	5 PDEs+ 5 closure laws	3 ODEs
Number of dimensions	1D	1D
Number of phases	2	1
liquid-liquid flows	yes	no
gas-liquid flows	yes	no
flow compressibility	yes	no
wave propagation	yes	no
gas migration	yes	no
liquid influx	yes	yes
gas influx	yes	no
flow pattern transitions	yes	no
variation in cross sectional area	yes	no
Isothermal condition assumption	yes	yes
Radially homogeneous flow assumption	yes	yes
Axial flow assumption	yes	yes

Remark 4.9. *The focus of this paper is not on controller design and the used controller does not necessarily provide a satisfactory pressure control performance. The focus is rather on assessing how certain model/system aspects, taken into account in the proposed model, affect closed-loop system performance.*

Following the work by [99], this control system is made up of two parts: 1) a proportional-integral (PI) controller that regulates the surface pressure p_c through the choke opening z_c as the control input and 2) an estimator that generates a reference for the controller from the surface and downhole measurements and also the reference given for the downhole pressure p^*_{dh}. The reference generator consists of a parametrized model, approximating the surface pressure based on a given reference for the downhole pressure, and an estimator that generates an estimate for the parameters of this model. This estimator is designed with a recursive least square (RLS) method with a forgetting factor, see [130]. Assuming laminar flow, the parametrized model is given by

$$p^*_c(t) = p^*_{dh}(t) - (J_c(t) - J_{bpp}(t))(1 - \theta_f(t))F - (1 - \theta_g(t))\, G, \tag{4.46}$$

where $F = \int_0^l \frac{32\mu_l\rho_0}{A_a(x)d_a^2(x)}dx$ and $G = g\int_0^l \rho_0 \sin(\theta_a(x))\, dx$, $\theta_f(t)$ and $\theta_g(t)$ are the to-be-estimated parameters, the estimates of which are indicated by $\hat{\theta}_f(t)$ and $\hat{\theta}_g(t)$, respectively. Here, we assume that the surface measurements are available at a high sampling rate while the downhole measurements are performed at a low rate, which is often the case in realistic drilling scenarios. Here, we take the downhole sampling period during normal operations to be $\Delta t_e = 20$ s. In practice, and especially in the case of long wells, there is also some delay in transmitting the downhole measurements to the surface because of using mud pulse telemetry. However, we here assume that the downhole data are immediately available after measurement. Moreover, we assume that the only choke, described by (4.22), has a linear characteristic in its operating range (i.e., $G_1(z_{c1}) = z_{c1}$ for $0 \le z_{c1} \le 1$, and $G_1(z_{c1}) = 0$ and $G_1(z_{c1}) = 1$ for $z_{c1} < 0$ and $z_{c1} > 1$, respectively). However, one can consider more complex characteristics for the choke through $G(\cdot)$.

4.6.3 Results for a choke plugging scenario.

Here, the results for a choke plugging scenario are shown. In this scenario, the choke flow factor k_c drops by 50% from its nominal value at $t = 400$ s. Because a laminar flow has weaker damping effects on propagating pressure waves and an objective of this section is to illustrate distributed aspects of the model, we consider a laminar flow with $n_l = 1$ along the entire flow path in this scenario. The corresponding results are reported in Fig. 4.8; on the left side of which are located the pressure signals, for both the design model and the simulation model, and on the right side snapshots of the liquid velocity along the flow path are shown. As seen from the left figure, the overall closed-loop responses in

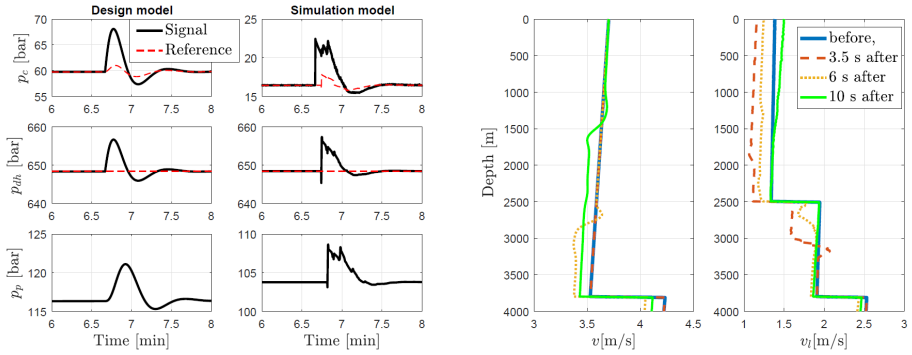

Figure 4.8: Simulation results for a choke plugging event: (left) the choke, downhole and pump pressure signals while comparing the design model to the simulation model, (right) Snapshots of the spatial velocity profile for the simulation model before and after the event.

both models are similar; the design and simulation models show close dynamical behaviours in this scenario. The difference between the two models in terms of the steady-state values of the pressures p_c and p_p is because the compressibility of the liquid is not captured in the design model.

The developed simulation model in this paper captures the wave propagations effects to a large extent and these effects can be clearly observed in the left figure (for the simulation model) in the time-delays that exist between the time when the pressure wave is generated at the surface and the times when it affects the downhole and pump pressures. In addition, the jagged behaviour observed in the pressure signals in this figure is due the wave propagation effects. A more insightful illustration of this effect is given in the right figure by the snapshots of the liquid velocity along the flow path at a variety of time instances. In particular, this figure illustrates the propagation and reflection of pressure waves when striking obstacles (such as the bit) and geometrical changes in the flow path. As can be seen, right before the choke gets plugged, the system is experiencing a (almost) steady-state condition. When the choke is plugged at $t = 400$ s, it causes some fluctuations in the velocity (and also pressure) profiles of the system that keep propagating along the system afterwards, until those are mitigated on a longer time scale, due to 1) frictional effects and control suppression and 2) numerical dissipations.

Summarising, these results show the value of the proposed model in the scope of the performance evaluation of MPD control systems.

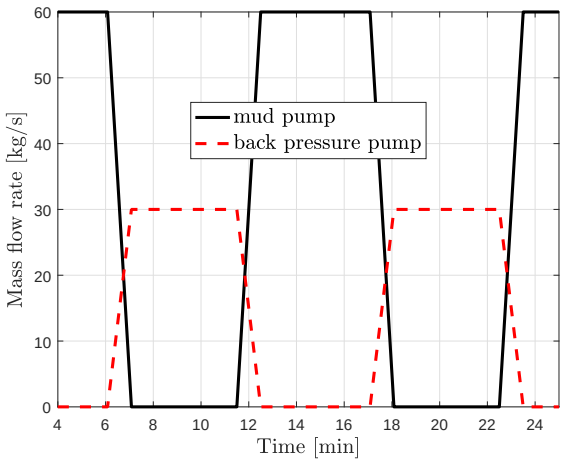

Figure 4.9: Mass flow rates of the mud pump and the back-pressure pump in the connection scenario.

4.6.4 Results for a pipe connection scenario.

Next, we present the simulation results in a pipe connection scenario. The results are presented for two sets of control parameters to further illustrate how neglecting the fast transients (such as wave propagation effects resulting from the distributed nature of the system) in the design model can deteriorate the closed-loop performance.

In this scenario, the mass flow rates of the mud pump and the back-pressure pump change as in Fig. 4.9. When the pump flow rate reaches a level less than half its nominal value, we stop sampling the downhole variables and updating the estimator, to replicate a realistic connection scenario during which the downhole measurements are not available because of the lack of mud circulation.

We first implement this scenario by considering the set of control parameters, referred to as the parameter set 1, which are designed to lead to slow and gentle control signals and a rather slow closed-loop system in terms of recovering from disturbances such as changes in the pump flow. We report the results of these simulations in Figs. 4.10 and 4.11. We apply the controller to both the design model and simulation models. It is observed from Fig. 4.10, that the simulation model exhibits a transient behaviour that is similar to that of the design model. This observation indicates that the simulation model reduces to the design model when the operations are performed slowly, such as in pipe connections. We have also shown the flow and pressure drop of the bit in Fig. 4.11. As expected, when the pressure drop is negative (i.e. $p_d(t,l) - p_a(t,0) < 0$) the flow through the bit becomes zero because of the non-return valve.

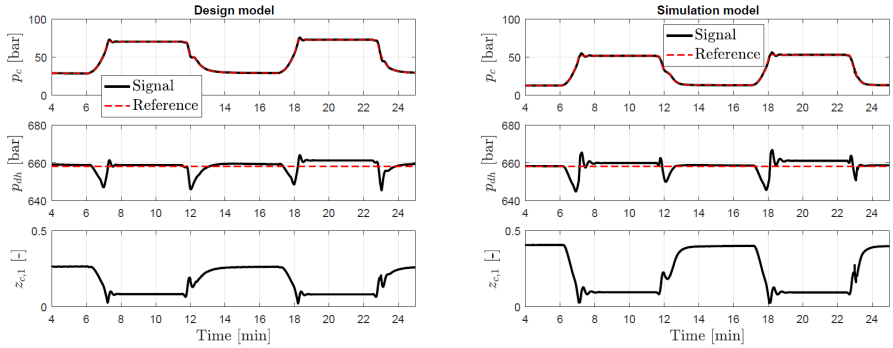

Figure 4.10: Simulation results for a pipe connection scenario for the parameter set 1: the choke, downhole and pump pressure signals from (left) the design model and (right) the simulation model.

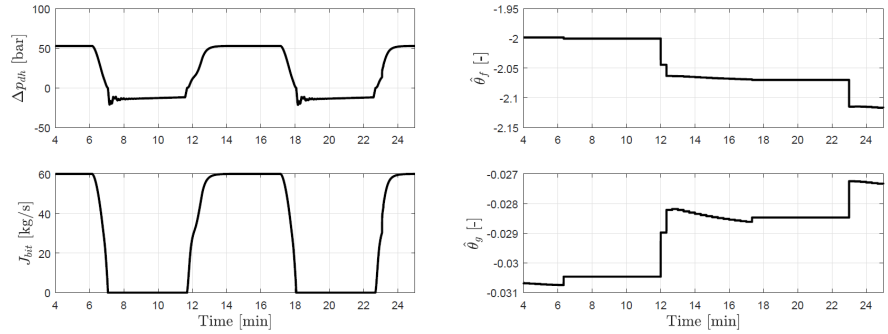

Figure 4.11: Simulation results for a pipe connection scenario for the parameter set 1: (left) the pressure drop over the bit and the flow through it, (right) the adaptive parameters of the estimator.

Results of these simulations can, however, be misleading as it might lead one to conclude that the distributed nature of a drilling system with MPD need not be taken into account while designing a pressure controller. To show that this can be a wrong conclusion and to further highlight the considerable effects of the distributed nature the system on the closed-loop performance, we repeat the same simulation scenario, but with the second set of control parameters, referred to as the parameter set 2, which should lead to faster control signals and better closed-loop performance in terms of recovering from disturbances. The results are plotted in Fig. 4.12. As expected, the closed-loop performance with the design model as the plant has improved. However, the response quality

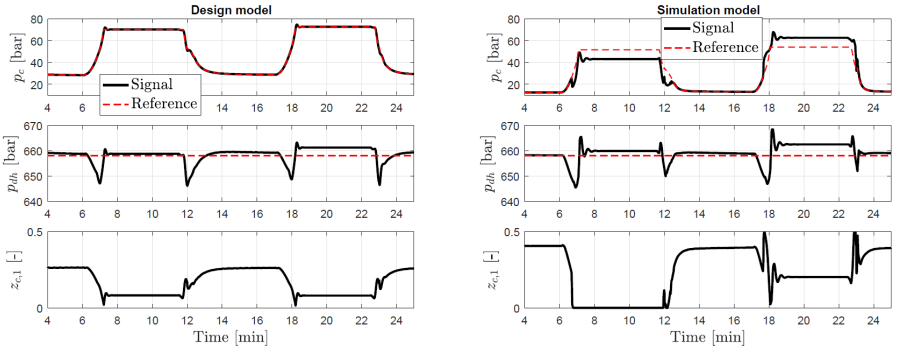

Figure 4.12: Simulation results for a pipe connection scenario for the parameter set 2: the choke, downhole and pump pressure signals from (left) the design model and (right) the simulation model.

when this controller is applied to the simulation model has degraded unexpectedly. This has three important implications: 1) the simulation model is more realistic than the design model and it provides a more accurate prediction of the flow and pressure behaviour in a drilling system. 2) One should not rely *only* on the theoretical results based on the design model and the simulations on a high-fidelity simulation model, as those might show perfect performance in some scenarios and poor performance in yet other scenarios. This can be because this controller with the parameter set 2 results in closed-loop dynamics which are too fast that the fast dynamical aspects of the drilling system (such as the pressure wave propagation effects) are no more negligible. 3) By having the time scale of the fast dynamics of the system, we can already predict intuitively which PI control parameters can result in a poor performance without a need to perform time-consuming simulations on the simulation model. However, doing so might be challenging or even impossible when the control system is more complicated. In such cases, one approach to determine the performance of the closed-loop system can be carrying out semi-analytical system analyses together with simulation studies on the simulation hydraulics model, which includes only the pressure control-aspects of MPD to allow for fast analyses and also simulations.

4.6.5 Results for a gas influx scenario.

This part illustrates the ability of the hydraulics model in capturing gas influx and migration scenarios in the annulus in a closed-loop setting with MPD. Throughout this scenario the pump flow rate is kept constant at $J_p = 60$ kg/s. A rapid 4% increase in the reservoir pressure p_r is applied at $t = 400$ s, resem-

bling the scenario of encountering a high pressure zone while drilling. Before this time, the flow is single-phase all along the flow path. Here, we assume that we can detect the resulting gas kick and also identify the new reservoir pressure some time after it begins. Afterwards, a new reference, larger than the reservoir pressure, is set for the downhole pressure to prevent further gas influxes into the wellbore. To prevent a potential control failure, this operation, changing p^*_{dh}, is performed slowly. We should mention that a common practice during a kick is to control the pump pressure rather than the downhole pressure.

The simulation results are depicted in Figs. 4.13 and 4.14. The choke, downhole and pump pressures are shown on the left side in Figs. 4.13. On the right side of this figure, we illustrate the choke opening together with the parameters of the estimator. We can clearly observe that when the gas reaches the surface, it leads to a rapid change in the choke opening. Moreover, the gas expansion in the annulus causes $\hat{\theta}_g$ to be increasing, as expected, for some time, given the fact that gas expansion phenomenon lowers the hydrostatic pressure in the annulus. Fig. 4.14 gives an illustration of the gas migration and its effects on the flow rate through the choke. We can see from snapshots of the gas void fraction α_g, on the right side of Fig. 4.14, that as the kick moves closer to the surface it expands more, due to a lower pressure. The gas expansion is also illustrated in Fig. 4.15. The gas expansion also increases the mass flow rate of the choke, as can be clearly observed from $J_c(t)$. On the contrary, when the kick reaches the surface and starts leaving the annulus through the choke, the mass flow rate of the choke rapidly drops, starting at around $t = 33.5$ min. The steady-state difference between the accumulative mass of the influx and the accumulative mass of the gas through the choke shows that the used scheme does not exactly preserve the mass flow rate. In order to obtain more accuracy in this respect, one may adapt well-balanced schemes that are more capable in preserving the steady-state response, see, e.g., [45]. In general, this type of scheme is, however, highly expensive computationally. It is observed that the control system successfully, in terms of remaining stable, circulated out the kick in this scenario. However, it is not unreasonable to expect the failure of the designed control system in events such as a choke plugging when there is a two-phase flow in the annulus. The reason for so is large changes in the system behaviour that can occur in such cases. In particular, even for small amounts of gas void fraction α_g, the sound velocity of the flow in the annulus drops drastically, causing the fast dynamics of the system not to be regarded as fast any more. This observation, again, implies the importance of performing a semi-analytical assessment of the performance of a pressure control system based on simulation models, in addition to simulation studies.

4.6.6 Simulations starting from a transient state.

In general, drilling systems with automated MPD are close to some steady-

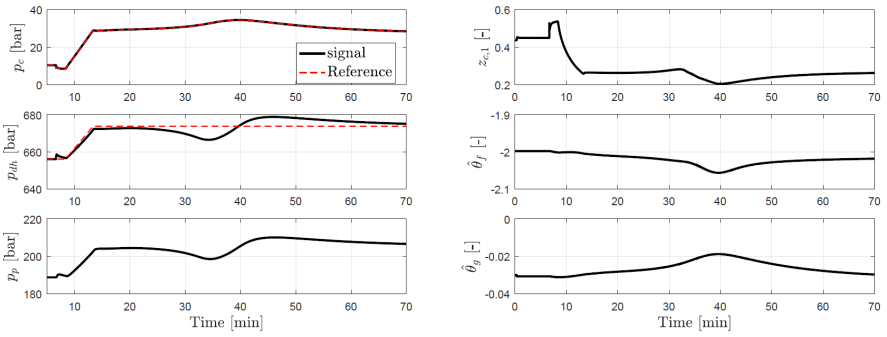

Figure 4.13: Simulation results for the gas influx scenario: (left) the choke, downhole and pump pressures, (right) the choke opening with the adaptive parameters of the estimator.

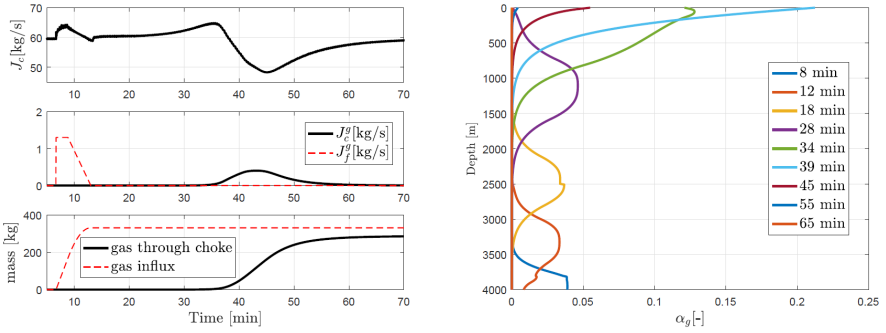

Figure 4.14: Simulation results for the gas influx scenario: (left) mixed mass flow rate downstream the choke J_c, mass flow rate of the gas influx J_f^g and that of the gas through the choke J_c^g, and accumulative mass of the gas influx and that of the gas through the choke, (right) snapshots of the gas void fraction α_g along the annulus.

state condition for most of the drilling time. Therefore, it is reasonable to start the simulation of many drilling scenarios from a steady state of the hydraulics model. Nonetheless, a reliable MPD control system should also show robustness to situations where the system is already in a transient state when the control system takes over the control task. Therefore, in this part, we present simulation results which have been started from a transient initial condition. To design a transient initial condition for these simulations, we consider a steady state of the model without any gas in the wellbore and, then add a pocket of gas to

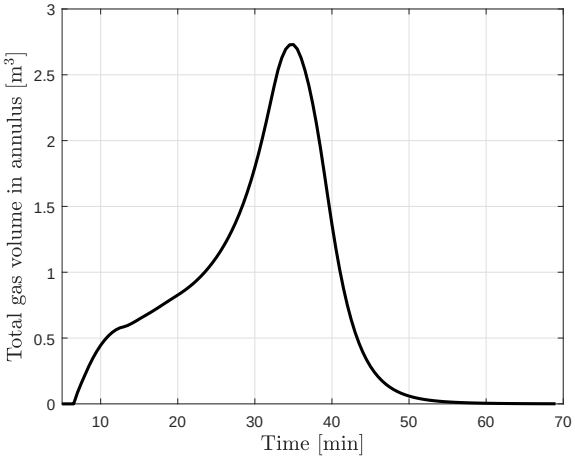

Figure 4.15: Simulation results for the gas influx scenario: the total volume of the gas in the wellbore over time.

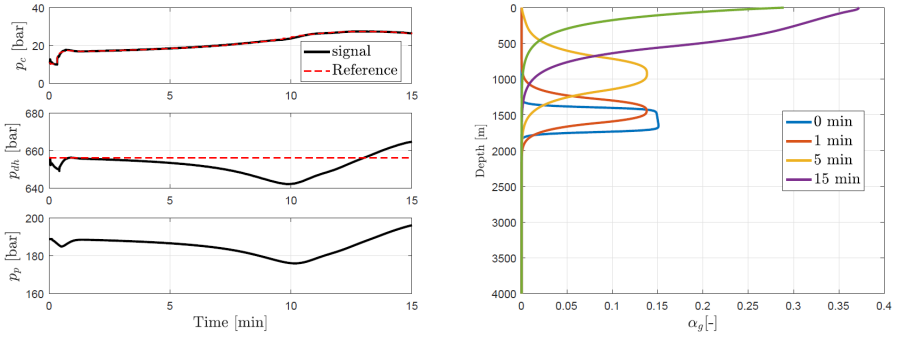

Figure 4.16: Simulation results for a transient initial condition: (left) the pump, downhole and choke pressure signals, (right) snapshots of the gas void fraction α_g along the annulus.

perturb it. The gas migration corresponding to this scenario is reported on the right side of Fig. 4.16. In this figure, the snapshot caught at the zero time shows the considered initial condition for α_g. The pump, downhole and choke pressure signals are reported on the left side of Fig. 4.16. These simulations show that this hydraulics model and the numerical tool can also be used to simulate drilling scenarios starting from a transient state.

4.7 Conclusions

A two-phase hydraulics model in the form of a two coupled systems of partial-differential equations has been derived for managed pressure drilling (MPD). The model complexity is limited by incorporating only the mass transport, pressure dynamics and other aspects of an MPD system that can impact the performance of a pressure control system in real world drilling scenarios. Therefore, it provides a basis for evaluating the performance of pressure control systems in virtual test scenarios. Moreover, an approach has been presented for numerical implementation of the model. Variations in the cross-sectional area of the flow path as well as the nonlinear boundary conditions are often not considered in the control design but do exist in reality and can significantly jeopardize the performance and stability of a pressure control system. These aspects have been captured in the model and are accounted for during numerical implementations. The predictive capability of the model and the performance of the numerical implementations have been demonstrated through illustrative case studies representing a choke plugging, connection and gas influx scenario. Through these studies, we have also demonstrated the importance of keeping the complexity of an MPD simulation low. However, the developed model is not suitable for handling scenarios related to vertical motions of the drillstring, such as washing-stand and tripping scenarios. Adding this aspect to the model is an ongoing research. Moreover, we have illustrated the high accuracy of the model by comparing it with field data from a real-life drilling well for single-phase scenarios. A direction of our current work is the further validation of the developed MPD model in this paper by comparing simulation results and drilling field data for two-phase flow drilling scenarios.

Chapter 5

Model Validation for Multi-phase Flow

The objective of this chapter is to experimentally validate the MPD model developed in Chapter 4 in case of liquid and gas influx scenarios. The MPD model is coupled with a reservoir model and a gas pump to, respectively, simulate a liquid and a gas influx. For the case of a liquid influx, to obtain a realistic reservoir model, a new model based on the conservation of mass and momentum is proposed, which outperforms the existing reservoir models in the literature in predicting the mass flow rate produced by the reservoir. The proposed model and the existing models are introduced in this chapter. Some uncertain constants of these models are identified through an optimization process according to the field data. For the case of a gas influx, a gas pump with known mass flow rate over time is (virtually) installed at the bottom of the well to simulate the gas injection from the reservoir into the annulus (no dynamics is present in this case). The comparison of the simulation results and the field data shows a close match and evidences the good predictive capabilities of the model-based simulation platform developed in Chapter 4.

5.1 Contingencies in MPD automation

Control of contingencies, i.e., unwanted or unforeseen situations during drilling, is a crucial step in MPD automation. In this chapter, we focus on the validation of the MPD model developed in Chapter 4 in case of a special contingency, when the fluid from the formation flows into the annulus, i.e., an influx phenomenon.

The International Association of Drilling Contractors (IADC) defines influx as the flow of fluids from the formation into the wellbore. An influx can occur when the operating Bottom-Hole Pressure (BHP) falls below the pore pressure of the formation. The BHP is maintained intentionally below the pore pressure

in case of Under Balanced Drilling (UBD) operations. An unintentional or an unwanted influx during MPD operations is called a kick [173]. The operating region where an influx or a kick can occur is shown in Figure 4.1 near the bit. In this region, the mass flow rate from the formation (a reservoir or a fracture) is denoted by $J_r^{g,l}$, where the subscript r denotes reservoir and superscripts g, l refer to gas and liquid, respectively. The approximation of the mass flow rate from the formation into the well-bore or vice versa is challenging due to the lack of measurements at the downhole of the wellbore and the lack of a representative dynamical model.

The influx, migrating up through the annulus, can be divided into two categories: standard and complex influx. When the influx is purely liquid, it is categorized as a standard one. When the flow from the formation contains gas, the gas expands as it is circulated out through the annulus. Moreover, the gas might dissolve in the liquid and get separated at the top. These features render this kind of influx complex. In both types of influx, a velocity difference between the two phases (two different liquids or mixture of gas and liquid) exists, which should be handled by a slip law between the two phases. This kind of flow is usually modeled by the DFM [128]. In the remainder of this chapter, we first analyze the case of liquid influx together with the validation of the model for such a scenario. Then, similarly, we deal with the case of a gas influx.

5.2 Validation in case of a liquid influx

During a standard drilling scenario, if the volumetric flow rate of the pump is not changed, the volumetric flow rate through the choke should also remain constant. If the flow rate through the choke changes significantly while the flow rate of the pump is constant, this can be related to an influx or loss. If the choke and pump pressure behave qualitatively similar to each other, this can also be attributed to a liquid influx. This feature is handy to detect a field data corresponding to this scenario. In this section, we first provide different techniques for approximating the influx flow rate exchanged between the wellbore and the formation. Then, we use an optimization procedure to identify the unknown parameters of the MPD model and the formation model to render the simulation data and the field data as close as possible to each other.

5.2.1 Field data for liquid influx

We consider a liquid influx occurring in a well with a short length ($600 - 800$ m) with one discontinuity in the diameter of the drillstring and one discontinuity in the diameter of the wellbore. We successively study two field data sets from this well, corresponding to single-phase flow and liquid-liquid two-phase flow. These field data are gathered from sensors installed at the pump and the location of the choke manifolds, reading pressure at the pump p_p, volumetric flow rate through

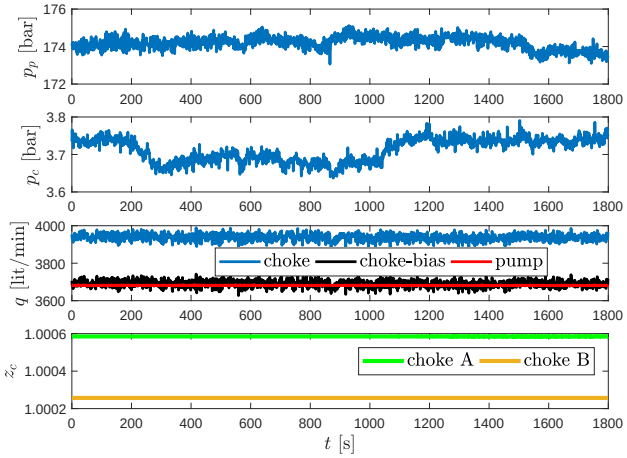

Figure 5.1: Field data for single phase flow (for "choke-bias", refer to Remark 5.1).

the pump q_p, (common) pressure before the choke manifolds p_c, the summation of volumetric flow rate through the choke manifolds q_c and also choke opening of the choke manifolds A and B installed at the wellhead, z_{cA} and z_{cB} (these two variables are dimensionless values representing the ratio of the open area for passing the flow through the valve and the full area).

For the field data of the single-phase scenario, we refer to Figure 5.1. This represents the case where the influx has not started yet. With this set of data, we can identify the unknown parameters in the MPD model related to the drilling equipment such as the bit and also the drilling fluid parameters.

Remark 5.1. *In the steady condition of single-phase flow, the mass flow rate of the choke and the mass flow rate of the pump should be equal. As the mud is almost incompressible, this equality should also hold for the volumetric flow rate. However, as the choke pressure is less than the pump pressure, the volumetric flow rate at the choke should be slightly higher than the volumetric flow rate at the pump. We use this fact to correct the choke flow rate readings from the sensors as the flow sensors are not calibrated well at the choke. Results shown for the liquid influx scenario based on this corrected version of the choke flow measurements are reported in Figure 5.1 under the legend "choke-bias".*

For the liquid-liquid flow scenario, we refer to Figure 5.2 which represents the field data after the influx occurrence from a fracture. At $t = 600$ s, the choke flow increases compared to its nominal value without changing the drilling condition. This shows that an influx from the fracture (any separation in the formation)

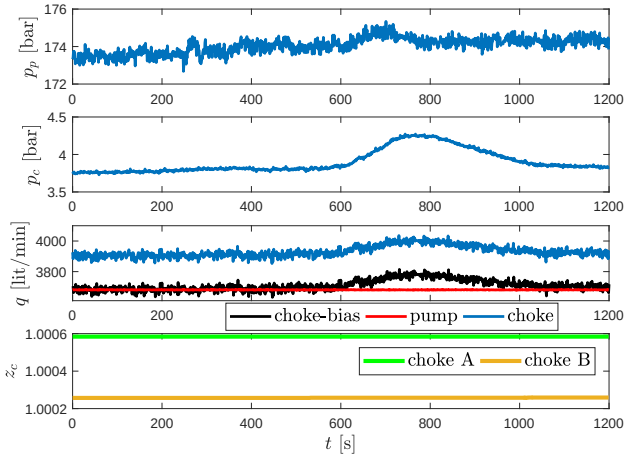

Figure 5.2: Field data for liquid-liquid flow.

has entered the wellbore. Due to the almost simultaneous increase of the pump pressure, choke pressure and the choke flow, this should be a liquid influx, which in this case is water.

5.2.2 Approximation of the influx flow rate

In this section, we introduce different techniques to approximate the mass or volumetric flow rate of the liquid influx: a data-based approach and three model-based approaches.

The data-based technique is based on the field data gathered from different sensors installed at the wellhead. Therefore, we do not introduce any dynamical model for the reservoir. This approach is then used as a testbed to qualify the accuracy of three different dynamical reservoir models, that are used to approximate the influx flow rate. These models, by replacing the simple static equation (4.21), interact with the hydraulics in the wellbore.

5.2.2.1 Data-based approach

To approximate the influx/loss flow rate, we consider

$$q_{\text{con}} = q_c - q_{c_{\text{nom}}}, \tag{5.1}$$

where $q_{c_{\text{nom}}}$ is the nominal value for the choke flow q_c, before the occurrence of the influx/loss and q_{con} represents the flow rate of the contingency (either influx or loss). If $q_{\text{con}} > 0$, an influx has occurred, otherwise, if $q_{\text{con}} < 0$, the drilling

mud has penetrated into the formation and a loss scenario is encountered. The integration of q_{con} over time represents the pit gain (i.e., an increase in level of mud tanks due to an increase in return mud volume). Then, to approximate the influx mass flow rate J_r^l, we have

$$J_r^l = \rho_{0_{\text{con}}} q_{\text{con}}, \tag{5.2}$$

where $\rho_{0_{\text{con}}}$ is the nominal density of the drilling mud in case of loss or the nominal density of the fluid in the formation in case of an influx.

5.2.2.2 Reservoir/fracture modeling

In contrast to the data-based approach, by assigning a dynamical model to the formation, the interaction between the wellbore and the formation determines the influx rate. Here, we assume that the influx comes from a fracture and we present three simple lumped fracture models to be validated against the field data. If the field data is obtained in a scenario with an influx coming from a porous formation, we use the model involving a distributed porous formation discussed in [174].

First-order fracture model

We denote with p_{fr} the pressure of the fracture far away from the wellbore, with m_{fr} the mass of the fluid in the fracture, with C the capacitance of the fracture, with R the resistance of fluid flow from the fracture to the wellbore, and with p_{dh} the bottom-hole pressure. The interaction between the wellbore and the fracture is schematically depicted in Figure 5.3. We introduce the following equations related to the capacitance and the resistance:

$$C = \frac{\mathrm{d}m_{\text{fr}}}{\mathrm{d}p_{\text{fr}}}, \tag{5.3}$$

$$R = \frac{p_{\text{fr}} - p_{\text{dh}}}{J_r^l}, \tag{5.4}$$

where "d" denotes the differential operator. Using the mass balance and (5.3), the rate of mass change inside the fracture can be written as follows:

$$C\mathrm{d}p_{\text{fr}} = -J_r^l \mathrm{d}t \rightarrow C\dot{p}_{\text{fr}} = -\frac{p_{\text{fr}} - p_{\text{dh}}}{R}.$$

Finally, by using the above mass balance and the resistance definition (5.4), the first-order dynamics of the fracture is defined as

$$\begin{cases} RC\dot{p}_{\text{fr}} = -p_{\text{fr}} + p_{\text{dh}}, \\ J_r^l = \dfrac{p_{\text{fr}} - p_{\text{dh}}}{R}. \end{cases} \tag{5.5}$$

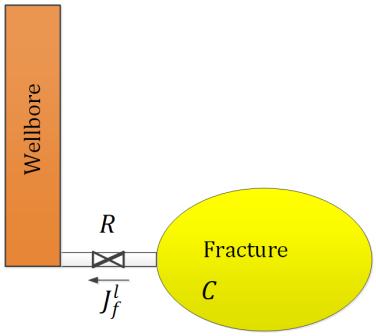

Figure 5.3: Interaction between the wellbore and the fracture, first-order fracture model.

This fracture model interacts with the MPD model via J_r^l. In this first-order fracture model, R, C, the initial value of p_{fr} and the time instant when the fracture model should be coupled with the wellbore dynamics (i.e., when the influx actually occurs) are unknown. These are later considered as tuning parameters of the model in Section 5.2.3.

Remark 5.2. *The mass flow rate equation in* (5.5) *resembles the Productivity Index (PI) relation [112] in reservoir engineering, with the PI equal to* $\frac{1}{R\rho_{fr}}$ *and* ρ_{fr} *being the density of the liquid in the formation.*

Constant terminal rate solution

The Constant Terminal Rate Solution (CTRS) is similar to the PI relation with a constant fracture pressure but with a time-varying PI [174]. The CTRS relation that approximates the volumetric flow rate of the influx is

$$q_{\text{influx}} = \frac{c_1}{c_2 + \ln(c_3 t)} \left(p_{\text{fr}} - p_{\text{dh}}\right), \tag{5.6}$$

where $q_{\text{influx}} = \frac{J_r^l}{\rho_{\text{fr}}}$ is the volumetric flow rate of the influx, c_1, c_2, c_3 are some unknown constants. In this relation, the pressure dynamics in the fracture are approximated by the time-dependent PI equal to $\frac{c_1}{c_2+\ln(c_3 t)}$. Similar to the first-order model, the unknown constants, p_{fr} and the time instant t at which the MPD dynamics and the CTRS connect to each other are free and will be tuned in Section 5.2.3.

Second-order fracture model

This model is based on mass and momentum conservation of laws and, similar to the first-order model, makes use of the capacitance relation (5.3).

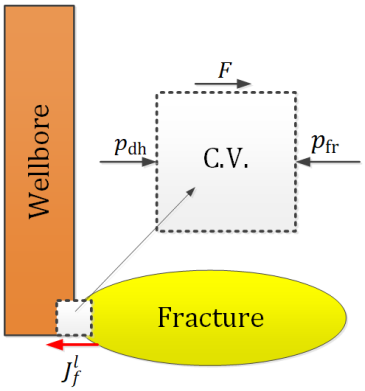

Figure 5.4: Interaction between the wellbore and the fracture, second-order model.

Similar to the first-order model, the mass conservation can be written as

$$J_r^l = -C\dot{p}_{\mathrm{fr}}. \tag{5.7}$$

To enforce the momentum conservation at the interaction point of the fracture and the wellbore, we consider a control-volume with a unit cross sectional area and apply

$$I\frac{\mathrm{d}}{\mathrm{d}t}\left(J_r^l\right) = (p_{\mathrm{fr}} - p_{\mathrm{dh}}) - F(J_r^l), \tag{5.8}$$

where I represents the inertia of the fluid and $F(\cdot)$ describes the friction opposing the influx flow, which can be written as $F(J_r^l) = \zeta J_r^l$ with a constant ζ. Combining (5.7) and (5.8) yields

$$\begin{cases} J_r^l = -C\dot{p}_{\mathrm{fr}}, \\ \ddot{p}_{\mathrm{fr}} = -\dfrac{p_{\mathrm{fr}} - p_{\mathrm{dh}}}{IC} - \dfrac{\zeta}{I}\dot{p}_{\mathrm{fr}}. \end{cases} \tag{5.9}$$

The free parameters in this model, C, I, ζ, initial value of p_{fr} and the coupling time, are later tuned to reproduce a result similar to the data-based approach. The interaction between the wellbore, the fracture and the Control-Volume (C.V.) is depicted in Figure 5.4.

Remark 5.3. *The second-order model shares similarities with the Kaasa model [99], which is usually used for simulating hydraulics inside a well by a lumped model.*

We now present an optimization method to calibrate the parameters of the MPD and fracture models to replicate the two field data sets shown in Figures 5.1 and 5.2.

5.2.3 Parameter identification for liquid influx

The unknown parameters of the fracture models and the MPD model to be identified can be divided into two categories: 1) parameters that can be identified from the steady field data shown in Figure 5.1 (speed of sound c_l, mud viscosity μ_l, nominal density of mud ρ_{0l}, nozzle area at the bit A_N and choke factor at different opening $K_c(z_c)$), and 2) parameters that are related to the liquid influx and can be identified from data represented in Figure 5.2. In order to obtain physical values for the parameters, the range of each parameter is restricted based on the prior knowledge from the drilling system. These free parameters are chosen through an optimization procedure such that the simulation results and the field data match as closely as possible.

5.2.3.1 Field data for the steady case

The following free parameters can be tuned based on the steady-state field data set. It should be noted that since we have one set of data and many tunable parameters, the identified parameters are not unique. We try to identify these values in a physically meaningful range.

Speed of sound c_l

Given the length of the well, the wave propagation from the choke to the pump takes around $2-3$ seconds. This is hard to be diagnosed from the data of this well as the resolution of the field data is 1 s which is also polluted with noise. So, it is hard to attribute the change in the pressure signals to the noise or to the wave reflection inside the wellbore. Hence, c_l is optimized in the range $[1000, 1200]$ m/s, which is the typical value for the speed of sound in the drilling mud.

Mud viscosity μ_l

Except at some periods at the beginning of the drilling, the pump flow is around 3700 lit/min. The total pressure drop of the friction and the drill bit can be identified from this data set. At this flow, both the frictional pressure drop and the drill bit pressure drop are related to the flow rate squared. Data sets with the same flow rate can not be used to identify the individual contribution of the friction and the drill bit to the pressure drop. This renders the differentiation between the effect of viscosity and nozzle area at the bit difficult. We use the nominal value of 40 cP for μ_l (which is a typical value of viscosity for drilling mud). For this case, as the flow in the drillstring is always in the turbulent region, and pipe roughness plays a crucial role in the turbulent friction model, which is unknown. In order to account for this important factor, the frictional

pressure drop $\Delta p_{\text{Friction}}$ is multiplied by a coefficient K_f, i.e.,

$$\Delta p_{\text{Friction}} = K_f g(\mu_l, v, d), \tag{5.10}$$

where $g(\cdot,\cdot,\cdot)$ gives the frictional pressure drop for a fixed pipe roughness, which depends on the viscosity of the liquid μ_l, the velocity of the liquid v and the hydraulic diameter of the pipe d. The coefficient K_f is considered as a free parameter to be tuned during the optimization procedure and viscosity μ_l is kept constant.

Mud density ρ_0

The nominal density of the drilling mud used in this well is around 1450 kg/m^3. As the density of the cuttings is around 2200 kg/m^3, ρ_0 can also be a bit higher than its nominal value. So, we optimize the mud density in the range [1400 − 1600] kg/m^3.

Nozzle area A_N

Since the pump flow rate is high, the flow in the drillstring is always in the turbulent region. By varying $\mu_l \in [10, 50]$ cP, the total friction loss changes in the range of $\Delta p_{\text{Friction}} \in [167, 175]$ bar. Based on the total gravitation gain $\rho g h \approx 110$ bar, the measurements of the downhole pressure (≈ 110 bar) and pump pressure ($p_p \approx 174$ bar), we have

$$p_p \approx p_{dh} - \rho g h + \Delta p_{\text{Friction}} + \Delta p_{\text{bit}}. \tag{5.11}$$

Substituting the above values in this relation shows $\Delta p_{\text{bit}} \approx 0$ and the nozzle area should be very close to the drillstring area at the bit location, while in practice, the nozzle area should be much lower than the drillstring area. To account for that, K_f introduced before is used to reduce the effect of friction ($K_f < 1$) and allocate more friction drop to the bit. Then, A_N is tuned manually to set a pressure drop in the range of $10 - 20$ bar over the bit, which is a typical value of pressure drop over the bit in such well configurations.

Choke constant $K_c(z_c)$

Pressure and flow rate sensors are installed both upstream (before) and downstream (after) the choke. Therefore, adequate data is available to train a nonlinear model (real choke manifolds represent nonlinear behavior) for the choke installed at the wellhead. To do so, a widely used choke model is assigned to the choke and its free parameter, $K_c(z_c)$, is trained over the available data. This nonlinear model is given by [128]:

$$q_c = K_c(z_c)\sqrt{\frac{2}{\rho_c}(p_c - p_0)}, \tag{5.12}$$

Table 5.1: Choke constant as a function of choke opening.

$z_{cA} + z_{cB}$	0	0.6	0.8	1.1	2
K_c	0	0	0.00052	0.0014	0.0036

where ρ_c, p_c, q_c are, respectively, the density of the mud at the choke, the pressure at the choke and the volumetric flow rate through the choke. Having access to data (q_c, p_c) with a different choke opening z_c ($z_c = z_{cA} + z_{cB}$ being the total opening of the choke A and B installed at the top of the well), $K_c(z_c)$ is obtained and presented in Table 5.1.

To summarize, values for c_l, K_f, ρ_0 are determined during an identification process, A_N is set manually to ensure a realistic pressure drop over the bit, μ_l is fixed on 40 cP and $K_c(z_c)$ is specified over the wide range of available data with different choke openings. Now, the procedure to obtain the influx-related parameters is explained.

5.2.3.2 Liquid influx-related parameters

In this section, based on the field data presented in Figure 5.2, the unknown parameters of the fracture models and the MPD model are identified.

Remark 5.4. *Since we study the liquid-liquid flow, the equation of state* (4.7b) *is replaced by a similar equation of state as in* (4.7a) *and therefore* (4.11) *changes accordingly. For the sake of brevity, we did not include these equations in this chapter.*

Speed of sound $c_{\mathbf{inf}}$

As the influx is water, the speed of sound in the influx medium is optimized in the range $[1000, 1200]$ m/s.

Influx viscosity $\mu_{\mathbf{inf}}$

This parameter is optimized between $[0.5, 1.5]$ cP as the viscosity of water is around 1 cP.

Influx nominal density $\rho_{0_{\mathbf{inf}}}$

This parameter varies between $[800, 1200]$ kg/m^3 due to the nominal density of water (1000 kg/m^3).

Influx rate approximation

The influx rate approximated by the data-based approach obtained by (5.1) and (5.2) is shown in Figure 5.5. For the model-based approaches, the interaction

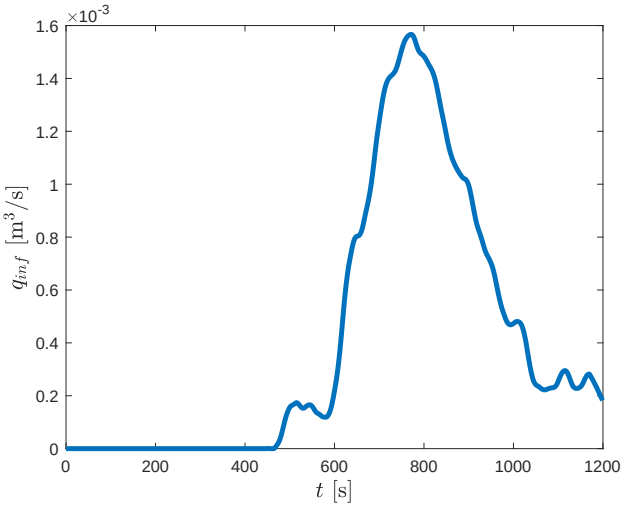

Figure 5.5: Influx rate approximated in the simulations by the data-based approach.

between the fracture model and the wellbore model by replacing equation (4.21) determines the influx rate.

Slip relation

In this work, we use the simple slip relation of Zuber-Findlay [69], $v_{\text{inf}} = Kv_{\text{mix}} + S$, where v_{inf} is the velocity of the reservoir liquid and v_{mix} is the mixture velocity of the mud and the reservoir liquid. Then the value for K and S are selected during an optimization procedure.

To summarize this section, the values of $c_{\text{inf}}, \mu_{\text{inf}}, \rho_{0\text{inf}}, K, S$ are determined during an optimization procedure. If model-based approaches are used to approximate the influx rate, the free parameters corresponding to each model are also tuned during the optimization procedure. The optimization problem is illustrated below.

5.2.3.3 Optimization procedure

For the optimization problem underlying the parameter identification, the following objective function is defined:

$$f(p_{p\text{Sim}}, p_{c\text{Sim}}) = \frac{\left\|p_{p\text{Field}} - p_{p\text{Sim}}\right\|_2}{\left\|p_{p\text{Field}}\right\|_2} + \frac{\|p_{c\text{Field}} - p_{c\text{Sim}}\|_2}{\|p_{c\text{Field}}\|_2}, \tag{5.13}$$

Table 5.2: Identification result for steady-state flow.

parameter	c_l [m/s]	K_f	A_N [m^2]	ρ_0 [kg/m^3]
nominal	1000	1	-	1450
optimized	1190.2	0.86	0.0014	1466

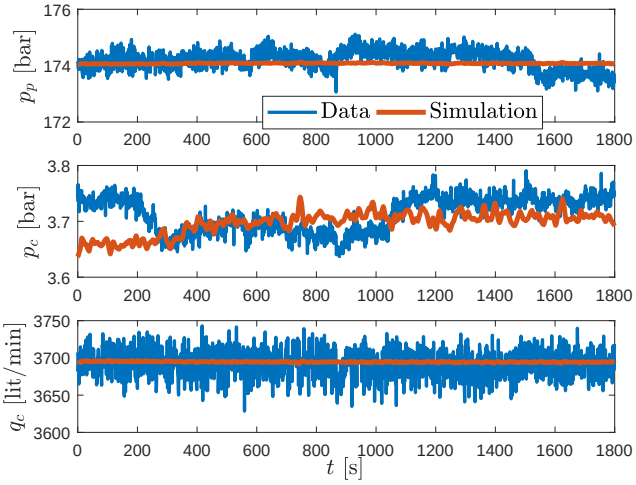

Figure 5.6: Comparison of field data and simulation model for single-phase flow.

where subscripts "Field" and "Sim" denote values obtained from the field data and the MPD model in Chapter 4, respectively, and $\|\cdot\|_2$ is the Euclidean norm of the signals discretized over time based on the sensor reading frequency. This objective function is aimed to be minimized by tuning the free parameters of the MPD model and the fracture models.

5.2.4 Validation results

For the purpose of model validation in steady-state conditions, we obtain the parameter values as reported in Table 5.2. The associated simulation results can be seen in Figure 5.6 in comparison to the field data. In this figure, pump pressure p_p, choke pressure p_c and volumetric flow rate through the choke q_c are depicted. All these values are approximated well with the MPD model using physical values for the optimized variables.

For the case where the influx occurs, we perform the identification for the two techniques introduced in Sections 5.2.2.1 and 5.2.2.2, the data-based approach and the model-based approach.

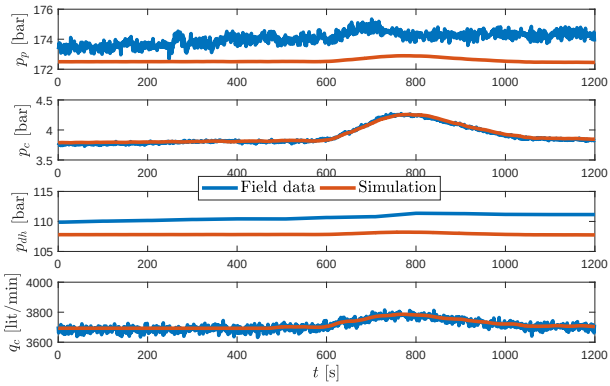

Figure 5.7: Comparison of field data and simulation for liquid-liquid flow.

5.2.4.1 Data-based approach in approximating the influx rate

For the purpose of validation in the liquid-liquid flow scenario by using the data-based approach discussed in Section 5.2.2.1, we obtain the parameter values as reported in Table 5.3. The associated simulation results can be seen in Figure 5.7.

The results associated to data-based approximation of the influx are well in agreement with the measured field data and this shows that influx rate approximated by this approach is close to the actual influx rate. This feature is used in the next section to construct the right model for the fracture.

5.2.4.2 Fracture models

According to the data presented in the previous section, it is confirmed that the influx profile in Figure 5.5 is close to the real influx from the reservoir. Now, among the proposed fracture models introduced in Section 5.2.2.2 and based on the measurement of the downhole pressure, we tune the free parameters of the fracture models to reproduce a similar influx profile. Then, the most reliable fracture model is interconnected with the wellbore dynamics by replacing (4.21).

Table 5.3: Parametric settings for liquid-liquid flow obtained with the data-based approach.

parameter	c_{inf} [m/s]	μ_{inf} [Pa.s]	$\rho_{0_{\text{inf}}}$ [kg/m^3]	K	S
nominal	1100	0.001	1000	-	-
optimized	1021.8	0.0011	1200	0.9745	0.4659

Remark 5.5. *In this well, "measurements while drilling" equipment were installed at the bit, and enabled the measurement of the downhole pressure. When the signal for p_{dh} is available, models* (5.5)*,* (5.6) *and* (5.9) *can be simulated without coupling the models to the wellbore dynamics via* (4.21)*.*

First-order dynamics

Free parameters in the model (5.5) are R, C, p_{fr} and the time instant at which the fracture dynamics is coupled with the wellbore dynamics. The best result achieved by tuning these parameters to fit the influx rate is shown in Figure 5.8 ($R = 5.7\times10^{6}$ 1/(m·s), $C = 5.3\times10^{-5}$ m·s^2, $p_{\text{fr}} = 185.6$ bar and $t_{couple} = 624$ s). Apparent from the result, the data coming from the dynamics occurring in the reservoir is quite different from the data of the data-based approach. Therefore, this fracture model is not selected to be coupled with the wellbore dynamics.

CTRS

For CTRS relation (5.6), in addition to p_{fr} and the time instant of dynamic coupling, the constants c_1, c_2, c_3 are also required to be tuned. The best fitting of the influx volumetric flow rate by this method is shown in Figure 5.8 ($c_1 = 1\times10^{-10}, c_2 = 0, c_3 = 100$, $p_{\text{fr}} = 197.8$ bar, $t_{couple} = 618$). Clearly, this method can not reliably approximate the influx well either. This result is very similar to the result of first-order model, except that the fracture pressure dynamics is replaced with the time-varying term $\dfrac{1}{c_2 + \ln(c_3(t - t_{couple}))}$.

Remark 5.6. *In both the first-order model and the CTRS, at the moment of coupling, the influx volumetric flow rate changes instantly, i.e., the acceleration of the control volume at the interaction of the fracture and the wellbore should be infinite despite the finite force acting on the fluid flow. Therefore, these models can not represent the real physics correctly.*

Second-order model

Free parameters in the second-order model (5.9) are $I, C, \zeta, p_{\text{fr}}$ and the coupling time instant. The best result achieved by tuning these parameters to fit the influx rate is shown in Figure 5.8 ($I = 7.8\times10^{6}$ m, $C = 0.0025$ m·s^2, $\zeta = 0.004$ m/s, $p_{\text{fr}} = 111.7$ bar and $t_{couple} = 582.6$ s). Apparent from the results, this model can achieve an influx profile similar to the data-based approach with a high accuracy. Therefore, this model is used to validate the field data.

Remark 5.7. *The main issue with these fracture models is the sensitivity of the results to the fracture and downhole pressure. A small change in the downhole pressure significantly affects the influx rate. Therefore, the parameters of the fracture model should be identified together with the wellbore dynamics. In the*

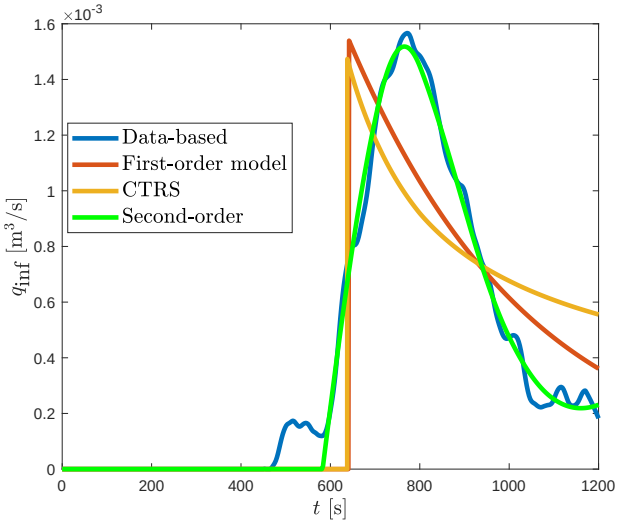

Figure 5.8: Comparison of data-based influx and approximation of influx via the first-order fracture model, the CTRS and the second-order fracture model.

following section, the results of the coupling of the second-order fracture model and the wellbore model are presented.

5.2.4.3 Validation of the interconnected wellbore-fracture dynamics by the second-order fracture model

In this section, the second-order fracture model (5.9) is coupled with the wellbore by replacing (4.21) to reproduce the field data in Figure 5.2. The optimized variables are reported in Table 5.4 and the comparison of the simulation with the field data is shown in Figure 5.9.

Table 5.4: Parametric settings for liquid-liquid flow obtained by the second-order fracture model.

parameter	c_{inf} [m/s]	μ_{inf} [Pa.s]	$\rho_{0\text{inf}}$ [kg/m^3]	K	S
nominal	1100	0.001	1000	-	-
optimized	1019.4	0.0006	1000	0.9873	0.18
parameter	p_{fr} [bar]	I [m]	C m·s^2	ζ [m/s]	t_{couple} [s]
nominal	111.7	-	-	-	-
optimized	109.3	8.7×10^6	0.0013	0.0028	600

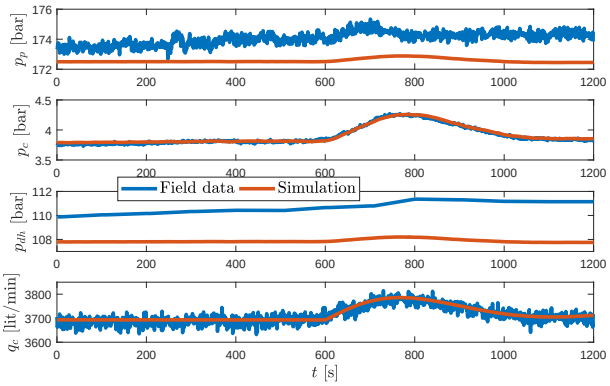

Figure 5.9: Comparison of field data and simulation for liquid-liquid flow with the second-order fracture model and with the parameter setting of Table (5.4).

Apparent from Figure 5.9, all components of the field data are approximated well with the parameter setting in Table 5.4. Better results can also be obtained by a better tuning of parameters on a richer field dataset if available. This comparison shows the predictability of the MPD model developed in Chapter 4 for standard influx scenarios. The capability of the MPD model is further investigated in the next section for the case of complex influx scenario.

5.3 Validation in case of a gas influx

In this section, all the steps required to validate the MPD model developed in Chapter 4, without any adaptation, in case of multi-phase flow scenarios are explained.

5.3.1 Field data

Finding suitable field data for a gas influx case is challenging. Measurements can be noisy and unreliable in case of gas and liquid mixture. Almost all measurement devices are designed for either gas or liquid, not for a mixture of those. Moreover, the gas influx in practice is usually accompanied with liquid influx and might also be dissolved in the drilling mud during the migration through the annulus. The coupling between the reservoir model and the wellbore model can be seen as another challenge for such complex influx scenario. These difficulties render finding a field data suitable for the purpose of accurate model validation virtually impossible. However, experimental data from test wells, which are designed to be useful for model validation, can be used. In this section, the data

presented in [5], [12] is used. The validation is performed in a different manner than in [5], which is discussed in Section 5.3.2.

The experimental data set is obtained from a well control test conducted at Louisiana State University. The test setup, illustrated in Figure 5.10, is detailed in [47]. An 11-bbl gas kick was simulated by injecting natural gas inside the 1.25-in tubing while water-based mud was continuously pumped through the annulus formed by the 3.5-in drill pipe and the 1.25-in tubing, with returns taken through the annulus between the 9.625-in casing and the 3.5-in drill pipe. A manually operated choke manifold was used to provide back-pressure, with the goal of keeping a constant drill pipe pressure throughout the gas circulation. This experiment represents a kick circulation commonly carried out in real practice. The choke opening, mud circulation and gas injection rate recorded during the test were used as inputs to the model and are shown in Figure 5.11. Well geometry, mud properties and drilling inputs are detailed in Table 5.5.

The measured data of the system is plotted in Figure 5.12. We point out an inconsistency in the data in the following remark.

Remark 5.8. *An inconsistency between the pressure and the flow in the zoomed part of the plot of the pump pressure data p_p in Figure 5.12 is detected. As the*

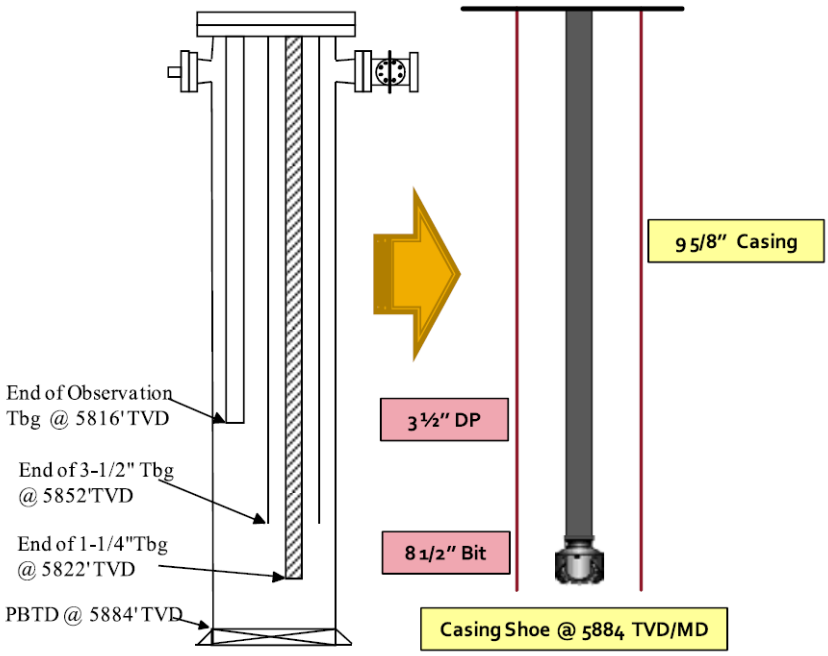

Figure 5.10: Schematic of the test well [5].

Table 5.5: Parameters of the experimental well.

parameter	Length [m]	ID [in]	OD [in]	c_l [m/s]	μ_l [cP]
value	1793	3.068 in	3.5	1202.5	8

pump flow rate is doubled in the period $[35, 55]$ *min, the frictional pressure drop should be quadrupled as the flow inside the drillstring is turbulent. We consider the approximate relation between pump pressure* p_p*, downhole pressure* p_{dh} *as follows:*

$$p_p \approx p_{dh} - \rho g h + \Delta p_{Friction} + \cancelto{0}{\Delta p_{bit}}. \tag{5.14}$$

As no bit is present in the test system, $\Delta p_{bit} \approx 0$*. In [5], it was approximated that* $p_{dh} - \rho g h \approx 27.6$ *bar after turning on the pump. Due to* $p_p \approx 60$ *bar after turning on the pump, the frictional pressure drop would be around 30-35 bar which is a reasonable value for such a well. Therefore, by almost doubling the pump flow rate, an increase of at least 90 bar is expected in the pump pressure.*

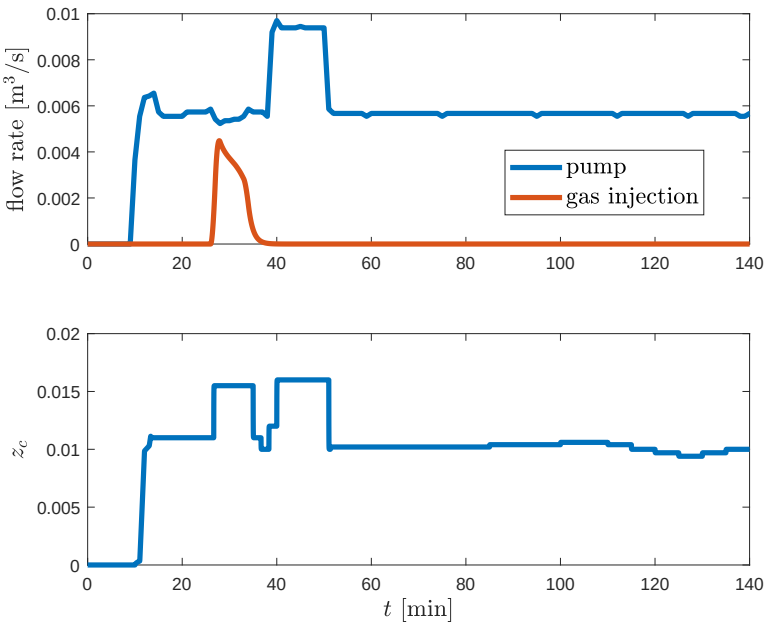

Figure 5.11: Inputs of the test well and simulation framework (z_c: choke opening).

