Modelling of coupled reservoir and wellbore for energy transition

Mansour Pour, K.

DOI
10.4233/uuid:d77e2228-fa44-4052-8c88-c5f394ba6d0c

Publication date
2023

Document Version
Final published version

Citation (APA)
Mansour Pour, K. (2023). *Modelling of coupled reservoir and wellbore for energy transition*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:d77e2228-fa44-4052-8c88-c5f394ba6d0c

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

MODELLING OF COUPLED RESERVOIR AND WELLBORE FOR ENERGY TRANSITION

MODELLING OF COUPLED RESERVOIR AND WELLBORE FOR ENERGY TRANSITION

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag, 31 oktober 2023 om 10.00 uur

door

Kiarash MANSOUR POUR

Master of Engineering in Energy Engineering,
Politecnico di Milano, Milan, Italy,
geboren te Tehran, Iran.

Dit proefschrift is goedgekeurd door de

promotor: Dr. D.V. Voskov
promotor: Prof. dr. D.F. Bruhn

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Dr. D.V. Voskov,	Technische Universiteit Delft
Prof. dr. D.F. Bruhn,	Technische Universiteit Delft
Onafhankelijke leden:	
Prof. dr. P.L.J. Zitha	Technische Universiteit Delft
Prof. dr. P.J. Vardon	Technische Universiteit Delft
Prof. dr. V. Demyanov,	Heriot-Watt university, Edinburgh
Dr. G. Rongier,	Technische Universiteit Delft
Dr. O. Møyner,	SINTEF, Norway

Keywords: Multi-segmented well, Trust-region nonlinear solver, PINNs, Machine learning, Geothermal simulation, carbon capture utilization, and storage, numerical simulation, Physics-informed neural networks

Printed by: ProefschriftMaken

Front & Back: Kiarash Mansour Pour, Kiarash Mansour Pour

ISBN 978-94-6366-761-6

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

This too shall pass.

Attar

When you do things from your soul, you feel a river moving in you, a joy.

Rumi

CONTENTS

SUMMARY

Borehole operations play a crucial role in managing various subsurface activities related to energy, including energy storage, geothermal energy production, CO_2 sequestration, oil and gas extraction, wastewater disposal, and thermal recovery processes. In recent times, intelligent well technologies, such as long deviated multi-lateral wells equipped with advanced inflow control valves, have been employed to enhance field profitability and operational reliability. Additionally, different wellbore designs are utilized to efficiently extract heat from the subsurface in advanced geothermal approaches. To ensure accurate modeling, prediction, and optimization of all these energy production processes, it is essential to employ fully-coupled models that accurately capture thermal multi-phase compositional flow and transport in both the reservoir and the boreholes.

In Chapter 2, we present a detailed explanation of our numerical framework and governing equations for both the reservoir and the wellbore. We introduce a novel molar formulation approach called operator-based linearization (OBL) that simplifies the construction of the Jacobian matrix and residuals. By employing the OBL approach, we transform the discretized nonlinear conservation equations into a quasi-linear form using state-dependent operators, which significantly enhances the simulation efficiency when dealing with highly nonlinear physical problems. We describe the operator forms of the mass, energy, and momentum conservation equations, taking into account both the wellbore and the reservoir. Furthermore, we provide a comprehensive description of the decoupled velocity formulation and the solution strategy, following the principles of the OBL approach.

In Chapter 3, we present a comprehensive discussion on the coupled wellbore and wellbore model, which is based on a multisegmented wellbore concept. We extend the OBL approach to encompass both the wellbore and the reservoir domains. By conducting a series of benchmark tests and comparing the results with those obtained from a legacy commercial simulator, we demonstrate that the extended OBL scheme significantly enhances computational efficiency while maintaining controlled accuracy and converging to the simulation results. Furthermore, the utilization of the multisegmented wellbore model enables us to incorporate more sophisticated wellbore configurations, including the co-axial wellbore model. Additionally, the decoupled velocity formulation exhibits flexibility and generalization, allowing us to effectively model reactive scenarios near the wellbore region, such as calcium carbonate dissolution tests.

In all simulation tests, OBL resolution plays an important role in the accuracy of the OBL approach. With increasing degrees of freedom in the simulation problem, DARTS (DelftAdvancedResearch TerraSimulator) shows its advantages to speed up the modeling process. OBL also facilitates the development of the nonlinear solver since the nonlinearity of the residual is now lumped into the operators. In Chapter 4, we developed the nonlinear solver embedded in an operator-based linearization framework based on the trust-region technique applied for CCUS processes. By tracking the nonlinear tra-

jectory and segmenting the problem's parameter space into a series of trust regions, we ensure that the hyperbolic operators maintain their second-order behavior, remaining either convex or concave. We approximate these trust regions during the solution process by identifying the boundaries of convex regions through directional derivative analysis. By conducting multiple trial trajectories on binary and ternary diagrams, we verify the correct detection of these boundaries by our algorithm. Moreover, our technique is computationally efficient as we do not compute the entire Hessian matrix but instead calculate the directional derivative while tracking the nonlinear update. Once all the boundaries along the nonlinear trajectory are detected, our proposed nonlinear solver locally constrains the composition update across the boundaries of these regions. We evaluate the performance of our nonlinear solver on various reservoir models, ranging from single cell to fully three-dimensional heterogeneous models. Our numerical results demonstrate that the trust-region solver effectively prevents overshoots in the nonlinear update, leading to superior convergence compared to conventional nonlinear solvers.

In Chapter 5, we investigated the application of deep learning for compositional transport simulation applied for CCUS. in the first part, we focused on physics-informed neural networks for compositional transport. In this study, we introduce a novel approach called sequential training PINN (Physics-Informed Neural Networks) for simulating two-phase transport in porous media. The core idea is to train the neural network to solve the partial differential equation (PDE) in successive time segments rather than attempting to train it for the entire time domain simultaneously. Our observations reveal that sequential training yields more accurate solutions compared to the standard training approach used in conventional two-phase problems.

Moreover, we extend the sequential training methodology to tackle compositional problems, where non-linearity becomes more pronounced due to complex phase transitions. To evaluate the effectiveness of our approach, we conducted tests on both miscible and immiscible scenarios. The results demonstrate that the sequential training approach outperforms the standard training method in terms of accuracy for both transport scenarios. In the second part of this chapter, we focus on the Application of Deep Neural Networks to the Operator Space of Nonlinear PDE for Physics-Based Proxy Modelling. In this study, we employ deep neural networks to approximate the operators of a nonlinear partial differential equation (PDE) within the Operator-Based Linearisation (OBL) simulation framework. Our aim is to identify the physical space for a physics-based proxy model with reduced degrees of freedom. To enhance the predictive accuracy of the proxy model, we utilize observations from a high-fidelity model within a supervised learning scheme to directly train the PDE operators.

Specifically, we focus on training the governing operators of a pseudo-binary gas vaporization problem using a transfer learning scheme. Our methodology involves a two-stage process. In the first stage, we utilize labeled data from the full compositional problem, as well as an analytical physics-based approximation of the operator space, to train the neural network. This initial training stage helps establish a foundation for the network's understanding of the problem.

In the second stage, we incorporate a fully implicit PDE solver directly into the neural network's loss function. This is achieved through the inclusion of a Lebesgue integration of the shocks in space and time. By integrating the PDE solver in the loss function, we can

effectively capture the complex dynamics of the system and further refine the network's predictive capabilities.

Overall, our methodology combines transfer learning, supervised learning, operator-based linearization, and the integration of a fully implicit PDE solver to improve the accuracy and performance of the physics-based proxy model.

Chapter 6 serves as the concluding chapter of this dissertation, providing an overview and summary of the key findings and contributions from each preceding chapter. Additionally, it explores potential future directions and aspects for further research in relation to each chapter's proposed ideas and approaches.

SAMENVATTING

Boorgatoperaties spelen een cruciale rol bij het beheer van verschillende ondergrondse activiteiten met betrekking tot energie, waaronder energieopslag, geothermische energieproductie, CO_2-vastlegging, olie- en gaswinning, afvalwaterverwijdering en thermische terugwinningsprocessen. De laatste tijd zijn intelligente puttechnologieën gebruikt, zoals lang afwijkende multilaterale putten die zijn uitgerust met geavanceerde instroomregelkleppen, om de winstgevendheid in het veld en de operationele betrouwbaarheid te verbeteren. Bovendien worden verschillende boorputontwerpen gebruikt om efficiënt warmte uit de ondergrond te halen in geavanceerde geothermische benaderingen. Om nauwkeurige modellering, voorspelling en optimalisatie van al deze energieproductieprocessen te garanderen, is het essentieel om volledig gekoppelde modellen te gebruiken die de thermische meerfasencompositiestroom en het transport in zowel het reservoir als de boorgaten nauwkeurig vastleggen.

In Hoofdstuk 2 presenteren we een gedetailleerde uitleg van ons numerieke raamwerk en heersende vergelijkingen voor zowel het reservoir als de boorput. We introduceren een nieuwe molaire formuleringsbenadering genaamd operator-gebaseerde linearisatie (OBL) die de constructie van de Jacobiaanse matrix en residuen vereenvoudigt. Door gebruik te maken van de OBL-benadering, transformeren we de gediscretiseerde niet-lineaire behoudsvergelijkingen in een quasi-lineaire vorm met behulp van toestandsafhankelijke operatoren, wat de simulatie-efficiëntie aanzienlijk verbetert bij het omgaan met zeer niet-lineaire fysieke problemen. We beschrijven de operatorvormen van de vergelijkingen voor massa-, energie- en momentumbehoud, rekening houdend met zowel de boorput als het reservoir. Verder geven we een uitgebreide beschrijving van de ontkoppelde snelheidsformulering en de oplossingsstrategie, volgens de principes van de OBL-benadering.

In Hoofdstuk 3 presenteren we een uitgebreide discussie over de gekoppelde boorput en het boorputmodel, dat gebaseerd is op een multigesegmenteerd boorputconcept. We breiden de OBL-benadering uit om zowel het boorgat als het reservoirdomein te omvatten. Door een reeks benchmarktests uit te voeren en de resultaten te vergelijken met die van een oude commerciële simulator, tonen we aan dat het uitgebreide OBL-schema de rekenefficiëntie aanzienlijk verbetert, terwijl de gecontroleerde nauwkeurigheid behouden blijft en convergeert naar de simulatieresultaten. Bovendien stelt het gebruik van het multigesegmenteerde boorputmodel ons in staat om meer geavanceerde boorconfiguraties op te nemen, waaronder het coaxiale boorputmodel. Bovendien vertoont de ontkoppelde snelheidsformulering flexibiliteit en generalisatie, waardoor we reactieve scenario's in de buurt van het boorgatgebied effectief kunnen modelleren, zoals tests voor het oplossen van calciumcarbonaat.

In alle simulatietesten speelt OBL-resolutie een belangrijke rol in de nauwkeurigheid van de OBL-benadering. Met toenemende vrijheidsgraden in het simulatieprobleem, toont DARTS zijn voordelen om het modelleringsproces te versnellen. OBL vergemak-

kelijkt ook de ontwikkeling van de niet-lineaire oplosser, aangezien de niet-lineariteit van het residu nu op één hoop wordt gegooid in de operatoren. In Hoofdstuk 4 hebben we de niet-lineaire oplosser ontwikkeld die is ingebed in een operatorgebaseerd linearisatieraamwerk op basis van de trust-region-techniek die wordt toegepast voor CCUS-processen. Door het niet-lineaire traject te volgen en de parameterruimte van het probleem te segmenteren in een reeks vertrouwensgebieden, zorgen we ervoor dat de hyperbolische operatoren hun gedrag van de tweede orde behouden en convex of concaaf blijven. We benaderen deze vertrouwensregio's tijdens het oplossingsproces door de grenzen van convexe regio's te identificeren door middel van directionele afgeleide analyse. Door meerdere proeftrajecten uit te voeren op binaire en ternaire diagrammen, verifiëren we de correcte detectie van deze grenzen door ons algoritme. Bovendien is onze techniek rekenkundig efficiënt omdat we niet de volledige Hessische matrix berekenen, maar in plaats daarvan de directionele afgeleide berekenen terwijl we de niet-lineaire update volgen. Zodra alle grenzen langs het niet-lineaire traject zijn gedetecteerd, beperkt onze voorgestelde niet-lineaire oplosser lokaal de compositie-update over de grenzen van deze regio's. We evalueren de prestaties van onze niet-lineaire oplosser op verschillende reservoirmodellen, variërend van enkele cellen tot volledig driedimensionale heterogene modellen. Onze numerieke resultaten tonen aan dat de trust-region-oplosser effectief overschrijdingen in de niet-lineaire update voorkomt, wat leidt tot superieure convergentie in vergelijking met conventionele niet-lineaire oplossers.

In Hoofdstuk 5 onderzochten we de toepassing van deep learning voor compositietransportsimulatie toegepast voor CCUS. in het eerste deel hebben we ons gericht op fysica-geïnformeerde neurale netwerken voor samenstellingstransport. In deze studie introduceren we een nieuwe benadering genaamd sequentiële training PINN (Physics-Informed Neural Networks) voor het simuleren van tweefasig transport in poreuze media. Het kernidee is om het neurale netwerk te trainen om de partiële differentiaalvergelijking (PDE) eerder in opeenvolgende tijdsegmenten op te lossen

PREFACE

Dear reader,

This dissertation is the product of the research work conducted in Delft Advanced Research Terra Simulator (DARTS) group in the Delft University of Technology between August 2018 and May 2023, under the supervision of my promotors, Dr. Denis Voskov and Prof. David Bruhn. I studied in the Reservoir Engineering section, part of the Department of Geoscience & Engineering within the Faculty of Civil Engineering & Geosciences. The objective of this research was designed to develop an efficient and accurate simulator to capture the highly nonlinear coupled wellbore and reservoir behavior. The Operator-based linearization (OBL) approach is applied to provide an approximate representation of the exact physics with controlled error. Machine learning and specifically neural networks have been applied for proxy modeling and speed-up. The scientific motivations of this research are carefully highlighted in Chapter 1.

In this book, most of the chapters were already published in journal articles or in conference papers throughout the years. In Chapter 2 where I described the methodology of this research, I endeavored to avoid the slightly inconsistent notations and repetitions; however, these small mistakes may still be present. I hope that all readers can enjoy reading this book.

I would like to utilize the remainder of this brief preface to share my personal experience as a Ph.D. candidate at TU Delft. I vividly remember the first day of reaching the Netherlands. I had a direct flight in the evening from Milan where I established some great friends and memories. It was not that easy to say goodbye to all the memories and great moments I had made in that city. When I arrived at Schipol it was my first time in Amsterdam. It was raining cats and dogs. Luckily, a taxi driver was waiting for me to take me home. Wow! I was pleasantly surprised by the fact that TU Delft arranged a taxi to pick me up from the airport. Then the taxi driver forgot the key to my house so he had to take me to a hotel nearby. The next day morning I went to the TU Delft and dived into the new journey of my life.

My doctoral study is a long story filled with unforgettable experiences. While there were moments of disappointment, I learned that the life of a Ph.D. student is not a sprint but rather an enduring journey. Over the past years, my main activities have revolved around extensive reading and paper writing, algorithm development, delivering presentations, and coding. Computer programming has consistently occupied a significant portion of my time, and it has grown into an activity that I unexpectedly developed a strong affinity for. Furthermore, I have taken on various responsibilities, including organizing practical sessions for master courses, delivering a few lectures, and assisting with student supervision during their thesis projects.

Apart from the demanding tasks of my Ph.D., the previous years have been brimming with delightful experiences. These encompass engaging in social gatherings with fellow

colleagues, joining football matches alongside Ph.D. students from various disciplines, and embarking on exhilarating runs with geo-runners.

I express my gratitude to you for choosing to review my thesis and I sincerely hope that you will discover valuable insights within its contents.

Kiarash Mansour Pour
Delft, June 2023

1

INTRODUCTION

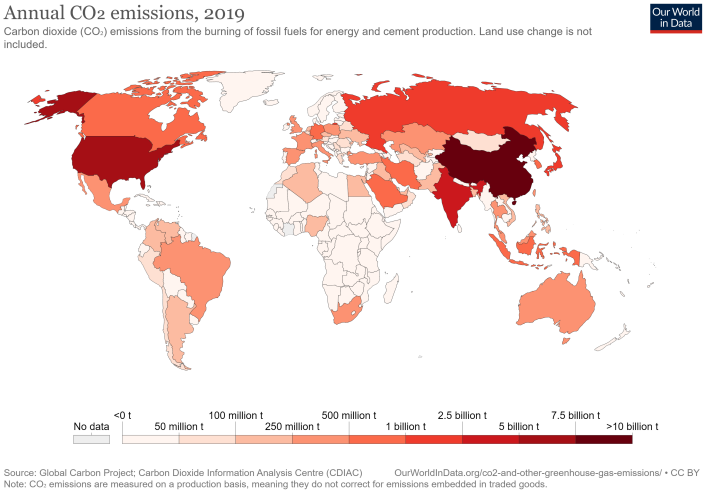
1.1. ENERGY TRANSITION

In recent decades, the world has witnessed a growing recognition of the urgent need to address climate change and reduce greenhouse gas emissions. As societies increasingly grapple with the consequences of fossil fuel dependence, an energy transition towards renewable sources has emerged as a pivotal solution to mitigate environmental impacts and ensure a sustainable future. Global climate change is one universal issue mainly caused by the emissions of greenhouse gases (GHGs). Carbon dioxide (CO_2) emissions are the principal driver of global warming. The burning of fossil fuels, such as coal, oil and gas, for energy and power, is the primary source of GHG emissions. According to *Our World in Data*, the global annual CO_2 emissions in 2019 is over 35 billion tonnes due to the burning of fossil fuels for energy and cement production (Fig. 1.1(a)) [1].

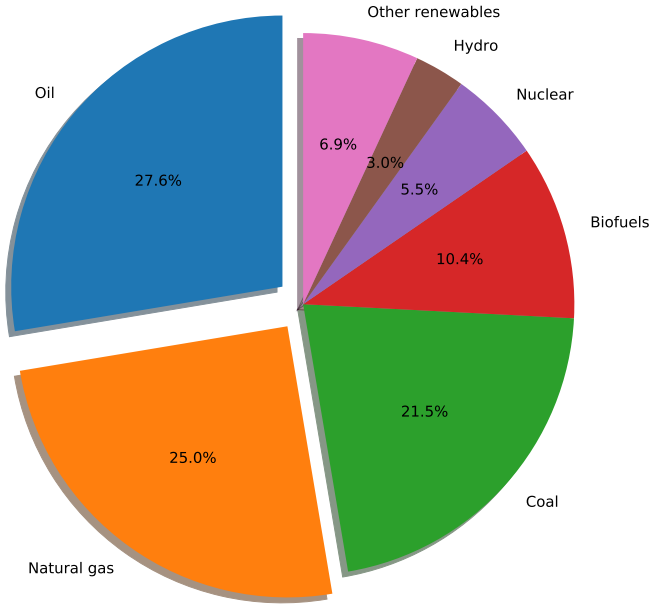
Subsurface can provide many resources for innovative low-carbon energy solutions such as geothermal energy production, hydrogen storage, carbon dioxide sequestration, etc. Geothermal energy, which has been ignored in many regions around the world for a long time, is now gaining more recognition [2, 3]. In addition to having a minimal carbon footprint, geothermal energy offers a reliable and stable source of power compared to other renewable energy sources like wind and solar energy. It can act as a base load and is not significantly affected by weather conditions or seasonal variations. The potential competitiveness of geothermal energy is highlighted by its widespread distribution and impressive reserves [4]. This substantial theoretical potential is visually depicted in Figure 1.2 which presents the global distribution of temperature gradients derived from heat flow calculations. Regions with higher temperature gradients typically indicate greater geothermal potential and naturally attract significant attention.

However, transitioning to low-carbon energy sources alone may not be sufficient to meet the ambitious emission reduction targets necessary to mitigate climate change effectively. The concentration of carbon dioxide in the atmosphere has already surpassed a safe level, necessitating additional efforts to remove this excess carbon actively. Carbon removal technologies, also known as negative emissions technologies, play a crucial role

1



(a)



(b)

Figure 1.1: (a) The yearly release of carbon dioxide (CO_2) into the atmosphere resulting from the combustion of fossil fuels for energy generation and the production of cement; data sourced from *Our World in Data (2019)* indicates the projected global primary energy consumption for the year 2040.; data *World Energy Outlook (2018)* by the *International Energy Agency*.

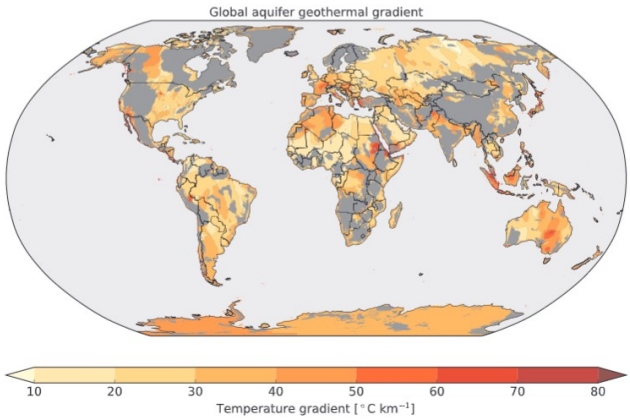


Figure 1.2: Geothermal gradients in aquifers around the world [5].

in this regard. These innovative approaches, including direct air capture, afforestation, enhanced weathering, and carbon capture, utilization, and storage (CCUS), offer the potential to extract carbon dioxide from the atmosphere and store it underground or utilize it in beneficial ways. By actively removing carbon from the atmosphere, we can compensate for remaining emissions and accelerate the transition towards a carbon-neutral or even carbon-negative future. There have been some CO_2 storage projects related to enhanced oil recovery (EOR) around the world since the 1970s (Table 1.1) in which the storage capacity is above 0.4 Mt/year according to the data from *International Energy Agency* [6]. These target geological formations, such as depleted oil and gas reservoirs, provide potential storage capacity for large-scale long-term CO_2 sequestration.

Undoubtedly, the energy transition towards a low-carbon future with active carbon removal and exploiting renewable geothermal energy poses significant challenges. A model that captures multiphase compositional flow and transport is required to simulate CO2 use and storage (CCUS) in subsurface reservoirs with complicated heterogeneous structures. Moreover, to model the geothermal energy production we need to consider the thermal effect. Lastly, It is inevitable to overlook the role of well and borehole operations for the successful management of any energy-related subsurface activities e.g., energy storage, geothermal energy production, carbon capture and sequestration, oil and gas production, wastewater disposal, and thermal recovery processes.

1.2. THE NEED FOR RESERVOIR SIMULATION

A comprehensive understanding of fluid and heat transport, their physical and chemical interactions, as well as their impact on geological formations is essential in the field of geoengineering, whether applied to energy production (such as hydrocarbons and geothermal energy) or storage (like CO_2 or hydrogen storage). Accurate and scalable models for simulating fluid and heat transport in subsurface porous media are of paramount importance to meet scientific, economic, and societal expectations in the successful development of energy resources and storage plans. These computer models and their

resulting predictions significantly contribute to the efficiency and safety of operations in production and storage facilities across various geo-engineering applications. These predictions offer valuable insights for optimizing hydrocarbon extraction, outlining energy production strategies, determining the lifespan of geothermal systems, evaluating the storage potential of underground formations for CO2 or hydrogen, and much more. Reservoir simulation is one essential tool to maximize both the economy of the field and the reliability of the operations of sustainable subsurface applications.

Numerical simulation, a tool developed by combining physics, mathematics, and computer programming, is an efficient way to understand complex fluid flow in subsurface reservoirs with applications to the evaluation of hydrocarbon recovery, energy efficiency, performance analysis, and various optimization problems [7–9]. It involves solving the partial differential equations (PDEs) governing coupled multiphase flow and transport in porous media with highly nonlinear physics [10, 11].

In this study, we are interested in a fully implicit coupling of the complex Thermal-Hydraulic-Chemical (THC) flow model in the wellbore and reservoir due to unconditional stability. In the fully implicit approach, we use Newton's method to linearize and solve a set of nonlinear equations. Linearization of discrete mass and energy governing equations of multiphase, multicomponent flow, and transport is a challenging task due to the highly nonlinear coupling and complex thermodynamic phase behavior that needs to resolve multiphase partitioning of different components at each nonlinear iteration to accurately evaluate the fluid/rock properties [12, 13].

The linearization stage for such problems is always a demanding task due to the complexity of Jacobian assembly in the presence of fully coupled physical-chemical interactions. A new approach for the linearization of governing equations called operator-based linearization (OBL), was proposed by [14] following ideas from tie-simplex parametrization [15, 16]. In this approach, the exact physics kernels of the governing partial differential equations were approximated using abstract algebraic operators. Later this technique was extended and implemented in the open-source Delft Advanced Research Terra Simulator [17]. DARTS is a scalable parallel modeling framework that aims to accelerate the simulation performance while capturing multi-physics geo-application processes such as hydrocarbon production[18, 19], geothermal energy extraction [20, 21] and CO_2 sequestration [22–24].

1.3. CHALLENGES

The wellbore plays an almost indispensable role in all geo-energy applications. To effectively design, predict, and optimize processes crucial for energy production, precise fully-coupled models for thermal multiphase flow in both the reservoir and boreholes are essential.

Modeling such wells presents several challenges. To begin with, accurately representing the complex physics involved in the wellbore, including thermal dynamics, multiphase flow, and multi-component interactions, is a formidable task. Furthermore, in applications related to energy transition, it is imperative to account for chemical interactions between the wellbore and the flowing fluids.

The interaction between the wellbore and the reservoir introduces additional layers of complexity. This complexity arises from the deviation of flow behavior in the

wellbore from Darcy's law. Consequently, when solving for the momentum equation, one must consider pressure losses resulting from friction, acceleration, and gravitational forces acting within the fluid. Additionally, the model needs to be sufficiently robust to accommodate more intricate and intelligent well topologies while maintaining reliability. Moreover, it should demonstrate computational efficiency while encompassing the complete range of physical phenomena.

Moreover, for the simulation of CO_2 utilization and storage (CCUS) in subsurface reservoirs with complicated heterogeneous structures, a model that includes multiphase compositional flow and transport is needed. The governing equations are highly nonlinear due to the complex thermodynamic behavior, which involves the appearance and disappearance of multiple phases. This nonlinearity causes difficulties for a nonlinear solver, leading to an increase in computational costs as smaller time steps are required for the simulation. Consequently, robust and efficient techniques are needed to solve the resulting nonlinear system of algebraic equations.

The literature provides various nonlinear solvers tailored to compositional formulations. An intriguing approach introduced by [25] involves the continuation method, which smoothly transitions a parameter from 0 to 1 over timesteps, enabling precise control of residuals during nonlinear trajectory integration in parameter space. A Dissipation-Based Nonlinear Solver, as recently proposed [26], utilizes numerical dissipation to construct a homotopy of discrete governing equations, incorporating a continuation parameter to limit dissipation and maintain solution accuracy. Additionally, the flux-based trust region method, initially developed by [27] for two-phase immiscible flow, demonstrates unconditional convergence by constraining saturation updates based on flux function inflection points. Extensions by [28–30] expanded this method to scenarios with buoyancy, capillary, and viscous forces, even for black oil three-phase physics. However, there's a notable gap in trust region solvers for molar formulations. While [31] employed tie-lines to parametrize compositional space in flow simulations, trust-region correction strategies for molar formulations proved less robust compared to those designed for natural formulations. [32], in detecting phase boundaries, overlooked inflection lines within two-phase zones, explaining the complexities and computational expenses associated with inflection line detection in compositional problems, necessitating second-order derivatives, and Hessian analysis.

Machine learning, particularly deep learning [33], has gained considerable traction in computer science and engineering fields. Physics-informed neural networks (PINNs) have emerged as a valuable tool for solving problems where engineering conservation equations and constitutive closure relationships are known, but labeled data is unavailable [34]. PINNs, constructed with multiple hidden layers and nonlinear activation functions, enable the approximation of complex nonlinear solutions. Consequently, they have found applications in diverse domains governed by differential equations, including the Euler equation [35, 36], gas dynamics [37, 38], water dynamics [39], and chemical kinetics [40, 41]. PINNs showcase their versatility in various applications such as data assimilation, parameter identification, and uncertainty quantification [34, 42–44].

Recent developments have extended PINNs to subsurface flow and transport, addressing two-phase immiscible transport in porous media, characterized by the nonlinear first-order hyperbolic Buckley-Leverett equation [45]. While standard PINNs ini-

tially struggled to find solutions for cases with steep saturation fronts and nonconvex flux functions, the addition of an artificial diffusion term to the conservation equation allowed them to approximate the true solution [45]. Alternative approaches have also been explored, such as the introduction of physics-informed attention-based neural networks (PIANNs) [46] and the incorporation of entropy and velocity constraints into the neural network residual [47]. However, saving a pure hyperbolic one-oder PDE remains a challenge for standard PINNs. Moreover, it has not been tested on compositional problems which is more complicated due to the phase appearance and disappearance.

Furthermore, compositional issues involve the interaction of numerous components, necessitating a set of equations equal to the number of components for each grid cell. This can result in high computational costs for a high-fidelity model fine-scale model.

1.4. THESIS OBJECTIVES

This thesis revolves around three distinct facets of reservoir simulation in the context of energy transition. First, we have developed a new computational framework in DARTS applying the general decoupled velocity formulation and extending OBL to couple well and reservoir model. Well and reservoir are both discretized similarly into nodes and connections following the general unstructured grid framework [connection list approach, e.g., 48] using the finite volume method. Total velocity serves as an additional nonlinear unknown written at each interface (connection) on the total computational domain and bounded by a suitable momentum equation. Similar to the staggered gridding method, this framework adopts a simultaneous approach by coupling the mass and energy balance equations at the center of each cell with the momentum balance equations at the cell interfaces. Moreover, transforming both reservoir and well nonlinear governing equations into an operator form benefits from OBL techniques and reduce further the computational cost related to linearization.

Next, we focus on accelerating near-wellbore modeling by developing an advanced nonlinear solver for high-fidelity forward simulation and a proxy model for inverse simulation. We first present an advanced nonlinear solver based on a generalization of the trust-region technique for compositional multiphase transport applied for CCUS. We investigate the nonlinearity of convective operators written in fractional flow form and detect boundaries of the trust region for the hyperbolic operator by assembling the directional approximation of the Hessian matrix. Next, we design the nonlinear solver in which we track the nonlinear trajectory for binary and ternary kernel in OBL parameter space and approximate these trust regions in the solution process via directional analysis of the derivative. By drawing some trial Newton trajectories on OBL parameter space, we observe that our directional analysis of derivatives predicts the boundaries of these trust regions correctly. Furthermore, it is less computationally expensive than computing the full Hessian matrix. In conclusion, we evaluate the effectiveness of the new nonlinear solver through extensive testing on various complex examples.

Furthermore, we shift our focus towards the application of machine learning techniques in porous media. A novel method utilizing physics-informed neural networks (PINNs) has been introduced as an alternative approach for solving partial differential equations (PDEs). Unlike conventional machine learning algorithms that heavily rely on labeled datasets for training, PINNs have the capability to train the neural network

using unlabeled data. The potential of this approach has been investigated in the context of multiphase flow and transport in porous media. However, it should be noted that when it comes to nonlinear hyperbolic transport equations, the performance of the solution deteriorates noticeably. In the first part, we introduce sequential training for the physics-informed neural network. The primary concept revolves around the retraining of the neural network to address the solution of partial differential equations (PDEs) by dividing the time domain into successive time segments, rather than training for the entire time domain concurrently. In the subsequent section, departing from the approach of PINNs, which relies on employing a neural network as a specialized solver to estimate the solution of a given partial differential equation (PDE), we adopted an alternative method. Our approach involved training the PDE operators using a neural network and then integrating them into the DARTS framework. By utilizing a fully implicit finite volume solver, we were able to effectively obtain the desired solution for the PDE.

To summarize, the main objective of this thesis is to develop a robust numerical framework to accurately model subsurface physical phenomena applied for energy transition applications. The research objectives addressed in this work are:

- Developing a high-fidelity framework to model coupled wellbore and reservoir for geo-energy applications.
- Accelerate the convergence of a high-fidelity model using the efficient nonlinear solver for OBL utilization in energy transition applications.
- Develop proxy modeling methodology using physics-driven machine learning for inverse modeling in reservoir simulation.