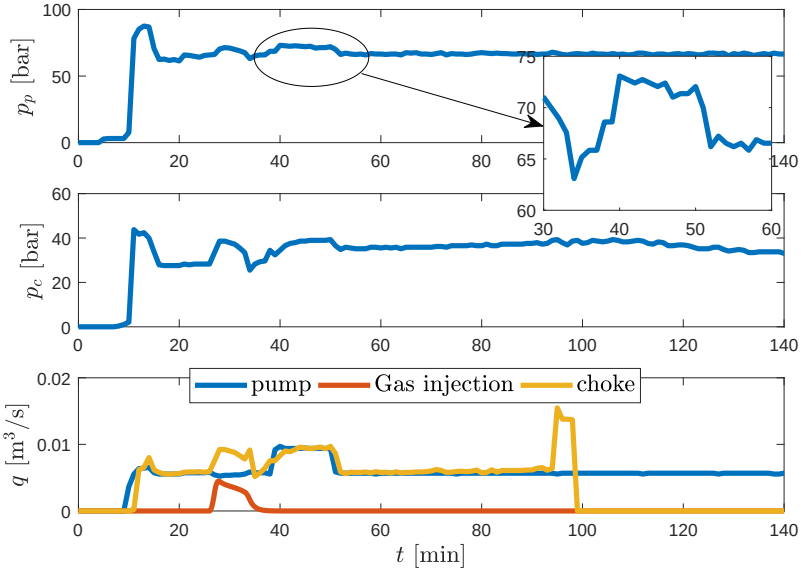

Figure 5.12: Measurements of the system (p_p: pump pressure, p_c: choke pressure, q: volumetric flow rate).

However, the measured data show that the pump pressure increases by less than 10 bar. Therefore, for this sensor, we only trust the steady-state measurements. Regarding other sensors, the fact that in the steady-state, the flow rate through the choke should be almost equal to the pump flow rate is confirmed by the measurements from the flow meters. Therefore, we trust the measurements recorded by the flow meters. We also trust the choke pressure sensor where typically more accurate sensors are installed at the choke manifold.

5.3.2 Model validation for complex influx/loss scenarios

In this section, we present results on the experimental validation of the MPD model presented in Chapter 4. In contrast with the validation for the liquid influx scenario, as the gas flow rate injection into the annulus is known over time, there is no need to introduce a model for the gas reservoir.

Remark 5.9. *A validation of a model similar to the DFM, namely a Reduced DFM, is done in [5]. In Table 5.6, we highlight the difference of the validation procedure in this section with [5].*

Remark 5.10. *Notably, in the field data, no drilling bit is presented while we use a drilling bit to connect the drillstring and the annulus. This bit simulates*

Table 5.6: Comparison of the validation in this study and the validation in [5].

Validation in [5]	Validation in this study
Validation of the Reduced DFM	Validation of the full DFM
Simulation of the flow only in the annulus	Simulation of the flow both in the drillstring and the annulus
State-dependent slip law	Zuber-Findlay slip law
No choke model	choke model (5.12)

the pressure drop due to the change in the direction of the flow going from the drillstring to the annulus.

5.3.3 Parameter identification for gas influx scenario

Here, we identify the unknown parameters used in the DFM either by the available measurements or during an optimization procedure. The choke model can be identified by the measured data and other parameters can be set during an optimization procedure. These parameters include speed of sound in the gas medium c_g, gas viscosity μ_g, pipe roughness coefficient K_f (similar to (5.10)), slip law constant parameters K, S and nozzle area at the bit A_N. For the optimization, the following objective function is minimized by tuning the free parameters:

$$f_g(p_{c\text{Sim}}) = \frac{\|p_{c\text{Field}} - p_{c\text{Sim}}\|_2}{\|p_{c\text{Field}}\|_2}. \tag{5.15}$$

Since the transients in the pump pressure sensor are not reliable, it is not included in the objective function unlike (5.13). Notably, for different choke openings z_c, K_c is obtained as explained in Section 5.2.3.1 and presented in Table 5.7.

Remark 5.11. *To obtain a computationally efficient solution to the optimization problem, the first 18 minutes of the data are not simulated. In this period, the fluid is stagnant and simple hydrostatic relations can replicate the field data. The first few minutes of the transients are also ignored to initialize the system from an almost steady condition. Moreover, as the gas reaches the topside measurement devices around $t = 90$ min (see Figure 5.12 for sudden increase in*

Table 5.7: Choke coefficient as a function of choke opening.

z_c	0	0.0102	0.011	0.012	0.0155	0.016	0.017
K_c	0	7.1e-5	7.8e-5	9.8e-5	10.8e-5	11.2e-5	12e-5

q_c and a sudden decrease afterwards), flow and pressure measurements at the choke side are not reliable due to the two-phase nature of the flow. Therefore, the simulations are stopped at this moment.

Remark 5.12. *Since we only have one set of data, not all parameters are uniquely identifiable and different parameter combination lead to similar objective function values, i.e., the objective function is not too sensitive with respect to parameters. Among these physically admissible parameters, the ones with better approximation of the pump pressure are selected.*

5.3.4 Validation results

For the purpose of validation in the two-phase flow scenario, we obtain the parameter values as reported in Table 5.8. The simulation results are depicted in Figure 5.13. As mentioned in Remark 5.11, we only run the simulations in the time window between $t = 18$ min and $t = 90$ min.

As can be observed from the results (except at the inconsistency mentioned in Remark 5.8 and the time window shown in the top part of Figure 5.12 related to the pump pressure p_p), the DFM with physically reasonable parameters is able to reproduce the field data.

The steady pump pressure shown in the top part of Figure 5.7 corresponds to the field data accurately. However, in the time window where the inconsistency has been detected (see Remark 5.8), the MPD model behaves physically justifiable. At this time window, the frictional pressure drop almost quadruples and

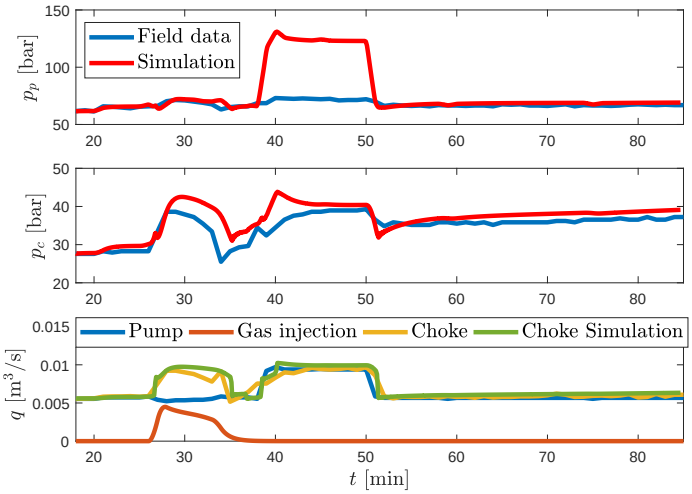

Figure 5.13: Comparison of field data and the simulation results.

Table 5.8: Optimization result for two-phase flow.

parameter	c_g [m/s]	μ_g [Pa.s]	K_f	K	S	A_N [m^2]
optimized	408.09	8e-4	2.5612	1.104	0.0486	0.00352

the pump pressure should change significantly. In the middle and bottom part of Figure 5.7, the choke pressure and the choke flow show a good agreement. The dynamics observed in the data at the choke are due to the gas migration and the pump flow rate change, all of which are captured well by the simulation of the model developed in Chapter 4. Further improvements can be achieved if a better trained choke model (5.12) is constructed using a richer data set.

To track the front of the migrating gas, the volume fraction of gas at different time instants is plotted in Figure 5.14. As expected, around $t = 90$ min, the gas is reaching the top of the annulus and the measurements become unreliable afterwards. The location of the gas front obtained by the MPD model is in a good agreement with the location of the gas front presented in [5].

5.4 Conclusion

The objective of this chapter has been the experimental validation of the MPD model developed in Chapter 4 in case of a liquid and a gas influx. For the case of a liquid influx, the MPD model is coupled with fracture models to simulate

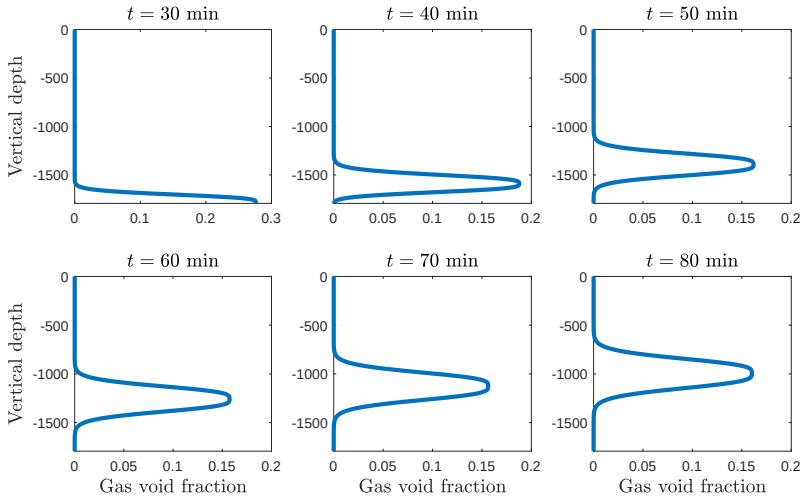

Figure 5.14: The distribution of gas volume fraction in different time instants.

the interaction between the wellbore and the reservoir. The existing models and a new proposed model, based on the mass and momentum conservation laws, aim to predict the mass flow rate from a formation flowing into the annulus. The proposed reservoir model based on a second-order dynamics predicts the mass flow rate of the influx qualitatively and quantitatively more accurately compared to the existing reservoir models. In case of a gas influx, finding a suitable field data is much more challenging. Therefore, a set of data obtained from an experimental well is used with a known gas flow rate injected to the annulus. Hence, no reservoir model is required and the explicit value of the gas influx is injected into the annulus at the bottom of the wellbore. The comparison of the simulation results and the field data showed good predictive capabilities of the MPD model in field circumstances. However, due to the lack of useful dataset in a drilling well, we were not able to validate the identified model over another dataset in the same drilling well.

Part II

Model Order Reduction

The MPD model in the case of single-phase flow and its utilized models in Part I are subject to model order reduction in this part. Since the RB method is not capable of dealing with time-varying, nonlinear and state-dependent boundary conditions, a new ansatz is suggested by decomposing the boundary dynamics and the internal dynamics. The reduction is applied to the high-dimensional linear system of the internal dynamics and the dimension of the nonlinear part of boundary dynamics is not reduced. The ansatz successfully captures nonlinear boundary conditions without generating non-physical spikes at the boundaries. Moreover, to predict the error due to reduction, a new error estimate is proposed for nonlinear systems with local nonlinearities at the boundaries. This error estimate hinges on the interconnection of an internally stable linear system with nonlinear dynamics. The error estimate efficiently predicts the accuracy loss after reduction.

To extend the error estimate for systems with non-Lipschitz distributed nonlinearities such as Burgers' equation, a loop transformation is performed between the linear and nonlinear systems. Still, this error estimate suffers from restriction due to the non-Lipschitz nonlinearities. To lift the restrictions with this type of error estimates, a new hierarchal error estimate for nonlinear systems is proposed based only on the snapshots of the system during the offline phase. Although both error estimates accurately predict the error due to the reduction, the hierarchal error estimate is applicable to a wider range of problems.

Finally, all developments in this part are put together to reduce the dimension of an MPD model for single-phase flow. There are many varying parameters in MPD. Challenging examples of such parameters are the location and number of area discontinuities in drillstring and annulus. To accommodate these parameters into RB structure, a local adaptive enrichment stage is added to the RB procedure. This stage should be performed when the location and number of area discontinuities vary. The proposed method accurately predicts the physics induced by these varying features while maintaining a reasonable small computational time.

Chapter 6

Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions

Many physical phenomena, such as mass transport and heat transfer, are modeled by systems of partial differential equations with time-varying and nonlinear boundary conditions. Control inputs and disturbances typically affect the system dynamics at the boundaries and a correct numerical implementation of boundary conditions is therefore crucial. However, numerical simulations of high-order discretized partial differential equations are often too computationally expensive for real-time and many-query analysis. For this reason, model complexity reduction is essential. In this paper, it is shown that the classical reduced basis method is unable to incorporate time-varying and nonlinear boundary conditions. To address this issue, it is shown that, by using a modified surrogate formulation of the reduced basis ansatz combined with a feedback interconnection and a input-related term, the effects of the boundary conditions are accurately described in the reduced-order model. The results are compared with the classical reduced basis method. Unlike the classical method, the modified ansatz incorporates boundary conditions without generating unphysical results at the boundaries. Moreover, a new approximation of the bound and a new estimate for the error induced by model reduction are introduced. The effectiveness of the error measures is studied through simulation case studies and a comparison with existing error bounds

This chapter is based on "M.H. Abbasi, L.Iapichino, B. Besselink, W.H.A.Schilders, N.van de Wouw, ***Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions***, *Computer Methods in Applied Mechanics and Engineering*, Volume 360, 1 March 2020, 112688".

and estimates is provided. The proposed approximate error bound gives a finite bound of the actual error, unlike existing error bounds that grow exponentially over time. Finally, the proposed error estimate is more accurate than existing error estimates.

6.1 Introduction

Simulations of high-fidelity dynamic models, especially for real-time and many-query analyses, entail highly expensive computations, rendering direct numerical simulation infeasible. To circumvent the so-called curse of dimensionality in such simulations, model order reduction has been employed in the literature and it is becoming an essential tool for controller design, multi-query, real-time simulations and model-based optimization [13]. Several techniques for generating reduced-order models have been proposed such as, e.g., the reduced basis method [89], balanced truncation [80], moment matching [13], and interpolation techniques based on input-output behavior such as IRKA [81]. In particular, the reduced basis method is an efficient approach for dealing with parameterized systems, where the parameters can be, for instance, the physical parameters of a system, geometry properties or boundary conditions [92].

The reduced basis method consists of an offline and online phase. In the offline phase, the essence of the underlying model is extracted by obtaining a, usually small, problem-specific set of basis functions. This can be carried out by computing the solution to the full-order model for a representative set of parameter values. The basis functions can be computed by different methods; the most popular ones are proper orthogonal decomposition (POD) and the greedy algorithm [65], [117]. After generating the basis functions, by exploiting the affine dependency of the full-order model on the varying parameters, a fast, easy-to-compute and reasonably accurate model is built in order to be used in the online phase. In the online phase, the computationally cheap yet accurate model can be solved for each new set of parameters with far less computational effort compared to the full-order model.

While the theory of the reduced basis method for a single, parabolic and elliptic partial differential equation (PDE) has already reached a mature stage [59], [78], [79], [89], systems of PDEs and single hyperbolic PDEs still present an open research field [84], [135], [148], [149]. For hyperbolic problems, shock-capturing model order reduction techniques such as the freezing method [135] and shifted-POD [148] are yet far from generic and still require many problem-specific adaptations and are not considered in this paper.

One critical issue to address in the reduced basis method is the incorporation of the boundary conditions. The reduced basis method is a powerful method for the case of constant boundary conditions. Due to the global nature of the basis functions and their independency over time, handling time- or state-dependent boundary conditions can be challenging [53]. Moreover, dealing with nonlinear

and implicit boundary conditions in the reduced-order model of a system of PDEs is still an open issue [82]. Nonetheless, boundary conditions of any system, either induced by the physical effects or by a controller, play a crucial role in the system dynamics.

In the reduced-order model, different approaches for dealing with parameterized Dirichlet boundary conditions have been introduced [52], [82]. In addition, the formulation of boundary conditions as a differential algebraic equation (DAE) has been discussed in [66]. Nevertheless, the correct implementation of boundary conditions of a hyperbolic system requires the use of characteristic waves [10], [71]. The governing equations of these waves include nonlinear ordinary differential equations (ODEs) [69], [71], introducing extra dynamics at the boundaries. This challenges the construction of reduced-order models. So, the correct implementation of the boundaries in the general case should be investigated more thoroughly.

Besides problems occurring in nonlinear, time-varying boundary conditions in the reduced basis method, an easy-to-compute a posteriori error bound (or estimate) is required, firstly, to speed up the basis generation in the greedy algorithm and, secondly, to assess the accuracy of the reduced solution generated by the reduced basis method. Numerous works are directed towards deriving a residual-based error bound (or estimate) for parabolic PDEs [31], [79], [89], [98], [117]. The error bounds in the parabolic context are well-developed while very few works have focused on developing an error bound (or estimate) for hyperbolic PDEs [84], [182]. The error bound in [84] is dependent on the norm of the state matrix; when this norm is larger than one, the error bound grows exponentially over time and is not useful for assessing the quality of the reduced-order model. The error estimate in [182] is derived by defining a dual system and building a connection between the norm of the residuals and the error in approximating the states. Moreover, this method is also based on the norm of the state matrix and this norm should satisfy some constraints. These constraints are not satisfied in the numerical examples shown in the current paper. So the ultimate problem is that in many cases for hyperbolic problems, no finite error bound can be constructed using existing approaches. Therefore, novel approaches for the construction of error bounds and estimates are needed. Hereto, we first propose a new approximate error bound that does not pertain to the norm of the state matrix. Second, in order to mitigate the conservativeness of the proposed error bound, a new error estimate is introduced by exploiting this error bound. The main contributions of the paper are described below.

In this paper, we propose an efficient strategy for dealing with time-varying and nonlinear boundary conditions. A modified ansatz representation for the reduced basis method, interconnected with a boundary condition solver, is proposed and tested on a hyperbolic system. This method draws inspiration from methods presented in [27], [28], [129] in the scope of balanced truncation for nonlinear systems. Furthermore, to certify the accuracy of the reduced-order

model, a new approximate error bound is proposed. However, this approximate error bound may still lead to conservative results. To sharpen the error bound, an error estimate is introduced based on the obtained approximate error bound and the simulations in the offline phase, similar to the one presented in [182], however, in a different norm realization. Two test cases are considered in this paper. First, a linear advection equation containing source terms with time-varying boundary conditions is investigated. The error bound is applied on this system and we show that, for a certain class of inputs, the proposed error bound is one of the sharpest possible error bounds. Second, the linearized isothermal Euler equations with nonlinear, time-varying boundary conditions are considered. The effect of the boundary conditions on the solution of the reduced-order model is investigated. The isothermal Euler equations with the utilized boundary conditions are commonly used in the simulation of gas and liquid flows in pipelines [126] and for hydraulics drilling simulations in single phase cases [129].

The outline of this paper is as follows. In Section 2, the general linear PDE together with the applied finite-volume scheme and the resulting full-order model are explained. In Section 3, the ingredients of the reduced basis method, the proposed ansatz and the reduced-order model are introduced. In Section 4, the error bound is formulated and the connection between the error bound and the error estimate is explained. In Section 5, numerical results for an advection equation with source term and the linearized isothermal Euler equations are presented. Finally, Section 6 closes the paper with some conclusions and perspectives on future work.

6.2 Problem statement

In this work, the following class of linear PDEs is studied:

$$\begin{cases} \dfrac{\partial u}{\partial t} + J\dfrac{\partial u}{\partial x} = S(u,t,x;\mu), \\ u(0,x;\mu) = u_0(x;\mu), \\ u(t,x_\partial;\mu) = u_\partial(t;\mu), \end{cases} \qquad t \in [0,T], \quad x \in [0,L], \tag{6.1}$$

where $u = u(t,x;\mu)$ is called conservative variable (since this variable should be conserved in each grid cell after the discretization) at time t and spatial location x for a set of parameters $\mu \in \mathcal{D} \subset \mathbb{R}^b$. Here, $\mathcal{D}$ is the domain of varying parameters and b is the number of varying parameters in the system. This system may also contain fixed parameters that do not vary from one simulation to another in a multi-query context. In addition, $J = J(\mu)$ is the Jacobian of the analytical flux function $f(u;\mu)$, i.e., $J(\mu) = \partial f/\partial u$ (J should be independent of u such that the PDE becomes linear), and $S(u,t,x;\mu)$ is the source term. Moreover, the final time of the simulation and the spatial domain length are denoted by T and L, respectively. To analyze the response, an initial condition $u_0(x;\mu)$ and a set of boundary conditions $u_\partial(t;\mu)$ on the boundary points x_∂

are required, which will be elaborated upon in Sections 6.5.1.1 and 6.5.1.2 for the specific cases of an advection equation with source term and the linearized isothermal Euler equations.

Definition 1. *The function $S(u,t,x;\mu)$ is affine with respect to parameters μ if it can be written as $S(u,t,x;\mu) = \sum_j S_u^j(u,t,x)\, S_\mu^j(\mu)$, with state-dependent functions $S_u^j(u,t,x)$ and parameter-dependent functions $S_\mu^j(\mu)$.*

Assumption 6.1. *In (6.1), J only depends on the parameters μ, not the states. J and S are affine with respect to parameters. Moreover, the eigenvalues of J are all real and distinct such that the system is hyperbolic. Finally, to have a linear system, all S_u^j in the expansion of Definition 1 are assumed to be a linear function of conservative variables u.*

Linear PDEs as in (6.1) with properties mentioned in Assumption 6.1 can be discretized with a linear scheme, which is described next.

6.2.1 Discretization using the upwind scheme

First-order Godunov-type schemes together with forward Euler integration numerically solve PDE (6.1) by (parameter dependency is not mentioned here for the ease of notation) [50], [139]

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(U_{i+1}^n, U_i^n) - \mathcal{F}(U_i^n, U_{i-1}^n)\right) + \Delta t\; S(U_i^n, t^n, x_i; \mu), \quad (6.2)$$

where $U_i^n,\ i = 1,\cdots,N_\delta,\ n = 0,\cdots,N_t - 1$, is the average of the conservative variables u over the i-th spatial grid cell at the time instant $t^n = n\Delta t$, with N_δ the number of spatial grid cells and N_t the number of time steps. Also, Δt and Δx denote the temporal and spatial resolution of the discretization. Next, $\mathcal{F}$ is the numerical flux function (discretized counterpart to the analytical flux function f in (6.1)) that is scheme-dependent. For instance, for the upwind scheme [50], [139], the numerical flux function is given by

$$\mathcal{F}(U_{i+1}, U_i) = A^+ U_i + A^- U_{i+1}, \quad (6.3)$$

where $A^+ = R\Lambda^+ R^{-1}$ and $A^- = R\Lambda^- R^{-1}$ with R matrix composed of the right eigenvectors of the Jacobian matrix J, and

$$\Lambda^+ = \begin{bmatrix} {\lambda_1}^+ & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & {\lambda_r}^+ \end{bmatrix}, \quad \Lambda^- = \begin{bmatrix} {\lambda_1}^- & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & {\lambda_r}^- \end{bmatrix}. \quad (6.4)$$

Here, $\lambda_k,\ k = 1,\cdots,r$ are the eigenvalues of the Jacobian matrix J with r as the dimension of the system (6.1). For every $k = 1,\cdots,r$, we denote $\lambda_k^+ =$

$\max(\lambda_k, 0)$ and $\lambda_k^- = \min(\lambda_k, 0)$. If the Jacobian matrix J is not diagonalizable (i.e., Assumption 6.1 is not valid) , other schemes such as the Lax-Friedrichs or Rusanov scheme [113] can be used.

The finite-volume discretization leads to a high-dimensional (though finite-dimensional) model, which will be referred to as the full-order model. This model is introduced in the upcoming section.

6.2.2 Full-order model

In many cases, the equations governing the boundary conditions are time-varying, coupled with the state variables, and/or nonlinear. As a result, the full-order model becomes nonlinear even if the internal dynamic is linear. As an example, since the system of PDEs of interest as in (6.1) is linear and the upwind scheme (6.3) is also linear, a finite-dimensional linear time-invariant (LTI) system is obtained after discretization. However, substituting $i = 1$ and $i = N_\delta$ in (6.2), the conservative variables U_0^n and $U_{N_\delta+1}^n$ are required to compute the solutions. These variables are the boundary values that can be regarded as inputs to the LTI system that render the entire system nonlinear. On the other hand, all nonlinearities enter the system at the boundaries. So, it is natural to decompose the full-order nonlinear model Σ_{nl} into a linear subsystem Σ_{lin} (representing the discretized PDEs in the internal domain) and a nonlinear subsystem Σ_{BC} (representing the boundary dynamics). This is schematically depicted in Figure 6.1 and the models Σ_{nl}, Σ_{lin} and Σ_{BC} are of the form

$$\Sigma_{nl}: \begin{cases} \Sigma_{lin}: \begin{cases} U^{n+1} = A(\mu)U^n + B(\mu)w^n \\ y^n = C_y U^n \\ z^n = C_z U^n \end{cases}, \\ \Sigma_{BC}: \begin{cases} V^{n+1} = G(V^n, z^n, u_c^n; \mu) \\ w^n = \mathcal{G}(V^n, u_c^n; \mu) \\ U_{BC}^n = C_{BC}\begin{bmatrix} (V^n)^T & (w^n)^T \end{bmatrix}^T \end{cases} \end{cases}. \tag{6.5}$$

Here, the superscript T denotes the transpose operator. As depicted in Figure 6.1 and apparent from (6.5), the linear subsystem Σ_{lin} solves for the states U^n. The input to this system comes from boundaries which are governed by Σ_{BC}. This subsystem takes auxiliary outputs z from the linear subsystem and the control inputs u_c from the user and then computes the variables w, which represents the input of the linear dynamics. The actual output y of the system is defined by the user while the auxiliary outputs z are determined based on the interconnection of the two subsystems. In addition, the boundary solver calculates the conservative variables at the boundary location U_{BC}, which will be used in the reduced-order model. Although the boundary values of the conservative variables, U_{BC}, do not play a role in the full-order model, these help us later

in the next section to incorporate the boundary conditions in the reduced-order model correctly.

To introduce the notation in (6.5), $A(\mu) \in \mathbb{R}^{N_\delta \times N_\delta}$, $B(\mu) \in \mathbb{R}^{N_\delta \times m}$, with m the dimension of the inputs w, are parameter-dependent state and input matrices, $U^n \in \mathbb{R}^{N_\delta}$ is the vector of state variables of the linear system Σ_{lin} at time step n. For this system to be stable, $A(\mu)$ should be a Schur matrix (i.e., all eigenvalues of $A(\mu)$ lie in the unit disc in the complex plane). Moreover, V^n is the vector of state variables in the dynamics of the boundary conditions governed by G, $w^n \in \mathbb{R}^m$ is the input of the linear system Σ_{lin}, which is computed by an output function $\mathcal{G}$ in Σ_{BC}. External (control) inputs u_c act at the boundaries. Finally, C_y and C_z are the output matrices for the output of interest y^n and auxiliary outputs z^n. In general, Σ_{nl} in (6.5) expresses a feedback interconnection between Σ_{lin} and Σ_{BC}. However, in some cases, there is no dynamics occurring at the boundaries and the boundary conditions are specified explicitly over time; and in some other cases, the boundaries are dynamically coupled with the output z^n as in Σ_{BC} in (6.5).

Clearly the dimension of Σ_{lin} scales with N_δ, which can be extremely large when fine spatial discretization meshes are used. On the other hand, the dimension of Σ_{BC} is typically of low order compared to N_δ. According to this observation, we assume that most of the computational time is allocated for solving Σ_{lin} rather than Σ_{BC}. This assumption is crucial to justify the speedup after model order reduction via the interconnection approach. To reduce the number of equations to be solved in Σ_{lin}, a reduction approach will be implemented on the linear subsystem, while the number of equations in Σ_{BC} is not changed. Moreover, the boundary incorporation in the reduced basis is another challenge that has to be handled. To deal with the complexity of the linear subsystem of the full-order model (6.5) and the boundary incorporation, the model order reduction technique is introduced next.

Assumption 6.2. *$A(\mu)$ and $B(\mu)$ in the linear subsystem Σ_{lin} in (6.5) are*

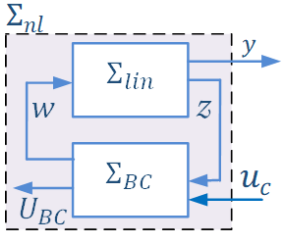

Figure 6.1: The schematic representation of the full-order nonlinear model Σ_{nl} consisting of a feedback interconnection of the linear subsystem Σ_{lin} and a boundary subsystem Σ_{BC}.

affine with respect to parameters μ. Moreover, $A(\mu)$ is Schur for all $\mu \in \mathcal{D}$.

Remark 6.3. *Since $A(\mu)$ is Schur for all $\mu \in \mathcal{D}$, Σ_{lin} is stable. However, the stability of Σ_{lin} does not guarantee the stability of Σ_{nl}.*

Remark 6.4. *In this study, only 1D problems are considered. For the 2D and 3D case, the number of equations to be solved at the boundary is still much lower than for the internal grids. If the boundaries in the higher-dimensional settings are nonlinear, reduction for simulation is less promising if most of the computational time is allocated in solving the boundary conditions. One alternative in the 2D and 3D cases would be the reformulation of the system as partial differential algebraic equations (PDAEs) and reduce the entire system, which is beyond the scope of this paper. For the reduction of parameter-independent PDAEs based on input-output behavior, refer to [146].*

Remark 6.5. *In this study, by using characteristic boundary equations [10], [69], boundary dynamics and the internal dynamics become consistent in the full-order model and no boundary layer appears in the solution.*

6.3 Boundary incorporation within the reduced basis method

The reduced basis method [84], [89] targets the reduction of parametrized problems requiring repeated evaluations or (faster than) real-time simulations, known as many-query and real-time analysis, respectively. This technique consists of two phases with two different objectives. The first phase is the offline phase, which captures the most dominant characteristics of the problem based on the solution of the physical full-order model, the so-called truth solutions, for specific parameter values. At this stage, the dimension N_δ of the full-order space exploited to find the truth solution is large, leading to expensive and time-consuming simulations. The reduced basis method approximates this space with an N-dimensional subspace (typically $N \ll N_\delta$). After finding this subspace, the required reduced-order operators are defined through Galerkin or Petrov-Galerkin projection onto this N-dimensional subspace. In the online phase, the reduced-order model is generated, rendering simulations faster as $N \ll N_\delta$ [89]. In other words, the online computational cost depends on N, rather than N_δ. This fact has two advantages. First, the solution in the online phase can be found faster compared to the full-order model, due to the small dimension of the reduced basis space and, second, the basis functions used for the projection are defined in the full-order space and allow to express the reduced basis solution in the full-order space as well. The dimension of the full-order space depends on the number of grid cells of the computational domain and can be increased to attain more accurate results. Even though increasing the number of grid cells increases the offline cost, it does not influence the online cost significantly. We note that Assumption 6.2 enables such offline-online decomposition [20], [89].

Numerically, the reduced basis solution of (6.5) in the finite-dimensional setting collected for all grid cells at time instant t^n is denoted by $\hat{U}^n(\mu)$. The reduced basis solution, $\hat{U}^n$, is computed as a linear combination of spatial- and time-dependent vectors as

$$\hat{U}^n(\mu) = \sum_{i=1}^{N} a_i^n(\mu)\phi_i, \tag{6.6}$$

where $\Phi = \{\phi_i,\ i = 1, \cdots, N\}$ is the set of N reduced basis vectors (i.e., reduced basis functions evaluated at the discrete computational domain). These basis vectors are the dominant modes of specific truth solutions of (6.5) for certain values of the parameters μ. The basis vectors ϕ_i can be obtained by using the POD-greedy algorithm in the offline phase [31], [89]. The POD approach to generate ϕ_i is illustrated in Algorithm 4 and POD-greedy is explained in Algorithm 5. The dynamics of the generalized coordinates, $a_i^n(\mu)$, can be obtained by Galerkin or Petrov-Galerkin projection of the problem (6.5) onto the space Φ.

Remark 6.6. *If nonlinearities exist or the system is non-affine with respect to parameters, the empirical interpolation method (EIM) or the discrete empirical interpolation method (DEIM) can be performed on the system to prepare it for Galerkin or Petrov-Galerkin projection onto the space Φ [44], [78], [79]. However, nonlinear and non-affine systems are not studied in this work.*

Remark 6.7. *The dyadic product of* (6.6) *may not be suitable for problems with shocks. In this study, the boundary conditions change smoothly and therefore no shock occurs in the system. For systems with shocks, other approaches such as Shifted-POD [148] or the freezing method [135] should be followed; however, their applicability is still limited.*

6.3.1 New ansatz formulation

The ansatz formulation (6.6) cannot capture time-varying boundary conditions [52]. This is attributed to the global and time-invariant nature of the reduced

Algorithm 4: POD algorithm, POD($\boldsymbol{U}, n_{\mathrm{POD}}$)

Input: Snapshots $\boldsymbol{U} = \{U^0, \cdots U^{N_t}\} \in \mathbb{R}^{N_\delta \times N_t}$, number of basis vectors n_{POD}

Output: $\phi_i \in \mathbb{R}^{N_\delta \times n_{\mathrm{POD}}}$

1 Perform a Singular Value Decomposition on the snapshots, $\boldsymbol{U} = MSV$

2 $\phi_i = M(:, 1 : n_{\mathrm{POD}})$ are the first n_{POD} vectors of the left singular vectors M

basis vectors. As an example, POD typically employs basis vectors that are constant over time. As a result, if the boundary values are constant over time and not parameter dependent, the boundary conditions are always satisfied. However, when the boundary conditions change over time, the reduced solution cannot cope with these changes since the coefficients of the basis vectors, a_i^n, are determined by the internal dynamics. This is more important when the system has two time-varying boundary conditions at both ends. For many industrial applications, such as flow inside a pipe [71] and managed pressure drilling (MPD) [129], boundary conditions are coupled with the state variables and are even nonlinear. Since external (control) inputs of such systems also usually act at the boundaries, correct implementation of the boundaries in the reduced-order model is crucial.

To handle the time dependency and nonlinearities associated with the boundaries, the dyadic form of the reduced basis ansatz (6.6) is changed to

$$\hat{U}^n(\mu) = U_B(\hat{U}^n_{BC}) + \sum_{i=1}^{N} a_i^n(\mu)\phi_i, \tag{6.7a}$$

$$U_B(\hat{U}^n_{BC})\Big|_{x_\partial} = \hat{U}^n_{BC}, \tag{6.7b}$$

$$\phi_i|_{x_\partial} = 0, \tag{6.7c}$$

where $\hat{U}^n_{BC}$ is the reduced solution at specified locations x_∂ on the boundary and $U_B(\hat{U}^n_{BC})$ is a vector (a function evaluated at the discrete computational domain) that enables the reduced basis solution $\hat{U}^n$ to satisfy the boundary conditions. In other words, the vector U_B encodes the exact satisfaction of only the boundary conditions at the reduced level as in (6.7b) (it can be an interpolation inside the internal domain). Then, (6.7c) states that ϕ_i should vanish at the location of the specified boundary conditions.

The logic behind choosing U_B is problem-specific and some examples are given in Section 6.5. In general, we propose to choose U_B by considering a linear interpolation between the boundaries x_∂ of the system (6.1).

One requirement for the correct incorporation of the boundary conditions is that the basis vectors in (6.7) should vanish at the boundaries (see (6.7c)). In order to satisfy this requirement, the snapshots for applying the POD are modified. Instead of applying the POD to the snapshots of the solution $U^n(\mu)$, the POD is applied to the set of modified snapshots $U^{n*}(\mu)$ defined as below,

$$U^{n*}(\mu) = U^n(\mu) - U_B(U^n_{BC}), \tag{6.8}$$

where U^n_{BC} is the truth solution at the boundaries. The modified snapshots $U^{n*}(\mu)$ vanish at the location of the specified boundaries, enforcing the POD to generate basis vectors that are zero at these boundaries.

The methodology to obtain the reduced-order model with the modified ansatz (6.7a) at the discrete level is illustrated next.

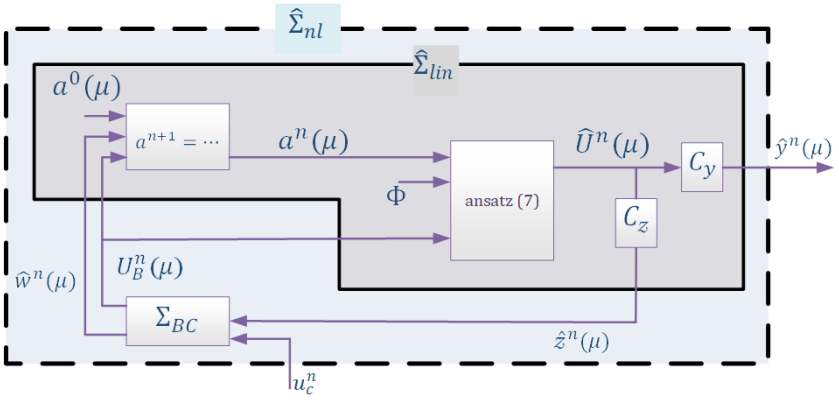

Figure 6.2: The schematic representation of handling boundary conditions in the reduced-order model in (6.9).

6.3.2 Reduced-order model by Galerkin projection

As mentioned earlier, we focus on linear PDEs as in (6.1), discretized with linear schemes such as the upwind scheme (6.3). By doing so, on the one hand, the equations governing the internal domain form an LTI system and thus the dimension of the equations can be reduced by the reduced basis method. On the other hand, the dynamics of the boundary conditions are time-variant, nonlinear, and act locally. As the correct implementation of the boundaries is uncompromisable, the number of equations in Σ_{BC} in (6.5) are not reduced. This method shares some features with the method introduced in [28], [29], [129] in the balanced truncation setting and in [66] in the finite-element setting.

The linear subsystem Σ_{lin} in (6.5) can be reduced to another linear system $\hat{\Sigma}_{lin}$ of lower order by the reduced basis method, but the local and finite boundary equations Σ_{BC} remain full-order. Generally speaking, the local nonlinearities of the boundary conditions are hard-coded in this approach. To provide some insights on this technique, the schematic view of the method is shown in Figure 6.2. This figure illustrates that auxiliary outputs, $\hat{z}^n = C_z\hat{U}^n$, that are necessary for computing the outputs of Σ_{BC}, w^n and $\hat{U}_{BC}^n$ (see also Figure 6.1), are provided via a feedback interconnection. Then, the outputs from Σ_{BC} are fed into the reduced linear system $\hat{\Sigma}_{lin}$ to incorporate the effect of the boundary conditions.

To obtain the reduced-order model corresponding to the full-order model (6.5), we drop the parameter dependency on μ for notational simplicity. The parameter dependency can be included straightforwardly using the affine property of the system, as explained in [84]. After finding the reduced basis space $\Phi \in \mathbb{R}^{N_\delta \times N}$, the Galerkin projection can be executed. After the projection, the

states of the reduced system $\hat{\Sigma}_{lin}$ are the generalized coordinates a_i^n. Following the explanation above regarding the reduction strategy and considering (6.7) and the orthogonality of Φ, we obtain the reduced-order system as follows:

$$\hat{\Sigma}_{nl}: \begin{cases} \hat{\Sigma}_{lin}: \begin{cases} a^{n+1} = \hat{A}_a a^n + \hat{B}\hat{w}^n + \hat{A}_{BC} U_B^n - \Phi^T U_B^{n+1}, \\ \hat{U}^n = U_B^n + \Phi a^n, \\ \hat{y}^n = C_y \hat{U}^n, \\ \hat{z}^n = C_z \hat{U}^n, \end{cases} \\ \Sigma_{BC}: \begin{cases} \hat{V}^{n+1} = G(\hat{V}^n, \hat{z}^n, u_c^n), \\ \hat{w}^n = \mathcal{G}(\hat{V}^n, u_c^n), \\ \hat{U}_{BC}^n = C_{BC} \begin{bmatrix} (\hat{V}^n)^T & (\hat{w}^n)^T \end{bmatrix}^T, \\ U_B^n = U_B(\hat{U}_{BC}^n), \end{cases} \end{cases} \tag{6.9}$$

where $\hat{A}_a = \Phi^T A \Phi$, $\hat{A}_{BC} = \Phi^T A$, $\hat{B} = \Phi^T B$, $a = [a_1, \cdots, a_N]^T$. The second equation in $\hat{\Sigma}_{lin}$ in (6.9) is the ansatz (6.7a). For an example of U_B, see (6.35) in Section 6.5.1.2 for the linear isothermal Euler equations. In (6.9), $\hat{A}_a \in \mathbb{R}^{N \times N}$ and $\hat{B} \in \mathbb{R}^{N \times m}$ are both of low dimension. Recall that N is the number of basis vectors with $N \ll N_\delta$ and m is the dimension of w^n. The initial condition of the system is $a^0 = \Phi^T U^0$ where U^0 is the initial condition of the original system. However, some dependencies on the actual degree of freedom N_δ still exist in (6.9) due to $\hat{A}_{BC}$, Φ^T and U_B. Resolving this issue is problem-specific, which is illustrated in Sections 6.5.1.1 and 6.5.1.2 for two test cases.

Remark 6.8. *In the reduced-order model, an inconsistency between the solution at the internal domain and the solution at the boundaries appears when Galerkin projection is applied (see Section 6.5.2). To resolve this, the effect of boundary conditions is subtracted from the internal domain and added later on by an input-related term to avoid boundary layers. One can also reformulate the problem with the boundary condition as a DAE problem. Note that in the current study, the boundary conditions are dynamic and not characterized by mere algebraic constraints. In a way, this type of decomposition preserves the DAE structure, but its numerical solution is derived based on the interconnection approach presented here. In addition, this also preserves the differentiation index of the DAE. If the full DAE is reduced, then the dynamics at the boundary might also change and inconsistencies might still appear. Here, only the differential part of the DAE is reduced and the boundary related part is retained in the exact form. Moreover, the numerical solution of the DAEs is hard to obtain as the problem might become stiff. These reasons have motivated us to pursue the decomposition approach.*

Remark 6.9. *System* (6.9) *still depends on the dimension of the full-order model due to the lifting in the second equation of $\hat{\Sigma}_{lin}$. This lifting can be incorporated*

in the output equations as $\hat{y}^n = C_y U_B^n + C_y \Phi a^n$ *(similar for* $\hat{z}^n$*) and precomputing* $C_y\Phi$ *(*$C_z\Phi$*) for use in the online phase.*

In the next section, the bound and estimate for the error induced by replacing the full-order model with the reduced-order model are discussed.

6.4 Error bound and estimate for the reduced-order model

In order to certify the accuracy of the reduced basis solution and to accelerate the offline phase by appropriate snapshot selection, an error bound (or estimate) has to be provided. In this study, we are interested in a reduced-order model that provides an accurate prediction of the output y (to be specified by the user of the reduction approach) of the full-order nonlinear model. To derive a bound for the output of a system, the concept of the ℓ_2-gain of the system is exploited.

Definition 2 ([101])**.** *The stable system* Σ_{lin} *as in* (6.5) *has a bounded* ℓ_2*-gain of less than* γ *from input* w *to output* y *if*

$$\sum_{n=0}^{\infty} \|y^n\|^2 \leq \gamma^2 \sum_{n=0}^{\infty} \|w^n\|^2, \tag{6.10}$$

for all trajectories of Σ_{lin} *with zero initial condition and where* $\|\cdot\|$ *is the Euclidean norm. We will use the shorthand notation* $\|y\|_{\ell_2} \leq \gamma \|w\|_{\ell_2}$*, where* $\|y\|_{\ell_2} = \sqrt{\sum_{n=0}^{\infty} \|y^n\|^2}$.

Definition 2 implies that for a bounded input with respect to the ℓ_2-norm, the output is also bounded with respect to the same norm. In other words, γ in the inequality (6.10) can be interpreted as a bound on the ratio between the energies of output and input. This kind of inequality can be defined for any input-output pair of the system. If the system is LTI and asymptotically stable, as we assume to be the case for Σ_{lin} in (6.5) for all $\mu \in \mathcal{D}$ (Assumption 6.2), it can be proved that the ℓ_2-gain of a linear system equates to the $\mathcal{H}_\infty$-norm of the system [160] for which computationally efficient tools are available [19], [34]. As the final system before applying the Galerkin projection should be cast into a linear model with respect to the states, the concept of ℓ_2-gain is also applicable in the reduced basis context. Here, we aim to exploit this notion to derive an error bound for substituting the full-order nonlinear model (6.5) with the reduced-order nonlinear model (6.9). For developing such an error bound, the dynamics that govern this error is required, which is the topic of Section 6.4.1. Then, we introduce an expensive yet accurate error bound in Section 6.4.2.2. As we are interested in easy-to-compute and cheap error *estimates*, a fast error estimate is introduced based on the error bound in Section 6.4.3.

6.4.1 Error dynamics

Recall that $\hat{U}^n$ is the result computed by the reduced basis method from (6.9). As this variable satisfies the reduced-order model, and not exactly the full-order model (6.5), by substituting $\hat{U}^n$ into (6.5), a residual denoted by $\mathcal{R}^n \in \mathbb{R}^{N_\delta}$, at each time step, appears in the formulation of the full-order dynamics

$$\Sigma_{nl}: \begin{cases} \Sigma_{lin}: \begin{cases} \hat{U}^{n+1} = A(\mu)\hat{U}^n + B(\mu)\hat{w}^n + \mathcal{R}^n, \\ \hat{y}^n = C_y\hat{U}^n, \\ \hat{z}^n = C_z\hat{U}^n, \end{cases} \\ \Sigma_{BC}: \begin{cases} \hat{V}^{n+1} = G(\hat{V}^n, \hat{z}^n, u_c^n; \mu), \\ \hat{w}^n = \mathcal{G}(\hat{V}^n, u_c^n; \mu), \\ \hat{U}_{BC}^n = C_{BC}\begin{bmatrix} (\hat{V}^n)^T & (\hat{w}^n)^T \end{bmatrix}^T. \end{cases} \end{cases} \tag{6.11}$$

It should be noted that the dynamics of the boundary conditions Σ_{BC} are fully preserved and no residual appears in their dynamics. By defining the reduction error by $e^n = U^n - \hat{U}^n$, the error dynamics can be obtained by using (6.11) and (6.5):

$$\Sigma_{nl}^e: \begin{cases} \Sigma_{lin}^e: \begin{cases} e^{n+1} = A(\mu)e^n + B(\mu)e_w^n - \mathcal{R}^n, \\ e_y^n = C_y e^n, \\ e_z^n = C_z e^n, \end{cases} \\ \Sigma_{BC}^e: \begin{cases} e_V^{n+1} = G(V^n, z^n, u_c^n; \mu) - G(\hat{V}^n, \hat{z}^n, u_c^n; \mu), \\ e_w^n = \mathcal{G}(V^n, u_c^n; \mu) - \mathcal{G}(\hat{V}^n, u_c^n; \mu), \\ e_{U_{BC}}^n = C_{BC}\begin{bmatrix} (e_V^n)^T & (e_w^n)^T \end{bmatrix}^T, \end{cases} \end{cases} \tag{6.12}$$

where other output errors are denoted as $e_{(\cdot)} = (\cdot) - \hat{(\cdot)}$ and $V^n = e_V^n + \hat{V}^n$. Thus, the error dynamics generally contain two inputs. One input is e_w^n, which affects the error dynamics if the boundary values are not estimated perfectly. Although the boundary dynamics are exactly preserved, due to the fact that boundary values are determined together with an approximation of the linear model, the boundary values are not exact either, which gives an error e_w^n. The other input of the error dynamics is the residual $\mathcal{R}^n$, which contributes to the error due to the inaccuracy of $\hat{U}^n$ in approximating U^n. Figure 6.3 explains the interconnection within (6.12). Next, bounds on the output error of the interconnected system (6.12) are introduced.

6.4.2 ℓ_2 and ℓ_∞ error bounds

As the error dynamics have the dimension of the full-order model, it cannot be used for efficiently determining the error and, therefore, bounds on the output of the error dynamics should be set. For systems with state-dependent boundary conditions, we propose the ℓ_2 error bound and for systems with boundary conditions explicitly stated over time, we propose the ℓ_∞ error bound as below.

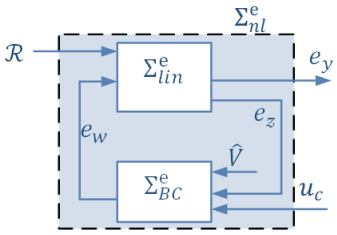

Figure 6.3: The schematic representation of the error dynamics (6.12) consisting of a feedback interconnection of the linear and nonlinear part.

6.4.2.1 ℓ_2 error bound

To impose a bound on the output e_y as in (6.12) by the ℓ_2-gain theory, we note that Assumption 6.2 implies the existence of gains $\gamma^{e_y e_w}$ and $\gamma^{e_y \mathcal{R}}$ such that

$$\|e_y\|_{\ell_2} \leq \gamma^{e_y e_w} \|e_w\|_{\ell_2} + \gamma^{e_y \mathcal{R}} \|\mathcal{R}_s\|_{\ell_2}, \tag{6.13}$$

where γ^{yu} is the $\mathcal{H}_\infty$-norm (ℓ_2-gain) of the system from input u to output y and $\mathcal{R}_s$ is the two-norm of the residual over the spatial domain at each time instant, i.e., $\mathcal{R}_s^n = \|\mathcal{R}^n\|$.

One ingredient for the calculation of the bound is the ℓ_2-norm of the error induced at the boundaries, $\|e_w\|_{\ell_2}$. This norm can be bounded by the Lipschitz constant of the nonlinear function governing the boundary. For instance in (6.12), we have:

$$\begin{aligned} e_w^n &= \mathcal{G}(V^n, u_c^n) - \mathcal{G}(\hat{V}^n, u_c^n), \\ \|e_w^n\| \leq L_{\mathcal{G}} \|e_V^n\| &\Longrightarrow \|e_w\|_{\ell_2} \leq L_{\mathcal{G}} \|e_V\|_{\ell_2}, \end{aligned} \tag{6.14}$$

where $L_{\mathcal{G}}$ is the global Lipschitz constant of the nonlinear function $\mathcal{G}(V^n, \cdot)$ with respect to its first argument.

Remark 6.10. *For some cases, the function $\mathcal{G}(V^n, \cdot)$ might not be globally Lipschitz. Then, a local Lipschitz constant can be employed. In this study, we approximate such local Lipschitz constant by*

$$L_{\mathcal{G}} \approx \max_n \left(\left| \frac{\partial \mathcal{G}}{\partial V} |_{V=\hat{V}^n} \right| \right), \tag{6.15}$$

which becomes more accurate if the solutions of the full-order and reduced-order systems are closer. The computation of $\|e_V\|_{\ell_2}$ in (6.14) is dependent on its governing dynamics G, which will be explained in one of the test cases in Appendix E. The assumption of reduced and full solution being close to each other becomes more realistic when the number of basis vectors increases. For a low number of

basis vectors, this assumption is not accurate and (6.13) *(combined with* (6.15)*) cannot be a true bound on the error. When the number of basis vectors is low, a possible alternative is to restrict the solution domain. Then, during the offline phase, the local Lipschitz constant over this restricted solution manifold can be approximated based on* (6.14) *and the available actual error (see Remark .1). In this approach, potentially larger discrepancies between the solutions of the full-order and reduced-order systems are always allowed at the expense of higher* $L_{\mathcal{G}}$*.*

Finally, the last ingredient for computing the error bound is $\|\mathcal{R}^n\|$ and consequently $\|\mathcal{R}_s\|_{\ell_2}$. This computation should also scale with N in the online phase to make the error bound computationally feasible. To analyze this, rearranging the first equation in (6.11) and dropping the parameter dependency yields:

$$\mathcal{R}^n = \hat{U}^{n+1} - (A\hat{U}^n + B\hat{w}^n). \tag{6.16}$$

By using the ansatz $\hat{U}^n = \Phi a^n + U_B^n$ and multiplying the above equation with its transpose, we obtain

$$\begin{aligned}
\|\mathcal{R}^n\|^2 = \mathcal{R}^{nT}\mathcal{R}^n = {} & a^{n+1^T}\Phi^T\Phi a^{n+1} + a^{n+1^T}\boxed{\Phi^T U_B^{n+1}} + \boxed{U_B^{n+1^T}\Phi}\, a^{n+1^T} + \\
& \boxed{U_B^{n+1^T}U_B^{n+1}} - \left(a^{n+1^T}\Phi^T A\Phi a^n + a^{n+1^T}\boxed{\Phi^T A U_B^n} + a^{n+1^T}\Phi^T B\hat{w}^n\right) - \\
& \left(\boxed{U_B^{n+1^T}A\Phi}\, a^n + \boxed{U_B^{n+1^T}AU_B^n} + \boxed{U_B^{n+1^T}B}\,\hat{w}^n\right) - a^{nT}\Phi^T A^T\Phi a^{n+1} \\
& - a^{nT}\boxed{\Phi^T A^T U_B^{n+1}} - \boxed{U_B^{nT}A^T\Phi}\, a^{n+1} - \boxed{U_B^{nT}A^T U_B^{n+1}} + \\
& \left(a^{nT}\Phi^T A^T A\Phi a^n + a^{nT}\boxed{\Phi^T A^T A U_B^n} + a^{nT}\Phi^T A^T B\hat{w}^n\right) + \\
& \left(\boxed{U_B^{nT}A^T A\Phi}\, a^n + \boxed{U_B^{nT}A^T AU_B^n} + \boxed{U_B^{nT}A^T B}\,\hat{w}^n\right) - \\
& (\hat{w}^n)^T B^T\Phi a^{n+1} - (\hat{w}^n)^T\boxed{B^T U_B^{n+1}} + \\
& \left((\hat{w}^n)^T B^T A\Phi a^n + (\hat{w}^n)^T\boxed{B^T A U_B^n} + (\hat{w}^n)^T B^T B\hat{w}^n\right).
\end{aligned} \tag{6.17}$$

Most operators in the above equation such as $\Phi^T A^T A\Phi \in \mathbb{R}^{N\times N}$ can be computed offline and used in the online phase for fast computation of the norm of the residual. The efficient computation of the terms in the dashed boxes will be explained in Section 6.5.1.1 and 6.5.1.2 as we define U_B for each test case separately. For more information, refer to [84]. As the two-norm of the residual over the spatial domain, $\mathcal{R}_s$, can be computed cheaply in the reduced basis context, its ℓ_2-norm over time is also cheaply computable.

6.4.2.2 ℓ_∞ error bound

For a single PDE with time-dependent (not state-dependent) boundary conditions, we have the value of w^n for all time instants. Therefore, the inclusion of Σ_{BC} is not required and we have $e_w^n = 0$ for all n. In this case, the error dynamic reduces to

$$\Sigma_{lin}^e : \begin{cases} e^{n+1} = A(\mu)e^n - \mathcal{R}^n \\ e_y^n = C_y e^n \end{cases}. \tag{6.18}$$

Based on the error dynamics (6.18), the ℓ_2 error bound (6.13) for these types of systems is as follows:

$$\|e_y\|_{\ell_2} \leq \gamma^{e_y \mathcal{R}} \|\mathcal{R}_s\|_{\ell_2}, \tag{6.19}$$

and the error bound can be compared with an error bound constructed by using the notion of input-to-state-stability (ISS) [101], which leads to a bound pointwise in (discretized) time. By solving (6.18) recursively over time for a given parameter μ, the error at each time instant is given by

$$e^{n+1} = A(\mu)^{\{n+1\}} e^0 - \sum_{i=0}^{n} A(\mu)^{\{n-i\}} \mathcal{R}^i, \tag{6.20}$$

where the notation $A^{\{n\}}$ is used to indicate the n-th power of the matrix A, i.e.,

$$A^{\{n\}} = \underbrace{A \times A \times \cdots A}_{n \text{ times}}. \tag{6.21}$$

The use of (6.20) leads to a bound on the norm of the error as

$$\left\|e^{n+1}\right\| \leq \kappa_{ISS} := \left\|A(\mu)^{\{n+1\}}\right\|_2 \left\|e^0\right\| + \sum_{i=0}^{n} \left\|A(\mu)^{\{n-i-1\}}\right\|_2 \mathcal{R}_s^i, \tag{6.22}$$

where $\|A\|_2$ is the two-norm of matrix A. The ISS-like bound (6.22), although expensive to compute due to calculation of $A(\mu)^{\{i\}}$ and the corresponding norms, is the sharpest possible error bound with the available information. The norm of the matrix should be calculated at each time step and this renders the usage of this bound computationally unattractive. However, this bound is a good reference measure for the performance of other error bounds. Instead, the error bound in (6.13) requires the norm calculation only once, i.e., much less computation than the bound (6.22). However, there are also two drawbacks of the error bound (6.13). First, calculation of $\mathcal{H}_\infty$-norms, even though only needed once, is expensive, and second, this error bound may be conservative. For resolving these two issues, an error estimate based on the error bound (6.13) is developed, as presented in the next section.

6.4.3 Error estimate for the reduced-order model

To resolve the problem of the expensive calculation of the $\mathcal{H}_\infty$-norm in the online phase, as needed in the ℓ_2 error bound (6.13), all required $\mathcal{H}_\infty-$norms, $\gamma^{e_y e_w}$ and $\gamma^{e_y \mathcal{R}}$, are computed and stored offline for all parameters in the discrete parameter domain $\mathcal{D}_h \subset \mathcal{D}$. These norms correspond to the ℓ_2-gain from different inputs to the output of Σ_{lin} that are present in (6.13) for all $\mu \in \mathcal{D}_h$. Here, as only the norms corresponding to the members of $\mathcal{D}_h$ are saved, memory demands are not increased. Then, in the online phase, for any new parameter that does not lie in $\mathcal{D}_h$, the closest element in $\mathcal{D}_h$ (in the Euclidean sense) is found and the norm of that parameter in $\mathcal{D}_h$ is used instead of the norm of the original parameter used in the online phase. If the distance from many parameter sets are the same, the one with the higher $\mathcal{H}_\infty$-norm is chosen.

Remark 6.11. *Since the $\mathcal{H}_\infty$-norms are not computed exactly, but rather computed approximately based on the discrete parameter domain, the expression* (6.13) *is not mathematically a bound and would be an approximation of the error bound. Henceforth, wherever we mention an error bound, we mean an approximation of the error bound.*

For resolving the problem of conservativeness, a similar approach to the one in [182] but in a different norm realization is used. Specifically, the bound (6.13) is adapted to the estimate

$$\|e_y\|_{\ell_2} \leq \kappa^{e_y} := \bar{\rho}\gamma^{e_y e_w}\|e_w\|_{\ell_2} + \bar{\rho}\gamma^{e_y \mathcal{R}}\|\mathcal{R}_s\|_{\ell_2}, \tag{6.23}$$

where $\bar{\rho}$ is the so-called transition factor that provides a measure of the average conservativeness of the bound (6.13). In particular, $\bar{\rho}$ can be computed after defining

$$\bar{\rho}_i^f = \max_n \frac{\sqrt{\sum_{j=1}^{n} \left\| e_y^j\left(\mu_i^*\right)\right\|^2}}{\gamma^{e_y e_w}\sqrt{\sum_{j=1}^{n} \left\|e_w^i\left(\mu_i^*\right)\right\|^2} + \gamma^{e_y \mathcal{R}}\sqrt{\sum_{j=1}^{n} \left(\mathcal{R}_s^j\left(\mu_i^*\right)\right)^2}}, \tag{6.24}$$

where e_y is the actual error computed in the offline phase for a parameter set μ_i^* as in (6.12). The parameter μ_i^* is the parameter selected during the greedy algorithm in the i-th stage of the greedy algorithm in offline phase, $i = 1, \cdots, N$. The denominator of (6.24) is motivated by (6.13).

There are two alternatives for calculating $\bar{\rho}$. The first one is that we choose the last entry of $\bar{\rho}^f$ as the transition factor,

$$\bar{\rho} = \bar{\rho}_N^f. \tag{6.25}$$

Algorithm 5: OFFLINE: Basis vector generation using POD-greedy

Input: $\mathcal{D}_h, \mu_0, \epsilon$

Output: $\Phi, \bar{\rho}, \gamma^{e_y e_w}, \gamma^{e_y \mathcal{R}}\ \forall \mu \in \mathcal{D}_h$

Compute the required $\mathcal{H}_\infty$-norms that appear in (6.13) for all $\mu \in \mathcal{D}_h$

Initialization: $i = 1, \mu_1^* = \mu_0, \Phi = [\,], \bar{\rho} = [\,]$, $\kappa^{e_y}(\mu^*) = 2\epsilon$

while $\kappa^{e_y}(\mu^*) > \epsilon$ **do**

  Compute the truth solution by the parameter set μ_i^* as in (6.5), $S_{FV}\left(\mu_i^*\right) = \{U^n\left(\mu_i^*\right)\}_{n=0}^{N_t}$,

  $\bar{U}\left(\mu_i^*\right) = S_{FV}\left(\mu_i^*\right) - \Phi\Phi^T S_{FV}\left(\mu_i^*\right)$,

  $\varphi_i = \mathrm{POD}(\bar{U}\left(\mu_i^*\right), 1)$ as in Algorithm 4,

  Enrich reduced basis space, $\Phi = [\Phi, \varphi_i]$, where φ_i is the first POD mode of $\bar{U}\left(\mu_i^*\right)$,

  For all members (or the last entry) of μ^*, solve (6.9) to compute $S_{RB}\left(\mu_i^*\right) = \{\hat{U}^n\left(\mu_i^*\right)\}_{n=1}^{N_t}$,

  Set $e_y\left(\mu_i^*\right) = C_y\left(S_{FV}\left(\mu_i^*\right) - S_{RB}\left(\mu_i^*\right)\right)$ and compute the denominator of (6.24) at each time instant,

  Compute $\bar{\rho}^f j$ as in (6.24) for $j = 1, \cdots, i$,

  Run the greedy algorithm for all $\mu \in \mathcal{D}_h$ and find $\mu_{i+1}^* = \arg\max_{\mu \in \mathcal{D}_h} \kappa^{e_y}(\mu)$, calculated by (6.23) (the error indicator here, κ^{e_y}, can be replaced by the actual error or any other error indicator),

  $i = i + 1$,

Compute the projected matrices of the reduced-order model.

The second one is

$$\bar{\rho} = \max(\bar{\rho}_i^f), \tag{6.26}$$

By embedding these transition factors into error estimates, the offline and online algorithms in the reduced basis method are illustrated in Algorithms 5 and 6. Other definitions of $\bar{\rho}^f$ can be straightforwardly tested, such as computing (6.24) only at the last time instant, which is a good option when the total error after the simulation horizon is interesting for the user. In this study, the options in (6.25) and (6.26) are tested. Obviously, using (6.26) leads to more conservative error estimates than (6.25).

In the next section, the effectiveness of the proposed error bound and estimate is tested in numerical case studies.

Algorithm 6: ONLINE: Reduced basis solution by Galerkin projection

Input: μ^o for online simulation, the required $\mathcal{H}_\infty$-norms for all $\mu \in \mathcal{D}_h$, the reduced basis space Φ, $\bar{\rho}$, projected matrices of the reduced-order model.

Output: $\hat{U}(\mu^o)$, κ^{e_y}

Find $\bar{\mu}^o = \arg\min_{\mu \in \mathcal{D}_h} \|\mu - \mu^o\|$

Combine the projected matrices considering affine property of the system (6.5) to obtain the low-order dynamics in (6.9),

for $n = 1 : N_t$, **do**

  Compute the RB generalized coordinates, $a^n(\mu^o)$ based on the reduced-order model (6.9),

  Compute the RB solution $\hat{U}^n(\mu^o)$ via (6.7)

  Compute the norm of the residual $\mathcal{R}_s^n$ as in (6.17),

  Compute the error estimator via (6.23) by using $\gamma^{yu}(\bar{\mu}^o)$ and $\mathcal{R}_s^n$,

  $n = n + 1$,

Remark 6.12. *To enable the computation of the reduced solution and the corresponding error estimate, the following items should be stored in the offline phase and used in the online phase:*

- *the parameter-independent operators obtained after the Galerkin projection, which are of the dimension of the reduced space,*
- *the ℓ_2-gains from different inputs to different outputs for all $\mu \in \mathcal{D}_h$.*

The gains in the second item are scalar constants and require only little memory to be stored.

6.5 Numerical case studies

In this section, first the PDE models are introduced and the associated adaptations are discussed. Then, the reduced basis solution with the classical and the proposed ansatz (equations (6.6) and (6.7)) are compared. Finally, the error bounds and estimates are applied on the discretized systems and the results are compared.

6.5.1 Case studies

As mentioned earlier, two well-known examples of models of the form (6.1) are an advection equation with a source term and the linearized isothermal Euler equations, which are discussed below.

6.5.1.1 Advection equation with a source term

We consider the following advection equation:

$$\begin{cases} \dfrac{\partial u}{\partial t} + c\dfrac{\partial u}{\partial x} = u, \\ u(0,x;\mu) = \mu_1, \\ u(t,0;\mu) = \mu_2(t), \end{cases} \qquad t \in [0,T],\ x \in [0,L], \tag{6.27}$$

where $u = u(t,x;\mu) \in \mathbb{R}$ is the scalar conservative variable (e.g. heat or pressure), c is the transport velocity of the conservative variable ($f(u) = cu, J = c$, as in (6.1)) and $\mu_2(0) = \mu_1$. To be able to numerically solve this equation, an initial condition $\mu_1 \in \mathbb{R}$ (initial condition is constant over the spatial domain) and a time-varying boundary condition $\mu_2(t) \in \mathbb{R}$ are defined. In this test case, $x_\partial = 0$ as in (6.1) and we consider the varying parameters $\mu := (\mu_1, \mu_2, c, L) \in \mathcal{D} \subset \mathbb{R}^4$. The other parameters in the system are fixed.

Within this test case, the boundary condition is just a function of time, and no dynamics occur at the boundary. Therefore, the full-order model (6.5) simplifies to:

$$\Sigma_{lin} : \begin{cases} U^{n+1} = A(\mu)U^n + B(\mu)w^n, \\ y^n = C_y U^n. \end{cases} \tag{6.28}$$

To take into account the time-varying boundary conditions, the vector U_B in (6.7) should be defined. For this test case, there is only one boundary condition in the left side of the spatial domain at $x_\partial = 0$, thus:

$$U_B(U^n_{BC}) = U_B(U^n|_{x_\partial = 0}) = \mu_2(t)\mathbf{1}, \tag{6.29}$$

where $\mathbf{1}$ is a vector of ones of dimension N_δ. By doing so, the vector U_B encodes the boundary condition and therefore $\hat{U}$ as in (6.7) correctly simulates the effects of the boundary. Hence, no dynamics occur at the boundary conditions as the boundary values are specified explicitly over time. Therefore, Σ_{BC} is omitted from (6.9) and this system reduces to

$$\hat{\Sigma}_{lin} : \begin{cases} a^{n+1} = \hat{A}_a a^n + \hat{B}\hat{w}^n + \hat{A}^*_{BC}\mu_2^n - \hat{B}^*_{BC}\mu_2^{n+1}, \\ \hat{U}^n = \mathbf{1}\mu_2^n + \Phi a^n, \\ \hat{y}^n = C_y \hat{U}^n, \end{cases} \tag{6.30}$$

where $\hat{A}^*_{BC} = \Phi^T A\mathbf{1}$ and $\hat{B}^*_{BC} = \Phi^T\mathbf{1}$ are both in $\mathbb{R}^N$. Now, the dynamics in (6.30) is completely independent of N_δ and offline-online decomposition is achieved thanks to the linearity and affine property of the full-order model (6.28) (Assumption 6.2). The error dynamics then is of the form (6.18) and the ℓ_2 error bound reads as (6.19). Therefore, in Algorithm 5 for this test case, we only need to calculate $\gamma^{e_y\mathcal{R}}$ for all parameters within $\mathcal{D}_h$ (recall that the error dynamics

(6.18) is parameter dependent). In addition, by the definition of $U_B^n = \mathbf{1}\mu_2^n$, the terms in the dashed box in the residual calculation (6.17) can be computed cheaply. For instance,

$$U_B^{n\,T} A^T A\Phi = \mu_2^n \mathbf{1}^T A^T A\Phi \in \mathbb{R}^N,$$

is computed without being dependent on the size of the full-order system N_δ. Therefore, the online residual calculation does not depend on the dimension of the full-order model.

This test case can assess the quality of the ℓ_2 error bound by comparison against the ISS-like bound (6.22). After promising performance of the approximate error bound (6.19) on this test case, the more general approximate error bound (6.13) is applied to the linearized isothermal Euler equations where the effect of the local nonlinearities plays a crucial role.

6.5.1.2 Isothermal Euler equations

To mathematically describe a single-phase flow inside a pipe, the isothermal Euler equations encompassing mass and momentum conservation laws are used [66]. This system of PDEs is described as

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \dfrac{\partial}{\partial x}(\rho v) = 0, \\ \dfrac{\partial}{\partial t}(\rho v) + c_l^2 \dfrac{\partial \rho}{\partial x} = \rho g \sin\theta - 32\dfrac{\zeta}{d^2}\dfrac{\rho v}{\rho_0}, \end{cases} \tag{6.31}$$

where $\rho = \rho(t, x; \mu)$ and $v = v(t, x; \mu)$ are the density and velocity of the fluid, respectively. Also, c_l is the speed of sound in the liquid medium, ρ_0 is the reference density around which the equation of state of the fluid is linearized, g is the gravitational acceleration, θ is the inclination of the pipe with respect to the horizontal direction, ζ is the viscosity of the fluid and d is the diameter of the pipe.

To compare it with (6.1), denote $u = [\rho \;\; \rho v]^T$, $f = [\rho v \;\; \rho c_l^2]^T$ and

$$S = \begin{bmatrix} 0 & \rho g \sin\theta - 32\dfrac{\zeta}{d^2}\dfrac{\rho v}{\rho_0} \end{bmatrix}^T, \quad J = \begin{bmatrix} 0 & 1 \\ c_l^2 & 0 \end{bmatrix}. \tag{6.32}$$

Henceforth, wherever the Euler equations are mentioned, we refer to system (6.31) unless otherwise noted.

Boundary conditions are selected to enable the simulation towards industrial applications such as MPD [129] and pipeline networks [66]. In this regard, for the left boundary conditions (at $x_\partial = 0$), a liquid pump is assigned as

$$q_p = A_p v_p, \tag{6.33}$$

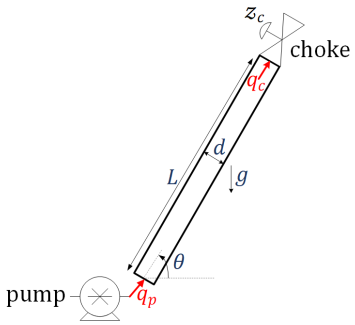

Figure 6.4: Pipe configuration for the Euler equations.

where q_p is the volumetric flow rate of the pump, A_p is the discharge area of the pump, and v_p is the liquid velocity at the pump. For the right boundary condition (at $x_\partial = L$), a choke manifold is located which is governed by the equation

$$q_c = K_c z_c \sqrt{\frac{2}{\rho_c}(p_c - p_{atm})}, \tag{6.34}$$

where $q_c, K_c, z_c, \rho_c, p_c$ and p_{atm} are the volumetric flow rate through the choke, the choke constant, the choke opening, the density at the choke inlet at the pipe side, the pressure at the choke inlet at the pipe side and the atmospheric pressure, respectively. This test case can be visualized as in Figure 6.4 with $x_\partial = \{0, L\}$.

However, to numerically solve the coupled system (6.31), the values for both ρ and ρv at each boundary are required while there is only one static relation at each boundary. For instance, at the left boundary condition, by assigning q_p, we know v at the boundary and ρ is unknown. In order to resolve this issue and render the dynamics at the boundary consistent with the internal dynamics, the characteristic-based boundary equations as in [69], [71] are unified with the static relations (6.33) and (6.34). These characteristic-based equations define G in (6.5).

In this test case, $\mu := (\rho_0, \theta, c_l, d, \nu, q, L) \in \mathcal{D} \subset \mathbb{R}^7$. Other remaining parameters, such as K_c, are fixed from one simulation to another. Finally, for numerical simulations, the initial condition of this system for any given parameter setting is the corresponding steady-state solution of the system. Then, by changing the control inputs u_c (q_p and z_c in this test case), the dynamics of the system are excited.

To incorporate boundary conditions for the linear isothermal Euler equation

(6.31), we take

$$U_B^n(\hat{U}_{BC}^n) = \left(1 - \frac{X}{L}\right)\hat{U}_{BC}^n|_{x_\partial=0} + \frac{X}{L}\hat{U}_{BC}^n|_{x_\partial=L}, \tag{6.35}$$

where $\hat{U}_{BC}^n|_{x_\partial=0}$ and $\hat{U}_{BC}^n|_{x_\partial=L}$, respectively, are the left and the right boundary values of the conservative variables (corresponding to U_{BC}^n in (6.5)) computed by the solution of the reduced-order model and $X \in \mathbb{R}^{N_\delta}$ is a vector containing the location of the grid cells. The logic behind choosing a linear function for incorporating the boundary conditions inside the domain originates from the steady-state solution of (6.31). In steady state, momentum is constant along the spatial domain, and due to the high value of the speed of sound, density varies almost linearly with respect to the spatial domain (this can be seen numerically in Section 6.5.2). In this way, the reduced basis approximation can capture the steady-state solution with a better accuracy compared to the situation where the classical ansatz (6.6) is used. Similar to the advection equation, by the definition of U_B^n, the residual (6.17) can be calculated cheaply. To include the effect of U_B^n in the reduced-order model and the detailed analysis for the error bound and estimate, refer to Appendix E.

6.5.1.3 Parameters for the numerical experiments

The numerical results are divided into two parts. In the first part, the effect of changing the ansatz is tested by comparing the use of (6.6) and (6.7) for both the advection and the Euler equations. It is shown that the proposed ansatz (6.7) with the interconnection shown in Figure 6.2 is capable of incorporating the time-varying and nonlinear boundary conditions in the advection and Euler equations while the reduced solution with the ansatz (6.6) generates discontinuity at the boundary locations. The simulation speedups in the online phase compared to the full-order solution are also reported without considering computations of the residuals, the error bound and estimate; it means that in Algorithm 6, lines 1, 6 and 7 are not included in the cpu time. In the second part, the performance of the error bound and estimate is analyzed. The error bound is compared with the error bound introduced in [84] (denoted by HO in the figures). For the advection equation, it is numerically confirmed, by comparing the results with the ISS-like bound (6.22), that the proposed error bound is one of the sharpest possible error bound on systems reaching steady state with constant inputs. This is due to the fact that the $\mathcal{H}_\infty$-norms of Σ_{lin} occur at zero frequency. Moreover, the error estimate of [182] (denoted by Dual in the figures) is compared with the proposed error estimate. In the following, the simulation parameters for both test cases are introduced.

Table 6.1: Variation range of the varying parameters in the advection equation.

parameter	c [m/s]	L [m]	μ_{BC}	μ_1
minimum	50	100	1	0
maximum	100	200	5	2
Online μ^o	85	125	4	1.5

Advection equation with the source term

The simulation parameters for the advection equation (6.27) along with the properties of the parameter domain are listed in Table 6.1, where the minimum and maximum value for each parameter are specified. The discrete parameter domain $\mathcal{D}_h$ is composed of 81 equidistant members in the parameter domain $\mathcal{D}$ (each parameter contains three equally distributed values in its region. In other words, $\mathcal{D}_h$ is a grid of 81 equidistant points in $\mathcal{D}$.). In the last row of Table 6.1, the set of parameters selected for the online simulation μ^o is reported, which does not lie in the discrete parameter domain. The number of spatial grid cells are $N_\delta = 500$, the time horizon T is 10 s and time steps are changed in a way such that at each simulation $c\Delta t/\Delta x = 0.9$ (this is the so-called CFL number determining the stability of the full-order model [69]). For the boundary conditions, we take

$$\mu_2(t) = \begin{cases} \mu_1, & 0 \leq t \leq 2, \\ \mu_{BC}, & 2 < t \leq 4, \\ 0, & 4 < t \leq 6, \\ \dfrac{\mu_{BC}}{2}, & 6 < t \leq 10. \end{cases} \tag{6.36}$$

Linearized isothermal Euler equations

For the Euler equations (6.31), the test case is shown in Figure 6.4. The first two rows of Table 6.2 indicate the range of variation of the varying parameters and the third row shows the values used in the online phase, μ^o. The discrete parameter domain $\mathcal{D}_h$ contains 128 members equally distributed in the parameter domain $\mathcal{D}$ (each parameter contains two equally distributed values in its region). Fixed parameters for the Euler equations are listed in Table 6.3.

Table 6.2: Variation range of the varying parameters in the Euler equations.

parameter	ρ_0 [kg/m^3]	θ	c_l [m/s]	d [m]	ζ [Pa.s]	q [m^3/s]	L [m]
minimum	800	60	1000	0.2	0.004	0.01	800
maximum	1200	90	1500	0.5	0.04	0.05	1200
Online μ^o	1000	72	1250	0.35	0.02	0.03	1000

The full-order model contains 500 spatial grid cells, $N_\delta = 500$, with the time horizon $T = 25$ s and time steps chosen such that at each simulation the CFL number is 0.99, i.e., $c_l \Delta t / \Delta x = 0.99$. The the boundary conditions are taken as

$$q_p(t) = \begin{cases} q & 0 \leq t \leq 3 \\ 0.2q & 3 < t \leq 25 \end{cases}, \; z_c(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0.1 & 1 < t \leq 25 \end{cases}. \tag{6.37}$$

Finally, it should be mentioned that in the simulation related to the Euler equations, the same number of basis vectors for density ρ and momentum ρv is used.

6.5.2 Illustration of the proposed ansatz formulation

In this section, the effect of the ansatz modification on the correct implementation of the boundary conditions is investigated.

Advection equation with the source term

The result of the reduced basis method with and without changing the ansatz at different time instants are compared in Figures 6.5 and 6.6. These figures compare the results of (6.27) solved by the finite-volume method (denoted by FV), the reduced basis method with the classical ansatz (6.6) (denoted by RB) and the reduced basis method of (6.27) with the proposed ansatz (6.7) (denoted by RB-ansatz) for a varying number of basis vectors, namely 10 and 30 basis vectors, respectively. The space of basis vectors in this section is not enriched with the initial conditions. As is apparent from the figures, the boundary condition (at $x_\partial = 0$) is not implemented correctly even by using 10 basis vectors with ansatz (6.6). This can be visually confirmed in the figures by the mismatch at the boundary at $x_\partial = 0$. This mismatch is due to the fact that the number of degrees of freedom to simultaneously satisfy the boundary conditions and the internal domain is not sufficient. In other words, in this case, the reduced basis compromises between the accuracy of the solution in the internal domain and the correct implementation of the boundaries. However, for the reduced basis with the ansatz (6.7), the boundary conditions are always implemented correctly, regardless of the number of basis vectors.

Moreover, the initial condition is captured more accurately with the ansatz (6.7). This means that the initial energy of the system is also captured better. It should be noted that the drawback in capturing the initial condition in the

Table 6.3: Fixed parameters for the Euler equations.

parameter	g [m/s^2]	p_{atm} [Pa]	K_c
value	9.81	10^5	0.00285

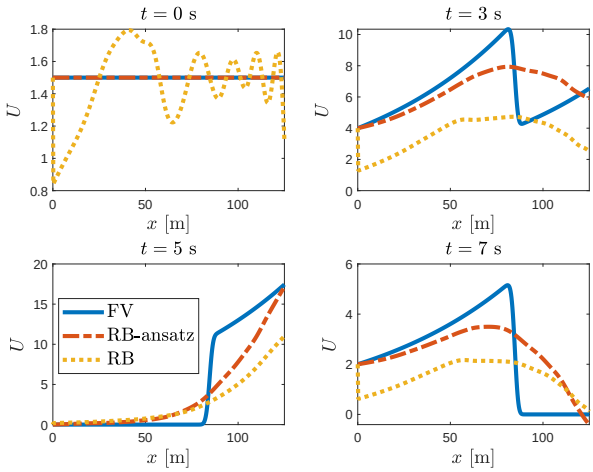

Figure 6.5: Incorporation of the boundaries by using 10 basis vectors for the advection equation; comparison of the finite-volume (FV) solution and the reduced basis method with (6.7) (RB-ansatz) and (6.6) (RB).

classical ansatz (6.6) can be resolved by enriching the space of the basis vectors with the initial condition.

To compare the speedups in the online phase and the full-order solution, see Table 6.4. As mentioned previously, computations of the error bound (estimate) are not included in the speedup calculation. In the online phase, the state for the entire domain is calculated to compare the solutions in the entire domain, which scales with the degrees of freedom of the original model and decreases the speedups. The other reason for the moderate speedups is the 1D type of problems considered in this paper, where the expected speedups are modest. The hyperbolicity of the advection equation can also be noted as another reason for moderate speedups.

Linearized isothermal Euler equations

Analogously, the result of the reduced basis method with the ansatz (6.6) and (6.7) at different time instants are compared in Figures 6.7 and 6.8 for the

Table 6.4: Speedup factors for the reduced basis method for the advection equation.

N	1	10	20	30	40	50
Speedup	16.8	8.2	7.9	6.2	5.5	5.3

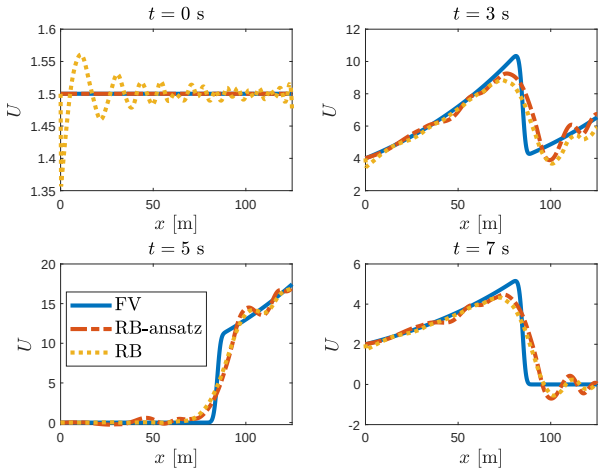

Figure 6.6: Incorporation of the boundaries by using 30 basis vectors for the advection equation; comparison of the finite-volume (FV) solution and the reduced basis method with (6.7) (RB-ansatz) and (6.6) (RB).

linearized isothermal Euler equations. The velocity of the liquid v is depicted in these figures when using 10 and 30 basis vectors.

Similar to the advection equation case study, the boundary condition treatment is not perfect with the classical ansatz (6.6) as mismatches at the two boundaries (at $x_\partial = 0$ and $x_\partial = L$) are present. Similar to the result for the advection equation, initial conditions, initial mass and initial energy of the system are captured better with the ansatz (6.7) compared to the ansatz (6.6).

To compare the speedups, refer to Table 6.5.

The relatively modest speedup factors can be attributed to the fact that, at each time step, a nonlinear function should be solved to compute the boundary conditions both for the full-order and the reduced-order model, which dominates the cpu time of both simulations. Similar to the advection equation, the full state is reconstructed in the online phase to compare the solutions in the entire domain, whose computation scales with the degrees of freedom of the original model and reduces the speedup. Hyperlic nature of the Euler equations is another reason of moderate speedup values.

Table 6.5: Speedup factors for the reduced basis method for the Euler equations.

N	1	10	20	30	40	50
Speedup	18.8	11.4	11.2	8.3	7.7	7.5

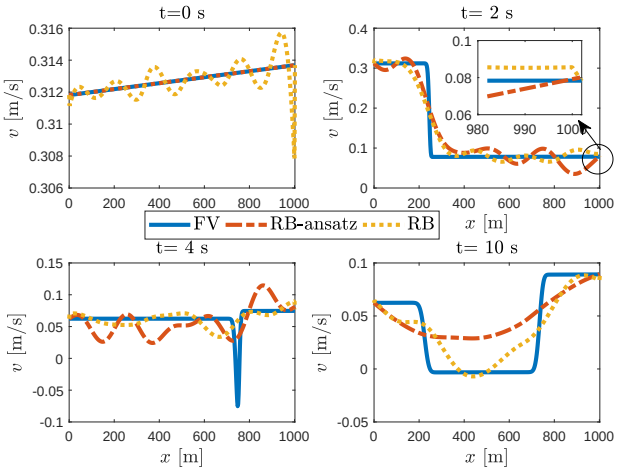

Figure 6.7: Incorporation of the boundaries by using 10 basis vectors for the Euler equations; comparison of the finite-volume (FV) solution and the reduced basis method with (6.7) (RB-ansatz) and (6.6) (RB).

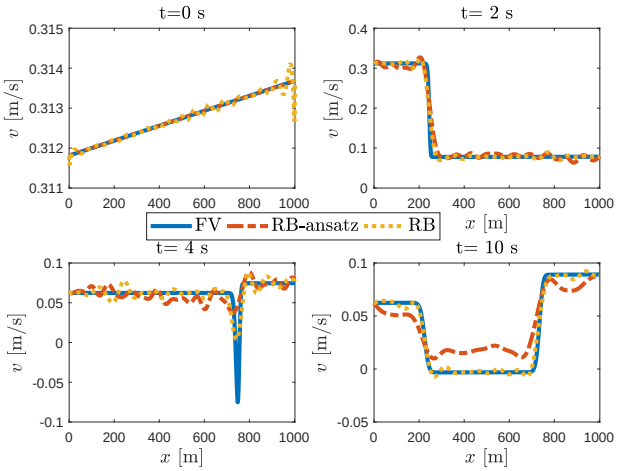

Figure 6.8: Incorporation of the boundaries by using 30 basis vectors for the Euler equations; comparison of the finite-volume (FV) solution and the reduced basis method with (6.7) (RB-ansatz) and (6.6) (RB).

All in all, the ansatz (6.7) captures the effect of the boundary conditions

better than the classical one (6.6), leading to a correct incorporation of the control inputs at the boundaries.

6.5.3 Performance of the error bound and estimate

In this section, the proposed error bound and estimate are compared with the actual error and also existing error bounds [84] and estimates [182] in the ℓ_2-norm. First, the results are presented for the advection equation with source term and, second, for the linearized isothermal Euler equations. The variables that are shown in the figures of this section are computed as

$$\mathcal{E}^n_{act} = \frac{1}{\sqrt{n}} \sqrt{\sum_{i=1}^{n} \left\| e_y^i \right\|^2}, \tag{6.38a}$$

$$\mathcal{E}^n_{\mathcal{H}} = \frac{1}{\sqrt{n}} \left(\gamma^{e_y e_w} \sqrt{\sum_{i=1}^{n} \left\| e_w^i \right\|^2} + \gamma^{e_y \mathcal{R}} \sqrt{\sum_{i=1}^{i=n} \left(\mathcal{R}_s^i \right)^2} \right), \tag{6.38b}$$

$$\mathcal{E}^n_{ISS} = \frac{1}{\sqrt{n}} \sqrt{\sum_{i=1}^{n} (\kappa^i_{ISS})^2}, \tag{6.38c}$$

where $\mathcal{E}_{act}$, $\mathcal{E}_{\mathcal{H}}$ and $\mathcal{E}_{ISS}$ are the indicators used for the actual error, the proposed error bound and ISS-like error bound. For the error bound and estimate introduced in [84], [182] (HO and Dual in the following figures, respectively), a similar approach to (6.38c) is followed to compute the variables shown in the figures of this section; instead of κ_{ISS}, the bound in those papers are used.

Advection equation with source term

In order to make a fair comparison between the proposed error bound in (6.19), which is a bound on the output error e_y, and the ISS-like bound in (6.22), which is a bound on the state error e, all states are considered as the output, i.e., $C_y = I_{N_\delta \times N_\delta}$ where I is the identity matrix. To have a zero error at the initial time and justify using the ℓ_2-gain notion (6.10), the space of basis vectors is enriched by the modified initial conditions, which is computed similarly to (6.8) with $t = 0$. The error bound comparisons are depicted in Figure 6.9. Obviously, the ISS-like bound is always the sharpest bound. On the one hand, the computation of the ISS-like bound is expensive due to the computation of the norms in (6.22) at each time step. On the other hand, the bound (6.19) is cheaper to compute as it calculates the $\mathcal{H}_\infty$-norms only once and also it is closer to the ISS-like bound compared to the HO bound after the initial transient. The proposed bound, unlike the HO bound, remains bounded for increasing time. Therefore, the proposed bound in this paper is one of the sharpest possible

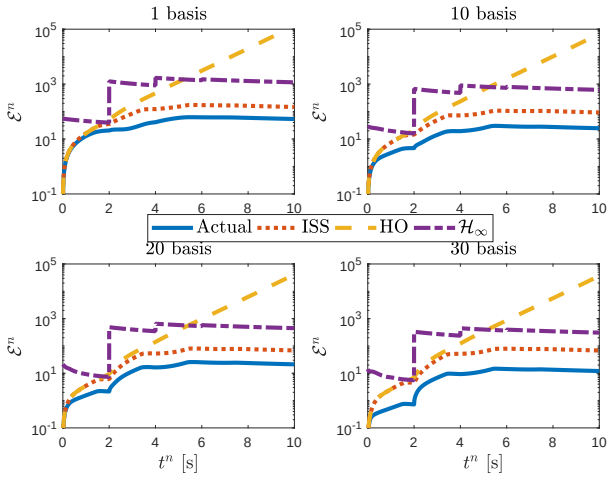

Figure 6.9: Comparison of the error bounds, $\mathcal{E}^n_{act}$ ('Actual'), $\mathcal{E}_{\mathcal{H}}$ ('$\mathcal{H}_\infty$'), $\mathcal{E}_{ISS}$ ('ISS') and 'HO', for the online parameter set; advection equation.

error bounds for steady-state certification. The jumps visible in Figures 6.9 and 6.10 in the error bounds and estimates are due to the fact that at these time instants the boundary values change drastically and therefore the residual changes significantly.

The comparison of the error *estimates* is depicted in Figure 6.10, revealing that the estimate (6.23) is very close to the actual error by selecting either option regarding $\bar{\rho}$. In this figure, the option of $\bar{\rho}$ as in (6.25) is shown by '$\mathcal{H}_\infty$-last' and $\bar{\rho}$ as in (6.26) is shown by '$\mathcal{H}_\infty$'. Clearly, both error estimates proposed here outperform the one proposed in [182] in the ℓ_2-norm realization. Moreover, '$\mathcal{H}_\infty$-last' provides a sharper estimate than '$\mathcal{H}_\infty$' (as expected). However, it was observed that sometimes the estimate '$\mathcal{H}_\infty$-last' underestimates the actual error. Nevertheless, it is always close to the actual error within a narrow band. For $\bar{\rho}$ as in (6.26), the error estimate never underestimated the actual error. It should be noted that the dual error introduced in [182] may perform better in other norms.

The dependency of the actual error and the error bounds and estimates as a function of the number of basis vectors is shown in Figure 6.11, where 'ℓ_2' is the bound computed by (6.38b) at $n = N_t$. This figure compares the maximum estimated error among the members in the discrete parameter domain during the greedy algorithm. The proposed error estimate with $\bar{\rho}$ as in (6.25) always approximates the maximum error in the parameter domain with high accuracy while the performance of the error estimate with $\bar{\rho}$ as in (6.26) and the error estimate in [182] are more conservative. Table 6.6 reports on the percentage

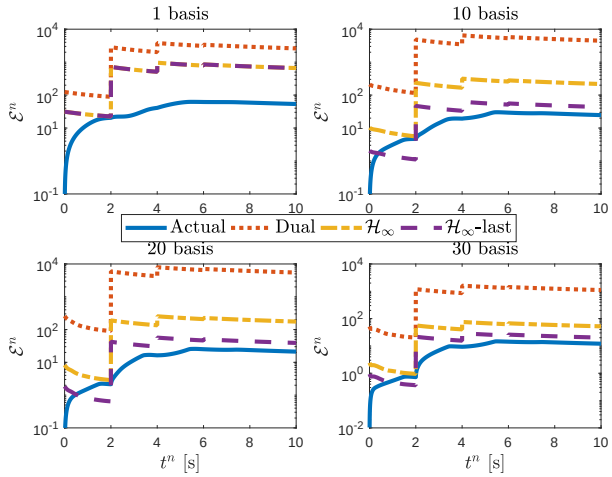

Figure 6.10: Comparison of the error estimates, $\mathcal{E}^n_{act}$ ('Actual'), 'Dual', $\bar{\rho} \times \mathcal{E}_{\mathcal{H}}$ ($\bar{\rho}$ as in (6.26) denoted by '$\mathcal{H}_\infty$' and as in (6.25) denoted by '$\mathcal{H}_\infty$-last' for the online parameter set; advection equation.

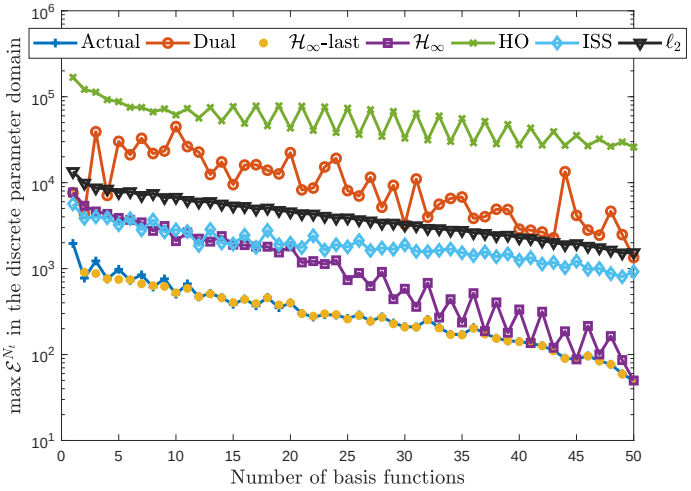

Figure 6.11: Comparison of the maximum of error bounds and estimates computed for all members of the parameter domain during the greedy algorithm; advection equation.

of the online calculation devoted to the calculation of the error estimate when varying the number of basis vectors. This table numerically confirms that the computational cost required for the online error bound computation is slightly larger than the one for the computation of the reduced-order solution due to the residual calculation. However, as shown before, the computations related to the residual calculation are independent of the dimension of the full-order model N_δ.

Linearized isothermal Euler equations

For this test case, as we can only approximate the Lipschitz constant of the nonlinear functions, no error bound is introduced. The detailed analysis of the error estimation computation can be found in Appendix E. Here, the pressure at the last grid cell, the grid cell before the choke manifold, is selected as the output. The comparison of the error estimates with the actual error is presented in Figure 6.12. Moreover, Figure 6.13 compares the maximum estimated error among the members of the discrete parameter domain during the greedy algorithm. Similar to the advection equation, the error indicator in Algorithm 5 is controlled by different approaches proposed and introduced in this paper. The proposed error estimate in this paper works effectively and always predicts the error with a good accuracy. Compared to the error estimate in [182], the estimation of the actual error has improved significantly in the ℓ_2-norm. Although this estimate is very effective, its applicability region is limited, which has been discussed at the end of Appendix E of this paper. The percentage of the CPU time allocated to calculate the error estimate in the online phase by varying N is listed in Table 6.7. Similar to the advection equation, calculation of the residual takes a significant percentage of the CPU time. The high ratio at the very low number of basis vectors relates to the very fast solution of the reduced-order model while the computation of the Lipschitz constants do not scale with the number of basis vectors.

Table 6.6: Percentage of the CPU time allocated to the computation of the error estimate during the online phase of the reduced-order model, the advection equation.

N	1	10	20	30	40	50
ratio	44%	75%	74%	72%	71%	70%

Table 6.7: Percentage of the CPU time allocated to the computation of the error estimate during the online phase of the reduced-order model, the Euler equations.

N	1	10	20	30	40	50
ratio	82%	71%	70%	70%	69%	69%

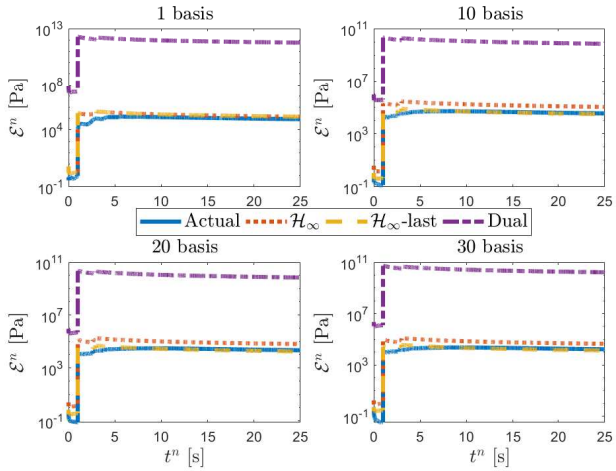

Figure 6.12: Comparison of the error estimates, $\mathcal{E}^n_{act}$ ('Actual'), 'Dual', $\bar{\rho} \times \mathcal{E}_{\mathcal{H}}$ ($\bar{\rho}$ as in (6.26) denoted by '$\mathcal{H}_\infty$' and as in (6.25) denoted by '$\mathcal{H}_\infty$-last' for the online parameter set; Euler equations.

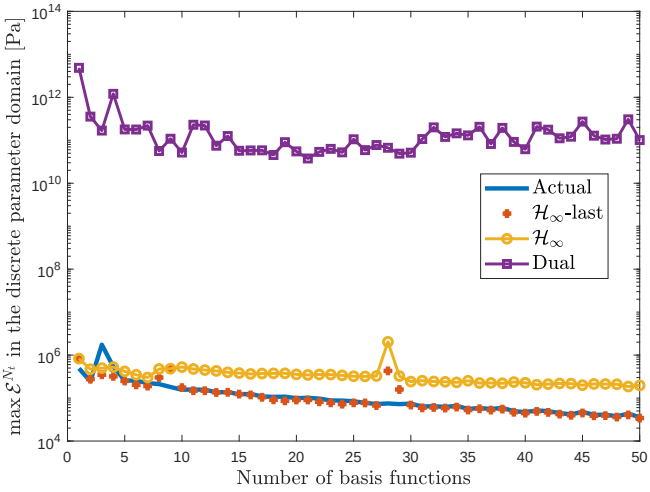

Figure 6.13: Comparison of the maximum of error estimates computed for all members of the parameter domain during the greedy algorithm; Euler equations.

6.5.4 Discussion

To conclude, as apparent from the results presented in this section, the ansatz (6.7) is capable of incorporating the boundary conditions effectively, both for the advection and the Euler equations. In addition, the proposed error bound (6.19) is very close to the ISS-like bound (6.22) in the advection equation and significantly computationally more attractive than the ISS-like bound (6.22). Finally, the computationally efficient error estimate (6.23) is sharp and able to predict the actual error with a reasonable accuracy.

6.6 Conclusion and future work

In this paper, the reduced basis method is applied to systems with time-varying and nonlinear boundary conditions. By changing the ansatz of the reduced basis, these types of boundary conditions are handled properly while mismatches and discontinuities are observed at the boundaries with the classical ansatz.

Moreover, a new residual-based error bound and estimate are proposed. Given the accessible information in the reduced basis context (the spatial two-norm of the residual at each time instant), this error bound is sharp in steady-state for a special class of systems; systems for which inputs are constant. To mitigate the conservativeness of the error bound, an accurate error estimate is introduced. The error estimate outperforms the existing error estimates in ℓ_2-norm realization.

For future work, the application of the approach on real applications such as a managed pressure drilling systems will be investigated. The interconnection approach of handling local nonlinearities at the boundaries will be extended to local nonlinearities inside the domain. The extension of the error bound and estimate for nonlinear systems is also being investigated.

Chapter 7

Error estimates for model order reduction of Burgers' equation

Burgers' equation is a nonlinear scalar partial differential equation, commonly used as a testbed for model order reduction techniques and error estimates. Model order reduction of the parameterized Burgers' equation is commonly done by using the reduced basis method. In this method, an error estimate plays a crucial rule in both accelerating the offline phase and quantifying the error induced after reduction in the online phase. In this study, we introduce two new estimates for this reduction error. The first error estimate is based on a Lur'e-type model formulation of the system obtained after the full-discretization of Burgers' equation. The second error estimate is built upon snapshots generated in the offline phase of the reduced basis method. The second error estimate is applicable to a wider range of systems compared to the first error estimate. Results reveal that when conditions for the error estimates are satisfied, the error estimates are accurate and work efficiently in terms of computational effort.

7.1 Introduction

Model order reduction of high-fidelity models is a necessary tool for enabling real-time simulation and controller design. These high-fidelity models are often the result of the discretization of Partial Differential Equations (PDEs) governing the physical phenomena. One way to reduce these models is the Reduced Basis (RB) method [84], consisting of decomposed offline and online phases. In the

This chapter is based on "M.H. Abbasi, L.Iapichino, B. Besselink, W.H.A.Schilders, N.van de Wouw, ***Error estimates for model order reduction of Burgers' equation***, *IFAC World Congress*, Berlin, July 2020".

offline phase of the RB method, RB functions for approximating the solution are generated. This phase contains computations whose complexity scale with the degrees of freedom of the original system, thus it is computationally expensive. In the online phase, the solution is approximated by a linear combination of the RB functions. The computations in this phase scale with the number of RB functions generated in the offline phase, which renders obtaining the solution of the reduced model computationally efficient. However, replacing a model with its reduced version leads to an error between the solution of the full-order model and the reduced one. To ensure the accuracy of the reduced solution, an error bound or estimate should be provided. In the RB context, the benefits of having such an error bound or estimate are twofold. First, an error bound (or estimate) in the RB technique can be used to accelerate the offline phase during the greedy algorithm [9]. Second, it certifies the accuracy of the solution that is obtained during the online phase. Therefore, developing a sharp error bound (or an accurate error estimate) is crucial within this approach.

To build an efficient yet accurate reduced-order model by the RB method and decompose the offline and online phases, nonlinear problems are hyper-reduced by using the Empirical Interpolation Method (EIM) [17] or its discrete counterpart, the Discrete Empirical Interpolation Method (DEIM) [44], combined afterwards with the RB method itself. EIM and DEIM require additional basis functions (called collateral basis functions) to approximate the nonlinear functions and these collateral basis functions are usually generated in the offline phase before the generation of the RB functions, which makes the offline phase even more expensive. To reduce the computation time, the collateral basis functions can be generated in parallel to the RB functions. To synchronize the RB function generation and the collateral basis function generation, various algorithms have been introduced; e.g. the PODEI algorithm by [63]. The inaccurate approximation of the nonlinear functions also plays a role in the final error induced by reduction, which has to be taken into account when building error estimates. To generate both collateral basis functions and RB functions, the solution snapshots of the full-order system of equations should be available.

In this paper, we focus on a hyperbolic PDE, Burgers' equation. Hyperbolic systems are commonly solved by Finite-Volume (FV) techniques that lead to state-space models of high order. The work on error bounds (or estimates) in the RB community for hyperbolic systems is still in the evolutionary stage, see [9], [84], [182] for some works. Methods introduced in these works are typically tailored to linear systems and not efficient if applied to nonlinear systems. Moreover, most of these techniques (except the method by [9]) utilize the norm of the state matrix of the discretized system. If the state matrix has a large norm (larger than one), these error bounds (estimates) are not valid and grow exponentially over time. The method introduced by [9] (which also works well if applied to systems with local nonlinearities) circumvents this issue by using the ℓ_2-norm of the system with respect to its inputs and outputs, as similarly

done in the balanced truncation method by [28], [29], [129]. In general, theoretical error estimates for nonlinear systems are lacking in the RB literature. In this paper, we aim to extend the methodology introduced in [9] from systems with local nonlinearities to systems with distributed nonlinearities. However, the error estimate of [9] cannot be efficiently used when strong nonlinearities (nonlinearities with high Lipschitz constant) are present in the system.

Therefore, in addition to the error estimate based on the ℓ_2-gain notion, an empirical error estimate is also introduced in this paper. This estimate is based on the snapshots generated in the offline phase of the RB method. This estimate does not suffer from restrictions of the previous error estimate. Most importantly, it does not require the residual calculation and it is tailored in a way that its computation is efficient, similar to the computation of the reduced-order solution.

The structure of this paper is as follows. In Section 7.2, Burgers' equation together with its discretization, which leads to the full-order model, is introduced. In Section 7.3, the model-order reduction approach used to obtain the reduced-order model is elaborated. In Section 7.4, the two error estimates for the nonlinear reduced-order model are discussed. In Section 7.5, numerical results are presented. Finally, Section 7.6 concludes the paper.

7.2 Burgers' equation

One of the simplest and yet fundamental nonlinear equations describing a conservative system is Burgers' equation, which is sometimes referred to as the scalar version of the Navier-Stokes equations ([136]). This equation is defined as

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(f(u)\right) = 0,\ t \in [0,T],\ x \in [0,L], \tag{7.1}$$

where $u := u(t,x;\boldsymbol{\mu})$ is the conservative variable and $f(u) = u^2/2$ is the flux function associated with Burgers' equation. Here, t represents time and T is the time horizon of the simulation. In addition, x denotes the spatial coordinate and L is the length of the spatial domain. Finally, $\boldsymbol{\mu} \in \mathcal{D}$ is a vector of parameters used in (7.1) that varies in a multi-query analysis within the parameter domain $\mathcal{D} \in \mathbb{R}^R$, with R the number of varying parameters. We assume that the initial condition and boundary condition are represented by these varying parameters. For the initial condition, we assume $u(0,x;\boldsymbol{\mu}) = \mu_1$, which is constant over the spatial domain. For the boundary condition at $x = 0$, we assume

$$u(t,0;\boldsymbol{\mu}) = \begin{cases} \mu_1, & t = 0, \\ \mu_2, & t > 0. \end{cases} \tag{7.2}$$

Therefore, in this study, we have $\boldsymbol{\mu} = [\mu_1, \mu_2]$.

Discretizing (7.1) with the Lax-Friedrichs scheme (see [73], [110]) leads to

$$\boldsymbol{U}^{n+1} = L_{lin}\boldsymbol{U}^n + BU_0^n - \frac{\Delta t}{4\Delta x}L_{nl}(\boldsymbol{U}^n)^2 + \frac{\Delta t}{2\Delta x}B(U_0^n)^2, \tag{7.3}$$

where $\boldsymbol{U}^n := [U_1^n, \cdots, U_{\mathcal{N}}^n]^T \in \mathbb{R}^{\mathcal{N}}$ is the vector containing U_i^n, the average of the conservative variable u over the i-th grid cell at the time instant $t^n := n\Delta t, n = \{0, \cdots, N_t\}$ with N_t number of time steps. Here, Δt and Δx refer to the temporal and spatial discretization intervals over time and space, respectively. The spatial discretization consists of cells $(x_{i-1/2}, x_{i+1/2}), i = 1, \cdots, \mathcal{N}$, with the length of Δx centered at $x_i = x_{i-1/2} + \Delta x/2$ and $\mathcal{N}$ spatial grid cells. Furthermore, $L_{lin}, L_{nl} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ are the operators acting on the linear and nonlinear part of the system that emerge after applying the full-discretization. Also, $U_0^n \in \mathbb{R}$ is the value of the conservative variable at the boundary $x = 0$ acting as the input into the system defined according to (7.2) and $B \in \mathbb{R}^{\mathcal{N}}$ is the input matrix corresponding to the boundary input. Moreover, the square operator $(\cdot)^2$ in (7.3) is interpreted element-wise. The nonlinearity associated with this equation is $g(U) = (U)^2$, where $g(\cdot)$ is a nonlinear operator. Then, system (7.3) is equivalent to the system Σ depicted in the left side of Figure 7.1, which comprises a linear subsystem Σ_{lin} and nonlinear subsystem Σ_{nl} given by

$$\begin{aligned}
\Sigma_{lin}: &\begin{cases}
\boldsymbol{U}^{n+1} = L_{lin}\boldsymbol{U}^n + BU_0^n - \dfrac{\Delta t}{4\Delta x}L_{nl}\boldsymbol{U}_{nl}^n \\
\qquad\qquad + \dfrac{\Delta t}{2\Delta x}B(U_0^n)^2, \\
y^n = C_y\boldsymbol{U}^n, \\
z^n = \boldsymbol{U}^n,
\end{cases} \\
\Sigma_{nl}: &\ \boldsymbol{U}_{nl}^n = g(z^n) = (z^n)^2.
\end{aligned} \tag{7.4}$$

Here, $y \in \mathbb{R}^w$ is the output of interest of the system (for instance, y can be the value of the conservative variable at the right-end of the spatial domain with $w = 1$) and $C_y \in \mathbb{R}^{w\times\mathcal{N}}$ is the corresponding output matrix. This full-order model has large dimension (i.e., $\mathcal{N}$ is large). Therefore, real-time simulations cannot be achieved unless powerful computational resources are at the disposal. Moreover, control design for such a complex system is generally infeasible. Hence, model order reduction should be applied to (7.4), which is the topic of the next section. The following assumption will be used throughout the paper.

Assumption 7.1. *The system matrix L_{lin} is Schur for all $\boldsymbol{\mu} \in \mathcal{D}$, i.e., Σ_{lin} in (7.4) is internally asymptotically stable.*

7.3 Model reduction

This section subsequently discusses the RB method, (D)EIM, and their combination, leading to a method for hyper-reduction of the nonlinear system (7.4).

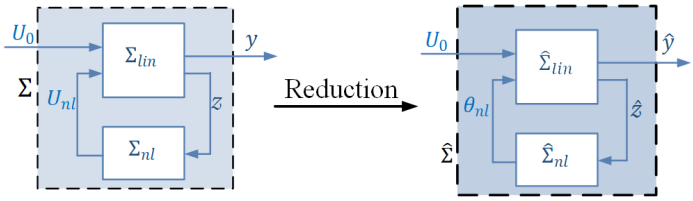

Figure 7.1: Left: The schematic representation of the the interconnection between the linear dynamics and the static nonlinearity. Right: The schematic representation of the interconnection between the reduced linear subsystem (obtained by the RB method) and the reduced nonlinear function (number of nonlinear equations is reduced by (D)EIM).

7.3.1 Reduced basis method

A powerful method for dimension reduction of a parameter-dependent dynamical system is the RB method. In the RB method, the system of equations is projected into a low dimensional space spanned by the solutions of the full-order model for specific members of the parameter domain.

As discussed in [9], handling time-varying boundary conditions within the RB method is vital as the (time-varying) control inputs commonly act at the boundaries. Tailoring the method in [9] to our case study, we introduce the RB ansatz

$$\hat{\boldsymbol{U}}^n(\boldsymbol{\mu}) = U_0^n(\boldsymbol{\mu})\mathbf{1} + \Phi a^n, \tag{7.5}$$

where $\hat{\boldsymbol{U}}^n \in \mathbb{R}^{\mathcal{N}}$ is the solution of the reduced-order model, and $\mathbf{1} \in \mathbb{R}^{\mathcal{N}}$ is a vector of ones that enables the RB solution $\hat{\boldsymbol{U}}^n(\boldsymbol{\mu})$ to satisfy the boundary condition (7.2) at all time instants. Then, the RB functions $\Phi \in \mathbb{R}^{\mathcal{N}\times N}$ should vanish at the location of the specified boundary condition (i.e., $\Phi|_{x=0} = 0$,), where N is the number of RB functions. Here, $a^n \in \mathbb{R}^N$ is the modal coordinate associated with the RB functions, which is the state of the reduced-order model. To generate the RB functions Φ vanishing at the location of the specified boundary, we modify the snapshots during the greedy algorithm for a selected parameter $\boldsymbol{\mu}^*$ and then apply the Proper Orthogonal Decomposition (POD) ([89]) on these modified snapshots (see Algorithm 7), defined as

$$\begin{aligned} \hat{\boldsymbol{U}}^{n,*}(\boldsymbol{\mu}^*) &= \hat{\boldsymbol{U}}^n(\boldsymbol{\mu}^*) - U_0^n(\boldsymbol{\mu}^*)\mathbf{1}, \\ \hat{\boldsymbol{U}}^*(\boldsymbol{\mu}^*) &= \{\hat{\boldsymbol{U}}^{n,*}(\boldsymbol{\mu}^*)\}, \quad \forall\, n = \{0, \cdots, N_t\}. \end{aligned} \tag{7.6}$$

Finally, "POD($\hat{\boldsymbol{U}}^*(\boldsymbol{\mu}^*), 1$)" obtained from Algorithm 7 yields an RB function. For more details, we refer to [9].

7.3.2 Empirical interpolation method

To handle the nonlinearities in (7.4), EIM is applied as in [17]. By using this method, a nonlinear function is replaced by a linear interpolation of collateral basis functions (basis functions generated by the EIM/DEIM), which are obtained during the offline phase. In the online phase, the coefficients for the linear interpolation of the collateral basis functions are chosen such that this interpolation becomes exact at some pre-selected points, the so-called interpolation points, along the spatial domain. The effect of the nonlinear function is then fed back into the linear system via the feedback interconnection as shown in the right side of Figure 7.1.

After applying EIM, the nonlinear function in (7.4) is approximated by a linear interpolation

$$(\boldsymbol{U}^n)^2 \approx q_{nl}\theta_{nl}^n, \tag{7.7}$$

where $q_{nl} \in \mathbb{R}^{\mathcal{N}\times M}$ is the matrix of collateral basis functions and $\theta_{nl} \in \mathbb{R}^M$ are the unknown coefficients of the collateral basis functions, to be calculated online. The collateral basis functions q_{nl} are obtained by applying POD (Algorithm 7) on the snapshots of the nonlinearities $g(z_n)$ for specific members of the parameter domain during the offline phase. The coefficients θ_{nl} in (7.7) are obtained during the online phase such that the interpolation is exact at M pre-selected points $\boldsymbol{X_m} = \{x_1, \cdots x_M\}$ where $x_i \in \mathbb{R}$ is the grid-cell number of the interpolation point selected at the i-th iteration (the selection procedure of such points is introduced later in Algorithm 8). Specifically, let $P = [e_{x_1}, \cdots e_{x_M}] \in \mathbb{R}^{\mathcal{N}\times M}$ where e_i is the i-th column of the identity matrix (of dimension $\mathcal{N} \times \mathcal{N}$). For the points $\boldsymbol{X_m}$, we have

$$(P^T\boldsymbol{U}^n)^2 = P^T q_{nl}\theta_{nl}^n, \tag{7.8}$$

stating that the interpolation is exact at X_m if $P^T q_{nl}$ is non-singular (θ_{nl}^n can then be computed from (7.8)). After approximating the nonlinearities in (7.4) with linear interpolation of the collateral basis functions, we can apply a Galerkin projection ([84]) to the system of equations, as explained in the next section.

Algorithm 7: POD algorithm, POD($\boldsymbol{U}, n_{\text{POD}}$)

Input: Snapshots $\boldsymbol{U}(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}\times N_t}$, number of basis vectors n_{POD}
Output: $\phi \in \mathbb{R}^{\mathcal{N}\times n_{\text{POD}}}$

1 Perform a Singular Value Decomposition on the snapshots, $\boldsymbol{U} = U_{\text{SVD}}SV$
2 $\phi = U_{\text{SVD}}(:, 1 : n_{\text{POD}})$ is the first n_{POD} vectors of the left singular vectors U_{SVD}.

7.3.3 RB-EIM combination

After applying EIM to the nonlinear parts of the dynamics, all the operators involved in the full-order model become linear and therefore the system can be efficiently projected onto a lower-dimensional subspace spanned by Φ. Substituting the ansatz (7.5) and the EIM approximation (7.7) in (7.4), applying a Galerkin projection on the resulting system and taking into account the orthogonality of the basis functions Φ, we obtain the reduced-order model

$$\begin{aligned}
\hat{\Sigma}_{lin}: &\begin{cases} a^{n+1} = \hat{L}_{lin}a^n + (\hat{B}+\hat{L}_{BC})U_0^n - \dfrac{\Delta t}{4\Delta x}\hat{L}_{nl}\theta_{nl}^n \\ \qquad\quad + \dfrac{\Delta t}{2\Delta x}\hat{B}(U_0^n)^2 - \Phi_{\mathbf{1}}U_0^{n+1}, \\ \hat{y}^n = U_0^n C_y \mathbf{1} + C_y \Phi a^n, \\ z_m^n = U_0^n P^T \mathbf{1} + P^T \Phi a^n, \end{cases} \\
\hat{\Sigma}_{nl}: &\begin{cases} \boldsymbol{U}_{nl}^n = g(z_m^n) = (z_m^n)^2, \\ \theta_{nl}^n = (P^T q_{nl})^{-1}\boldsymbol{U}_{nl}^n, \end{cases}
\end{aligned} \tag{7.9}$$

where $\hat{y}$ is an approximation of y and $\hat{L}_{lin} = \Phi^T L_{lin}\Phi$, $\hat{L}_{nl} = \Phi^T L_{lin} q_{nl}$, $\hat{B} = \Phi^T B$, $\hat{L}_{BC} = \Phi^T L_{lin}\mathbf{1}$, and $\Phi_{\mathbf{1}} = \Phi^T\mathbf{1}$. Finally, z_m^n is the value of the reduced solution at the pre-selected points $\boldsymbol{X_m}$. None of the computations in (7.9) scales with the actual degrees of freedom $\mathcal{N}$ and therefore the model is in a reduced form. It should be noted that the boundary-related terms in (7.9) (such as $U_0^n P^T\mathbf{1}$) are due to the ansatz used in (7.5) and the segregation of the boundary condition from the solution. In this work, to synchronize the generation of the RB functions Φ and the collateral basis functions q_{nl}, the PODEI algorithm in [63] is used, which is mentioned in Algorithm 8 together with the greedy algorithm and the selection of the interpolation points. Now, the reduced-order model is available and the error estimates can be introduced.

7.4 Error estimates

In this section, we introduce two types of error estimates. In the first one, we build the error dynamics and propose an estimate based on the ℓ_2-gain notion. In the second one, we use the solutions of the full-order model generated in the offline phase to obtain an empirical error estimate.

7.4.1 Error estimate based on the ℓ_2-gain notion

As shown in (7.9), the interconnection of the RB method and EIM can be represented as a Lur'e-type system as shown in the right side of Figure 7.1. The error estimate introduced here relies on the notion of small-gain condition of the error dynamics ([28]), to be introduced here. If this condition is not satisfied,

Algorithm 8: PODEI-Greedy algorithm

Input: $\mathcal{D}_h$(discretized version of$\mathcal{D}$), $N, \boldsymbol{\mu}^1 \in \mathcal{D}_h$
Output: $\Phi, q_{nl}, \boldsymbol{X_m}, B_{nl}$

Set $\Phi = \{\}, q_{nl} = \{\}, \boldsymbol{X_m} = \{\}, B_{nl} = 1,$
for *k=1 to N-1* **do**
  Solve (7.4) for $\boldsymbol{\mu}^k$ to obtain $\boldsymbol{U} = [\boldsymbol{U}^0, \cdots, \boldsymbol{U}^{N_t}]$ and $\boldsymbol{U}_{nl} = [\boldsymbol{U}_{nl}^0, \cdots, \boldsymbol{U}_{nl}^{N_t}]$
  Generate $\boldsymbol{U}^* = \boldsymbol{U} - [U_0^0, \cdots, U_0^{N_t}]\mathbf{1}$ and $\bar{\boldsymbol{U}}_{nl} = \boldsymbol{U}_{nl} - q_{nl} B_{nl}^{-1} \boldsymbol{U}_{nl}(\boldsymbol{X_m}, :)$
  Set $\bar{\boldsymbol{U}} = \boldsymbol{U}^* - \Phi\Phi^T \boldsymbol{U}^*$
  $\Phi \leftarrow \text{orth}\left\{\Phi \cup \text{POD}(\bar{\boldsymbol{U}}, 1)\right\}$ and $q_{POD} = \text{POD}(\bar{\boldsymbol{U}}_{nl}, 1)$
  $\sigma_M = \left(q_{nl}(\boldsymbol{X_m}, :)\right)^{-1} q_{POD}(\boldsymbol{X_m})$
  $r_d = q_{POD} - q_{nl}\sigma_M$
  $\boldsymbol{X_m} \leftarrow \{\boldsymbol{X_m} \cup \arg\max(r_d)\},$
  $q_{nl} \leftarrow \{q_{nl} \cup \frac{r_d}{\max r_d}\}, B_{nl} = q_{nl}(\boldsymbol{X_m}, :)$
  Based on Φ and q_{nl}, perform the error estimates to find the worst approximated solution and find $\boldsymbol{\mu}^{k+1}$ and $e(\boldsymbol{\mu}^{k+1})$
  if $e(\boldsymbol{\mu}^{k+1}) > e(\boldsymbol{\mu}^k)$ **then**
    $q_{nl} = q_{nl}(:, 1 : \text{end} - 1)$, $\boldsymbol{X_m} = \boldsymbol{X_m}(1 : \text{end} - 1)$, $B_{nl} = q_{nl}(\boldsymbol{X_m}, :)$

the error estimate presented here cannot be used. To enable cheap computation of the residual, the following assumption is used.

Assumption 7.2 ([63])**.** *We assume the exactness of the EIM approximation for a certain number of collateral RB functions; i.e., there exists a positive integer $M^* > M$ with the set of enriched collateral basis functions by q_{nl}^* and the corresponding coefficients by ${\theta_{nl}^*}^n$, such that*

$$(\hat{\boldsymbol{U}}^n(\mu))^2 = q_{nl}^* {\theta_{nl}^*}^n(\mu) \quad \forall n = 1, \cdots, N_t, \quad \textit{and} \quad \mu \in \mathcal{D}. \tag{7.10}$$

Claim 7.3. *Let $\boldsymbol{U}^n$ be obtained from* (7.4) *and $\hat{\boldsymbol{U}}^n$ be obtained from* (7.9) *and* (7.5) *with $n = 1, \cdots, N_t$ under the same initial condition and the same boundary input U_0^n. We define the residual $\mathcal{R}^n$ by inserting the RB solution $\hat{\boldsymbol{U}}^n$ into (7.4) as follows:*

$$\begin{aligned}\mathcal{R}^n =& \hat{\boldsymbol{U}}^{n+1} - \Big(L_{lin}\hat{\boldsymbol{U}}^n + BU_0^n - \frac{\Delta t}{4\Delta x} L_{nl} g(\hat{\boldsymbol{U}}^n) \\ &+ \frac{\Delta t}{2\Delta x} B(U_0^n)^2\Big).\end{aligned} \tag{7.11}$$

We assume the Lipschitz continuity L_g for the nonlinear function $\boldsymbol{e}_g^n = g(\boldsymbol{e}^n)$, i.e., $\left\|\boldsymbol{e_g^n}\right\| \leq L_g \left\|\boldsymbol{e^n}\right\|$. An estimate of the error bound of $\|\boldsymbol{e}_y\|_{\ell_2}$ with $\boldsymbol{e}_y := y - \hat{y}$ is given by

$$\begin{aligned} \left\|\boldsymbol{e_y}\right\|_{\ell_2} &\leq \kappa(\boldsymbol{\mu}) \|\mathcal{R}\|_{\ell_2} \\ \text{with} \quad \kappa(\boldsymbol{\mu}) &:= \gamma^{\boldsymbol{e_y}\mathcal{R}} + \frac{\gamma^{\boldsymbol{e_y e_g}} L_g \gamma^{\boldsymbol{e}\mathcal{R}}}{1 - L_g \gamma^{\boldsymbol{ee_g}}}, \end{aligned} \tag{7.12}$$

with $\|(\cdot)\|_{\ell_2} := \sqrt{\sum_{n=0}^{\infty} \|(\cdot)\|^2}$ and γ^{yu} denoting the ℓ_2-norm of the system from input u to the output y.