1.5. THESIS OUTLINE

This thesis comprises six chapters following journal articles published or presented at the conferences. The thesis starts with an introductory chapter 1. Chapter 2 describes the numerical models, including the conservation equations used for general-purpose coupled reservoir and wellbore simulation and their operator forms within the Operator-based linearization framework.

In chapter 3, we have developed a new computational framework that can simulate thermal multiphase, multi-component flow both along the wellbore and the reservoir. The implementation is based on an operator-based linearization method in which the governing equations are represented in an operator form. This framework was tested for several complex physical kernels including thermal compositional multiphase reactive flow and transport. The proposed model was validated using a comparison with analytic and numerical results.

In chapter 4, we focus on the simulation of multiphase multicomponent flow applied for carbon capture utilization and storage simulation. We investigate the nature of nonlinearities in CCUS simulations and suggest solutions to a general compositional problem. We present an advanced nonlinear solver based on a trust-region technique aimed at solving multiphase multi-component flow and transport problems more efficiently. The trust region solver is based on the analysis of multi-dimensional tables connected

to highly nonlinear convection operators parameterized in physical space. These operators are associated with the governing equations and are built for a newly introduced Operator-Based Linearization approach.

In chapter 5 we introduce the proxy model and explore the application of deep neural network for multiphase compositional transport. We develop a novel training scheme for a physics-informed neural network (PINNs) that overcomes the limitations of standard PINNs for capturing a shock. The main concept is to retrain the neural network to solve the PDE over successive time segments rather than train for the entire time domain at once. We observe that sequential training can capture the solution more accurately concerning the standard training method. In the second part of this chapter, we utilize deep neural networks to approximate operators of a nonlinear partial differential equation (PDE), within the Operator-Based Linearisation (OBL) simulation framework, and discover the physical space for a physics-based proxy model with reduced degrees of freedom.

Finally, chapter 6 concludes the work and defines perspectives for further research.
9/

Table 1.1: CCUS projects around the world according to the data from *International Energy Agency* [6]

Country	Project	Operation date	Source of CO_2	CO_2 capture capacity (Mt/year)	Primary storage type
United States (US)	Terrell natural gas plants (formerly Val Verde)	1972	Natural gas processing	0.5	EOR
US	Enid fertiliser	1982	Fertiliser production	0.7	EOR
US	Shute Creek gas processing facility	1986	Natural gas processing	7.0	EOR
Norway	Sleipner CO_2 storage project	1996	Natural gas processing	1.0	Dedicated
US/Canada	Great Plains Synfuels (Weyburn/Midale)	2000	Synthetic natural gas	3.0	EOR
Norway	Snohvit CO_2 storage project	2008	Natural gas processing	0.7	Dedicated
US	Century plant	2010	Natural gas processing	8.4	EOR
US	Air Products steam methane reformer	2013	Hydrogen production	1.0	EOR
US	Lost Cabin Gas Plant	2013	Natural gas processing	0.9	EOR
US	Coffeyville Gasification	2013	Fertiliser production	1.0	EOR
Brazil	Petrobras Santos Basin pre-salt oilfield CCS	2013	Natural gas processing	3.0	EOR
Canada	Boundary Dam CCS	2014	Power generation (coal)	1.0	EOR
Saudi Arabia	Uthmaniyah CO_2 -EOR demonstration	2015	Natural gas processing	0.8	EOR
Canada	Quest	2015	Hydrogen production	1.0	Dedicated
United Arab Emirates	Abu Dhabi CCS	2016	Iron and steel production	0.8	EOR
US	Petra Nova	2017	Power generation (coal)	1.4	EOR
US	Illinois Industrial	2017	Ethanol production	1.0	Dedicated
China	Jilin oilfield CO_2 -EOR	2018	Natural gas processing	0.6	EOR
Australia	Gorgon Carbon Dioxide Injection	2019	Natural gas processing	3.4-4.0	Dedicated
Canada	Alberta Carbon Trunk Line (ACTL) with Agrium CO_2 stream	2020	Fertiliser production	0.3-0.6	EOR
Canada	ACTL with North West Sturgeon Refinery CO_2 stream	2020	Hydrogen production	1.2-1.4	EOR

*Here only projects where CO_2 capture capacity is higher than 0.4 Mt/year is included in the table.

2

GOVERNING EQUATIONS FOR COUPLED WELL AND RESERVOIR

Summary

A detailed description of the governing equations for the thermal multiphase compositional formulation of the coupled wellbore and reservoir systems is provided. It presents a thorough explanation of the wellbore momentum equation in addition to a full description of the momentum equations within the reservoir. Mass, energy, and momentum conservation equations are presented in operator form, following the concept of the OBL approach. Furthermore, a detailed explanation is provided for the solving strategy employed by the decoupled velocity engine. This approach considers velocity as an additional degree of freedom, enabling a more accurate simulation of the flow dynamics inside the wellbore. By treating velocity as a separate parameter, the decoupled velocity engine strategy offers enhanced precision in capturing the fluid flow and momentum equation inside the wellbore. Additionally, the conventional DARTS solution strategy is also discussed. In addition to the solving strategies, the sparse matrix structure utilized in both the decoupled velocity engine and the conventional DARTS solution strategy is explained.

2.1. GOVERNING EQUATIONS FOR FULLY COUPLED APPROACH

For the investigated domain with volume Ω, bounded by surface Γ, the mass and energy conservation can be expressed in a uniformly integral way along with the proper momentum equation as shown in the table. 2.1

Description	Equation
Conservation of mass and energy	$\frac{\partial}{\partial t}\int_{\Omega} M^k d\Omega + \int_{\Gamma}^{n_p} F^k.nd\Gamma = \int_{\Omega} q^k$
Mass accumulation	$M^k(\omega, \upsilon_m) = \phi \sum_{j=1} x_{cj}\rho_j s_j$
Energy accumulation	$M^{ke}(\omega, \upsilon_m) = \phi \sum_{j=1}^{n_p} x_{cj}\rho_j s_j U_j - (1-\phi)U_r$
Mass flux	$F^k(\xi, \omega, \upsilon_m) = \sum_{j=1}^{n_p} x_{cj}\rho_j \upsilon_j$
Energy flux	$F^{ke}(\xi, \omega, \upsilon_m) = \sum_{j=1}^{n_p} x_{cj}\rho_j \upsilon_j h_j + k\nabla T$
Reservoir momentum equation	$\upsilon_j = -\frac{\mathbf{K}k_{rj}}{\mu_j}(\nabla p_p - \rho_j g \nabla D)$
Well momentum equation	$\frac{\partial p^w}{\partial z} = -\rho_m g - \frac{\partial p^w}{\partial t}(\rho_m V_m) - \frac{\partial(\rho_m V_m^2)}{\partial z} - \frac{f_{tp}\rho_m V_m^2}{2d_{in}} - R_m$

Table 2.1: Mass, energy, and momentum balance equations

Here, we introduce all terms in the equations as functions of spatial coordinate $\boldsymbol{\xi}$ and physical state $\boldsymbol{\omega}$:

- $\phi(\boldsymbol{\xi}, \boldsymbol{\omega})$- porosity,
- $x_{cj}(\boldsymbol{\omega})$- the mole fraction of component c in phase j,
- $s_j(\boldsymbol{\omega})$ - phase saturations,
- $\rho_j(\boldsymbol{\omega})$- phase molar density,
- $\mathbf{v}_j(\boldsymbol{\xi}, \boldsymbol{\omega})$- phase velocity,
- $U_j(\boldsymbol{\omega})$ - phase internal energy,
- $U_r(\boldsymbol{\omega})$ - rock internal energy,
- $h_j(\boldsymbol{\omega})$ - phase enthalpy,
- k - thermal conduction.

Next, we formulate a one-dimensional momentum balance equation for the entire fluid in the wellbore. Assuming the coordinate z points along the well, this equation is given by [49] and [50].

$$\frac{\partial p^w}{\partial z} = -\rho_m g - \frac{\partial p^w}{\partial t}(\rho_m V_m) - \frac{\partial(\rho_m V_m^2)}{\partial z} - \frac{f_{tp}\rho_m V_m^2}{2d_{in}} - R_m \tag{2.1}$$

We focus on well momentum equation in table 2.1, where $\rho_m(\boldsymbol{\omega})$ is the total mixture density, g is the gravitational acceleration in the z direction, d_{in} is the internal diame-

ter of the well, where f_{tp} is the friction factor, which is a function of the dimensionless Reynolds number Re (ratio of inertial forces to viscous forces), d is the diameter of the segment, and V_m is the velocity of the fluid mixture in the segment.

$$f_{tp} = \begin{cases} 16/Re, & Re \leq 2000 \text{ (laminar flow)} \\ 1/\left(-3.6\log\left(\frac{6.9}{Re} + (\frac{\epsilon}{3.7d})^{\frac{10}{9}}\right)\right)^2, & Re \geq 4000 \text{ (Torbulent flow)} \\ 16/2000 + k_f(R_e - 2000), & 2000 < Re < 4000 \end{cases} \tag{2.2}$$

The general momentum equation 2.1 can be simplified to the steady state momentum equation as follows

$$\frac{\partial p^w}{\partial z} = (\frac{\partial p}{\partial z})_h + (\frac{\partial p}{\partial z})_f + (\frac{\partial p}{\partial z})_a. \tag{2.3}$$

The equation above simply states the total pressure loss over any control volume (segment) of the well as the sum of three components' forces (acceleration, friction, and hydrostatic).

The momentum equation for the reservoir part is simplified to Darcy's law, and phase velocity is computed as:

$$v_j = -\frac{\mathbf{K}k_{rj}}{\mu_j}(\nabla p_j - \rho_j g \nabla D). \tag{2.4}$$

where

- $\mathbf{K}(\boldsymbol{\xi})$ – permeability tensor,
- $k_{rj}(\boldsymbol{\omega})$ – relative permeability,
- $\mu_j(\boldsymbol{\omega})$ – phase viscosity,
- $\boldsymbol{p}_j(\boldsymbol{\omega})$ – vector of pressures in phase j,
- $\rho_j(\boldsymbol{\omega})$ – phase density,
- $\boldsymbol{d}(\boldsymbol{\xi})$ – vector of depths (positive downwards).

The main source of nonlinearity is related to the use of the Fully Implicit Method (FIM) for time approximation of the governing equations which requires the flux term in governing Equation 2.1 to be defined based on of the nonlinear unknowns at a new timestep $(n+1)$. The closure assumption of instantaneous thermodynamic equilibrium further increases the nonlinearity. We used the overall molar formulation suggested by [51]. In this formulation, the following system must be solved at any grid block contain-

ing a multiphase (n_p) multi-component n_c mixture:

$$F_c = z_c - \sum_{j=1}^{n_p} v_j x_{cj} = 0, \tag{2.5}$$

$$F_{c+n_c} = f_{c1}(p,T,x_1) - f_{cj}(p,T,x_j) = 0, \tag{2.6}$$

$$F_{j+n_c*n_p} = \sum_{c=1}^{n_c} (x_{c_1} - x_{cj}) = 0, \tag{2.7}$$

$$F_{n_p+n_c*n_p} = \sum_{j=1}^{np} v_j - 1 = 0. \tag{2.8}$$

Here $z_c = \Sigma x_{cj}\rho_j s_j / \rho_j s_j$ is overall composition and $f_{cj}(p,T,x_{cj})$ is the fugacity of component c in phase j. The solution of this system is called a multiphase flash [52] and needs to be applied at every nonlinear iteration [13]. The solution provides molar fractions for each component x_{cj} and phase fraction v_j.

2.2. OBL FORMULATION

Unlike the conventional DARTS formulation, in which the problem's primary unknowns were written on the cell centers and velocity was not one of them, the decoupled velocity formulation considers velocity as an additional nonlinear unknown (state) of the problem written at each interface between two nodes. On each node, the following independent variables are defined:

- $\omega = [\ p_i,\ h_i,\ \mathbf{z}_i\ ]$ which corresponds to pressure, enthalpy, and the vector of overall molar fractions respectively.

Note that the above variables are all defined at the center of the node. On each connection, we define total velocity:

- $\boldsymbol{v}_m$, Total velocity

2.2.1. GOVERNING EQUATIONS FOR FULLY COUPLED APPROACH

After discretizing governing Equations 2.1, using the finite volume scheme and backward Euler approximation in time, we transform the mass and energy residuals into an operator form as follows:

$$R_{nm}(\omega, v) = V_n\phi\Big(\alpha_c(\omega) - \alpha_c(\omega^n)\Big) + \Delta t \sum_l \beta_c^l(\omega)\, v_m^l(\boldsymbol{\xi}, \omega) = 0. \tag{2.9}$$

In Eq.2.9, operators read as:

$$\beta_c(\omega) = \sum_j x_{cj} f_j \rho_j, \tag{2.10}$$

$$\alpha_c(\omega) = \big(1 + c_r(p - p_{ref})\big) \sum_j x_{cj}\rho_j s_j, \tag{2.11}$$

$$\tag{2.12}$$

The discretized energy conservation equation in operator form can be written as:

$$\begin{aligned} R_{ne}(\boldsymbol{\omega}, v) &= V_n\phi\Big(\alpha_f(\boldsymbol{\omega}) - \alpha_f(\boldsymbol{\omega}^n)\Big) + (1-\phi)V_n U_r\Big(\alpha_{er}(\boldsymbol{\omega}) - \alpha_{er}(\boldsymbol{\omega}^n)\Big) \\ &+ \Delta t\sum_l \beta_e^l(\boldsymbol{\omega})\, v_m^l(\boldsymbol{\xi}, \boldsymbol{\omega}) \\ &+ \Delta t\sum_l \Gamma^l(T_i - T_j)\Big(\phi_0\gamma_{ef}(\boldsymbol{\omega}) + (1-\phi_0)k_r\alpha_{er}(\omega)\Big) = 0. \end{aligned} \tag{2.13}$$

$$\beta_e(\boldsymbol{\omega}) = \sum_j x_{cj} f_j \rho_j h_j, \tag{2.14}$$

$$\alpha_f(\boldsymbol{\omega}) = \Big(1 + c_r(p - p_{ref})\Big)\sum_j x_{cj}\rho_j s_j U_p, \tag{2.15}$$

$$\alpha_{er}(\boldsymbol{\omega}) = \frac{1}{1 + c_r(p - p_{ref})}, \tag{2.16}$$

$$\gamma_{ef} = \Big(1 + c_r(p - p_{ref})\Big)\sum_j s_j k_j, \tag{2.17}$$

The momentum equations are defined at the interface (connections of two nodes). For each connection between node block i and j, we write a discrete momentum equation in residual form as follows depending on the connection whether it is between wells or reservoir blocks.

$$\mathbf{R_c} = \begin{cases} v_{mix} + T_c\lambda(\omega)(P_i - P_j), & \text{reservoir connection,} \\ P_i - P_j - (\Delta P_h + \Delta P_f + \Delta P_h), & \text{well connection.} \end{cases} \tag{2.18}$$

In the Darcy velocity equation 2.18, T_c is the transmissibility and $\lambda(\boldsymbol{\omega})$ is the total mobility operator of the upwind grid block. For the momentum equation in a wellbore, ΔP^w is the pressure drop between two nodes, and ΔP_h^w, ΔP_f^w and ΔP_a^w are the hydrostatic, frictional and acceleration components of the pressure drop, respectively. For simplicity, we only take into account the friction losses. The frictional pressure difference between two nodes is defined as

$$\Delta P_f^w = (\frac{2 f_{tp}\rho^{seg} V_m^2}{d})\Delta x_i, \tag{2.19}$$

where f_{tp} is the friction factor which is a function of the dimensionless Reynolds number Re (ratio of inertial forces to viscous forces) explained previously, d is the diameter of the segment, and V_m is the velocity of the fluid mixture in the segment connection.

The hydrostatic pressure difference between two nodes is:

$$\Delta P_h^w = \rho_t g \Delta H, \tag{2.20}$$

Acceleration pressure losses are given by the following formula:

$$\Delta P_a^w = \frac{2 m_{in} V_m}{A}, \tag{2.21}$$

where $m_{in} = \sum_p \rho_p(\omega) Q_p$ is the mass flow rate of the mixture entering the segment defined as

$$m_{in} = \sum_{i}^{Nperf} \left(WI(p_u^w - p_{perf})\lambda(\omega)\rho_{tot}(\omega) \right)_i, \tag{2.22}$$

where ΔH is the height difference between two nodes and ρ_t is the total mass density. p_u^w is the pressure of the upwind node for connection c between two well segments and P_{Perf} is the pressure in the perforated reservoir block.

2.2.2. INTERPOLATION

The combination of different physical properties into a single nonlinear operator allows us to simplify the complicated nonlinear physics and implementation of the generic linearization approach. Instead of performing complex evaluations of properties and their derivatives with respect to nonlinear unknowns during the simulation, we can parameterize operators in the physical space at the pre-processing stage or adaptively with a limited number of supporting points [14]. The parameter space depends on the physical problem. For strongly nonlinear functions (e.g., capillary pressure), it is necessary to select a reasonable OBL resolution to characterize the physical space. A coarse OBL resolution may cause a larger deviation in the solutions [14, 24]. The governing equations are written in the form of state-dependent operators by following the OBL approach. The state-dependent operators can be parameterized (Fig. 2.1) with respect to nonlinear unknowns in multi-dimension tables under different resolutions. The values and derivatives of the operators in the parameter space can be interpolated and evaluated based on supporting points (Fig. 2.2). For the adaptive parameterization technique [14], the supporting points are calculated 'on the fly' and stored for later re-usage, which can save time for parameterization in high-dimension parameter space (i.e. in multi-component compositional simulations). At the same time, the Jacobian assembly becomes flexible with the OBL approach, even for very complex physical problems.

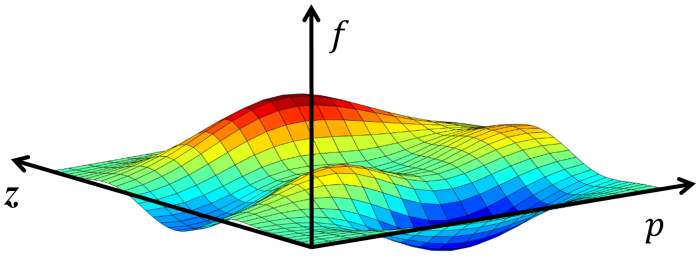


Figure 2.1: Parameterization of the arbitrary operator f based on pressure P and composition Z in 2D space with predefined OBL resolution.

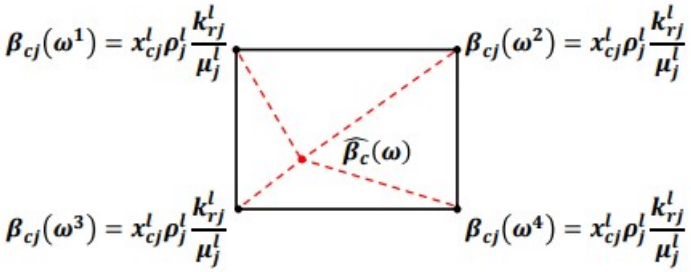


Figure 2.2: Schematic of interpolation process for β operator

2.3. SOLUTION OF EQUATIONS

2.3.1. DECOUPLED VELOCITY ENGINE

The fully coupled method is a commonly used method to solve coupled problems implicitly. In this method, equations for all the subproblems are solved simultaneously. We assemble our global Jacobian matrix based on the variables defined on each node and connection

$$\begin{cases} R_N(\omega^{n+1}, \boldsymbol{v}_t^{n+1}) = 0, & \text{Node equations;} \\ R_C(\omega^{n+1}, \boldsymbol{v}_t^{n+1}) = 0, & \text{Connection equations. 2.18} \end{cases} \tag{2.23}$$

Newton's method is applied to the entire system and the Newton update can be obtained by solving a single linear system:

$$\begin{bmatrix} J_{NN} & J_{NC} \\ J_{CN} & J_{CC} \end{bmatrix} \begin{bmatrix} \delta\omega \\ \delta \boldsymbol{v}_t \end{bmatrix} = - \begin{bmatrix} R_N \\ R_C \end{bmatrix}, \tag{2.24}$$

The coupled system will be solved until reaching the global convergence criteria:

$$\begin{cases} ||R_N(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_n, & \text{Node equations;} \\ ||R_C(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_c, & \text{Connection equations.} \end{cases} \tag{2.25}$$

We subdivide the node equations and connection equations of the reservoir into the node or well parts and define the specific tolerance for each subdomain as follows

$$\begin{cases} ||R_{NR}(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_{nr}, & \text{Node equations for reservoir blocks} \\ ||R_{NW}(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_{nw}, & \text{Node equations for well blocks} \\ ||R_{CR}(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_{cr}, & \text{Connection equations for reservoir interfaces} \\ ||R_{CW}(\omega^{n+1}, \boldsymbol{u}_t^{n+1})|| < \epsilon_{cw}, & \text{Connection equations for well interfaces} \end{cases} \tag{2.26}$$

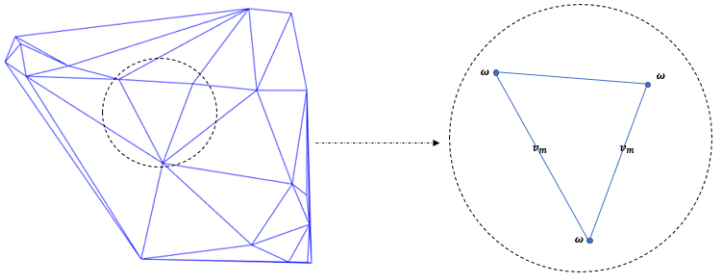


Figure 2.3: Location of the unknowns on general unstructured domain in decoupled velocity framework

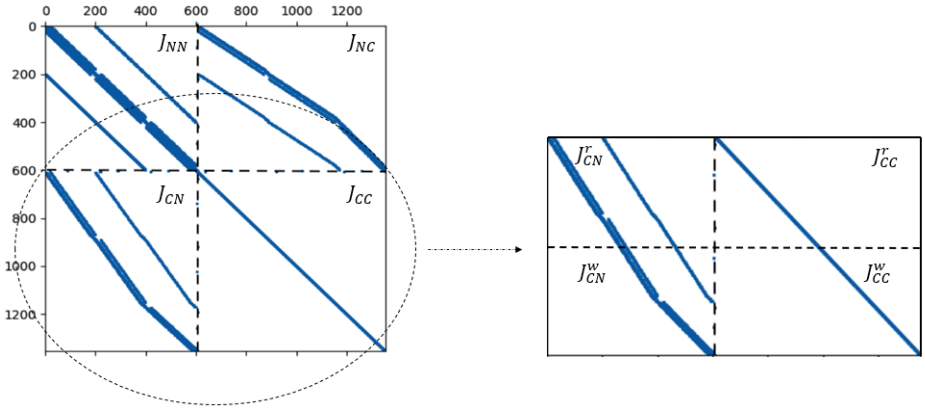


Figure 2.4: Decoupled velocity Jacobian matrix for 3-D reservoir, zooming into momentum equation part

Figure 2.3 shows the grid domain and the location of the primary unknowns.

The global Jacobian matrix for a decoupled velocity engine is shown in 2.4. The global Jacobian matrix has a multilevel matrix structure. It can be divided into two parts at the first level. The upper row is related to the derivatives of the node variables, while the lower row is related to the derivatives of the connection variables. Each part in the second level can be further subdivided into the reservoir part and the well parts. Figure 2.4 shows how the matrix can be further subdivided into connection equations with respect to the reservoir and well parts by zooming in on the momentum parts.

2.3.2. CONVENTIONAL ENGINE

In this section, we describe different types of linearization using the general algebraic form of a governing equation.

To solve nonlinear equations 2.1, we need to linearize them. The conventional approach in reservoir simulation is based on the application of the Newton-Raphson method. In each iteration of this method, we need to solve a linear system of equations of the following form:

$$\mathbf{J}(\omega^k)(\omega^{k+1} - \omega^k) = -\mathbf{r}(\omega^{\mathbf{k}}), \tag{2.27}$$

where J is the Jacobian defined at nonlinear iteration step k. Note that in a conventional engine, velocity is not an additional degree of freedom and all the unknowns ω are defined at the cell center.

2.3.3. MATRIX STORAGE IN DARTS

COMPRESSED SPARSE ROW (CSR)

DARTS offers support for different matrix formats, including Block Compressed Sparse Row BCSR and COMPRESSED SPARSE ROW (CSR). To gain a comprehension of the BCSR format, let's first delve into the CSR format. The CSR format is widely utilized due to its ability to represent sparse matrices of various types [53]. In decoupled velocity information, due to the fact that unknowns are defined are not all define in different zone, we cannot exploit from the block structure of the matix so it makes us to store it in CSR format. CSR is extensively employed in a wide range of simulation applications. Numerous algorithms have been devised specifically for the CSR format, showcasing its significant benefits. In decoupled velocity formulation, because the unknowns are distributed across various parts of the domain (with velocity information at the interface and the remaining components at the nodes), we are unable to take advantage of the matrix's block structure. As a result, we are compelled to store it in the Compressed Sparse Row (CSR) format.

Let's commence by providing an overview of the CSR format. To exemplify the CSR format, a 5x5 sparse matrix with 11 non-zero elements (referenced as 2.28) is utilized. In the CSR format, three arrays are employed, as presented in Table 2.2 and 2.3. These arrays are defined as follows for a matrix with dimensions of nrow x ncol and nnz non-zero elements, considering a 0-based indexing and labeling scheme.

$$\begin{bmatrix} 5 & 8 & 0 & 0 & 0 \\ 5 & 0 & 6 & 0 & 0 \\ 0 & 0 & 12 & 22 & 9 \\ 0 & 0 & 11 & 0.1 & 0 \\ 99 & 0 & 0 & 0 & 0.5 \end{bmatrix}, \tag{2.28}$$

Label	0	1	2	3	4	5
row_ptr	0	2	4	7	9	11

Table 2.2: Row pointer array of the CRS format

- The "value array" contains all the non-zero elements in the matrix, arranged row by row. Its length is always equal to the number of non-zero elements (nnz).
- The row pointer (row ptr) is a 0-based index array that marks the starting position of the first non-zero value in each row. It has a length of nrow + 1, starting with 0 and ending with nnz.

Label	0	1	2	3	4	5	6	7	8	9	10
Col_idx	0	1	0	1	2	3	4	2	3	0	4
value	5	8	5	6	12	22	9	11	0.1	99	0,5

Table 2.3: Column index and value arrays of the CRS format

- The column index (col ind) is a 0-based index array of the same length as the "value array." It stores the column index for each non-zero value.

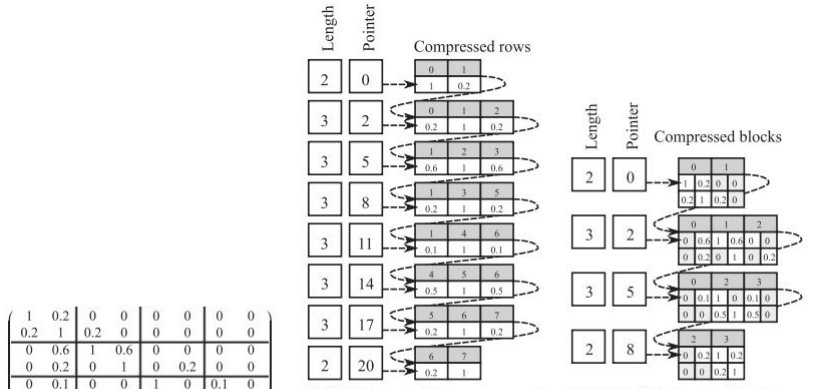
To locate a non-zero element, the "row pointer array" and the "column index array" are utilized. The "value array" stores the non-zero element found at that specific location.

Constructing a CSR matrix involves two steps. Firstly, the pointer arrays for column indices (column-idx) and row pointers (row-ptr) are created by scanning the equations and identifying the positions of the non-zero elements. Secondly, the value array of the CSR matrix is populated. The elements appear in different orders, requiring complex indexing operations to insert them. These two steps are collectively referred to as the Jacobian matrix assembly process.

BLOCK COMPRESSED SPARSE ROW (BCSR)

BCSR format extends the capabilities of the CSR format. In this scheme, the input matrix of size M x M is divided into blocks of P x Q elements. Each non-empty block is stored similarly to the CSR method, as depicted in Fig. 2.5a and 2.5c. Each block row consists of several non-empty blocks, and each block contains a certain number of non-zero elements. It's important to note that zero elements within a block are explicitly stored. In the BCSR format, the column index is stored for each block, while for each block row, its length and the pointer to the first block in that row are indicated, as illustrated in Fig. 2.5c. Storing the values of a block consecutively in memory allows for reduced cache misses [54] when utilized on CPUs, and it enhances the efficiency of memory transactions on GPUs. The block-compressed row matrix format, known as BCSR, offers computational efficiency. The positioning of pointers in the pointer matrix is determined by cell-based and connection-based numbering. Most algorithms that can utilize the standard CSR matrix format can be adapted for the block CSR matrix format with minimal modifications. The key requirement is to replace arithmetic operations with small matrix operations [55]. In conventional DARTS engine since all the unknowns are written at the center of each node, the jacobian matrix have block structures. Therefore, we can exploit from its block entities and store the matrix in BSCR format.

Due to the structure and sparsity of the matrix, we construct our Jacobian matrix in the CSR format within the decoupled velocity formulation. In the case of the conventional engine, (BCSR) format is employed due to the block structure of the Jacobian matrix.



(a) Sparse matrix divided in blocks of size 2×2.

(b) CSR. Rows in (a) are compressed and each non-zero element is stored with its column index (gray cells). For each row, its length and pointer to the first element are stored.

(c) BCSR. 'Block rows' in (a) are compressed, and each *non-empty* block is stored along with its column index (gray cells). For each block row, its length and pointer to the first block are stored.

Figure 2.5: Compressed Sparse Row (CSR) and Block Compressed Sparse Row (BCSR) storage schemes [56]

3

COUPLED RESERVOIR AND WELLBORE

Summary

Well, borehole operations play an important role in all these applications. In order to operate wells intelligently, there must be a robust simulation technology that captures physics and the expected production scenario. In this study, we design a numerical framework for predictive simulation and monitoring of injection and production wells based on the general multi-segment well model. In our simulation model, wells are segmented into connected control volumes similar to the finite-volume discretization of the reservoir. Total velocity serves as an additional nonlinear unknown and it is constrained by the momentum equation. Moreover, transforming nonlinear governing equations for both reservoir and well into linearized equations benefits from operator-based linearization (OBL) techniques and reduce further the computational cost of simulation. This framework has been tested for several complex physical kernels including thermal compositional multiphase reactive flow and transport which covers a large range of energy transition applications including geothermal energy production and CO_2 sequestration. The proposed model is validated using a comparison with analytic and numerical results.

The material presented in this chapter has been published in Geoenergy Science and Engineering 2023, [57].

3.1. INTRODUCTION

Borehole operations are a key component for the management of any energy-related subsurface activities such as energy storage, geothermal energy production, CO_2 sequestration, oil and gas production, wastewater disposal, and thermal recovery processes. Recently, intelligent wells such as long deviated multi-lateral wells with sophisticated inflow control valves have been used to maximize both the economy of the field and the reliability of the operation. Besides, various wellbore designs are used for the efficient production of heat from the subsurface in advanced geothermal approaches. The design, prediction, and optimization of all the processes of interest in energy production require accurate fully-coupled models for thermal multi-phase flow in both the reservoir and the boreholes.

There are multiple challenges with the adequate modeling of such wells. Firstly, capturing accurately the physics (thermal, multiphase flow, multi-component) in the wellbore is complex. Moreover, chemical interactions between the wellbore and flowing fluids cannot be ignored in energy transition applications. The coupling between the wellbore and the reservoir introduces additional complications. The complexity stems from the fact that the flow through the wellbore does not follow Darcy's law. This means that the momentum equation must be solved considering pressure losses due to friction, acceleration, and gravitational forces acting in the fluid. Moreover, the model should be reliable enough to honor more sophisticated intelligent well topology such as multi-lateral deviated wellbore.

Several coupled reservoir and well models were proposed to simulate the complex physics present in the wellbore. The most common approach for modeling a well in reservoir simulation is the standard well model that considers the well as a point source or sink term in the perforated reservoir block [58]. However, the standard well model is blind to the actual physics of the wellbore. Moreover, it cannot capture the complex well network topology which may include chokes, valves, and various surface facilities. To overcome the above drawback, the multisegmented well (ms-well) model was proposed [59–61]. In the ms-well model, the borehole is discretized into several segments with the fluid velocity, node pressure, and other properties simulated along with the wellbore geometry. The benefits of the general ms-well model are its flexible approximation of the actual geometry of each wellbore and its handling of the complex well topology and controls in the pipeline network. Moreover, introducing velocity as an additional degree of freedom allows us to account for various pressure losses by solving the momentum equation.

Multiple solution strategies for coupled reservoir and well models were proposed in the literature. Livescu *et al.* [62] proposed a semianalytical thermal multiphase wellbore-flow model for general-purpose simulation. Pan and Oldenburg [63] proposed the hybrid implicit (semi-explicit) solution for the transient momentum equation in the geothermal wellbore. In addition, a few fully coupled numerical models with different strategies for thermal wellbore flow have been suggested [59, 61, 64]. All these models operate at various simplifying assumptions for wellbore physics and wellbore coupling with the reservoir. There are different formulations for geothermal wellbore simulation exist. Multiple solution strategies for stand-alone geothermal wellbore simulation that neglect wellbore coupling have been recommended. [65] makes an in-depth review of the

geothermal wellbore simulation. The analytical solution for coaxial wellbore simulation was suggested by [66]. Besides a limitation in the coupling, most existing strategies exclude chemical interactions.

In this study, we are interested in a fully implicit coupling of the complex Thermal-Hydraulic-Chemical flow model in the wellbore and reservoir due to unconditional stability. In the fully implicit approach, we use Newton's method to linearize and solve a set of nonlinear equations. Linearization of discrete mass and energy governing equations of multiphase, multicomponent flow, and transport is a challenging task due to the highly nonlinear coupling and complex thermodynamic phase behavior that needs to resolve multiphase partitioning of different components at each nonlinear iteration to accurately evaluate the fluid/rock properties [12, 13].

The linearization stage for such problems is always a demanding task due to the complexity of Jacobian assembly in the presence of fully coupled physical-chemical interactions. A new approach for the linearization of governing equations called operator-based linearization (OBL), was proposed by [14] following ideas from tie-simplex parametrization [15, 16]. In this approach, the exact physics kernels of the governing partial differential equations were approximated using abstract algebraic operators. Later this technique was extended and implemented in the open-source Delft Advanced Research Terra Simulator [17]. DARTS is a scalable parallel modeling framework that aims to accelerate the simulation performance while capturing multi-physics geo-application processes such as hydrocarbon production[18, 19], geothermal energy extraction [20, 21] and CO_2 sequestration [22–24].

In this work, we develop a new computational framework in DARTS applying the general decoupled velocity formulation and extend OBL to couple well and reservoir model. Well and reservoir are both discretized similarly into nodes and connections following the general unstructured grid framework [connection list approach, e.g., 48] using the finite volume method. Total velocity serves as an additional nonlinear unknown written at each interface (connection) on the total computational domain and bounded by a suitable momentum equation. Similar to the staggered gridding method, this framework adopts a simultaneous approach by coupling the mass and energy balance equations at the center of each cell with the momentum balance equations at the cell interfaces. Moreover, transforming both reservoir and well nonlinear governing equations into an operator form benefits from OBL techniques and reduce further the computational cost related to linearization.

The chapter is organized as follows. First, the governing equations describing thermal, multiphase multi-component flow in the wellbore and the reservoir are presented in detail. Next, we present our decoupled velocity design and OBL solution strategies for solving coupled wellbore and reservoir equations. We first test the accuracy and consistency of the method through a set of benchmark tests. Next, more complex numerical experiments are conducted to take into account a more realistic wellbore coupled with a field model.

3.2. WELL MODELLING

In this section, we describe two types of well model used in reservoir simulation.

3.2.1. STANDARD WELL MODEL

In reservoir simulation, the standard model treats a well as a source, or sink term that is added to the grid blocks penetrated by the well [67]. At best, the well is treated as a boundary condition in a manner that ignores the details of fluid flow in the wellbore itself. The standard well (StdWell) model has been widely used in reservoir simulation. The standard model had been extended to handle a single lateral well, which can be vertical, tilted, or horizontal. In the StdWell model, density variations along the wellbore length are accounted for approximately. Some extended standard well models can deal with friction using simplified treatments of frictional losses [68] . We are interested in efficient and accurate representation of the flow in complex wells; however, a detailed understanding of the standard well model is a necessary prerequisite for such models.

The standard well model represents the most basic approach, employing numerous assumptions and approximations to simplify the depiction of physics within the wellbore. Based on the preceding discussion, it is important to acknowledge the following limitations associated with the standard well model:

- The standard well model solely accounts for hydrostatic pressure changes within the wellbore.
- It approximates the treatment of fluid density by employing a one-iteration lag, which is not an accurate representation.
- Transient effects in the wellbore and the consideration of phase holdup are entirely neglected in the standard well model.

3.2.2. GENERAL MULTISEGMENTED WELLBORE MODEL

We developed the general Ms-well model as a solution to the limitations encountered in the standard well model. Ms-well is discretized into nodes and connections in a similar fashion to a reservoir using a finite-volume scheme. Any well segment can be connected with an arbitrary number of reservoir control volumes, representing well perforations. We write the control equation either as a pressure control also known as BHP control for the ghost cell (Fig. 3.1) or we write it as rate control and solve it for the first connection as follows:

$$\begin{cases} R_1^{seg} = \omega_1^{seg} - \omega_{target}, & \text{BHP control} \\ R_1^{conn} = \frac{v_j^{s,j}}{\rho_j^{sc}} . (\sum_c \sum_p \rho_p Q_p x_{x,p}) - Q_j^{target}, & \text{Rate control} \end{cases} \tag{3.1}$$

As shown in figure 3.1, each well segment can have zero, one, or multiple perforations. Each segment of the ms-well wellbore is a separate object that can have different geometrical properties from other segments. Currently, in DARTS, two types of tube and annulus segments are available as shown in Fig. 3.2.

This model offers a range of advanced features, including:

- The model supports general branching, enabling the connection of any well node to multiple other nodes. This feature offers great flexibility, allowing for the definition of facility models with intricate geometries. As a result, a more accurate approximation of the physical reality can be achieved.

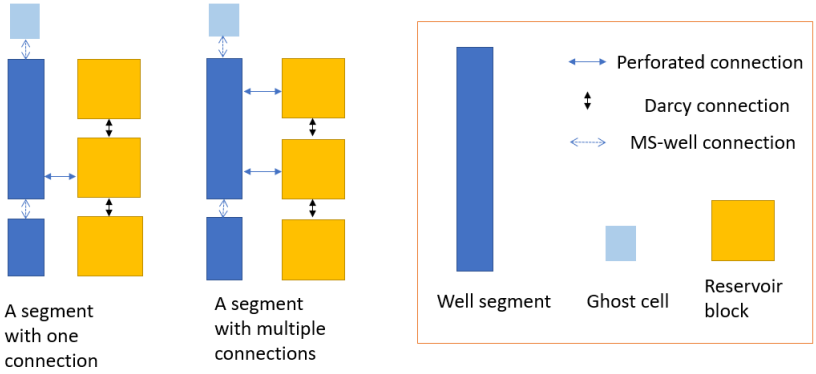
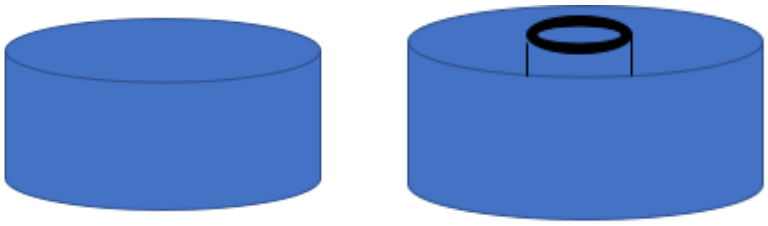


Figure 3.1: ms-well-reservoir network schematic



(a) Tube pipe (b) Annulus pipe

Figure 3.2: Well segment type

- The model incorporates loops with arbitrary flow directions, wherein the flow direction is dynamically determined during the Newton iteration process. The actual flow direction is indicated by the sign of the mixture flow rate assigned to each connection. If the flow rate is positive, it aligns with the initially assumed direction, while a negative flow rate indicates that the actual direction is opposite to the assumed direction.

- The model includes special segments with diverse functionalities, such as separators and valves. In a general multi-segment (MS) well, any segment has the potential to be designated as a special segment, allowing for distinct property calculations compared to ordinary segments. The accumulation terms within the mass conservation equations and the local constraints can be customized to suit the specific functionality of each segment.

3.3. NUMERICAL RESULTS

We test the proposed implementation by comparing it with analytical solutions or numerical results using conventional simulation techniques.

3.3.1. VERIFICATION OF HEAT LOSS MODEL

In this test, we validate our numerical model with the analytical heat loss model in which the overall resistance of the well is calculated with an analogy to the resistance circuits as follows [69]

$$R_{total} = \frac{1}{2\pi L}\Big(\frac{1}{h_f r_i} + \frac{1}{\lambda_p} ln(\frac{r_o}{r_i}) + \frac{1}{h_p r_o} + \frac{1}{\lambda_{ins}} ln(\frac{r_{ins}}{r_o}) + \frac{1}{h_{rc} r_{ins}}\Big). \tag{3.2}$$

$$q = \frac{(\Delta T)_{total}}{R_{total}}, \tag{3.3}$$

Here h_f is the heat transfer coefficient of heat transfer between the fluid inside the pipe and the wall, h_{pi} is the coefficient of heat transfer across any deposits of scale and dirt at the inside wall of the pipe and insulation, r_i is the inner radius of the pipe, r_0 is the outer radius of the pipe and essentially the inner radius of the insulation, r_{ins} is the external radius of the insulation, and λ_p and λ_{ins} are the thermal conductivities of the pipe and insulation, q is the overall heat loss.

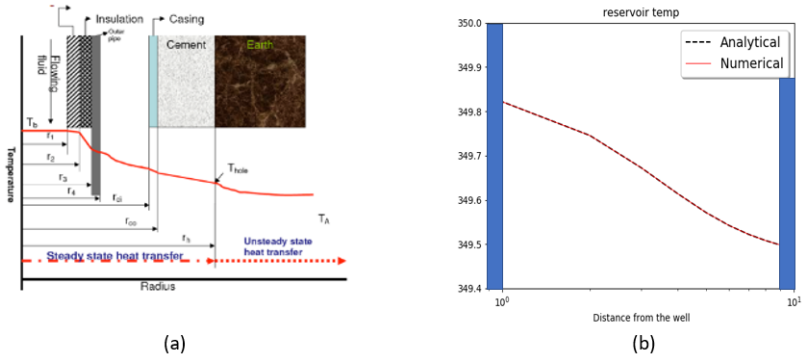


Figure 3.3: (a) wellbore-reservoir heat losses schematic and (b) comparison of analytical model[70] vs numerical solutions

In figure. 3.3 we mimic the wellbore connection to the reservoir with different resistance materials. This test case considers a 1000m in length 1D homogeneous reservoir with constant permeability and porosity of $K = 10$ mD and $\phi = 0.25$ and two ms-well under bhp control in the left and right parts of the domain. We ran the simulation for 1000 days. In figure 3.3.b we compare the numerical solution vs analytical solution of the temperature profile due to the heat losses. It can be seen that both solutions match closely.

3.3.2. SIMPLE 3D RESERVOIR WITH MS-WELL

In this test case, the reservoir dimension is $3 \times 10 \times 10$ [taking from SPE1 benchmark, see 71] with lateral permeabilities of $K_x = 100$, and K_y 500 mD, while the vertical permeability was set as $Kz = K_x/100$. We inject water in pressure 405 bar and produce oil. the reservoir is initially in 400 pressure bar and we produce at 395 bar for the producing

well. Two vertical multi-segmented wells with three segments each are placed at the opposite corners of the model. Each segment is connected to the corresponding layer with different well indices 10, 20, and 30.

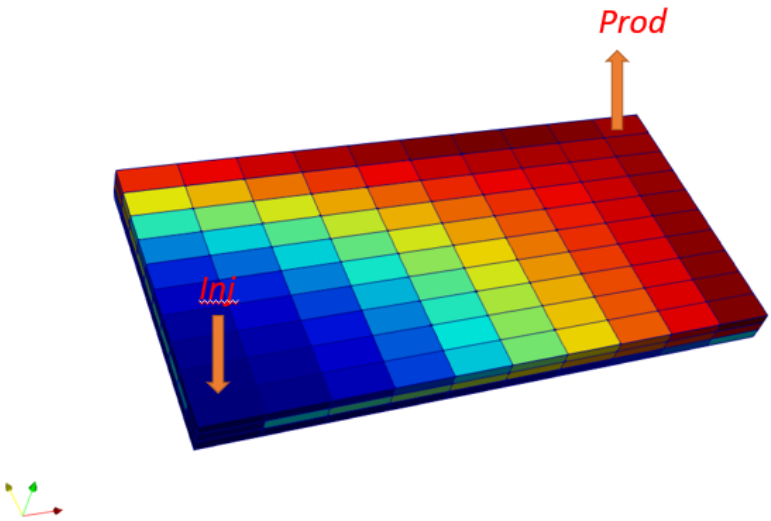


Figure 3.4: Coupled Reservoir-ms-well model

Initially, the reservoir is saturated with oil at 400 bar. We inject water in the injection ms-well at 405 pressure and produce at 395 bar at the production ms-well. We run the simulation for 4000 days. The comparison between DARTS and The Automatic Differentiation General Purpose Research Simulator (AD-GPRS) with the ms-well model is shown in figure 3.5. From this graph, we can observe that there is a perfect match between both DARTS and ADGPRS for this model. You can see that the breakthrough happens quite quickly in the model following the most permeable layer and gradually increasing while oil production is following a backward trend. Due to the changing mobility, the injection rate is growing non-monotonically. Notice that the production rate is shown in negative volume when the injection rate is shown in positive.

Figure 3.6 compares the snapshot of the overall molar fraction of water profile at time 80 days in the production well segments. Here again, we compare solutions produced by ADGPRS and DARTS. Except for a small difference in segment 2, all results are matching quite close. Finally, in figure 3.7 we compare the performance of DARTS vs ADGPRS. In both simulators, a nonlinear tolerance of 10^{-3} is set for mass and momentum conservation. For this test case, the nonlinear solver using Newton's method and the local chop of overall fractions with a maximum allowable update of 0.1 is selected for both ADGPRS and DARTS. In DARTS, the direct SUPERLU solver is used to obtain the linear solution for decoupled velocity formulations. In order to make a fair comparison, a similar setting

was employed in ADGPRS.

We can see that the number of Newton iterations and CPU time are lower in DARTS which can be explained by the utilization of molar nonlinear formulation and the OBL approach.

The nonlinear convergence for simulation problems involving complex physical phenomena, such as gravity, capillarity, and chemical reactions, still remains a challenging problem. Recently, several advanced nonlinear formulations were developed to successfully address these complex problems [72, 73]. However, most of the advanced nonlinear solvers for general-purpose simulation have been developed for natural formulation [74] with explicit correction of saturation. An advanced simulation strategy for the molar formulation is still required to improve the convergence of the nonlinear solutions, especially in the presence of complex physics. The proposed OBL approach for the molar formulation provides a unique tool for an efficient representation of the complex nonlinear physics of the simulation problem.

3.3.3. HETEROGENEOUS LOW-ENTHALPY GEOTHERMAL MODEL

In this test case, we show the capabilities of the decoupled velocity engine for the simulation of a geothermal reservoir with heterogeneous properties. A single-layer model extracted from a synthetic geological setting of the West Netherlands Basin [21] is chosen here. The model has dimensions of 60 × 40 × 1 and a grid size of 30m × 30m × 1m. Two horizontal multi-segmented wells with 10 segments each are placed at the left and right center of the model. Figure 3.8 illustrates the permeability and porosity distribution as well as well locations of the model. The initial pressure of the reservoir is 200 bar, with a temperature of 348 K. We inject water at 308 K with a constant pressure of 250 bars and produce with BHP control at 125 bars.

Figure 3.9(a) shows the pressure and enthalpy distribution after 3650 days and compares it with the conventional DARTS engine without considering velocity as an additional nonlinear unknown following approach suggested in [75]. In this approach, the wellbore is discretized and modeled using Darcy's law. As we can see, a good match is obtained using this approach conventional DARTS engine with an error of less than 0.05%. Figure 3.9(b) compares the well temperature profile along the wellbore comparing the solution using ms-well with the solution using a simplified well model. It is clear that for conventional (near steady-state) geothermal simulation, the pseudo-porous medium approach is accurate enough and can be used instead of the more expensive ms-well approach with a decoupled momentum equation.

In Table 3.1, we compare the performance of decoupled velocity formulation with ms-well vs conventional DARTS formulation using a pseudo-porous medium approach. We can see that the number of nonlinear iterations and CPU time are lower in the case of the conventional approach. The primary reason for this is that the decoupled velocity approach involves an additional degree of freedom with respect to velocity, increasing each iteration's computational cost. Furthermore, the conventional approach typically requires fewer nonlinear iterations because of the simplified momentum equation.

DARTS formulation	nonlinear iteration	CPU time[sec]
Pseudo-porous medium	73	8.44
Decoupled velocity	160	21.4

Table 3.1: Comparison between decoupled velocity and conventional DARTS formulation with pseudo porous media assumptions

3.3.4. REACTIVE TRANSPORT TEST CASE

One of the major issues in geothermal systems is related to reactive transport. Precipitation inside the geothermal well can influence heavily the production results. Moreover, dissolution in the near-wellbore area due to acidification can, on the contrary, increase productivity. All these processes involve the chemical alternation of fluid and solid components near or inside the wellbore. Currently, the implemented ms-well technology is only tested with the dissolution model in DARTS and in this test, we demonstrate this capability.

Here we use the decoupled velocity with an unstructured mesh and demonstrate how it can capture the near-wellbore dissolution in carbonates. For this test case, we analyze the dissolution of calcite which can be written as a simple kinetic equation:

$$CaCO_3 \leftrightarrow Ca^{2+} + CO_3^{2-}, \tag{3.4}$$

We treat this reaction as an equilibrium one. The equilibrium relations are defined by the law of mass action and are given as

$$Q - K_q = \prod_{c=1}^{C} \alpha_c^{c_{cq}} - K_q = 0 \tag{3.5}$$

Here, α_c is the activity of component c, Q_q is the reaction quotient and K_q is the equilibrium reaction quotient or equilibrium solubility limit in the case of dissolution/precipitation of minerals. The rigorous description of reactive transport modeling in DARTS is described in [22]. The injection well is perforating the left boundary and the production well is located on the right boundary of the reservoir.

The model has a dimension of 100 by 100 meters. A constant permeability of 1mD is used with random noise of 5%. Figure 3.10(a) shows both the permeability of the reservoir and the locations of the wells and figure 3.10 (b) shows the unstructured grid domain of the model. The proposed model aims to simulate the phenomenon of unstable wormhole formation that occurs due to minor permeability perturbations near the injector wellbore. The reservoir features an injector well on one side that is perforated across its entire thickness, and a producer well on the other side, spanning the entire thickness of the reservoir. This example model has dimensions of 100 meters by 100 meters. Table 3.2 specifies the simulation parameter for this test case. Figure 3.11 illustrates the solution of solid overall molar fraction $CaCO_3$ at 3 three different times: 0.13 t_d, 0.25 t_d, and at t_d where t_d is the dimensionless time.

It is clear that chemical interactions can be incorporated into the developed ms-well extension of the DARTS framework. In future work, we are going to implement precipita-

tion of minerals in the wellbore and near-well reservoir to accurately capture injectivity decline in the geothermal wells.

Parameter	Value
Initial porosity ϕ_0 [-]	0.26
Initial pressure, p_0 [bar]	95
Initial overall fraction $[H_2O, Ca^{2+} + CO_3^{-2}, CaCO_3]$	[0.09, 0.01, 0.8901]
Injection overall fraction $[H_2O, Ca^{2+} + CO_3^{-2}, CaCO_3]$	[0.8, 0.01, 0.18]
Solubility constant, K_{sp}[-]	55
Simulation time, t [days]	1.8e5
Injection pressure, BHP_{inj} [bar]	135
Production pressure, BHP_{prod} [bar]	65

Table 3.2: Simulation parameters for reactive transport test case

3.3.5. CLOSED-LOOP WELLBORE MODEL

In this example, we will test the developed framework for modeling advanced geothermal setups. Figure 3.12 schematically illustrates a single closed-loop also known as coaxial wellbore setup. In order to, model such wells, the wellbore is discretized into nodes and connected to the reservoir using a finite-volume scheme. Any well segment can be connected with an arbitrary number of reservoir control volumes, representing conductive heat flux. There are currently two types of tube and annulus segments available in DARTS that allow us to model coaxial wellbores. In order to model such well, we need to create a special connection list. Figure 3.13 schematically illustrates the discretized wellbore and corresponding connection list for such wells.

We made two different scenarios for two different flow directions and investigate the effect of the heat conduction (different resistance) between the annulus and the reservoir part. In this test case, the reservoir dimensions are $1 \times 1 \times 100$ with a grid of size $0.30 \times 30 \times 6.96$ m with impermeable layers. The ms-well with 100 segments for annulus and the inner tube is fully (conductively) connected to the 100 reservoir blocks. We ran the test case for 2.4 hours while taking into account various conductive heat transfer coefficients, t_d between the annulus and the reservoir. In this test case, we also assume a perfect insulation between the annulus and the inner tube ($t_i = 0$).

In Figure 3.14 we can see the temperature profile along the annulus and the inner tube while we inject from the annulus and produce from the inner tube. We can see that by increasing the t_d the temperature profile goes slightly upward, meaning lower resistance between the annulus and the reservoir and thus more heat conduction. Figure 3.15 corresponds to the reverse flow direction in which we inject from the inner tube and produce from the annulus. As we can see there is no effect of the t_d on the injection well as there is no contact between the inner tube and the reservoir. On the other hand,

the temperature profile along the annulus is nonlinear and temperature increases by increasing the conductivity between the annulus and the reservoir. The obtained solution qualitatively follows the results reported in [76] for a similar coaxial setup.

3.4. DISCUSSION AND CONCLUSION

We have developped a novel computational framework capable of simulating the complex Thermal-Hydraulic-Chemical (THC) multiphase multi-component fully-coupled flow within both the wellbore and the reservoir. This framework is built upon the operator-based linearization (OBL) method employed in the Delft Advanced Research Terra Simulator (DARTS). The OBL approach involves expressing the governing equations in operator form, greatly simplifying the solution of highly nonlinear equations with intricate physical properties.

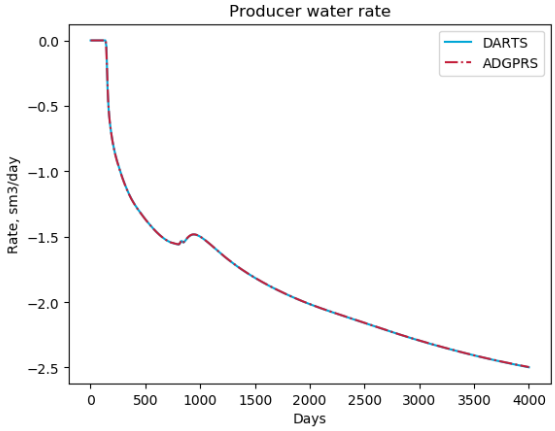
In our proposed framework, we extend the OBL technique to encompass both the governing equations of the reservoir and the wellbore. During the simulation, we utilize multilinear interpolation to efficiently compute the corresponding values and derivatives of operators, reducing the computational load associated with linearization. Our simulation model is constructed on a general unstructured grid framework, where the wellbore is divided into segments following a similar approach to the finite-volume discretization used for the reservoir. Additionally, the total velocity is introduced as an additional nonlinear variable, constrained by the momentum equation, allowing for the formulation of a suitable momentum equation for the wellbore.

Initially, we validate the ms-well model's accuracy by comparing its thermal, two-phase immiscible physics solution to the Automatic Differentiation General Purpose Research Simulator (ADGPRS). The comparison aligns with the ms-well model's performance in both simulation frameworks. Furthermore, we examine DARTS and ADGPRS simulations, highlighting DARTS' advantages in terms of reduced CPU time and nonlinear iterations due to distinct nonlinear formulations and the OBL approach. We also extend testing to more intricate thermal physics, assessing the engine's performance for geothermal scenarios involving a heterogeneous reservoir, comparing it to a pseudo-porous medium approach. In cases without transient effects, the pseudo-porous medium approach provides comparable solutions at a lower computational cost.Additionally, we assess the framework's ability to model calcite dissolution near the wellbore on an unstructured mesh, successfully capturing the dissolution effect. Precise wellbore discretization enables the modeling of advanced well technologies, like co-axial wellbores. We conduct a test case for a single closed-loop wellbore, performing multiple numerical experiments and sensitivity analyses regarding factors affecting heat extraction.

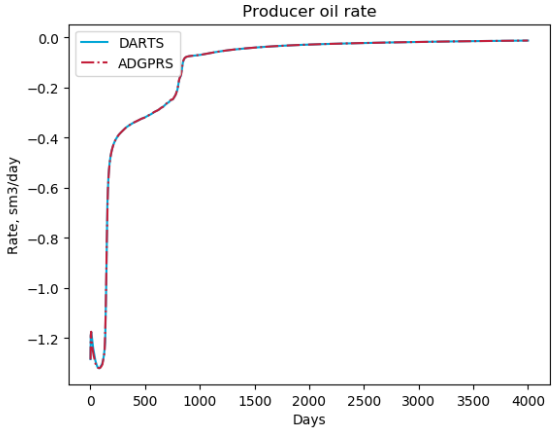
Our study has highlighted potential areas for future research. One direction involves converting the drift-flux model for wellbore multiphase flow into operator form and incorporating complex chemical interactions in the wellbore and reservoir. Another area of interest is exploring iterative linear solvers for decoupled velocity engines on both CPU and GPU platforms. The nonlinear nature of coupled wellbore and reservoir simulations can impede convergence, making the development of a nonlinear solver crucial.

Expanding the Trust-region method, integrated into the operator-based linearization framework, to encompass coupled wellbore and reservoir simulations holds promise. Moreover, the decoupled velocity formulation separates the momentum equation, mass,

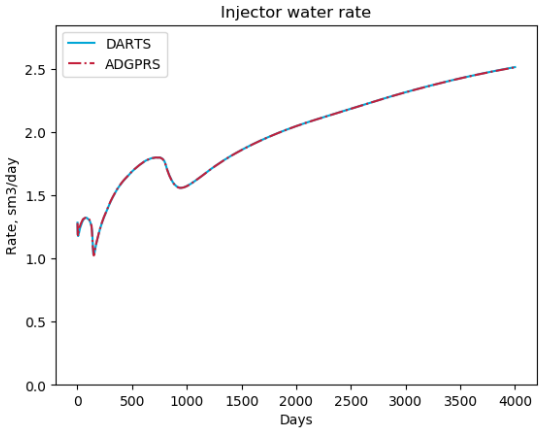
and energy, allowing for a local nonlinear solver that can address convergence issues. We also aim to extend the model to encompass advanced well-network topologies, considering surface capabilities, chokes, and valves.



(a)



(b)



(c)

Figure 3.5: Benchmark testcase with ADGPRS for SPE1 reservoir

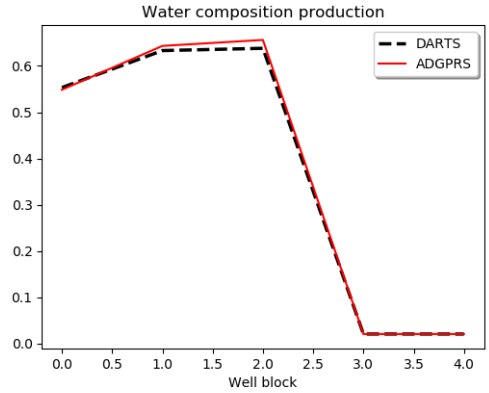


Figure 3.6: Water overall molar fraction inside the wellbore after 80 days

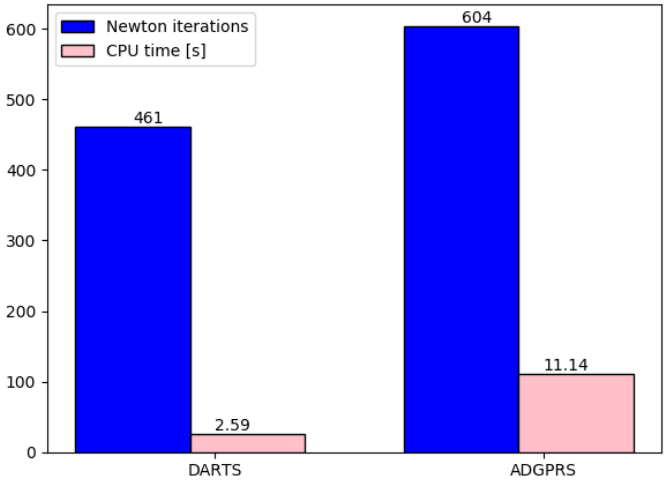


Figure 3.7: Performance comparison between DARTS and AGPRS

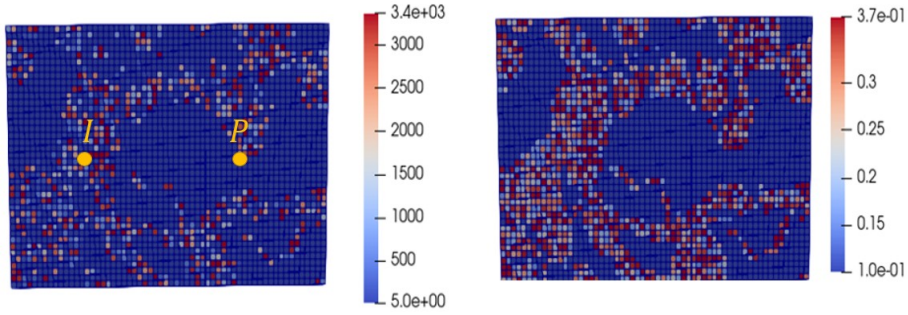
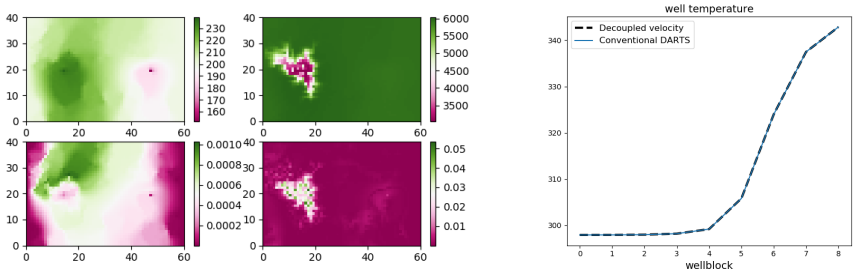
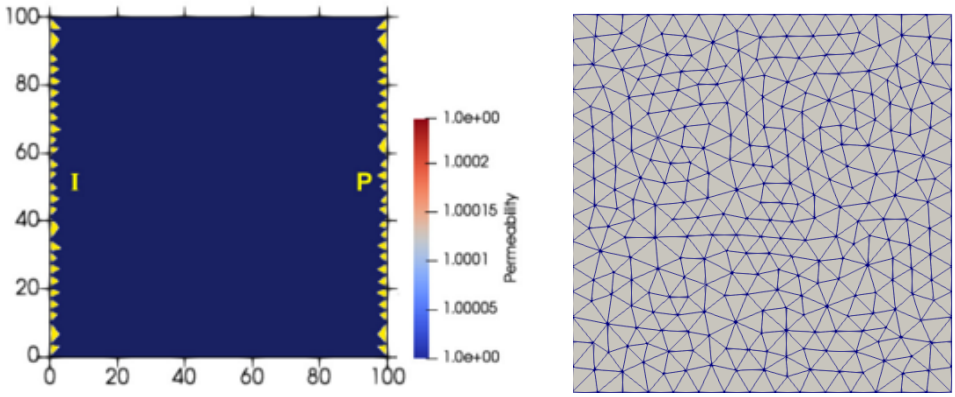


Figure 3.8: permeability (right figure) and porosity distribution with well locations (left figure)



(a) (b)

Figure 3.9: (a) Comparison 2-D heterogeneous pressure and low enthalpy solution obtained with the decoupled velocity and the pseudo-porous medium approach (b) Temperature profile along the wellbore for both approaches.