Derivation: To define the error estimate, the error dynamics is defined by subtracting (7.11) from the full-order model (7.3)

$$\boldsymbol{e}^{n+1} = L_{lin}\boldsymbol{e}^n - \frac{\Delta t}{4\Delta x} L_{nl}((\boldsymbol{U}^n)^2 - (\hat{\boldsymbol{U}}^n)^2) - \mathcal{R}^n, \tag{7.13}$$

with $\boldsymbol{e} := \boldsymbol{U} - \hat{\boldsymbol{U}}$. By denoting $(\boldsymbol{U}^n)^2 - (\hat{\boldsymbol{U}}^n)^2$ as $\boldsymbol{e}_g^n$ and rewriting the dynamics in the feedback interconnected form, we obtain the error system Σ^e with its linear and nonlinear subsystems given as follows:

$$\begin{aligned} \Sigma_{lin}^e : &\begin{cases} \boldsymbol{e}^{n+1} = L_{lin}\boldsymbol{e}^n + \dfrac{\Delta t}{4\Delta x} L_{nl}\boldsymbol{e}_g^n - \mathcal{R}^n, \\ \boldsymbol{e}_y^n = C_y \boldsymbol{e}^n, \\ \boldsymbol{e}_z^n = \boldsymbol{e}^n, \end{cases} \\ \Sigma_{nl}^e : &\boldsymbol{e}_g^n = f(\hat{\boldsymbol{U}}, \boldsymbol{e}_z) = g(\boldsymbol{e}_z + \hat{\boldsymbol{U}}) - g(\hat{\boldsymbol{U}}). \end{aligned} \tag{7.14}$$

This feedback interconnection is depicted in Figure 7.2. Notably, the relation in Σ_{nl}^e holds regardless of using EIM as we have already lifted the solution to the full-order space. The effect of inaccurate approximation of the nonlinearities plays a role in the residual calculation, which is explained later in this section.

In the online phase, however, we do not have access to the values for $(\boldsymbol{U}^n)^2$ since the actual solution is not known. Therefore, an estimation of the output should be defined as we cannot simulate these error dynamics in a computationally efficient manner.

Following the idea introduced by [9] for linear systems and assuming Σ_{lin} is asymptotically stable (Assumption 7.1), an error bound on the ℓ_2-norm of the error signal can be computed as follows:

$$\left\|\boldsymbol{e_y}\right\|_{\ell_2} \leq \gamma^{\boldsymbol{e_y}\mathcal{R}} \|\mathcal{R}\|_{\ell_2} + \gamma^{\boldsymbol{e_y e_g}} \left\|\boldsymbol{e_g}\right\|_{\ell_2}. \tag{7.15}$$

This ℓ_2-norm is equal to the $\mathcal{H}_\infty$-norm of the linear system (7.14) with respect to the same input and output ([101]). Apart from the gains, in order to compute

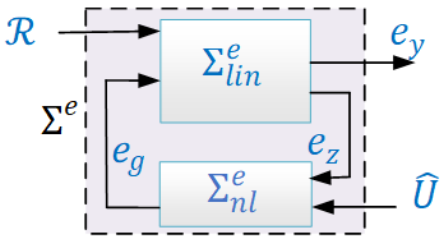

Figure 7.2: The schematic representation of the feedback interconnection for the error dynamics.

this error bound, both $\|\mathcal{R}^n\|$ and $\left\|\boldsymbol{e}^n_{\boldsymbol{g}}\right\|$ should be computed in a computationally efficient manner.
To compute the norm of the residual, we decompose the residual into a linear and a nonlinear part as below:

$$\mathcal{R}^n = \mathcal{R}^n_{lin} + \mathcal{R}^n_{nl}, \tag{7.16}$$

where

$$\begin{aligned}\mathcal{R}^n_{lin} &= \hat{\boldsymbol{U}}^{n+1} - \Big(L_{lin}\hat{\boldsymbol{U}}^n + BU^n_0 - \frac{\Delta t}{4\Delta x}L_{nl}q_{nl}\theta^n_{nl} \\ &\qquad + \frac{\Delta t}{2\Delta x}B(U^n_0)^2\Big), \\ \mathcal{R}^n_{nl} &= -\frac{\Delta t}{4\Delta x}L_{nl}(q_{nl}\theta^n_{nl} - (\hat{\boldsymbol{U}}^n)^2).\end{aligned} \tag{7.17}$$

In computing the two-norm of the residual $\mathcal{R}^n$, it is necessary to compute $\mathcal{R}^n_{nl}$, which is time-consuming due to the presence of the nonlinear term $(\hat{\boldsymbol{U}}^n)^2$. To avoid this computational issue, following the idea presented by [63], this term is calculated empirically by using Assumption 7.2. This assumption requires the reduced-order problem to be solved once more with an enriched set of collateral basis functions. Employing this assumption in the equation governing $\mathcal{R}^n_{nl}$ leads to

$$\mathcal{R}^n_{nl} = -\frac{\Delta t}{4\Delta x}L_{nl}(q_{nl}\theta^n_{nl} - q^*_{nl}{\theta^*_{nl}}^n). \tag{7.18}$$

The other required quantity for calculating the error estimate via (7.15) is $\|\boldsymbol{e_g}\|_{\ell_2}$. As $\boldsymbol{e^n_g} := \boldsymbol{e^n}$ represents the error in approximating the nonlinear function, we have

$$\left\|\boldsymbol{e^n_g}\right\| \leq L_g \left\|\boldsymbol{e^n}\right\|, \tag{7.19}$$

where L_g is an approximation of the local Lipschitz constant of the nonlinear operator g. The inequality (7.19) implies

$$\left\|\boldsymbol{e_g}\right\|_{\ell_2} \leq L_g \left\|\boldsymbol{e}\right\|_{\ell_2}. \tag{7.20}$$

Similar to (7.15), we have

$$\left\|\boldsymbol{e}\right\|_{\ell_2} \leq \gamma^{\boldsymbol{e}\mathcal{R}} \|\mathcal{R}\|_{\ell_2} + \gamma^{\boldsymbol{eeg}} \left\|\boldsymbol{e_g}\right\|_{\ell_2}. \tag{7.21}$$

Combining (7.20) and (7.21), and assuming that the small-gain condition $L_g \gamma^{\boldsymbol{eeg}} < 1$ holds, leads to

$$\left\|\boldsymbol{e}\right\|_{\ell_2} \leq \frac{\gamma^{\boldsymbol{e}\mathcal{R}}}{1 - L_g \gamma^{\boldsymbol{eeg}}} \|\mathcal{R}\|_{\ell_2}. \tag{7.22}$$

Finally, the use of this result in (7.15) gives (7.12). ■

Remark 7.4. *Exact satisfaction of Assumption 7.2 requires $M^* = \mathcal{N}$, which renders the error estimate expensive. In the results presented in this paper, we set $M^* = M + 1$. Therefore, $\|\mathcal{R}^n\|$ is computed cheaply and the ℓ_2-norm can be calculated.*

Remark 7.5. *To compute ${\mathcal{R}^n}^T \mathcal{R}^n$, some operators such as $\Phi^T L_{lin} L_{nl} q_{nl} \in \mathbb{R}^{N \times M}$ should be pre-computed during the offline phase and stored for usage during the online phase. Now, the two-norms of $\mathcal{R}^n$ can be computed with computations that scale at most with the dimension of q^*_{nl} or Φ, which is still much lower than the number of actual degrees of freedom of the high-fidelity scheme. For the details of residual calculation, we refer to [9]. Using Remark 7.4 instead of $M^* = \mathcal{N}$ renders the bound* (7.12) *to be an error estimate, not an actual error bound.*

Remark 7.6. *As the nonlinear operator for Burgers' equation $g(U) = (U)^2$ is not globally Lipschitz, we have to restrict the solution domain to be able to define a finite L_g. Note that the inequality* (7.19) *holds only locally as the value of L_g depends on the magnitude of $\boldsymbol{U}$, which restricts the range of $\boldsymbol{U}$ in the simulations. Assuming $\boldsymbol{e^n_z} := \boldsymbol{e^n}$ to be small and estimating the Lipschitz constant by the derivative of the nonlinear function $L_g = 2 \max_{i,n} U^n_i$ reveals that*

$$\max_{i,n} U^n_i < \frac{1}{2\gamma^{\boldsymbol{eeg}}}, \tag{7.23}$$

ensures that the small-gain condition in (7.14) *is satisfied.*

To enlarge and shift the applicability region, a loop transformation can be pursued as follows.

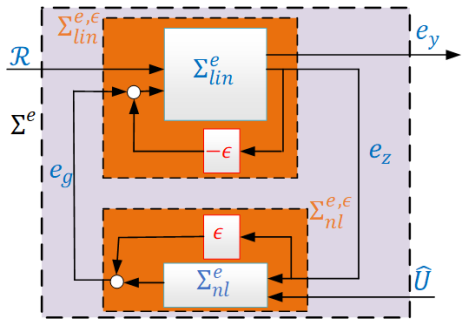

Figure 7.3: The schematic representation of the feedback interconnection within the error dynamics after loop transformation.

7.4.1.1 Loop transformation

The range of the applicability of the small-gain condition can be enlarged by using a so-called loop transformation (see [101]). In this section, we aim to apply this transformation to the feedback interconnection in (7.14) induced by the EIM and RB methods.

The loop transformation changes the interconnection in Figure 7.2 to Figure 7.3. The error dynamics after the loop transformation can be written as

$$\begin{aligned}
&\Sigma^{e,\epsilon}_{lin}: \begin{cases} \boldsymbol{e}^{n+1} = (L_{lin} - \epsilon\dfrac{\Delta t}{4\Delta x}L_{nl})\boldsymbol{e}^n + \dfrac{\Delta t}{4\Delta x}L_{nl}\boldsymbol{e}^n_g - \mathcal{R}^n, \\ \boldsymbol{e}^n_y = C_y\boldsymbol{e}^n, \\ \boldsymbol{e}^n_z = \boldsymbol{e}^n, \end{cases} \\
&\Sigma^{e,\epsilon}_{nl}: \boldsymbol{e}^n_g = g(\boldsymbol{e}_z + \hat{\boldsymbol{U}}) - g(\hat{\boldsymbol{U}}) + \epsilon\boldsymbol{e}_z.
\end{aligned} \tag{7.24}$$

It should be noted that Σ^e in (7.14) and (7.24) are exactly the same. The constant ϵ should be defined such that it minimizes the product $L_g\gamma^{ee_g}$ and therefore enlarges the applicability region while also reducing the conservatism in the small-gain condition and the estimate (7.12). For the parameterized system (7.24), the following minimization problem is solved to obtain ϵ,

$$\begin{aligned}
&\epsilon = \arg\min_{\epsilon} \left(\sum_i \left(\max\left(|2\boldsymbol{U}(\boldsymbol{\mu}^i) + \epsilon| \right) \times \gamma^{e_z e_g}(\boldsymbol{\mu}^i) \right) \right) \\
&\text{s.t.} \quad \forall \boldsymbol{\mu}^i \in \mathcal{D}_h \quad \begin{cases} \rho(L_{lin} - \epsilon\dfrac{\Delta t}{4\Delta x}L_{nl}) < 1 \\ \max(|2\boldsymbol{U}(\boldsymbol{\mu}^i) + \epsilon|) \times \gamma^{e_z e_g}(\boldsymbol{\mu}^i) < 1 \end{cases}
\end{aligned} \tag{7.25}$$

where $\rho(\cdot)$ is the spectral radius of a matrix and $\mathcal{D}_h$ is the discrete version of the varying parameter domain $\mathcal{D}$. For the test case under study, we have designed

the experiments such that

$$\min(\mu_1, \mu_2) \leq \boldsymbol{U}^n(\boldsymbol{\mu}^i) \leq \max(\mu_1, \mu_2). \tag{7.26}$$

The constraints in the minimization problem (7.25) ensure that for each parameter setting, first, the linear part of the error dynamics $\Sigma_{lin}^{e,\epsilon}$ is stable, and second, the interconnection of the linear subsystem $\Sigma_{lin}^{e,\epsilon}$ and the nonlinear subsystem $\Sigma_{nl}^{e,\epsilon}$ is also stable. In order to render the computations tractable, we terminate the minimization problem as soon as the constraints are satisfied.

Due to the fact that the nonlinear part of the system is not globally Lipschitz, a restriction on the region of the solution still holds after determining ϵ. In other words, to satisfy the small-gain condition, for all members of the parameter domain, we require (based on the second constraint in (7.25))

$$-\frac{1}{2\gamma^{e_z e_g}(\boldsymbol{\mu})} - \frac{1}{2}\epsilon < u(x,t;\boldsymbol{\mu}) < \frac{1}{2\gamma^{e_z e_g}(\boldsymbol{\mu})} - \frac{1}{2}\epsilon. \tag{7.27}$$

Therefore, the parameters, boundary conditions and initial conditions should be chosen in a way that the satisfaction of (7.27) would be possible. Based on the knowledge of the dependence of the ℓ_2-gains on ϵ and the variation of initial and boundary conditions, one can a priori have an insight whether this condition can be satisfied or not.

However, the error estimate (7.12), even with this loop transformation, can lead to conservative results. To alleviate the conservativeness, we tighten (sharpen) the error estimate as below.

7.4.1.2 Sharpening the error estimate

To resolve the problems of expensive calculation of the ℓ_2-gain of the system and conservativeness of the error estimate, we follow [9]. The main idea is that in the offline phase, the average of the conservatism of the error estimate is known and we can sharpen the error estimate in the online phase according to the experience in the offline phase. To do so, the error gain κ in (7.12) is multiplied by a reduction factor to obtain

$$\kappa^{e_y}(\boldsymbol{\mu}) = \bar{\rho}\kappa(\boldsymbol{\mu}), \tag{7.28}$$

where $\kappa^{e_y}\|\mathcal{R}\|_{\ell_2}$ is an estimate of $\|\boldsymbol{e}_y\|_{l_2}$, which is calculated based on κ in (7.12). To define $\bar{\rho}$, we first introduce the variable $\bar{\rho}_i^f$ as a measure of the conservatism

$$\bar{\rho}_i^f = \frac{\left\|\boldsymbol{e_y}(\boldsymbol{\mu}^{*,i})\right\|_{\ell_2}}{\left(\gamma^{\boldsymbol{e_y}\mathcal{R}} + \frac{\gamma^{\boldsymbol{e_y e_g}} L_g \gamma^{e\mathcal{R}}}{1 - L_g \gamma^{\boldsymbol{e e_g}}}\right) \|\mathcal{R}(\boldsymbol{\mu}^{*,i})\|_{\ell_2}}, \tag{7.29}$$

where $\boldsymbol{e_y}$ is the actual error computed in the offline phase for a parameter set $\boldsymbol{\mu}^{*,i}$ selected at the i-th stage of the greedy algorithm in the offline phase. The

denominator of (7.29) is motivated by (7.12). Then, $\bar{\rho}$ is defined as

$$\bar{\rho} = \max_i(\bar{\rho}_i^f). \tag{7.30}$$

In Section 7.5, the performance of the error estimate is investigated numerically. For the detailed algorithm of this error estimate, we refer to [9], where it is limited to systems without distributed nonlinearities.

7.4.2 Empirical error estimate

The underlying idea for the empirical error estimate is similar to the idea used for finding the contributed error from EIM ([63]) and the idea presented by [85].

Claim 7.7. *In the offline phase, we enrich RB functions from dimension N to dimension N' and the collateral basis functions from dimension M to dimension M' such that, based on the snapshots of previously selected parameters during the greedy algorithm, the following relation holds with $\eta_{N,M}^{N',M'} < 1$:*

$$\|y - \hat{y}_{N',M'}\|_{\ell_2} \leq \eta_{N,M}^{N',M'} \|y - \hat{y}_{N,M}\|_{\ell_2}, \tag{7.31}$$

where y is the actual output computed from (7.4) and $\hat{y}_{N,M}$ is obtained from (7.9) with N RB functions and M collateral basis functions. An output error estimate can be defined as

$$\|y - \hat{y}_{N,M}\|_{\ell_2} \leq \frac{\zeta_{N,M}^{N',M'}}{1 - \eta_{N,M}^{N',M'}}, \tag{7.32}$$

with

$$\zeta_{N,M}^{N',M'} = \|\hat{y}_{N',M'} - \hat{y}_{N,M}\|_{\ell_2}. \tag{7.33}$$

Derivation: To increase the accuracy in the offline phase, based on the snapshots of the current selected parameter $\boldsymbol{\mu}^{*,i}$ in the i-th iteration of the greedy algorithm, we enrich Φ and q_{nl} step by step. During the greedy algorithm, we increase N' and M' until $\eta_{N,M}^{N',M'}$ in (7.31) becomes smaller than 1 for all parameters whose corresponding full-solution is available. Therefore, for any (N, M), we can find (N', M') such that $\eta_{N,M}^{N',M'} < 1$. This condition bears similarities with the small-gain condition introduced in the first error estimate in this paper. Now, in the offline phase, corresponding to each (N, M), a pair of (N', M') and the value of $\eta_{N,M}^{N',M'}$ are known.

In the online phase, two reduced solutions with (N, M) and (N', M') basis functions should be solved. After obtaining these two computationally cheap solutions, we set

$$\zeta_{N,M}^{N',M'} = \|\hat{y}_{N',M'} - \hat{y}_{N,M}\|_{\ell_2}. \tag{7.34}$$

Algorithm 9: Empirical error estimate

Input: $q_{nl}, \Phi, \boldsymbol{X_m}$, parameters selected in the previous greedy iteration $\boldsymbol{\mu}^*$ and their corresponding full solutions

Output: $N', M', \eta_{N,M}^{N',M'}$

1 Set $N' = N$ and $M' = M$,
2 Based on the recently selected parameters, enrich $\Phi(N' = N' + 1)$ and $q_{nl}, \boldsymbol{X_m}(M' = M' + 1)$
3 compute $\eta^{N',M'} = \|y - \hat{y}_{N',M'}\|_{\ell_2}$ and $\eta_{N,M} = \|y - \hat{y}_{N,M}\|_{\ell_2}$ for all members of $\boldsymbol{\mu}^*$,
4 Set $\eta = \max\left(\frac{\eta^{N',M'}}{\eta_{N,M}}\right)$
5 **if** $\eta < 1$ **then**
6 $\quad \eta_{N,M}^{N',M'} = \eta$
7 **else**
8 $\quad$ Go back to step 2

Then, based on the following inequality

$$\|y - \hat{y}_{N,M}\|_{\ell_2} \leq \|y - \hat{y}_{N',M'}\|_{\ell_2} + \|\hat{y}_{N',M'} - \hat{y}_{N,M}\|_{\ell_2}, \tag{7.35}$$

and taking into consideration from the offline phase that $\|y - \hat{y}_{N',M'}\|_{\ell_2} \leq \eta_{N,M}^{N',M'} \|y - \hat{y}_{N,M}\|_{\ell_2}$, we finally obtain

$$\|y - \hat{y}_{N,M}\|_{\ell_2} \leq \frac{\zeta_{N,M}^{N',M'}}{1 - \eta_{N,M}^{N',M'}}. \tag{7.36}$$

The reason for having $\eta_{N,M}^{N',M'} < 1$ shows itself here to have finite and positive error estimate. ■

For the implementation of this error estimate, refer to Algorithm 9.

7.5 Numerical results

The simulation parameters in the online phase $\boldsymbol{\mu}^o$ for Burgers' equation along with the parameter domain are listed in Table 7.1, where the minimum and maximum value for each parameter are specified. The discrete parameter domain is composed of 8 equidistant members in the parameter domain. In the last row of Table 7.1, the set of parameters selected for the online simulation $\boldsymbol{\mu}^o$ is reported, which does not lie in the discrete parameter domain. This kind of parameter setting ensures that $4 \leq u(t, x; \boldsymbol{\mu}) \leq 7$ for all $(t, x) \in [0, T] \times [0, L]$. The number of spatial grid cells is $\mathcal{N} = 250$, the time horizon T is 50 s and time step is $\Delta t = 0.01$ s. The output is the value of the conservative variable at $x = L$.

Table 7.1: Test case parameter range for Burgers' equation.

parameter	L [m]	μ_1	μ_2
minimum	100	4	6
maximum	110	5	7
Online $\boldsymbol{\mu}^o$	105	4.5	6.5

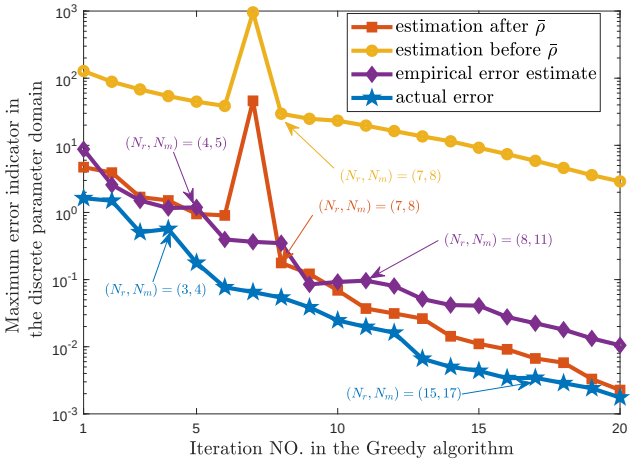

Figure 7.4: Maximum error in the discrete parameter domain during the greedy algorithm.

The effect of using the actual error, the error estimates based on the ℓ_2-gain notion (with and without the reduction factor $\bar{\rho}$ in (7.30)) and the empirical error estimate in the greedy algorithm of PODEI algorithm (Algorithm 8) as in [63] is shown in Figure 7.4. Clearly, the error estimates accurately approximate the maximum error in the parameter domain. The "estimation before $\bar{\rho}$" is the most conservative error estimate, which is due the high conservativeness in the ℓ_2-gain notion. In the eighth iteration, using the ℓ_2-based error estimate, the collateral basis function is inconsistent with the RB basis functions, which reduces the accuracy of the RB solution and increases the residual values. Discarding the collateral basis functions, as in PODEI algorithm, resolves the problem for the next iteration. In general, the accuracy of the RB solution increases by enriching the RB and collateral basis functions.

In the online phase by using 20 RB functions and 20 collateral basis functions, the time-wise evolution of the solution is shown in Figure 7.5 at four different time instants in comparison with the FV solution. The speedup factor is reported in Table 7.2 (without including the error estimate computational time). The

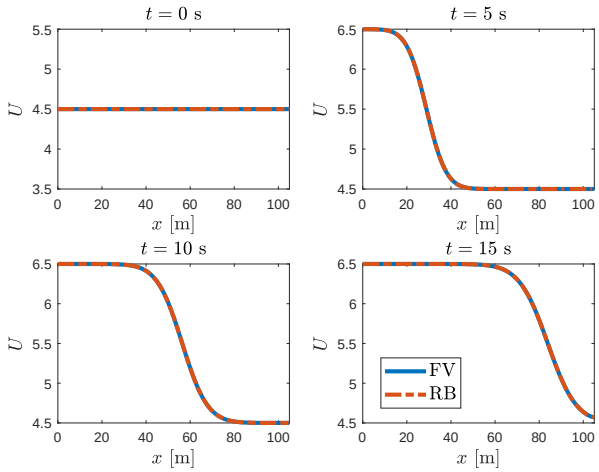

Figure 7.5: Comparison of the full-order and low-order solutions over time using 20 RB functions and 20 collateral basis functions.

moderate speedup is due to the hyperbolic, nonlinear and 1D nature of the original problem. The effect of the number of RB functions in the induced error due to the reduction for the parameters used in the online phase is shown in Figure 7.6.

The results of this section verify that both error estimates perform successfully in estimating the maximum error during the greedy algorithm in the offline phase and also estimating the error for a new parameter setting during the online phase. However, the estimate based on the ℓ_2-gain notion suffers from restricted applicability to satisfy the small-gain condition. This becomes even more restricted in the case of stronger nonlinearities (nonlinearities with higher local Lipschitz constant). On the other hand, in the empirical error estimate, we only need to find (N', M') to be sufficiently large to satisfy the condition on $\eta_{N,M}^{N',M'} < 1$. Apart from this condition that should be resolved in the offline phase, there is no restriction on the applicability of the method in the online phase.

Table 7.2: Speedup factors for the reduced basis method for Burgers' equation.

$N = M$	1	5	10	15	20
Speedup	17.4	4.2	4	3.4	3

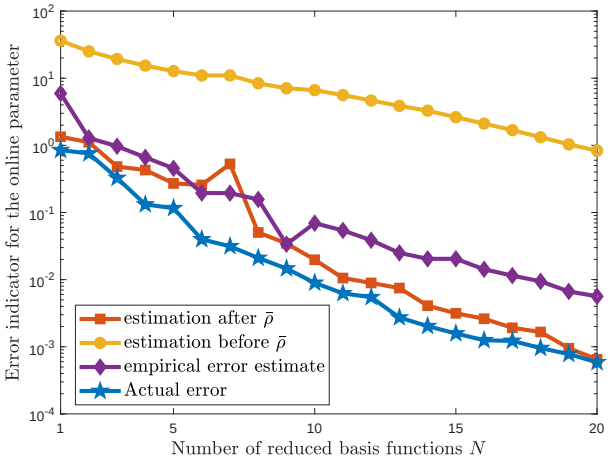

Figure 7.6: Error evolution by increasing the number of basis functions.

7.6 Conclusion

In this paper, a new perspective on the interaction between EIM and RB methods is introduced. First, a new error estimate based on a Lur'e type formulation of nonlinear Burgers' equation is defined. This estimate is rigorous, accurate and effective, but has limited applicability due to satisfying a small-gain condition. Furthermore, it requires another reduced-order model to be solved to approximate the residual. To circumvent the small-gain condition issue, hinged on the snapshots generated in the offline phase, an empirical error estimate is introduced that does not suffer from the restrictions of the first error estimate. Both error estimates work efficiently in terms of computational effort and accuracy. The empirical error estimate is faster and also applicable on a wider range of problems than the error estimate proposed on the basis of ℓ_2-gain notion.

Chapter 8

Reduced Basis Method for Managed Pressure Drilling Based on a Model with Local Nonlinearities

To circumvent restrictions of conventional drilling methods, such as slow control actions and inability to drill depleted reservoirs, a drilling method called Managed Pressure Drilling (MPD) has been developed. In MPD, single-phase flow processes can be modeled as a feedback interconnection of a high-order linear system and a low-order nonlinear system. These nonlinearities appear locally both inside and at the boundaries of the computational domain. To obtain a fast simulation platform for real-time purposes (e.g. online model-based controller implementation), model order reduction is required for MPD. However, the local nonlinearities render applying model order reduction techniques challenging. In this study, a new approach is proposed to deal with such nonlinearities within the Reduced Basis (RB) context and it is successfully tested on a model for MPD. Contrary to the classical RB technique, the proposed approach not only does not generate non-physical spikes at the locations of these local nonlinearities but also yields high speedup factors. The obtained reduced-order model can be used for efficient online simulation and controller design for drilling systems with MPD.

This chapter is based on "M.H. Abbasi, L.Iapichino, S.Naderi Lordejani, W.H.A.Schilders, N.van de Wouw, ***Reduced Basis Method for Managed Pressure Drilling Based on a Model with Local Nonlinearities***, *International Journal for Numerical Methods in Engineering*, in press, 2020."

8.1 Introduction

The design of pressure control systems for drilling with Managed Pressure Drilling (MPD) requires an accurate model of the hydraulics of the system. Moreover, training novice drillers for drilling operations, simulating well control incidents in a controlled environment and also monitoring drilling wells are other important reasons for hydraulics modeling in drilling. However, the desire for accurate modeling of drilling systems typically leads to highly complex models. In particular, this type of modeling for MPD automation gives rise to models involving parameterized, nonlinear, non-conservative hyperbolic Partial Differential Equations (PDEs) completed by nonlinear and implicit boundary conditions [5], [128], [129]. The governing equations can become even more complicated when the flow path cross-section is discontinuous along the well. All these features of the model render its numerical simulation computationally expensive and also make the controller design for the system cumbersome. Moreover, an accurate and fast assessment of the downhole pressure is crucial in drilling processes. A tool that facilitates this assessment can be a significant help/support for the drilling procedure. For instance, in case of emergencies, an easy-to-use and much faster than real-time simulation framework can provide insights into effects of a potential reaction to contingencies occurring during drilling. This reduces the probability of making hazardous decisions and hence increases the safety of the drilling rig. To provide such efficient model-based simulation tool, Model Order Reduction (MOR) [13] can be very useful.

MOR techniques aim at the automatic construction of reduced-complexity models that combine high predictive capacity and low complexity. These techniques drastically reduce the size of the problem and, subsequently, its computational cost at a price of a minimal and quantifiable loss of model accuracy. The resulting low-complexity models can then be used for simulation, control, optimization, parameter estimation and inverse modeling [13]. Among several existing techniques for reducing the dimension of expensive and complex computational models, the Reduced Basis (RB) [89] method is an efficient approach for dealing with parameterized systems. In particular, the MPD model described in [128] can be considered as a parameterized system where the (varying) parameters can be the properties of the drilling mud or the wellbore geometry.

The use of the RB method for MPD models is not straightforward due to the time-dependent, nonlinear and state-dependent boundary conditions. The underlying idea of the RB method is to find the solution of the system as a combination of some precomputed functions, called RB functions. Handling boundary conditions becomes challenging due to the global nature of the RB functions and their independency over time [53]. To resolve this issue, a lifting method [9] is implemented here and tested for an industry-relevant MPD application. Within this lifting method, the effect of the boundary conditions is segregated from the internal domain and is incorporated later as a state-dependent and locally acting

input.

Besides the nonlinearities at the boundaries, local nonlinearities can occur inside the computational domain. One realistic example is represented by the discontinuities in the cross-section of the flow path in a drilling system [10]. These local nonlinearities generate discontinuities in the solution of the state variables. While various reduction techniques for distributed nonlinearities, such as Empirical Interpolation Method (EIM) [17] and Discrete EIM (DEIM) [44], have been developed, little attention,to the best of our knowledge, has been paid to the issue of local nonlinearities in the RB setting. To take into account the local nonlinearities inside the domain, the RB functions are enriched with some local basis functions with compact support (henceforth called the localized enrichment). This method shares similarities with the approaches introduced in the balanced truncation community [27], [129].

In this paper, to the best of our knowledge, for the first time the RB method is applied to a drilling system with MPD and the ansatz introduced in [9] is tested for dealing with implicit and nonlinear boundary conditions governing the hydraulics dynamics of the drilling system. Moreover, a new methodology is introduced to automatically capture the local nonlinearities inside the computational domain by enriching RB functions with specific local basis functions.

The outline of this paper is as follows. In Section 8.2, a model for MPD is briefly introduced. In Section 8.3, the new approach to use the RB method for capturing local nonlinearities in MPD is proposed. In Section 8.4, numerical results comparing the full-order model and the reduced-order model are presented. Finally, conclusions and future works are presented in Section 8.5.

8.2 Problem Statement

The industrial problem under investigation is a drilling system, with a special focus on MPD. The configuration of the system is illustrated in Figure 8.1. A drilling liquid known as mud is pumped into a pipe at high pressure; this pipe is called the drillstring. At the bottom of the drillstring, the mud leaves the drillstring through nozzles located inside the drill bit and enters the area between the drillstring and the wellbore known as the annulus. It then flows up through the annulus and carries the rock cuttings out of the well. In MPD, the annulus is sealed off from the surroundings at the top with a Rotating Control Device (RCD) and the mud circulates out of the well through a choke valve. The circulation path of the mud can be observed by following the green arrows in Figure 8.1. Furthermore, the diameters of the drillstring and the wellbore experience sudden variations along the well; the diameter profile of the drillstring and the wellbore are shown in red (denoted by d) and blue lines (denoted by D_{out}), respectively, in Figure 8.1. The parameters L and θ denote the well length and the well inclination with respect to the horizontal line. In this paper, a single-phase flow model for MPD is studied. For a more comprehensive explanation of

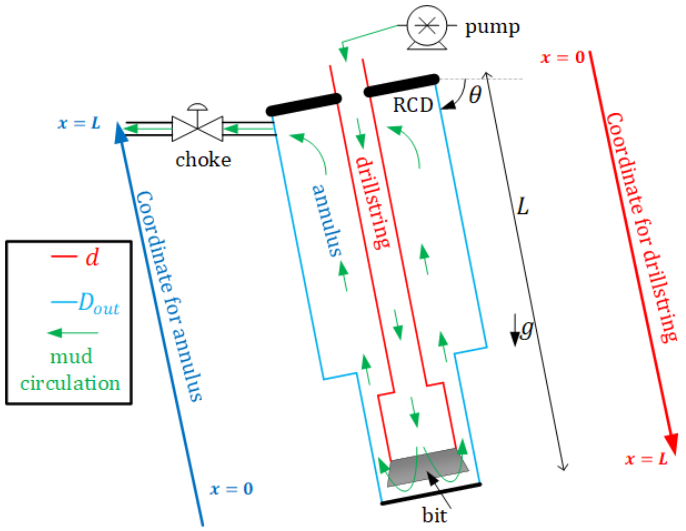

Figure 8.1: A schematic configuration of an MPD system.

MPD systems and multi-phase flow modeling, the reader is referred to [128]. The mathematical formulation employed to model this system is described below.

8.2.1 Mathematical modeling for MPD

To model the drilling hydraulics, the drillstring and annulus can be treated as two pipes connected at the bit and the hydraulics within MPD can be modeled by the so called U-tube modeling approach [128]. Different models have been introduced for the mathematical representation of the single-phase flow inside each pipe [99], [128]. One of the most widely used models is obtained by simplifying the isothermal Euler equations and accounting for the area variation of the drillstring and the wellbore. This model is described by the following system of PDEs:

$$\begin{aligned}
&\frac{\partial(\boldsymbol{u}A)}{\partial t}+\frac{\partial(f(\boldsymbol{u},x))}{\partial x}=\boldsymbol{S}(\boldsymbol{u},x)\rightarrow\\
&\begin{cases}
\dfrac{\partial(\rho A)}{\partial t}+\dfrac{\partial(\rho A v)}{\partial x}=0,\\
\dfrac{\partial(\rho A v)}{\partial t}+c_l^2\dfrac{\partial(\rho A)}{\partial x}=A\left(F(\boldsymbol{u},x)+G(\boldsymbol{u},x)\right)+p\dfrac{\partial A}{\partial x},
\end{cases}
\qquad x\in[0,L], t\in[0,T],
\end{aligned} \tag{8.1}$$

where $\rho(t,x)$, $v(t,x)$ and $A(x)$ are density, velocity and the cross-sectional area of each pipe, respectively. Here, t and x denote the temporal variable and the spatial variable, respectively. Parameters T and L, respectively, denote

the temporal and spatial length of the computational domain. In addition, $\boldsymbol{u} = [\rho,\ \rho v]^T$ (noteworthy, $m(t,x) := \rho(t,x)v(t,x)$ is the momentum of the flow, and superscript $(\cdot)^T$ shows the transpose operator). The source term $\boldsymbol{S}(\boldsymbol{u},x)$ consists of a friction term $F(\boldsymbol{u},x)$, a gravitation term $G(\boldsymbol{u},x)$ and the effect of variable cross-sectional area. The flux function $f(\boldsymbol{u},x)$ attributed to (8.1) is $f(\boldsymbol{u},x) = [\rho A v,\ \rho A c_l^2]^T$ where c_l is the speed of sound in the fluid medium.

In case of laminar flow, we have

$$F(\boldsymbol{u},x) = -32\frac{\gamma}{D_h^2}\frac{\rho v}{\rho_{0_s}}, \tag{8.2}$$

where $D_h(x)$ is the hydraulic diameter of each pipe, ρ_{0_s} is the reference density and γ is the viscosity of the liquid. For gravity, we set

$$G(\boldsymbol{u},x) = \pm\rho g \sin\theta, \tag{8.3}$$

where g is the gravitational acceleration and θ is the inclination of the pipe. To calculate pressure, the Equation Of State (EOS) for the fluid can be used, which reads as follows:

$$p = p_0 + c_l^2(\rho - \rho_{0_s}), \tag{8.4}$$

where $p(t,x)$ is the pressure and p_0 is the reference pressure. Although system (8.1) is parameter-dependent (e.g., the inclination θ and the properties of the fluid such as c_l and ρ_{0_s} can vary), the parameter dependency is not shown for the sake of simplicity. The parameter dependency is made explicit later in Section 8.3.

Remark 8.1. *System* (8.1) *is solved once for the drillstring and once for the annulus. These two solutions then interact at the bit. For the simulation of the hydraulics in the drillstring, we set* $D_h(x) = d(x)$ *where* $x = 0$ *and* $x = L$ *correspond, respectively, to the pump and bit location in the drillstring. Additionally, for gravitational source term, we choose "+" sign in* (8.3)*. For the annulus, we set* $D_h(x) = D_{out}(x) - d(x)$ *with* $x = 0$ *indicating the bit location and* $x = L$ *indicating the choke location with "-" sign in the gravity term* (8.3)*.*

Remark 8.2. *Area discontinuities, such as those presented in Figure 8.1, leads to an impulse in the right-hand side of the momentum equation in* (8.1)*, leading to a discontinuity in the density and velocity. At these locations, where* v *is discontinuous and* $\partial v/\partial x$ *is an impulse, the simplified isothermal Euler equations as in* (8.1) *are not accurate. At these locations, the original isothermal Euler equations as in [57] are exploited to solve the fluid flow.*

The Finite-Volume (FV) method is usually employed to numerically solve this type of system of PDEs [113]. In the following section, the numerical framework to solve system (8.1) is discussed.

8.2.2 Finite-volume discretization

To numerically solve (8.1), a Gudonov-type scheme is employed as follows

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{F}\left(U_i^n, U_{i+1}^{n,*}\right) - \mathcal{F}\left(U_{i-1}^{n,*}, U_i^n\right)\right) + \Delta t \;\left(F\left(U_i^n, x_i\right) + G\left(U_i^n, x_i\right)\right), \tag{8.5}$$

where the discrete field variable $U_i^n = [\rho_i^n,\ \rho_i^n v_i^n]^T$, $i \in \{1, \cdots N_\delta\}$, $n \in \{1, \cdots N_t\}$, is the average of field variables $\boldsymbol{u} = [\rho,\ \rho v]^T$ over the i-th grid cell at the time instant $t^n := n\Delta t$. Moreover, N_δ and N_t are, respectively, the number of spatial and temporal grid cells of the discretized computational domain. Also, $\mathcal{F}(\cdot,\cdot)$ is the scheme-specific numerical flux function (to be introduced in Section 8.2.4). Here, Δt and Δx refer to the temporal and spatial discretization step sizes, respectively. It should be noted that parameter dependency is not mentioned in (8.5) for the sake of notational simplicity.

The effects of area variation are incorporated by the starred variables $(U_{i+1}^{n,*}, U_{i-1}^{n,*})$ appearing in (8.5). These variables can be computed by solving the following system of equations at each time step (refer to [103] for details),

$$\rho_{i+1}^{n,*} v_{i+1}^{n,*} A_i = \rho_{i+1}^n v_{i+1}^n A_{i+1}, \tag{8.6a}$$

$$c_l^2 \ln \rho_{i+1}^{n,*} + \frac{(v_{i+1}^{n,*})^2}{2} = c_l^2 \ln \rho_{i+1}^n + \frac{(v_{i+1}^n)^2}{2}. \tag{8.6b}$$

After obtaining $\rho_{i+1}^{n,*}$ and $v_{i+1}^{n,*}$, $U_{i+1}^{n,*} = [\rho_{i+1}^{n,*},\ \rho_{i+1}^{n,*} v_{i+1}^{n,*}]^T$ is constructed at each time step n ($U_{i-1}^{n,*}$ is formed in the same manner). Since this approach is developed only to consider the effects of diameter discontinuity, system (8.6) is solved only at the locations of area variations. At other locations in the spatial domain where the area does not change over the interfaces, we set $U_{i+1}^{n,*} = U_{i+1}^n$ and $U_{i-1}^{n,*} = U_{i-1}^n$.

Remark 8.3. *The set of equations (8.6) are derived according to the isothermal Euler equations, not (8.1), see Remark 8.2.*

8.2.3 Initial and boundary conditions

In order to be able to numerically solve system (8.1), initial conditions and boundary conditions should be imposed. Initial conditions are chosen to the steady-state solution of system (8.1) corresponding to a selected set of parameters. Then, by changing the inputs of the system (i.e., the boundary inputs), the dynamics of the system are excited.

System (8.1) consists of two first-order PDEs; hence, two physical boundary conditions for each pipe should be defined. For the drillstring, the governing equation of a pump comprises the left boundary condition while the drilling bit equation governs the right boundary condition. For the annulus, on the other

hand, the drilling bit constitutes the left boundary, while the right boundary is specified by a choke valve, see Figure 8.1. The governing equations of the pump, bit and choke are summarized in Table 8.1. There, q_p, A_p and v_p represent the volumetric flow rate of the pump, the cross-sectional area at the pump and the velocity of the liquid at the pump. Moreover, $\Delta p_b, \rho_b, \dot{m}_b, A_N$ and C_D denote the pressure drop over the bit, density at the drillstring side of the bit, the mass flow rate through the bit, the total area of the nozzle holes of the bit and the nozzle coefficient of the bit. Finally, $q_c, K_c, z_c, \rho_c, p_c$ and p_{atm} are the volumetric flow rate through the choke, the choke constant, the choke opening, the density at the choke inlet, the pressure at the choke inlet and atmospheric pressure, respectively. Control inputs of the system are q_p and z_c. For a more comprehensive description of the boundary conditions, refer to [128].

Remark 8.4. *It should be noted that a Non-Return Valve (NRV) is always installed above the drilling bit inside the annulus. This valve allows the drilling mud to flow only from the drillstring into annulus, not vice versa. The NRV model is introduced in [128].*

8.2.4 Upwind scheme

All numerical test cases in this paper are performed by an upwind scheme [50]. The corresponding numerical flux function in (8.5) holds

$$\mathcal{F}(U_i^n, U_{i+1}^n) = \mathcal{A}^+ U_i^n + \mathcal{A}^- U_{i+1}^n, \tag{8.7}$$

where

$$\mathcal{A}^- = \frac{1}{2}\begin{bmatrix} -c_l & 1 \\ c_l^2 & -c_l \end{bmatrix}, \quad \mathcal{A}^+ = \frac{1}{2}\begin{bmatrix} c_l & 1 \\ c_l^2 & c_l \end{bmatrix}. \tag{8.8}$$

By inserting (8.7) into (8.5), the following equation is attained by compressing the notation ($F_i^n := F\left(U_i^n, x_i\right)$ and similarly for G_i^n):

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{A}^- U_{i+1}^{n,*} - \mathcal{A}^+ U_{i-1}^{n,*} + \left(\mathcal{A}^+ - \mathcal{A}^-\right) U_i^n\right) + \left(F_i^n + G_i^n\right)\Delta t. \tag{8.9}$$

Using (8.9), one can find the vector of discrete field variables over the i-th cell at each time step. Augmenting all $U_i^n,\ i \in \{1, \cdots N_\delta\},\ n \in \{1, \cdots N_t\}$, over

Table 8.1: Governing equations of the pump, bit and choke.

pump	bit	choke
$q_p = A_p v_p$	$\Delta p_b = \frac{1}{2\rho_b}\left(\frac{\dot{m}_b}{A_N C_D}\right)^2$	$q_c = K_c z_c \sqrt{\frac{2}{\rho_c}\left(p_c - p_{\text{atm}}\right)}$

all cells and taking into account the variation of the diameter (by solving (8.6)) yields the following algebraic equations

$$\begin{aligned}\boldsymbol{\rho}^{n+1} = &\left(\mathrm{I} + c_l\frac{\Delta t}{\Delta x}L_1\right)\boldsymbol{\rho}^n + \frac{\Delta t}{2\Delta x}L_2\boldsymbol{m}^n \\ &+ c_l\frac{\Delta t}{\Delta x}L_1^*\xi\rho^{+,n} + \frac{\Delta t}{2\Delta x}L_2^*\xi m^{+,n} \\ &+ \frac{c_l}{2}\frac{\Delta t}{\Delta x}B_0\rho_0^n + \frac{c_l}{2}\frac{\Delta t}{\Delta x}B_{N_\delta+1}\rho_{N_\delta+1}^n + \frac{1}{2}\frac{\Delta t}{\Delta x}B_0 m_0^n - \frac{1}{2}\frac{\Delta t}{\Delta x}B_{N_\delta+1}m_{N_\delta+1}^n,\end{aligned} \tag{8.10}$$

$$\begin{aligned}\boldsymbol{m}^{n+1} = &\left(\frac{c_l^2}{2}\frac{\Delta t}{\Delta x}L_2 \pm g\sin\theta\Delta t\mathrm{I}\right)\boldsymbol{\rho}^n + \\ &\left(\mathrm{diag}\left(1 - 32\frac{\gamma}{\rho_{0_s}\boldsymbol{D}_{\boldsymbol{h}}^2}\right)\Delta t + c_l\frac{\Delta t}{\Delta x}L_1\right)\boldsymbol{m}^n + \frac{c_l^2}{2}\frac{\Delta t}{\Delta x}L_2^*\xi\rho^{+,n} + \\ &c_l\frac{\Delta t}{\Delta x}L_1^*\xi m^{+,n} - \frac{c_l^2}{2}\frac{\Delta t}{\Delta x}B_0\rho_0^n - \frac{c_l^2}{2}\frac{\Delta t}{\Delta x}B_{N_\delta+1}\rho_{N_\delta+1}^n + \frac{c_l}{2}\frac{\Delta t}{\Delta x}B_0 m_0^n + \\ &\frac{c_l}{2}\frac{\Delta t}{\Delta x}B_{N_\delta+1}m_{N_\delta+1}^n,\end{aligned} \tag{8.11}$$

where $\boldsymbol{\rho}, \boldsymbol{m}, \boldsymbol{D_h} \in \mathbb{R}^{N_\delta}$ are, respectively, the vector-valued averages of the density and momentum and the vector containing diameter values at all grid cells gathered in a vector of dimension N_δ (number of grid cells). Matrix I is the square identity matrix of dimension N_δ. The function "diag" in (8.11) generates a square matrix with the diagonal terms according to its argument. Furthermore, ρ_0^n and m_0^n are the left boundary values of each pipe at time instant with index n while $\rho_{N_\delta+1}^n$ and $m_{N_\delta+1}^n$ are the right boundary values of each pipe at time instant with index n. We recall that in the term $32\gamma/(\rho_{0_s}\boldsymbol{D}_{\boldsymbol{h}}^2)$ in (8.11), we set $\boldsymbol{D_h} = \boldsymbol{d}$ for the drillstring and $\boldsymbol{D_h} = \boldsymbol{D_{out}} - \boldsymbol{d}$ for the annulus ($\boldsymbol{d}$ is the discrete version of $d(x)$ and similarly for $\boldsymbol{D_{out}}$). Also, we have

$$L_1 = \begin{bmatrix} -1 & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & -1 \end{bmatrix}_{N_\delta\times N_\delta}, \quad L_2 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}_{N_\delta\times N_\delta},$$

$$B_0 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{N_\delta\times 1}, \quad B_{N_\delta+1} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{N_\delta\times 1}. \tag{8.12}$$

In (8.10), (8.11), the variables with the superscript "+", $(\cdot)^{+,n} \in \mathbb{R}^{2N_d}$ with N_d number of diameter discontinuities along the spatial domain, are defined as:

$$(\cdot)^{+,n} = (\cdot)^{*,n} - (\cdot)^{n}, \tag{8.13}$$

where the variable $(\cdot)^n$ is subtracted from the starred variable $(\cdot)^{*,n}$ computed by (8.6).

The operators L_1^* and L_2^* incorporate the nonlinearities due to discontinuities in the diameter. These operators are tridiagonal matrices with zero diagonal terms and nonzero entities at the off-diagonal entities corresponding to the locations of area discontinuities. These nonzero terms are respectively a copy of the off-diagonal terms of L_1 and L_2 at the neighboring cells of the interface where the area discontinuity occurs; the other off-diagonal entities are set to zero. The operator $\xi \in \mathbb{R}^{N_\delta \times 2N_d}$ in (8.10),(8.11) is composed of the canonical vectors at the neighboring cells of the interface with the area discontinuity. For instance, if the area variation only occurs at the interface $x_{i+1/2}$ of the discretized domain, we have:

$$L_1^* = \begin{array}{c} \\ 1 \\ \vdots \\ i \\ i+1 \\ \vdots \\ N_\delta \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & \dots & i & i+1 & \dots & N_\delta \end{array} \\ \begin{bmatrix} 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \frac{1}{2} & \cdots & 0 \\ 0 & \cdots & \frac{1}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \quad \xi = [e_i,\ e_{i+1}], \tag{8.14}$$

where $e_i \in \mathbb{R}^{N_\delta}$ is the i-th canonical vector (a zero vector with only one nonzero element at the i-th entry).

To show the internal dynamics (8.10),(8.11), boundary equations and the area discontinuity solver (8.6) in a compact form, a description of the discretized system dynamics is shown below in the form of a feedback interconnection:

$$\Sigma : \begin{cases} \Sigma_{lin} : \begin{cases} \boldsymbol{\rho}^{n+1} = A_{11}\boldsymbol{\rho}^n + A_{12}\boldsymbol{m}^n + B_1 V^n + B_1^* W^n, \\ \boldsymbol{m}^{n+1} = A_{21}\boldsymbol{\rho}^n + A_{22}\boldsymbol{m}^n + B_2 V^n + B_2^* W^n, \\ y^n = C[\boldsymbol{\rho}^{nT} \quad \boldsymbol{m}^{nT}]^T, \\ y_V^n = C_V[\boldsymbol{\rho}^{nT} \quad \boldsymbol{m}^{nT}]^T, \\ y_W^n = C_W[\boldsymbol{\rho}^{nT} \quad \boldsymbol{m}^{nT}]^T, \end{cases} \\ \Sigma_{nl} : \begin{cases} V^n = h(y_V^{n-1}, V^{n-1}, u_c^{n-1}), \\ W^n = \mathcal{G}(y_W^n), \end{cases} \end{cases} \tag{8.15}$$

where $h(\cdot,\cdot,\cdot)$ is the solver of the dynamics occurring at the boundaries, u_c contains the control inputs acting at the boundaries (in MPD, $u_c = [q_p,\ z_c]$),

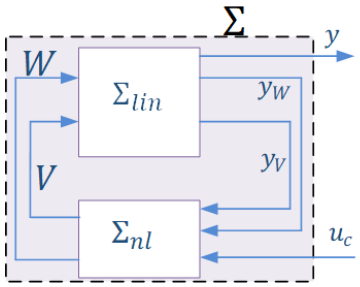

Figure 8.2: A schematic presentation of the full-order model consisting of a feedback interconnection of the linear and nonlinear part.

and $\mathcal{G}(\cdot)$ is the solver of (8.6). We have set $W^n = [\rho^{+,n^T},\ m^{+,n^T}]^T, V^n = [\rho_0^n,\ \rho_{N_\delta+1}^n,\ m_0^n,\ m_{N_\delta+1}^n]^T$. The other matrices such as A_{11} are defined according to (8.10)-(8.11) and $y^n \in \mathbb{R}^{N_y}$ is a vector of the outputs of interest with the dimension N_y. The other outputs of Σ_{lin}, y_V^n and y_W^n (correspondingly the output matrices C_V and C_W), are respectively required for the computations of the boundary conditions and the area discontinuities. This interconnection is shown in Figure 8.2.

It should be noted that the area variation changes the wave reflection pattern inside the system. It also significantly changes the velocity profile and therefore the pressure drop due to the friction. The formulation (8.15) allows to capture these effects even when the order of the system is reduced by the RB method.

Remark 8.5. *Before going to the model order reduction part, let us add the parameter dependency into* (8.15),

$$\Sigma(\mu): \begin{cases} \Sigma_{lin}(\mu): \begin{cases} \boldsymbol{\rho}^{n+1} = A_{11}(\mu)\boldsymbol{\rho}^n + A_{12}(\mu)\boldsymbol{m}^n + B_1(\mu)V^n + B_1^*(\mu)W^n, \\ \boldsymbol{m}^{n+1} = A_{21}(\mu)\boldsymbol{\rho}^n + A_{22}(\mu)\boldsymbol{m}^n + B_2(\mu)V^n + B_2^*(\mu)W^n, \\ y^n = C[\boldsymbol{\rho}^{n^T} \quad \boldsymbol{m}^{n^T}]^T, \\ y_V^n = C_V[\boldsymbol{\rho}^{n^T} \quad \boldsymbol{m}^{n^T}]^T, \\ y_W^n = C_W[\boldsymbol{\rho}^{n^T} \quad \boldsymbol{m}^{n^T}]^T, \end{cases} \\ \Sigma_{nl}(\mu): \begin{cases} V^n = h(y_V^{n-1}, V^{n-1}, u_c^{n-1}; \mu), \\ W^n = f(y_W^n; \mu), \end{cases} \end{cases} \tag{8.16}$$

where μ is a vector in $\mathcal{D} \in \mathbb{R}^b, b \geq 0$. Here, $\mathcal{D}$ is a continuous space containing varying parameters of the system.

To summarize, system (8.16) is linear except at the locations of area discontinuities and at the boundaries. The high-dimensional nature of the system dynamics Σ is only present in Σ_{lin} and not in the nonlinear part Σ_{nl} and this motivates the system complexity reduction by reducing only the linear part. Thus,

developing a MOR method to reduce the linear part and simultaneously account for this type of local nonlinearities is essential, which is the topic of the next section. We in particular consider the case in which the computational effort to solve for the dynamics of Σ_{lin} is significantly higher than that required for Σ_{nl}. We care to stress that this is the case in many application scenarios in which Σ_{lin} expresses the discretized, large-scale, dynamics of an internal domain and Σ_{nl} expresses nonlinear, though low-dimensional, dynamics (or even static relationships) characterizing local nonlinearities. In such relavant scenarios, substantial computational efficiency gains can be expected after reduction. The reduction approach can still be pursued if this assumption is violated, but the computational gain may be less significant. In support of the MOR paradigm, we adopt the following assumptions.

Assumption 8.6. *To enable efficient reduction of* (8.16)*, all operators in* $\Sigma_{lin}(\mu)$ *should be affine with respect to the parameters* μ*, i.e.,*

$$A_{11}(\mu) = \sum_j \Theta_j(\mu) A_{11}^j, \tag{8.17}$$

where $\Theta_j(\mu)$ *are scalar functions of* μ *and* A_{11}^j *are matrices independent of the parameter* μ*.*

Remark 8.7. *If Assumption 8.6 is not satisfied, the operators in* (8.16) *can be approximated by an affine expression by using EIM [17].*

8.3 Reduced Order Model Formulation

The RB method targets parameterized problems requiring repeated evaluations (many-query analysis) or (faster than) real-time simulations (real-time analysis). It typically consists of an offline and an online stage. During the offline stage, an RB space Φ is generated and several parameter independent operators related to system (8.16) are evaluated to be used in the online stage. In the online stage, the reduced solution can be obtained much faster than the one with a classical (e.g., FV) numerical simulation with a reasonable accuracy.

In this paper, we propose an additional stage, called "intermediate" stage, necessary to sufficiently handle the internal nonlinearities of system (8.16) (i.e., nonlinearities due to the discontinuous diameter of the drillstring and the annulus). In this section, we explain the three mentioned stages of the method: offline, intermediate, and online stage. The requirements of each stage are explained within each section. Finally, the computational decomposition of these stages is illustrated.

Remark 8.8. *In this study, during the offline stage, we propose to consider a reference domain (constant cross-sectional area along the well) excluding nonlinearities in the internal domain and including the discontinuous cross-sectional*

area in the intermediate stage to be used in the online stage, which is further explained in the sequel.

8.3.1 Offline stage

During the offline stage, a set of N RB functions is computed. Although N is typically much smaller than the degrees of freedom of classical numerical scheme (e.g., FV), this stage can be computationally expensive, since the RB functions are obtained via Proper Orthogonal Decomposition (POD) on the solutions of (8.16) for a specific set of parameter values. The selection of these parameter values is performed by the Greedy algorithm. The entire procedure is called POD-Greedy [89] and is explained in Section 8.3.1.3.

8.3.1.1 Proper orthogonal decomposition

Let assume that the solution of (8.16) for an arbitrary parameter value μ has been computed. POD extracts the most energetic modes of the snapshots which represent the set of RB functions denoted by ϕ. Mathematically speaking, POD solves a minimization problem as follows [33]

$$\min_{\phi \subset H} \left\| \boldsymbol{U} - \phi \phi^T \boldsymbol{U} \right\|_F^2, \tag{8.18}$$

where $\boldsymbol{U} = [\boldsymbol{U}^0,\ \boldsymbol{U}^1,\ \cdots,\ \boldsymbol{U}^{N_t}]$ with $\boldsymbol{U}^n \in \mathbb{R}^{N_\delta}$ is the numerical solution at all spatial grid points at time instant n, $\mathcal{H}$ is the Hilbert space and $\|\cdot\|_F$ is the Frobenius norm. The POD algorithm (to be used in a Greedy fashion later) is summarized in Algorithm 10, where ϕ is computed based on the discrete field variables $\boldsymbol{U}$ and the number of desired RB vectors. In Algorithm 10, the number of POD modes is either given by the user or determined by the decay of singular values. As we use this algorithm in a Greedy fashion, we show POD algorithm for the former option.

Before discussing the POD-Greedy approach, we discuss the incorporation of the nonlinear boundary conditions.

Algorithm 10: POD algorithm, $\text{POD}(\boldsymbol{U}, n_{\text{POD}})$

Input: Snapshots $\boldsymbol{U}(\mu) \in \mathbb{R}^{N_\delta \times N_t}$, number of basis vectors n_{POD}
Output: $\phi \in \mathbb{R}^{N_\delta \times n_{\text{POD}}}$

1 Perform a Singular Value Decomposition on the snapshots, $\boldsymbol{U} = U_{\text{SVD}} S V$
2 $\phi = U_{\text{SVD}}(:, 1 : n_{\text{POD}})$ is the first n_{POD} vectors of the left singular vectors U_{SVD}.

8.3.1.2 Nonlinearities at the boundaries

To handle the nonlinearities at the boundaries, the following RB ansatz is used by lifting the reduced solution [9]

$$\begin{aligned} \hat{\boldsymbol{U}}^n(\mu) &= \boldsymbol{U}_B^n + \sum_{i=1}^{N} a_i^n(\mu)\phi_i, \\ \phi_i|_{x=\{0,L\}} &= 0, \end{aligned} \tag{8.19}$$

where U_B^n is a vector that enables the RB solution $\hat{\boldsymbol{U}}^n(\mu)$ to satisfy the boundary conditions ($\hat{\boldsymbol{U}}$ can be either $\hat{\boldsymbol{\rho}}$ or $\hat{\boldsymbol{m}}$). In other words, the vector $\boldsymbol{U}_B^n$ encodes the exact satisfaction of the boundary conditions. The RB vectors $\Phi = \{\phi_i,\ i = 1, \cdots, N\}$ are the set of N RB functions evaluated at the discrete spatial points, which are obtained by the POD (Algorithm 10) applied to modified snapshots as below. To incorporate the boundary conditions correctly, ϕ_i should vanish at the location of boundaries, see (8.19). To do so, we modify the snapshots as below (step 5 of Algorithm 11):

$$\bar{U}^n(\mu) = U^n(\mu) - U_B^n. \tag{8.20}$$

Since $\bar{U}^n$ vanishes at the boundaries, applying POD to this snapshots yields basis vectors that vanish at the boundaries as well.

Finally, $a_i, i = \{1, 2, \cdots, N\}$, which is the new state variable of the ROM, is the modal coordinate corresponding to ϕ_i. The governing dynamics of the modal coordinates are explained in Section 8.3.3.

The vector $\boldsymbol{U}_B^n$ should be exact at the boundaries and can be an interpolation in the internal domain. For instance, for the density variable, we set

$$\boldsymbol{\rho}_B^n = \left(\mathbf{1} - \frac{X}{N_\delta + 1}\right)\rho_0^n + \frac{X}{N_\delta + 1}\rho_{N_\delta+1}^n, \tag{8.21}$$

where $X = [1 \cdots N_\delta]^T \in \mathbb{R}^{N_\delta}$ and $\mathbf{1} \in \mathbb{R}^{N_\delta}$ is a vector of ones. This expression at the boundaries leads to $\boldsymbol{\rho}_B^n|_{x=0} = \rho_0^n$ and $\boldsymbol{\rho}_B^n|_{x=L} = \rho_{N_\delta+1}^n$, which satisfies the requirements for the new ansatz (8.19). A similar expression can be used for $\boldsymbol{m}_B^n$.

Remark 8.9. *A similar lifting approach can be applied to 2D and 3D cases. The modified snapshots and correspondingly the RB functions then should vanish at the boundary lines or surfaces.*

8.3.1.3 POD-Greedy approach

The POD-Greedy Algorithm is summarized in Algorithm 11. To enable the implementation of Algorithm 11, we need to introduce a finite parameter domain $\mathcal{D}_h \subset \mathcal{D}$ which serves as a computational surrogate for $\mathcal{D}$ (which is the continuous

parameter domain). We assume that the set of first i parameter values have been selected and the first set of RB vectors in Φ has been computed. The next parameter value μ_{i+1} is the one corresponding to the worst approximated RB solution of (8.22) among all the members of the discrete parameter domain $\mathcal{D}_h$ (which is determined in step 10 of Algorithm 11). For this parameter value, we solve the full-order model (8.15) and gather the snapshots of density and momentum. The worst approximation is determined by evaluating the output error between the RB solution of system (8.22) and the full order solution of (8.16). An error bound or an error estimate can be useful to drastically alleviate the computational effort of the error exploration among the domain $\mathcal{D}_h$ [9] (if the error estimate can be computed at low computational cost, i.e., the computation of the error estimate does not scale with the dimension of the full-order model). Other empirical error estimates can also be useful for nonlinear problems, such as those in [8], [85]. In this study, however, during the offline stage, the actual error in the output is considered as the measure to select the next parameter. Using the actual error during the Greedy algorithm is called the strong Greedy approach [32] and it generally is computationally expensive. It is known that the Greedy approach has a higher convergence rate compared to POD and nested POD algorithms [84]. Moreover, saving all snapshots for all members of the parameter domain for all time steps induces stringent memory requirements while for the strong Greedy algorithm, we only need to save the output of the system. The POD-strong Greedy algorithm terminates when the maximum error is smaller than a given tolerance or the number of RB vectors is equal to a given number of RB vectors.