(a) (b)

Figure 3.10: (a) Reservoir permeability map with specified well locations and (b) discretization mesh

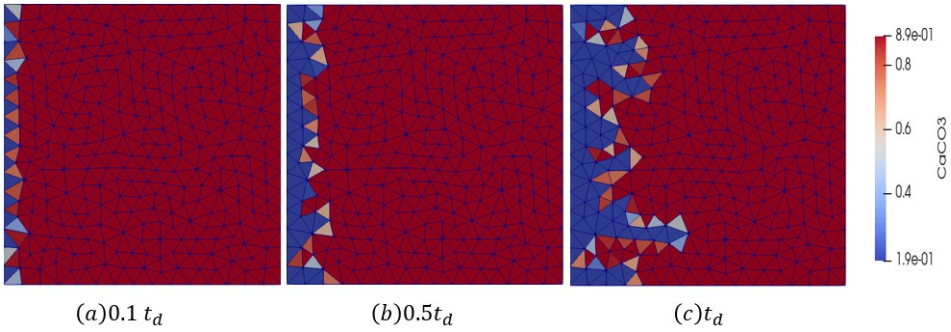


Figure 3.11: Calcium Carbonate dissolution of the wormhole model at three different time snapshots

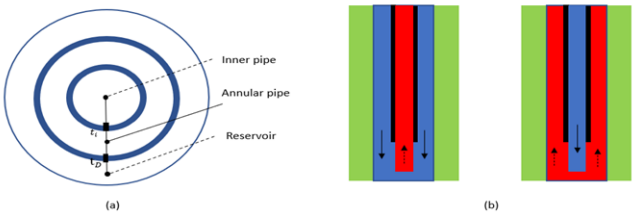


Figure 3.12: (a) Cross section of a pipe-in-pipe single closed-loop wellbore (b) schematic diagram of a single-well closed loop for different flow direction

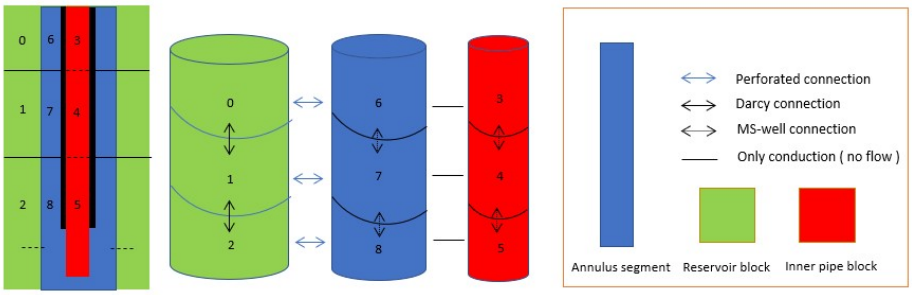


Figure 3.13: Connection list of the coaxial wellbore network

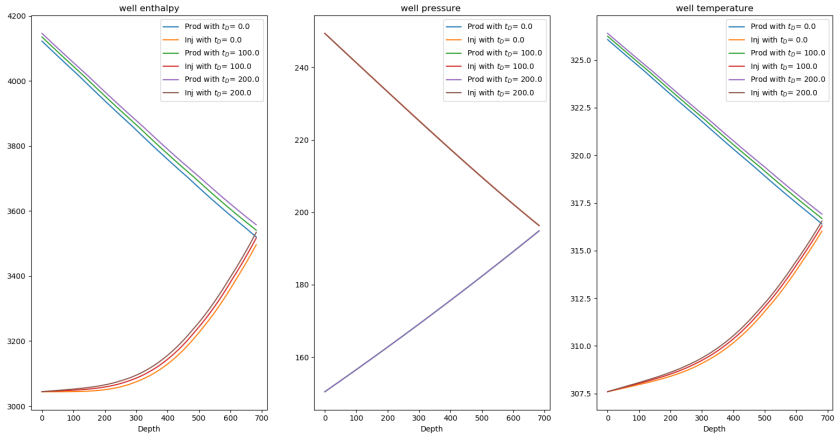


Figure 3.14: Comparison of water temperature profile under different thermal conductivity between annulus and reservoir while injecting from annulus pipe

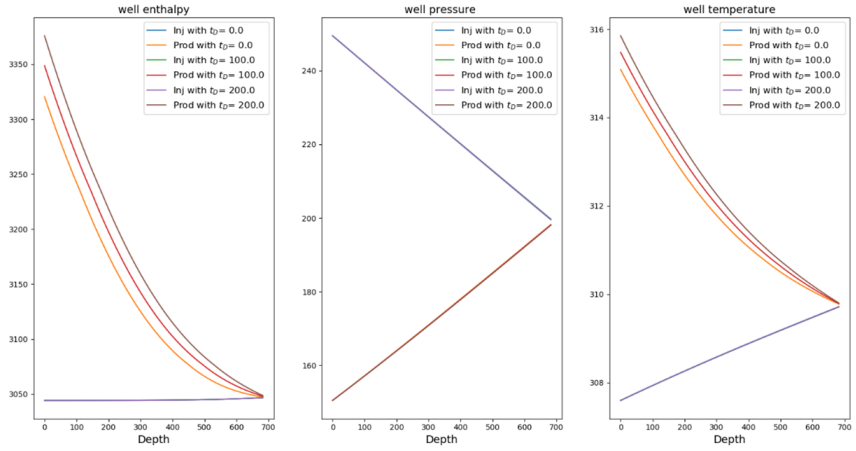


Figure 3.15: Comparison of water temperature profile under different thermal conductivity between annulus and reservoir while injecting from inner pipe

4

NONLINEAR SOLVER BASED ON TRUST REGION APPROXIMATION

Summary

Reservoir simulation of CO_2 utilization and storage (CCUS) in subsurface reservoirs with complex heterogeneous structures requires a model that captures multiphase compositional flow and transport. Accurate simulation of these processes necessitates the use of stable numerical methods that are based on an implicit treatment of the flux term in the conservation equation. Due to the complicated thermodynamic phase behavior, including the appearance and disappearance of multiple phases, the discrete approximation of the governing equations is highly nonlinear. Consequently, robust and efficient techniques are needed to solve the resulting nonlinear system of algebraic equations. In this study, we present a powerful nonlinear solver based on a generalization of the trust-region technique for compositional multiphase flows. The approach is designed to embed a newly introduced Operator-Based Linearization technique and is grounded on the analysis of multi-dimensional tables related to parameterized convection operators. We split the parameter space of the nonlinear problem into a set of trust regions where the convection operators preserve the second-order behavior (i.e., they remain positive or negative definite). We approximate these trust regions in the solution process by detecting the boundary of convex regions via analysis of the directional derivative. We tested the performance of the proposed nonlinear solver for various scenarios.

The material presented in this chapter has been published in Geoenergy Science and Engineering 2023, [24].

4.1. INTRODUCTION

Carbon emissions reduction has become a high priority as the world strives to mitigate global warming. Carbon dioxide capture, utilization, and storage (CCUS) is one of the promising technologies to minimize the amount of greenhouse gases entering the atmosphere. CCUS includes the capture of carbon dioxide and its associated compounds from producing sources, compression, transportation, and use of the captured CO_2 for operations such as permanent storage in deep underground geological formations and increased hydrocarbon recovery in existing oil fields. A model that captures multiphase compositional flow and transport is required to simulate CO_2 use and storage (CCUS) in subsurface reservoirs with complicated heterogeneous structures.

Compositional simulation is based on the solution of the discretized governing equations describing the mass, energy and momentum transfer in the reservoir. Explicit schemes have severe timestep size restrictions and are impractical for large-scale detailed reservoir models with Courant-Friedrichs-Lewy (CFL) numbers that vary by several orders of magnitude across the domain. As a result, in practice, the fully-implicit method (FIM) is preferred.

After the discretization of the governing Partial Differential Equations is complete, a nonlinear system needs to be linearized. The most frequently used sets of variables for linearization are based on natural [77] and molar formulations [51, 78] which include phase-dependent or mass-dependent variables respectively. Typically, linearization is done using a version of the Newton-based method, which demands the assembly of the Jacobian and the residual for the combined system of equations. A previous timestep solution is used as an initial guess for the nonlinear solver. Due to the nonlinear nature of the equations and dependency on the initial guess, Newton's method is not guaranteed to converge for larger timesteps [79]. Once the solution of the linearized system is obtained, the nonlinear unknowns are updated and nonlinear iterations are repeated until convergence is achieved.

Heuristic strategies are utilized to select timesteps in reservoir simulation practice [10]. The use of such heuristics often yields to timesteps that are either too conservative (i.e., small) or too large which in turn leads to wasted nonlinear iterations [25]. The limitation of timestep selection can be overcome by applying an advanced nonlinear solution strategy.

There are several nonlinear solvers described in the literature for the compositional formulation. One of the promising ideas is the continuation method proposed by [25] that introduces a continuous parameter changing between 0 to 1 through the timestep. This approach controls the residual through continuous integration along the nonlinear trajectory in parameter space. Recently, [26] introduced a Dissipation-Based Nonlinear Solver for compositional transport. By using numerical dissipation, the approach creates a homotopy of the discrete governing equations. A continuation parameter is included to limit dissipation and ensure that the converged solution's accuracy is not harmed.

Another approach is the flux-based trust region (TR) method for the natural formulation, proposed initially by [27] for two-phase immiscible flow with the S-shape fractional flow curves. Their work demonstrates that unconditional convergence is possible by limiting saturation updates based on the inflection point of the flux function. Later, [28, 29] extended the flux-based trust region for two-phase immiscible flow and transport where

buoyancy, capillary, and viscous forces are present. More recently, [30] extended this work for black oil three-phase physics with a non-smooth flux function.

Even though different trust region nonlinear solvers were developed for the natural formulation, there is a lack of such strategies for the molar formulation. [31] demonstrated a compositional flow simulation employing tie-lines to parametrize the compositional space. A version of trust-region correction has been developed for molar formulation along the key tie-lines [80] but was not robust enough in comparison with techniques proposed for the natural formulation. [32] designed a nonlinear solver that detects phase boundaries. They focused their research on chopping at phase boundaries, ignoring the inflection line within the two-phase zone. They highlighted that detecting the inflection line for compositional problems required a second-order derivative and the Hessian analysis, which is more expensive and challenging to compute.

Recently, a new approach for the linearization of governing equations, called operator-based linearization (OBL), was proposed by [81]. In this approach, the exact physics of the simulation model was approximated using abstract algebraic operators. Later this technique was extended and implemented in the open-source Delft Advanced Research Terra Simulator (DARTS). In the OBL approach, the parameterization is performed depending on the conventional molar unknowns (pressure and overall composition). Using the OBL approach, the nonlinearity of the residual is translated into the operators. Consequently, by analyzing the nonlinearity of the operators, one can understand the major source of nonlinearity in the discrete residual equations. This greatly facilitates the design of a nonlinear solver for this framework.

In this work, we present an advanced nonlinear solver based on a generalization of the trust-region technique for compositional multiphase transport applied for CCUS. First, we investigate the nonlinearity of convective operators written in fractional flow form and detect boundaries of the trust region for the hyperbolic operator by assembling the directional approximation of the Hessian matrix. Next, we design the nonlinear solver in which we track the nonlinear trajectory for binary and ternary kernel in OBL parameter space and approximate these trust regions in the solution process via directional analysis of the derivative. By drawing some trial Newton trajectories on OBL parameter space, we observe that our directional analysis of derivatives predicts the boundaries of these trust regions correctly. Furthermore, it is less computationally expensive than computing the full Hessian matrix. Finally, we test the performance of the new nonlinear solver for several complex examples.

4.2. NONLINEAR ANALYSIS OF CONVECTIVE (FLUX) OPERATOR

For simplicity, we assume in the following derivations that the system is incompressible, which limits the analysis to the convection operators (Flux operator) β_c only as the main source of nonlinearity in the residual equation. 2.9 written in total velocity formulation. As we can see in Fig. 4.1, for a binary compositional kernel, we have two kinks in addition to the inflection point of the fractional curve. These two points correspond to bubble and dew points compositions where the phase transition occurs. Kinks have different properties than inflection points and usually have a negative impact on nonlinear convergence [82]. There is a discontinuity in derivative at the point of kinks and thus there is an abrupt change in concavity and residual.

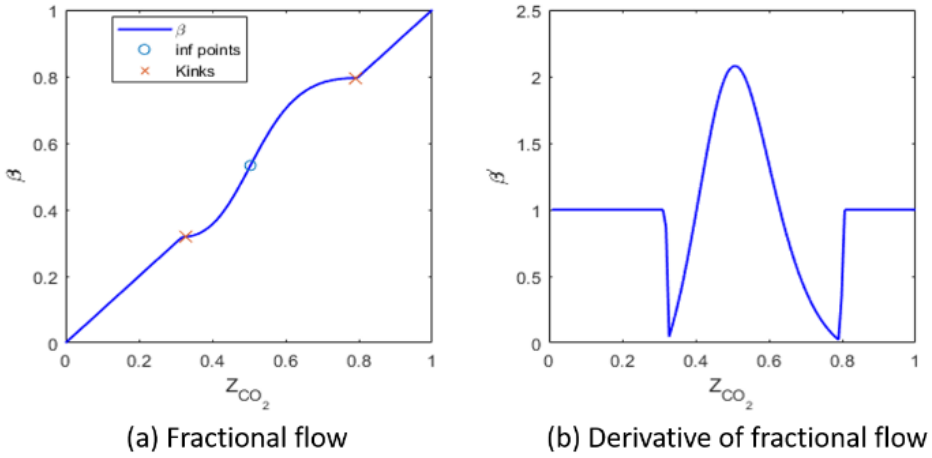


Figure 4.1: Binary incompressible compositional kernel

Kinks and inflection points in parameter space dictate the boundaries of trust regions. For general multicomponent systems, we estimate the inflection point(s) based on the analysis of the Hessian of the convective operator. The Hessian matrix is a way of organizing all the second partial derivative information of a multivariable function. The general Hessian matrix for a convection operator can be written as:

$$\mathbf{H}(\omega) = \mathbf{J}\left(\nabla[\beta_c(\omega)]\right) = \begin{bmatrix} \frac{\partial^2 \beta_1}{\partial \omega_1^2} & \frac{\partial^2 \beta_1}{\partial \omega_1 \partial \omega_2} & \cdots & \frac{\partial^2 \beta_1}{\partial \omega_1 \partial \omega_c} \\ \frac{\partial^2 \beta_1}{\partial \omega_2 \partial \omega_1} & \frac{\partial^2 \beta_1}{\partial \omega_2^2} & \cdots & \frac{\partial^2 \beta_1}{\partial \omega_2 \omega_1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \beta_1}{\partial \omega_c \partial \omega_1} & \frac{\partial^2 \beta_1}{\partial \omega_c \partial \omega_2} & \cdots & \frac{\partial^2 \beta_1}{\partial \omega_c^2} \end{bmatrix}. \tag{4.1}$$

In this work, we focus on the analysis of binary and ternary compositional problems and evaluate the Hessian matrix with respect to hyperbolic variables z_c using the finite difference method.

In our analysis of the ternary system, we are interested in the variation of convective operators with respect to z_1 and z_2. Accordingly, we construct the Hessian matrix for the fixed pressure as follows:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \beta}{\partial z_1^2} & \frac{\partial^2 \beta}{\partial z_1 \partial z_2} \\ \frac{\partial^2 \beta}{\partial z_2 \partial z_1} & \frac{\partial^2 \beta}{\partial z_2^2} \end{bmatrix}. \tag{4.2}$$

For ternary systems, we uniformly discretize the parameter space and compute the Hes-

sian numerically as follows

$$\left(\frac{\partial^2\beta}{\partial z_1^2}\right)_{i,j} = \frac{(\frac{\partial\beta}{\partial z_1})_{i+1,j} - (\frac{\partial\beta}{\partial z_1})_{i,j}}{\Delta z}, \tag{4.3}$$

$$\left(\frac{\partial^2\beta}{\partial z_2^2}\right)_{i,j} = \frac{(\frac{\partial\beta}{\partial z_2})_{i,j+1} - (\frac{\partial\beta}{\partial z_2})_{i,j}}{\Delta z}, \tag{4.4}$$

$$\left(\frac{\partial^2\beta}{\partial z_2\partial z_1}\right)_{i,j} = \left(\frac{\partial^2\beta}{\partial z_1\partial z_2}\right)_{i,j} = \left(\frac{(\frac{\partial\beta}{\partial z_1})_{i,j+1} - (\frac{\partial\beta}{\partial z_1})_{i,j}}{\Delta z}\right), \tag{4.5}$$

where i and j corresponds to the coordinates of the hypercube centers for axes z_1 and z_2 respectively. Next, for each point in the centers at the interface of parameterized hypercubes, we define a quadratic form

$$Q = dz\mathbf{H}dz'. \tag{4.6}$$

After calculating Q for all points in the parameter space, we identify trust regions. If Q changes the sign from positive to negative, it indicates that our operator goes from positive definite to negative definite and changes its convex condition.

Figure 4.2 shows the Hessian diagram for all three convection operators (β_1, β_2, β_3) and the phase diagram corresponding to that ternary kernel. In the phase diagram, the red color corresponds to the two-phase region and the blue color corresponds to the single-phase region. In the Hessian diagram, each color corresponds to different convex conditions of the flux operators. Comparing the Hessian diagram to the phase diagram, it is clear that there is an abrupt change in concavity (kink) on the boundaries between single-phase and two-phase regions. Moreover, there is an inflection line in the two-phase region for each component that segments the two-phase zone into a concave and a convex part.

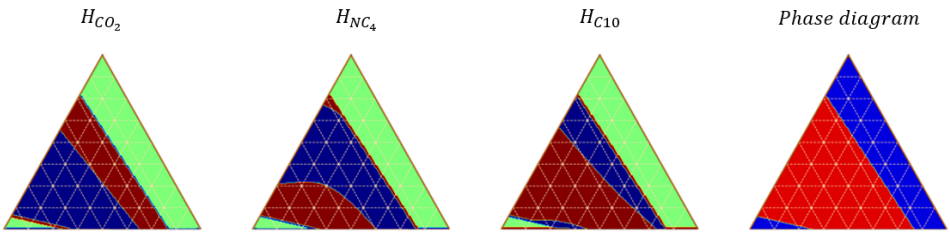


Figure 4.2: Hessian for three convective operators and phase diagram for incompressible ternary kernel

All computations of special features in the phase diagram depend on the OBL resolution. Since all properties including convective operator β_i are defined based on interpolation among supporting points in the OBL mesh, any changes in convexity can be detected based on the numerical approximation of second derivatives. It is clear that these calculations as well as the performance of the nonlinear solver are dependent on the OBL resolution. In Appendix C, we show the sensitivity of inflection point definition

and nonlinear convergence based on OBL resolution. More sensitivity analysis of the nonlinear convergence can be found in [81] and [83]

It is computationally expensive and time-consuming to compute the Hessian at each point in the parameter space. Checking the convexity condition only for the interfaces through which Newton's trajectory passes excludes the need to study the entire parameter space. However, the quadratic form of the operator must be evaluated using equation 4.6, which requires the computation of the full Hessian matrix. The size of the Hessian matrix is $n_c \times n_c$, where n_c is the number of components. As a result, n_c^2 evaluations of first derivatives are required for hessian matrix assembly. To avoid calculating the entire Hessian matrix, in the directional analysis of the second derivative, we only need to evaluate derivatives in the direction of the given interface that the nonlinear solver passes. As an example, given the simple Newton trajectory in figure 4.3 that passes interface L in the direction of Z_1, the second order directional second derivative is calculated as follows:

$$\partial^2_{Z_1}\beta(\omega) = \frac{(\frac{\partial\beta}{\partial z_1})_{i+1} - (\frac{\partial\beta}{\partial z_1})_i}{\Delta z_1}, \qquad \omega = [Z_1, ..., Z_{nc}]. \tag{4.7}$$

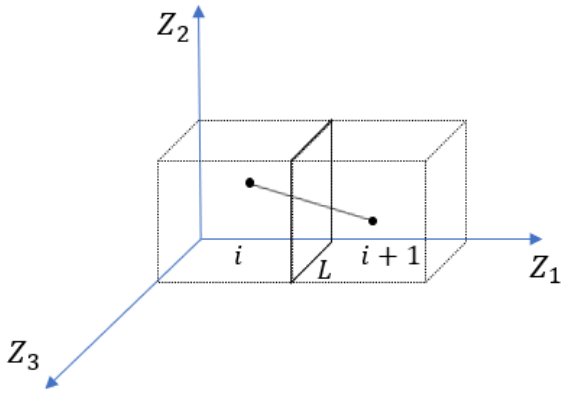


Figure 4.3: Newton trajectory passing interface L between two cells in the parameter space

The advantage of using equation 4.7 is that we only need to evaluate derivatives once to calculate the second-order directional derivative regardless of the dimension of the operator's space n_c.

4.3. NONLINEAR SOLVER FOR OBL FRAMEWORK

Before we present our Trust Region (TR) nonlinear Newton solver in detail, we provide an overview of the state-of-the-art damping strategies for Newton's solver. Newton update can be written at every nonlinear iteration in the general form as follows:

$$\Delta\omega = -\Phi \mathbf{J}^{-1}\mathbf{r}, \tag{4.8}$$

where **J** and **r** are the Jacobian and and the residual. Here Φ is the diagonal matrix:

$$\Phi = \text{diag}(\phi_1, \phi_2, ..., \phi_n), \phi_i \in [0, 1]. \tag{4.9}$$

Global and local nonlinear solvers can be seen as different methods to damp the Newton updates [84] by specifying the diagonal matrix Φ:

- *STD* The standard Newton's solver where the updated matrix is the identity matrix.
- *Global chop* In the global-chop nonlinear strategy, all entries of the diagonal are identical, implying that the Newton direction is simply scaled by a constant factor.
- *Local chop* In the local-chop nonlinear solver, the diagonal scaling entries depend on a cell-by-cell basis to limit the local compositional update.

The benefit of global chop is that the nonlinear update direction is not changing. However, it could be quite restrictive due to a significant number of resulting nonlinear iterations. Local chop on the other hand is more effective due to the local adjustment of the update. However, it may lead to inconsistency in the nonlinear update due to the restrictions of the hyperbolic transport solution. Next, we will present a Trust Region (TR) solver which yields a consistent nonlinear update.

4.3.1. TRUST REGION SOLVER FOR OBL FRAMEWORK

In previous sections, the sources of nonlinear convergence failure, such as kinks and inflection lines encoded in the convective operator, were discussed. All of these features can be discovered by applying the Hessian analysis of the convective operator. With these kinks and inflection lines, the nonlinear space is divided by multiple subregions, which are referred to as trust regions (TR). The key idea is that any nonlinear update (in terms of composition) is not allowed to cross the boundary of any trust region too far. If an intersection is found, the nonlinear update's size is reduced until the updated composition barely crosses the boundary.

The full Hessian evaluation in the trust-region identification is a time-consuming procedure. To detect the trust regions more efficiently, we apply a directional analysis of the second derivative while tracking the nonlinear update and passing each interface in the OBL parametrization. The procedure is as follows:

1. detect OBL interfaces along the nonlinear update trajectory in the parameter space (see Appendix 1 for detailed algorithm detecting OBL interfaces based on tracking of nonlinear trajectory in parameter space),
2. computes directional second derivative for each convection operator at each crossing interface of OBL space with finite difference method,
3. detects inflection point(s) and kinks based on the second derivative information,
4. limits the local nonlinear update by the location inside the trust region.

Next, we illustrate our approach with examples and compare a calculation of special points detected by two types of analysis: reference results based on the construction of the full Hessian matrix for binary and ternary systems and directional analysis based on the second derivative only. In Fig. 4.4, we detect an inflection line position in the parameter space based on the convex condition approximated by the quadratic form of the Hessian matrix and compare it to the inflection point detected during tracking the nonlinear update. In this example, we use a two-phase immiscible physical kernel with viscosity dependent on pressure, which results in a pressure-dependent inflection line. Note that the exact location of the detection points can be lightly shifted from the line position due to the discrete approximation of the directional derivative. Another observation is that the pressure impact can be taken into consideration when detecting the inflection point using the directional derivative.

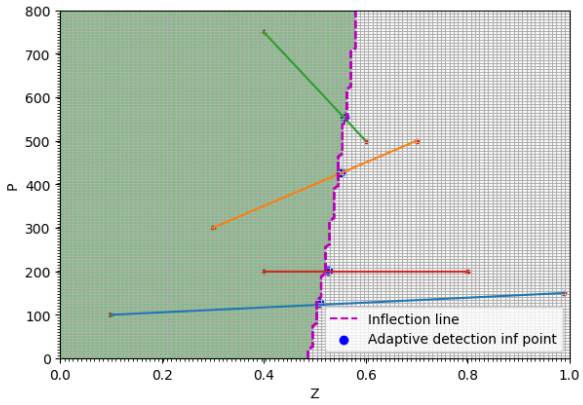


Figure 4.4: Adaptive detection of inflection point versus the rigorous inflection line calculated by the full Hessian assembly for several random trajectories in parameter space

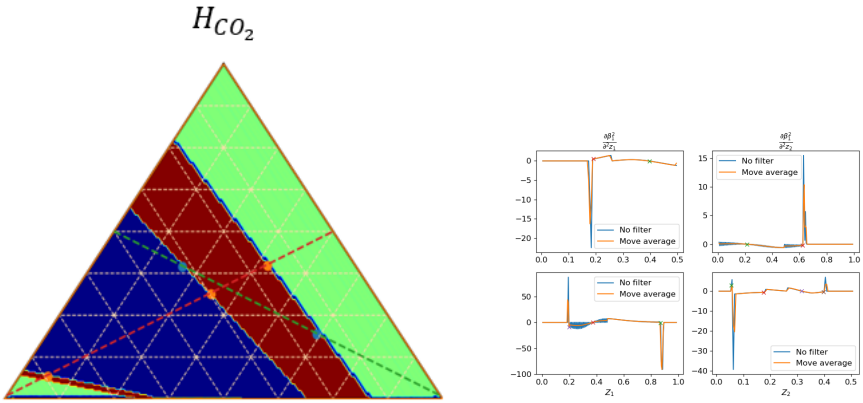
Similarly, for a ternary compositional kernel, we show in Fig. 4.5(a) that special points can be detected correctly using directional derivatives along the nonlinear update. However, some numerical artefacts and noise are usually present in the computation of numerical derivatives in directional analysis. To overcome this problem, we use the moving average algorithm [85] to smooth second-order directional derivatives as shown in Fig. 4.5(b).

4.4. PERFORMANCE OF NONLINEAR SOLVER

In this section, the performance of the proposed nonlinear solver is demonstrated for several simplified and realistic modeling setups.

4.4.1. SINGLE CELL ANALYSIS

We can derive some essential conclusions for simulation problems by investigating the nonlinear behavior of a single-cell compositional transport problem. Here our goal is



(a) Trial trajectories on H_{CO_2}

(b) Directional derivatives of the random trajectories.

Figure 4.5: Full and directional evaluations of second order behavior for ternary compositional kernel

to find the solution z^{n+1} from the initial guess $z^{n+1,0}$ for a given boundary condition on the left and right sides of the cell, defined as $\beta_{c_i L}$ and P_R, respectively. The imposed flux is from left to right. Here we inject pure gas mixture with the composition of (CO_2, C_{10}) = (1, 0) into the reservoir initialized with the composition of (0, 1) with the K-values equivalent to $K = \{2.5, 0.3\}$ and the viscosity ratio of 10 between the oil and gas phase.

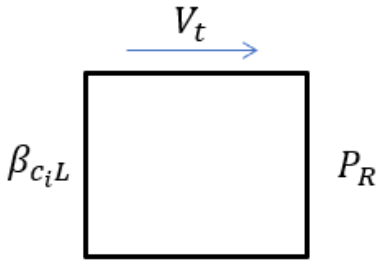
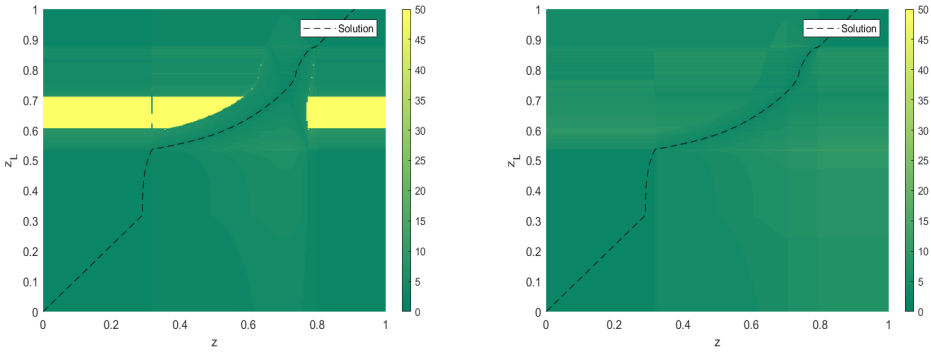


Figure 4.6: Single-cell setup with left and right boundary condition. Flow is from left to right with fixed total velocity V_t and fixed total flux $\beta_{c_i L}$.

We investigate the convergence map of the pure Newton's solver and trust region solver for binary compositional kernels. In this setup, we fixed the right boundary conditions as $z_r = 1$ and study the convergence of the solution using different nonlinear solvers for all possible starting points $(z^{n+1,0}, z_l) \in (0,1) \times (0,1)$. The maximum nonlinear iteration for all these test cases is equal to 50. We perform our analysis for two different dimensionless timesteps expressed as C=$\frac{V_t \mathrm{dt}}{\mathrm{dx}}$, with t in days and x in meters and V_t velocity in meters per seconds.

As seen in Fig. 4.7, once the solution is in the two-phase area, pure Newton's update struggles to converge in a maximum number of iterations. The color indicates the num-

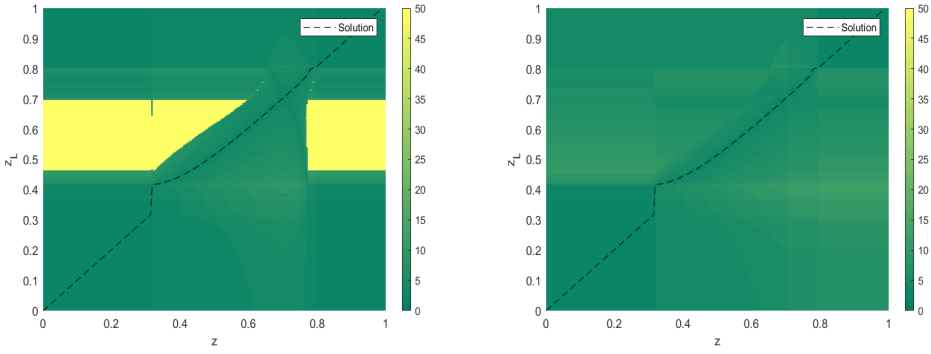


(a) Convergence map for pure newton

(b) Convergence map for TR algorithm

Figure 4.7: Convergence map for compositional binary problem with C= $\frac{V_t \mathrm{dt}}{\mathrm{dx}}$ = 10

ber of Newton iterations. A maximum number of iterations is set at 50. Another point to note is that once the solution is in the single-phase zone (linear fractional flow), Newton's approach guarantees nonlinear convergence. The proposed nonlinear solver, on the other hand, always has a limited number of nonlinear iterations.



(a) Convergence map for pure newton

(b) Convergence map for TR algorithm

Figure 4.8: Convergence map for compositional binary problem with C= $\frac{V_t \mathrm{dt}}{\mathrm{dx}}$ = 100

As we increase the timestep, we can observe that for pure Newton's update, the yellow region (corresponding to the non-convergence zone) increases again in the two-phase region. However, once the solution is in the single-phase region pure Newton is able to find the solution. And again, the proposed Trust Region solver is globally convergent for all initial guesses.

4.4.2. FRONT PROPAGATION IN A SINGLE FRACTURE

In fractured reservoirs, the speed of transport front propagation between matrix and fracture is significantly different due to the large contrast in permeabilities. Here we imitate this process for a one-dimensional reservoir by running the simulation with small timesteps to develop a resolved displacement solution at a particular time. Next, we restart the simulation from this distribution for one control timestep only and account for the number of nonlinear iterations required to converge the solution. We repeat this procedure by gradually increasing the size of the control timestep and detecting the change in the number of nonlinear iterations. Fig. 4.9 illustrates CO_2 front propagation test case. This test case considers a 1000m in length 1D homogeneous reservoir with constant permeability and porosity of $K = 10$ mD and $\phi = 0.3$.

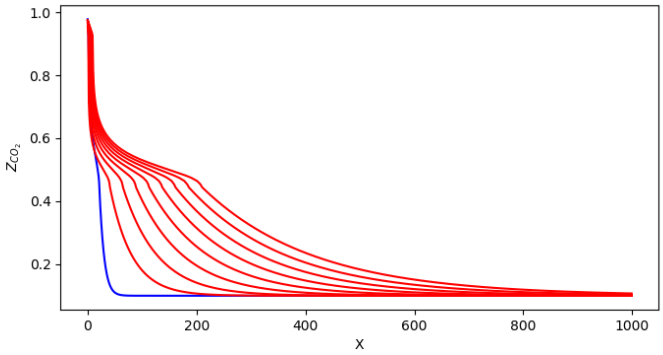


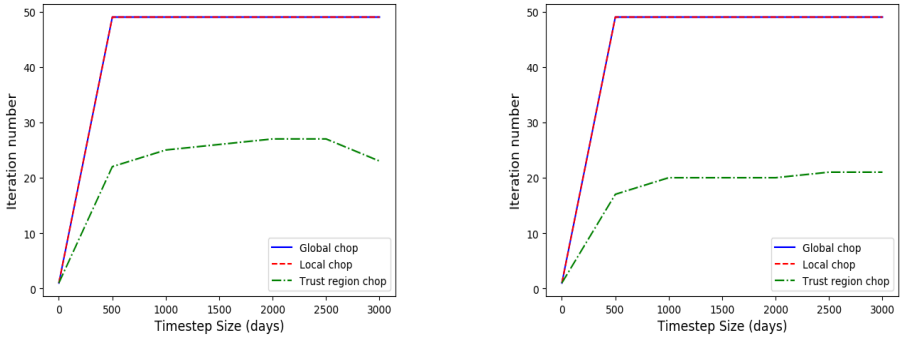
Figure 4.9: CO_2 front propagation for different timesteps. The blue curve represents the resolved displacement solution, and the red curves represent the progressive control timestep solutions.

Next, we use multiple physical kernels of increasing complexity to test the performance of a nonlinear solver based on directional analysis in the suggested numerical setup. We also compare our trust-region nonlinear method against global and local chop with $\Delta z = 0.1$ as the highest tolerable Δz.

For binary systems, we test the performance of nonlinear solvers for the flow of two-phase immiscible fluids and miscible fluids with phase behavior controlled by constant K-values with $K = \{2.5, 0.3\}$. Initially, the 1D domain is fully saturated by the non-wetting phase and we inject a wetting phase at the left boundary. We ran the simulation with a small timestep for 1000 days. Next, we restart by enlarging the timesteps. Figure 4.10 shows the fracture test results comparing different nonlinear solvers for binary kernels. We set the maximum number of nonlinear iterations to 50 for all test cases.

It is clear that the trust region solver performs better (provides fewer nonlinear iterations) for both immiscible and miscible kernels. Here global and local chops only provide a convergence strategy for timesteps corresponding to $\Delta t = 500$ days and fail to converge for larger timesteps.

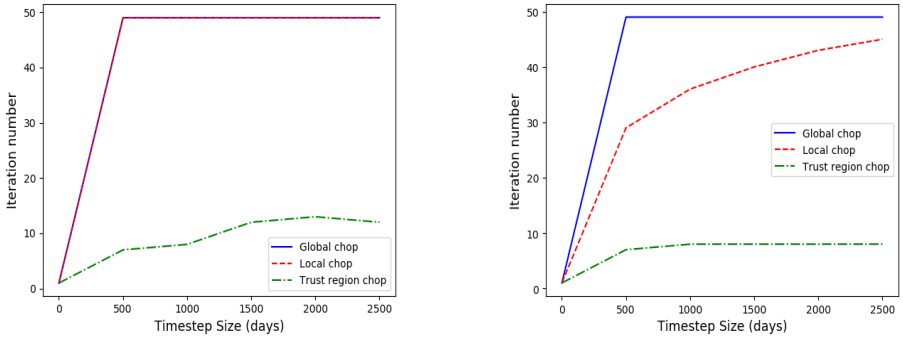
Furthermore, we run front propagation in a single fracture test case for ternary and quaternary kernels. We inject at 135 bar using bottom hole pressure control and initial reservoir pressure of 95 atm. For the ternary kernel, we inject the gas stream of



(a) Immiscible kernel (b) Miscible kernel

Figure 4.10: Comparison of different nonlinear solvers for binary kernel

$\{CO_2, NC_4, C_{10}\}$ with the composition of $z_{inj} = \{0.98, 0.01, 0.01\}$ into the reservoir initialized with $z_{ini} = \{0.1, 0.2, 0.7\}$. Simillarly, in 4-component system we inject gas stream of $\{CO_2, NC_4, C_{10}, C_2\}$ with the composition of $z_{inj} = \{0.98, 0.001, 0.001, 0.01\}$ into the reservoir initialized with $z_{ini} = \{0.1, 0.2, 0.2, 0.5\}$. Figure 4.11 illustrates the performance of different nonlinear solvers for a single fracture test case. In both the ternary and four-component test cases, the TR solver converges with fewer nonlinear iterations.



(a) Ternary kernel (b) 4 components

Figure 4.11: Comparison of different nonlinear solvers for ternary and quaternary kernel

4.4.3. FULL COMPOSITIONAL SIMULATION

In this test case, we run the simulation study for multiple timesteps until reaching the final time. We start the simulation with a small timestep ($\Delta t = 1$ day), if nonlinear iteration converges, we double the timestep for the next timestep and if it does not converge

we cut the timestep to half. The maximum Newton iteration is again $N = 50$. In this test case, we compare four different nonlinear solvers: *STD*, *TR*, *Global* and *local* chop with a damping factor of 0.1.

1D HOMOGENEOUS MODEL

This is the ternary 1D full simulation test case. The domain is discretized into 1000 grid cells with a homogeneous permeability and porosity of $K = 10$ mD and $\phi = 0.3$. Components are $\{CO_2, NC_4, C_{10}\}$. We inject a gas mixture of composition (0.98, 0.01, 0.01) into the (0.1, 0.25, 0.65) under BHP control (constant pressure 405 bar) at the left boundary and produce it under constant pressure 395 bar at the right boundary.

Figure 4.12 illustrates the cumulative number of nonlinear iterations for different final times and compares different nonlinear solver's performances. It is clear that the TR solver has superior performance with respect to other solvers. Figure 4.13 compares the composition front for different nonlinear solvers. As we can see, the STD solver can capture the shock accurately. On the other hand, the TR solver smears out the shock since it is capable of converging for a more aggressive timestep and consequently larger time-truncation error is present in the solution. The smearing effect of time truncation is most noticeable at the shock. Because the other solvers cannot converge for larger time steps, they reduce timesteps, resulting in less truncation error and better composition front evolution. While time truncation error analysis is beyond the scope of this paper, having a robust nonlinear solver allows us to choose timestep from practical considerations rather than the necessity to perform a simulation with smaller timesteps.

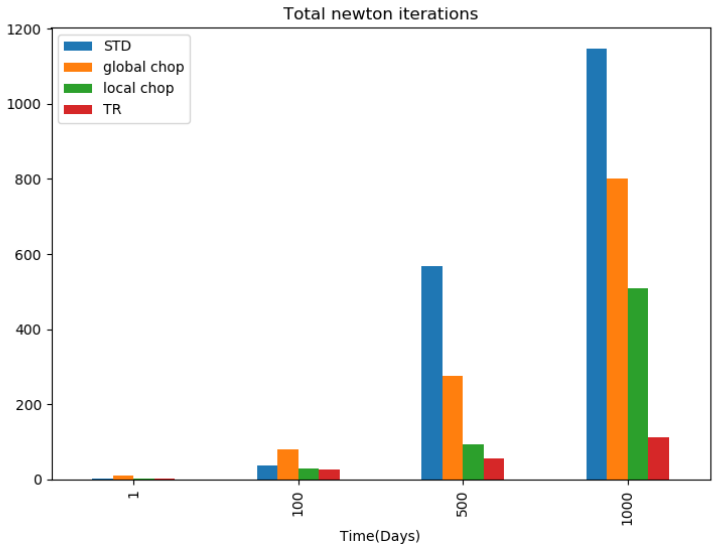


Figure 4.12: Comparison of cumulative Newton iterations for different nonlinear solver for ternary kernel

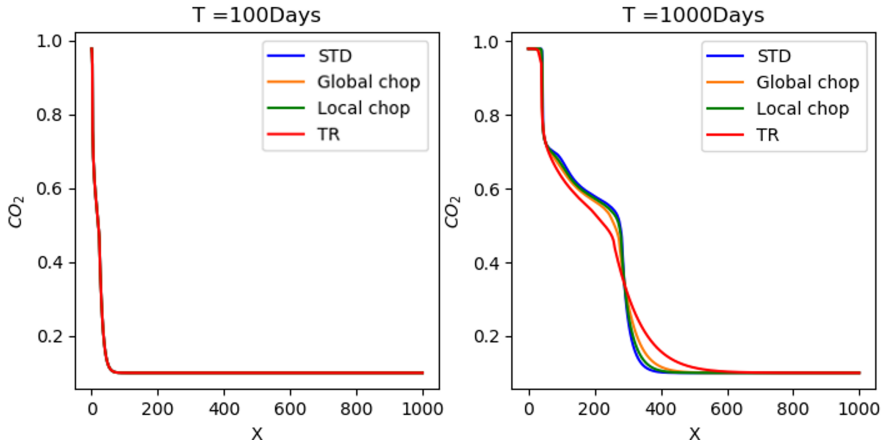


Figure 4.13: Comparison of solution for different nonlinear solvers for ternary kernel

4.4.4. 3D HETEROGENOUS MODEL

In this test case, we compared the performance of nonlinear solvers for a heterogeneous reservoir. We used a channelized Egg model described in [86]. The dimension of the model is $60 \times 60 \times 7 = 25200$ grid cells of which 18553 cells are active, with a grid size of 8m × 8m × 4m. The porosity is constant ($\phi = 0.2$). Figure 4.14 illustrates the permeability and well locations. The injection well is located in the left upper part and the production well is at the right bottom of the domain. The initial pressure of the reservoir is 400 bar. We inject a gas mixture of composition (0.98, 0.01, 0.01) into the (0.1, 0.25, 0.65) at constant pressure 405 bar and produce at 395 bar. We run the simulation for different nonlinear solvers several times.

Figure 4.15 summarizes the nonlinear iterations for different nonlinear solvers. The TR solver takes a lower number of nonlinear iterations to converge compared to the other solvers. In this test case, the STD solver takes the highest number of Newton iterations since it relies on a pure Newton update. Local chop performs slightly better than global chop in this setting but still worse than TR. We take the solution of STD as a reference solution for the time-truncation error analysis of the solution for the single top layer of the model. From the figure 4.16 we can see that the error is mainly localized around the trailing and leading shocks. Similar to the 1D homogeneous model test case, the time truncation error affects the accuracy of the solution since the TR solver can converge for more aggressive timesteps. You can see that the difference in the 2D heterogeneous solution is less pronounced than the one detected in homogeneous 1D.

4.5. NONLINEAR PRECONDITIONING

The proposed trust-region nonlinear solver guarantees to converge for arbitrary timesteps. However, its application may become time prohibitive due to an excessive number of nonlinear iterations for large timesteps. To overcome this issue, we proposed the nonlinear preconditioning strategy described in this section. In the OBL framework, we look

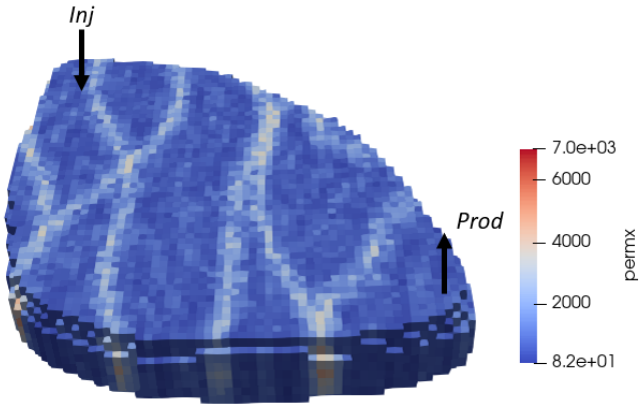


Figure 4.14: Permeability K_x map of egg model

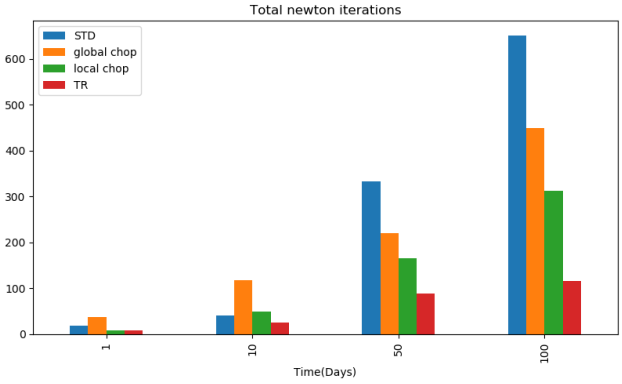


Figure 4.15: Comparison of cumulative Newton iteration for different nonlinear solver for egg-model at different time

into the sequential update on nonlinear iterations where changes in transport unknowns in the downstream block depend on the derivatives of the convective operator at the current block. If composition in the block is defined at the residual values (when convection for the phase is absent), the update in the downstream block is zero, and the transport wave cannot propagate on this iteration. As a result, the nonlinear solver needs as many iterations as many blocks at residual compositions it needs to propagate.

This problem is especially pronounced in fractured reservoirs, where the transport front has to propagate for a large number of grid-blocks once it reaches fracture. Here, we perform the analysis for the propagation of the nonlinear wave in the OBL approach which is a further extension of the analysis performed in [87]. To simplify derivations, we assume that the model is limited by a 1D reservoir with Cauchy boundary conditions on the left and right sides. This reduces the spatial discretization, which yields the following equation in vector form (the length of the vector corresponds to the number of

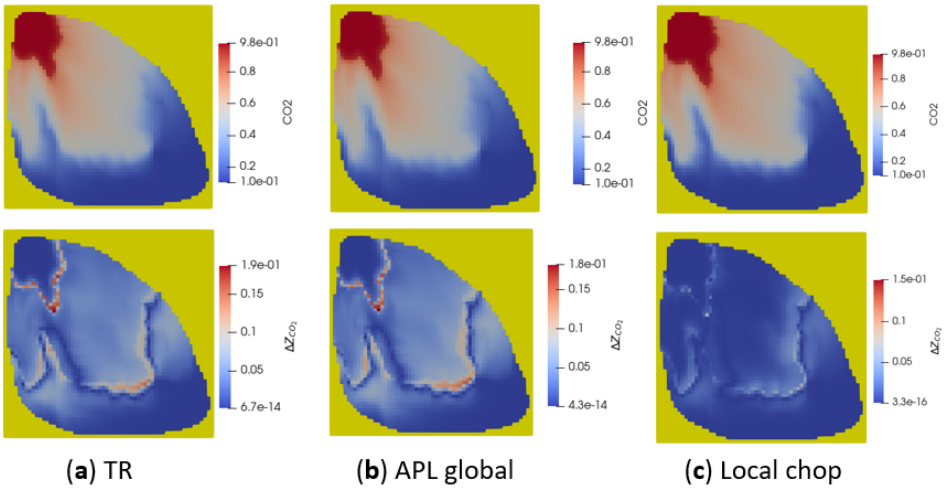


Figure 4.16: Comparison of CO_2 saturation maps at the time of $T = 100$ days (1st row). The second row shows the delta of gas saturation between each model and the STD case

components n_c) for the block i:

$$\mathbf{r}_i(\boldsymbol{\omega}_{i-1},\boldsymbol{\omega}_i,\boldsymbol{\omega}_{i+1},\boldsymbol{\omega}_i^n) = (\boldsymbol{\alpha}(\boldsymbol{\omega}_i) - \boldsymbol{\alpha}(\boldsymbol{\omega}_i^n))a_i - \boldsymbol{\beta}(\boldsymbol{\omega}_i)b_{i^+}(\boldsymbol{\omega}_i,\boldsymbol{\omega}_{i+1}) + \boldsymbol{\beta}(\boldsymbol{\omega}_{i-1})b_{i^-}(\boldsymbol{\omega}_i,\boldsymbol{\omega}_{i-1}), \tag{4.10}$$

where

$$a_i = \phi_0 V_i, \tag{4.11}$$

$$b_{i^+}(\boldsymbol{\omega}_i,\boldsymbol{\omega}_{i+1}) = \Delta t T_{i,i+1}(p_{i+1} - p_i)\Lambda(\boldsymbol{\omega}_i), \tag{4.12}$$

$$b_{i^-}(\boldsymbol{\omega}_i,\boldsymbol{\omega}_{i-1}) = \Delta t T_{i-1,i}(p_i - p_{i-1})\Lambda(\boldsymbol{\omega}_{i-1}). \tag{4.13}$$

For simplicity, we assume a homogeneous reservoir with V, ϕ_0 and T constants. Equation 4.10 can be written for an internal reservoir block as

$$\mathbf{r}_i = (\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_i^n) + \gamma(\boldsymbol{\beta}_i b_{i+} + \boldsymbol{\beta}_{i-1} b_{i^-}). \tag{4.14}$$

Here

$$\boldsymbol{\alpha}_i = \boldsymbol{\alpha}(\boldsymbol{\omega}_i), \quad \boldsymbol{\beta}_i = \boldsymbol{\beta}(\boldsymbol{\omega}_i), \quad \gamma = \Delta t \frac{T^{ab}}{\phi_0 V}, \tag{4.15}$$

and

$$b_{i^+} = (p_i - p_{i+1})\Lambda(\boldsymbol{\omega}_i), \quad b_{i^-} = (p_i - p_{i-1})\Lambda(\boldsymbol{\omega}_i - 1). \tag{4.16}$$

Now the internal Jacobian row of the equation can be written as:

$$\begin{bmatrix} \gamma \mathbf{B}_{i-1} b_{i^-} + \gamma \boldsymbol{\beta}_{i-1} \times \mathbf{b}'_{i-,i-1} \\ \mathbf{A}_i + \gamma(\mathbf{B}_i b_{i+} + \boldsymbol{\beta}_i \times \mathbf{b}'_{i+,i} + \boldsymbol{\beta}_{i-1} \times \mathbf{b}'_{i-,i}) \\ \gamma \boldsymbol{\beta}_i \times \mathbf{b}'_{i+,i+1} \end{bmatrix}^T, \tag{4.17}$$

where

$$\mathbf{A}_i = \left[\frac{\partial \boldsymbol{\alpha}_i}{\partial \boldsymbol{\omega}_i}\right] = \left[\frac{\partial \alpha_c}{\partial p_i}\frac{\partial \alpha_c}{\partial z_{i,1}}...\frac{\partial \alpha_c}{\partial z_{i,n_c-1}}\right], c = 1,...,n_c, \tag{4.18}$$

$$\mathbf{B}_i = \left[\frac{\partial \boldsymbol{\beta}_i}{\partial \boldsymbol{\omega}_i}\right] = \left[\frac{\partial \beta_c}{\partial p_i}\frac{\partial \beta_c}{\partial z_{i,1}}...\frac{\partial \beta_c}{\partial z_{i,n_c-1}}\right], c = 1,...,n_c, \tag{4.19}$$

$$\mathbf{b}'_{i^-,i-1} = \left[\frac{\partial b_{i-}}{\partial \boldsymbol{\omega}_{i-1}}\right]^T = (p_i - p_{i-1})\frac{\partial \Lambda_{i-1}}{\partial \boldsymbol{\omega}_{i-1}} - \begin{bmatrix}\Lambda_{i-1}\\ 0\\ \vdots\\ 0\end{bmatrix}, \tag{4.20}$$

$$\mathbf{b}'_{i-1,i} = \left[\frac{\partial b_{i-}}{\partial \boldsymbol{\omega}_i}\right]^T = \begin{bmatrix}\Lambda_{i-1}\\ 0\\ \vdots\\ 0\end{bmatrix}, \tag{4.21}$$

$$\mathbf{b}'_{i+,i} = \left[\frac{\partial(-b_{(i+1)^-})}{\partial \boldsymbol{\omega}_{(i+1)-1}}\right] = -\mathbf{b}'_{(i+1)-,(i+1)^-1}, \tag{4.22}$$

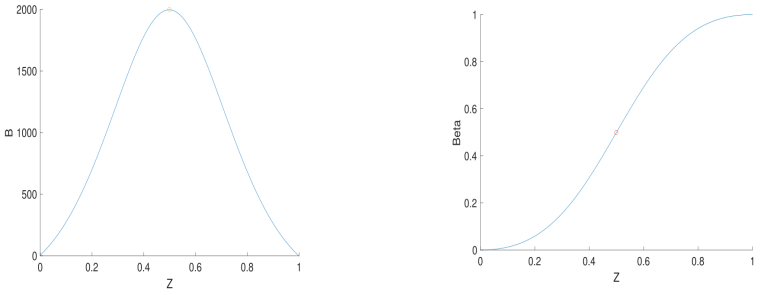
$$\mathbf{b}'_{i+,i+1} = \left[\frac{\partial(-b_{(i+1)^-})}{\partial \boldsymbol{\omega}_{(i+1)}}\right] = -\mathbf{b}'_{(i+1)-,(i+1)}. \tag{4.23}$$

For binary systems, under the assumption of an incompressible fluid, α_c and β_c are equivalent to overall composition z_c and the compositional fractional flow curve (F_c) respectively. When the residual saturation is zero at the initial conditions, the β-operator becomes zero since the compositional fractional flow is zero at residual saturation equals to zero. Therefore, the internal Jacobian for the nonlinear iteration can be written as:

$$\begin{bmatrix}\gamma \mathbf{B}_{i-1} b_{i^-} & 1 + \gamma(\mathbf{B}_i b_{i+}) & 0\end{bmatrix}, \tag{4.24}$$

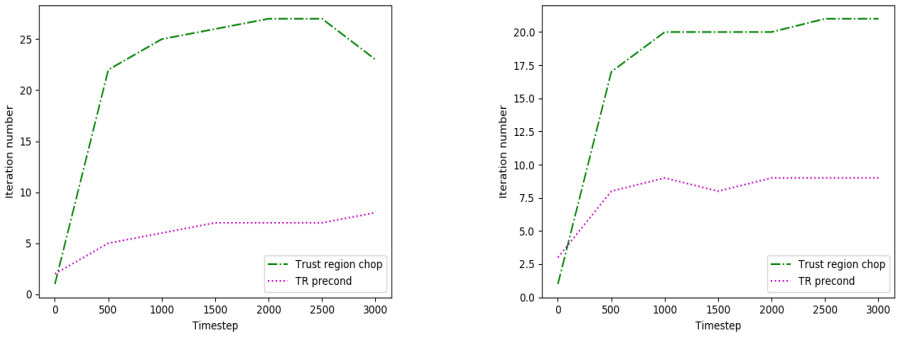
In Eq. 4.24, if **B** becomes zero at the end point of fractional flow for the component, the internal Jacobian matrix becomes the identity matrix that does not allow the composition to propagate downstream. Therefore, to maximize the propagation for the composition front downstream in a single iteration, we need to maximize the **B** term.

From Fig. 4.17 for the incompressible version of dead-oil kernel (Check Appendix A.4 for fluid description) without the dependency of viscosity on pressure, it is clear that the **B** is maximum at the inflection point. Therefore, to maximize the composition propagation in more than one gridblock in one iteration, we introduced the inflection point of the β-operator for each gridblock as an initial guess. Fig. 4.18 compares our nonlinear solver performance with and without preconditioning for both dead-oil with pressure variation and binary compositional kernel. We can see that by applying the preconditioning strategy, the number of iterations for increasing timestep grows slower for the preconditioned system. In Appendix A.3, you can find the preconditioning performance's sensitivity to the transport function parameters.



(a) (b)

Figure 4.17: (a) The derivative of the β-operator with respect to z for dead-oil system (b) Fractional flow for dead-oil system



(a) Dead-oil kernel (b) Compositional kernel

Figure 4.18: TR nonlinear solver with/without preconditioning

4.6. DISCUSSION AND CONCLUSION

For simulation of CO_2 use and storage (CCUS) in subsurface reservoirs with complicated heterogeneous structures, a model that includes multiphase compositional flow and transport is needed. The discretized governing equations are highly nonlinear, and Newton's technique is frequently used to solve them. Newton's solution technique does not ensure convergence and is extremely dependent on the timestep choice. In this work, we investigate the nature of nonlinearities in CCUS simulations and suggest solutions to a general compositional problem.

Operator based linearization Facilitates the nonlinear analysis of the discrete residual since the nonlinearty is lumped in the operators and one can understand the nonlinearity by analysing a nonlineairty of the operator. We started by analyzing the nonlinearity of the convective flux operator for binary and ternary kernels. Two significant attributes of the flux term result in challenges when it comes to achieving nonlinear con-

vergence with the Newton solver:

- **Kinks line**

  Kinks manifest themselves when there are discontinuities in the first-order derivatives of a function. They coincide with sudden changes in the curvature of the flux function and represent a significant contributor to Newton oscillations. Comparing the phase diagram and hessian diagram clearly illustrates that The occurrence of kinks arises mainly due to the the crossing of phase boundaries.

- **Inlflection line** For the scalar function, the inflection point is the point where the function changes its concavity conditions. for multi-variable functions, the convexity of the function depends on the quadratic forms of the Hessian matrix. Our analysis of the hessian illustrates that each component has its own unique inflection line in the parameter space. in a general case inflection line depends on pressure and temperature.

Detecting inflection and kink lines and requires rigorous hessian assembly which is computationally expensive and cumbersome. We proposed the directional derivative algorithm to estimate the trust-region boundaries instead of full hessian analysis to overcome the computational cost.

We present an advanced nonlinear solver based on a trust-region technique aimed to solve multiphase multi-component transport problems. The trust region solver is based on the analysis of multi-dimensional tables connected to parameterized highly nonlinear convection operators. These operators are associated with the governing equations and are built for a newly introduced Operator-Based Linearization approach. The inflection line and kinks in the parameter space determine the delineation of the trust-region zones. According to our nonlinear study of convective operators for binary and ternary systems, each component has its inflection line within the two-phase region. In addition, kink lines appear when phase boundaries are crossed. These boundaries could change in the parameter space of the problems based on the direction of the Newton trajectory.

We track the nonlinear trajectory and segment the parameter space of the problem into a set of trust regions where the hyperbolic operators keep their second-order behavior (i.e., they remain either convex or concave). We approximate these trust regions in the solution process by detecting the boundary of convex regions via analysis of the directional derivative. By drawing multiple trial trajectories on binary and ternary diagrams we observe that our algorithm can detect these boundaries correctly. Moreover, it is less computationally expensive since we do not compute the entire hessian in our technique and instead compute the directional derivative while tracking the nonlinear update. After detecting all the boundaries along the nonlinear trajectory, the proposed nonlinear solver locally constrains the update of the overall compositions across the boundaries of these regions. We tested our nonlinear solver for several reservoir models starting from the single cell to a fully 3D heterogeneous model. Our numerical results show that the trust-region solver avoids overshoots in the nonlinear update which lead to superior convergence in comparison to conventional nonlinear solvers.

A promising future research direction could be to reduce the computational overhead related to locating the trust-region boundaries and carrying out the chopping. While we showed that directional derivative is cheaper than full Hessian assembly, there is still additional overhead since detection of special points (kinks and inflection) are happening on the fly during simulation. The major cause of the overhead is due to the tracking the Newton trajectory in nonlinear operator space. One possibility is that once the special points have been identified, we save the value to avoid recalculation for the next iteration. Another possible future project would be to combine the TR solver with another type of solver. Based on our single cell analysis, we can see that once the solution is in a single-phase region, conventional Newton solvers work, so we can switch between different solvers adaptively if we observe Newton solver is struggling.

5

PROXY MODELLING BASED ON MACHINE LEARNING

Summary

CO_2 utilization and storage (CCUS) simulation in subsurface reservoirs with complex heterogeneous structures necessitates a model that can capture multiphase compositional flow and transport. The governing equations are highly nonlinear due to the complex thermodynamic behavior which involves the appearance and disappearance of multiple phases. Accurate simulation of these processes requires the use of stable numerical methods. While Machine Learning (ML) approaches have been used to solve a variety of nonlinear computational problems, a new approach based on physics-informed neural networks (PINNs) has been proposed for solving partial differential equations. Unlike typical ML algorithms that require a large dataset for training, PINNs can train the network with unlabelled data. The applicability of this method has been explored for multiphase flow and transport in porous media. However, for nonlinear hyperbolic transport equations, the solution degrades significantly. In this work, we propose sequential training PINNs to simulate two-phase transport in porous media. The main concept is to retrain the neural network to solve the PDE over successive time segments rather than train for the entire time domain simultaneously.

In the second section, we explore the application of neural networks for training the operator space of the PDE. We start with training with labeled data for the classic S-shape fractional flow operator. Next, we extend the training for a reduced-order proxy model for multi-dimensional highly heterogeneous reservoirs.

Part of The material presented in this chapter has been submitted to the Journal of Machine Learning for Modeling and Computing and the second part has been published in SPE Reservoir Simulation Conference 2023

5.1. APPLICATION OF PHYSICS-INFORMED NEURAL NETWORKS FOR MULTIPHASE FLOW

5.1.1. INTRODUCTION

The global attempt to mitigate the effects of climate change has increased due to the urgency of lowering carbon emissions. Carbon dioxide capture, utilization, and storage (CCUS) is one of the existing technologies that has substantial potential for lowering greenhouse gas emissions. The capture of carbon dioxide from industrial sources, compression, transportation, and subsequent utilization for operations such as permanent storage in deep underground geological formations and increased oil recovery in depleted areas are all part of CCUS. However, proper modeling of multiphase compositional flow and transport in underground reservoirs with complex heterogeneous structures is required to simulate effectively CO_2 use and storage (CCUS).

Compositional simulation deals with the modeling of the flow of multiple phases in a porous medium. The interactions between the hydrocarbon phases take place at the interplay of phase behavior, flow, and transport. Compositional simulation continues to be a challenging problem. Complexities are mainly due to nonlinear couplings between multi-phase multi-component flow in porous media with thermodynamic phase behavior [13, 88]. Conventional compositional simulation is based on the solution of the discretized governing equations describing the mass, energy, and momentum transfer in the reservoir either implicitly or explicitly. The fully-implicit method (FIM) is preferred in practice, with the nonlinear system solved by a Newton method due to the fewer restrictions on timestep size. In practical applications, the highly nonlinear nature of problems involving kinks and inflection points poses a significant challenge for Newton's method to achieve convergence, particularly when large timesteps are utilized. Consequently, to overcome this limitation, alternative nonlinear solvers have been proposed [24, 26, 27] or smaller timesteps are selected for simulations.

Machine learning (ML) techniques, particularly deep learning [33], are gaining prominence in the computer science and engineering fields. Notably, physics-informed neural networks (PINNs) are being utilized to solve problems where there is knowledge of engineering conservation equations and constitutive closure relationships, but without labeled data [34]. By constructing neural networks with several hidden layers, coupled with nonlinear activation functions, complex nonlinear solutions can be approximated. As a result, PINNs have been employed to tackle diverse applications governed by differential equations, such as the Euler equation [35, 36], gas dynamics [37, 38], water dynamics [39] and chemical kinetics [40, 41]. PINNs have demonstrated their versatility in several applications, including data assimilation and numerical models, parameter identification (i.e., solving inverse problems) [34, 42], and uncertainty quantification [43, 44].

The application of PINNs has been widely explored in subsurface flow and transport. Two-phase immiscible transport in porous media is typically described by a nonlinear first-order hyperbolic PDE, also known as the Buckley-Leverett equation. Standard physics-informed neural networks (PINNs) have been utilized to model this phenomenon [45]. The authors demonstrated that PINNs cannot find the solution in the case of the steep saturation front with the nonconvex flux function [45]. Only after an

artificial diffusion term was added to the original conservation equation, the neural networks solution managed to approximate the true solution [45]. There were multiple attempts to solve this problem by modifying the loss function. Rodriguez-Torrado *et al.* [46] author proposed a new neural network architecture known as physics-informed attention-based neural networks (PIANNs), which is a blending of recurrent neural networks and attention mechanisms. Fraces and Tchelepi [47] introduced yet another solution to the B-L problem. Their method entails embedding the entropy and velocity constraints into the neural network residual.