Remark 8.10. *The POD-strong Greedy is not computationally feasible in general, but the goal of the paper is to show that it is possible to use a reduced model to compute accurate solutions, not accounting for an effective offline stage (i.e., using weak greedy approach).*

By inserting the ansatz (8.19) into (8.15), applying a standard Galerkin projection and taking into account the orthonormality of Φ, a reduced-order model is obtained as below:

$$\hat{\Sigma}: \begin{cases} \hat{\Sigma}_{lin}: \begin{cases} a_\rho^{n+1} = \hat{A}_{11} a_\rho^n + \hat{A}_{12} a_m^n + \hat{B}_1 \hat{V}^n - \hat{F}_1 \hat{V}^{n+1}, \\ a_m^{n+1} = \hat{A}_{21} a_\rho^n + \hat{A}_{22} a_m^n + \hat{B}_2 \hat{V}^n - \hat{F}_2 \hat{V}^{n+1}, \\ \hat{y}^n = CC_0 \begin{bmatrix} \rho_0^n & m_0^n \end{bmatrix}^T + CC_{N_\delta} \begin{bmatrix} \rho_{N_\delta+1}^n & m_{N_\delta+1}^n \end{bmatrix}^T + \\ \qquad CC_\phi \begin{bmatrix} a_\rho^{n\,T} & a_m^{n\,T} \end{bmatrix}^T, \\ \hat{y}_V^n = C_V C_0 \begin{bmatrix} \rho_0^n & m_0^n \end{bmatrix}^T + C_V C_{N_\delta} \begin{bmatrix} \rho_{N_\delta+1}^n & m_{N_\delta+1}^n \end{bmatrix}^T + \\ \qquad C_V C_\phi \begin{bmatrix} a_\rho^n & a_m^n \end{bmatrix}^T, \end{cases} \\ \Sigma_{nl}: \hat{V}^n = h(\hat{y}_V^{n-1}, \hat{V}^{n-1}, u_c^{n-1}), \end{cases} \tag{8.22}$$

where the hatted operators $\hat{(\cdot)}$ are obtained by Galerkin projection of the full-order operators on the space spanned by the orthogonal RB vectors Φ_ρ and Φ_m,

and

$$C_0 = \begin{bmatrix} \mathbf{1} - \dfrac{X}{N_\delta + 1} & \mathbf{0}_{N_\delta \times 1} \\ \mathbf{0}_{N_\delta \times 1} & \mathbf{1} - \dfrac{X}{N_\delta + 1} \end{bmatrix}, \quad C_{N_\delta} = \begin{bmatrix} \dfrac{X}{N_\delta + 1} & \mathbf{0}_{N_\delta \times 1} \\ \mathbf{0}_{N_\delta \times 1} & \dfrac{X}{N_\delta + 1} \end{bmatrix}, \tag{8.23}$$
$$C_\phi = \begin{bmatrix} \Phi_\rho & \mathbf{0}_{N_\delta \times N_m} \\ \mathbf{0}_{N_\delta \times N_\rho} & \Phi_m \end{bmatrix}.$$

Operators such as $CC_0 \in \mathbb{R}^{N_y \times 2}$ are of low dimensions and can be defined in the offline stage. For more information, refer to [9]. The initial conditions for the system are also obtained by a Galekin projection of the initial condition; for instance, $a_\rho^0 = \Phi_\rho^T \boldsymbol{\rho}^0$. It should be recalled that in the offline stage, the diameter is considered constant along each pipe.

Note that in step 8 of Algorithm 11, the POD is performed on the interpolation error instead of the modified snapshots to consistently add new information at each step of the POD-strong Greedy procedure. To apply POD (step 8 of Algorithm 11), a singular value decomposition (SVD) is performed on the modified snapshots as demonstrated in Algorithm 10. The set of RB vectors is enriched with the first left singular vector. For more information, refer to [89]. At this point, if the maximum number of RB vectors is not yet reached or the expected maximum error among members of the discrete parameter domain $\mathcal{D}_h$ is not achieved, we proceed in the same way to compute μ_{i+2}.

8.3.2 Intermediate stage

To incorporate the effect of the spatially-dependent diameters, an extra stage is added to the standard RB method, which is an intermediate stage coming in between the offline and the online stage. Although the computation of this step scales with the dimension of the full-order mode, this stage has to be performed when the diameter profile is changed. If this stage is not included, whenever the diameter profile is changed, the entire offline stage should be repeated. In the intermediate stage, operators related to the diameter are defined and the RB vectors are enriched with some local basis vectors with compact support. Then, the enriched set of basis vectors is orthonormalized. This is explained in Algorithm 12 and in the following section.

8.3.2.1 Localized enrichment

For capturing the internal nonlinearities (discontinuous diameter inside the domain), a local enrichment of RB vectors is introduced here, which also shares similarities with the EIM approach for the distributed nonlinearities. The generic EIM procedure consists of an offline stage to generate the collateral basis functions and detect some interpolation points [17]. Then, the distributed nonlinearities are approximated by linear interpolation of these collateral basis functions.

Algorithm 11: POD–strong Greedy

Input: $\mathcal{D}_h, N, \mu_1 \in \mathcal{D}_h$ (arbitrary), ϵ_y
Output: Φ

Set $\Phi = \{\}, k = 1$, and $\mathcal{E}_k > \epsilon_y$,
Solve (8.16) $\forall \mu \in \mathcal{D}_h$ to compute $y(\mu)$,
while $\mathcal{E}_k > \epsilon_y$ **do**
  Find $U^n(\mu_k)$ as the solution of (8.16), $n = \{1, \cdots, N_t\}$,
  $\bar{U}^n(\mu_k) = U^n(\mu_k) - U_B^n$,
  $\bar{\boldsymbol{U}}(\mu_k) = [\bar{U}^0(\mu_k), \cdots, \bar{U}^{N_t}(\mu_k)]$,
  $\bar{\boldsymbol{e}}^n(\mu_k) = \bar{\boldsymbol{U}}(\mu_k) - \Phi\Phi^T\bar{\boldsymbol{U}}(\mu_k)$,
  $\Phi \leftarrow \text{orth}\,\{\Phi \cup \text{POD}(\bar{\boldsymbol{e}}^n, 1)\}$ which orthonormalizes the enriched set of basis function (see Algorithm 10 for POD),
  Solve (8.22) $\forall \mu \in \mathcal{D}_h$ to compute $\hat{y}(\mu)$,
  $\mu_{k+1} = \arg\max_{\mu \in \mathcal{D}_h} \frac{1}{\sqrt{N_t}} \sqrt{\sum_{n=0}^{N_t} \|y^n(\mu) - \hat{y}^n(\mu)\|_2^2}$ and
  $\mathcal{E}_{k+1} = \frac{1}{\sqrt{N_t}} \sqrt{\sum_{n=0}^{N_t} \|y^n(\mu_{k+1}) - \hat{y}^n(\mu_{k+1})\|_2^2}$
  $k + k + 1$

The coefficients of the interpolation are computed such that the interpolation is exact at the interpolation points. Generally, EIM is tailored for distributed nonlinearities, not for local nonlinearities. Here, by a similar formulation of the problem suited to EIM (see (8.10),(8.11) where operator ξ appears in the equations), the nonlinearity computation is decoupled from the linear subsystem.

In the current problem setting, the MOR method is supposed to be flexible and deal with any generic cross-sectional area of the domain. The nonlinearities do not enter the problem until the area discontinuity is introduced by the user in the intermediate stage. Here, unlike the procedure in the EIM approach, the interpolation points are already known and selected at the locations of area discontinuity. Moreover, the local basis vectors are the same as ξ introduced in (8.10) and (8.11).

The locations of area discontinuities can be detected as $x_m^* = \{x_{k+1/2} | D_h(x_k) \neq D_h(x_{k+1})\}$. Therefore, the interpolation points are defined as $x_m = \{x_k, x_{k+1} | D_h(x_k) \neq D_h(x_{k+1}), k = 1, \cdots, N_\delta - 1\}$. Then, the local basis vectors are defined as the collocation of canonical vectors corresponding to the locations in x_m, i.e., $\xi = \{e_k, e_{k+1} | D_h(x_k) \neq D_h(x_{k+1}), k = 1, \cdots, N_\delta - 1\}$, where e_k is the canonical vector with the nonzero entity at k-th element.

If no area variation occurs, the local basis vector is a null vector. The local nonlinearities of area variation, $\mathcal{J} \in \mathbb{R}^{N_\delta}$ for instance for the density, can be

approximated by

$$\mathcal{J} = \xi \rho_d^{+,n}, \tag{8.24}$$

where $\rho_d^{+,n}$ is calculated at the locations of x_m as in (8.13). In other words, $\mathcal{J}$ is a vector with nonzero elements at the locations of area discontinuities and zero elements elsewhere. In this way, we approximate the local nonlinearities by an affine combination of pre-computed basis vectors ξ and local values of the field variables. This also enables the exact interpolation at the location of area variation.

At this stage, RB vectors are enriched with local basis vectors to take into account the effect of area variations and also the reduced operators are re-defined. To account for the discontinuities in the area, the set of continuous RB vectors, obtained during the offline stage, are enriched by local discontinuous basis vectors with compact support. For instance, for an area variation at the discrete point $x_{i+1/2}$, one of the two basis vectors shown in Figure 8.4 is added to the set of RB vectors and then the orthogonalization is applied. These discontinuous basis vectors account for the discontinuities generated in the solution due to the discontinuous diameter and these should also vanish at the boundaries. This is explained in Algorithm 12. The motivation behind choosing the discontinuous basis vectors shown in Figure 8.4 is explained in the following remark.

Remark 8.11. *In the case of a discontinuous diameter, the snapshots in $\boldsymbol{U}$, and therefore the RB vectors in ϕ, are also discontinuous. Since, in the offline stage, local discontinuities are ignored, we should account for these nonlinearities in the intermediate stage. Assume that we have the steady-state solution as the initial condition in the online stage. Then, we should enrich the current set of RB vectors such that the following cost function:*

$$\left\| \bar{\boldsymbol{U}}_{ss} - \phi \phi^T \bar{\boldsymbol{U}}_{ss} \right\|_F^2, \tag{8.25}$$

is minimized with respect to ϕ, where $\bar{\boldsymbol{U}}_{ss}$ is the modified steady-state solution after subtracting the effect of the boundary conditions similar to (8.20)*. For an arbitrary set of parameters, this modified steady-state solution for the momentum for example resembles Figure 8.3. The optimum way to minimize this objective function is to enrich the set of RB vectors with the modified steady-state solution. However, this solution is not known beforehand. Therefore, local basis vectors should be added to the set of RB vectors to reproduce at least the steady-state solution. To this end, the set of basis vectors introduced in Figure 8.4 are added. These basis vectors are designed to take into account the discontinuities in the solution, while the continuous RB vectors determine the solution between two adjacent discontinuities.*

Remark 8.12. *The selection of the local basis vectors is problem specific. For 2D and 3D cases, based on the difference between the solution in the absence*

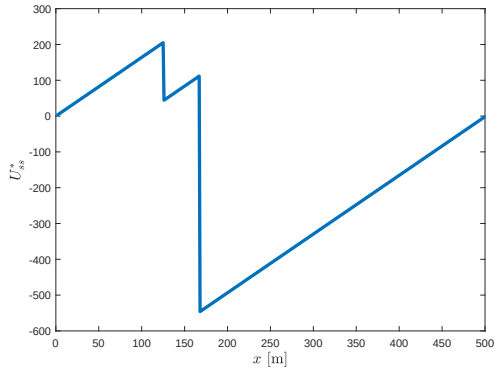

Figure 8.3: Modified steady-state solution for momentum $\bar{U}_{ss}$ for an arbitrary set of parameters.

and presence of the local nonlinearities, some basis surfaces or volumes should be defined accordingly.

8.3.3 Online stage

After obtaining the enriched basis vectors $\bar{\Phi}$ from Algorithm 12, the linear part of system (8.16) is reduced from Σ_{lin} to $\hat{\Sigma}_{lin}$ by a standard Galerkin projection into the space spanned by $\bar{\Phi}$ while the local nonlinear dynamics Σ_{nl} (area discontinuities and boundary equations) remain exactly as in the original model (8.16). A schematic view of the reduced model is shown in Figure 8.5. This figure illustrates that auxiliary outputs that are necessary for computing the output of Σ_{nl}, y_W^n and y_V^n, are provided via a feedback interconnection. Then,

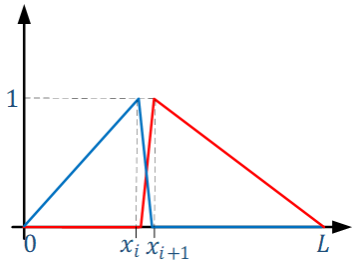

Figure 8.4: The discontinuous basis vector ϕ_i^d to take into account the discontinuous area at $x = x_{i+1/2}$.

Algorithm 12: Intermediate stage including local enrichment

Input: Continuous RB vectors Φ, diameter profile d, D_{out}, FV operators as in (8.10),(8.11)

Output: Enriched set of basis vectors $\bar{\Phi}$

Calculate the hydraulics diameter D_h for each pipe,
$x_m = \{\}, \zeta = \{\}, \Phi^d = \{\}$
for $k \leftarrow 1$ *to* $N_\delta - 1$ **do**
    if $D_h(x_k) \neq D_h(x_{k+1})$ **then**
        $x_m = [x_m, \{x_k, x_{k+1}\}]$
        $\zeta = [\zeta, \{e_k, e_{k+1}\}]$
        Compute the discontinuous basis vectrors $\Phi^d = [\Phi^d, \phi_k^d]$ as explained in Figure 8.4,
Enrich $\bar{\Phi} = [\Phi, \Phi^d]$ and perform an orthonormalization based on the Gram-Schmidt procedure [84].

the outputs from Σ_{nl} are fed into the reduced linear system $\hat{\Sigma}_{lin}$ to incorporate the effect of the boundary conditions and area variation.

Similar to (8.22) and by embedding the effect of discontinuous diameter, we

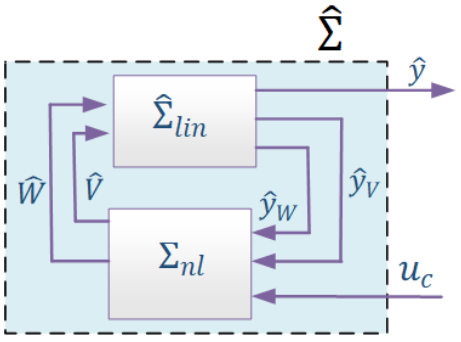

Figure 8.5: The schematic representation of the reduced order model.

have

$$\hat{\Sigma}:\begin{cases}\hat{\Sigma}_{lin}:\begin{cases}a_\rho^{n+1}=\hat{A}_{11}a_\rho^n+\hat{A}_{12}a_m^n+\hat{B}_1\hat{V}^n+\hat{B}_1^*\hat{W}^n-\hat{F}_1\hat{V}^{n+1},\\ a_m^{n+1}=\hat{A}_{21}a_\rho^n+\hat{A}_{22}a_m^n+\hat{B}_2\hat{V}^n+\hat{B}_2^*\hat{W}^n-\hat{F}_2\hat{V}^{n+1},\\ \hat{y}^n=CC_0\begin{bmatrix}\rho_0^n & m_0^n\end{bmatrix}^T+CC_{N_\delta}\begin{bmatrix}\rho_{N_\delta+1}^n & m_{N_\delta+1}^n\end{bmatrix}^T+\\ \qquad CC_\phi\begin{bmatrix}a_\rho^{nT} & a_m^{n\,T}\end{bmatrix}^T,\\ \hat{y}_V^n=C_VC_0\begin{bmatrix}\rho_0^n & m_0^n\end{bmatrix}^T+C_VC_{N_\delta}\begin{bmatrix}\rho_{N_\delta+1}^n & m_{N_\delta+1}^n\end{bmatrix}^T+\\ \qquad C_VC_\phi\begin{bmatrix}a_\rho^n & a_m^n\end{bmatrix}^T,\\ \hat{y}_W^n=C_WC_0\begin{bmatrix}\rho_0^n & m_0^n\end{bmatrix}^T+C_WC_{N_\delta}\begin{bmatrix}\rho_{N_\delta+1}^n & m_{N_\delta+1}^n\end{bmatrix}^T+\\ \qquad C_WC_\phi\begin{bmatrix}a_\rho^{nT} & a_m^{n\,T}\end{bmatrix}^T,\end{cases}\\ \Sigma_{nl}:\begin{cases}\hat{V}^n=h(\hat{y}_V^{n-1},\hat{V}^{n-1},u_c^{n-1}),\\ \hat{W}^n=\mathcal{G}(\hat{y}_W^n),\end{cases}\end{cases}\tag{8.26}$$

where the effect of discontinuous diameter is incorporated by operators such as $\hat{B}_1^*$ and variables $\hat{W}^n$. Recall that operators such as $CC_0 \in \mathbb{R}^{N_y\times 2}$ are of low dimension and are defined in the offline stage. Operators such as C_WC_0 are defined in the intermediate stage due to their dependencies over the locations of area variations. In the next section, the notion of offline-online decomposition, including the intermediate stage taking into account the area variation, is explained.

8.3.4 Offline-intermediate-online decomposition

In the offline stage, no area variation is considered. It means that B_i^* in (8.16) is zero during the offline stage and the RB vectors are generated without considering the discontinuous diameter. The affine property of the model with respect to the parameters (Assumption 8.6) leads to the offline-online decomposition of the computational costs [89].

Since changing the locations of the diameter discontinuity changes the structure of the wave propagation inside the domain, the diameter profile of the drillstring and the annulus is not a function of the spatial coordinate x in the offline stage. If the locations of area variation change from one simulation to another in the offline stage, the snapshots and, therefore, the RB vectors experience discontinuity at different locations. Then, reconstructing the solution from these RB vectors leads to a stair-case problem. This problem also arises in many other applications [180]. To avoid this issue, no area discontinuity is considered in the offline stage. It means that the snapshots and therefore the RB vectors, which are obtained by performing a POD on the modified snapshots, are continuous. To obtain the parameters corresponding to the worst RB approximation during the Greedy algorithm, the actual error (as in line 10 of Algorithm 12) or an estimate of the error indicated in [9], [182] can be used.

However, as diameters are spatially dependent and should be set by the user before running the online simulation, we define an intermediate stage between the offline and the online stages. In this intermediate stage, the diameter is defined by the user. Also, the operators in (8.26) that are dependent on this diameter are defined in this stage. Although this stage involves computation scaled by the actual degrees of freedom of the original system, this is done only once per each change in the diameter profile. If this stage is absent, the entire expensive offline stage should be repeated for each change in the diameter profile, which would be far more expensive computationally.

In the online stage, by changing any of the varying parameters as well as the diameter profile, the solution can be obtained accurately in a computationally cheap fashion. However, if the locations of the area variation vary, the computations in the intermediate stage should be repeated while the information computed during the offline stage are still useful and valid for the new scenario.

8.4 Numerical results

In this section, the numerical results are divided into two categories. First, results are obtained by using the same external control inputs (as u_c^n in (8.16) and (8.26)) during the offline and online stages (both stages simulate one drilling scenario). Second, without varying the offline stage, the external control inputs in the online stage are changed to simulate another scenario in drilling. The two test cases mentioned above are relevant to two real drilling scenarios, which are explained below.

Connection: The action of adding a stand of drillpipe to the drillstring to continue drilling deeper is called connection. To do so, the pump is stopped and choke is partially closed to maintain the downhole pressure in a specific range. A new drillpipe is attached to the top of the drillstring to elongate the drillstring. After this, the pump is turned on and the choke is opened up and the drilling procedure resumes [128], [155].

Choke plugging: As the mud along with the cuttings travels up along the annulus, the mixture passes through the choke manifold. When some of the cuttings get stuck in the choke, due to their weight and change of flow direction, the flow area of the choke decreases if no action is taken by the operator. Any change in the choke area might affect the choke pressure and consequently the bottomhole pressure. The contingency of having some cuttings getting stuck in the choke yielding in the unplanned reduction of the flow area in the choke manifold is called choke plugging [128], [155].

8.4.1 Training for connection scenario

In this study, the varying parameters are the mud viscosity γ, the well length L, the well inclination θ, the speed of sound c_l, the reference density of the

mud ρ_{0s}, and the drillstring diameter d and wellbore diameter D_{out}. The initial condition of the system (8.16) is its steady-state solution corresponding to a specific parameters of interest or the parameters selected during the Greedy algorithm. Then, by changing the inputs to the system (q_p and z_c), the dynamics of the system are excited. In other words, the dynamic solution is a perturbation of the steady-state solution.

Table 8.2 contains the range of the varying parameters. Due to memory constraints, only 128 equispaced samples are considered in the discrete parameter domain. There are also some fixed parameters from one simulation to another, which are listed in Table 8.3. In the last row of Table 8.2, the set of parameters selected for the online simulation μ^o is reported, which does not lie in the discrete parameter domain $\mathcal{D}_h$. The diameters used in the online stage are shown in Figure 8.6. The number of grid cells is $N_\delta = 500$ and the fixed time horizon is $T = 50$ s. In each simulation, the time-step Δt is changed such that $c_l \Delta t / \Delta x = 0.8$, which is the CFL number [113] associated to each simulation. The output of interest is the pressure in the last 10 percent of the domain in the annulus (i.e., downhole in the well) where the pressure needs to be calculated accurately. The boundary conditions variability over time during offline stage are shown in Figure 8.7. This type of input simulates the connection scenario commonly performed in practice.

Remark 8.13. *As the length of the spatial domain L varies, we scale the RB vectors on a unit length and scale back for any new given length.*

Remark 8.14. *The interconnected model $\hat{\Sigma}$ may get unstable due to the interconnection of the area discontinuity and the reduced linear subsystem $\hat{\Sigma}_{lin}$. One way to reduce the gain of this feedback interconnection is to reduce the CFL*

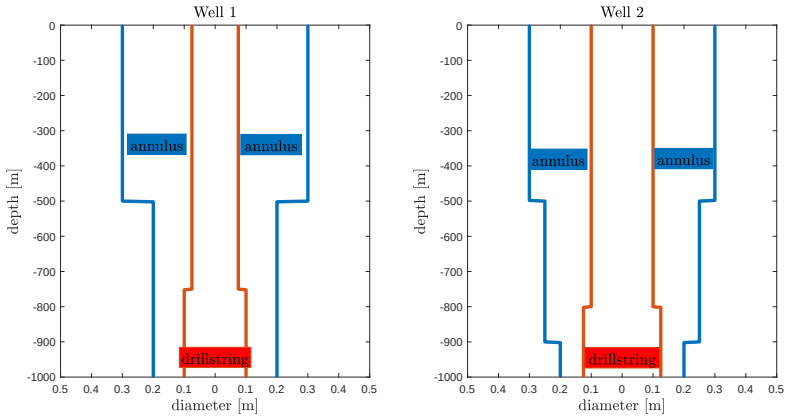

Figure 8.6: Diameter configuration for the online test cases.

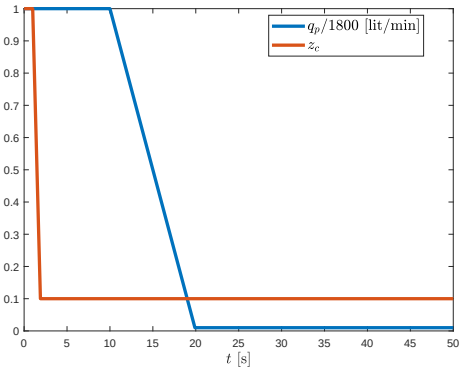

Figure 8.7: Input signals for the connection scenario.

Table 8.2: Range of varying parameters.

parameter	ρ_{0_s} [Kg/m^3]	θ [deg]	c_l [m/s]	L [m]	γ [Pa.s]	d [m]	D_{out} [m]
minimum	800	60	1300	800	0.004	0.05	0.15
maximum	1200	90	1500	1200	0.04	0.125	0.4
Online μ^o	1000	75	1400	1000	0.02	Well 1 Figure 8.6	Well 1 Figure 8.6

number (for CFL= 0.8, we did not encounter instability in the simulations).

The maximum of the error indicator computed in the POD-strong Greedy algorithm (line 10 of Algorithm 11) during the offline stage among all the members of the training set $\mathcal{D}_h$ is plotted in Figure 8.8 where no area variation is imposed.

A comparison of the pressure profile between the FV and RB solutions is shown in Figure 8.9 at four different time instants during the online stage where the area discontinuity of Well 1 is considered. It should be noted due to the low compressibility of the drilling mud, that the pressure does not vary significantly over the discontinuous area and its discontinuties are not visible in Figure 8.9. The maximum error in the downhole pressure approximation is less than 1 bar. The dimension of the full-order model is 2000 while the dimension of the reduced-order model is 80 plus 6 basis vectors added in the intermediate stage

Table 8.3: Fixed parameters for the simulations.

parameter	g [m/s^2]	p_{atm} [Pa]	K_c	A_N [in^2]	C_D
value	9.81	10^5	0.00285	1.1562	0.8

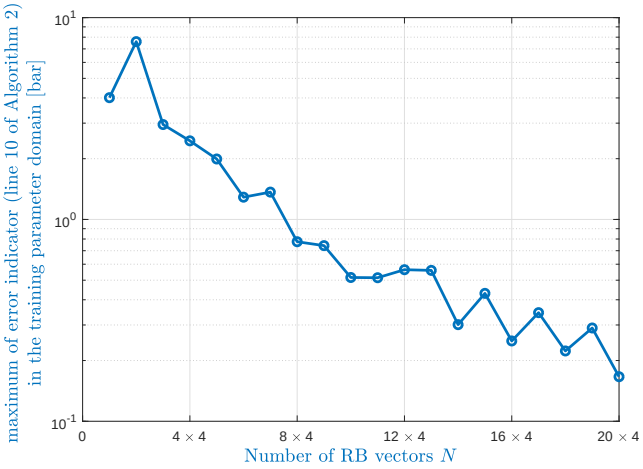

Figure 8.8: Error in approximating pressure profile by the RB solution during the offline stage with spatially constant diameters.

to capture the area discontinuity (one diameter discontinuity in the drillstring for both momentum and density and two in the annulus for both momentum and density). To decrease the approximation error, the number of RB vectors should be increased. However, increasing the number of RB vectors does not resolve the so-called Gibbs phenomenon due to the hyperbolic nature of the system under investigation [66]. In addition to the pressure, Figures 8.10 and 8.11 show the comparison of the velocity profiles between the FV and the RB solution. To show the applicability of the method, the RB results are also compared with the POD results applied to the snapshots of the full-order model with discontinuity. Apparently, POD works better; however, when the location of the area discontinuity varies, the entire offline stage should be implemented again. The aforementioned Gibbs phenomenon is more visible in Figures 8.10 and 8.11. However, in the MPD context, the approximation of pressure is much more important than approximating the velocity. The inaccuracies observed in Figures 8.10, 8.11 are due to the hyperbolic nature of the system, which is beyond the scope of this paper. It can be inferred from the results that the proposed methodology for capturing the local nonlinearities works efficiently and accurately and can be a promising approach to speed up real-life drilling operations.

The time-wise comparison of the bottomhole pressure computed by the full-order and reduced-order model together with the approximation error is depicted in Figure 8.12. Apparent from this figure, the wave reflection occurred due to the area variation has been captured accurately by the reduced-order model (see

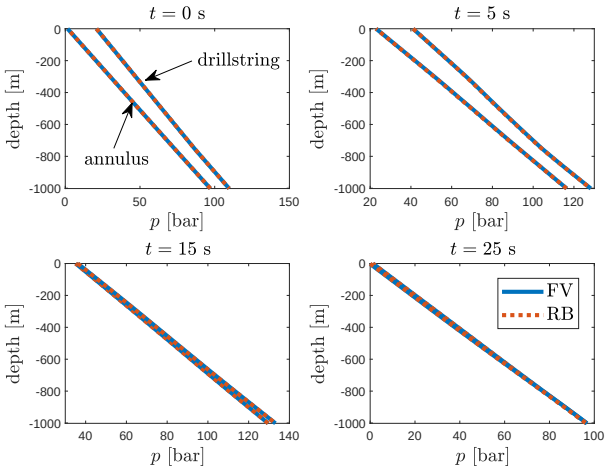

Figure 8.9: Comparison between pressure profiles obtained by the FV and RB solution for the online parameter μ^o as in Table 8.2, connection scenario.

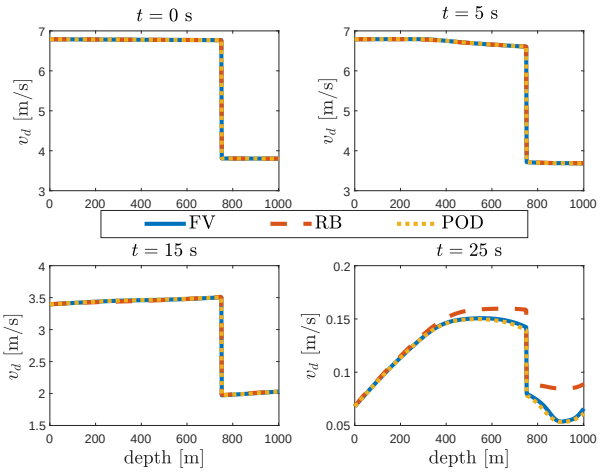

Figure 8.10: Comparison between velocity profiles obtained by the FV solution, the RB solution and direct POD applied to the discontinuous solution in the drillstring for the online parameter μ^o as in Table 8.2, connection scenario.

the step-wise increase in the downhole pressure over time, which is due to the wave reflection at the area discontinuities and boundaries). This phenomenon

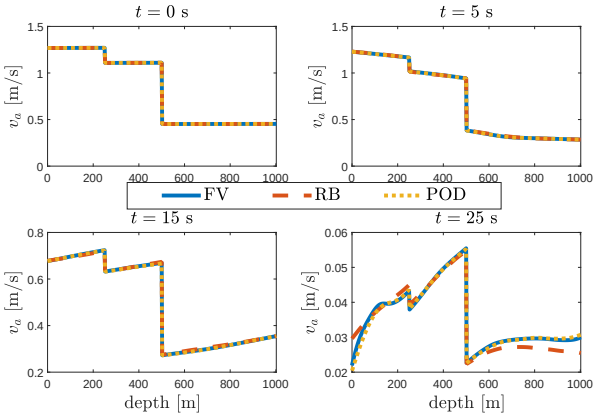

Figure 8.11: Comparison between velocity profiles obtained by the FV solution, the RB solution and direct POD applied to the discontinuous solution in the annulus for the online parameter μ^o as in Table 8.2, connection scenario.

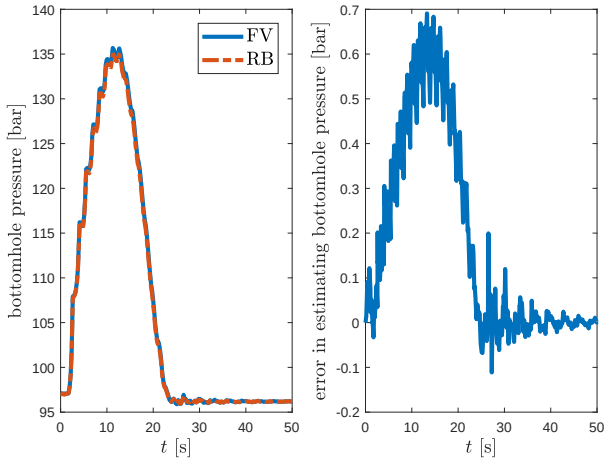

Figure 8.12: Time-wise comparison of the bottomhole pressure between the FV and RB solution for the online parameter μ^o as in Table 8.2, connection scenario.

significantly changes the dominant resonance frequency of the system, which has to be captured by the reduced-order model. The time-wise change of the velocity at the choke together with the approximation error can be seen in Figure 8.13, which further confirms the accuracy of the reduced-order model.

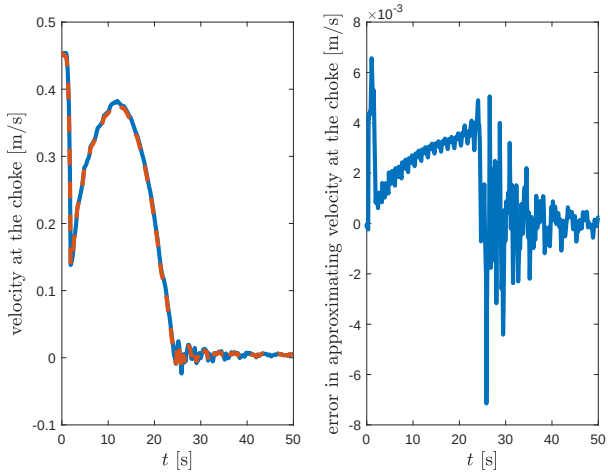

Figure 8.13: Time-wise comparison of the velocity at the choke between the FV and RB solution for the online parameter μ^o as in Table 8.2, connection scenario.

The CPU time allocated to solve the offline stage, the intermediate stage, the online stage and the full-order model together with the speedups are reported in Table 8.4 by using different numbers of RB functions. Number of RB functions for each state is the same and therefore the total number of RB functions is multiplied by four. Since there is one diameter discontinuity in the drillstring and two diameter discontinuities in the annulus, six extra local basis functions are added to enrich the previous set of RB functions. The speedups are obtained by comparing the CPU time of solving the full-order model and the CPU time of solving both the intermediate and the online stages. This shows that by using this approach, a speedup in the computational time of obtaining the reduced-order solution (intermediate and online stages together) up to 70 can be achieved while maintaining a high accuracy in the output of interest. The large CPU time of the offline stage (around 17 hours) is due to the high dimension of the parameter domain and the requirement to solve the full-order model (without area discontinuity) 128 times.

To further evaluate the performance of the proposed method, five random points in the parameter domain (outside the training parameter domain) are selected and new simulations for the connection scenario in the two geometries shown in Figure 8.6 are computed and compared with the full-order solutions. In Table 8.5, the following relative error indicator for the different parameter

samples is reported:

$$e = \sqrt{\frac{\sum_{n=0}^{N_t} \|y^n(\mu) - \hat{y}^n(\mu)\|_2^2}{\sum_{n=0}^{N_t} \|y^n(\mu)\|_2^2}}. \tag{8.27}$$

The reported relative errors confirm the accuracy of the approach for different well geometry configuration and different parameter samples. The results also reveal that the relative error is higher if the number of area discontinuities is higher due to the many discontinuities in the state variables.

8.4.2 Choke plugging scenario

In the previous section, both the offline stage and the online stage are computed with the same external control inputs; only the parameters of the simulation are changed. In this section, based on the offline stage performed on connection scenario, we simulate a choke plugging scenario (with different inputs compared to the offline stage) and show the results.

In a choke plugging scenario, the choke opening decreases due to cuttings getting trapped inside the choke. As there are two parallel chokes installed at the choke manifold, the plugged choke is closed and the stand-by choke is opened. Then, the personnel cleans the plugged choke. To simulate this effect, the choke opening z_c is reduced for some period of time and then increased to the previous level. Throughout all these actions, the pump flow is constant. The input signals are shown in Figure 8.14.

Table 8.4: CPU time comparison of the full-order model of MPD and its reduced version for the online parameter μ^o as in Table 8.2 (N: number of basis functions in the online stage).

Full-order			368.11 s		
Offline			16 h 46 min		
N	$1 \times 4 + 6$	$5 \times 4 + 6$	$10 \times 4 + 6$	$15 \times 4 + 6$	$20 \times 4 + 6$
Intermediate	0.066 s	0.068 s	0.068 s	0.072 s	0.075 s
Online	4.15 s	4.18 s	4.75 s	4.95 s	5.1 s
Speedup	87.3	86.6	76.4	73.3	71.1

Table 8.5: Error indicator (8.27) for different random parameter samples (outside the discrete parameter domain) in the two geometry configurations shown in Figure 8.6 tested for connection scenario.

sample number	1	2	3	4	5
well 1	0.0028	0.0033	0.0025	0.0035	0.0031
well 2	0.0079	0.0091	0.0072	0.0100	0.0090

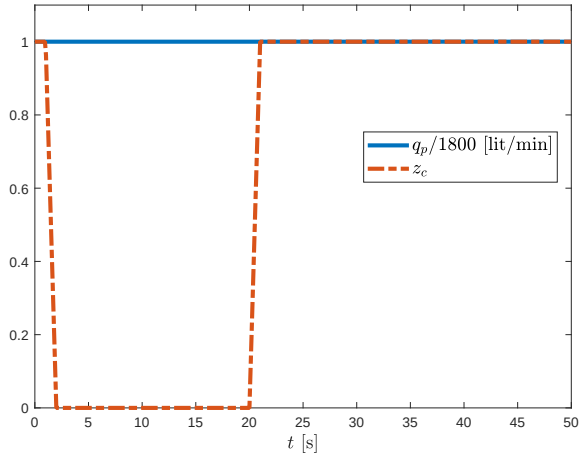

Figure 8.14: Input signals for the choke plugging scenario.

Similar to the previous test case, the time-wise comparisons of the bottom-hole pressure and the velocity at the choke together with their corresponding approximation errors are shown in Figures 8.15 and 8.16. The results confirm that, although the training (the offline stage) is performed for a connection scenario, it can simulate other drilling scenarios. Speedup in this case is similar to Table 8.4 of the previous test case.

The results presented in this section confirm that the proposed method successfully captures the effect of the local nonlinearities inside the computational domain and at the boundaries of the system during MOR.

8.5 Conclusion

In this paper, a new approach for addressing the issue of localized nonlinearities with the RB method has been proposed. These nonlinearities appear either at the boundaries of the system or inside the computational domain. As the boundary conditions for industrial systems are often nonlinear, it is vital to change the reduced basis ansatz, such as the one suggested in this paper. For the nonlinearities inside the domain, a local enrichment approach bearing similarities with EIM is incorporated and its outputs are coupled with the rest of the system dynamics. Using this approach, crucial underlying physics of the phenomena such as mass conservation at a particular point and wave reflections are preserved during MOR. Any other spatially-dependent local nonlinearities can be handled by this method. Locating pump stations and orifices in a pipeline can be seen as other applications which may benefit from the proposed approach.

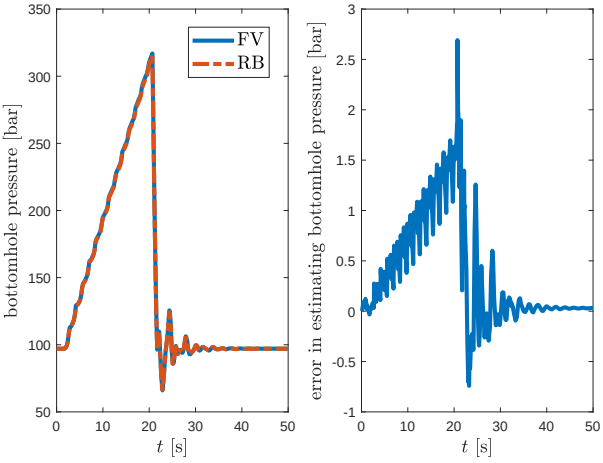

Figure 8.15: Time-wise comparison of the bottomhole pressure between the FV and RB solution for the online parameter μ^o as in Table 8.2, choke plugging scenario.

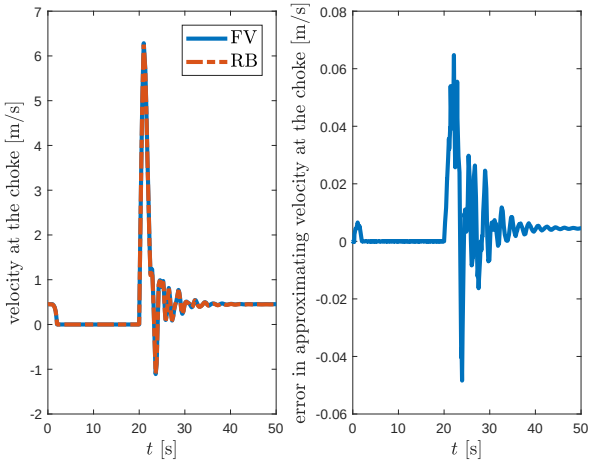

Figure 8.16: Time-wise comparison of the velocity at the choke between the FV and RB solution for the online parameter μ^o as in Table 8.2, choke plugging scenario.

The speedup obtained by using the proposed approach can expedite, for instance, optimization for well planning, real-time simulations, and the imple-

mentation of (model-predictive and optimal) control techniques.

Part III

Port-Hamiltonian Systems

The final part of the thesis provides the pH formulation of models involved in MPD. In this part, two commonly used multi-phase flow models, the TFM and DFM, are investigated and pH modelling of both is presented. As the single-phase flow also abides by pH formalism, all dynamical models involved in MPD can be reformulated with such a structure. To create a pH-MPD model, these pH models are interconnected through the boundary condition of a drilling bit. In order to preserve the total power and obtain an aggregated pH model for the MPD model, the drilling bit should also preserve power. The conditions under which this boundary constitutes a power-preserving interconnection are derived in this chapter. The pH structure can be exploited for passivity-preserving MOR techniques in future studies.

Chapter 9

Port-Hamiltonian Formulation of Two-phase Flow Models

Two-phase flows are frequently modelled and simulated using the Two-Fluid Model (TFM) and the Drift Flux Model (DFM). This paper proposes Stokes-Dirac structures with respect to which port-Hamiltonian representations for such two-phase flow models can be obtained. We introduce a non-quadratic candidate Hamiltonian function and present dissipative Hamiltonian representations for both models. We then use the structure of the corresponding formally skew-adjoint operator to derive a Stokes-Dirac structure in the scope of the two variants of multi-phase flow models. Moreover, we present a numerical counter example to demonstrate that only a special form of the DFM (without slip between the phases) can be cast in a port-Hamiltonian representation and that the DFM with the Zuber-Findlay slip conditions is not an energy consistent model for two-phase flow.

9.1 Challenges in the DFM reduction

Applying projection-based model order reduction techniques, such as the reduced basis method, to the DFM does not typically yield a stable system due to the highly nonlinear and coupled nature of the DFM. It is challenging, if not impossible, to identify a suitable basis (of reduced dimension) to approximate the solution space, irrespective of the number of basis functions used. The difficulties in the reduction of the DFM, based on the author's experience and the

This chapter is based on "H. Bansal, P. Schulze, M.H. Abbasi, H. Zwart, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***Port-Hamiltonian Formulation of Two-phase Flow Models***, Submitted to *Systems & Control Letters*, 2020".

conducted simulations, can be listed as below:

- The DFM contains, at the infinite dimensional level, strongly coupled nonlinear terms, which affect the solution dramatically;
- The primitive variables are a highly nonlinear function of conservative variables;
- Discretization techniques used for the DFM introduce even more local and distributed nonlinearities;
- Some of the nonlinearities introduced after the discretization, e.g. "max" function in (2.23), has to be applied at the interface of the two neighboring cells of the FV discretization, rendering the computational reduction technique less efficient;
- Boundary conditions are also a highly nonlinear function of conservative variables;
- The coupled nature of the DFM might require different accuracy for different conservative variables. Even considering this feature does not help to recover stability.

These features of the DFM make it challenging to construct an accurate and reduced model concurrently. Discarding some nonlinearities would not lead to a realistic discretization of the DFM. On the other hand, we could use interpolation techniques, such as Empirical Interpolation Method (EIM) [17], to tackle the nonlinearities. However, the stability of the Reduced-Order Model (ROM) is observed only when a high number interpolation points is used in the EIM approximation. Moreover, even in very limited stable test cases, the ROM goes to a non-physical steady-state solution unless a high number of EIM interpolation points is used. This will reduce the final gain in computational efficiency and sometimes even increases the required computational time compared to solving the full-order model. Therefore, we propose in this chapter a pH framework for DFM and TFM. This can serve as a stepping stone towards Model Order Reduction (MOR) while keeping essential properties of systems such as stability or passivity.

9.2 Introduction

In this paper, we develop a port-Hamiltonian (pH) formulation for modelling multi-phase flow dynamics in pipes. Multi-phase flows are important in a large range of industrial applications, such as within the oil and gas industry, chemical and process industry (including heat-pumping systems) as well as the safety analysis of nuclear power plants [5], [6], [143]. Within the oil and gas industry, such models are used for virtual drilling scenario testing [5], [6]. The multi-phase

aspect is particularly relevant in these applications in case of gas influx occurring from a reservoir.

A pH model formulation is known to provide a modular framework for multi-physics and interconnected systems [122]. The pH structure allows for non-zero energy flow through the boundary and guarantees power preservation [154]. Moreover, structure-preserving methods for discretization and the model order reduction of infinite-dimensional pH systems can preserve certain original system-theoretic properties such as stability and passivity [43], [138]. Additionally, the pH framework supports the development of control strategies [119].

In the literature, the infinite-dimensional pH structure has been exploited in several domains of science and engineering. For instance, some well-known fluid dynamical systems such as the shallow water equations [138], reactive Navier Stokes equations [11], and reaction diffusion processes [184] have already been formulated in the pH formalism. Such a representation is also prevalent in the fields of structural dynamics [119] and fluid-structure interaction [36].

Multi-phase flows are mathematically governed by conservation laws. Several conservation laws have previously been converted to pH representations [121], [177]. Some work on Hamiltonian modeling for multi-phase hydrodynamics has been done in [90]. However, (dissipative) Hamiltonian representations do not exist for the Two-Fluid Model (TFM) and the Drift Flux Model (DFM) [68]. Moreover, until now, to the best of our knowledge, pH modeling for fluid dynamics only encompasses single-phase models [181].

Matrix/operator theory for *linear* distributed parameter port-Hamiltonian systems on one-dimensional domains is owed to some pioneering works [77], [97]. The central theme of the current paper is to extend and propose modifications to the existing theory for *non-linear* distributed parameter systems. We exploit the existing theory in the scope of linear systems and arrive at new results from an operator theoretic viewpoint, including further generalizations in the scope of non-linear distributed parameter port-Hamiltonian systems.

The main contributions of this paper are as follows: *(i)* (dissipative) Hamiltonian representations of the TFM and the DFM, and *(ii)* proposition of state-dependent Stokes Dirac structures for both the TFM and the DFM along with the proof of the corresponding representation obtained in the scope of the TFM.

The paper is organized as follows. In Section 9.3, we introduce the two mathematical models governing 1-D multi-phase flow dynamics and mention under which conditions these are equivalent. The (dissipative) Hamiltonian representations of these models are presented in Section 9.4. Then, the corresponding geometric properties are discussed and proved in Section 9.5. This section also includes a non-unique parametrization of the boundary port-variables. Afterwards, Section 9.6 deals with the reasons behind formulating the DFM without slip between the two phases in a pH representation instead of a general DFM with the Zuber-Findlay slip conditions. Finally, Section 9.7 closes with conclusions.

Notations: We first introduce few notations that are used in the sequel.

$\mathcal{L}^2(\Omega)$ is the space of square-integrable functions over the spatial domain Ω, and

$$\mathcal{L}^2(\Omega)^p = \mathcal{L}^2(\Omega) \times \mathcal{L}^2(\Omega) \times ... \times \mathcal{L}^2(\Omega) \quad \text{(p-times)}. \tag{9.1}$$

$H^1(\Omega)$ denotes the Sobolev space of functions that also possess a weak derivative. Furthermore, $H_0^1(\Omega)$ denotes the functions in $H^1(\Omega)$ that have zero boundary values. $H^1(\Omega)^p$ is defined in a manner anologous to $\mathcal{L}^2(\Omega)^p$. And, $\mathbb{R}$ denotes the space of real numbers.

9.3 Multi-phase flow models

In this section, we present two sets of nonlinear conservation laws, namely, the TFM and the DFM.

9.3.1 Two-Fluid Model (TFM)

The TFM is a set of Partial Differential Equations (PDEs) and algebraic closure relations. The PDEs expressing mass and momentum conservation for each phase are as follows:

$$\partial_t \left(\alpha_g \rho_{\mathrm{g}}\right) + \partial_x \left(\alpha_g \rho_{\mathrm{g}} v_{\mathrm{g}}\right) = 0, \tag{9.2a}$$
$$\partial_t \left(\alpha_\ell \rho_\ell\right) + \partial_x \left(\alpha_\ell \rho_\ell v_\ell\right) = 0, \tag{9.2b}$$
$$\partial_t \left(\alpha_g \rho_{\mathrm{g}} v_{\mathrm{g}}\right) + \partial_x \left(\alpha_g \rho_{\mathrm{g}} v_{\mathrm{g}}^2\right) = -\partial_x \left(\alpha_g p\right) + M_{\mathrm{g}}, \tag{9.2c}$$
$$\partial_t \left(\alpha_\ell \rho_\ell v_\ell\right) + \partial_x \left(\alpha_\ell \rho_\ell v_\ell^2\right) = -\partial_x \left(\alpha_\ell p\right) + M_\ell, \tag{9.2d}$$

where $t \in \mathbb{R}_{\geq 0}$ and $x \in [a, b]$ are, respectively, the temporal and spatial variables (a and b refer to the location of the left and the right boundary of the one-dimensional spatial domain). The model contains seven unknown variables, namely, liquid and gas void fraction, α_ℓ and α_g, liquid and gas phase velocity, v_ℓ and v_{g}, liquid and gas phase density, ρ_ℓ and ρ_{g}, and the common pressure p.

To complete the model, we use one set of the most widely applied closure laws as in [68]:

$$\alpha_g + \alpha_\ell = 1, \tag{9.3a}$$
$$M_{\mathrm{g}} + M_\ell = 0, \tag{9.3b}$$
$$M_{\mathrm{g}} = p\partial_x \alpha_g + M_{ig}, \tag{9.3c}$$
$$M_{ig} = b_g^M (v_\ell - v_{\mathrm{g}}), \quad \text{with } b_g^M \geq 0, \tag{9.3d}$$
$$\rho_{\mathrm{g}} = \frac{p}{c_g^2}, \tag{9.3e}$$
$$\rho_\ell = \rho_{\ell 0} + \frac{p - p_{\ell 0}}{c_\ell^2}, \tag{9.3f}$$

where (9.3a) expresses that any pipe segment is occupied by the combination of gas and liquid. The terms M_{g} and M_ℓ with the constant b_g^M in (9.3b)–(9.3d) account for the force interaction between the phases. Finally, (9.3e)–(9.3f) define the equation of state of each phase with the reference density and pressure as $\rho_{\ell 0}$ and $p_{\ell 0}$, and c_g and c_ℓ are the constant speeds of sound in the gas and liquid phase, respectively.

Remark 9.1. *We do not consider gravitational and frictional effects in the above TFM description for the sake of simplicity. However, in principle, the TFM can be formulated with the additional terms accounting for these effects [68].*

The TFM, governed by the set of equations (9.2) and (9.3), can be written in terms of only four physical variables. We introduce the following shorthand notations: $m_g := \alpha_{\mathrm{g}}\rho_{\mathrm{g}}$ and $m_\ell := \alpha_\ell\rho_\ell$.

Assumption 9.2. *The gas void fraction, the liquid void fraction, the liquid and the gaseous phase densities along with $\beta = \rho_{\ell 0}c_\ell^2 - p_{\ell 0}$ are positive.*

Lemma 9.3. *.model By considering m_g, m_ℓ, v_g and v_ℓ as state variables, the system of equations* (9.2) *and* (9.3) *can be re-written in the following form:*

$$\partial_t m_{\mathrm{g}} + \partial_x \left(m_{\mathrm{g}} v_{\mathrm{g}}\right) = 0, \tag{9.4a}$$

$$\partial_t m_\ell + \partial_x \left(m_\ell v_\ell\right) = 0, \tag{9.4b}$$

$$\partial_t v_{\mathrm{g}} + \partial_x \left(\frac{v_{\mathrm{g}}^2}{2}\right) = -c_g^2 \partial_x \left(\ln p\right) + \frac{b_g^M}{m_{\mathrm{g}}} v_r, \tag{9.4c}$$

$$\partial_t v_\ell + \partial_x \left(\frac{v_\ell^2}{2}\right) = -c_\ell^2 \partial_x \left(\ln\left(p+\beta\right)\right) - \frac{b_g^M}{m_\ell} v_r, \tag{9.4d}$$

where $v_r = (v_\ell - v_{\mathrm{g}})$, and

$$p\left(m_{\mathrm{g}}, m_\ell, \alpha_g\right) = m_{\mathrm{g}} c_g^2 + m_\ell c_\ell^2 - \beta\left(1-\alpha_{\mathrm{g}}\right), \tag{9.5}$$

$$\alpha_g\left(m_{\mathrm{g}}, m_\ell\right) = -m_{\mathrm{g}}\frac{c_g^2}{2\beta} - m_\ell\frac{c_\ell^2}{2\beta} + \frac{1}{2} + \sqrt{\left(m_{\mathrm{g}}\frac{c_g^2}{2\beta} + m_\ell\frac{c_\ell^2}{2\beta} - \frac{1}{2}\right)^2 + m_{\mathrm{g}}\frac{c_g^2}{\beta}}. \tag{9.6}$$

We refer the reader to [6] for the detailed proof of the expression for $\alpha_g\left(m_{\mathrm{g}}, m_\ell\right)$. In summary, the set of equations (9.4) is equivalent to (9.2) and (9.3).

9.3.2 Drift Flux Model (DFM)

The DFM can be obtained from the TFM via a slip relation of the form

$$v_{\mathrm{g}} - v_\ell = \Phi\left(m_{\mathrm{g}}, m_\ell, v_{\mathrm{g}}\right), \tag{9.7}$$

where m_{g} and m_ℓ are as introduced above. Since the slip relation (9.7) determines the coupling between the velocities of the two phases, only one momentum equation is required contrary to the two momentum equations in the TFM (9.2). Several models of the form (9.7) exist depending on the choice of the function Φ [68]. In the simplest case, without slip, $\Phi := 0$. Another case is the Zuber-Findlay relation [68]:

$$\Phi := \frac{(K-1)v_{\mathrm{g}} + S}{K\alpha_\ell} \to v_{\mathrm{g}} = K(\alpha_g v_{\mathrm{g}} + \alpha_\ell v_\ell) + S, \tag{9.8}$$

where K and S are flow-regime dependent parameters, which are assumed to be constant in this study.

Using the abbreviations $I_g := m_{\mathrm{g}} v_{\mathrm{g}}$ and $I_\ell := m_\ell v_\ell$, the governing equations for the DFM are:

$$\partial_t m_{\mathrm{g}} + \partial_x I_g = 0, \tag{9.9a}$$
$$\partial_t m_\ell + \partial_x I_\ell = 0, \tag{9.9b}$$
$$\partial_t \left(I_g + I_\ell\right) + \partial_x \left(I_g v_{\mathrm{g}} + I_\ell v_\ell\right) = -\partial_x p + Q_g + Q_v \tag{9.9c}$$

completed with closure equations (9.3a), (9.3e), (9.3f), (9.7) and gravitational effects Q_g and frictional effects Q_v defined by [69]:

$$Q_g = -g\left(m_{\mathrm{g}} + m_\ell\right)\sin\theta, \tag{9.10a}$$
$$Q_v = -\frac{32\mu_m(\alpha_g v_{\mathrm{g}} + \alpha_\ell v_\ell)}{d^2}, \tag{9.10b}$$

with gravitational constant g, space-dependent pipe inclination $\theta(x)$, mixture viscosity $\mu_m > 0$, and pipe diameter d.

Remark 9.4. *Similar to Lemma 9.3, the governing equations* (9.9) *associated with* $v := v_{\mathrm{g}} = v_\ell$ *(DFM without slip), the closure equations* (9.3a), (9.3e), (9.3f) *and* (9.10)*, upon elimination of auxiliary variables, can be rewritten as a system of PDEs with as many unknowns as equations. We omit the discussion for the sake of brevity.*

The TFM can be adapted to behave exactly like the DFM if the term M_{ig} in (9.3d) is replaced with the term stated in the following theorem. For the proof, we refer to [68].

Theorem 9.5. *Under zero gravitational and frictional effects, the DFM* (9.9) *together with* (9.3a) *and* (9.7) *is equivalent to the TFM* (9.2) *with* (9.3a)–(9.3c)*, and*

$$M_{ig} = -\alpha_g\alpha_\ell\frac{\rho_{\mathrm{g}} - \zeta\rho_\ell}{m_{\mathrm{g}} + \zeta m_\ell}\partial_x p - \frac{m_{\mathrm{g}} m_\ell}{m_{\mathrm{g}} + \zeta m_\ell}\Big(v_\ell\partial_x v_\ell -$$

$$\zeta v_{\mathrm{g}} \partial_x v_{\mathrm{g}} + \mu_g \partial_x (m_{\mathrm{g}} v_{\mathrm{g}}) + \mu_\ell \partial_x (m_\ell v_\ell) \Big), \quad (9.11)$$

with $\mu_g := \frac{\partial \Phi}{\partial m_{\mathrm{g}}}$, $\mu_\ell := \frac{\partial \Phi}{\partial m_\ell}$, $\zeta := 1 - \frac{\partial \Phi}{\partial v_{\mathrm{g}}}$.

Remark 9.6. *The equivalence of the DFM and the TFM can also be shown in the presence of gravitational and frictional effects; see [68], for further details.*

The model equivalence, stated above, will play a crucial role in drawing a conclusion about the behavior of the Hamiltonian along the solutions of the DFM by using the theoretical analysis conducted for the TFM (see Section 9.6).

9.4 Dissipative Hamiltonian Formulations

Port-Hamiltonian (pH) systems have several useful properties for system analysis and control. Basic properties of pH systems include passivity and compositionality. The pH model formulation is appealing as it helps to characterize the energy exchange across the boundaries and thus accounts for the interaction between the system and the environment. Such a framework generalizes the classical Hamiltonian framework by the definition of boundary ports. We restrict ourselves to pH systems (with state-variable z) of the form

$$\begin{aligned} \partial_t z &= \Big(\mathcal{J}(z) - \mathcal{R}(z)\Big) \delta_z \mathcal{H}(z), \\ \begin{pmatrix} f_\partial \\ e_\partial \end{pmatrix} &= \mathcal{M} \begin{pmatrix} (\delta_z \mathcal{H}(z))(b) \\ (\delta_z \mathcal{H}(z))(a) \end{pmatrix}, \end{aligned} \quad (9.12)$$

where $\mathcal{H}$ is the Hamiltonian functional, $\delta_z \mathcal{H}(z)$ its variational derivative, and $\mathcal{M}$ is a state-dependent bijective mapping. Furthermore, for every z, $\mathcal{J}(z)$ is formally skew-adjoint with respect to the L^2 inner product, i.e., for e_1, e_2 sufficiently smooth and zero at the boundary there holds

$$\int_\Omega e_1^T (\mathcal{J}(z)) e_2 \mathrm{d}x + \int_\Omega e_2^T (\mathcal{J}(z)) e_1 \mathrm{d}x = 0, \quad (9.13)$$

where Ω refers to the spatial domain, and $\mathcal{R}$ is formally self-adjoint with respect to the L^2 inner product and positive semi-definite. Finally, f_∂, e_∂ are the boundary ports.

The dissipation inequality, which expresses that energy cannot be generated within the system, is a property which directly follows from the definition of a pH system. In particular, ignoring the boundary conditions,

$$\begin{aligned} \frac{\mathrm{d}\mathcal{H}}{\mathrm{d}t} &= \int_\Omega (\delta_z \mathcal{H}(z))^T \partial_t z \ \mathrm{d}x = \int_\Omega (\delta_z \mathcal{H}(z))^T (\mathcal{J}(z) - \mathcal{R}(z)) \delta_z \mathcal{H}(z) \ \mathrm{d}x \\ &= \int_\Omega (\delta_z \mathcal{H}(z))^T (-\mathcal{R}(z)) \delta_z \mathcal{H}(z) \ \mathrm{d}x \leq 0. \end{aligned} \quad (9.14)$$

Thus, $\mathcal{R}$ is the dissipative component of the system. In the presence of boundary conditions, the behavior of the Hamiltonian along the solutions of the mathematical model is governed by the following balance equation:

$$\frac{\mathrm{d}\mathcal{H}}{\mathrm{d}t} = \int_\Omega (\delta_z \mathcal{H}(z))^T (-\mathcal{R}(z)) \delta_z \mathcal{H}(z) \, \mathrm{d}x + \text{b.t.}, \tag{9.15}$$

where b.t. denotes the boundary terms. Normally f_∂, e_∂ are chosen such that the boundary terms equal $\langle f_\partial, e_\partial \rangle$ w.r.t. some inner product. In our case, this will be the standard inner product on Euclidean space. Associated to the operators $\mathcal{J}$ and $\mathcal{R}$, we can identify an underlying geometric object called Stokes-Dirac structure. This is crucial as the pH systems can be defined with respect to these infinite-dimensional Stokes-Dirac structures [64]. Often, this structure is only associated to $\mathcal{J}$. This geometric object yields a manner to describe the boundary port variables, i.e., f_∂ and e_∂, see (9.12).

We first introduce (dissipative) Hamiltonian representations, i.e., without boundary effects for the mathematical models under consideration. The resulting formally skew-adjoint operators and formally self-adjoint operators are used as a tool to derive a non-canonical Stokes-Dirac structure, and hence the boundary port variables.

In the models discussed in Section 9.3, the Hamiltonian is dependent on the kinetic, gravitational potential and internal energy. To derive the internal energy of the system, consider the following remark.