Recently, a sequential training scheme has been proposed for PINNs [89, 90]. In this approach, unlike training for the entire spatiotemporal domain, we discretize the time domain and march in time to reach the end time. Mattey and Ghosh [89] demonstrates the effectiveness of the sequential training approach for Allen Cahn and Cahn Hilliard equations. Krishnapriyan *et al.* [90] demonstrated the method's effectiveness in solving a one-dimensional reaction-diffusion problem. The present study aims to investigate the efficacy of sequential training in the context of hyperbolic transport equations in porous media. This approach is inspired by the conventional practice of using Newton's method to solve partial differential equations in reservoir simulation, while also taking into consideration the fact that reducing the timestep can help improve the effectiveness of Newton's method. Our work suggests a sequential training scheme with a dynamic timestep that reduces the number of timesteps required for training, instead of uniformly marching in time.

Through the analysis of a single timestep residual, our findings demonstrate that the nonlinearity of the flux becomes increasingly dominant as the timestep increases. This nonlinearity can be translated into the residual of the loss function. As a result, the final value of the loss function is higher for larger timesteps. Next, we extend the sequential training approach to compositional problems, in which nonlinearity is more pronounced due to complex phase transitions. In compositional problem, the presence of discontinuous points that coincide with the phase transition zone can severely hinder the convergence of the loss function for neural networks. To mitigate this issue, we utilize a continuous sigmoid approximation of the fractional flow. Finally, we test the performance of the sequential training approach on pure hyperbolic transport in porous media in the 1D domain, ranging from the classic Buckley-Leverett problem to both immiscible and miscible compositional transport. Our findings demonstrate that the sequential training method outperforms the standard training approach in terms of accuracy, for both immiscible and miscible transport scenarios.

5.1.2. GOVERNING EQUATIONS

We consider the general form of the transport equations for an isothermal multiphase compositional problem with n_p phases and n_c components that can be written as:

$$\frac{\partial}{\partial t}(\phi \sum_{j=1}^{n_p} x_{cj}\rho_j S_j) + div \sum_{j=1}^{n_p} x_{cj}\rho_j \mathbf{v}_j + \sum_{j=1}^{n_p} x_{cj}\rho_j \tilde{q}_j = 0, \quad c = 1,\dots,n_c. \tag{5.1}$$

Here, ϕ is porosity, x_{cj} is the mole fraction of component c in phase j, S_j is the phase saturation of phase j, ρ_j is phase molar density, $\mathbf{v}_j$ is phase velocity and $\tilde{q}_j$ is phase rate per unit volume. Darcy's law is applied to describe how each phase flows:

$$\mathbf{v}_j = -\mathbf{K}\frac{\mathbf{k_{rj}}}{\mu_\mathbf{j}}(\nabla \mathbf{p_j} - \rho_\mathbf{j}\nabla \mathbf{d}), \tag{5.2}$$

where

- $\mathbf{K}$ – permeability tensor,
- k_{rj} – relative permeability,
- μ_j – phase viscosity,
- $\mathbf{p_j}$ – vector of pressures in phase j,
- ρ_j – phase density,
- $\mathbf{d}$ – vector of depths (positive downwards).

The closure assumption of instantaneous thermodynamic equilibrium further increases the nonlinearity. We used the overall molar formulation suggested by [51]. In this formulation, the following system must be solved at any grid block containing a multiphase (n_p) multi-component n_c mixture:

$$r_c = Z_c - \sum_{j=1}^{n_p} v_j x_{cj} = 0, \tag{5.3}$$

$$r_{c+n_c} = f_{c1}(p,T,x_1) - f_{cj}(p,T,x_j) = 0, \tag{5.4}$$

$$r_{j+n_c*n_p} = \sum_{c=1}^{n_c}(x_{c_1} - x_{cj}) = 0, \tag{5.5}$$

$$r_{n_p+n_c*n_p} = \sum_{j=1}^{np} v_j - 1 = 0. \tag{5.6}$$

Here $Z_c = \sum x_{cj}\rho_j s_j / \sum \rho_j s_j$ is overall composition and $f_{cj}(p,T,x_{cj})$ is the fugacity of component c in phase j. The solution of this system is called a multiphase flash [52] and needs to be applied at every nonlinear iteration [13]. The solution provides molar fractions for each component x_{cj} and phase fraction v_j. The above system of equations provides a complete mathematical statement for multi-phase multi-component transport. Here we concentrate on two-phase, two-component incompressible transport for miscible and immiscible cases.

$$\frac{\phi \partial Z_c}{\partial t} + v_t \frac{\partial F_c}{\partial x} = q_c, \quad x \in \Omega, \quad c \in \{H_2O, CO_2\} \tag{5.7}$$

where Z_c is the overall composition of component c, v_t is the total velocity. q_c is the source, sink term. F_c is the fractional flow of the component c which is defined as:

$$\begin{cases} F = x_c(1 - f(s)) + y_c f(s), & \text{Two-phase zone} \\ F = Z, & \text{Single-phase zone} \end{cases} \tag{5.8}$$

$$f_\alpha = \frac{\lambda_\alpha}{\lambda_{CO_2} + \lambda_{water}}, \quad \alpha = \{H_2O, CO_2\} \tag{5.9}$$

where x_c and y_c are molar fractions of component c in the liquid and gas phases, respectively. $\lambda_\alpha = (kk_{r\alpha})/\mu_\alpha$ stands for the phase mobility, μ_α is the viscosity of the phase, $k_{r\alpha}(S_\alpha)$ is the relative phase permeability. For the immiscible two-phase Buckley-Leverett transport test case, the fractional flow F_c is equivalent to f_α. The initial and boundary conditions are:

$$\begin{cases} Z_c(x,t) = 0, \ \forall x \ \& \ t = 0; & \text{Initial condition} \\ Z_c(x,t) = 1, \ x = 0 \ \& \ t > 0 & \text{Boundary condition} \end{cases} \tag{5.10}$$

5.1.3. STD-PINNS SOLUTION

[34] proposed that the solution of the partial differential equation (PDE) could be approximated by a deep neural network through the loss function of the neural network. In a standard PINNs solution, a neural network is trained for the entire spatial-temporal domain. Let's examine the standard format of a PDE

$$Z_c^t + \mathcal{N}(Z_c) = 0, \tag{5.11}$$

where Z(x,t) denotes the latent (hidden) solution (overall composition), superscript $(\bullet)^t$ is a time derivative. $\mathcal{N}[.]$ is a nonlinear differential operator. By adopting the methodology of [34], the solution Z(t, x) to the PDE is estimated using a deep neural network that relies on a group of parameters denoted as θ. In simpler terms, the PDE solution is expressed as a sequence of function compositions.

Deep neural networks are composed of n_l series of functions

$$Z_\theta(\mathbf{X}) = z_{n_l}(z_{n_{l-1}}(...(z_2(z_1(\mathbf{X}))))) \tag{5.12}$$

$$z_i(\mathbf{X}) = \sigma(\mathbf{W_i X} + \mathbf{b_i}) \qquad i = 1, ..., n_l. \tag{5.13}$$

where each hidden layer consists of a stack of artificial neurons that process input feature matrix $\mathbf{X}$ as the weighted sum of weights $\mathbf{W_i}$ and biases $\mathbf{b_i}$ before passing through activation function σ ($tanh$ in our study), θ is the ensemble of all the model parameters based on weight and biases:

$$\theta = \{W_1, W_2, .., W_{n_l}, b_1, b_2, ..., b_{n_l}\}. \tag{5.14}$$

To provide the neural network with the physics specified by the PDE, we define the residual of the PDE as the left-hand side of Equation 5.11 and replace Z_c with $\tilde{Z}_c$.

$$R_\theta(x,t) := (\tilde{Z}_c)_t + \mathcal{N}(\tilde{Z}) = 0, \tag{5.15}$$

Here, $\tilde{Z}_c(x,t)$ is the PDE solution which is approximated by a neural network. The loss function of the neural network is made of three error conditions:

$$L_{tot} = L_i + L_{ii} + L_{iii}, \tag{5.16}$$

Each error is:

$$\begin{cases} L_i = \frac{\sum_K^{N_i} \tilde{Z}(x_k^i,0) - Z_k^i}{N_i}, & \text{MSE of the initial condition,} \\ x_k^i \in \Omega\,, & \\ L_{ii} = \frac{\sum_K^{N_b} \tilde{Z}(x_k^b,t) - Z(x_k^b,t_k)}{N_b}, & \text{MSE of the boundary condition,} \\ (\mathbf{x_k^b},\, t_k^b) \in \Gamma \times (0,T]\,, & \\ L_{iii} = \frac{\sum_K^{N_i} (R(x_k^t,t_k^k))}{N}, & \text{MSE of the residual of the PDE.} \\ (\mathbf{x_k^r},\, t_k^b) \in \Gamma \times (0,T], & \end{cases} \tag{5.17}$$

Here we introduce N_i, N_b and N_r which are the number of initial, boundary, and residual collocation points, where Γ is the boundary of Ω, Z_k^i is the given initial condition at $(x_k^i, 0)$. The superscript $(\bullet)^b$, $(\bullet)^i$, $(\bullet)^r$ stands for a boundary, initial and residual conditions of the PDE. In our study, we focus on the 1-D equation 5.7 with the initial and boundary conditions given in equation 5.10.

Figure 5.1 shows schematically the fully connected neural network architecture.

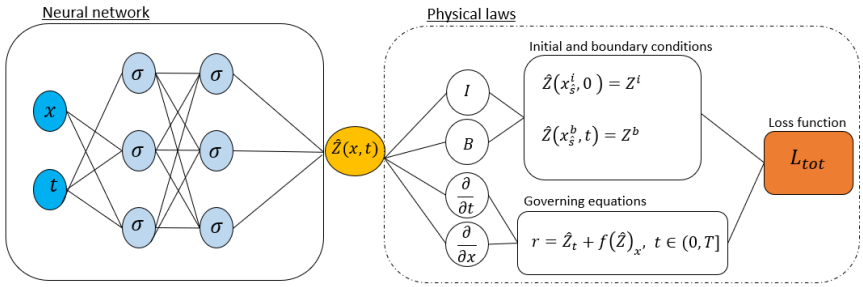


Figure 5.1: Standard PINNs architecture

5.1.4. SEQUENTIAL TRAINING WITH DYNAMIC TIME STEPPING

Unlike standard PINNs training, we train for the entire domain at once and discretize the time domain into several segments. Note that this strategy is similar to the numerical methods used in scientific computing, where space-time problems are typically harder to solve than time-marching methods. Figure 5.2 illustrates the sequential training scheme versus the standard training scheme.

In a dynamic time-stepping scheme, the first-time step, $[0, T_1]$ is solved with a similar loss function as std-PINNs in Equation.5.16.

For all the subsequent time segments, we utilize the following loss function:

$$L_{tot}^{\Delta T_n} = L_i + L_{ii} + L_{iii} + L_{iv}, \tag{5.18}$$

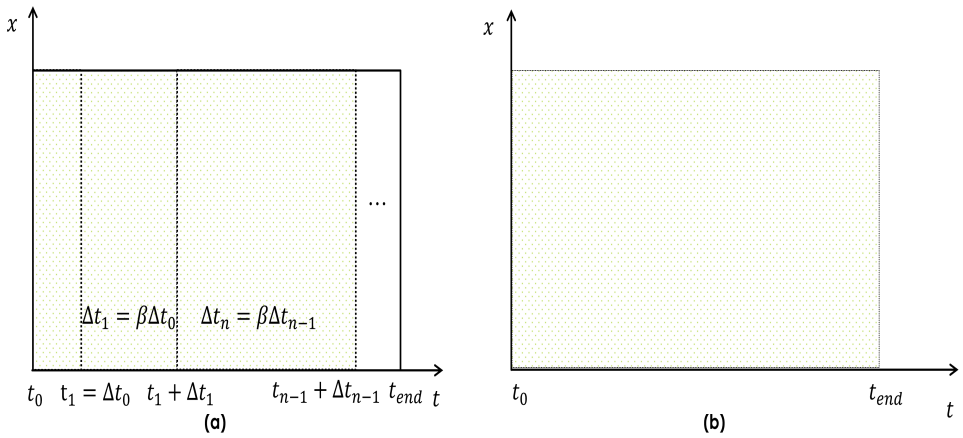


Figure 5.2: Training scheme (a) dynamic sequential time-stepping scheme (b) standard training scheme

Each element of the loss function is the mean squared error (MSE) of different terms as:

$$
\begin{cases}
L_i = \frac{\sum_K^{N_i} \tilde{Z}(x_k^i, T_{n-1}) - Z_k^i}{N_i}, & \text{MSE of the initial condition,} \\
x_k^i \in \Omega\,, & \\
L_{ii} = \frac{\sum_K^{N_b} \tilde{Z}(x_k^b, t) - Z(x_k^b, t_k)}{N_b}, & \text{MSE of the boundary condition,} \\
(\mathbf{x_k^b},\, t_k^b) \in \Gamma \times (T_{n-1}, T_n], & \\
L_{iii} = \frac{\sum_K^{N_i} (R(x_k^t, t_k^k))}{N}, & \text{MSE of the residual of the PDE,} \\
(\mathbf{x_k^r},\, t_k^r) \in \Gamma \times (T_{n-1}, T_n], & \\
L_{iv} = \tilde{Z}(x,t) - \underline{\tilde{Z}}(x,t), & \text{MSE of solution of previous time.} \\
(\mathbf{x_k^s},\, t_k^s) \in \Omega \times (0, T_{n-1}]\,, &
\end{cases}
\tag{5.19}
$$

Here (x_k^i, T_{n-1}) is used to denote the collection of points where the calculation of the error on the initial condition is evaluated. $(\mathbf{x_k^b}, t_k^b)$ is the set of points where the error on the boundary conditions is calculated during the specified time interval $(T_{n-1}, T_n]$. The points on grid $(\mathbf{x_k^s}, t_k^s)$ store the solution obtained during the n-th segment on the interval $(0, T_{n-1}]$, for its usage in the $(n)^{th}$ segment. By incorporating L_{iv} terms into the loss functions, the neural network can ensure backward compatibility [89], meaning that the single neural network can replicate the solution from all the previous time segments while solving the PDE for a specific time segment.

Next, we introduce a dynamic time-stepping scheme that instead of marching in time uniformly for training, we will make dynamic sequential training. Dynamic time-stepping allows us to train for a larger timestep and consequently smaller training time. The algorithm is as follows:

- We start with the small timestep,
- If the loss function decreases as a predefined tolerance, we multiply the next timestep to the fixed ratio β
- If it fails to reach the tolerance, we divide the next timestep by the same constant β
- If the maximum timestep Δt_{max} is reached, we keep it for further training until reaching the final time of the simulation.

Figure 5.3 shows the neural network of the sequential training with backward-compatibility and dynamic time-stepping scheme over the interval $[t_{n-1}, t_n]$.

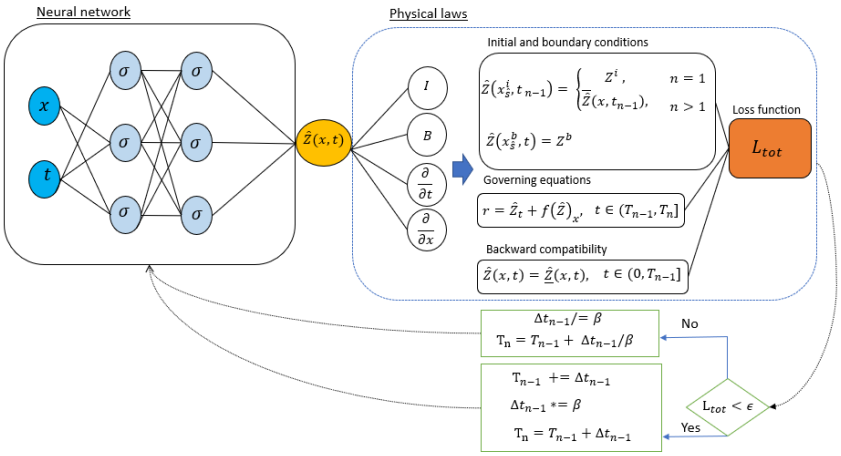


Figure 5.3: Sequential PINNs architecture with backward compatibility and dynamic time-stepping scheme

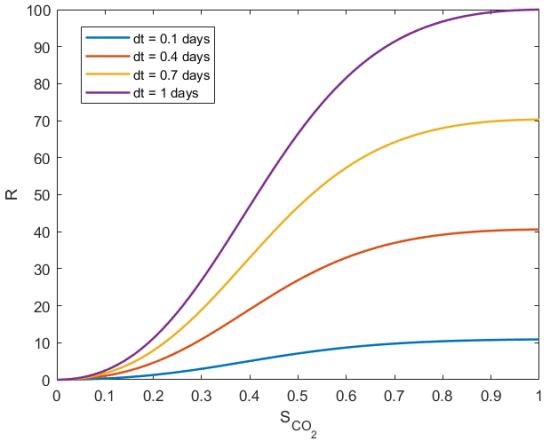
5.1.5. RESULTS

SINGLE TIMESTEP TRAINING

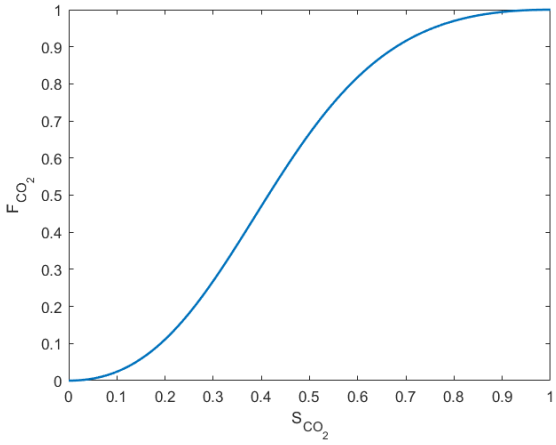
To motivate our sequential approach, we consider the following transport equation with a fixed left boundary:

$$R_{CO_2} = S_{CO_2} + \frac{\Delta t}{\Delta x}\left(F_r(S_{CO_2}) - F_l\right), \tag{5.20}$$

We investigated the nonlinearity of the residual over four progressive time steps and analyzed the relationship between timestep size and nonlinearity. Figure 5.4 shows that as the timestep size increases, the nonlinearity of the residual also increases. This is due to the nonlinearity of the flux function, which plays a crucial role in determining the nonlinearity of the residual. The figure further demonstrates that the residual becomes more significant and more closely aligned with the flux function as the timestep size increases.



(a)



(b)

Figure 5.4: (a) Residual for multiple timesteps, (b) Flux function

Here, we make a test case in which we train for a single progressive timestep and observe the behavior of loss function for the Buckley–Leverett problem. From figure 5.5, we can observe that the final value of the loss function evolves with the training for the larger timesteps. We used fully connected neural networks with 8 layers and each layer has 20 neurons. We use the L-BFGS-B optimizer. The training data points of the neural network are recorded in table 5.1.

FULL 1-D SIMULATION BUCKLEY-LEVERETT TEST CASE

Here we compare the solution of seqPINN with dynamic timstepping and stdPINN for the two-phase immiscible Buckley–Leverett problem. Initially, the 1D domain is fully

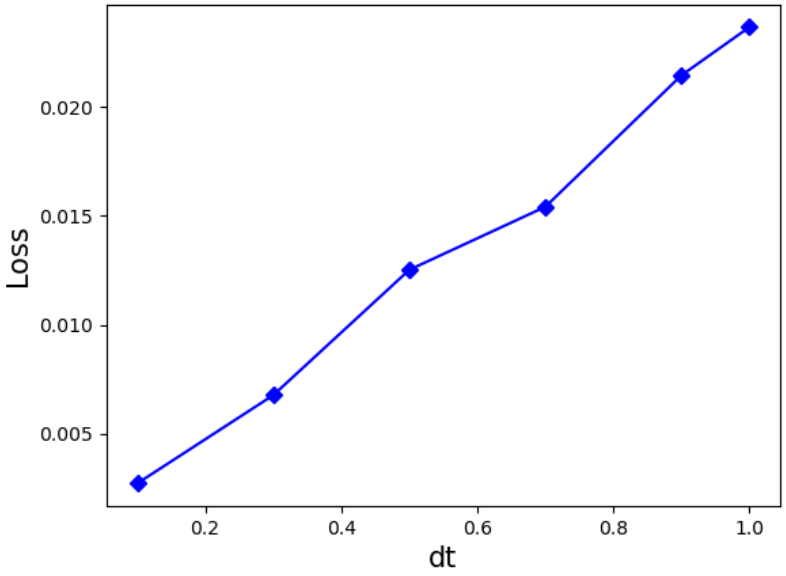


Figure 5.5: Evolution of loss function for different timesteps

Variable	Description	Number
N_i	Initial condition points	300
N_b	Boundary condition points	300
N_r	Collocation points	$10000 \times \Delta t$

Table 5.1: Description of training data

saturated by the non-wetting phase and we inject a wetting phase at the left boundary. We used fully connected neural networks with 8 layers and each layer has 20 neurons. We use the L-BFGS-B optimizer. The activation is represented by the hyperbolic tangent function (tanh)..

The tolerance for the dynamic sequential time-stepping is set as 3.5e-3 in this test case. Figure 5.6 compares the precise analytical solution to the solution predicted by the standard PINNs at time intervals t = 0.1, 0.3, and 0.7. As we can see, the standard PINNs cannot find the solution of the front accurately. Figure 5.7 compares the analytical solution to the solution of the PINNs with dynamic sequential training. As we can see, sequential training is capable of predicting the solution more accurately with respect to the standard training scheme. As we can see sequential training can capture the front accurately. Figure 5.8 shows the full solution in spatiotemporal for the Buckley-Leverett problem.

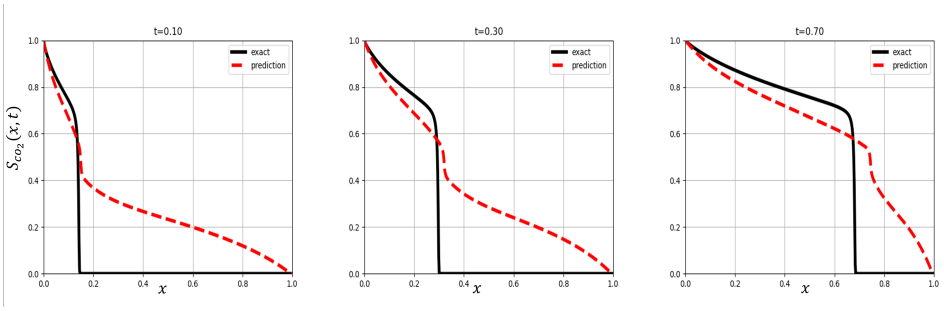


Figure 5.6: Solution of the PINNs with standard training scheme

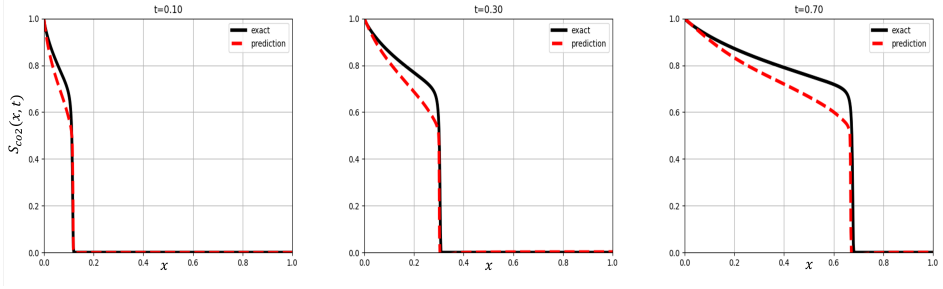


Figure 5.7: Solution of the PINNs with sequential training scheme

FULL 1-D SIMULATION COMPOSITIONAL TEST CASE

In the compositional test case, we see two non-differentiable points related to phase changes in fractional flow formulations of compositional transport. In the two-phase region, the fractional flow has an S-shaped curve, whereas it is linear in single-phase conditions. The discontinuous derivatives in the flow function can largely degrade the convergence of the loss function for the neural networks. To overcome this issue, we propose the smooth formulation of the fractional flow using the sigmoid step function:

$$F_{smooth} = H \times F + (1-H) \times Z, \tag{5.21}$$

$$H = Sigmoid(S) - Sigmoid((S-1)), \tag{5.22}$$

$$Sigmoid = \frac{1}{1+e^{-\alpha S}}, \tag{5.23}$$

$$\tag{5.24}$$

where F is the non-smooth fractional flow given by equation 5.8. The accuracy of the smooth approximation highly depends on the parameter α which indicates how steep the sigmoid and step function could be. From figure 5.9 we can observe that, the higher the α, the better the approximation.

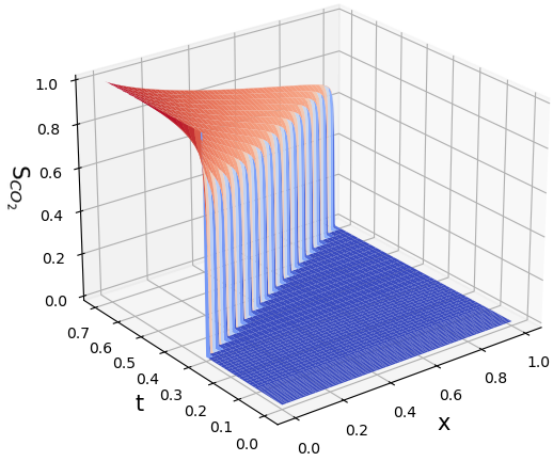


Figure 5.8: Solution in spatiotemporal space

NEAR-MISCIBLE TEST CASE

Here we test the sequential PINNs training on miscible fluids with phase behavior controlled by constant K-values with $K = \{2.5, 0.3\}$. Initially, the 1D domain is fully saturated by the non-wetting phase and we inject a wetting phase at the left boundary. We use $\alpha = 20$ for the sigmoid approximation of the fractional flow as a trade-off between loss and accuracy. In this particular test case, the tolerance value for the dynamic sequential time-stepping is set to 7e-3. Figure 5.11 and 5.10 show the solution of the sequential training versus standard training respectively. We can observe that the sequential training scheme can predict the solution better than the standard training scheme.

IMMISCIBLE TEST CASE

We test sequential PINNs training on quasi-immiscible compositional transport with phase behavior regulated by constant K-values of 2 and 0.002. The non-wetting phase first completely saturated the 1D domain, and we injected a wetting phase at the left border. For the sigmoid approximation, we use $\alpha = 20$. In this particular test case, the tolerance value for the dynamic sequential time-stepping is set to 7e-3. Figures 5.13, 5.12 illustrate and compare the PINNs solution trained sequentially and standard versus the analytical solution for three different times $t = 0.1$, 0.3, and 0.7. We can observe that the sequential training scheme can capture the shock more accurately with respect to the standard training scheme. However, there is still a small difference between the analytic solution and the PINNs solution with the sequential training scheme.

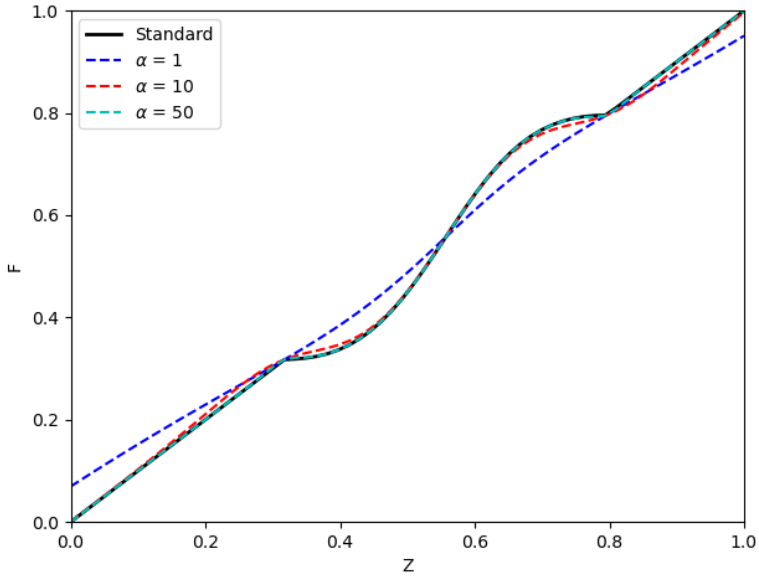


Figure 5.9: Standard fractional flow versus smooth approximation of fractional flow using a sigmoid function with different alpha coefficient

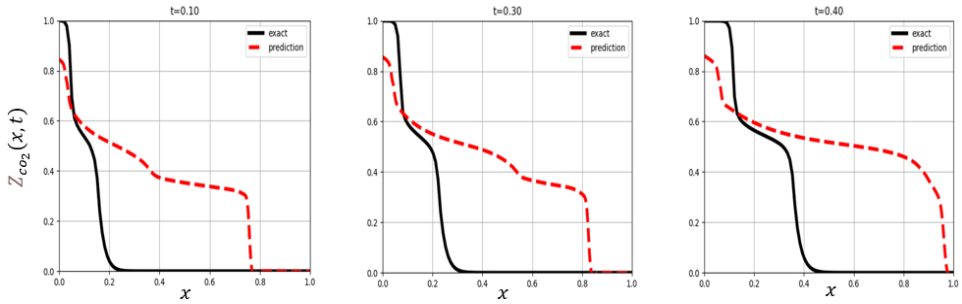


Figure 5.10: Solution of the PINNs with standard training scheme and dynamic time-stepping

5.2. APPLICATION OF DEEP NEURAL NETWORKS FOR PHYSICS-BASED PROXY MODELLING

5.2.1. INTRODUCTION

In this section, we will discover another utilization of neural networks for proxy modeling in compositional simulation. A reduced number of components increases the efficiency of compositional simulation by reducing the number of equations that need to be solved. In the lumping technique, the problem is reformulated by presenting the formu-

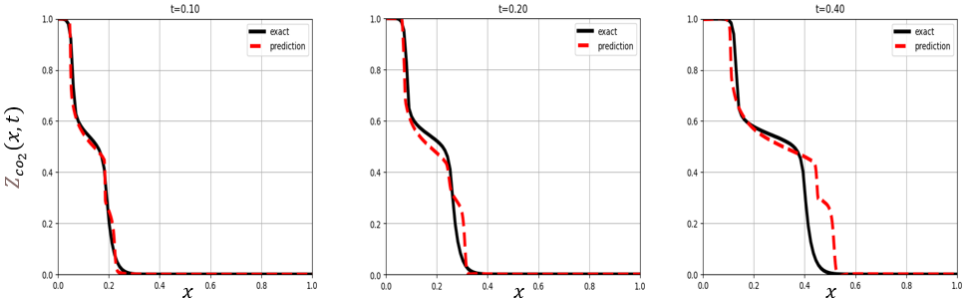


Figure 5.11: Solution of the PINNs with sequential training scheme and dynamic time-stepping

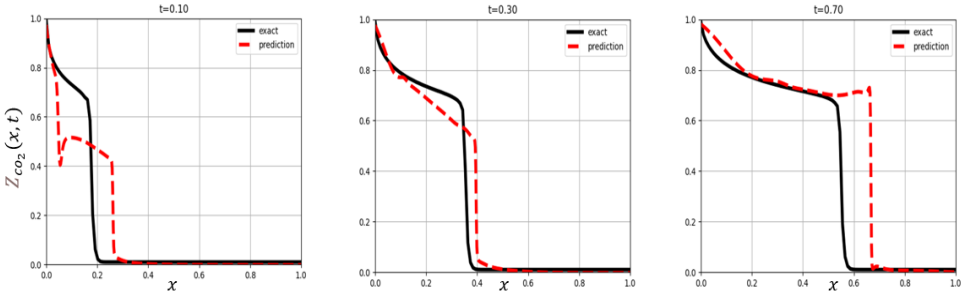


Figure 5.12: Solution of the PINNs for compositional immiscible test case with standard training scheme and dynamic time-stepping

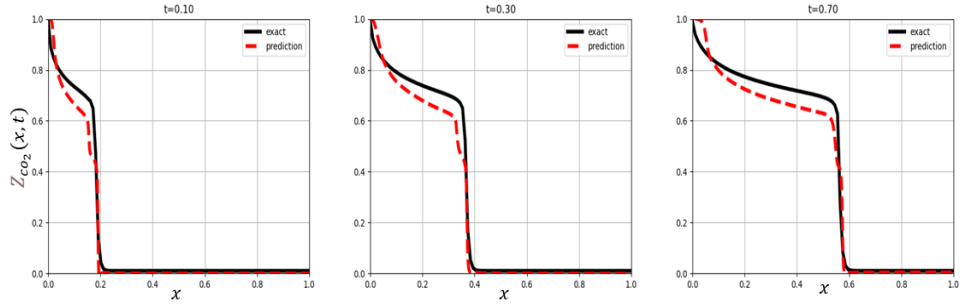


Figure 5.13: Solution of the PINNs for a compositional immiscible test case with sequential training scheme and dynamic time-stepping

a

lation with a limited number of equivalent components and delumping the components at the production stream [91, 92]. Further techniques rely on a numerical representation of the method of characteristics. Tang and Zick [93] propose a limited compositional reservoir simulator that solves a four-component problem by solving an equivalent pseudo-ternary problem. Ganapathy and Voskov [94] proposes a physics-based method for proxy modeling, termed multi-scale reconstruction in physics. In their method,

a pseudo-binary proxy model is used to solve any transport problem with an arbitrary number of components. Their model is applied to four and eight-component systems with either the equation of state or constant K-value thermodynamics. It accurately locates the major features of the compositional displacement profile [95].