Remark 9.7. *The internal energy $u_i, i \in \{\ell, g\}$, can be interpreted as the energy causing the expansion of the i-th compressed phase or compression of the i-th expanded phase. In order to derive this energy component, the Gibbs relation [41] under barotropic and isentropic flow considerations for an infinitesimal part of the phase is used, i.e.,*

$$\rho_i^2 \mathrm{d}u_i = p \mathrm{d}\rho_i, \qquad i \in \{\ell, g\}.$$

Using (9.3e)–(9.3f) *and integrating the above equation leads to*

$$u_\ell = -\frac{p_{\ell 0}}{\rho_\ell} + c_\ell^2 \ln \rho_\ell + \frac{\rho_{\ell 0} c_\ell^2}{\rho_\ell} + K_1, \tag{9.16a}$$

$$u_g = c_g^2 \ln \rho_\mathrm{g} + K_2, \tag{9.16b}$$

where K_1 and K_2 are the integration constants.

Considering the total energy of the system (neglecting the gravitational potential energy), we define a candidate for the Hamiltonian as follows:

$$\mathcal{H} := \int_\Omega \left(\alpha_g \rho_\mathrm{g} \frac{v_\mathrm{g}^2}{2} + \alpha_\ell \rho_\ell \frac{v_\ell^2}{2} + \alpha_g \rho_\mathrm{g} u_g + \alpha_\ell \rho_\ell u_\ell \right) \mathrm{d}x, \tag{9.17}$$

where $\Omega = [a, b]$ refers to the spatial domain.

Inserting (9.16) into (9.17), the Hamiltonian for a flow across a (unit) constant cross-section takes the following form:

$$\mathcal{H} := \int_\Omega \Big(\alpha_g \rho_{\mathrm{g}} \frac{v_{\mathrm{g}}^2}{2} + \alpha_\ell \rho_\ell \frac{v_\ell^2}{2} + \alpha_g \rho_{\mathrm{g}} (c_g^2 \ln \rho_{\mathrm{g}} + K_2) + \alpha_\ell \rho_\ell \left(c_\ell^2 \ln \rho_\ell + K_1 \right) + \alpha_\ell (c_\ell^2 \rho_{\ell 0} - p_{\ell 0}) \Big) \, \mathrm{d}x. \quad (9.18)$$

It should be noted that when $\rho_i \to 0$, $\rho_i \ln \rho_i \to 0$. The term $\rho_i \ln \rho_i$ is bounded from below, i.e., $\rho_i \ln \rho_i \geq -1/e$. So, the Hamiltonian (9.18) is bounded from below. Due to the high bulk modulus of the liquid phase, we usually have $\rho_{\ell 0} c_\ell^2 \gg p_{\ell 0}$ [69]; therefore, the positivity of the Hamiltonian (9.18) can be ensured by appropriately choosing K_1 and K_2 or even adding some constants to the Hamiltonian. For simplicity, we set $K_1 := 0$ and $K_2 := 0$ henceforth.

Remark 9.8. *The discussion in the above paragraph is reasonable from a physical perspective. However, numerically, solutions of the TFM and DFM may not be guaranteed to have non-negative density and non-negative void fractions.*

9.4.1 Dissipative Hamiltonian Formulation for the Two-Fluid Model

We now present the dissipative Hamiltonian framework for the TFM.

Theorem 9.9. *The governing equations* (9.2) *together with the closure equations* (9.3) *can be written in the following dissipative Hamiltonian form:*

$$\partial_t q = (\mathcal{J}_T(q) - \mathcal{R}_T) \, \delta_q \mathcal{H}(q) \quad (9.19)$$

with $q = [q_1,\ q_2,\ q_3,\ q_4]^T := [m_{\mathrm{g}},\ m_\ell,\ I_g,\ I_\ell]^T$, *the Hamiltonian functional* (9.18), *and where*

$$\mathcal{J}_T(q) = -\begin{bmatrix} 0 & 0 & \partial_x(q_1 \cdot) & 0 \\ 0 & 0 & 0 & \partial_x(q_2 \cdot) \\ q_1 \partial_x(\cdot) & 0 & \partial_x(q_3 \cdot) + q_3 \partial_x(\cdot) & 0 \\ 0 & q_2 \partial_x(\cdot) & 0 & \partial_x(q_4 \cdot) + q_4 \partial_x(\cdot) \end{bmatrix}$$

is a formally skew-adjoint operator with respect to the $\mathcal{L}^2$ *inner product, and*

$$\mathcal{R}_T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_g^M & -b_g^M \\ 0 & 0 & -b_g^M & b_g^M \end{bmatrix}$$

is a symmetric and positive semi-definite matrix.

Proof: Similar to (9.4), the TFM with respect to the state variables q can be straightforwardly formulated. We omit the model reformulation here for the sake of brevity.

The Hamiltonian (9.18) in terms of q_1, q_2, q_3 and q_4 is re-written as follows:

$$\mathcal{H}(q_1,q_2,q_3,q_4) := \int_\Omega \frac{q_3^2}{2q_1} + \frac{q_4^2}{2q_2} + q_1 c_g^2 \ln\left(\frac{p}{c_g^2}\right) + q_2 c_\ell^2 \ln\left(\frac{p+\beta}{c_\ell^2}\right) + (1-\alpha_g)\,\beta \;\mathrm{d}x, \quad (9.20)$$

where p and α_g can be replaced by the relations (9.5) and (9.6), respectively.

The variational derivatives are:

$$\frac{\delta\mathcal{H}}{\delta q_1} = -\frac{1}{2}\frac{q_3^2}{q_1^2} + c_g^2 \ln\left(\frac{p}{c_g^2}\right) + c_g^2, \quad \frac{\delta\mathcal{H}}{\delta q_3} = \frac{q_3}{q_1},$$
$$\frac{\delta\mathcal{H}}{\delta q_2} = -\frac{1}{2}\frac{q_4^2}{q_2^2} + c_\ell^2 \ln\left(\frac{p+\beta}{c_\ell^2}\right) + c_\ell^2, \quad \frac{\delta\mathcal{H}}{\delta q_4} = \frac{q_4}{q_2}.$$

For the sake of brevity, we omit detailed calculations here. Instead, we argue that the TFM exhibits similarities in structure with the model presented in [166], where the Hamiltonian structure was discussed for single-phase dynamics. The TFM with $b_g^M = 0$ can be viewed as two separately existing phases. The contributions due to the non-zero b_g^M enter into the dissipation matrix $\mathcal{R}_T$. The proof of the symmetric and positive semi-definite nature of $\mathcal{R}_T$ is straightforward.

The operator $\mathcal{J}_T$ is formally skew-adjoint (with respect to the $\mathcal{L}^2$ inner product). To prove formal skew-adjointness of $\mathcal{J}_T$, we check whether $\langle \mathbf{e^1}, \mathcal{J}_T\mathbf{e^2}\rangle_{\mathcal{L}^2(\Omega)} + \langle \mathcal{J}_T\mathbf{e^1}, \mathbf{e^2}\rangle_{\mathcal{L}^2(\Omega)} = 0$ for smooth $\mathbf{e^1}, \mathbf{e^2}$ which are zero at the boundary, where we define $\mathbf{e^i} = (e_1^i, e_2^i, e_3^i, e_4^i)^T$. Here, the variable e_j^i refers to the j-th element of $\mathbf{e^i}$. $\mathcal{J}_T$ is formally skew-adjoint with respect to the $\mathcal{L}^2$ inner product as

$$\begin{aligned}
-\langle \mathbf{e^1}, \mathcal{J}_T\mathbf{e^2}\rangle_{\mathcal{L}^2(\Omega)} - \langle \mathcal{J}_T\mathbf{e^1}, \mathbf{e^2}\rangle_{\mathcal{L}^2(\Omega)} = \int_\Omega\; & e_1^1\partial_x(q_1e_3^2) + q_1e_3^2\partial_x e_1^1 + \\
& e_1^2\partial_x(q_1e_3^1) + q_1e_3^1\partial_x e_1^2 + e_2^1\partial_x(q_2e_4^2) + q_2e_4^2\partial_x e_2^1 + \\
& e_2^2\partial_x(q_2e_4^1) + q_2e_4^1\partial_x e_2^2 + e_3^1\left[\partial_x(q_3e_3^2) + q_3\partial_x e_3^2\right] + \\
& e_3^2\left[\partial_x(q_3e_3^1) + q_3\partial_x e_3^1\right] + e_4^1\left[\partial_x(q_4e_4^2) + q_4\partial_x e_4^2\right] + \\
& e_4^2\left[\partial_x(q_4e_4^1) + q_4\partial_x e_4^1\right] dx = \Big(\begin{bmatrix} e_1^1 & e_2^1 & e_3^1 & e_4^1 \end{bmatrix} \underbrace{\begin{bmatrix} 0 & 0 & q_1 & 0 \\ 0 & 0 & 0 & q_2 \\ q_1 & 0 & 2q_3 & 0 \\ 0 & q_2 & 0 & 2q_4 \end{bmatrix}}_{\mathrm{Q}} \begin{bmatrix} e_1^2 \\ e_2^2 \\ e_3^2 \\ e_4^2 \end{bmatrix}\Big)\Big|_a^b,
\end{aligned} \quad (9.21)$$

which vanishes under our assumptions on the boundary conditions. □

9.4.2 Dissipative Hamiltonian Formulation for the Drift Flux Model

So far, we focused on the dissipative Hamiltonian representation for the TFM. We will now deal with the DFM under gravitational and frictional effects, and present a corresponding dissipative Hamiltonian formulation. For the DFM, we focus only on a case in which there is no slip between the phases, i.e., $v := v_{\mathrm{g}} = v_\ell$ (the reason for adopting this no-slip assumption is provided in Section 9.6). Since gravitation is considered, the gravitational potential energy needs to be added to the Hamiltonian. The Hamiltonian now takes the following form:

$$\mathcal{H}_D(m_{\mathrm{g}}, m_\ell, v) = \int_\Omega m_{\mathrm{g}} \frac{v^2}{2} + m_\ell \frac{v^2}{2} + m_\ell c_\ell^2 \ln\left(\frac{p+\beta}{c_\ell^2}\right) + m_{\mathrm{g}} c_g^2 \ln\left(\frac{p}{c_g^2}\right) + \alpha_\ell \beta + (m_{\mathrm{g}} + m_\ell)\left(\int_a^x g \sin(\theta(\xi)) \mathrm{d}\xi\right) \mathrm{d}x. \quad (9.22)$$

Using the above candidate Hamiltonian function $\mathcal{H}_D$, a dissipative Hamiltonian representation of a special case of the DFM is shown below.

Theorem 9.10. *The governing equations* (9.9) *together with* $v := v_{\mathrm{g}} = v_\ell$ *(case of no slip), the closure equations* (9.3a), (9.3e), (9.3f) *and* (9.10) *can be written in dissipative Hamiltonian form as follows:*

$$\partial_t z_D = \left(\mathcal{J}_D(z_D) - \mathcal{R}_D(z_D)\right) \delta_{z_D} \mathcal{H}_D(z_D) \quad (9.23)$$

with $z_D := [m_{\mathrm{g}},\ m_\ell,\ v]^T$, *the Hamiltonian functional* (9.22), *where*

$$\mathcal{J}_D(z_D) = -\begin{bmatrix} 0 & 0 & \partial_x\left(\frac{m_{\mathrm{g}}}{m_{\mathrm{g}}+m_\ell}\cdot\right) \\ 0 & 0 & \partial_x\left(\frac{m_\ell}{m_{\mathrm{g}}+m_\ell}\cdot\right) \\ \frac{m_{\mathrm{g}}}{m_{\mathrm{g}}+m_\ell}\partial_x(\cdot) & \frac{m_\ell}{m_{\mathrm{g}}+m_\ell}\partial_x(\cdot) & 0 \end{bmatrix}$$

is a formally skew-adjoint operator with respect to the $\mathcal{L}^2$ *inner product, and*

$$\mathcal{R}_D(z_D) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{32\mu_m}{d^2(m_{\mathrm{g}}+m_\ell)^2} \end{bmatrix}$$

is a symmetric and positive semi-definite matrix.

Proof: First note that, using (9.9a) and (9.9b), the left-hand side of equation (9.9c) can be rewritten as

$$(m_{\mathrm{g}} + m_\ell)\,\partial_t v + v\partial_t\,(m_{\mathrm{g}} + m_\ell) + \partial_x\left((m_{\mathrm{g}} + m_\ell)\,v^2\right)$$

$$= (m_{\mathrm{g}} + m_\ell) \left(\partial_t v + \partial_x \left(\frac{v^2}{2} \right) \right).$$

Thus, instead of (9.9c) we can also consider

$$\partial_t v + \partial_x \left(\frac{v^2}{2} \right) = \frac{1}{m_{\mathrm{g}} + m_\ell} \left(-\partial_x p + Q_g + Q_v \right). \tag{9.24}$$

The variational derivatives of $\mathcal{H}_D$ are given by:

$$\begin{aligned}
\frac{\delta \mathcal{H}_D}{\delta m_{\mathrm{g}}} &= c_g^2 \ln \left(\frac{p}{c_g^2} \right) + \frac{v^2}{2} + c_g^2 + \int_a^x g \sin(\theta(\xi)) \mathrm{d}\xi, \\
\frac{\delta \mathcal{H}_D}{\delta m_\ell} &= c_\ell^2 \ln \left(\frac{p+\beta}{c_\ell^2} \right) + \frac{v^2}{2} + c_\ell^2 + \int_a^x g \sin(\theta(\xi)) \mathrm{d}\xi, \\
\frac{\delta \mathcal{H}_D}{\delta v} &= (m_{\mathrm{g}} + m_\ell)\, v.
\end{aligned}$$

Next, we prove the claim equation by equation. The first line of (9.23) reads

$$\partial_t m_{\mathrm{g}} = -\partial_x \left(\frac{m_{\mathrm{g}}}{m_{\mathrm{g}} + m_\ell} (m_{\mathrm{g}} + m_\ell)\, v \right) = -\partial_x \left(m_{\mathrm{g}} v \right). \tag{9.25}$$

Similarly, the second line is

$$\partial_t m_\ell = -\partial_x \left(\frac{m_\ell}{m_{\mathrm{g}} + m_\ell} (m_{\mathrm{g}} + m_\ell)\, v \right) = -\partial_x \left(m_\ell v \right). \tag{9.26}$$

Let us introduce a short-hand notation $G = \int_a^x g \sin(\theta(\xi)) \mathrm{d}\xi$. Then, the third line yields

$$\begin{aligned}
\partial_t v = &- \frac{m_{\mathrm{g}}}{m_{\mathrm{g}} + m_\ell} \partial_x \left(c_g^2 \ln \left(\frac{p}{c_g^2} \right) + \frac{v^2}{2} + c_g^2 + G \right) \\
&- \frac{m_\ell}{m_{\mathrm{g}} + m_\ell} \partial_x \left(c_\ell^2 \ln \left(\frac{p+\beta}{c_\ell^2} \right) + \frac{v^2}{2} + c_\ell^2 + G \right) \\
&- \frac{32 \mu_m}{d^2 \left(m_{\mathrm{g}} + m_\ell \right)^2} (m_{\mathrm{g}} + m_\ell)\, v \\
= &- \partial_x \left(\frac{v^2}{2} \right) - \frac{1}{(m_{\mathrm{g}} + m_\ell)} (\partial_x p + Q_g + Q_v).
\end{aligned} \tag{9.27}$$

The claim follows by observing that (9.25), (9.26), and (9.27) are identical to (9.9a), (9.9b), and (9.24), respectively.

The symmetric and positive semi-definite nature of $\mathcal{R}_D$ follows immediately from the positivity of μ_m. The formal skew-adjointness of $\mathcal{J}_D$ essentially follows

from integration by parts and neglecting the boundary conditions. The operator $\mathcal{J}_D$ contains terms similar to the skew-adjoint operator $\mathcal{J}_T$, the formal skew-adjointness of which was discussed extensively in the proof of Theorem 9.9. For the sake of brevity, we refer the reader to follow similar lines of reasoning to show the formal skew-adjointness of $\mathcal{J}_D$. □

9.5 Geometrical properties of the system: Stokes-Dirac structures

We now define a geometric structure, a generalization of symplectic and Poisson structures, called a Stokes-Dirac structure.

Definition 9.11. *[64], [77] Consider $\mathcal{F}$ and $\mathcal{E}$ as real Hilbert spaces which are isometrically isomorphic. The subspace $\mathcal{D} \subset \mathcal{F} \times \mathcal{E}$ is a Stokes-Dirac structure if $\mathcal{D} = \mathcal{D}^\perp$, where $\mathcal{D}^\perp$ denotes the orthogonal complement which is defined as*

$$\mathcal{D}^\perp := \{(\tilde{\mathbf{f}}, \tilde{\mathbf{e}}) \in \mathcal{F} \times \mathcal{E} \mid \ll (\tilde{\mathbf{f}}, \tilde{\mathbf{e}}), (\mathbf{f}, \mathbf{e}) \gg = 0 \quad \forall (\mathbf{f}, \mathbf{e}) \in \mathcal{D}\}. \tag{9.28}$$

Here, $\ll (\tilde{\mathbf{f}}, \tilde{\mathbf{e}}), (\mathbf{f}, \mathbf{e}) \gg$ is defined as follows:

$$\ll (\tilde{\mathbf{f}}, \tilde{\mathbf{e}}), (\mathbf{f}, \mathbf{e}) \gg := \langle \tilde{\mathbf{f}} \mid \mathbf{e} \rangle + \langle \mathbf{f} \mid \tilde{\mathbf{e}} \rangle, \tag{9.29}$$

where the notation $\langle \mathbf{f} \mid \mathbf{e} \rangle$ indicates a non-degenerate bilinear form defined on the bond space $\mathcal{B} = \mathcal{F} \times \mathcal{E}$.

This structure relates the composing elements of a system in a power-conserving manner [138]. Such geometric structures often have a compositionality property [7], [106], [138].

For (f, e) element of a Stokes-Dirac structure, it is easy to see that $\langle f \mid e \rangle = 0$, and thus there is a close relation to (formally) skew-adjoint operators, see also (9.13). However, if $f = \mathcal{J}e$ for all $(f, e) \in D$, and J is formally skew-adjoint, then $\mathcal{D} \subset \mathcal{D}^\perp$. To make such a $\mathcal{D}$ into a Stokes-Dirac structure, it is required that $\mathcal{D} = \mathcal{D}^\perp$ holds. The formally skew-adjoint part of a pH system will form the foundation of the associated Stokes-Dirac structure, as we will show as well.

Non-linearity encoded within the Hamiltonian along with a linear Stokes-Dirac structure constitutes a favorable representation of PDEs. Such a structure facilitates the analysis of non-linear systems as the linearity of the Stokes-Dirac structure can be exploited to assess system behavior. Stokes-Dirac structures can also be used to formulate boundary control systems [77].

In the existing results [64], [77], [175], the skew-adjoint operator yields a symmetric bilinear form on the space of the boundary variables. An important tool used in that framework is the trace operator, which, in earlier works [64], [77], [175], requires that the effort variables $\mathbf{e}$ belong to the function class $H^1(\Omega)$. Given the state-dependent nature of skew-adjoint operators in (9.19) and (9.23)

(unlike in [77]), a combination of the states and the effort variables have to belong to the function class $H^1(\Omega)$ or suitable conditions have to be imposed on the state variables in order to have effort variables belonging to the function class $H^1(\Omega)$ (see Theorems 9.14 and 9.16). Boundary port-variables have been parametrized in [77] using the trace operators. However, such an elegant parametrization is limited to the case of a non-singular matrix Q (synonymous to (9.21)) arising in linear problems with state-independent operators. To the best of our knowledge, the work [175] is the only work in the scope of parametrization of boundary port-variables for a singular matrix Q, thereby enlarging the class of systems that can be dealt. Villegas in [175] demonstrated the approach to define the non-degenerate bilinear form under singular Q and consequently modified the definition of the boundary port-variables. However, [175] was limited to the setting of *state-independent* Stokes-Dirac structures. In this work, we extend the definition of boundary port-variables to eventually obtain *state-dependent* Stokes-Dirac structures with boundary ports for non-linear problems with non-quadratic Hamiltonian functional. It should be mentioned that the authors in [154] have also considered state-dependent Stokes-Dirac structures for problems (for instance, ideal isentropic fluid) with non-quadratic Hamiltonian functional by using a differential geometric viewpoint. We, contrarily, use the matrix or operator-theoretic viewpoint in the consideration of such geometric structures in the scope of the compressible two-phase flow models.

Remark 9.12. *Boundary port-variables, in our setting, will remain unchanged in the presence of dissipation. This is only true since our resistive operator ($\mathcal{R}$) does not include any differential operator. In general, the boundary ports could also include contributions from the resistive part. In this work, we only consider Stokes-Dirac structures without accounting for resistive ports (for the above mentioned reason) and finally arrive at a definition of the boundary port-variables, which is practical for pH representations.*

We recall the following fundamental lemma of calculus of variations.

Lemma 9.13. *If the pair $(h, m) \in \mathcal{L}^2(\Omega)^2$ satisfies*

$$\int_a^b [h(x)\partial_x f(x) + m(x)f(x)]\mathrm{d}x = 0, \tag{9.30}$$

for all $f \in H_0^1(\Omega)$, then

$$h \in H^1(\Omega), \quad \textit{and} \quad \partial_x h = m(x). \tag{9.31}$$

Lemma 9.13 will be extensively used in order to prove that a certain structure is a Stokes-Dirac structure.

Using the above mathematical preliminaries, we first propose a Stokes-Dirac structure for the TFM and present a corresponding proof, and then we propose it for the DFM without slip.

9.5.1 Stokes-Dirac structure representation for the Two-Fluid Model

We, first, introduce the notations

$$\mathbf{f}_t = \begin{bmatrix} f_{m_\mathrm{g}} & f_{m_\ell} & f_{I_g} & f_{I_\ell} & f^B_{a,t} & f^B_{b,t} \end{bmatrix}^T, \tag{9.32a}$$

$$\mathbf{e}_t = \begin{bmatrix} e_{m_\mathrm{g}} & e_{m_\ell} & e_{I_g} & e_{I_\ell} & e^B_{a,t} & e^B_{b,t} \end{bmatrix}^T, \tag{9.32b}$$

$$\mathbf{f}_{tr} = \begin{bmatrix} f_{m_\mathrm{g}} & f_{m_\ell} & f_{I_g} & f_{I_\ell} \end{bmatrix}^T, \tag{9.32c}$$

$$\mathbf{e}_{tr} = \begin{bmatrix} e_{m_\mathrm{g}} & e_{m_\ell} & e_{I_g} & e_{I_\ell} \end{bmatrix}^T \tag{9.32d}$$

with $\mathbf{f}_t \in \mathcal{F}_t$, $\mathbf{e}_t \in \mathcal{E}_t$ where $\mathcal{F}_t = \mathcal{E}_t = \mathcal{L}^2(\Omega)^4 \times \mathbb{R}^2 \times \mathbb{R}^2$. On $\mathcal{F}_t \times \mathcal{E}_t$ the following non-degenerate bilinear form is defined:

$$\langle \mathbf{f}_t \mid \mathbf{e}_t \rangle = \int_\Omega (f_{m_\mathrm{g}} e_{m_\mathrm{g}} + f_{m_\ell} e_{m_\ell} + f_{I_g} e_{I_g} + f_{I_\ell} e_{I_\ell}) \mathrm{d}x + (f^B_{b,t})^T e^B_{b,t} + (f^B_{a,t})^T e^B_{a,t}. \tag{9.33}$$

Using these notations, the Stokes-Dirac structure corresponding to the dissipative Hamiltonian representation of the TFM can be expressed as follows.

Theorem 9.14. *Consider $\mathcal{F}_t$ and $\mathcal{E}_t$ as introduced above. Moreover, assume that $m_\mathrm{g}, m_\ell, I_g, I_\ell =: q_1, q_2, q_3, q_4 \in H^1(\Omega)$. We also assume that $q_1, q_2 > 0$ on Ω. Then, the linear subset $\mathcal{D}_t \subset \mathcal{F}_t \times \mathcal{E}_t$ defined as follows:*

$$\begin{aligned} \mathcal{D}_t = \Big\{ (\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{F}_t \times \mathcal{E}_t \mid \mathbf{e}_{tr} \in H^1(\Omega)^4, \mathbf{f}_{tr} = \mathcal{J}_t(q)\mathbf{e}_{tr}, \\ \begin{pmatrix} f^B_{b,t} \\ e^B_{b,t} \end{pmatrix} = \begin{pmatrix} f^B_{b1,t} \\ f^B_{b2,t} \\ e^B_{b1,t} \\ e^B_{b2,t} \end{pmatrix} = \left(\begin{pmatrix} q_1 & 0 & q_3 & 0 \\ 0 & q_2 & 0 & q_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e_{m_\mathrm{g}} \\ e_{m_\ell} \\ e_{I_g} \\ e_{I_\ell} \end{pmatrix} \right) (b), \\ \begin{pmatrix} f^B_{a,t} \\ e^B_{a,t} \end{pmatrix} = \begin{pmatrix} f^B_{a1,t} \\ f^B_{a2,t} \\ e^B_{a1,t} \\ e^B_{a2,t} \end{pmatrix} = \left(\begin{pmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ q_1 & 0 & q_3 & 0 \\ 0 & q_2 & 0 & q_4 \end{pmatrix} \begin{pmatrix} e_{m_\mathrm{g}} \\ e_{m_\ell} \\ e_{I_g} \\ e_{I_\ell} \end{pmatrix} \right) (a) \Big\}, \end{aligned} \tag{9.34}$$

where

$$\mathcal{J}_t(q) = - \begin{bmatrix} 0 & 0 & \partial_x(m_g \cdot) & 0 \\ 0 & 0 & 0 & \partial_x(m_\ell \cdot) \\ m_g \partial_x(\cdot) & 0 & \partial_x(I_g \cdot) + I_g \partial_x(\cdot) & 0 \\ 0 & m_\ell \partial_x(\cdot) & 0 & \partial_x(I_\ell \cdot) + I_\ell \partial_x(\cdot) \end{bmatrix} \tag{9.35}$$

is a Stokes-Dirac structure with respect to the symmetric pairing given by

$$\ll (\mathbf{f}_t, \mathbf{e}_t), (\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \gg = \langle \mathbf{f}_t \mid \tilde{\mathbf{e}}_t \rangle + \langle \tilde{\mathbf{f}}_t \mid \mathbf{e}_t \rangle, \\ (\mathbf{f}_t, \mathbf{e}_t), (\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \in \mathcal{F}_t \times \mathcal{E}_t, \quad (9.36)$$

where the pairing $\langle \cdot \mid \cdot \rangle$ *is given in* (9.33).

Proof: The proof is divided into two parts. We first prove that $\mathcal{D}_t \subset \mathcal{D}_t^{\perp}$.

We consider two pairs of flow and effort variables belonging to the Stokes-Dirac structure, i.e., $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ and $(\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \in \mathcal{D}_t$. Using the earlier introduced notations, we obtain:

$$\begin{aligned} \ll (\mathbf{f}_t, \mathbf{e}_t), (\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \gg = & \int_\Omega (f_{m_\mathrm{g}} \tilde{e}_{m_\mathrm{g}} + f_{m\ell} \tilde{e}_{m\ell} + f_{I_g} \tilde{e}_{I_g} + f_{I_\ell} \tilde{e}_{I_\ell}) \mathrm{d}x + \\ & \int_\Omega (\tilde{f}_{m_\mathrm{g}} e_{m_\mathrm{g}} + \tilde{f}_{m\ell} e_{m\ell} + \tilde{f}_{I_g} e_{I_g} + \tilde{f}_{I_\ell} e_{I_\ell}) \mathrm{d}x + \\ & (f_{a,t}^B)^T \tilde{e}_{a,t}^B + (f_{b,t}^B)^T \tilde{e}_{b,t}^B + (\tilde{f}_{a,t}^B)^T e_{a,t}^B + (\tilde{f}_{b,t}^B)^T e_{b,t}^B. \end{aligned} \quad (9.37)$$

Substituting the mappings between the flow and the effort variables, the total sum within the integrals of (9.37) becomes

$$\begin{aligned} & \Big[-\partial_x(q_1 e_{I_g}) \tilde{e}_{m_\mathrm{g}} - \partial_x(q_2 e_{I_\ell}) \tilde{e}_{m\ell} + \Big(-q_1 \partial_x e_{m_\mathrm{g}} - \partial_x(q_3 e_{I_g}) - q_3 \partial_x e_{I_g} \Big) \tilde{e}_{I_g} + \\ & \Big(-q_2 \partial_x e_{m\ell} - \partial_x(q_4 e_{I_\ell}) - q_4 \partial_x e_{I_\ell} \Big) \tilde{e}_{I_\ell} \Big] + \Big[-\partial_x(q_1 \tilde{e}_{I_g}) e_{m_\mathrm{g}} - \partial_x(q_2 \tilde{e}_{I_\ell}) e_{m\ell} + \\ & \Big(-q_1 \partial_x \tilde{e}_{m_\mathrm{g}} - \partial_x(q_3 \tilde{e}_{I_g}) - q_3 \partial_x \tilde{e}_{I_g} \Big) e_{I_g} + \Big(-q_2 \partial_x \tilde{e}_{m\ell} - \partial_x(q_4 \tilde{e}_{I_\ell}) - q_4 \partial_x \tilde{e}_{I_\ell} \Big) e_{I_\ell} \Big) \Big] \\ & = -\partial_x(q_1 \tilde{e}_{m_\mathrm{g}} e_{I_g}) - \partial_x(q_1 e_{m_\mathrm{g}} \tilde{e}_{I_g}) - \partial_x(q_2 e_{I_\ell} \tilde{e}_{m\ell}) - \partial_x(q_2 \tilde{e}_{I_\ell} e_{m\ell}) \\ & - \partial_x(q_3 e_{I_g} \tilde{e}_{I_g}) - \partial_x(q_3 e_{I_g} \tilde{e}_{I_g}) - \partial_x(q_4 e_{I_\ell} \tilde{e}_{I_\ell}) - \partial_x(q_4 e_{I_\ell} \tilde{e}_{I_\ell}). \end{aligned}$$

Performing integration on the above expression, it equals minus the last expressions in (9.37) and hence, $\mathcal{D}_t \subset \mathcal{D}_t^{\perp}$. This concludes the first part of the proof.

We now prove the converse part, i.e., $\mathcal{D}_t^{\perp} \subset \mathcal{D}_t$. For this, we follow the steps similar to Proposition 4.1 in [64]. The proof consists of several repeated steps, which are summarized below. We take $(\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \in \mathcal{D}_t^{\perp}$ i.e., $(\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \in \mathcal{F}_t \times \mathcal{E}_t$ such that $\ll (\mathbf{f}_t, \mathbf{e}_t), (\tilde{\mathbf{f}}_t, \tilde{\mathbf{e}}_t) \gg = 0 \quad \forall (\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$. To this end, we use the freedom in the choice of the effort variables and exploit Lemma 9.13.

Step 1: Let $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m\ell}, e_{I_g}, e_{I_\ell} = 0$ and $e_{m_\mathrm{g}}(a) = e_{m_\mathrm{g}}(b) = 0$. Using (9.37), we find that

$$\int_\Omega -(q_1 \partial_x e_{m_\mathrm{g}}) \tilde{e}_{I_g} + \tilde{f}_{m_\mathrm{g}} e_{m_\mathrm{g}} \mathrm{d}x = 0 \quad \forall e_{m_\mathrm{g}} \in H_0^1(\Omega). \quad (9.38)$$

Lemma 9.13 gives

$$q_1\tilde{e}_{I_g} \in H^1(\Omega) \quad \text{and} \quad \tilde{f}_{m_g} = -\partial_x(q_1\tilde{e}_{I_g}). \tag{9.39}$$

Using $q_1 \in H^1(\Omega)$ along with $q_1 > 0$ on Ω, we obtain that $\tilde{e}_{I_g} \in H^1(\Omega)$.
Step 2: Considering $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g}, e_{I_g}, e_{I_\ell} = 0$ and $e_{m_\ell} \in H_0^1(\Omega)$, we have by (9.37) that

$$\int_\Omega -(q_2\partial_x e_{m_\ell})\tilde{e}_{I_\ell} + \tilde{f}_{m_\ell} e_{m_\ell} \mathrm{d}x = 0 \quad \forall e_{m_\ell} \in H_0^1(\Omega). \tag{9.40}$$

Now using Lemma 9.13 leads to

$$q_2\tilde{e}_{I_\ell} \in H^1(\Omega) \quad \text{and} \quad \tilde{f}_{m_\ell} = -\partial_x(q_2\tilde{e}_{I_\ell}). \tag{9.41}$$

As before, using $q_2 \in H^1(\Omega)$ along with $q_2 > 0$ on Ω, we have that $\tilde{e}_{I_\ell} \in H^1(\Omega)$.
Step 3: For $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g}, e_{m_\ell}, e_{I_\ell} = 0$ and $e_{I_g} \in H_0^1(\Omega)$, we obtain:

$$\int_\Omega -\partial_x(q_1 e_{I_g})\tilde{e}_{m_g} - \partial_x(q_3 e_{I_g})\tilde{e}_{I_g} - (q_3\partial_x e_{I_g})\tilde{e}_{I_g} + \tilde{f}_{I_g} e_{I_g} \mathrm{d}x = 0 \quad \forall e_{I_g} \in H_0^1(\Omega).$$

We rewrite the above equation as follows:

$$\int_\Omega -(\partial_x q_1)(e_{I_g}\tilde{e}_{m_g}) - (\partial_x q_3)(e_{I_g}\tilde{e}_{I_g}) - (\partial_x e_{I_g}) \cdot \left(q_1\tilde{e}_{m_g} + 2q_3\tilde{e}_{I_g}\right) + \tilde{f}_{I_g} e_{I_g} \mathrm{d}x = 0 \quad \forall e_{I_g} \in H_0^1(\Omega).$$

As a result of Lemma 9.13, we have that $q_1\tilde{e}_{m_g} + 2q_3\tilde{e}_{I_g} \in H^1(\Omega)$. Moreover, we obtain the following identity:

$$\tilde{f}_{I_g} = -\partial_x(q_1\tilde{e}_{m_g} + 2q_3\tilde{e}_{I_g}) + \tilde{e}_{m_g}\partial_x q_1 + \tilde{e}_{I_g}\partial_x q_3. \tag{9.42}$$

Using $q_1, q_3, \tilde{e}_{I_g} \in H^1(\Omega)$ and that $q_1 > 0$, it can easily be deduced that $\tilde{e}_{m_g} \in H^1(\Omega)$, and so (9.42) can be written as

$$\tilde{f}_{I_g} = -q_1\partial_x\tilde{e}_{m_g} - \partial_x(q_3\tilde{e}_{I_g}) - q_3\partial_x\tilde{e}_{I_g}. \tag{9.43}$$

Step 4: Considering $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g}, e_{m_\ell}, e_{I_g} = 0$ and $e_{I_\ell} \in H_0^1(\Omega)$, we obtain:

$$\int_\Omega -\partial_x(q_2 e_{I_\ell})\tilde{e}_{m_\ell} - \partial_x(q_4 e_{I_\ell})\tilde{e}_{I_\ell} - (q_4\partial_x e_{I_\ell})\tilde{e}_{I_\ell} + \tilde{f}_{I_\ell} e_{I_\ell} \mathrm{d}x = 0 \quad \forall e_{I_\ell} \in H_0^1(\Omega).$$

Re-writing the above equation as in the previous step and using Lemma 9.13, we have that $q_2\tilde{e}_{m_\ell} + 2q_4\tilde{e}_{I_\ell} \in H^1(\Omega)$ and also obtain:

$$\tilde{f}_{I_\ell} = -\partial_x(q_2\tilde{e}_{m_\ell} + 2q_4\tilde{e}_{I_\ell}) + \tilde{e}_{m_\ell}\partial_x q_2 + \tilde{e}_{I_\ell}\partial_x q_4. \tag{9.44}$$

Using $q_2, q_4, \tilde{e}_{I_\ell} \in H^1(\Omega)$ and that $q_2 > 0$, it can easily be deduced that $\tilde{e}_{m_\ell} \in H^1(\Omega)$ and so

$$\tilde{f}_{I_\ell} = -q_2\partial_x\tilde{e}_{m_\ell} - \partial_x(q_4\tilde{e}_{I_\ell}) - q_4\partial_x\tilde{e}_{I_\ell}. \tag{9.45}$$

Step 5: Let $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_\ell} = e_{I_g} = e_{I_\ell} = 0$ and $e_{m_g}(a) = 0, e_{m_g}(b) \neq 0$. Using the procedure outlined above, we obtain the following identity: $\tilde{e}^B_{b1,t} = \tilde{e}_{I_g}\,|_b$.
Step 6: Let $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g} = e_{I_g} = e_{I_\ell} = 0$ and $e_{m_\ell}(a) = 0, e_{m_\ell}(b) \neq 0$. We now observe that $\tilde{e}^B_{b2,t} = \tilde{e}_{I_\ell}\,|_b$ holds.
Step 7: Let $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g} = e_{m_\ell} = e_{I_\ell} = 0$ and $e_{I_g}(a) = 0, e_{I_g}(b) \neq 0$. Using the outlined procedure, we now obtain:

$$-(q_1\tilde{e}_{m_g}e_{I_g})\,|_b - (q_3\tilde{e}_{I_g}e_{I_g})\,|_b + \tilde{f}^B_{b1,t}e_{I_g}\,|_b = 0. \tag{9.46}$$

Finally, we obtain the following identity:

$$\tilde{f}^B_{b1,t} = \left(q_1\tilde{e}_{m_g} + q_3\tilde{e}_{I_g}\right)|_b\,. \tag{9.47}$$

Step 8: Let $(\mathbf{f}_t, \mathbf{e}_t) \in \mathcal{D}_t$ with $e_{m_g} = e_{m_\ell} = e_{I_g} = 0$ and $e_{I_\ell}(a) = 0, e_{I_\ell}(b) \neq 0$. Using the outlined procedure, we now obtain the following identity:

$$\tilde{f}^B_{b2,t} = \left(q_2\tilde{e}_{m_\ell} + q_4\tilde{e}_{I_\ell}\right)|_b\,. \tag{9.48}$$

The boundary port-variables $f^B_{a1.t}, f^B_{a2,t}, e^B_{a1,t}$ and $e^B_{a2,t}$ can be obtained in a manner similar to the one outlined for computing the boundary port-variables at the right boundary of the spatial domain Ω.

Thus, in summary we have shown $\mathcal{D}_t^\perp \subset \mathcal{D}_t$ and, hence, $\mathcal{D}_t$ is a Stokes-Dirac structure. □

Remark 9.15. *The formally skew-adjoint operator $\mathcal{J}_T(q)$ in Theorem 9.9 is equal to the skew-adjoint operator $\mathcal{J}_t(q)$ associated to the Stokes-Dirac structure representation in Theorem 9.14. These operators are found to be equal only because of the assumptions on the state variables q; see Theorem 9.14 for details. In general, the formally skew-adjoint operator and the skew-adjoint operator associated to the Stokes-Dirac structure representation need not be the same. For instance, see Theorem 9.16.*

We now discuss the representation of the Stokes-Dirac structure corresponding to the skew-adjoint operator $\mathcal{J}_D$ in the scope of the Drift Flux Model without slip.

9.5.2 Stokes-Dirac structure representation for the Drift Flux Model

We introduce the notations

$$\mathbf{f}_d = \begin{bmatrix} f_{m_g,d} & f_{m_\ell,d} & f_{v,d} & f^B_{a,d} & f^B_{b,d} \end{bmatrix}^T, \tag{9.49a}$$

$$\mathbf{e}_d = \begin{bmatrix} e_{m_g,d} & e_{m_\ell,d} & e_{v,d} & e^B_{a,d} & e^B_{b,d} \end{bmatrix}^T, \tag{9.49b}$$

$$\mathbf{f}_{dr} = \begin{bmatrix} f_{m_g,d} & f_{m_\ell,d} & f_{v,d} \end{bmatrix}^T, \tag{9.49c}$$

$$\mathbf{e}_{dr} = \begin{bmatrix} e_{m_g,d} & e_{m_\ell,d} & e_{v,d} \end{bmatrix}^T. \tag{9.49d}$$

A Stokes-Dirac structure for the dissipative Hamiltonian representation of the DFM can be expressed as follows.

Theorem 9.16. *Consider $\mathcal{F}_d = \mathcal{E}_d = \mathcal{L}^2(\Omega)^3 \times \mathbb{R}^2$. We assume that $A_g := \frac{m_g}{m_g+m_\ell}$, $A_\ell := \frac{m_\ell}{m_g+m_\ell} \in H^1(\Omega)$. We also consider that the non-degenerate bilinear form on $\mathcal{F}_d \times \mathcal{E}_d$ is defined in the following way:*

$$\langle \mathbf{f}_d \mid \mathbf{e}_d \rangle = \int_\Omega (f_{m_g,d} e_{m_g,d} + f_{m_\ell,d} e_{m_\ell,d} + f_{v,d} e_{v,d}) \mathrm{d}x + f^B_{b,d} e^B_{b,d} + f^B_{a,d} e^B_{a,d}. \tag{9.50}$$

Then, the linear subset $\mathcal{D}_d \subset \mathcal{F}_d \times \mathcal{E}_d$ given by

$$\mathcal{D}_d = \Big\{ (\mathbf{f}_d, \mathbf{e}_d) \in \mathcal{F}_d \times \mathcal{E}_d, \begin{pmatrix} A_g e_{m_g,d} + A_\ell e_{m_\ell,d} \\ e_{v,d} \end{pmatrix} \in H^1(\Omega)^2, \mathbf{f}_{dr} = \mathcal{J}_d(z_D) \mathbf{e}_{dr},$$
$$\begin{pmatrix} f^B_{a,d} \\ e^B_{a,d} \end{pmatrix} = \left(\begin{pmatrix} -A_g & -A_\ell & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e_{m_g,d} \\ e_{m_\ell,d} \\ e_{v,d} \end{pmatrix} \right) (a),$$
$$\begin{pmatrix} f^B_{b,d} \\ e^B_{b,d} \end{pmatrix} = \left(\begin{pmatrix} A_g & A_\ell & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e_{m_g,d} \\ e_{m_\ell,d} \\ e_{v,d} \end{pmatrix} \right) (b) \Big\}, \tag{9.51}$$

where

$$\mathcal{J}_d(z_D) = \begin{pmatrix} 0 & 0 & -\partial_x(A_g \cdot) \\ 0 & 0 & -\partial_x(A_\ell \cdot) \\ -D(A_g \cdot) \& D(A_\ell \cdot) & & 0 \end{pmatrix}$$

is a Stokes-Dirac structure with respect to the symmetric pairing given by, see (9.50)*:*

$$\ll (\mathbf{f}_d, \mathbf{e}_d), (\tilde{\mathbf{f}}_d, \tilde{\mathbf{e}}_d) \gg = \langle \mathbf{f}_d \mid \tilde{\mathbf{e}}_d \rangle + \langle \tilde{\mathbf{f}}_d \mid \mathbf{e}_d \rangle,$$
$$(\mathbf{f}_d, \mathbf{e}_d), (\tilde{\mathbf{f}}_d, \tilde{\mathbf{e}}_d) \in \mathcal{F}_d \times \mathcal{E}_d. \tag{9.52}$$

The action of the operator $D(A_g\cdot)\&D(A_\ell\cdot)$ is given by

$$D(A_g\cdot)\&D(A_\ell\cdot)\begin{pmatrix} e_{m_{\mathrm{g}},d} \\ e_{m_\ell,d}\end{pmatrix} = \partial_x(A_g e_{m_{\mathrm{g}},d} + A_\ell e_{m_\ell,d}) - e_{m_{\mathrm{g}},d}\partial_x A_g - e_{m_\ell,d}\partial_x A_\ell. \quad (9.53)$$

Remark 9.17. *This can be considered as a special case of the extended structure shown in [16] in the context of spatially-varying cross-section. We skip the proof of Theorem 9.16 and instead refer to [16] and use similar lines of reasoning.*

We have shown Stokes-Dirac structure representations for both dissipative Hamiltonian formulations of the mathematical models under consideration.

9.6 Special case considerations for the DFM

In this section, we disqualify the DFM with the Zuber-Findlay slip conditions as an energy consistent model for two-phase flow, and, thus, motivate the reasons behind considering the DFM without slip.

We recall the dissipation inequality obeyed by the TFM (see Theorem 9.9). Under the imposition of periodic boundary conditions, the time derivative of the Hamiltonian (9.18) can be expressed using (9.3d) as follows:

$$\begin{aligned}\frac{\mathrm{d}\mathcal{H}}{\mathrm{d}t} &= -\int_\Omega (\delta_q\mathcal{H}(q))^T(\mathcal{R}_T)\delta_q\mathcal{H}(q)\ \mathrm{d}x = -\int_\Omega b_g^M\,(v_{\mathrm{g}}-v_\ell)^2\ \mathrm{d}x \\ &= -\int_\Omega M_{ig}(v_\ell - v_{\mathrm{g}})\ \mathrm{d}x \le 0.\end{aligned} \quad (9.54)$$

The equivalence between the TFM and DFM, discussed in Section 9.3, gives a better understanding of the DFM, especially when comparing the energy considerations between these two models since the only difference is how the term M_{ig} is chosen. In the TFM, it is chosen to be proportional to the slip velocity $v_\ell - v_{\mathrm{g}}$ with a non-negative coefficient of proportionality b_g^M. This linear relationship has been chosen to enforce an entropy inequality [95] and it is the basic ingredient to show that the Hamiltonian is non-increasing along solutions, see Theorem 9.9. However, to imitate the behavior of the DFM from the TFM, the expression for M_{ig} in (9.11) is much more complex and it is challenging to analytically investigate the sign of the term $\int_\Omega M_{ig}(v_\ell - v_{\mathrm{g}})\ \mathrm{d}x$ that appears in (9.54).

If the term $\int_\Omega M_{ig}(v_\ell - v_{\mathrm{g}})\ \mathrm{d}x$ is always positive, it can be claimed that the dissipation inequality $\mathrm{d}\mathcal{H}/\mathrm{d}t \le 0$ also holds for the (general) DFM (using (9.54)). It is worth recalling that the dissipation inequality $\mathrm{d}\mathcal{H}_D/\mathrm{d}t \le 0$ holds for the DFM under zero slip considerations (see Theorem 9.10).

As the theoretical assessment of the term $\int_\Omega M_{ig}(v_\ell - v_{\mathrm{g}})\ \mathrm{d}x$ for the model with non-zero slip is highly involved, we investigate its behavior numerically.

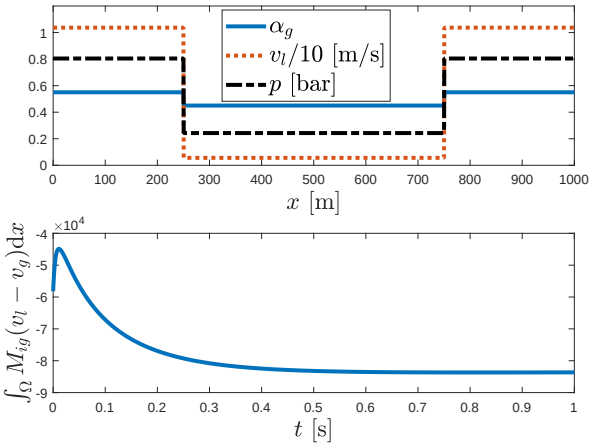

Figure 9.1: (top) Initial condition and (bottom) the temporal evolution of $\int_\Omega M_{ig}(v_\ell - v_\mathrm{g})\mathrm{d}x$ for the DFM with periodic boundary conditions.

In order to calculate μ_g, μ_ℓ and ζ as in Theorem 9.5, the same expressions as computed in [68] are used. The Rusanov scheme [152] together with Zuber-Findlay slip (with $K = 1.07$ and $S = 0.216$ m/s cf. (9.8)) is used to solve the DFM numerically in a horizontal 1000 m-long spatial domain with the spatial and temporal step size of 0.5 m and 0.0005 s, $p_{\ell 0} = 1$ bar, $\rho_{\ell 0} = 1000$ kg/m^3, $c_\ell = 1000$ m/s, and $c_g = 316$ m/s. We consider periodic boundary conditions with the initial condition as shown in Figure 9.1. We use this test case to draw a concrete conclusion on the sign of $\int_\Omega M_{ig}\,(v_\ell - v_\mathrm{g})\,\mathrm{d}x$. As obvious from Figure 9.1, we have found a counter example for which this integral is negative for all time instants.

The numerical results indicate that the proposed Hamiltonian $\mathcal{H}_D$ with periodic boundary considerations does not guarantee the non-increasing behavior of the Hamiltonian functional along solutions of the DFM. A possible underlying reason for this effect could be that the Hamiltonian (9.22) (under zero gravitational contribution) is not suitable for the DFM with the Zuber-Findlay slip. However, the Hamiltonian $\mathcal{H}_D$ has the interpretation of the energy. The increment in this energy along the solutions in principle disqualifies the DFM for such slip conditions as an energy-consistent model for two-phase flow. Hence, we do not consider the general case of the DFM and only focus on a special case of the model, i.e., the model without slip.

9.7 Conclusions

We introduced a dissipative Hamiltonian formulation for two variants of multi-phase flow models, i.e., the Two-Fluid Model (TFM) and the no-slip Drift Flux Model (DFM) across a constant cross-section. Moreover, we presented Stokes-Dirac structure representations corresponding to the skew-adjoint operators obtained both for the TFM and for the DFM without slip (under certain choice of state-variables) along with the proof of corresponding representation for the TFM. Port-Hamiltonian representations for the multi-phase models are implicitly represented in terms of the Stokes-Dirac structures. Additionally, we numerically reasoned, by exploiting a connection to the TFM, to support the consideration of the DFM without slip.

Elegantly parametrizing the boundary port-variables for a class of state-dependent Stokes-Dirac structures is one important research direction for the future. The construction of structure-preserving surrogate models will be another focus of future work. This will open up possibilities for the analysis and control of complex physical systems.

Chapter 10

Power-Preserving Interconnection of Single- and Two-Phase Flow Models for Managed Pressure Drilling

Many complex systems are modeled by a network of different subsystems, each having their underlying mathematical model representations. Energy-based modeling of each of these subsystems can yield a port-Hamiltonian (pH) representation. In this paper, a single-phase flow model, a dissipative mathematical component and a two-phase flow model are interconnected to model hydraulics for Managed Pressure Drilling (MPD) applications. These subsystems are interconnected in a power-preserving manner to build an aggregated pH system for real-life MPD scenarios. We prove that the interconnection junction connecting the single- and two-phase flow models is conditionally power-preserving.

10.1 Introduction

Port-Hamiltonian (pH) systems have recently received a lot of attention for modeling physical phenomena governed by nonlinear Partial Differential Equations (PDEs) and ordinary differential equations [64], [153]. A pH realization offers a suitable description of the components for the modeling, analysis and controller design [64].

Controllers for PDEs are generally designed for finite-dimensional state-space

This chapter is based on "M.H. Abbasi, H. Bansal, H. Zwart, L. Iapichino, W.H.A. Schilders, N. van de Wouw, ***Power-Preserving Interconnection of Single- and Two-Phase Flow Models for Managed Pressure Drilling***, *American Control Conference*, Denver, U.S.A., July 2020".

model descriptions obtained after a low-resolution spatial discretization of the PDEs, which lack part of the information (such as mass conservation) present in the infinite-dimensional representation. Next to PDE control techniques such as optimal control, backstepping [104] and adjoint methods [123], recently, researchers have been investigating control strategies for pH representations [64]. A pH framework enables controller design based on energy consideration by different techniques such as energy-shaping [118], and interconnection and damping assignment [137]. In addition, the Hamiltonian defined in pH framework represents a good candidate for the Lyapunov function, rendering the physics-based control design and the stability proof more tangible [119].

One interesting feature of pH systems is power preservation. A key property of pH systems is that the interconnection of such systems still preserve the pH structure if the interconnection is performed in a power-preserving manner [42], [64]. This compositionality feature enables to interconnect the individual pH subsystems to define an aggregated pH system. A lot of work has already been done in the past in the scope of integration of finite-dimensional and infinite-dimensional components [42], [64]. The key point in aggregating different pH subsystems is the identification of the interconnection structure and casting this interconnection into a power-preserving structure.

Hydraulics in an Managed Pressure Drilling (MPD) can be characterized by interconnection of subsystems governed by a single-phase flow in one pipe and a two-phase flow in another pipe (see Figure 10.1), and, mathematical models governed by nonlinear ordinary differential equations or static equations [5]. A single-phase flow is usually modelled by the isothermal Euler equations, which obeys a pH formalism [181]. For two-phase flow modelling, the Two-Fluid Model (TFM) and the Drift Flux Model (DFM) are typically employed [5]. Recently, it has been shown that the TFM and a DFM without slippage between the two phases can also be cast in the pH formalism [15]. Drilling with MPD is composed of single- and two-phase flow pH realizations, which can be interconnected via MPD equipment (bit) in a power-preserving manner to form an aggregated pH system.

We employ the existing theory to interconnect (individual) mathematical subsystem models to construct an aggregated model for MPD. To the best of our knowledge, compositional pH modeling for MPD is taken up for the first time in this paper. A compositional pH representation of the MPD model is useful when it needs to be connected to other systems such as a reservoir model, where each system is characterized by a particular energy property. The compositional structure, presented in this work, can be viewed as a stepping stone towards a holistic control paradigm for MPD scenarios. To the best of our knowledge, most controllers for MPD are designed based on a lumped-parameter models approximating the hydraulics and ignoring the fast pressure dynamics [5]. The framework introduced in this paper enables an energy-based controller design while taking all (infinite-dimensional) dynamics into account.

The contribution of this study is two-fold. First, a power-preserving interconnection at the junction connecting the single- and two-phase flow models are provided and, second, a power-preserving condition for a typical junction used in MPD [128] interconnecting these two models is derived. Outside this conditional power-preserving region, the interconnection junction generates power, which renders the junction model (that connects the two subsystems) non-physical.

The structure of this paper is as follows. In Section 10.2, a brief introduction to MPD is provided. In Section 10.3, single- and two-phase flow models together with their pH formalisms are introduced. In Section 10.4, the interconnection of the single- and two-phase flow models together with the conditional power preservation of the interconnection junction is discussed. In Section 10.5, the power-preserving interconnection of the systems in a real-life drilling scenario is investigated. Finally, Section 10.6 concludes the paper.

10.2 Managed pressure drilling

The industrial problem under investigation is a drilling system, with a special focus on MPD. The configuration of the system is illustrated in Figure 10.1. A drilling liquid known as mud is pumped into a pipe, called the drillstring, at high pressure. At the bottom of the drillstring, the mud leaves the drillstring through nozzles created inside the drill bit and enters the area between the drillstring and the wellbore, known as the annulus. It then flows up through the annulus and carries the rock cuttings out of the well. In MPD, the annulus is sealed off from the surroundings at the top with a Rotating Control Device (RCD in Figure 10.1) and the mud circulates out of the well through a choke valve. The circulation path of the mud can be observed by following the green arrows in Figure 10.1. Usually a flow from the formation containing gas and liquid (this formation is named reservoir in Figure 10.1) occurs at the bottomhole of the annulus, leading to multi-phase flow in the annulus. For a more comprehensive explanation of MPD, single-phase and two-phase flow modeling, we refer to [128].

Remark 10.1. *If no contingency happens during drilling, the flows inside the drillstring and annulus are of a single-phase flow nature. In case of liquid influx, we assume the reservoir produces the same liquid as the drilling mud. If a gas influx occurs in the formation, the flow inside only the annulus involves two phases.*

Notation: The following short-hand notations are used in the paper. x_a, x_d: spatial coordinate in the annulus and drillstring, $x_d \in \Omega_d = [0, L]$ and $x_a \in \Omega_a = [0, L]$: spatial domain in the drillstring and annulus, respectively. $(\cdot)^B_{L,d}$: variable $(\cdot)$ at the boundary $x_d = L$, $(\cdot)^{B,i}_{0,a}$: the i-th component of the decomposed variable $(\cdot)$ at the boundary $x_a = 0$, $(\cdot)|_{L,d}$: variable $(\cdot)$ at $x_d = L$, $(\cdot)|^{L,d}_{0,d}$:= $(\cdot)|_{L,d} - (\cdot)|_{0,d}$, $(\cdot)^{B,r,g}_{0,a}, (\cdot)^{B,r,\ell}_{0,a}$: boundary variables at the reservoir for the gas

and liquid phase, respectively. Subscripts $(\cdot)_\ell$ and $(\cdot)_g$ refer to the values for liquid and gaseous phase, respectively. $\mathbb{R}$ denotes the space of real numbers.

10.3 Port-Hamiltonian Models

In this section, we briefly introduce the isothermal Euler equations and the TFM. The pH formulation of each of these models is presented in this section. It should be noted that modeling MPD in 1D captures the most important hydraulics features of drilling [5]. Therefore, the governing PDEs and models for the MPD equipment are presented in 1D.

10.3.1 Isothermal Euler Equations

Isothermal Euler equations [113] are typically employed to model single-phase flow inside the drillstring [128]. This model encompasses a coupled mass balance and momentum balance equation. For a drillstring with constant cross-sectional area A_d and a constant inclination of the pipe θ, see Figure 10.1, under the assumption of laminar flow, the isothermal Euler equations read as follows:

$$\partial_t \rho + \partial_{x_d} (\rho v) = 0, \tag{10.1a}$$

$$\partial_t (\rho v) + \partial_{x_d} \left(\rho v^2 + p\right) = -32 \frac{\mu_\ell v}{d_d^2} + \rho g \sin\theta, \tag{10.1b}$$

where $t \in \mathbb{R}_{\geq 0}$ and x_d denote the temporal and spatial variables in the drillstring, respectively (see Figure 10.1). Variables $\rho(t, x_d)$, $v(t, x_d)$ and $p(t, x_d)$ refer to

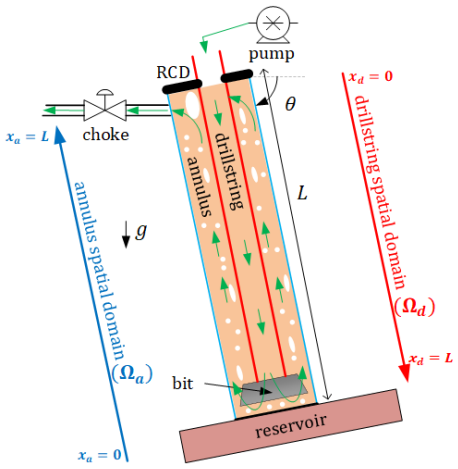

Figure 10.1: A drilling well with MPD equipment.

density, velocity and pressure of the mud inside the drillstring, respectively. Moreover, μ_ℓ and d_d denote viscosity of the mud and hydraulic diameter of the drillstring, respectively, and g is the gravitational acceleration. To complete the set of equations, an Equation of State (EOS) is provided as $p = (\rho - \rho_0)c_\ell^2 + p_0$ with constants ρ_0 and p_0 (respectively the density and pressure around which the EOS is linearized), and c_ℓ being the speed of sound in the mud.

The Hamiltonian function for (10.1) in the state variables $z := [\rho, v]^T$ is

$$\mathcal{H}(z) := A_d \int_{\Omega_d} \rho \frac{v^2}{2} + \rho c_\ell^2 \ln \rho + (c_\ell^2 \rho_0 - p_0) - \rho g x \sin\theta \, \mathrm{d}x. \tag{10.2}$$

In the following theorem, the pH formulation corresponding to (10.1) is introduced.