In this work, the neural network is used to train the operator space of the physics-based proxy of Ganapathy and Voskov [94] and improve performance for a compressible, gas-injection problem at isothermal conditions. Our approach embeds neural networks in the nonlinear operator of the proxy model to improve the performance of the proxy model. Our method is implemented in TensorFlow and utilizes automatic differentiation and operator-based linearization approaches. Transfer training is employed for increased computational efficiency and solution accuracy.

5.2.2. OPERATOR TRAINING FOR OBL APPROACH

In contrast to PINNs, our approach involves employing a neural network as a solver to estimate the solution of the partial differential equation (PDE). In this particular section, we employed a neural network to train the PDE operators, which were then integrated into the DARTS framework. By utilizing a fully implicit finite volume solver, we obtained the solution of the PDE. Figure 5.16 illustrates the neural network architecture for training the fractional flow operator. Figure 5.15 compares the analytical fractional flow versus the neural network prediction of the fractional flow.

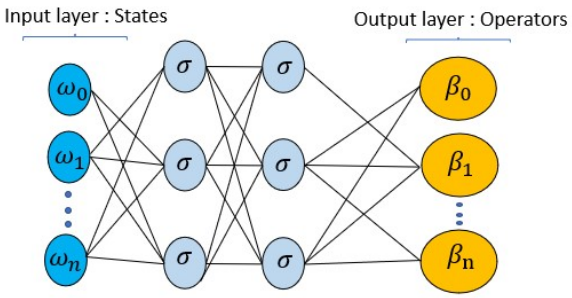


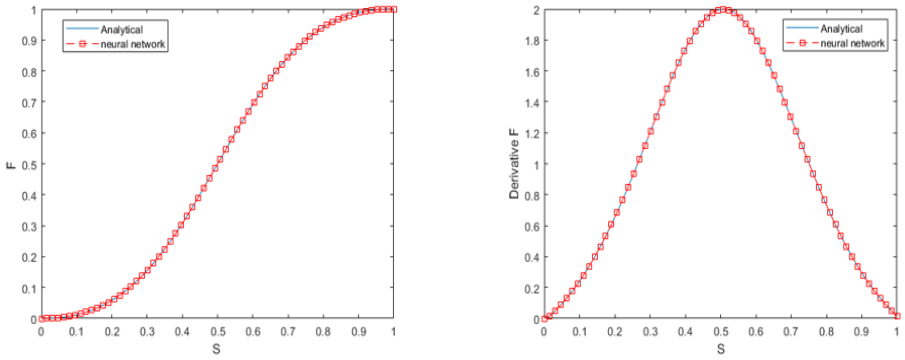
Figure 5.14: Neural network architecture for operator training

We start with simple training with the S-shape fractional flow for two-phase immiscible fluid and extend the training for more complicated operator space corresponding to the reduced order modeling of the compositional transport.

5.2.3. 1-D TRAINING

BUCKLEY-LEVERETT WITH NEURAL NETWORK IN JACOBIAN ASSEMBLY

In this particular test case, our objective is to train a neural network to approximate both the nonlinear operator of the partial differential equation (PDE) and its derivative. Instead of employing multi-linear interpolation in the operator space, we utilize the neural network to obtain the operator's value as well as its derivative. Figure 5.15 compares the neural network prediction of fractional flow versus the analytical solution.



(a) Fractional flow (b) Derivative of the Fractional flow

Figure 5.15: Comparison of neural network prediction of fractional flow versus analytical solution

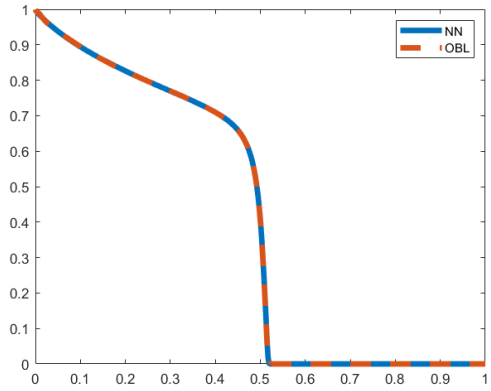


Figure 5.16: The solution of Buckley-Leverett is being compared to an assembly-embedded neural network.

Table 5.2 compares the total number of Newton iterations between interpolation during the Jacobian and assembling and assessing using the neural network. We can clearly see that the number of Newton iterations is higher if we call the neural network. Moreover, for higher dimension operator space while there are more components, training for all the partial derivatives is computationally costly. As a result, in the following part, we solely train neural networks for evaluating the supporting points of operator space, and we use multi-linear interpolation to obtain the values along with their derivative throughout the assembly.

5.2.4. TWO-STAGE TRAINING

A supervised learning strategy is used in our study to adjust the operator space of the aforementioned proxy model and increase its prediction performance. This is accom-

Operator evaluation	nonlinear iteration
Interpolation	323
NN	380

Table 5.2: Comparison of the number of Newton iterations during interpolation versus neural network evaluation

plished by embedding neural networks into the PDE's nonlinear operator and applying the forward solution in a loss function regularization term. Our method combines the distance between the proxy model's leading and trailing shocks with respect to the reference model and penalizes the loss function so that the neural network learns the operator space with the least shock misfit. This can be effectively achieved by using the Lebesgue integration instead of the Riemann integral in the loss function.

A neural network, f_θ is applied to the β and Λ operators of the PDE. A two-step transfer learning approach is used for training. First, neural networks are trained using labeled data, which consists of a $N_o \times N_o$ OBL mesh of data points with an input matrix $\mathbf{X}$ that corresponds to state variables pressure and composition. In the first step, the loss function is the mean squared error of the proxy model operator space's analytical approximation, $f(\omega)$, and the neural network's predictions, $f_\theta(\omega)$:

$$L_1 = \frac{1}{N_o^2} \sum_{i=1}^{N_o^2} (f^i(\omega) - f_\theta^i(\omega))^2. \tag{5.25}$$

The model weights are copied/transferred to a new neural network in the second step, and a fully implicit solver is included in the loss function. The solver use the anticipated operator space to find the pseudo-composition z_r. The quality of the resultant solution is then evaluated by computing the Lebesgue integral between $z_r(x)$ and $z_{ref}(x)$ over N_t time steps and adding it to the loss.

$$L_2 = \frac{1}{N_t N_f} \sum_{t=1}^{N_t} \sum_{z=1}^{N_z} \sqrt{(x_z^t - x_{zr}^t)^2}. \tag{5.26}$$

This factor in the loss function penalizes the neural network if the distance between the shocks of the reference and proxy models is high because Δx must approach zero for every given time-step.

$$\Delta x = -\lambda \left(\int \frac{\partial F_\theta}{\partial z_r} \partial t - \int \frac{\partial F}{\partial z_{ref}} \partial t \right) \tag{5.27}$$

The new loss function combines both parts: $L = L_1 + L_2$.

After addressing the proxy and computing the loss function, TensorFlow utilizes reverse-mode automatic differentiation to facilitate back-propagation and update the ensemble of weights and biases, denoted as θ in the literature (TF). This process effectively incorporates the analytical approximation of the operator space into the neural network. The training phase consists of 600 iterations, employing the Adam stochastic optimizer with

a fixed learning rate of 0.001. The training specifically focuses on a one-dimensional homogeneous reservoir with a constant permeability. Additionally, the training procedure is described in terms of the neural network architecture, optimizer, and activation function. The architecture mirrors that of [96] and [97]., comprising 8 hidden layers, each containing 20 neurons. The optimizer employs standard gradient descent with a consistent learning rate of 0.0001. The choice of a small learning rate is intentional as it is crucial for the operator space to preserve the characteristics of shocks in compliance with the velocity constraint of the method of characteristics. The neural network weights are initialized randomly using the Xavier/Glorot initialization technique. The activation function, denoted as σ, is represented by the hyperbolic tangent function (tanh), which introduces nonlinearity to the system.

PSEUDO-BINARY MODEL

In the study by Ganapathy and Voskov [94], the four-component gas vaporization problem is tackled using the multi-scale reconstruction of physics (MSRP) method. This approach is applied to handle the specific scenario involving injection gas and initial oil compositions, which are outlined in table 5.3. The reference solution to this problem is shown in figure 5.17. The injection gas consists of 97% of methane while the initial oil composition consists of a mixture of all four components. In the compositional space, a tie-line represents a line where the liquid and gas phases achieve thermodynamic equilibrium at a specific pressure and temperature. This equilibrium condition is determined by the equality of fugacities (f) for each component in both phases, serving as a stability criterion.

In a quaternary displacement, the solution is defined according to three key tie-lines, namely the initial, injection and cross-over tie-lines. A tie-line is a line in the compositional space where the liquid and gas phases are in thermodynamic equilibrium for a given pressure and temperature and is expressed by the stability condition by imposing that the fugacities f of each component in both phases must be equal.

In the context of gas injection processes, the compositional path traverses the two-phase zone by extending the initial and injection tie-lines, resulting in discontinuities in both the compositional and saturation profiles, commonly known as shocks. As a result, the compositional profile can be divided into three distinct zones based on their phase states: the single-phase vapor zone, the single-phase liquid zone, and the two-phase zone. These zones are delineated by the leading and trailing shocks. Downstream of the leading shock, the mixture exhibits a liquid phase with the initial composition conditions, whereas upstream of the trailing shock, the composition consists solely of component C_1. Consequently, the region between these two shocks contains both phases simultaneously. Within the two-phase zone, the compositional profile comprises tie-line and non-tie-line rarefactions that connect the tie-lines and follow a specific compositional path.

Since the solution of the compositional problem is characterized by the leading and trailing shocks, a reduced-order proxy model by [94] is used to partially construct part of the solution. This method is able to reduce a compositional problem with an arbitrary number of components to a pseudo-binary problem where the number of components is reduced to two and thus the number of equations is reduced to $2 \times n_b$. Their analytic

Table 5.3: Vaporizing gas drive

Component	C1	CO2	nC4	nC10
Initial	0.10	0.18	0.37	0.35
Injection	0.97	0.01	0.01	0.01
K-value	2.5	1.5	0.5	0.05

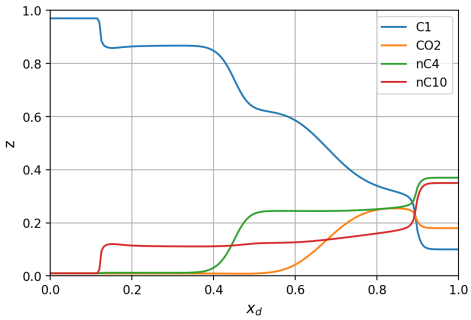


Figure 5.17: Composition (mole fraction) profile for a four-component vaporizing gas drive. The profile is characterized by the leading shock and trailing shock located at approximately $x_d = 0.9$ and $x_d = 0.12$. From left to right, they delineate three zones where single-phase vapor, both phases and single-phase liquid is present.

approach relies on the fact that the displacement path can be constructed according to key tie-lines of the system as the leading- and trailing shocks can only enter and leave the two-phase zone along tie-line extensions [98]. The first stage of the MSRP reconstruction is able to accurately reconstruct a part of the conservative solution as it manages to locate the leading and trailing shocks in space and time for an incompressible transport problem. Here their model is adapted to include flow as well as transport and used for an isothermal compressible gas vaporization problem. The location of the shocks for the injection component C_1 is reconstructed.

The pseudo-operator space for a given component is parameterized according to pseudo-composition z_r and pressure p. The lever rule is applied with fixed partitioning coefficients computed at initial and injection compositions and the resulting gas fraction and saturation are used to define the operator space for the initial and injection tie-lines $\beta^{ini/inj}$.

$$z_r = z_i \tag{5.28}$$

$$v = \frac{z_r - x_i}{y_i - x_i} \tag{5.29}$$

Subsequently, the pseudo-operator space $\beta_r(\omega)$ is established by constructing the convex hull from the combination of all key tie-lines (eq. 5.30). Within this resulting operator space, the transition from the initial to the injection tie-lines occurs at the intersection point in the nonlinear, two-phase zone.

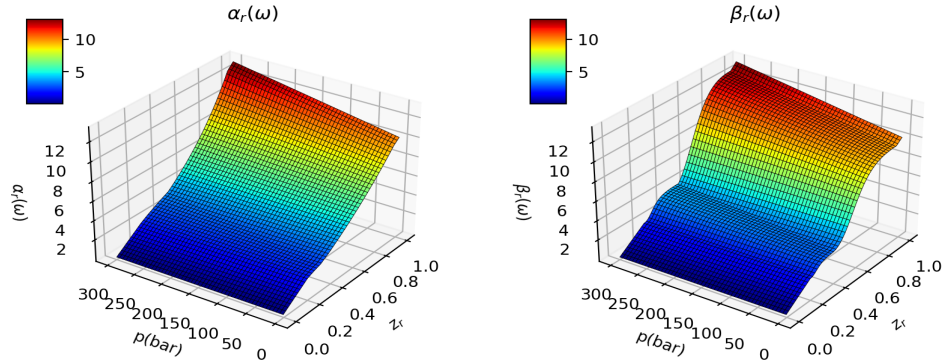
$$\beta_r(\omega) = conv(\beta_i^{ini} \cup \beta_i^{inj}) \tag{5.30}$$

Similarly, $\alpha_r(\omega)$ is defined as the product of the total density and pseudo-composition z_r.

$$\rho_{tot,r} = conv(\rho_{tot}^{ini} \cup \rho_{tot}^{inj}) \tag{5.31}$$

$$\alpha_r(\omega) = \rho_{tot,r} z_r \tag{5.32}$$

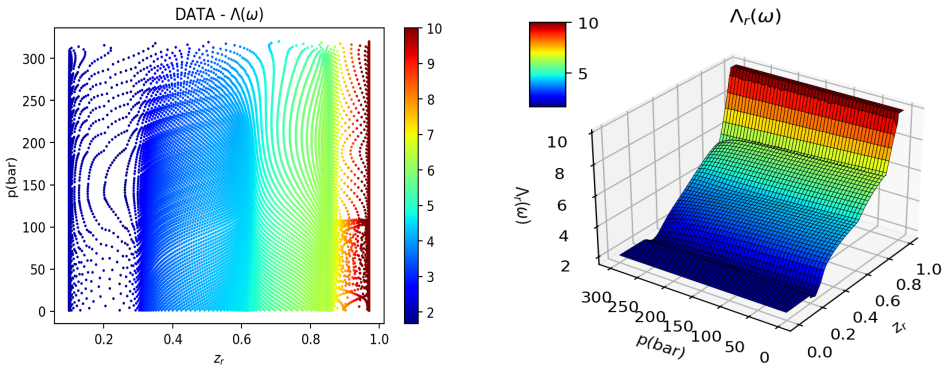
The analytical construction is illustrated for the flux operator in one dimension (incompressible) in figure 5.20 and in two dimensions (compressible) in figure 5.18(a). The pseudo-operator $\alpha_r(\omega)$ is shown in figure 5.18(b).



(a) Accumulation operator

(b) Flux operator.

Figure 5.18: Operator space of the binary proxy model.



(a) Collected data points from the reference simulation.

(b) Resulting interpolated operator space. In extremum, the total mobility is equal to 2 and 10 in the single-phase liquid and vapor zones.

Figure 5.19: Total mobility operator $\Lambda_r(\omega)$.

The total mobility operator is constructed by tabulating values from the reference solution based on the state variables. To ensure comprehensive coverage of the entire

parameter space, the reference simulation is conducted with varying bottom-hole pressure controls. Subsequently, the scattered data points are utilized to generate a structured OBL mesh through the application of a linear interpolant. Figure 5.19 illustrates both the collected data points and the resulting interpolation for the total mobility operator. The operator space of the total mobility operator $\Lambda(\omega)$ and the flux operator $\beta(\omega)$

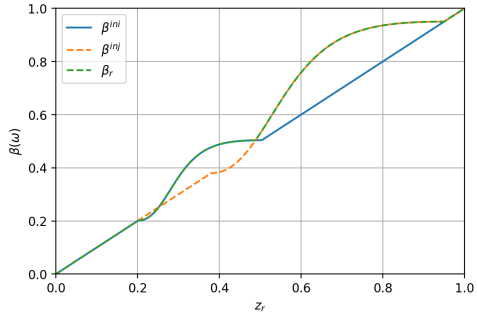
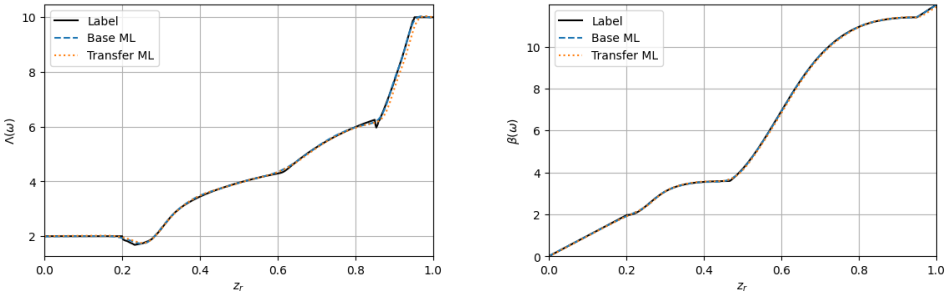


Figure 5.20: The convex hull of the union of the initial and injection tie-lines is taken to define the pseudo-operator space.

is adapted with our transfer training approach.

In the first step, the neural network is exclusively trained to approximate the OBL mesh with labeled data from the analytical construction. At this stage, it is found that the neural network has a smoothing effect on the operator space which can be seen in figure 5.21. This figure compares the operator space at base ML training with the labels. Additionally, due to the structured nature of the training data, it is found that the neural network is able to accurately approximate OBL grids of different densities.

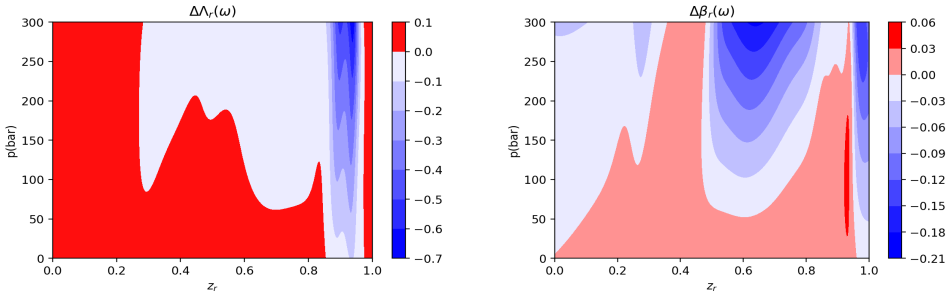


(a) Total mobility operator.

(b) Flux operator.

Figure 5.21: Comparison of the operator space before (base ML) and after transfer training (transfer ML) to the labels ($p = 180bar$).

In the second step, the estimated operator space is used within a fully implicit solver embedded in the loss function of the neural network as described previously. The solver utilizes the OBL approach as it interpolates required values from the vertices of the OBL



(a) Total mobility operator.

(b) Flux operator.

Figure 5.22: Difference of the operator space after training. The greatest changes in the operator space are concentrated in the area relevant to the trailing shock.

mesh. Training is carried out for a one-dimensional homogeneous reservoir containing 200 grid blocks where the permeability is equal to $100mD$ and the length of the reservoir is $200m$. During the training period, the bottom-hole pressures are fixed at the limits of the OBL mesh, namely $1bar$ and $300bar$. The Lebesgue integral of the leading and trailing shocks is computed at 10 discrete time steps ranging from $t = 1day$ to $10days$. The range at which the Lebesgue integral of misfit is calculated lies between $z_r = 0.90$ and 0.96 for the trailing shock and between $z_r = 0.11$ and 0.30 for the leading shock.

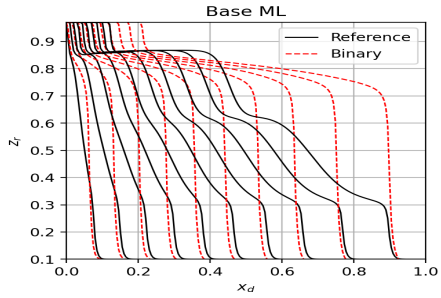


Figure 5.23: Forward solution of the binary model versus the reference solution, before transfer training (base ML).

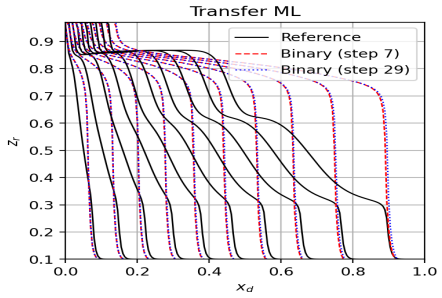


Figure 5.24: Forward solution of the binary model versus the reference solution, after transfer training (transfer ML).

Figure 5.25 presents the plot depicting the evolution of the loss function during the second step. Additionally, figure 5.26 illustrates the progression of the misfit for the trailing and leading shocks at each time step (ts) and training iteration. Notably, a global minimum is attained at training step 7, beyond which further improvements become marginal. The misfit of both the leading and trailing shocks decreases, indicating an enhancement in the accuracy of the proxy model compared to the conservative reference solution for n_c components. Particularly, the estimation of the trailing shocks exhibits significant improvement, as evident in figure 5.24, while the position of the leading shock

undergoes modest changes. Consequently, changes in the operator space, encompassing both the total mobility and flux operators, are most pronounced in the region pertinent to the trailing shock, characterized by high pressure and overall composition. A visual representation of the changes in the operator space before and after training can be observed in figure 5.22.

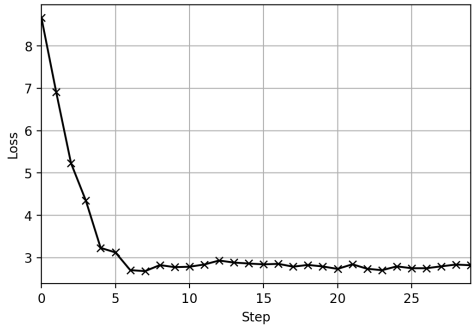


Figure 5.25: Evolution of the loss function.

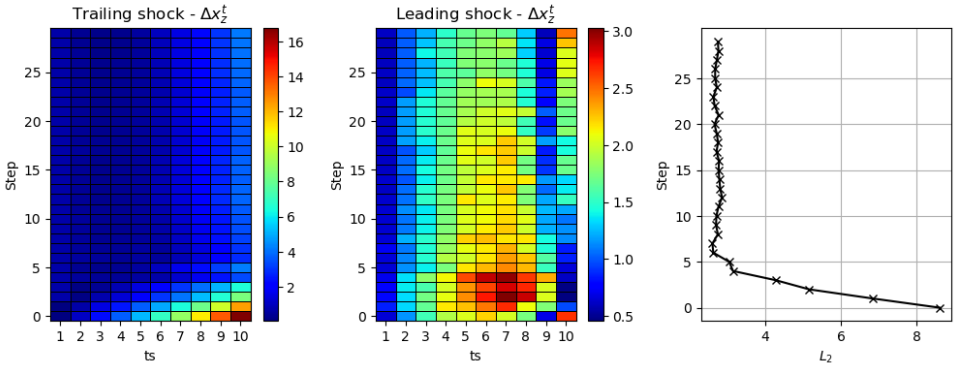


Figure 5.26: Evolution of the Lebesgue integral of the trailing and leading shocks that make up the L_2-term of the loss function.

5.2.5. HORIZONTAL LAYERS OF SPE10

The SPE10 geological model employs a trained model on two-dimensional heterogeneous layers. These models consist of (220×60) cells and represent the Tarbert formation (top 35 layers) and the Upper Ness formation (bottom 50 layers), depicting prograding near shore and fluvial environments [99]. The cell sizes are $\Delta x = 3m$, $\Delta y = 6m$, and $\Delta z = 0.6m$. A 5-spot well pattern is utilized, with an injection well situated in the middle and fixed bottom hole pressure of 300 bars injecting 97 % methane. Additionally, there are 4 production wells located at the corners, operating at 1 bar. The breakthrough times of initial and final breakthroughs are recorded for each model. These breakthroughs align with the leading and trailing shock breakthroughs at any of the four wells. Furthermore, the accuracy of predicting the shocks' positions in space and time is assessed by analyzing the phase state of each grid cell.

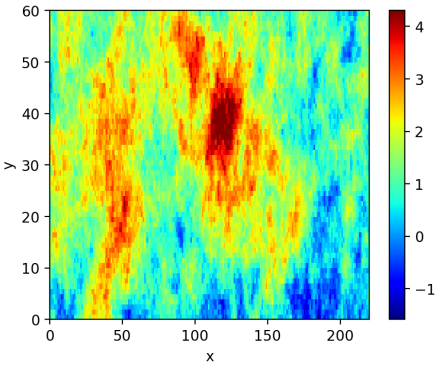
To assess the performance of each model, the error in phase-state classification of the generated compositional profile is quantified. The phase-state maps distinguish three distinct zones within the reservoir, representing single- and two-phase areas where gas (2), oil and gas (1), or oil (0) are present. The classification of each cell's state is based on the tie-line end points of the injection and initial tie-lines. The model's error is then measured as the ratio of misclassified grid blocks compared to the reference model, divided by the total number of grid blocks.

$$T(S_r(x,y,t) - S_{ref}(x,y,t)) = 1 \qquad for \qquad S_r - S_{ref} \neq 0 \tag{5.33}$$

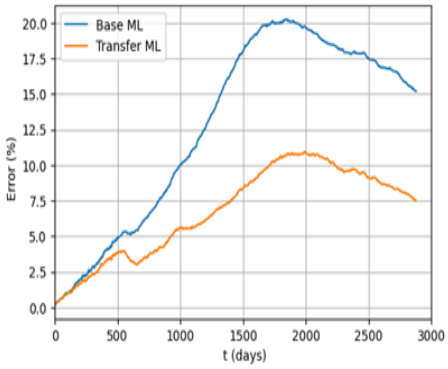
$$Error(t) = \frac{\sum_i^{n_b} T(S_{r,i} - S_{ref,i})}{n_b} \tag{5.34}$$

Figures 5.32(a) and 5.32(b) illustrate the distribution of the phases and the error of the proxy model before and after training for layer 7 of the SPE10 model. The boundary between the zones delineates the location of the leading and trailing shock within the reservoir and thus the error gives an approximation of how well the location of the shock is approximated in space and time. From the error maps, it is observed that the estimation of the location of the trailing shock is visibly improved for the trained model whereas the estimation of the leading shock remains the same or is slightly better. The evolution of the error as the function of time is reported in figure 5.27(b) and the permeability of layer 7 is shown in figure 5.27(a). The error increases as the displacement front grows and the region of displacement increases.

The performance before and after training of the proxy model is compared for the top 15 layers of the SPE10 model. The breakthrough times of the leading and trailing shocks are reported per layer in figures 5.28(a) and 5.28(b) together with the corresponding error of the phase-state classification. It is found that the trained model consistently outperforms the model trained at the first stage in terms of error and breakthrough time of the trailing shock while the breakthrough time of the leading shock effectively remains the same as the base ML model already makes a good estimation. The error of the trained model at breakthrough remains below 7.5%. The average difference in breakthrough time for the trained and untrained models with respect to the reference model is $293 days$ versus $570 days$ for the trailing shocks and $15 days$ versus $16 days$ for the leading shock.



(a) *Log*10 of the permeability distribution of layer 7 of the SPE10 model.

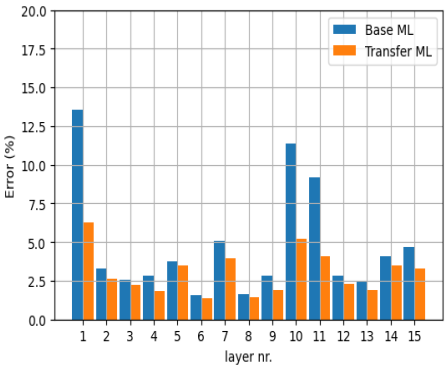


(b) Error of the base ML and transfer ML models.

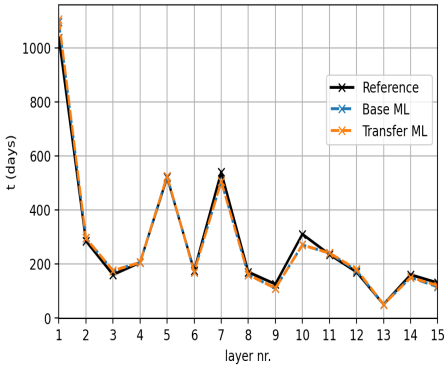
Figure 5.27: Evolution of the error for layer 7 of the SPE10 model before (base ML) and after transfer training (transfer ML).

5

Figures 5.30 and 5.31 present the reported first and last breakthrough times per layer for the bottom 15 layers of the SPE10 model. The analysis reveals that the estimation of the leading shock remains largely unchanged, whereas there is a consistent improvement in estimating the trailing shock. It is important to note that when production wells are situated outside sand channels, the breakthrough time for both the trailing and leading shocks becomes significantly delayed. As a result, breakthrough times are reported in logarithmic form to better capture the variations.

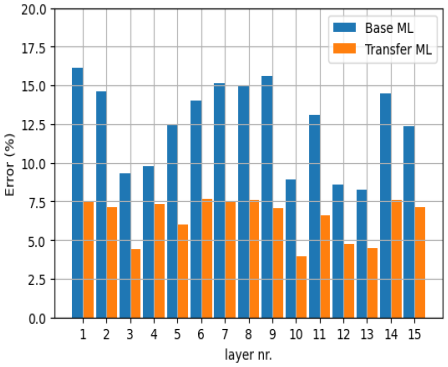


(a) Error of the phase-state classification at the breakthrough of the leading shock.

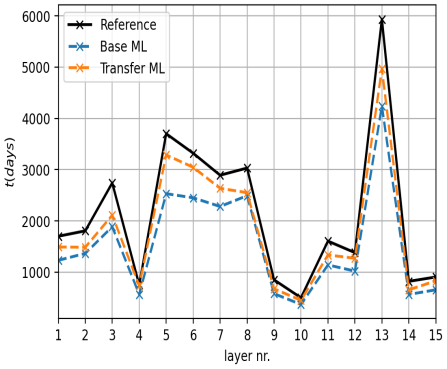


(b) Breakthrough times of the trailing shock at one of the wells.

Figure 5.28: The breakthrough time of the leading shock at one of the wells and the corresponding error of the phase-state maps is assessed for the top 15 layers of the SPE10 model.

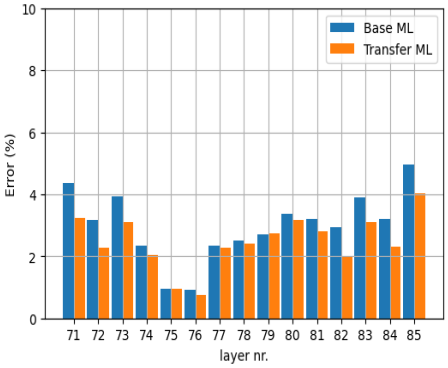


(a) Error of the phase-state classification at the breakthrough of the trailing shock.