Theorem 10.2. *The governing equations* (10.1) *together with the EOS* $p = (\rho - \rho_0)c_\ell^2 + p_0$ *can be written in the following dissipative pH formulation:*

$$\partial_t z = (\mathcal{J}_d - \mathcal{R}_d(z))\, \delta_z \mathcal{H}(z), \tag{10.3a}$$

$$\text{with } \mathcal{J}_d = -\frac{1}{A_d}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \frac{\partial}{\partial x_d}, \mathcal{R}_d(z) = \begin{bmatrix} 0 & 0 \\ 0 & \frac{32\mu_\ell}{A_d d_d^2 \rho^2} \end{bmatrix}, \tag{10.3b}$$

where $z := [\rho,\ v]^T$ *and* $\mathcal{H}(z)$ *is given by* (10.2)*. This equation is completed by the power conjugated input* u *and output* y *at the boundaries with coupling relations as follows:*

$$\begin{pmatrix} y_{L,d}^B \\ u_{L,d}^B \end{pmatrix} = \begin{pmatrix} \frac{1}{A_d}\delta_\rho \mathcal{H}(z) \\ \delta_v \mathcal{H}(z) \end{pmatrix} |_{L,d}, \begin{pmatrix} y_{0,d}^B \\ u_{0,d}^B \end{pmatrix} = \begin{pmatrix} -\frac{1}{A_d}\delta_\rho \mathcal{H}(z) \\ \delta_v \mathcal{H}(z) \end{pmatrix} |_{0,d}. \tag{10.4}$$

Proof. Using (10.1a) and the EOS, the momentum equation (10.1b) can be written in terms of ρ and v as

$$\partial_t v = -\partial_{x_d}(\frac{v^2}{2} + c_\ell^2 \ln \rho) - 32\frac{\mu_\ell v}{d_d^2 \rho} + g \sin\theta. \tag{10.5}$$

The variational derivatives of Hamiltonian (10.2) are:

$$\delta_\rho \mathcal{H} = A_d(\frac{v^2}{2} + c_\ell^2 \ln \rho + c_\ell^2 - g x_d \sin\theta), \quad \delta_v \mathcal{H} = A_d \rho v. \tag{10.6}$$

The physical interpretation of $\delta_v \mathcal{H}$ is the mass flow rate of the mud passing spatial location x_d at time t. The energy per unit mass multiplied by the cross section A_d can be inferred from $\delta_\rho \mathcal{H}$. Equations in (10.3) are obtained by a straightforward replacement of $e_d := [\delta_\rho \mathcal{H},\ \delta_v \mathcal{H}]^T$ in (10.1a) and (10.5). Boundary conditions (10.4) are obtained by satisfying the following relation for the

time derivative of the Hamiltonian $\dot{\mathcal{H}}$ and the power through the boundaries of the drillstring $\mathcal{P}^B$,

$$\begin{aligned}\dot{\mathcal{H}} + \mathcal{P}^B = &- \int_{\Omega_d} e_d^T \ \mathcal{R}_d \ e_d \ \mathrm{d}x + \left(\frac{1}{A_d}\delta_\rho \mathcal{H}(z)\delta_v \mathcal{H}(z)\right)|_{0,d} - \\ &\left(\frac{1}{A_d}\delta_\rho \mathcal{H}(z)\delta_v \mathcal{H}(z)\right)|_{L,d} + y_{0,d}^B u_{0,d}^B + y_{L,d}^B u_{L,d}^B \\ = &- \int_{\Omega_d} e_d^T \ \mathcal{R}_d \ e_d \ \mathrm{d}x \leq 0.\end{aligned} \tag{10.7}$$

The last inequality is due to the positive semi-definite nature of $\mathcal{R}_d(z)$. □

10.3.2 Two-Fluid Model

The TFM can be defined in terms of PDEs expressing mass and momentum conservation laws for each phase in the annulus with constant cross-sectional area A_a, constant hydraulic diameter d_a and constant inclination θ as follows [168]:

$$\partial_t \left(\alpha_i \rho_i\right) + \partial_{x_a} \left(\alpha_i \rho_i v_i\right) = 0, \tag{10.8a}$$

$$\begin{aligned}\partial_t \left(\alpha_i \rho_i v_i\right) + \partial_{x_a} \left(\alpha_i \rho_i v_i^2\right) = &-\partial_x \left(\alpha_i p_a\right) + \\ &M_i - \alpha_i \rho_i g \sin\theta - 32 \frac{\mu_i \alpha_i v_i}{d_a^2},\end{aligned} \tag{10.8b}$$

where $i \in \{\ell, g\}$ and x_a is the spatial variable in the annulus (see Figure 10.1). The model contains seven unknown variables, namely, liquid and gas void fraction, α_ℓ and α_g, liquid and gas phase velocity, v_ℓ and v_g, liquid and gas phase density, ρ_ℓ and ρ_g, and the common pressure in the annulus p_a. To complete the model, we use a set of widely applied closure laws as in [68]:

$$\alpha_g + \alpha_l = 1, \tag{10.9a}$$

$$M_g + M_\ell = 0, M_g = p_a \partial_x \alpha_g + b_g^M (v_\ell - v_g), \tag{10.9b}$$

$$\rho_g = \frac{p_a}{c_g^2}, \rho_\ell = \rho_0 + \frac{p_a - p_0}{c_\ell^2}, \tag{10.9c}$$

where (10.9a) expresses that any pipe segment is occupied by the combination of gas and liquid. The terms M_g and M_l with the constant $b_g^M \geq 0$ in (10.9b) account for the force interaction between the phases. Finally, (10.9c) define the barotropic EOS of each phase with c_g the constant speed of sound in the gaseous phase.

The Hamiltonian for the flow inside the annulus with $z_a := [m_g,\ m_\ell,\ v_g,\ v_\ell]^T$ takes the following form (with $m_g := \alpha_g \rho_g$ and $m_\ell := \alpha_\ell \rho_\ell$):

$$\mathcal{H}_a(z_a) := A_a \int_{\Omega_a} m_g \frac{v_g^2}{2} + m_l \frac{v_\ell^2}{2} + m_g c_g^2 \ln \rho_g + \rho_\ell c_\ell^2 \ln \rho_\ell + \alpha_\ell (c_\ell^2 \rho_0 - p_0) - (\rho_\ell + m_g)\, g(L-x) \sin\theta \, \mathrm{d}x. \quad (10.10)$$

Notably, variables $\rho_\ell, \rho_g, \alpha_\ell$ can be written in terms of m_g and m_ℓ, see [15]. In the following theorem, the pH formulation corresponding to (10.8) and (10.9) is presented.

Theorem 10.3. *The governing equations* (10.8) *associated with the closure equations* (10.9) *can be written in the dissipative pH formulation as follows:*

$$\partial_t z_a = (\mathcal{J}_a - \mathcal{R}_a(z_a))\, \delta_{z_a} \mathcal{H}_a(z_a), \quad (10.11a)$$

$$\mathcal{J}_a = -\frac{1}{A_a} \begin{bmatrix} \mathbf{0} & I \\ I & \mathbf{0} \end{bmatrix} \frac{\partial}{\partial x_a}, \mathcal{R}_a(z_a) = \frac{1}{A_a} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tau \end{bmatrix}, \quad (10.11b)$$

where $z_a := [m_g, \ \rho_\ell, \ v_g, \ v_\ell]^T$ *and* $\mathcal{H}_a$ *is given by* (10.10), $\tau = \begin{bmatrix} \frac{b_g^M}{m_g^2} + \frac{32\mu_g\alpha_g}{m_g^2 d^2} & -\frac{b_g^M}{m_g \rho_\ell} \\ -\frac{b_g^M}{m_g \rho_\ell} & \frac{b_g^M}{m_\ell^2} + \frac{32\mu_l\alpha_\ell}{m_l^2 d^2} \end{bmatrix}$, *and* $\mathbf{0}$ *and* I *denote the* 2×2 *zero and identity matrix, respectively. This pH formulation is completed by the power conjugated inputs* u *and outputs* y *at the boundary coupled with relations as follows:*

$$\begin{pmatrix} y_{0,a}^{B,1} \\ y_{0,a}^{B,2} \\ u_{0,a}^{B,1} \\ u_{0,a}^{B,2} \end{pmatrix} = \begin{pmatrix} -\frac{1}{A_a}\delta_{m_g}\mathcal{H}_a \\ -\frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a \\ \delta_{v_g}\mathcal{H}_a \\ \delta_{v_\ell}\mathcal{H}_a \end{pmatrix} |_{0,a}, \begin{pmatrix} y_{L,a}^{B,1} \\ y_{L,a}^{B,2} \\ u_{L,a}^{B,1} \\ u_{L,a}^{B,2} \end{pmatrix} = \begin{pmatrix} \frac{1}{A_a}\delta_{m_g}\mathcal{H}_a \\ \frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a \\ \delta_{v_g}\mathcal{H}_a \\ \delta_{v_\ell}\mathcal{H}_a \end{pmatrix} |_{L,a}. \quad (10.12)$$

Proof. Rewriting momentum equations (10.8b) in terms of v_g and v_ℓ leads to

$$\partial_t v_i + \partial_{x_a}\left(\frac{v_i^2}{2}\right) = -c_i^2 \partial_{x_a}(\ln \rho_i) \pm \frac{b_g^M}{m_i}(v_\ell - v_g) - g\sin\theta - 32\frac{\mu_i v_i}{\rho_i d_a^2}, \quad (10.13)$$

where "+" is used for $i = g$ and "−" is used for $i = \ell$. Using the Hamiltonian (10.10) (for details, see [15]),

$$\begin{cases} \delta_{m_i}\mathcal{H}_a = A_a(\frac{v_i^2}{2} + c_i^2 \ln \rho_i + c_i^2 - g(L - x_a)\sin\theta), \\ \delta_{v_i}\mathcal{H}_a = A_a m_i v_i, \quad i \in \{\ell, g\}. \end{cases} \quad (10.14)$$

Similar to the isothermal Euler equations, $\delta_{v_i}\mathcal{H}_a$ represents the mass flow rate of the phase i. Similarly, The energy per unit mass of phase i multiplied by the

cross section A_a can be inferred from $\delta_{m_i}\mathcal{H}_a$. Straightforward replacement of these relations into the original equations give the asserted equations. Similar to Theorem 10.2, the boundary terms can be obtained from ($\mathcal{P}_a^B$ is the power through the boundaries of annulus)

$$\begin{aligned}\dot{\mathcal{H}}_a + \mathcal{P}_a^B = &- \int_{\Omega_a} e_a^T\ \mathcal{R}_a\ e_a\ \mathrm{d}x + \left(\frac{1}{A_a}\delta_{m_g}\mathcal{H}_a\delta_{v_g}\mathcal{H}_a\right)|_{0,a} + \\ &\left(\frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a\delta_{v_\ell}\mathcal{H}_a\right)|_{0,a} - \left(\frac{1}{A_a}\delta_{m_g}\mathcal{H}_a\delta_{v_g}\mathcal{H}_a\right)|_{L,a} - \\ &\left(\frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a\delta_{v_\ell}\mathcal{H}_a\right)|_{L,a} + y_{0,a}^{B,1}u_{0,a}^{B,1} + y_{0,a}^{B,2}u_{0,a}^{B,2} + \\ &y_{L,a}^{B,1}u_{L,a}^{B,1} + y_{L,a}^{B,2}u_{L,a}^{B,2} = - \int_{\Omega_a} e_a^T\ \mathcal{R}_a\ e_a\ \mathrm{d}x \leq 0,\end{aligned}$$

with $e_a = [\delta_{m_g}\mathcal{H}_a, \delta_{m_\ell}\mathcal{H}_a, \delta_{v_g}\mathcal{H}_a, \delta_{v_\ell}\mathcal{H}_a]^T$. □

Remark 10.4. *PH properties for the isothermal Euler equations in [181] and the TFM in [15] are proved for only unit cross section without frictional and gravitational source terms. Moreover, a different choice of the state variables than in [15] is employed to express the pH realization of the TFM.*

Remark 10.5. *It can easily be proved that $\mathcal{J}_d$ in* (10.3) *and $\mathcal{J}_a$ in* (10.11) *are formally skew-adjoint operators [64]. Moreover, $\mathcal{R}_d$ and $\mathcal{R}_a$ in the same equations are symmetric positive semi-definite.*

10.4 Power-Preserving Interconnection

In this section, the drillstring, the drill bit and the annulus are connected in a power-preserving manner. First, the boundary conditions are introduced. Then, the dissipation of energy through the bit is studied. Finally, the power-preserving structure of the interconnection junction (the bit and the summation junction at the reservoir, see Figure 10.2) is investigated.

10.4.1 Boundary conditions of the single- and two-phase flow models in MPD

The boundary conditions governing the fluid flow in the drillstring and the annulus follow from Figure 10.1. The governing equations of the pump, bit and choke are summarized in Table 10.1. For the pump, $\dot{m}_p, A_p, \rho_p$ and v_p represent the mass flow rate, the cross-sectional area, the density and the velocity of the liquid through the pump, respectively. For the bit, $\Delta p_b, \rho_b, \dot{m}_b, A_N$ and C_D denote the pressure drop over the bit, density at the drillstring side of the bit, the mass flow rate through the bit, the total area of the nozzles of the bit and

the nozzle coefficient, respectively. For the choke, $\dot{m}_c, \rho_c, p_c$ and p_0 are the mass flow rate, the density, the pressure at the choke inlet and atmospheric pressure, respectively. Finally, K_c and z_c, are the choke constant and the choke opening, respectively.

First, we define the boundary conditions for the drillstring to be used in (10.4).

Pump: At the pump location, we have $A_p = A_d, v_p = v|_{0,d}$ and $\rho_p = \rho|_{0,d}$. Hence

$$\delta_v \mathcal{H}|_{0,d} = (\rho A_d v)|_{0,d} = \dot{m}_p(t). \tag{10.15}$$

The boundary condition at the left side of the spatial domain Ω_d is assigned. Input and output variables at this boundary can be defined with (10.15) and (10.4).

Bit and reservoir: Through the bit, the mass conservation holds and the pressure drop is governed by the bit equation. Moreover, the flow that passed through the bit is then mixed with the known liquid mass flow rate $\dot{m}_\ell(t)$ and the gaseous mass flow rate $\dot{m}_g(t)$ coming out of the reservoir. Then, the mixture enters the annulus. Considering $\Delta p_b = p|_{L,d} - p|_{0,a}, \dot{m}_b = (\rho A_d v)|_{L,d}, \rho_b = \rho|_{L,d}$, we have

$$\begin{cases} p|_{L,d} - p|_{0,a} = \dfrac{1}{2\rho|_{L,d}} \left(\dfrac{\delta_v \mathcal{H}|_{L,d}}{A_N C_D} \right)^2, \\ \delta_v \mathcal{H}|_{L,d} + \dot{m}_\ell(t) = \delta_{v_\ell} \mathcal{H}_a|_{0,a}, \\ \delta_{v_g} \mathcal{H}_a|_{0,a} = \dot{m}_g(t). \end{cases} \tag{10.16}$$

Remark 10.6. *To solve the TFM, typically either 2 boundary conditions are specified at the left side of the domain and 2 at the right side or 3 boundary conditions at the left side and one at the right [70]. In this paper, we consider the 2-2 case. In* (10.16)*, one equation corresponds to the right boundary for the isothermal Euler equations in the drillstring and two equations correspond to the left boundary of the TFM in the annulus.*

Remark 10.7. *Above the bit and inside the drillstring, a non-return valve is installed to restrict the flow in one direction only from the drillstring to the*

Table 10.1: Governing equations of the pump, bit and choke.

Equipment	Governing equation
pump	$\dot{m}_p = A_p \rho_p v_p$
bit	$\Delta p_b = \frac{1}{2\rho_b} \left(\frac{\dot{m}_b}{A_N C_D} \right)^2$
Choke	$\dot{m}_c = K_c z_c \sqrt{2\rho_c (p_c - p_0)}$

annulus. When this valve is open, the pressure drop over the bit is governed by the bit equation in Table 10.1. When this valve is closed, the drillstring and the annulus become disconnected. Then, the right boundary condition for the drillstring becomes $\dot{m}_b(t) = (\rho A_d v)|_{L,d} = 0$ *and therefore* $\delta_v \mathcal{H}|_{L,d} = 0$. *The left boundary condition for the annulus changes to* $\delta_{v_\ell} \mathcal{H}_a|_{0,a} = \dot{m}_\ell(t)$ *and* $\delta_{v_g} \mathcal{H}_a|_{0,a} = \dot{m}_g(t)$.

Choke: For the TFM at the choke, we have two boundary conditions, i.e., the explicit value of gas void fraction over time and the nonlinear choke equation. Following the same procedure in [128], we rewrite $\dot{m}_c$ as the mass flow rate of the mixture, $\dot{m}_c = (\rho_\ell A_a v_\ell)|_{L,a} + (\rho_g A_a v_g)|_{L,a}$. We also replace ρ_c with the mixture density, $\rho_c = \alpha_\ell \rho_\ell + \alpha_g \rho_g$. Therefore, we have:

$$\begin{aligned} (\delta_{v_\ell} \mathcal{H}_a + \delta_{v_g} \mathcal{H}_a)|_{L,a} &= K_c z_c \sqrt{2(m_g + \rho_\ell)|_{L,a} (p_a|_{L,a} - p_0)}, \\ \alpha_g|_{L,a} &= g(t), \end{aligned} \tag{10.17}$$

where $g(t)$ is a function of time, explicitly specifying the gas void fraction at the choke.

Remark 10.8. *Notably, boundary conditions* (10.16) *and* (10.17) *form an implicit function of variational derivative of Hamiltonian* (10.2) *and* (10.10) *with respect to* z *and* z_a, *respectively.*

Remark 10.9. *For the case of 2-2 boundary conditions specified above, setting the boundary inputs to zero leads to* $\dot{\mathcal{H}}_a = 0$ *in the absence of dissipation. For the case of 3-1 boundary conditions, setting the boundary inputs to zero will not yield the same result and it is not clear how energy of the system evolves over time. This complicates the energy perspective presented in this paper for 3-1 case.*

10.4.2 Dissipativity of power through the bit

The power-preserving structure of the single-phase flow model and the TFM are shown in Section 10.3 by the corresponding dissipative pH formulation of (10.3)-(10.4) and (10.11)-(10.12).

In this section, we derive conditions under which the interconnections, shown in Figure 10.2, are power-preserving. To have power-preserving interconnections, we define the following interconnections between input ports u and output ports y of different components shown in Figure 10.2:

$$\begin{aligned} &\begin{cases} u_{d,bit} = y^B_{L,d} \\ y_{d,bit} = -u^B_{L,d} \end{cases}, \begin{cases} u_{a,bit} = y_1 \\ y_{a,bit} = -u_1 \end{cases}, \begin{cases} u_2 = y^{B,2}_{0,a} \\ y_2 = -u^{B,2}_{0,a} \end{cases}, \\ &\begin{cases} u_3 = y^{B,r,\ell}_{0,a} \\ y_3 = -u^{B,r,\ell}_{0,a} \end{cases}, \begin{cases} u^{B,r,g}_{0,a} = y^{B,1}_{0,a} \\ y^{B,r,g}_{0,a} = -u^{B,1}_{0,a} \end{cases}. \end{aligned} \tag{10.18}$$

The power preservation of all these connections can be easily checked [64], e.g., $u_2 y_2 + u_{0,a}^{B,2} y_{0,a}^{B,2} = 0$. If the power is preserved through the bit, then the entire aggregated system preserves power. Therefore, we only focus on deriving the condition of power preservation through the bit.

Remark 10.10. *Due to the directions shown in Figure 10.1, the positive direction is assumed from pump to the bit and from the bit to the choke. Therefore, the incoming power into bit flows from the drillstring and the outgoing power from the bit enters the annulus.*

The physical nature of the bit dictates the outgoing power to be less than the incoming power (see Figure 10.2)

$$\mathcal{P}_{bit} := u_{a,bit} y_{a,bit} - u_{d,bit} y_{d,bit} \leq 0. \tag{10.19}$$

The incoming power $u_{d,bit} y_{d,bit}$ is related to the physical variables of the system via (10.18). To relate the outgoing power $u_{a,bit} y_{a,bit}$ to the model-dependent variables, the power preservation across the summation junction is written as

$$\begin{aligned} &u_1 y_1 + u_2 y_2 + u_3 y_3 = 0 \xrightarrow{(10.18)} \\ &(-u_{a,bit} y_{a,bit}) + (-u_{0,a}^{B,2} y_{0,a}^{B,2}) + (-u_{0,a}^{B,r,\ell} y_{0,a}^{B,r,\ell}) = 0, \end{aligned} \tag{10.20}$$

where $u_{0,a}^{B,\mathrm{r},\ell} := -\dot{m}_\ell(t)$ and, by considering the definition of the boundary conditions in the pH formulations (10.4) and (10.12), we have $u_{0,a}^{B,2} = \dot{m}_b + \dot{m}_\ell(t)$ and $y_{0,a}^{B,2} = -\frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a|_{0,a}$. For the summation junction, we use a 1-junction principle where all outputs are equal and the summation of all inputs equal to zero. This leads to $y_{0,a}^{B,\mathrm{r},\ell} = -\frac{1}{A_a}\delta_{\rho_\ell}\mathcal{H}_a|_{0,a}$, Then by using (10.16) in (10.20), we obtain,

$$u_{a,bit} y_{a,bit} = \frac{1}{A_a}\dot{m}_b \ \delta_{\rho_\ell}\mathcal{H}_a|_{0,a}. \tag{10.21}$$

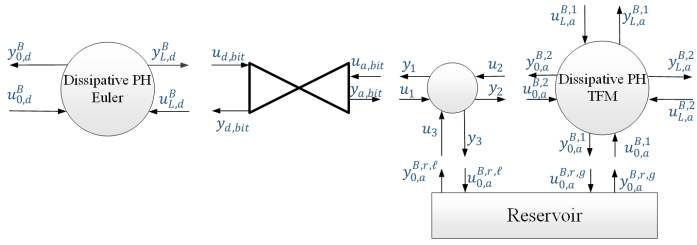

Figure 10.2: The power-preserving interconnection of different components of a drilling well.

Substituting (10.21), (10.16) into (10.19) yields

$$\mathcal{P}_{bit} = \dot{m}_b \left(\frac{1}{A_a} \delta_{\rho_\ell} \mathcal{H}_a|_{0,a} - \frac{1}{A_d} \delta_\rho \mathcal{H}|_{L,d} \right). \tag{10.22}$$

Note that when the non-return valve is closed, the two systems become isolated and the summation of the power change of both is less than the summation of the input-output conjugated energy of each pipe. As the non-return valve is open, $\dot{m}_b > 0$ holds. To ensure the power-preserving property across the bit, we must ensure $\mathcal{P}_{bit} \leq 0$. As a result of this property and by using (10.22), we have

$$\frac{1}{A_d} \delta_\rho \mathcal{H}|_{L,d} - \frac{1}{A_a} \delta_{\rho_\ell} \mathcal{H}_a|_{0,a} \geq 0. \tag{10.23}$$

Replacing the terms describing the variational derivative of Hamiltonian with respect to state variables from (10.6) and (10.14) leads to

$$\begin{aligned} &(\frac{v^2}{2} + c_\ell^2 \ln \rho - gL \sin\theta)|_{L,d} - (\frac{v_\ell^2}{2} + c_\ell^2 \ln \rho_\ell - gL \sin\theta)|_{0,a} \geq 0 \rightarrow \\ &\frac{1}{2} M_d^2 (-\frac{M_a^2}{M_d^2} + 1) + \ln \frac{\rho|_{L,d}}{\rho_\ell|_{0,a}} \geq 0, \end{aligned} \tag{10.24}$$

where $M_d = \frac{v|_{L,d}}{c_\ell}, M_a = \frac{v_\ell|_{0,a}}{c_\ell}$ are the Mach numbers of the flow at the outlet of the drillstring and at the inlet of the annulus near the bit. To further simplify the relation, we use the bit equation,

$$\Delta p_b = (\rho|_{L,d} - \rho_\ell|_{0,a}) c_\ell^2 = \frac{1}{2} \rho|_{L,d} (\frac{A_d}{A_N C_D})^2 v|_{L,d}^2 \rightarrow \tag{10.25a}$$

$$\frac{\rho_\ell|_{0,a}}{\rho|_{L,d}} = 1 - \frac{1}{2} (\frac{A_d}{A_N C_D})^2 M_d^2, \tag{10.25b}$$

The mass conservation across the bit (10.16) can also be simplified to

$$\begin{aligned} &(A_d \rho v)|_{L,d} + \dot{m}_\ell(t) = (A_a \rho_\ell v_\ell)|_{0,a} \rightarrow A_d M_d + \frac{\dot{m}_\ell(t)}{\rho c_\ell} = A_a \frac{\rho_\ell}{\rho} M_a \xrightarrow{(10.25)} \\ &M_a = \frac{A_d}{A_a} \frac{M_d}{1 - \frac{1}{2} (\frac{A_d}{A_N C_D})^2 M_d^2} + M_{\ell_r}, \end{aligned} \tag{10.26}$$

where $M_{\ell_r} := \frac{\dot{m}_\ell(t)}{A_a c_\ell \rho_\ell|_{0,a}}$ is the Mach number at the interface of the reservoir and annulus. Finally by using (10.26), the inequality (10.24) simplifies to

$$\mathcal{M} := \frac{1}{2} M_d^2 ((\frac{A_d}{A_a} \frac{1}{1 - \frac{1}{2} (\frac{A_d}{A_N C_D})^2 M_d^2} + \frac{M_{\ell_r}}{M_d})^2 - 1) + \ln(1 - \frac{1}{2} (\frac{A_d}{A_N C_D})^2 M_d^2) \leq 0. \tag{10.27}$$

For the bit, connecting the drillstring and the annulus, to be power-preserving, the inequality (10.27) should hold.

10.5 Numerical example

In this section, a real drilling well is considered and the region where the inequality (10.27) holds is investigated. The corresponding geometry and bit property are studied to define the power-preserving operational region. Outside this region, the bit model should be adjusted to abide power preservation.

Remark 10.11. *In drilling operations, the velocity inside the drillstring is typically around 1 m/s while the speed of sound in the mud is around 1000 m/s. Therefore, for drilling applications,* $M_d \approx 0.001$.

The geometry and equipment properties of the drilling platform are given by

$$\begin{aligned} &d_d = 76.2 \text{ mm}, d_{od} = 241.3 \text{ mm}, d_w = 444.5 \text{ mm}, \\ &A_N = 1418.7 \text{ mm}^2, C_D = 0.8, \end{aligned} \tag{10.28}$$

where d_{od} and d_w are, respectively, the outer diameter of the drillstring and the diameter of the wellbore.

For this well and this drill bit, to render the argument inside the logarithmic function in (10.27) to be positive, we observe that $M_d < 0.35$ should hold. As shown in the top part of Figure 10.3, in this restricted region for M_d with no flow from the reservoir (which is true in the normal drilling scenarios), (10.27) always holds for $\dot{m}_\ell = 0$ for the drilling well under consideration and the bit model is indeed dissipative (power-preserving). This might be the experimental condition under which the model for the bit was derived.

In case of contingencies, where the fluid of the reservoir flows into the annulus, condition (10.27) is not always satisfied in the restricted region for M_d, as shown in the bottom part of Figure 10.3. When the reservoir also contains liquid, the velocity of this flow should be less than the velocity of the flow coming through the bit. This situation most probably occurs when the drilling process and the mud injection are halted ($M_d = 0$) and a new pipe section is added to the drillstring to increase its length to drill further (this is called a *connection* scenario in practice). If the reservoir is producing liquid during connection, this inequality does not hold for sure. Therefore, the bit model presented in Table 10.1 must not be used to simulate the hydraulics in this situation. Notably, in cases when velocity of the flow from the reservoir is much higher than the velocity of the flow passing through the bit, usually the non-return valve is closed and the two subsystems become isolated. This situation, however, requires more investigation. These bit models are usually derived by curve fitting to experimental data obtained under certain conditions. To adapt the bit model, experiments should be designed in such a way that the inequality (10.27) is violated and a new model should be fitted to the new data.

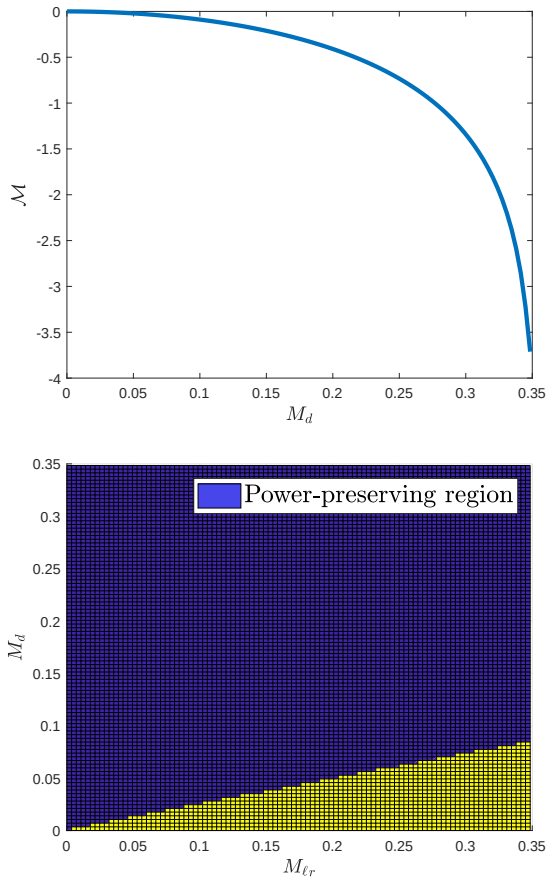

Figure 10.3: Top: The value of the function $\mathcal{M}$ in (10.27) for different admissible Mach number with $\dot{m}_\ell = 0$, Below: Power-preserving region for different Mach numbers of M_d and M_{ℓ_r} satisfying inequality (10.27).

10.6 Conclusion

In this paper, two pH models for the (single- and two-phase) flow dynamics in MPD with nonlinear boundary conditions are interconnected by a nonlinear drill bit model. To render the aggregated system power-preserving, the mathematical model of the bit, used to interconnect the two pipes, obeys power preservation under a certain condition. However, this conditional power preservation does not restrict the normal drilling operation region. The drill bit model restricts the drilling operation where liquid influx from reservoir flows into the wellbore.

In such cases, velocity of the drilling mud at the bit inside the annulus should be higher than the velocity of the liquid influx. Outside this region, the power preservation of the bit model might be violated. The framework proposed in this paper enables an energy-based controller design for MPD while taking the infinite-dimensional nature of the dynamics into account.

Chapter 11

Conclusion and Outlook

11.1 Conclusions

In this thesis, research results towards order reduction of discretized models for MPD are presented. The first step towards this goal was deriving a representative model of the hydraulics occurring during drilling with MPD. Then, this highly nonlinear, non-conservative model was discretized and numerically solved accurately. The following contributions were presented in Part I of the thesis,

1. Numerical solver dealing with the non-conservative part of the system due to the discontinuous geometry of pipes,

2. Numerical solver dealing with the detrimental effects of the source terms on the numerical solution,

3. Implementing nonlinear boundary conditions of MPD to obtain a simulator for hydraulics in MPD,

4. Validation of the MPD model with field data in case of single-phase flow, liquid influx and gas influx.

The discontinuous geometry of the wellbore increases the complexity of the hydraulics equations by adding a non-conservative term. This term cannot be handled by classical numerical methods as non-physical jumps will appear in the solution. To resolve this issue, in Chapter 2, a new numerical method was proposed to deal with this problem. The results confirm that the proposed method captures the effects of the discontinuous geometry accurately and the induced physics due to this feature is well represented in the numerical simulations.

It is well-known in the literature that the presence of source terms in the equations will induce drift of the numerical steady-state solution away from the analytical (true) steady-state solution. This is due to the fact that the source terms are treated in a trivial manner, where the detrimental effects of the source term will be canceled only by increasing the number of discretization cells. This, however, increases the computational time and the memory demand dramatically. In cases where the effect of the source terms is significant, source terms should be accounted for in a more accurate way. In Chapter 3, a new method to deal with these source terms is proposed. This method approximates the true steady-state solution with significantly higher accuracy compared to the classical methods. Moreover, the method is faster than the classical methods by using lower number of grid-cells while the same level of accuracy on the steady-state solutions of the two methods is imposed.

MPD is governed by the hydraulics in the drillstring and the annulus connected with highly nonlinear boundary conditions. Incorporating the effect of these boundary conditions is a non-trivial task. In addition, a non-return valve is installed inside the drillstring, which has to be modeled. In Chapter 4, the governing models, the numerical techniques, the incorporation of boundary conditions, and the modeling of the non-return valve are discussed. In addition, the numerical results in case of single-phase flow are validated against the field data obtained from real drilling wells. This comparison reveals that the derived models of the MPD and the components such as the non-return valve are physically valid and the numerical techniques successfully replicate the field data.

As the final chapter of Part I, the validation of the MPD model against the multi-phase flow data is carried out in Chapter 5. In this chapter, the MPD model is adapted for the case of liquid (different from the drilling mud) and gas influx phenomena. To accommodate the time-varying influx in case of liquid influx, a new reservoir model is proposed, which accurately approximates the influx flow rate if tuned correctly. It was confirmed that the numerical results are in a good agreement with the data collected from the sensors installed at real drilling rigs.

The discretized model developed in Part I is of high dimension. Exploiting this model for multi-query simulation scenarios for drilling optimization or for simulation-based controller design is by no means efficient. This brings many questions into the research framework of this thesis, which were answered in Part II by the contributions listed as below:

5. Proposing a new RB ansatz and the interconnection of the internal and boundary dynamics to capture the effect of nonlinear, state-dependent boundary conditions,

6. Quantification of accuracy loss induced by reduction in nonlinear systems with local nonlinearities,

7. Extension of the applicability region of the error estimate for systems with distributed nonlinearities,
8. Proposing an empirical hierarchical error estimate for systems with distributed nonlinearities,
9. Extension of the RB method for MPD while considering discontinuous geometry of the pipes and wellbore.

Since the MPD model in case of single-phase flow contains many parameters that vary from one simulation to another, the RB method is exploited in Chapter 6 to reduce the number of equations involved in solving the hydraulics in MPD. The presented results confirm the efficiency of this method in accelerating the solution of the MPD-relevant hydraulics. However, this acceleration always comes at a price, the accuracy loss. Due to the presence of the nonlinearities associated with this problem at local points in the spatial domain, the existing error bounds and estimates do not work well. Therefore, a new effcient error estimate for this system is proposed to quantify the error induced by the reduction.

The error estimate proposed in Chapter 6 is not tailored for nonlinear systems with distributed nonlinearities. In Chapter 7, this error estimate is extended to more general nonlinear systems by some adaptations. However, the error estimate suffers from a severe restriction when applied to nonlinear systems with nonlinearities that are not globally Lipschitz. To circumvent this issue, a hierarchical error estimate based on the snapshots of the original system is proposed, which even performs more efficiently and robustly compared to the previous error estimate.

The RB method developed in Chapter 6 is applied to the MPD model for the case of single-phase flow in Chapter 8. The reduced version of the MPD is still capable of capturing the physics occurring in MPD, especially the physics induced by the discontinuous geometry of the well. The RB method also offers high speedup values while losing little accuracy in predicting pressure in the downhole region of the wellbore.

The implementation of the RB method yields disappointing results for the multi-phase flow scenarios. This is due to the highly nonlinear nature of the dynamics. There are few alternatives to go about this problem and one is the reformulation of the problem as a pH system. This will be the stepping stone for discretization and reduction of pH systems by structure-preserving techniques. These techniques preserve some system theoretic properties of the system, such as stability and passivity, through reduction. In Chapter 9, we showed that multi-phase flow models abide the pH formalism. Then Chapter 10 was devoted to explore the compositionality feature of pH systems. The sub-models involved in the MPD model are reformulated as pH systems and interconnected through power-preserving interconnections. The model developed in this chapter, followed by a structure-preserving discretization, can then be used for structure-preserving model order reduction techniques. The reduced model obtained after

this hierarchy of procedures guarantee stability. Moreover, the model developed here can be used for controller design.

11.2 Outlook and Future Research

Although many methods have been tested, extended and tailored for the problem at hand, still many aspects of this study can be extended to fulfill the goals.

The drift flux model is not accurate in many flow regimes of the multi-phase flow [127]. In these regimes, Two-fluid model should be simulated to obtain reliable results. Moreover, the well-balanced property should be extended for the two-fluid model. The well-balanced property should be upgraded for the case where discontinuous diameter is present in the system and also for higher order of accuracy. To improve the model validation, many datasets are required from a drilling well where all scenarios have actually occurred. If these data are available, similar reservoir models can be proposed for gas influx scenario.

In this thesis, error estimates are proposed which only approximate the accuracy loss. However, error bounds are required to certify the accuracy level after reduction. Tight and cheaply computable error bounds and estimates can be a topic of future research. Moreover, accompanying RB methods with stability preservation techniques for nonlinear systems opens up many possibilities for future research.

After casting the sub-models of MPD in the pH formalism, structure-preserving techniques can be employed to discretize these pH models. During the discretization, one should give special attention to the non-polynomial nature of the Hamiltonian of the system. After finding an accurate discretization technique, to achieve a stable ROM, structure-preserving MOR techniques with the focus on the passivity and stability should be exploited.

Apart from efforts in a pH route to avoid instability of the reduction of multi-phase flow models, non-intrusive (data-based) MOR approaches can also be followed. This thread shares similarities with Machine Learning and also System Identification Theory. Through this method, data obtained from the simulation of the full-order model plays a crucial role in finding a low-dimensional state-space system and the model equations themselves are ignored in the reduction process.

As the final remark on the future research, we can refer to a hot research thread, reduction of the number of equations describing advection-dominated problems. The existing methods [135], [148], however, are far from being practical and almost all can be applied only to problems with periodic boundary conditions. These problems have no counterpart in real-world applications. Moreover, most of the existing techniques apply only to systems governed by a single PDE and fail for systems with coupled PDEs. However, if these severe restrictions of these techniques are resolved, they can be applied to the MPD test case.

Appendix

A Initial conditions away from steady state

As mentioned in Section 2.3.4, for initial conditions far from the steady-state solution or simulating dynamics including abrupt changes in the input variables, the model-based modification (the fourth approach proposed in this paper) does not perform well. Due to the fact that equations in the DFM (2.6) are rather complicated, it is hard to find a framework for defining the applicability region of this method. Below, a brief explanation and an example are given for the case when the model-based modification cannot be applied.

Different from the results presented in the main text, here we analyze the algebraic constraints (2.34) in more detail and provide a qualitative insight why the algebraic constraints do not have a solution when the initial conditions for system (2.6) are far away from the steady-state solution. Similar analysis can be performed on the algebraic constraints (2.31).

To find the solution of the algebraic constraints, seven equations (three from (2.34) and four from closure laws of (2.2)-(2.5)) should be solved simultaneously to find the seven unknowns of primitive variables. Due to the nonlinearity of equations, these may have more than one solution or may not have any solution. This is investigated in this appendix by applying the following steps:

1. Since there are seven unknown variables in the equations, it is challenging to analyze the solutions. As a first step, we express the equations only in one variable, particularly in pressure.

2. The resulting single equation is a nonlinear function of pressure, which is hard to be analyzed. Therefore, the solution of that equation is investigated numerically.

Assume we aim to find the starred values $U^*_{M_1}$ from U_{M_1} by substituting (2.35a) and (2.35b) into (2.35c) and keeping pressure as the only variable. For reducing the number of variables and simplifying the equations, no slip between the

phases, i.e., $\hat{u} := u_l = u_g$, is considered (it should be noted this is just an assumption for simplifying the computational procedure while the algebraic constraints of (2.34) are derived for general case of slip). Then, the simplified algebraic constraints of (2.35) changes to:

$$(\alpha_l \rho_l \hat{u})^*_{M_1} = B, \tag{1a}$$

$$(\alpha_g \rho_g \hat{u})^*_{M_1} = C, \tag{1b}$$

$$(\alpha_l \frac{\rho_l}{\rho_g} \frac{\hat{u}^2}{2} + \alpha_g \frac{\hat{u}^2}{2} + c_g^2 \ln \rho_g)^*_{M_1} = D, \tag{1c}$$

with

$$B := \frac{(\alpha_l \rho_l \hat{u} A)_{M_1}}{A_{M_2}} \tag{2a}$$

$$C := \frac{(\alpha_g \rho_g \hat{u} A)_{M_1}}{A_{M_2}} \tag{2b}$$

$$D := (\alpha_l \frac{\rho_l}{\rho_g} \frac{\hat{u}^2}{2} + \alpha_g \frac{\hat{u}^2}{2} + c_g^2 \ln \rho_g)_{M_1}. \tag{2c}$$

By embedding (1a) and (1b) into (1c) and using the closure laws of (2.2)-(2.5) and keeping only pressure as the variable (denoting $p^*_{M_1}$ by p), a nonlinear function of pressure is obtained as follows:

$$\Phi(p) := \frac{1}{2}\varphi_1(p)\varphi_2^2(p) + c_g^2 \ln \frac{p}{c_g^2} - D, \tag{3}$$

where

$$\varphi_1(p) := \frac{\frac{p}{c_l^2} + F}{\frac{p}{c_g^2}} + \frac{C}{B}\left(\frac{p}{c_l^2} + F\right) \frac{1}{\frac{p}{c_g^2} + \frac{C}{B}\left(\frac{p}{c_l^2} + F\right)} \left(1 - \frac{\frac{p}{c_l^2} + F}{\frac{p}{c_g^2}}\right), \tag{4a}$$

$$\varphi_2(p) := \frac{B\frac{p}{c_g^2} + C\left(\frac{p}{c_l^2} + F\right)}{\frac{p}{c_g^2}\left(\frac{p}{c_l^2} + F\right)}, \tag{4b}$$

with $F := \rho_0 - p_0/c_l^2$. Now, the solution of the algebraic constraints of (2.35) correspond to the root of the function Φ in (3). Although function Φ is highly nonlinear and analytical investigation of the roots is challenging, analyzing the roots of the function Φ is easier compared to investigating the solution to the constraints (2.35). It can be shown that this equation either has one root, two roots or no root at all. Analytical investigation of the condition of having a root,

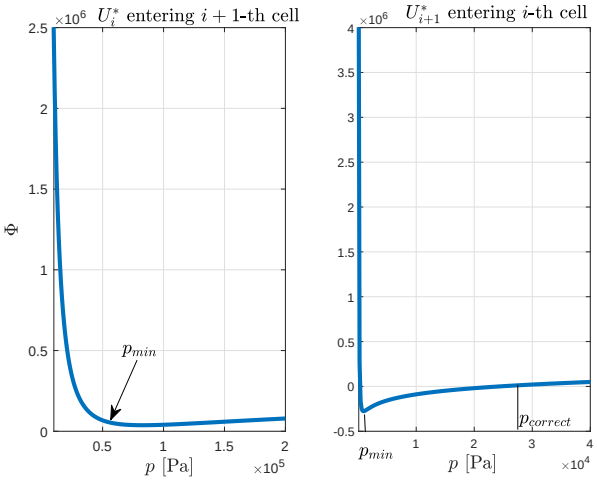

Figure 1: Evolution of Φ function with pressure.

however, is difficult, if not impossible. As the only variable in the function Φ is pressure, the roots can be found numerically by varying the pressure from zero to infinity and then we can analyze how many roots the function Φ has under different conditions.

This function tends to infinity as the pressure tends to zero or infinity when $B, C, D > 0$. In other words, for $p \to 0$ or $p \to \infty$, it holds that $\Phi \to \infty$. Thus, for Φ to have any real roots, the minimum of function Φ should be less than zero, which is highly dependent on the constant values of B, C, D, apparent from the dependence of the function on these constants. Hence, the necessary and sufficient condition for having solution(s) to the introduced algebraic constraints is $\Phi(p_{min}) < 0$ with $\Phi'(p_{min}) = 0$, where p_{min} is the unique value at which the minimum of Φ is attained. As mentioned previously, the analytical assessment of the roots of Φ is challenging, thus we tackle this numerically as follows with an example.

Consider the following test case as an example to show the fact that those algebraic constraints sometimes have multiple solutions and sometimes do not have any solution. If starting from the initial condition of shock tube problem presented in Section 2.4.2 accompanied with discontinuous area reduction at the middle cell of the pipe, say at the i-th cell, from $A_i = 1.5$ m^2 to $A_{i+1} = 1$ m^2, the evolution of function Φ with respect to p (as pressure varies from zero to infinity) for finding U_i^* and U_{i+1}^* values can be shown in Figure 1.

It can be noted that for finding the U_i^*, we do not have any solution, therefore the model-based modification fails to predict the solution (see the left plot in Figure 1 that function Φ does not intersect the horizontal axis). In case of DFM,

due to the complexity of the equations, it is hard to find a generic condition under which the constraint equations have solutions; unlike the case for Euler equations for which the condition is derived in [103]. However, a qualitative insight in the condition for the existence of a numerical solution can be provided as below. If we start exactly from steady-state solution, since the three algebraic constraints of (1) are defined by the steady-state equations, these equations are already satisfied. If starting close to steady states, the residuals of these equations are small and we are still in the region where the algebraic constraints can be satisfied by small changes in the primitive variables. In contrary, if we start from an initial condition far from steady-states, the residual of the algebraic constraints are large and it might be impossible to satisfy all seven equations simultaneously (in other words, $\Phi(p_{min}) > 0$ due to the values of B, C, D in (3)). Therefore, it is probable not to obtain solution if we start from inconsistent initial condition, i.e., initial conditions far away from the steady-state solution. Analogous analyses can be carried out for abrupt dynamic changes in the simulation.

For U^*_{i+1}, we have two solutions (see right plot in Figure 1). In case of having two solutions, one solution corresponds to the subsonic flow and the other to the supersonic flow. Following the same admissibility criterion of [103], a root which lies in the same side of p_{min} is chosen, meaning that

$$(p_{correct} - p_{min})(p_{i+1} - p_{min}) > 0 \tag{5}$$

where $p_{correct}$ is the correct root of function Φ that should be selected. Relying on this analysis, the second root should be selected in this simulation and the first root is ruled out.

Since in some cases, especially when the imposed assumptions (2.33) are not valid, we may not have the solutions of those algebraic relations, it is highly recommended to use the numerical method proposed in this paper when we start close to steady-state solution of the PDE. In industrial applications, it is often the case that system is simulated from steady states (or from rest) and the underlying dynamics are some perturbations to the steady-state solution. So, this is not a restrictive assumption in many cases. However, satisfying those algebraic relations to simulate the correct transient behavior comes at the expense of losing the solution in problems starting far from steady state. We leave further analysis of finding a better approach to future works. One tentative remedy could be a hybrid approach, combining the third and fourth approach proposed in this paper.

B Reference solution for the transient simulation

As there is no Riemann solution for the DFM with variable cross-section, another method for having a reference solution for transient case is used. In this method, at the location of area discontinuity, the pipe is divided into two different pipes that are connected to each other by some boundary conditions. For instance, the pipe in Figure 2.4 is broken into two different connected pipes, as shown in Figure 2.

Within each smaller pipe, classical schemes with high resolution can be applied as each pipe has a constant area. At the interconnected boundary, at least six equations should be defined to obtain three primitive variables for each pipe. Then, by using the closure laws (2.2)-(2.5), all primitive variables can be obtained.

From the left pipe L^p, two characteristic-based boundary conditions corresponding to the waves moving downstream similar to (2.16) and (2.17) are written. From the right pipe R^p, one characteristic-based boundary condition moving upstream can be written similar to (2.18). Three more equations are required, which are obtained by integrating system (2.6) over the area discontinuity,

$$\int_{\delta x} \frac{\partial(\alpha_l \rho_l A)}{\partial t} dx + \int_{\delta x} \frac{\partial(\alpha_l \rho_l u_l A)}{\partial x} dx = 0, \tag{6a}$$

$$\int_{\delta x} \frac{\partial(\alpha_g \rho_g A)}{\partial t} dx + \int_{\delta x} \frac{\partial(\alpha_g \rho_g u_g A)}{\partial x} dx = 0, \tag{6b}$$

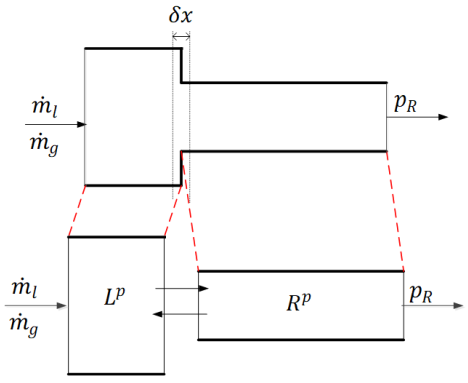

Figure 2: Decomposition of pipes into many connected smaller pipes.

$$\int_{\delta x} \frac{\partial\Big((\alpha_l\rho_l u_l + \alpha_g\rho_g u_g)A\Big)}{\partial t}dx + \int_{\delta x} \frac{\partial\Big((\alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2 + p)A\Big)}{\partial x}dx = \int_{\delta x} p\frac{\partial A}{\partial x}dx, \tag{6c}$$

where δx is a very narrow band around each area discontinuity. For $\delta x \to 0$, the integrals related to the time derivations vanish and (6) simplifies to the following algebraic relations

$$(\alpha_l\rho_l u_l A)_{R^p} = (\alpha_l\rho_l u_l A)_{L^p}, \tag{7a}$$

$$(\alpha_g\rho_g u_g A)_{R^p} = (\alpha_g\rho_g u_g A)_{L^p}, \tag{7b}$$

$$\int_{\delta x}\left(\frac{\partial\left((\alpha_l\rho_l u_l^2 + \alpha_g\rho_g u_g^2)A\right)}{\partial x} + A\frac{\partial p}{\partial x}\right)dx = 0, \tag{7c}$$

where the subscript R^p and L^p refer to the right and left pipe. For no-slip condition $u_l = u_g = \hat{u}$, based on Lemma 2.5 and the algebraic equation (2.31c), we have

$$(\alpha_l\rho_l\hat{u}A)_{R^p} = (\alpha_l\rho_l\hat{u}A)_{L^p}, \tag{8a}$$

$$(\alpha_g\rho_g\hat{u}A)_{R^p} = (\alpha_g\rho_g\hat{u}A)_{L^p}, \tag{8b}$$

$$\left(\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)_{R^p} = \left(\alpha_l\rho_l\hat{u}A(\frac{\hat{u}^2}{2} + c_l^2\ln\rho_l) + \alpha_g\rho_g\hat{u}A(\frac{\hat{u}^2}{2} + c_g^2\ln\rho_g)\right)_{L^p}. \tag{8c}$$

The other three equations have also been obtained. Then,the six boundary equations are complete and the simulation in each pipe can be solved with a high resolution Rusanov scheme to deliver the reference solution.

C Approximate solution for turbulent flows

In this section, we present an approximate solution to (4.6) based on a first-order Taylor series expansion. From (4.6), we define

$$H(\nu) := \frac{1}{\sqrt{\nu}} + 4\log\left(\frac{0.27\varepsilon}{D_{eff}} + \frac{1.26^{n_m^{-1.2}}}{\left(\nu^{\left(1-\frac{n_m}{2}\right)}\mathrm{Re}\right)^{n_m^{-0.75}}}\right), \tag{9}$$

such that $H(\nu) = 0$. We take

$$\nu = \nu_0 - \Delta\nu, \tag{10}$$

where

$$\nu_0 = \frac{1}{16\left(\log\left(\frac{0.27\varepsilon}{D_{eff}} + \frac{5.74}{\mathrm{Re}^{0.9}}\right)\right)^2}, \tag{11}$$

is a well-known approximate solution to Colebrook equation, which is recovered from (9) for $n_m = 1$, see [164]. Moreover, $\Delta\nu$ is a parameter that is to be approximated. Now, using a Taylor expansion of $H(\nu)$ around ν_0, we obtain

$$H(\nu) \approx H(\nu_0) - H'(\nu_0)\Delta\nu, \tag{12}$$

where $H'(\nu_0) = \frac{dH}{d\nu}(\nu_0)$ is available analytically. Following (12), designing $\Delta\nu$ as

$$\Delta\nu = \frac{H(\nu_0)}{H'(\nu_0)}, \tag{13}$$

leads to $H(\nu_0 - \Delta\nu) \approx 0$. Substituting this design of $\Delta\nu$ into (10) further leads the following explicit approximation for the friction factor:

$$\nu \approx \nu_0 - \frac{H(\nu_0)}{H'(\nu_0)}. \tag{14}$$

One may expand this equation to obtain a closed-form description of it to reduce the computational burden during simulations. Our numerical evaluations verified the high accuracy of this approximate solution over a wide range of Reynold's numbers and $0.6 < n_m < 1.4$.

D Derivation of the dynamical bit equation

The bit equation is obtained by considering a control volume filled with only liquid over the bit, as illustrated in Fig. 3, and averaging the momentum conservation equation for liquid over it. For the first and second half of this control volume, each of the length $\Delta l/2$, located in the drilling and annulus, we respectively obtain

$$\frac{\Delta l}{2}\frac{dz(t)}{dt} = A_d(L)\Bigg(\frac{z^2(t)}{A_d(L)\rho(t,L-\Delta l/2)} - \frac{z^2(t)}{A_d(l)\rho(t,L)} + P_d(t,L-\Delta l/2) - P_d(t,L)\Bigg) + \bar{s}_d(t), \tag{15}$$

$$\frac{\Delta l}{2}\frac{dz(t)}{dt} = -A_a(0)\Bigg(\frac{z^2(t)}{A_a(0)\rho_a(t,\Delta l/2)} - \frac{z^2(t)}{A_a(0)\rho_a(t,0)} + P_a(t,\Delta l/2) - P_a(t,0)\Bigg) + \bar{s}_a(t), \tag{16}$$

where $z(t)$ is an approximation of the average mass flow rate in this control volume, which also gives an approximation of the flow through the bit. Note that, we have also assumed a single-phase flow in the half of this control volume that is located in the annulus. Moreover, $\bar{s}_d = A_d(L)\int_{L-\Delta l/2}^{L} s_d(u^d,t,x)\,dx$ and $\bar{s}_a = A_a(0)\int_0^{\Delta l/2} s_a(u^a,t,x)\,dx$. These terms can be approximated as follows:

$$\bar{s}_d(t) \simeq A_d(L)\frac{\Delta l}{2}s_d\left(U_{d,N},t,L-\frac{\Delta l}{2}\right), \quad \bar{s}_a(t) \simeq A_a(0)\frac{\Delta l}{2}s_a\left(U_{a,1},t,\frac{\Delta l}{2}\right). \tag{17}$$

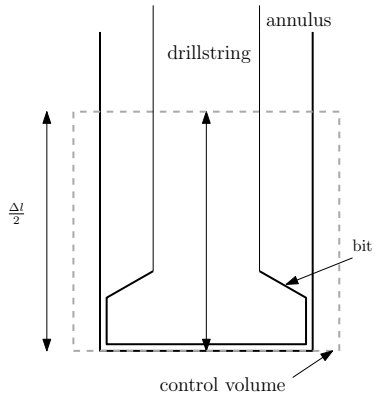

Figure 3: A schematic of the control volume assumed over the bit to facilitate solving the boundary equations at the bit.

Next, considering that $\rho(t, L - \Delta l/2) \simeq \rho(t, L)$ and $\rho_a(t, \Delta l/2) \simeq \rho_a(t, 0)$, subtracting (15) from (16) results in

$$A_a(0)P_a(t,0) + A_d(L)P_d(t,L) = A_a(0)P_a(t,\Delta l/2) + A_d(L)P_d(t,L-\Delta l/2) - \bar{s}_d(t) + \bar{s}_a(t). \tag{18}$$

Now, given the bit equation in (4.18), we have

$$P_d(t,L) - P_a(t,0) = \frac{1}{2\rho(t,L)}\left(\frac{z(t)}{c_d A_n}\right)^2, \quad z > 0. \tag{19}$$

If we solve (18) and (19) for $P_d(t, L)$ and $P_a(t, 0)$ and substitute the solution into the summation of (15) and (16), we obtain the bit equation (4.39).

E An error estimate for the linearized isothermal Euler equations

In Section 6.5.1.1, the procedure for the computation of the error bound and error estimate for the advection equation (6.27) was elaborated. This equation consists of a linear system with time-varying boundary conditions. In this section, the procedure for computing the error estimate for the Euler equations (6.31) is illustrated. This test case is also an internally linear system but with time-varying, nonlinearly state-dependent boundary conditions. The pump boundary located at $x_\partial = 0$ and the choke boundary at $x_\partial = L$ lead to nonlinear time-varying boundary conditions. Both of the boundary conditions are coupled with the states of the system. As the Lipschitz constant for the local nonlinearity at $x_\partial = L$ is not exactly known and has to be estimated, the error bound is not a true upper bound and is somehow an estimate of the upper bound. Due to this fact, we just propose an error estimate for this test case. The entire procedure for the error estimation computation is detailed below.

Following the same idea for advection equation, we can derive the error dynamics for the Euler equations. Pursuing the same steps explained in Section 6.4.1, the following error dynamics is obtained:

$$\Sigma_{nl}: \begin{cases} \Sigma^e_{lin}: \begin{cases} e^{n+1}_\rho = A_{11}e^n_\rho + A_{12}e^n_m + B_1 e^n_{V_1} + F_1 e^n_{W_2} - \mathcal{R}^n_\rho, \\ e^{n+1}_m = A_{21}e^n_\rho + A_{22}e^n_m + B_2 e^n_{V_1} + F_2 e^n_{W_2} - \mathcal{R}^n_m, \\ e^n_y = C_y[e^T_\rho \ e^T_m]^T, \\ e_z = [e^n_{\rho_1} \ e^n_{\rho_{N_\delta}} \ e^n_{m_1} \ e^n_{m_{N_\delta}}]^T = C_z[e^T_\rho \ e^T_m]^T, \end{cases} \\ \Sigma^e_{BC}: \begin{cases} x_\partial = 0: \begin{cases} e^{n+1}_{V_2} = \alpha_1 e^n_{V_2} + \alpha_2 e^n_{\rho_1} + \alpha_3 e^n_{m_1}, \\ e^n_{V_1} = Q^n e^n_{V_2}, \end{cases} \\ x_\partial = L: \begin{cases} e^{n+1}_{W_1} = \beta_1 e^n_{W_1} + \beta_2 e^n_{\rho_{N_\delta}} + \beta_3 e^n_{m_{N_\delta}}, \\ e^n_{W_2} = h(W^{n+1}_1, z^n_c) - h(\hat{W}^{n+1}_1, z^n_c), \end{cases} \\ e^n_{U_{BC}} = C_{BC}[(e^n_V)^T \ (e^n_w)^T]^T. \end{cases} \end{cases} \tag{20}$$

Here, ρ is a vector of averaged densities within the grid cells, and $m = \rho v$ is momentum. Next, V_i and $W_i, i \in \{1, 2\}$, are the expressions emerging from the finite-volume discretization necessary for solving the boundary conditions. It should be noted that $W^n_1 = e^n_{W_1} + \hat{W}^n_1$. Moreover, $e_{(\cdot)} = (\cdot) - \hat{(\cdot)}$ is the difference between the full-order and reduced-order solution. The variable Q^n represents the time variation related to the pump boundary condition (6.33) and the function $h(\cdot, \cdot)$ encompasses the effect of the nonlinear choke boundary condition (6.34).

In addition, α_i and $\beta_i, i \in \{1, 2, 3\}$ are some constants and A_{ij}, B_i and $F_i, i, j \in \{1, 2\}$ are matrices generated as the result of the finite-volume discretization. $\mathcal{R}_{(\cdot)}$ is the residual attributed to the equation solving for the variable $(\cdot)$. The nonlinearities and time variation associated with the boundary conditions does not fit into the LTI representation of the system. Then, by the

interconnection in Figure 6.3, the effects of the boundaries are included in the system as auxiliary inputs. To make a connection between (20) and (6.12), set

$$\begin{aligned} e &= \begin{bmatrix} e_\rho \\ e_m \end{bmatrix}, A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, B = \begin{bmatrix} B_1 & F_1 \\ B_2 & F_2 \end{bmatrix}, w = \begin{bmatrix} V_1 \\ W_2 \end{bmatrix}, \\ V &= \begin{bmatrix} V_2 \\ W_1 \end{bmatrix}, G = \begin{bmatrix} \alpha_1 V_2^n + \alpha_2 \rho_1^n + \alpha_3 m_1^n \\ \beta_1 W_1^n + \beta_2 \rho_{N_\delta}^n + \beta_3 m_{N_\delta}^n \end{bmatrix}, \mathcal{G} = \begin{bmatrix} Q^n V_2^n \\ h\left(W_1^n, z_c^n\right) \end{bmatrix}. \end{aligned} \tag{21}$$

Finally, C_z and C_{BC} in (20) are selected to generate the required auxiliary outputs and C_y is defined by the user to yield the output of interest. Here, the pressure at the choke is chosen as the output. A good indicator for the pressure is density, therefore $C_y = [\mathbf{0}_{N_\delta - 1},\ 1,\ \mathbf{0}_{N_\delta}]$, where $\mathbf{0}_{N_\delta} \in \mathbb{R}^{1 \times N_\delta}$ is a vector of zeros.

Considering the following state of vectors and rewriting the whole error dynamics (20), we have

$$\begin{cases} \bar{e} = \begin{bmatrix} e_{V_2} & e_\rho^T & e_m^T & e_{W_1} \end{bmatrix}^T, \\ \bar{e}^{n+1} = \bar{A}\bar{e}^n + B_{\mathcal{R}}\mathcal{R}^n + \bar{B}_{V_1} e_{V_1}^n + \bar{B}_{W_2} e_{W_2}^n, \\ \bar{e}_y^n = \bar{C}_y \bar{e}^n, \end{cases} \tag{22}$$

with

$$\begin{aligned} \bar{A} &= \begin{bmatrix} \alpha_1 & \alpha_2 E_1 & \alpha_3 E_1 & 0 \\ \mathbf{0} & A_{11} & A_{12} & \mathbf{0} \\ \mathbf{0} & A_{21} & A_{22} & \mathbf{0} \\ 0 & \beta_2 E_N & \beta_3 E_N & \beta_1 \end{bmatrix}, B_{\mathcal{R}} = \begin{bmatrix} \mathbf{0} \\ -I_{2N_\delta \times 2N_\delta} \\ \mathbf{0} \end{bmatrix}, \bar{B}_{V_1} = \begin{bmatrix} 0 \\ B_1 \\ B_2 \\ 0 \end{bmatrix}, \\ \bar{B}_{W_2} &= \begin{bmatrix} 0 \\ F_1 \\ F_2 \\ 0 \end{bmatrix}, \mathcal{R} = \begin{bmatrix} \mathcal{R}_\rho \\ \mathcal{R}_m \end{bmatrix}, \end{aligned} \tag{23}$$

where $\mathbf{0}$ is a zero vector with appropriate dimension and $E_1 = [1\ 0\ \cdots\ 0]$, $E_N = [0\ \cdots\ 0\ 1] \in \mathbb{R}^{N_\delta}$. The nonlinear part then includes only the static function $\mathcal{G}$ as in (21).

Relying on the same idea of (6.13) and decomposing the inputs of the linear system into e_{V_1} and e_{W_2}, the error bound for (22) is obtained as

$$\|\bar{e}_y\|_{\ell_2} \leq \gamma^{\bar{e}_y \mathcal{R}} \|\mathcal{R}_s\|_{\ell_2} + \gamma^{\bar{e}_y V_1} \|e_{V_1}\|_{\ell_2} + \gamma^{\bar{e}_y W_2} \|e_{W_2}\|_{\ell_2}. \tag{24}$$

If we can compute the right-hand side of (24), we can use (6.23) to compute the error estimate. To this end, we need the following relations (note that Q^n is a scalar),

$$\begin{aligned} \left\|e_{V_1}^n\right\| \leq |Q^n| \left\|e_{V_2}^n\right\| \quad &\rightarrow \|e_{V_1}\|_{\ell_2} \leq \|Q\|_{\ell_\infty} \|e_{V_2}\|_{\ell_2}, \\ \left\|e_{W_2}^n\right\| \leq \eta_{W_2 W_1} \left\|e_{W_1}^n\right\| \quad &\rightarrow \|e_{W_2}\|_{\ell_2} \leq \eta_{W_2 W_1} \|e_{W_1}\|_{\ell_2}, \end{aligned} \tag{25}$$

where $\eta_{W_2W_1}$ is the local Lipschitz constant of the nonlinear function $h(\cdot,\cdot)$ with respect to its first argument, which is defined similar to (6.15). As mentioned earlier, due to this approximation of the Lipschitz constant, we do not calculate any error bound for the Euler equations and only calculate an error estimate.

Remark .1. *An empirical alternative (which is not tested in this paper) for computing the Lipschitz constant can be provided due to the fact that we have the truth solution in the offline phase. In the offline phase, for the set of parameters μ^*, $\eta_{W_2W_1}$ can be found. One can use the Lipschitz constant either for the last selected set of parameters or the maximum Lipschitz constant for all the previously selected set of parameters.*

For the sake of notation, henceforth, we set $\eta := \|\eta_{W_2W_1}\|_{\ell_\infty}$ and $Q := \|Q\|_{\ell_\infty}$. Thus, we have:

$$\begin{aligned}\|e_{W_1}\|_{\ell_2} &\leq \gamma^{W_1\mathcal{R}} \|\mathcal{R}_s\|_{\ell_2} + \gamma^{W_1W_2} \|e_{W_2}\|_{\ell_2} + \gamma^{W_1V_1} \|e_{V_1}\|_{\ell_2}\,,\\ \|e_{W_2}\|_{\ell_2} &\leq \eta\gamma^{W_1\mathcal{R}} \|\mathcal{R}_s\|_{\ell_2} + \eta\gamma^{W_1W_2} \|e_{W_2}\|_{\ell_2} + \eta\gamma^{W_1V_1} \|e_{V_1}\|_{\ell_2}\,.\end{aligned} \tag{26}$$

Then,

$$\|e_{W_2}\|_{\ell_2} \leq \frac{\eta}{1-\eta\gamma^{W_1W_2}} \left(\gamma^{W_1\mathcal{R}} \|\mathcal{R}_s\|_{\ell_2} + \gamma^{W_1V_1} \|e_{V_1}\|_{\ell_2}\right). \tag{27}$$

Similarly, for e_{V_1}, we have

$$\|e_{V_1}\|_{\ell_2} \leq \frac{Q}{1-Q\gamma^{V_2V_1}} \left(\gamma^{V_2\mathcal{R}} \|\mathcal{R}_s\|_{\ell_2} + \gamma^{V_2W_2} \|e_{W_2}\|_{\ell_2}\right). \tag{28}$$

Finally, by inserting (27) into (28), we get

$$\|e_{V_1}\|_{\ell_2} \left(1-\sigma_1\sigma_2\gamma^{V_2W_2}\gamma^{W_1V_1}\right) \leq \sigma_1 \left(\gamma^{V_2\mathcal{R}} + \sigma_2\gamma^{V_2W_2}\gamma^{W_1\mathcal{R}}\right) \|\mathcal{R}_s\|_{\ell_2}\,, \tag{29}$$

where

$$\begin{aligned}\sigma_1 &= \frac{Q}{1-Q\gamma^{V_2V_1}},\\ \sigma_2 &= \frac{\eta}{1-\eta\gamma^{W_1W_2}}.\end{aligned} \tag{30}$$

For using this bound, these norms in Algorithm 5 should be calculated: $\gamma^{\bar{e}_y\mathcal{R}}, \gamma^{\bar{e}_yV_1}, \gamma^{\bar{e}_yW_2}, \gamma^{W_1\mathcal{R}}, \gamma^{W_1W_2}, \gamma^{W_1V_1}, \gamma^{V_2\mathcal{R}}, \gamma^{V_2W_2}$. Due the special structure of the model and also parameter settings, in all simulations we observe that $\sigma_1 \approx 1$ and $\gamma^{V_2W_2}\gamma^{W_1V_1} \approx 1$. Then, the conservativeness and applicability of the error estimate are highly dependent on σ_2 and hence on η and, consequently, on the choke operating condition. By having the upper bound on $\|e_{V_1}\|_{\ell_2}$, the upper bounds on $\|e_{W_2}\|_{\ell_2}$ and then $\|\bar{e}_y\|_{\ell_2}$ are computable by subsequently using

(28) and (27) into (24). It should be noted that for using the error estimate, the condition

$$1 - \sigma_1 \sigma_2 \gamma^{V_2 W_2} \gamma^{W_1 V_1} > 0, \tag{31}$$

should hold, which depends on the system and boundary condition properties such as pump and choke characteristics; otherwise, the error estimate cannot be used (by applicability we mean that the condition (31) should be satisfied on the solution manifold). This condition originates from the small-gain condition on the interconnection of the system dynamics with the boundary dynamics [27], see Figure 6.3.

To resolve this issue, we propose to use a hybrid error estimate; meaning that at those regions that the proposed error estimate does not exist due to violating the small-gain condition, we can switch to the error estimate presented in [182]. In the test cases of this paper, the condition (31) was always satisfied and this hybrid method has not been used. Other approaches will be investigated in future works.

Summary

Modeling and Order Reduction for Hydraulics Simulation in Managed Pressure Drilling

Societal uses for the drilling of deep wells are abundant and without exception have enormous impact on global economies. The future sustainability of the harvesting of these resources requires the exploitation of difficult-to-access, unconventional reserves. The drilling of deep wells for these purposes is characterized by high complexity, high uncertainty, high risk and high cost. In particular, one threat for the economically feasible and environmentally safe harvesting of such energy and mineral reserves is pressure control. Maintaining the downhole pressure within the pressure limits of the well, i.e., between the pore and fracture pressure, is critical for the safety of drilling operations. To successfully control the downhole pressure, a drilling technique called Managed Pressure Drilling (MPD) has been recently developed. In support of MPD application, advanced tools for virtual drilling scenario testing are needed, especially during the drilling operation to evaluate the effects of a potential action. The main objectives of this thesis are: i) developing a new hydraulics model for MPD together with new numerical techniques, ii) developing efficient model order reduction techniques with error estimates, and iii) introducing port-Hamiltonian (pH) formulations of the models to preserve key properties after reduction.

To develop a virtual drilling well, a hydraulics model for MPD operations should be developed. The hydraulics model for MPD consists of a single- and a multi-phase flow model connected through nonlinear equations describing the MPD equipment. The obtained model, however, suffers from many different mathematical complexities, rendering the classical numerical techniques inadequate to discretize and numerically simulate such a model. Therefore, different numerical techniques have been developed or extended to support numerical simulation with the aim to: i) capture the physics induced by a discontinuous well geometry, ii) have a significantly more accurate steady-state solution in the presence of friction and gravity, and iii) incorporate the nonlinear state-dependent boundary conditions. Comparisons of the numerical solutions against the field

data gathered from real-world drilling wells reveal the validity of the model and also the accuracy of the numerical solutions.

The discretized model for MPD is typically computationally expensive to be run in real time, rendering the numerical model not suitable for simulation-based controller design. Moreover, while optimizing the design process of the drilling procedure, any change in the parameters of the system, such as the geometry of the well, requires performing many expensive and time-consuming simulations. To overcome these issues, model order reduction of the parameterized model for MPD is beneficial. The Reduced Basis (RB) method is utilized for the reduction of single-phase flow models. Nonetheless, the RB method is tailored for Dirichlet time-invariant and Neumann boundary conditions. To deal with the highly nonlinear, state-dependent boundary conditions employed in MPD, a modified version of the RB method has been proposed. Apart from the reduction, the RB method for nonlinear systems lacks error bounds and estimates to quantify the accuracy loss due to the reduction. In this regard, a new error estimate has been developed based on a Lure type model formulation for the discretized model, which can be efficiently extended to other systems of such a form. Results show that the error estimate approximates the actual error with a high accuracy. However, the condition to guarantee the existence of the error estimate based on this approach is not always satisfied. To generalize the proposed approach to highly nonlinear systems, a new hierarchical error estimate has been proposed based on the available simulation data in the RB method. Furthermore, a new RB method has been developed to capture discontinuous features of the drilling well geometry. Simulations confirm that this new method captures the physics in MPD even in the presence of discontinuous well geometry.

In contrast to the efficient performance of the RB method for single-phase flow models, due to the high complexity of multi-phase flow models, applying RB method to these models may generate an unstable reduced-order system. To preserve key system properties such as stability after reduction, a pH model formulation has been investigated. This type of systems are proven to be passive and can preserve passivity through structure-preserving reduction techniques. PH model formulations have been proposed for single- and multi-phase flow models used in MPD modeling. These pH models are interconnected in a power-preserving manner, yielding an aggregated pH system model for MPD. This aggregated system can then be discretized and used for model-order reduction, and also controller design, while preserving desired information of the system at the infinite-dimensional level.

چکیده

مدلسازی و کاهش مرتبه به منظور شبیه سازی هیدرولیکی در حفاری مدیریت فشار

نیازهای جامعه برای حفر چاه های عمیق فراوان است و بدون استثنا تأثیر زیادی بر اقتصاد جهانی می گذارد. پایداری برداشت از این منابع در آینده مستلزم بهره برداری از ذخایر غیرقابل دسترسی و غیرمعمول است. حفر چاه های عمیق برای این اهداف دارای پیچیدگی بالا ، عدم اطمینان زیاد ، ریسک و هزینه بالا می باشد. به خصوص، یکی از تهدیدات برای برداشت مقرون به صرفه و سالم از نظر زیست محیطی از این ذخایر انرژی و معدنی، کنترل فشار چاه است. حفظ فشار چاه در محدوده معین ، بین فشار منافذ و فشار شکستگی دیواره، برای ایمنی عملیات حفاری بسیار مهم است. برای کنترل موفقیت آمیز فشار چاه، اخیراً یک تکنیک حفاری به نام حفاری مدیریت فشار گسترش داده شده است. به این منظور، ابزارهایی برای آزمایش سناریوی حفاری مجازی مورد نیاز است، به ویژه در حین عملیات حفاری برای ارزیابی اثرات یک واکنش بالقوه. اهداف اصلی این پایان نامه عبارتند از: الف) تهیه یک مدل هیدرولیک جدید برای حفاری مدیریت فشار به همراه تکنیک های جدید عددی ، ب) توسعه تکنیک های کاهش مرتبه مدل با استفاده از برآورد خطا ، و پ) معرفی فرمولاسیون مدل‌های پورت‌همیلتون برای حفظ خواص کلیدی بعد از کاهش مرتبه.

برای توسعه مجازی یک چاه حفاری، باید یک مدل هیدرولیک برای عملیات حفاری مدیریت فشار تدوین شود. مدل هیدرولیک برای حفاری مدیریت فشار شامل یک مدل جریان تک فاز و چند فاز می‌باشد که از طریق معادلات غیرخطی توصیف کننده تجهیزات حفاری مدیریت فشار به یکدیگر متصل می‌شوند. با این حال، مدل به دست آمده از بسیاری از پیچیدگی های ریاضی رنج می برد، و تکنیک های عددی کلاسیک برای تفسیر و شبیه سازی عددی چنین مدل ناکافی می باشند. بنابراین ، تکنیک های عددی مختلفی برای شبیه سازی عددی توسعه و یا گسترش یافته‌اند. اهداف مورد نظر عبارتند از: الف) دربرگرفتن فیزیک ناشی از هندسه ناپیوسته چاه، ب) جواب حالت پایدار بسیار دقیق تر در حضوراصطکاک و گرانش، و پ) در بر گرفتن شرایط مرزی غیرخطی وابسته به حالت سیستم. مقایسه راه حل های عددی با داده‌های میدانی جمع‌آوری‌شده از چاه‌های حفاری در دنیای واقعی ، اعتبار مدل و همچنین صحت راه‌حل‌های عددی را نشان می‌دهد.

مدل گسسته‌شده برای حفاری مدیریت فشار معمولاً از نظر محاسباتی برای اجرای آنی بسیار سنگین است، و به این دلیل مدل عددی برای طراحی کنترلر مبتنی بر شبیه‌سازی

مناسب نمی‌باشد. علاوه بر این ، هنگام بهینه‌سازی فرآیند طراحی روش حفاری، هرگونه تغییر در پارامترهای سیستم مانند هندسه چاه ، نیاز به انجام بسیاری از شبیه‌سازی‌های سنگین و زمان بر دارد. برای غلبه بر این مسائل ، کاهش مرتبه برای مدل پارامتریزه شده برای حفاری مدیریت فشار سودمند است. در این پایان‌نامه، از روش پایه کاهش‌یافته برای کاهش مرتبه مدل‌های جریان تک‌فاز استفاده شده است. با این وجود، روش پایه کاهش یافته برای شرایط مرزی بدون تغییر زمانی دیریچله و نیومن طراحی شده است. برای مقابله با شرایط مرزی غیرخطی و وابسته به حالت سیستم بکار رفته در حفاری مدیریت فشار، یک نسخه اصلاح شده از روش پایه کاهش یافته ارائه شده است. ورای کاهش مرتبه، روش پایه کاهش یافته برای سیستمهای غیرخطی فاقد تخمین خطا برای حدس میزان خطا ناشی از کاهش مرتبه است. در این راستا، یک تخمین خطای جدید بر اساس مدل اتصال یافته برای مدل گسسته ساخته شده است که می تواند به صورت کارآمد به سایر سیستمهای چنین فرمی گسترش یابد. نتایج نشان می دهد که تخمین خطا با دقت بالا خطای واقعی را تقریب می‌زند. با این حال ، شرط تضمین وجود تخمین خطا بر اساس این رویکرد، همیشه ارضا نمی‌شود. برای تعمیم رویکرد پیشنهادی به سیستمهای بسیار غیرخطی ، تخمین خطای سلسله مراتبی جدیدی بر اساس داده های شبیه سازی موجود در روش پایه کاهش یافته ارائه شده است. علاوه بر این، یک روش جدید پایه کاهش یافته برای دربرگرفتن ویژگی های ناپیوسته هندسه چاه حفاری توسعه داده شده است. شبیه سازی‌ها بیانگر این نکته است که این روش جدید حتی در صورت وجود هندسه ناپیوسته چاه ، فیزیک مشمول در حفاری مدیریت فشار را دربرمیگیرد.

برخلاف عملکرد کارآمد روش پایه کاهش یافته برای مدل‌های جریان تک فاز، به دلیل پیچیدگی بالای مدل‌های چند فاز، استفاده از روش پایه کاهش یافته برای این مدل‌ها ممکن است یک سیستم کاهش یافته ناپایدار ایجاد کند. برای حفظ خواص کلیدی سیستم مانند پایداری پس از کاهش مرتبه، فرمولاسیون مدل پورت‌همیلتونین مورد بررسی قرار گرفته است. این نوع سیستم ها منفعل می‌باشند و می تواند انفعال را از طریق تکنیک‌های کاهش مرتبه حافظ ساختار حفظ کنند. فرمولاسیون مدل پورت-همیلتونین برای مدل‌های جریان تک‌فاز و چند فاز مورد استفاده در مدل سازی حفاری مدیریت فشار ارائه شده است. این مدل های پورت‌همیلتونین به روش حافظ انرژی به هم متصل شده اند و یک مدل سیستم پورت‌همیلتونین جامع برای حفاری مدیریت فشار حاصل می‌شود. این سیستم جامع سپس می تواند گسسته شده و برای کاهش مدل مرتبه و همچنین طراحی کنترلر مورد استفاده قرار گیرد، در حالی که اطلاعات مورد نظر از سیستم در سطح معادلات پاره ای را حفظ می‌کند.

Acknowledgment

Hereby, I want to play a flashback at the wonderful PhD journey of mine and thank everyone who had an impact on my life during the past four years. I cannot remember a moment during these four years without their presence.

First of all, I must thank my promotors, Wil Schilders and Nathan van de Wouw, and my friend/daily supervisor, Laura Iapichino. I vividly remember my on-site interview in Eindhoven for this position with Wil and Nathan, which took place in the Department of Mechanical Engineering. While Nathan was asking hard questions, Wil was typing on his laptop and I was thinking with myself, "oh my god, he is typing whatever I am saying!" Later on in the project, I found out that he was simply replying to his emails! Luckily they offered me this position, over which I spent four wonderful years of my life. Wil always encouraged me to go to conferences and workshops, to expand my network and learn the new trends in the field of MOR. I am specifically thankful for the Durham symposium. His activities also inspired me to co-organize some events, such as the Model Order Reduction Summer School in Eindhoven. He always pushed me to invest in the stock market, which I will do in the near future. I cannot be more thankful for my second promotor, Nathan. As a supervisor and also a leader, I admire his critical thinking and profound knowledge. He was keen to know every detail of the subject at which he was not an expert. His critical questions made me think more deeply into the problems and resulted in more structured research. As a person, he is very thoughtful and considerate. I could not wish for a better person to work with. My daily supervisor and my friend, Laura, discussed with me on any problems I had during the research and also my personal life. With her kind heart and infinite patience, she helped me a lot in making thoughtful decisions. With her receiving invitations to different conferences, I got the chance to open my eyes to the world of MOR. I should also thank other supervisors who were involved in this project, Bart Besselink, Glenn-Ole Kaasa, Florent Di Meglio, and Ulf Jakob Aarsnes. They were always available for discussions, willing to share their knowledge and ideas with me.

Next, I would like to thank my parents and siblings for their dedication and encouragement, not only during these four years but also throughout my life. It

would never be possible for me to stand here without their generous support. With my little brother, Mohammad Hassan, moving to Norway, we have had provided the most support for each other. I must also thank him for the great work that led to the cover of this thesis.

Then, I should thank my dear friends, Sajad and Naveen, who were involved in the HYDRA project. We have had amazing years of research and fun together. Jimmys' nights are unforgettable. I hope you the best for your future life. I would also like to thank Marie Sklodowska-Curie Innovative Training Networks (ITN-EID) that gathered us together and sponsored this project. I would like to thank Harshit as well for his scientific and organizational collaboration.

Kelda Drilling Control AS has a special place in my heart, who hosted me for 1.5 years in an energetic environment. They taught me many subjects which are important in the industry but being neglected in science. Glenn-Ole, Geir Arne, Linn and Marcela helped me a lot in the initiative process of moving to Norway. Daily conversations with Christian and Jon Age helped me finding a good path in my research. Lunch discussions with Christian and Geir Arne taught me many lessons.

I would like to thank CASA for the amazing 2.5 years. I must say many thanks to our secretary, Enna, who is the best in her job. Being committed to her job, she resolved all bureaucratic and administrative problems I had, even when I was in Norway. I would like to thank other secretaries, Jolijn and Diane, who shared the workload with Enna to make CASA a cozy home for everyone. A big *thank you* goes to my former and current office mates, Melania, Nitin, Harshit, Amir, Faeze, Guilia and Perfect. I always enjoyed talking to Rene, Giovanni, Xingang, Rien, Arthur, Koondi, Masoud, Hamid, Ayda and Annastasiia. I should specifically thank my colleagues and SSC fellows, Lotte and Jasper. We always pushed each other to achieve a higher level of fitness. My great thanks go to Barry for his dedicated support with the first phase of my PhD research. CASA will always remain in my heart.

The huge community of Iranians in Eindhoven helped me to truly feel at home. Very special thanks to my first friend in Eindhoven, Hadi. We have shared many unforgettable moments, most of which took place at SSC. Amin joined us later to make a perfect group. Thank you, Amin! Futsal was my main hobby in Eindhoven and helped me make friends who have a special place in my heart. Reza (the captain) and Amir, we made a gang full of sarcastic fun. We are going to continue the same manner throughout our lives. Thanks to you two for these moments! I should thank all members of SHAAKHAA team for these great years, Morteza, Amir Reza, Reza Z., Peyman, Amir Abbas and Mojtaba. I thank all futsal players during the weekends; to name a few, Mohsen, Farzad, Mehdi, Afonso and Celal.

Finally, I would like to thank Prof. Gianluigi Rozza, Prof. Bernhard Maschke, Prof. Barry Koren and Dr. Glenn-Ole Kaasa for having agreed to take part in my doctorate committee and reviewing my thesis.

Curriculum Vitae

Mohammad Hossein Abbasi was born on August 10, 1989, in Eilam, Iran. He received his B.Sc. and M.Sc. in Mechanical Engineering from Sharif University of Technology, Tehran, Iran, in 2011 and 2013, respectively. His bachelor's graduation project focused on theoretical study and design of a human dummy model for crash tests, which received the best thesis award of the department. Later on, he contributed to the construction of the human dummy and carrying out a crash test. His M.Sc. graduation project focused on theoretical and experimental study of slosh phenomena in flying objects, which was carried out at Center of Excellence in Design, Robotics and Automation of Sharif University of Technology. Between 2013 and 2016, he was mainly working in companies and research institutes such as Durali's System Design and Automation, and Niroo Research Institute.

Since October 2016, he started his Ph.D. project within Center for Analysis, Scientific Computing and Applications at the Department of Mathematics and Computer Science at TU Eindhoven, under the supervision of Wil Schilders, Nathan van de Wouw and Laura Iapichino. His research project was part of HYDRA research program funded by European Union's Horizon 2020 Framework Program. HYDRA was focused on different aspect of model order reduction techniques for hydraulics in managed pressure drilling. Mohammad's project was focused on the development of model order techniques suitable for simulation. The results of his research are printed in this dissertation.

Mohammad's research interests include dynamics and control, model order reduction, data-based and dynamical modeling, and computational fluid dynamics.

Bibliography

[1] U. J. F. Aarsnes, B. Akmee, A. Ambrus, and O. M. Aamo, "Robust controller design for automated kick handling in managed pressure drilling," *Journal of Process Control*, volume 47, pages 46–57, November 2016.

[2] U. J. F. Aarsnes, A. Ambrus, F. Di Meglio, A. Karimi Vajargah, O. Morten Aamo, and E. van Oort, "A simplified two-phase flow model using a quasi-equilibrium momentum balance," *International Journal of Multiphase Flow*, volume 83, pages 77–85, Jul. 2016.

[3] U. J. F. Aarsnes, F. Di Meglio, R. Graham, and O. M. Aamo, "A methodology for classifying operating regimes in underbalanced-drilling operations," *SPE Journal*, volume 21, number 02, pages 423–433, April 2016, SPE-178920-PA.

[4] U. J. F. Aarsnes, T. Flåtten, and O. M. Aamo, "Review of two-phase flow models for control and estimation," *Annual Reviews in Control*, volume 42, pages 50–62, 2016.

[5] U. J. F. Aarsnes, *Modeling of two-phase flow for estimation and control of drilling operations*, eng. PhD thesis, Norges Teknisk Naturvitenskapelige Universitet, 2016.

[6] U. J. F. Aarsnes, F. D. Meglio, S. Evje, and O. M.Aamo, "Control-oriented drift-flux modeling of single and two-phase flow for drilling," *American Society of Mechanical Engineers, San Antonio, Texas, USA*, p. V003T37A003, 2014.

[7] M. H. Abbasi, H. Bansal, H. Zwart, L. Iapichino, W. Schilders, and N. van de Wouw, "Power-preserving interconnection of single- and two-phase flow models for managed pressure drilling," *Accepted for American Control Conference (ACC)*, 2020.

[8] M. H. Abbasi, L. Iapichino, B. Besselink, W. Schilders, and N. van de Wouw, "Error estimates for model order reduction of burgers' equation," *IFAC World Congress*, 2020 https://research.tue.nl/en/publications/error-estimates-for-model-order-reduction-of-burgers-equation.

[9] M. H. Abbasi, L. Iapichino, B. Besselink, W. Schilders, and N. van de Wouw, "Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions," *Computer Methods in Applied Mechanics and Engineering*, volume 360, 112 688, Jul. 2020.

[10] M. H. Abbasi, S. Naderi Lordejani, N. Velmurugan, C. Berg, L. Iapichino, W. Schilders, and N. van de Wouw, "A Godunov-type scheme for the drift flux model with variable cross section," *Journal of Petroleum Science and Engineering*, volume 179, pages 796–813, August 2019.

[11] R. Altmann and P. Schulze, "A port-hamiltonian formulation of the navier-stokes equations for reactive flows," *Systems & Control Letters*, volume 100, 51–55, 2017.

[12] A. Ambrus, U. J. F. Aarsnes, A. Karimi Vajargah, B. Akbari, E. van Oort, and O. M. Aamo, "Real-time estimation of reservoir influx rate and pore pressure using a simplified transient two-phase flow model," *Journal of Natural Gas Science and Engineering*, volume 32, pages 439–452, May 2016.

[13] A. Antoulas, *Approximation of large-scale dynamical systems*. Society for Industrial and Applied Mathematics, 2005.

[14] E. Audusse, F. Bouchut, M. Bristeau, R. Klein, and B. Perthame, "A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows," *SIAM Journal on Scientific Computing*, volume 25, number 6, pages 2050–2065, January 2004.

[15] H. Bansal, P. Schulze, M. Abbasi, H. Zwart, L. Iapichino, W. Schilders, and N. van de Wouw, "Port-Hamiltonian formulation of two-phase flow models," Available online at https://www.win.tue.nl/~hbansal/draft-version-to-be-submitted.pdf, 2020.

[16] H. Bansal, H. Zwart, L. Iapichino, W. Schilders, and N. van de Wouw, "Port-Hamiltonian modelling of fluid dynamics models with variable cross-section," *Submitted to 24th International Symposium on Mathematical Theory of Networks and Systems, MTNS*, 2020.

[17] M. Barrault, Y. Maday, N. Nguyen, and A. Patera, "An empirical interpolation method: Application to efficient reduced-basis discretization of partial differential equations," *Comptes Rendus Mathematique*, volume 339, number 9, pages 667–672, November 2004.

[18] M. Baudin, C. Berthon, F. Coquel, R. Masson, and Q. H. Tran, "A relaxation method for two-phase flow models with hydrodynamic closure law," en, *Numerische Mathematik*, volume 99, number 3, pages 411–440, January 2005.

[19] M. N. Belur and C. Praagman, "An efficient algorithm for computing the $\mathcal{H}_\infty$-norm," *IEEE Transactions on Automatic Control*, volume 56, number 7, pages 1656–1660, Jul. 2011.

[20] P. Benner and T. Stykel, *Model or reduction of differential-algebraic equations: a survey.* en, series Surveys in Differential-Algebraic Equations IV. Springer, 2017.

[21] J. P. Berberich, P. Chandrashekar, and C. Klingenberg, "A general well-balanced finite volume scheme for euler equations with gravity," in *Theory, Numerics and Applications of Hyperbolic Problems I*, C. Klingenberg and M. Westdickenberg, Eds., Cham: Springer International Publishing, 2018, pages 151–163.

[22] C. Berg, J. Å. Stakvik, B. Lie, K. Vaagsaether, and G. O. Kaasa, "Pressure wave propagation in Managed Pressure Drilling- model comparison with real life data," in *Proceedings of the 60th Conference on Simulation and Modelling, 13-16 August 2019, Sweden*, 2019.

[23] C. Berg, G. A. Evjen, N. Velmurugan, and M. Culen, "The Influx-Management Envelope Considering Real-Fluid Behavior," *SPE Drilling & Completion*, December 2019, SPE-198916-PA.

[24] C. Berthon and C. Chalons, "A fully well-balanced, positive and entropy-satisfying godunov-type method for the shallow-water equations," *Mathematics of Computation*, volume 85, number 299, pages 1281–1307, 2016.

[25] C. Berthon, C. Chalons, S. Cornet, and G. Sperone, "Fully well-balanced, positive and simple approximate Riemann solver for shallow water equations," en, *Bulletin of the Brazilian Mathematical Society*, volume 47, number 1, pages 117–130, March 2016.

[26] C. Berthon, "Stability of the MUSCL schemes for the Euler equations," *Communications in Mathematical Sciences*, volume 3, number 2, pages 133–157, Jun. 2005.

[27] B. Besselink, N. van de Wouw, and H. Nijmeijer, "Model reduction for nonlinear systems with incremental gain or passivity properties," *Automatica*, volume 49, number 4, pages 861–872, April 2013.

[28] B. Besselink, N. van de Wouw, and H. Nijmeijer, "Model reduction for a class of convergent nonlinear systems," *IEEE Transactions on Automatic Control*, volume 57, number 4, pages 1071–1076, 2012.

[29] B. Besselink, N. van de Wouw, and H. Nijmeijer, "Model reduction for nonlinear systems with incremental gain or passivity properties," *Automatica*, volume 49, pages 861–872, 2013.

[30] S. Bhagwat and A. Ghajar, "A flow pattern independent drift flux model based void fraction correlation for a wide range of gas–liquid two phase flow," *International Journal of Multiphase Flow*, volume 59, number Supplement C, pages 186–205, February 2014.

[31] M. Billaud-Friess and A. Nouy, "Dynamical model reduction method for solving parameter-dependent dynamical systems," *SIAM Journal on Scientific Computing*, volume 39(4), 1766–1792, April 2016.

[32] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk, "Convergence rates for greedy algorithms in reduced basis methods," *SIAM Journal on Mathematical Analysis*, volume 43, number 3, pages 1457–1472, 2011.

[33] J. Borggaard, Z. Wangb, and L. Zietsman, "A goal-oriented reduced-order modeling approach for nonlinear systems," *Computers and Mathematics with Applications*, volume 71, pages 2155–2169, 2016.

[34] N. A. Bruinsma and M. Steinbuch, "A fast algorithm to compute the $\mathcal{H}_\infty$-norm of a transfer function matrix," *Systems & Control Letters*, volume 14, number 4, pages 287–293, April 1990.

[35] C Calgaro, E Creus, T Goudon, and Y Penel, "Positivity-preserving schemes for euler equations: Sharp and practical CFL conditions," en, *Journal of Computational Physics*, volume 234, pages 27–35, February 2013.

[36] F. Cardoso-Ribiero, D. Matignon, and V. Pommier-Buddinger, "Modeling of a fluid-structure coupled system using port-Hamiltonian formulation," *IFAC-PapersOnLine*, volume 48, number 13, 217–222, 2015.

[37] M. A. D. de Carvalho, G. F. de Moraes Oliveira, L. D. Fernandes, A. L. Martins, and M. P. Vega, "Two-phase flow model validation during conventional / Pressurized Mud Cap Drilling (PMCD) scenarios," *Journal of Petroleum Science and Engineering*, volume 172, pages 314–326, January 2019.

[38] M. J. Castro, T. Morales de Luna, and C. Parés, "Well-balanced schemes and path-conservative numerical methods," in *Handbook of Numerical Analysis*, series Handbook of numerical methods for hyperbolic problems, R. Abgrall and C.-W. Shu, Eds., volume 18, Elsevier, January 2017, pages 131–175.

[39] M. Castro Diaz, J. Lopez-Garcia, and C. Pares, "High order exactly well-balanced numerical methods for shallow water systems," en, *Journal of Computational Physics*, volume 246, 242–264, 2013.

[40] E. Cayeux, B. Daireaux, E. Dvergsnes, A. Leulseged, B. Bruun, and M. Herbert, "Advanced drilling simulation environment for testing new drilling automation techniques and practices," *SPE Drilling & Completion*, volume 27, number 04, pages 559–573, December 2012, SPE-150941-PA.

[41] Y. Cengel and M. A. Boles, *Thermodynamics: An Engineering Approach*. McGraw-Hill Education, New York City, NY, USA, 2006.

[42] J. Cervera, A. van der Schaft, and A. Banos, "Interconnection of port-Hamiltonian systems and composition of dirac structures," *Automatica*, volume 43, number 2, pages 212–225, 2007.

[43] S. Chaturantabut, C. Beattie, and S. Gugercin, "Structure-preserving model reduction for nonlinear port-hamiltonian systems," *SIAM Journal on Scientific Computing*, volume 38, number 5, B837–B865, 2016.

[44] S. Chaturantabut and D. Sorensen, "Nonlinear model reduction via discrete empirical interpolation," *SIAM Journal on Scientific Computing*, volume 32, number 5, pages 2737–2764, January 2010.

[45] A. Chertock, S. Cui, A. Kurganov, and T. Wu, "Well-balanced positivity preserving central-upwind scheme for the shallow water system with friction terms," en, *International Journal for Numerical Methods in Fluids*, volume 78, number 6, pages 355–383, Jun. 2015.

[46] A. Chertock, S. Cui, A. Kurganov, S. N. Ozcan, and E. Tadmor, "Well-balanced schemes for the Euler equations with gravitation: Conservative formulation using global fluxes," *Journal of Computational Physics*, volume 358, pages 36–52, April 2018.

[47] J. E. Chirinos, J. R. Smith, and D. A. Bourgoyne, "A simplified method to estimate peak casing pressure during mpd well control," *SPE Annual Technical Conference Exhibition*, 2011.

[48] J. Choi, E. Pereyra, C. Sarica, H. Lee, I. Jang, and J. Kang, "Development of a fast transient simulator for gasliquid two-phase flow in pipes," en, *Journal of Petroleum Science and Engineering*, volume 102, pages 27–35, February 2013.

[49] S. Clain and D. Rochette, "First- and second-order finite volume methods for the one-dimensional nonconservative Euler system," *Journal of Computational Physics*, volume 228, number 22, pages 8214–8248, December 2009.

[50] P. Colella, "Multidimensional upwind methods for hyperbolic conservation laws," *Journal of Computational Physics*, volume 87, number 1, pages 171–200, March 1990.

[51] C. Cook, S. Balachandar, J. Chung, and L. Vu-Quoc, "A generalized Characteristic-Based Split projection method for Navier-Stokes with real fluids," *International Journal of Heat and Mass Transfer*, volume 124, pages 1045 –1058, 2018.

[52] A. Cosimo, A. Cardona, and S. Idelsohn, "General treatment of essential boundary conditions in reduced order models for non-linear problems," en, *Advanced Modeling and Simulation in Engineering Sciences*, volume 3, number 1, 3:7, December 2016.

[53] A. Cosimo, A. Cardona, and S. Idelsohn, "Improving the k-compressibility of hyper reduced order models with moving sources: Applications to welding and phase change problems," *Computer Methods in Applied Mechanics and Engineering*, volume 274, pages 237–263, Jun. 2014.

[54] R. Courant, K. Friedrichs, and H. Lewy, "Uber die partiellen differenzengleichungen der mathematischen physik," *Mathematische Annalen (in German)*, volume 100, number 1, 32–74, 1928.

[55] R. Courant, K. Friedrichs, and H. Lewy, "On the partial difference equations of mathematical physics," *IBM Journal of Research and Development*, volume 11, number 2, pages 215–234, 1967.

[56] F. Crouzet, F. Daude, P. Galon, J.-M. Hrard, O. Hurisse, and Y. Liu, "Validation of a two-fluid model on unsteady liquidvapor water flows," *Computers and Fluids*, volume 119, pages 131–142, Sep. 2015.

[57] D. H. Cuong and M. D. Thanh, "A Godunov-type scheme for the isentropic model of a fluid flow in a nozzle with variable cross-section," *Applied Mathematics and Computation*, volume 256, pages 602–629, April 2015.

[58] C. G. da Silva Santim and E. S. Rosa, "Roe-type Riemann solver for gas–liquid flows using drift-flux model with an approximate form of the jacobian matrix," en, *International Journal for Numerical Methods in Fluids*, volume 80, number 9, pages 536–568, March 2016.

[59] C. Daversin and C. Prud'homme, "Simultaneous empirical interpolation and reduced basis method for non-linear problems," *Comptes Rendus Mathematique*, volume 353, number 12, pages 1105–1109, December 2015.

[60] F. Di Meglio, G. O. Kaasa, and N. Petit, "A first principle model for multiphase slugging flow in vertical risers," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, December 2009, pages 8244–8251.

[61] F. Di Meglio, G. O. Kaasa, N. Petit, and V. Alstad, "Slugging in multiphase flow as a mixed initial-boundary value problem for a quasilinear hyperbolic system," in *Proceedings of the 2011 American Control Conference*, 2011, pages 3589–3596.

[62] F. Di Meglio, "Dynamics and control of slugging in oil production," Theses, École Nationale Supérieure des Mines de Paris, Jul. 2011.

[63] M. Drohmann, B. Haasdonk, and M. Ohlberger, "Reduced Basis Approximation for Nonlinear Parametrized Evolution Equations based on Empirical Operator Interpolation," *SIAM Journal on Scientific Computing*, volume 34, number 2, A937–A969, January 2012.

[64] V. Duindam, A. Macchelli, S. Stramigioli, and H. Bruyninckx, *Modeling and control of complex physical systems: the port-Hamiltonian approach*. Springer, 2009.

[65] J. Edmonds, "Matroids and the greedy algorithm," *Mathematical Programming*, volume 1, number 1, pages 127–136, December 1971.

[66] H. Egger, T. Kugler, B. Liljegren-Sailer, N. Marheineke, and V. Mehrmann, "On structure-preserving model reduction for damped wave propagation in transport networks," *SIAM Journal on Scientific Computing*, volume 40(1), 331–365, April 2017.

[67] S. Evje and K. Fjelde, "On a rough AUSM scheme for a one-dimensional two-phase model," *Computers and Fluids*, volume 32, number 10, pages 1497–1530, December 2003.

[68] S. Evje and T. Flatten, "On the wave structure of two-phase flow models," *SIAM Journal of Applied Mathematics*, volume 67, number 2, pages 487–511, 2007.

[69] S. Evje and K. K. Fjelde, "Hybrid flux-splitting schemes for a two-phase flow model," en, *Journal of Computational Physics*, volume 175, number 2, pages 674–701, January 2002.

[70] A. Figueiredo, R. Baptista, F. de Freitas Rachid, and G. Bodstein, "Numerical simulation of stratified-pattern two-phase flow in gas pipelines using a two-fluid model," *International Journal of Multiphase Flow*, volume 88, pages 30–49, 2017.

[71] K. K. Fjelde and K. H. Karlsen, "High-resolution hybrid primitive–conservative upwind schemes for the drift flux model," *Computers and Fluids*, volume 31, number 3, pages 335–367, March 2002.

[72] T. Flåtten and S. T. Munkejord, "The approximate riemann solver of Roe applied to a drift-flux two-phase flow model," en, *ESAIM: Mathematical Modelling and Numerical Analysis*, volume 40, number 4, pages 735–764, Jul. 2006.

[73] K. O. Friedrichs, "Symmetric hyperbolic linear differential equations," en, *Communications on Pure and Applied Mathematics*, volume 7, number 2, pages 345–392, May 1954.

[74] S. L. Gavrilyuk and J. Fabre, "Lagrangian coordinates for a drift-flux model of a gas-liquid mixture," *International Journal of Multiphase Flow*, volume 22, number 3, pages 453–460, Jun. 1996.

[75] J.-M. Godhavn, "Control requirements for automatic managed pressure drilling system," *SPE Drilling & Completion*, volume 25, number 03, pages 336–345, Sep. 2010, SPE-119442-PA.

[76] J.-M. Godhavn, A. Pavlov, G.-O. Kaasa, and N. L. Rolland, "Drilling seeking automatic control solutions," *IFAC Proceedings Volumes*, volume 44, number 1, pages 10842 –10 850, 2011, 18th IFAC World Congress.

[77] Y. L. Gorrec, H. Zwart, and B. Maschke, "Dirac structures and boundary control systems associated with skew-symmetric differential operators," *SIAM Journal on Control and Optimization*, volume 44, number 5, 1864–1892, 2005.

[78] M. Grepl, "Model order reduction of parametrized nonlinear reaction–diffusion systems," *Computers and Chemical Engineering*, volume 43, pages 33–44, August 2012.

[79] M. A. Grepl, "Certified reduced basis methods for nonaffine linear time-varying and nonlinear parabolic partial differential equations," *Mathematical Models and Methods in Applied Sciences*, volume 22, number 03, page 1 150 015, March 2012.

[80] S. Gugercin and A. C. Antoulas, "A survey of model reduction by balanced truncation and some new results," *International Journal of Control*, volume 77, number 8, pages 748–766, May 2004.

[81] S. Gugercin, A. C. Antoulas, and C. Beattie, "$\mathcal{H}_2$ model reduction for large-scale linear dynamical systems," *SIAM Journal on Matrix Analysis and Applications*, volume 30, number 2, pages 609–638, 2008.

[82] M. D. Gunzburger, J. S. Peterson, and J. N. Shadid, "Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data," *Computer Methods in Applied Mechanics and Engineering*, volume 196, number 4, pages 1030–1047, January 2007.

[83] Y. Guo, T. Xiong, and Y. Shi, "A positivity-preserving high order finite volume compact-WENO scheme for compressible Euler equations," *Journal of Computational Physics*, volume 274, pages 505–523, October 2014.

[84] B. Haasdonk and M. Ohlberger, "Reduced basis method for finite volume approximations of parametrized linear evolution equations," en, *ESAIM Mathematical Modelling and Numerical Analysis*, volume 42, number 2, pages 277–302, March 2008.

[85] S. Hain, M. Ohlberger, M. Radic, and K. Urban, "A hierarchical a posteriori error estimator for the reduced basis method," *Advances in Computational Mathematics*, pages 1–24, January 2019.

[86] N. Hammouda and Y. F. Rao, "The effect of the advanced drift-flux model of ASSERT-PV on critical heat flux, flow and void distributions in CANDU bundle subchannels," *Annals of Nuclear Energy*, volume 104, number Supplement C, pages 197–213, Jun. 2017.

[87] A. Harten, P. Lax, and B. van Leer, "On upstream differencing and Godunov-type schemes for hyperbolic conservation laws," *SIAM Review*, volume 25, number 1, pages 35–61, January 1983.

[88] P. Helluy, J.-M. Hérard, and H. Mathis, "A well-balanced approximate Riemann solver for compressible flows in variable cross-section ducts," *Journal of Computational and Applied Mathematics*, volume 236, number 7, pages 1976–1992, January 2012.

[89] J. S. Hesthaven, G. Rozza, and B. Stamm, *Certified reduced basis methods for parametrized partial differential equations*, en, series SpringerBriefs in Mathematics. Springer International Publishing, 2016.

[90] D. Holm and B. Kupershmidt, "Hydrodynamics and electrohydrodynamics of adiabatic multiphase fluids and plasmas," *International Journal of Multiphase Flow*, volume 12, number 4, 667–680, 1986.

[91] J. A. Holmes, "Description of the Drift Flux Model in the Loca Code Relap-UK," in *Proceedings of the Conference on Heat and Fluid Flow in Water Reactor Safety*, Manchester, UK: Institution of Mechanical Engineers paper C206/77, Sep. 1977, pages 103–108.

[92] L. Iapichino, A. Quarteroni, and G. Rozza, "Reduced basis method and domain decomposition for elliptic problems in networks and complex parametrized geometries," *Computers and Mathematics with Applications*, volume 71, number 1, pages 408–430, January 2016.

[93] IRIS, *Test Wells - IRIS*, http://www.iris.no/research/ullrigg/ullrigg-drilling-rig/test-wells, 2017, accessed 28 January 2019.

[94] K Isao and I Mamoru, "Drift flux model for large diameter pipe and new correlation for pool void fraction," en, *International Journal of Heat and Mass Transfer*, volume 30, number 9, pages 1927–1939, Sep. 1987.

[95] M. Ishii and T. Hibiki, *Thermo-Fluid Dynamics of Two-Phase Flow*. Springer, New York, NY, USA, 2006.

[96] M. Ishii and T. Hibiki, "One-Dimensional Drift-Flux Model," en, in *Thermo-Fluid Dynamics of Two-Phase Flow*, M. Ishii and T. Hibiki, Eds., New York, NY: Springer New York, 2011, pages 397–436.

[97] B. Jacob and H. Zwart, *Linear port-Hamiltonian systems on infinite dimensional spaces*. Springer, New York, 2012.

[98] A. Janon, M. Nodet, and C. Prieur, "Certified reduced-basis solutions of viscous Burgers equation parametrized by initial and boundary values," *ESIAM: Mathematical Modelling and Numerical Analysis*, volume 47, number 2, pages 317–348, 2013.

[99] G.-O. Kaasa, Ø. N. Stamnes, O. M. Aamo, and L. S. Imsland, "Simplified hydraulics model used for intelligent estimation of downhole pressure for a managed-pressure-drilling control system," en, *SPE Drilling & Completion*, volume 27, number 01, pages 127–138, March 2012.

[100] R. Käppeli and S. Mishra, "Well-balanced schemes for the Euler equations with gravitation," *Journal of Computational Physics*, volume 259, pages 199–219, February 2014.

[101] H. K. Khalil, *Nonlinear systems*, English, 3 edition. Upper Saddle River, N.J: Pearson, December 2001.

[102] Z Kowalczuk and M Tatara, "Analytical steady-state model of the pipeline flow process," en, in *2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*, Miedzyzdroje, Poland: IEEE, August 2018, pages 182–187.

[103] D. Kröner and M. Thanh, "Numerical solutions to compressible flows in a nozzle with variable cross-section," *SIAM Journal on Numerical Analysis*, volume 43, number 2, pages 796–824, January 2005.

[104] M. Krstic and A. Smyshlyaev, *Boundary control of PDEs: a course on backstepping designs*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2008.

[105] A. Kurganov and E. Tadmor, "New high-resolution central schemes for nonlinear conservation laws and convection–diffusion equations," *Journal of Computational Physics*, volume 160, number 1, pages 241–282, May 2000.

[106] M. Kurula, A. van der Schaft, and H. Zwart, "Composition of infinite dimensional dirac structures," *17th International Symposium on Mathematical Theory of Networks and Systems, MTNS*, 27–32, 2006.

[107] A. C. V. M. Lage, K. K. Fjelde, and R. W. Time, "Underbalanced Drilling Dynamics: Two-Phase Flow Modeling and Experiments," *SPE Journal*, volume 8, number 01, pages 61–70, March 2003, SPE-83607-PA.

[108] I. S. Landet, H. Mahdianfar, U. J. F. Aarsnes, A. Pavlov, and O. M. Aamo, "Modeling for MPD operations with experimental validation," in *IADC/SPE Drilling Conference and Exhibition*, Society of Petroleum Engineers, California, SPE-150461-MS, 2012.

[109] I. S. Landet, A. Pavlov, and O. M. Aamo, "Modeling and control of heave-induced pressure fluctuations in managed pressure drilling," *IEEE Transactions on Control Systems Technology*, volume 21, number 4, pages 1340–1351, Jul. 2013.

[110] P. D. Lax, "Weak solutions of nonlinear hyperbolic equations and their numerical computation," en, *Communications on Pure and Applied Mathematics*, volume 7, number 1, pages 159–193, 1954.

[111] B. J. Lee, E. F. Toro, C. E. Castro, and N. Nikiforakis, "Adaptive Osher-type scheme for the Euler equations with highly nonlinear equations of state," *Journal of Computational Physics*, volume 246, pages 165–183, August 2013.

[112] H. Lee and J. Lee, "Design of Iterative Sliding Mode Observer for Sensorless PMSM Control," *IEEE Trans. Contr. Syst. Technol.*, volume 21, number 4, pages 1394–1399, 2013.

[113] R. Leveque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge university press, 2002.

[114] M.-S. Liou and C. J. Steffen, "A New Flux Splitting Scheme," *Journal of Computational Physics*, volume 107, number 1, pages 23–39, Jul. 1993, bibtex: AUSM_original_euler.

[115] M. Liou and C. Steffen, "A New Flux Splitting Scheme," *Journal of Computational Physics*, volume 107, number 1, pages 23–39, Jul. 1993.

[116] S Livescu, "Mathematical modeling of thixotropic drilling mud and crude oil flow in wells and pipelinesa review," *Journal of Petroleum Science and Engineering*, volume 98, pages 174–184, 2012.

[117] S. Lorenzi, A. Cammi, L. Luzzi, and G. Rozza, "POD-Galerkin method for finite volume approximation of Navier–Stokes and RANS equations," *Computer Methods in Applied Mechanics and Engineering*, volume 311, pages 151–179, November 2016.

[118] A. Macchelli, Y. L. Gorrec, and H. Ramirez, "Boundary energy-shaping control of an ideal compressible isentropic fluid in 1-d," *IFAC-PapersOnLine*, volume 50, number 1, pages 5598–5603, 2017.

[119] A. Macchelli and C. Melchiorri, "Modeling and control of the timoshenko beam. the distributed port Hamiltonian approach," *SIAM Journal on Control and Optimization*, volume 43, number 2, pages 743–767, 2004.

[120] R. Malekzadeh, S. P. C. Belfroid, and R. F. Mudde, "Transient drift flux modelling of severe slugging in pipeline-riser systems," *International Journal of Multiphase Flow*, volume 46, number Supplement C, pages 32–37, November 2012.

[121] V. Martins, B. Maschke, and Y. L. Gorrec, "Hamiltonian approach to the stabilization of systems of two conservation laws," *IFAC Proceedings Volumes*, volume 43, number 14, 581–586, 2010.

[122] B. Maschke and A. van der Schaft, "Port-controlled hamiltonian systems: Modelling origins and system-theoretic properties," *IFAC Proceedings Volumes*, volume 25, 359 – 365, 1992.

[123] C. Meyer, R. Arnd, and T. Fredi, "Optimal control of pdes with regularized pointwise state constraints," *Computational Optimization and Applications*, volume 33, number 2-3, pages 209–228, 2006.

[124] V. Michel-Dansac, C. Berthon, S. Clain, and F. Foucher, "A well-balanced scheme for the shallow-water equations with topography or manning friction," *Journal of Computational Physics*, volume 335, pages 115–154, April 2017.

[125] V. Michel-Dansac, C. Berthon, S. Clain, and F. Foucher, "A well-balanced scheme for the shallow-water equations with topography," *Computers and Mathematics with Applications*, volume 72, number 3, pages 568–593, August 2016.

[126] A Morin and G. Reigstad, "Pipe networks: Coupling constants in a junction for the isentropic euler equations," in *3rd Trondheim Gas Technology Conference*, Trondheim, Norway: Energy Procedia, 2015, pages 140–149.

[127] S. T. Munkejord, "Analysis of the two-fluid model and the drift-flux model for numerical calculation of two-phase flow," eng, PhD thesis, Norges Teknisk-Naturvitenskapelige Universitet, Fakultet for Ingeniørvitenskap og Teknologi, 2006.

[128] S. Naderi Lordejani, M. H. Abbasi, N. Velmurgan, C. Berg, J. Å. Stakvik, B. Besselink, L. Iapichino, F. Di Meglio, W. Schilders, and N. van de Wouw, "Modeling and numerical implementation of managed pressure drilling systems for evaluating pressure control systems," *SPE Drilling & Completion*, 2020.

[129] S. Naderi Lordejani, B. Besselink, M. H. Abbasi, G. O. Kaasa, W. H. A. Schilders, and N. van de Wouw, "Model order reduction for managed pressure drilling systems based on a model with local nonlinearities," *IFAC-PapersOnLine*, 3rd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production OOGP 2018, volume 51, number 8, pages 50–55, January 2018.

[130] O. Nelles, *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*, en. Berlin Heidelberg: Springer-Verlag, 2001.

[131] A. Nikoofard, U. J. F. Aarsnes, T. A. Johansen, and G. O. Kaasa, "State and parameter estimation of a drift-flux model for underbalanced drilling operations," *IEEE Transactions on Control Systems Technology*, volume 25, number 6, pages 2000–2009, 2017.

[132] A. Nikoofard, U. J. F. Aarsnes, T. A. Johansen, and G. O. Kaasa, "Estimation of states and parameters of a drift-flux model with unscented kalman filter," *IFAC Papers Online*, 2nd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production OOGP 2015, volume 48, number 6, pages 165–170, January 2015.

[133] S. Noelle, Y. Xing, and C.-W. Shu, "High-order well-balanced finite volume WENO schemes for shallow water equation with moving water," *Journal of Computational Physics*, volume 226, number 1, pages 29–58, Sep. 2007.

[134] G. Nygaard and G. Nævdal, "Nonlinear model predictive control scheme for stabilizing annulus pressure during oil well drilling," *Journal of Process Control*, volume 16, number 7, pages 719–732, August 2006.

[135] M. Ohlberger and S. Rave, "Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing," *Comptes Rendus Mathematique*, volume 351, pages 901–906, April 2013.

[136] P. Orlandi, "The Burgers equation," en, in *Fluid Flow Phenomena: A Numerical Toolkit*, series Fluid Mechanics and Its Applications, Dordrecht: Springer Netherlands, 2000, pages 40–50.

[137] R. Ortega, A. van der Schaft, B. Maschke, and G. Escobar, "Interconnection and damping assignment passivity-based control of port-controlled hamilonian systems," *Automatica*, volume 38, pages 585–596, 2002.

[138] R. Pasumarthy and A. van der Schaft, "A finite dimensional approximation of the shallow water equations: The port-hamiltonian approach," *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA*, 3984–3989, 2006.

[139] S. Patankar, *Numerical heat transfer and fluid flow*, en. Taylor & Francis, 1980.

[140] T. Pedersen, U. J. F. Aarsnes, and J.-M. Godhavn, "Flow and pressure control of underbalanced drilling operations using nmpc," *Journal of Process Control*, volume 68, pages 73 –85, 2018.

[141] T. Pedersen, J.-M. Godhavn, and J. Schubert, "Supervisory control for underbalanced drilling operations," *IFAC-PapersOnLine*, volume 48, number 6, pages 120 –127, 2015, 2nd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production OOGP 2015.

[142] R. PompIlio, "Experimental analysis of air water two phase flow in vertical large pipes and development of drift flux models," eng, Laurea Magistrale / Specialistica, Politecnico Di Milano, Italy, April 2014.

[143] A. Poullikkas, "Effects of two-phase liquid-gas flow on the performance of the nuclear reactor cooling pumps," *Progress in Nuclear Energy*, volume 42, number 1, pages 3–10, 2003.

[144] M. Prebeg, T. Flåtten, and B. Müller, "Boundary and source term treatment in the large time step method for a common two-fluid model," in *Proceedings of the 11th International Conference on CFD in the Minerals and Process Industries, Melbourne, Australia (2015)*, 2015.

[145] M. Prebeg, T. Flåtten, and B. Müller, "Large time step roe scheme for a common 1D two-fluid model," *Applied Mathematical Modelling*, volume 44, pages 124–142, April 2017.

[146] A. Quarteroni, A. Manzoni, and F. Negri, *Reduced Basis Methods for Partial Differential Equations*, en, series La Matematica per il 3+2. Springer, 2016.

[147] T. D. Reed and A. A. Pilehvari, "A new model for laminar, transitional, and turbulent flow of drilling muds," in *SPE Production Operations Symposium*, Society of Petroleum Engineers. SPE-25456-MS, 1993.

[148] J. Reiss, P. Schulze, J. Sesterhenn, and V. Mehrmann, "The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena," *SIAM Journal on Scientific Computing*, volume 40(3), 1322–1344, December 2018.

[149] D. Rim, S. Moe, and R. LeVeque, "Transport reversal for model reduction of hyperbolic partial differential equations," *SIAM / ASA Journal on Uncertainty Quantification*, volume 6(1), pages 118–150, January 2018.

[150] R. Rommetveit and E. H. Vefring, "Comparison of results from an advanced gas kick simulator with surface and downhole data from full scale gas kick experiments in an inclined well," SPE-22558-MS, Society of Petroleum Engineers, January 1991.

[151] R. Rommetveit and E. Vefring, "Comparison of results from an advanced gas kick simulator With surface and downhole data from full scale gas kick experiments in an inclined well," Society of Petroleum Engineers, January 1991.

[152] V. Rusanov, *Calculation of Interaction of Non-steady Shock Waves with Obstacles*. National Research Council of Canada, 1962.

[153] A. van der Schaft and D. Jeltsema, "Port-Hamiltonian systems theory: An introductory overview," *Foundations and Trends in Systems and Control*, volume 1, number 2, pages 173–378, 2014.

[154] A. van der Schaft and B. Maschke, "Hamiltonian formulation of distributed-parameter systems with boundary energy flow," *Journal of Geometry and Physics*, volume 42, 166–194, 2002.

[155] Schlumberger. (2019). Schlumberger oilfield glossary. https://www.glossary.oilfield.slb.com/, [Online]. Available: https://www.glossary.oilfield.slb.com/ (visited on 09/30/2019).

[156] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, "Solving boundary value problems for ordinary differential equations in matlab with bvp4c," en, October 2000.

[157] Y. Shekari, E. Hajidavalloo, and M. Behbahani-Nejad, "Reduced order modeling of transient two-phase flows and its application to upward two-phase flows in the under-balanced drilling," *Applied Mathematics and Computation*, volume 224, pages 775–790, November 2013.

[158] S. Simon and J. C. Mandal, "A simple cure for numerical shock instability in the HLLC Riemann solver," *Journal of Computational Physics*, volume 378, pages 477–496, February 2019.

[159] A. Singer, D. GIillespie, J. Norbury, and R. S. Eisenberg, "Singular perturbation analysis of the steady-state Poisson–Nernst–Planck system: Applications to ion channels," *European Journal of Applied Mathematics*, volume 19, number 5, pages 541–569, 2008.

[160] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*, en, 2nd edition. Wiley, 2005.

[161] H. L. Smith and P. Waltman, "Perturbation of a globally stable steady state," *Proceedings of the American Mathematical Society*, volume 127, number 2, pages 447–453, 1999.

[162] Ø. N. Stamnes, J. Zhou, G. O. Kaasa, and O. M. Aamo, "Adaptive observer design for the bottomhole pressure of a managed pressure drilling system," in *2008 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008, pages 2961–2966.

[163] T. Strecker, O. M. Aamo, and H. Manum, "Simulation of heave-induced pressure oscillations in herschel-bulkley muds," *SPE Journal*, volume 22, number 05, pages 1635–1653, October 2017, SPE-185947-PA.

[164] P. Swanee and A. K. Jain, "Explicit equations for pipeflow problems," *Journal of the hydraulics division*, volume 102, number 5, 1976.

[165] P. K. Sweby, "High resolution schemes using flux limiters for hyperbolic conservation laws," *SIAM Journal on Numerical Analysis*, volume 21, number 5, pages 995–1011, 1984.

[166] E. Tassi, "Hamiltonian closures in fluid models for plasmas," *The European Physical Journal D*, volume 71, page 269, 2017.

[167] M. D. Thanh, "On a two-fluid model of two-phase compressible flows and its numerical approximation," *Communications in Nonlinear Science and Numerical Simulation*, volume 17, number 1, pages 195–211, January 2012.

[168] I. Tiselj and S. Petelin, "Second order numerical method for two-fluid model of air-water flow," *Nuclear Society of Slovenia*, 1995.

[169] J. E. Udegbunam, K. K. Fjelde, S. Evje, and G. Nygaard, "On the advection-upstream-splitting-method hybrid scheme: A simple transient-flow model for managed-pressure-drilling and underbalanced-drilling applications," *SPE Drilling and Completion*, volume 30, number 02, pages 98–109, Jun. 2015.

[170] J. E. Udegbunam, K. K. Fjelde, S. Evje, and G. Nygaard, "On the advection-upstream-splitting-method hybrid scheme: a simple transient-flow model for managed-pressure-drilling and underbalanced-drilling applications," *SPE Drilling & Completion*, volume 30, number 02, pages 98–109, Jun. 2015, SPE-168960-PA.

[171] B. Van Leer, "Towards the ultimate conservative difference scheme III. upstream-centered finite-difference schemes for ideal compressible flow," *Journal of Computational Physics*, volume 23, number 3, pages 263–275, March 1977.

[172] P. A. Varadarajan and P. S. Hammond, "Numerical scheme for accurately capturing gas migration described by 1D multiphase drift flux model," *International Journal of Multiphase Flow*, volume 73, number Supplement C, pages 57–70, Jul. 2015.

[173] N. Velmurugan, "Deliverable D5.2:definition case study on standard (non-complex) influx/loss scenarios," 2017, HYDRA project.

[174] N. Velmurugan and F. Di Meglio, "Transient modelling of influx and observer implementation for estimation while drilling," in *IEEE Conference on Decision and Control (CDC)*, 2018.

[175] J. Villegas, *Port-Hamiltonian approach to distributed parameter systems.* PhD thesis, 2007.

[176] N. Vishnumolakala, E. Gildin, and S. Noynaert, "A simulation environment for automatic managed pressure drilling control," *IFAC-PapersOnLine*, volume 48, number 6, pages 128 –133, 2015, 2nd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production OOGP 2015.

[177] N. M. T. VU, L. Lefevre, and B. Maschke, "Port-Hamiltonian formulation for systems of conservation laws: Application to plasma dynamics in tokamak reactors," *IFAC Proceedings Volumes*, volume 45, 108–113, 2012.

[178] G. B. Wallis, *One Dimensional Two-Phase Flow*, eng. Mcgraw-Hill, 1656.

[179] N. Wei, C. Xu, Y. Meng, G. Li, X. Ma, and A. Liu, "Numerical simulation of gas-liquid two-phase flow in wellbore based on drift flux model," *Applied Mathematics and Computation*, volume 338, pages 175 –191, 2018.

[180] G. Welper, "Interpolation of Functions with Parameter Dependent Jumps by Transformed Snapshots," *SIAM Journal on Scientific Computing*, volume 39, number 4, A1225–A1250, January 2017.

[181] H. de Wilde, *Port-Hamiltonian discretization of gas pipeline networks*. MSc. Thesis Applied Mathematics, Groningen University, 2015.

[182] Y. Zhang, L. Feng, S. Li, and P. Benner, "An efficient output error estimation for model order reduction of parametrized evolution equations," *SIAM Journal on Scientific Computing*, volume 37, number 6, B910–B936, January 2015.

[183] V. D. Zhibaedov, N. A. Lebedeva, A. A. Osiptsov, and K. F. Sin'kov, "On the hyperbolicity of one-dimensional models for transient two-phase flow in a pipeline," en, *Fluid Dynamics*, volume 51, number 1, pages 56–69, January 2016.

[184] W. Zhou, B. Hamroun, Y. L. Gorrec, and F. Couenne, "Infinite dimensional port Hamiltonian representation of reaction diffusion processes," *IFAC-PapersOnLine*, volume 48, 476–481, 2015.

[185] N. Zuber and J. Findlay, "Average volumetric concentration in two-phase flow systems," *Journal of Heat Transfer*, volume 87, number 4, pages 453–468, November 1965.