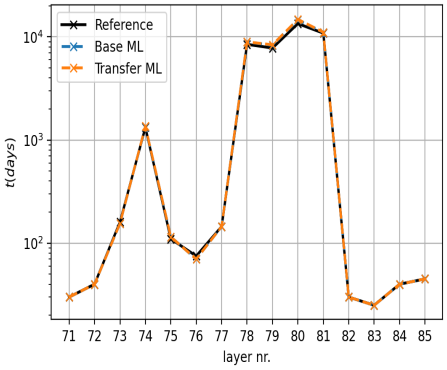


(b) Breakthrough times of the trailing shock at one of the wells.

Figure 5.29: The breakthrough time of the trailing shock at one of the wells and the corresponding error of the phase-state maps is assessed for the bottom 15 layers of the SPE10 model.



(a) Error of the phase-state classification at the breakthrough of the trailing shock.

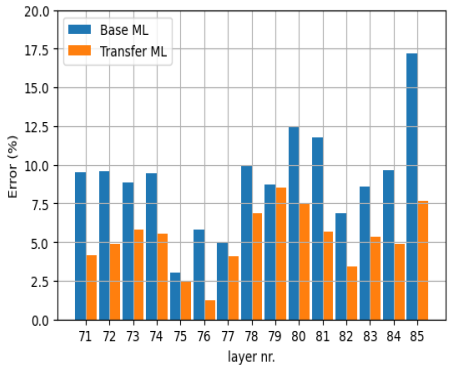


(b) Breakthrough times of the trailing shock at one of the wells.

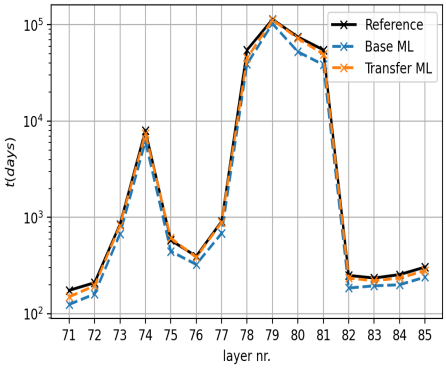
Figure 5.30: The breakthrough time of the trailing shock at one of the wells and the corresponding error of the phase-state maps is assessed for the bottom 15 layers of the SPE10 model.

5.3. DISCUSSION AND CONCLUSION

In this chapter, We explore deep learning for the simulation of CCUS. In the first part of the chapter, we investigated the application of a Physics-informed neural network (PINNs) for two-phase fluid in porous media. A Physics-Informed Neural Network (PINN) is a machine learning approach that combines neural networks with physical laws to solve complex scientific and engineering problems, ensuring predictions adhere to the governing physics. It minimizes a loss function that includes data fitting from the boundary and initial conditions and PDE constraint terms during training. Standard PINNs

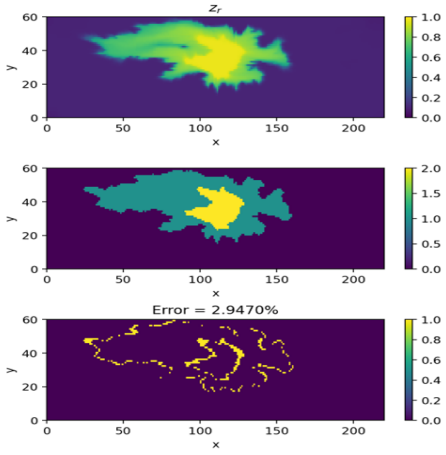


(a) Error of the phase-state classification at the breakthrough of the trailing shock.

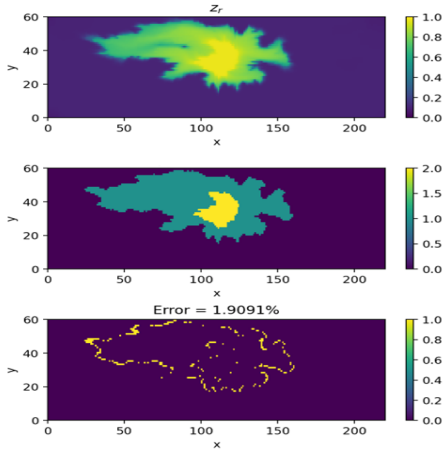


(b) Breakthrough times of the trailing shock at one of the wells.

Figure 5.31: The breakthrough time of the trailing shock at one of the wells and the corresponding error of the phase-state maps is assessed for the bottom 15 layers of the SPE10 model.



(a) Base ML proxy model.



(b) Transfer ML proxy model.

Figure 5.32: Distribution of the phase-states and the corresponding error at t = 150 days for layer 7 of the SPE10 model. The phase-states correspond to single- and two-phase zones where gas (2), oil and gas (1), or oil (0) occupy the grid block. The transfer ML model indicates a marked improvement in the estimation of the trailing shock.

has limitation in capturing the transport problem with hyperbolic characters in porous media. In our single timestep analysis, we can clearly see as a proof of concept that when training for bigger timesteps, the loss function increases. That motivates us to use a sequential training scheme with backward compatibility as an alternative to standard training to overcome the standard training limitation. We proposed dynamic time-

stepping instead of uniform time-stepping for training. Our training method starts with a small step. If the loss goes below a certain point, we make the next step bigger by a factor of β. If not, we make the step smaller by dividing it by β. This adaptive approach helps us find the right answer faster while saving computing time and costs. Additionally, we applied the sequential training approach to the challenging miscible binary compositional test case, which includes complex kink points that appear due to the phase transition. Kink points pose a greater challenge for the optimizer since these points are not differentiable. To overcome this, we introduced a sigmoid function to continuously approximate the fractional flow function. Our findings indicate that the seqPINNs method performs better than the conventional approach in terms of accurately capturing solutions.

In the second part of this chapter, we explored the capability of the neural network for training operators. In this chapter instead of using PINNs to solve directly the PDE, we use neural network to obtain the supporting points for OBL. Next, we run DARTS to get the full solution to the problem. The advantage of this approach is that we gain benefits from robust and fast DARTS simulation instead of using PINNs as a solver. We started with training a convective flux operator in the form of fractional flow for two-phase immiscible transport. We also made a test case that we trained the derivatives and we evaluated the derivatives and operators based on the neural network instead of interpolation. The result indicated that using a neural network instead of interpolation increases a Newton iteration. Afterward, we focused only on obtaining supporting points by neural network and used the classic OBL approach (interpolation) to get the values of the operator and its derivatives in Jacobian assembly. We further explored the neural network for training more sophisticated operator of a proxy model for compositional problem based on Multiscale Reconstruction in Physics. We employ a physics-based deep learning strategy in which we apply neural networks to the operator space of the proxy model and integrate the Lebesgue integration of the forward solution into the loss function. This results in enhanced breakthrough time estimations, benefiting both 1-D scenarios and the complex fifteen-layer SPE10 model. This approach can be expanded to create a versatile physical proxy model with reduced degrees of freedom, trained using data points generated by a high-fidelity physical model or derived from real-world observations.

6

RECAPITULATION AND CONCLUDING REMARKS

6.1. COUPLED WELLBORE-RESERVOIR MODELLING

Wellbore operations play a pivotal role in the management of diverse subsurface geo-energy applications, including energy storage, geothermal energy production, CO_2 sequestration, oil and gas extraction, wastewater disposal, and thermal recovery techniques. In recent times, advanced well technologies, like multi-lateral wells equipped with sophisticated inflow control valves, have been employed to enhance both the economic efficiency of field operations and operational reliability. Furthermore, different wellbore designs (such as co-axial wellbore) are utilized to optimize the extraction of subsurface heat in cutting-edge geothermal applications. To successfully design, predict, and optimize the various processes essential for energy production, precise fully-coupled models for thermal multi-phase flow within both the reservoir and boreholes are imperative

We have developed a new computational framework that can simulate Thermal-Hydraulic-Chemical (THC) multiphase multi-component fully-coupled flow in the wellbore and the reservoir. The implementation is based on an operator-based linearization (OBL) method used in the Delft Advanced Research Terra Simulator (DARTS). In the OBL approach, the governing equations are represented in operator form which significantly simplifies the solution of highly nonlinear governing equations with complex physics. In the proposed framework, the OBL technique is extended to both the governing equations of the reservoir and the wellbore. During the simulation, multilinear interpolation is used to interpolate the corresponding values and derivatives of operators, which reduces the computational cost related to linearization. Our simulation model is built on a general unstructured grid framework, in which the wellbore is divided into segments that follow a similar scheme as the finite-volume discretization used for the reservoir. The total velocity serves as an additional nonlinear unknown that is constrained by the momentum equation, allowing for writing a suitable momentum equation for a well-

bore.

First, we verified the accuracy of the ms-well model by comparing a solution for thermal, two-phase immiscible physics with the Automatic Differentiation General Purpose Research Simulator (ADGPRS). Our test produces comparable results to an accurate ms-well model in both simulation frameworks. The performance comparison of DARTS and ADGPRS simulation shows that DARTS allows for less CPU time and nonlinear iterations due to the different nonlinear formulations and OBL approach. We further tested the framework for more complex physics considering thermal effects. We assess the engine's performance for geothermal physics using a heterogeneous reservoir and compare the results to the conventional method with a pseudo-porous medium approach. For cases when transient effects can be ignored, a pseudo-porous medium approach produces a solution comparable to the accurate well model with a lower computational cost. Besides, we test the developed framework in modeling the calcite dissolution in the near-well region on unstructured mesh and the results show that the framework could capture the dissolution effect near the wellbore. Accurate discretization of the wellbore allows us to model sophisticated well technologies such as co-axial wellbores. We made a test case for a single closed loop wellbore and ran multiple numerical experiments and sensitivity analyses on various factors that affect heat extraction from the co-axial wellbore.

6.2. TRUST-REGION NONLINEAR SOLVER EMBEDDED IN OBL FRAMEWORK

Reducing carbon emissions has become a top global priority in the effort to combat climate change. Carbon dioxide capture, utilization, and storage (CCUS) stands out as a promising approach to limit the release of greenhouse gases into the atmosphere. CCUS encompasses the entire process of capturing carbon dioxide and its related compounds from emission sources, compressing it, transporting it, and utilizing it for various purposes. These uses include permanent storage in deep underground geological formations and enhancing the recovery of hydrocarbons in existing oil fields. To effectively model CCUS in subsurface reservoirs with complex, non-uniform structures, it is essential to employ a comprehensive model that accounts for multiphase compositional flow and transport.

The discretized governing equations are highly nonlinear, and Newton's technique is frequently used to solve them. Newton's solution technique does not ensure convergence and is extremely dependent on the timestep choice. In this work, we investigate the nature of nonlinearities in CCUS simulations and suggest solutions to a general compositional problem. We present an advanced nonlinear solver based on a trust-region technique aimed to solve multiphase multi-component transport problems. The trust region solver is based on the analysis of multi-dimensional tables connected to parameterized highly nonlinear convection operators. These operators are associated with the governing equations and are built for a newly introduced Operator-Based Linearization approach. The inflection line and kinks in the parameter space determine the delineation of the trust-region zones. According to our nonlinear study of convective operators for binary and ternary systems, each component has its inflection line within the two-phase region. In addition, kink lines appear when phase boundaries are crossed.

These boundaries could change in the parameter space of the problems based on the direction of the Newton trajectory.

We track the nonlinear trajectory and segment the parameter space of the problem into a set of trust regions where the hyperbolic operators keep their second-order behavior (i.e., they remain either convex or concave). We approximate these trust regions in the solution process by detecting the boundary of convex regions via analysis of the directional derivative. By drawing multiple trial trajectories on binary and ternary diagrams we observe that our algorithm can detect these boundaries correctly. Moreover, it is less computationally expensive since we do not compute the entire hessian in our technique and instead compute the directional derivative while tracking the nonlinear update. After detecting all the boundaries along the nonlinear trajectory, the proposed nonlinear solver locally constrains the update of the overall compositions across the boundaries of these regions. We tested our nonlinear solver for several reservoir models starting from the single cell to a fully 3D heterogeneous model. Our numerical results show that the trust-region solver avoids overshoots in the nonlinear update which lead to superior convergence in comparison to conventional nonlinear solvers.

6.3. PHYSICS-INFORMED NEURAL NETWORKS FOR CCUS

Machine learning techniques, particularly deep learning, are gaining prominence in the fields of computer science and engineering. Notably, physics-informed neural networks (PINNs) are being employed to address problems where knowledge of engineering conservation equations and constitutive closure relationships is available, even in the absence of labeled data. By constructing neural networks with multiple hidden layers and nonlinear activation functions, complex nonlinear solutions can be effectively approximated. Consequently, PINNs have found utility in a wide range of applications governed by differential equations, including the Euler equation, gas dynamics, water dynamics, and chemical kinetics. PINNs have showcased their versatility in various applications, encompassing data assimilation, parameter identification (solving inverse problems), and uncertainty quantification.

The application of PINNs has been extensively explored in the domain of subsurface flow and transport. For instance, two-phase immiscible transport in porous media, typically described by the nonlinear first-order hyperbolic PDE known as the Buckley-Leverett equation, has been modeled using standard PINNs [45]. However, it was observed that PINNs struggle to find a solution in cases involving steep saturation fronts with nonconvex flux functions. A potential remedy involved the introduction of an artificial diffusion term into the original conservation equation, enabling neural networks to approximate the true solution. Further innovative solutions have been proposed, such as physics-informed attention-based neural networks (PIANNs) [46], which blend recurrent neural networks and attention mechanisms, and methods that embed entropy and velocity constraints into the neural network residual [47].

We investigated the application of a Physics-informed neural network (PINNs) for two-phase fluid in porous media. While standard PINNs have difficulties solving hyperbolic PDEs with non-convex flux functions, we suggested a sequential training scheme as an alternative. We can overcome this obstacle by training for shorter time intervals and marching in time dynamically. The sequential training scheme begins with a small

time step, and if the loss function decreases to a predefined tolerance, we multiply the next time step by a constant parameter, denoted as β. However, if the loss function fails to decrease to the tolerance within the given number of epochs, we divide the time interval by β. This adaptive approach allows for efficient convergence to an accurate solution while avoiding unnecessary computations and minimizing computational costs. Furthermore, we extended the sequential training strategy to the miscible binary compositional test case, where there are kink points that made it even more difficult for the optimizer to find the solution. To address this, we proposed a sigmoid function to continuously approximate the fractional flow function.

Within our study, we conducted several numerical experiments to evaluate the effectiveness of our proposed sequential training scheme for simulating CO_2 utilization and storage in subsurface reservoirs with complex heterogeneous structures. Firstly, we designed a single-time step numerical test case to demonstrate that increasing the time step leads to an increase in the final loss function. This observation is attributed to the heightened non-linearity of the residual as the time step is increased, resulting in a more challenging optimization problem that negatively impacts the performance of the optimizer. Subsequently, we conducted a 1D full simulation test case to compare the standard and sequential training schemes for both immiscible and miscible test cases. Our results show that the proposed seq-PINNs approach outperforms the standard approach in terms of solution capture accuracy.

Furthermore, we implemented the sequential training method on more challenging miscible binary compositional test cases characterized by intricate kink points arising from phase transitions in the flux function. Kink points present a heightened challenge for optimization as they lack differentiability. To address this issue, we introduced a sigmoid function to provide a continuous approximation of the fractional flow function. Our results demonstrate that the seq-PINNs approach outperforms the conventional method in accurately representing solutions.

6.4. APPLICATION OF DEEP NEURAL NETWORKS FOR PHYSICS-BASED PROXY MODELLING

We delved into the potential of neural networks for training operators. Instead of using Physics-Informed Neural Networks (PINNs) to directly tackle Partial Differential Equations (PDEs), we harnessed neural networks to procure key support points for Operator-Based Learning (OBL). Subsequently, we used (DARTS) to obtain the solution. This approach has an advantage because it uses the fast and robust DARTS simulation instead of PINNs as solvers. We started by training a convective flux operator, adopting the form of fractional flow, particularly relevant for two-phase immiscible transport. A noteworthy test case involved training derivatives and subsequently evaluating these derivatives and operators based on the neural network, eliminating the need for interpolation. The results illustrated that replacing interpolation with a neural network led to an increased number of Newton iterations. We then concentrated solely on obtaining support points through the neural network, utilizing the conventional OBL approach (interpolation) to acquire operator values and their derivatives during Jacobian assembly.

Compositional challenges emerge from the interactions of various components, de-

manding a set of equations that matches the number of components for every grid cell. This can lead to significant computational expenses, especially when dealing with a fine-scale high-fidelity model. To address this, we have employed a proxy model constructed using multiscale techniques in physics, and we trained the operator within this framework. This strategy incorporates physics-based deep learning, applying neural networks to the operator space of the proxy model and integrating the Lebesgue integration of the forward solution into the loss function. The outcome was more accurate breakthrough time estimations, benefiting both 1-D scenarios and the intricate fifteen-layer SPE10 model. The applicability of this approach extends to the creation of a versatile physical proxy model with reduced degrees of freedom, trained using data points generated by a high-fidelity physical model or derived from real-world observations.

6.5. FUTURE PERSPECTIVES

6.5.1. COUPLED-WELLBORE AND RESERVOIR

Our study has identified several promising areas for future research. One potential direction is to extend the drift-flux model for multiphase flow in the wellbore into an operator form and integrate more complex chemical interactions in both the wellbore and reservoir. Another promising area is to investigate the development of an iterative linear solver for a decoupled velocity engine on both CPU and GPU platforms.

The nonlinear nature of coupled wellbore and reservoir simulations with complex physical models can cause the nonlinear solver to struggle and slow down the convergence process. Therefore, developing a nonlinear solver for this framework is an important future direction of our research. Extending the Trust-region method, which is integrated into the operator-based linearization framework, to include both the coupled wellbore and reservoir simulations would be a promising direction to pursue.

Furthermore, decoupled velocity formulation separates the momentum equation from mass and energy, as well as the well and reservoir regions, providing a promising solution for developing a local nonlinear solver that can identify areas with convergence issues and resolve them. Additionally, we aim to extend the model to include more advanced well-network topologies, taking into account surface capabilities, chokes, and valves.

6.5.2. ADAPTIVE NONLINEAR SOLVER IN DARTS

A promising future research direction could be to reduce the computational overhead related to locating the trust-region boundaries and carrying out the chopping. While we showed that directional derivative is cheaper than full Hessian assembly, there is still additional overhead since the detection of special points (kinks and inflection) are happening on the fly during simulation. The major cause of the overhead is due to the tracking of the nonlinear trajectory in nonlinear operator space.

One possibility is that once the special points have been identified, we save the value to avoid recalculation for the next iteration. Another possible future project would be to combine the TR solver with another type of solver. Based on our single-cell analysis, we can see that once the solution is in a single-phase region, conventional Newton solvers work, so we can switch between different solvers adaptively if we observe non-

linear solvers struggling.

6.5.3. EXTENSION OF PINNS TO MORE COMPONENT SYSTEMS

Our study identifies several promising directions for future research. Firstly, an important avenue for further exploration would be to extend the neural network to an arbitrary number of components, thereby enhancing the model's capacity to simulate complex multiphase compositional flows. Secondly, another possible future project would be to extend the model to higher dimensions and predict the solution in 2D and 3D space with heterogeneous reservoir structures.

NOMENCLATURE

Physical Symbols

M	accumulation term for mass
F	flux term for mass
q	source/sink term for mass
ρ_p	phase density
n_p	number of fluid phases
n_c	number of components
n_i	exponent for phase relative permeability
z_c	component overall molar fraction
x_{cj}	molar fraction of component c in phase p
s_j	phase saturation
S_{gr}	residual gas saturation
S_{or}	residual oil saturation
S_{wc}	connate water saturation
ϕ	effective rock porosity
ϕ_0	initial rock porosity
c_r	rock compressibility
p_{ref}	reference pressure
$\boldsymbol{u_j}$	phase velocity
$\boldsymbol{v_t}$	total velocity
$\mathbf{K}$	full permeability tensor
k_{rj}	phase relative permeability
k_{ro}	oil relative permeability
k_{rw}	water relative permeability
μ_j	phase viscosity
p_j	phase pressure
p_c	capillary pressure
p_d	capillary entry pressure
p_n	pressure for non-wetting phase
p_w	pressure for wetting phase
$\boldsymbol{\gamma}_p$	phase specific weight
D	vertical depth vector (up-down oriented)
D_{cj}	diffusive coefficient
ρ_t	total fluid density
r	rate for kinetic reaction

σ_{ca}	water/gas surface tension
ψ_c	fugacity coefficient of the gas phase
h_c	Henry's constant
κ_c	activity coefficient
K_c	phase-equilibrium constant of component

Linearization Operators

$\alpha(\omega)$	mass accumulation operator
$\alpha_{ef}(\omega)$	phase energy accumulation operator
$\alpha_{er}(\omega)$	rock energy accumulation operator
$\beta(\omega)$	mass flux operator
$\beta_e(\omega)$	energy flux operator
$\gamma(\omega)$	mass diffusion operator
$\delta(\omega)$	reaction operator
$\delta_e f(\omega)$	phase thermal conduction operator
$\chi(\omega)$	mass gradient operator
$\zeta_p^{vol}(\omega)$	volumetric well rate operator
$\zeta_p^{vol}(\omega)$	mass well rate operator
$\delta_p(\omega)$	phase density operator
$\xi(\omega)$	capillarity operator

Other Symbols

Γ^l	fluid transmissibility
Γ_T	rock thermal transmissibility
$\Phi_{p,ij}$	potential difference of phase p between block i and j

REFERENCES

[1] H. Ritchie and M. Roser, *Co_2 and greenhouse gas emissions,* Our World in Data (2020), https://ourworldindata.org/co2-and-other-greenhouse-gas-emissions.

[2] J. W. Lund and T. L. Boyd, *Direct utilization of geothermal energy 2015 worldwide review,* Geothermics **60**, 66 (2016).

[3] P. Vardon, D. Bruhn, A. Steiginga, B. Cox, H. Abels, A. Barnhoorn, G. Drijkoningen, E. Slob, and K. Wapenaar, *A geothermal well doublet for research and heat supply of the tu delft campus,* (2020), arXiv:2003.11826 [physics.geo-ph] .

[4] R. Bertani, *Geothermal power generation in the world 2010–2014 update report,* Geothermics **60**, 31 (2016).

[5] J. Limberger, T. Boxem, M. Pluymaekers, D. Bruhn, A. Manzella, P. Calcagno, F. Beekman, S. Cloetingh, and J.-D. van Wees, *Geothermal energy in deep aquifers: A global assessment of the resource base for direct heat utilization,* Renewable and Sustainable Energy Reviews **82**, 961 (2018).

[6] IEA, *Ccus in clean energy transitions,* International Energy Agency (2020), https://www.iea.org/reports/ccus-in-clean-energy-transitions.

[7] M. Todd, P. O'dell, and G. Hirasaki, *Methods for increased accuracy in numerical reservoir simulators,* Society of Petroleum Engineers Journal **12**, 515 (1972).

[8] A. Spillette, J. Hillestad, and H. Stone, *A high-stability sequential solution approach to reservoir simulation,* in *Fall Meeting of the Society of Petroleum Engineers of AIME* (Society of Petroleum Engineers, 1973).

[9] G. Thomas and D. Thurnau, *Reservoir simulation using an adaptive implicit method,* Society of Petroleum Engineers Journal **23**, 759 (1983).

[10] K. Aziz, *Petroleum reservoir simulation,* Applied Science Publishers **476** (1979).

[11] K. H. Coats, L. Thomas, and R. Pierson, *Compositional and black oil reservoir simulation,* in *SPE Reservoir Simulation Symposium* (Society of Petroleum Engineers, 1995).

[12] D. Collins, L. Nghiem, Y.-K. Li, and J. Grabonstotter, *An Efficient Approach to Adaptive- Implicit Compositional Simulation With an Equation of State,* SPE Reservoir Engineering **7**, 259 (1992).

[13] D. Voskov and H. Tchelepi, *Comparison of nonlinear formulations for two-phase multi-component eos based simulation,* Journal of Petroleum Science and Engineering **82-83**, 101 (2012).

[14] D. V. Voskov, *Operator-based linearization approach for modeling of multiphase multi-component flow in porous media,* Journal of Computational Physics **337**, 275 (2017).

[15] A. Iranshahr, D. V. Voskov, and H. A. Tchelepi, *A negative-flash tie-simplex approach for multiphase reservoir simulation,* SPE Journal **18**, 1140 – 1149 (2013).

[16] A. Iranshahr, D. V. Voskov, and H. A. Tchelepi, *Tie-simplex based compositional space parameterization: Continuity and generalization to multiphase systems,* AIChE Journal **59**, 1684 – 1701 (2013).

[17] DARTS, *Open Delft Advanced Research Terra Simulator,* GitLab repository (2023).

[18] X. Lyu, D. Voskov, J. Tang, and W. R. Rossen, *Simulation of foam enhanced-oil-recovery processes using operator-based linearization approach,* SPE Journal **26**, 2287 – 2304 (2021).

[19] X. Lyu, M. Khait, and D. Voskov, *Operator-based linearization approach for modeling of multiphase flow with buoyancy and capillarity,* SPE Journal **26**, 1858 – 1878 (2021).

[20] M. Khait and D. Voskov, *Operator-based linearization for efficient modeling of geothermal processes,* Geothermics **74**, 7 (2018).

[21] Y. Wang, D. Voskov, M. Khait, and D. Bruhn, *An efficient numerical simulator for geothermal simulation: A benchmark study,* Applied Energy **264**, 114693 (2020).

[22] K. Kala and D. Voskov, *Element balance formulation in reactive compositional flow and transport with parameterization technique,* Computational Geosciences **24**, 609 – 624 (2020), cited by: 12; All Open Access, Green Open Access, Hybrid Gold Open Access.

[23] X. Lyu, D. Voskov, and W. R. Rossen, *Numerical investigations of foam-assisted co2 storage in saline aquifers,* International Journal of Greenhouse Gas Control **108**, 103314 (2021).

[24] *Nonlinear solver based on trust region approximation for co2 utilization and storage in subsurface reservoir,* Geoenergy Science and Engineering **225**, 211698 (2023).

[25] R. Younis, *Modern advances in software and solution algorithms for reservoir simulation*, Ph.D. thesis, Stanford University (2011).

[26] J. Jiang and H. Pan, *Dissipation-Based Nonlinear Solver for Fully Implicit Compositional Simulation,* SPE Journal , 1 (2022), https://onepetro.org/SJ/article-pdf/doi/10.2118/209233-PA/2661304/spe-209233-pa.pdf .

[27] P. Jenny, H. A. Tchelepi, and S. H. Lee, *Unconditionally convergent nonlinear solver for hyperbolic conservation laws with s-shaped flux functions,* Journal of Computational Physics **228**, 7497 (2009).

[28] X. Wang and H. Tchelepi, *Trust-region newton solver for multiphase flow and transport in heterogeneous porous media,* (2013) p. 358 – 377.

[29] B. Li and H. A. Tchelepi, *Unconditionally convergent nonlinear solver for multiphase flow in porous media under viscous force, buoyancy, and capillarity,* (2014) p. 404 – 411, cited by: 10; All Open Access, Gold Open Access.

[30] O. Møyner, *Nonlinear solver for three-phase transport problems based on approximate trust regions,* Computational Geosciences **21**, 999 – 1021 (2017), cited by: 11.

[31] D. Voskov and H. Tchelepi, *Compositional space parametrization for miscible displacement simulation,* Transport in Porous Media **75**, 111 – 128 (2008), cited by: 45.

[32] O. Khebzegga, A. Iranshahr, and H. Tchelepi, *A nonlinear solver with phase boundary detection for compositional reservoir simulation,* Transport in Porous Media **137**, 707 – 737 (2021).

[33] Y. Lecun, Y. Bengio, and G. Hinton, *Deep learning,* Nature **521**, 436 – 444 (2015).

[34] M. Raissi, P. Perdikaris, and G. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,* Journal of Computational Physics **378**, 686 – 707 (2019).

[35] A. D. Jagtap, Z. Mao, N. Adams, and G. E. Karniadakis, *Physics-informed neural networks for inverse problems in supersonic flows,* Journal of Computational Physics **466**, 111402 (2022).

[36] A. D. Jagtap, E. Kharazmi, and G. E. Karniadakis, *Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems,* Computer Methods in Applied Mechanics and Engineering **365**, 113028 (2020).

[37] Q. Lou, X. Meng, and G. E. Karniadakis, *Physics-informed neural networks for solving forward and inverse flow problems via the boltzmann-bgk formulation,* Journal of Computational Physics **447** (2021), 10.1016/j.jcp.2021.110676.

[38] M. De Florio, E. Schiassi, B. D. Ganapol, and R. Furfaro, *Physics-informed neural networks for rarefied-gas dynamics: Thermal creep flow in the Bhatnagar–Gross–Krook approximation,* Physics of Fluids **33** (2021), 10.1063/5.0046181.

[39] A. D. Jagtap, D. Mitsotakis, and G. E. Karniadakis, *Deep learning of inverse water waves problems using multi-fidelity data: Application to serre–green–naghdi equations,* Ocean Engineering **248** (2022), 10.1016/j.oceaneng.2022.110775.

[40] W. Ji, W. Qiu, Z. Shi, S. Pan, and S. Deng, *Stiff-pinn: Physics-informed neural network for stiff chemical kinetics,* Journal of Physical Chemistry A **125**, 8098 – 8106 (2021).

[41] S. Kim, W. Ji, S. Deng, Y. Ma, and C. Rackauckas, *Stiff neural ordinary differential equations,* Chaos: An Interdisciplinary Journal of Nonlinear Science **31** (2021), 10.1063/5.0060697.

[42] L. Yang, X. Meng, and G. E. Karniadakis, *B-pinns: Bayesian physics-informed neural networks for forward and inverse pde problems with noisy data,* Journal of Computational Physics **425** (2021), 10.1016/j.jcp.2020.109913.

[43] R. Tipireddy, D. A. Barajas-Solano, and A. M. Tartakovsky, *Conditional karhunen-loève expansion for uncertainty quantification and active learning in partial differential equation models,* Journal of Computational Physics **418**, 109604 (2020).

[44] Z. Mao, A. D. Jagtap, and G. E. Karniadakis, *Physics-informed neural networks for high-speed flows,* Computer Methods in Applied Mechanics and Engineering **360**, 112789 (2020).

[45] O. Fuks and H. A. Tchelepi, *Limitations of physics informed machine learning for nonlinear two-phase transport in porous media,* Journal of Machine Learning for Modeling and Computing **1**, 19 (2020).

[46] R. Rodriguez-Torrado, P. Ruiz, L. Cueto-Felgueroso, M. C. Green, T. Friesen, S. Matringe, and J. Togelius, *Physics-informed attention-based neural network for hyperbolic partial differential equations: application to the buckley–leverett problem,* Scientific Reports **12** (2022), 10.1038/s41598-022-11058-2.

[47] C. G. Fraces and H. Tchelepi, *Physics informed deep learning for flow and transport in porous media,* (2021).

[48] K.-T. Lim, *A new approach for residual and jacobian arrays construction in reservoir simulators,* (1995), 10.2118/16976-PA.

[49] T. Hibiki and M. Ishii, *One-dimensional drift–flux model for two-phase flow in a large diameter pipe,* International Journal of Heat and Mass Transfer **46**, 1773 (2003).

[50] R. Issa and M. Kempf, *Simulation of slug flow in horizontal and nearly horizontal pipes with the two-fluid model,* International Journal of Multiphase Flow **29**, 69 (2003).

[51] D. Collins, L. Nghiem, Y.-K. Li, and J. Grabenstetter, *Efficient approach to adaptive-implicit compositional simulation with an equation of state,* SPE Reservoir Engineering (Society of Petroleum Engineers) **7**, 259 – 264 (1992).

[52] M. L. Michelsen, *The isothermal flash problem. part ii. phase-split calculation,* Fluid Phase Equilibria **9**, 21 (1982).

[53] R.Pozo, K.A.Remington, and A.Lumsdaine., *Sparselib++ v. 1.5 sparse matrix class library reference guild,* (1986).

[54] G. I. Goumas, K. Kourtis, N. Anastopoulos, V. Karakasis, and N. Koziris, *Understanding the performance of sparse matrix-vector multiplication,* 16th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008) , 283 (2008).

[55] Y. Jiang, *Techniques for Modeling Complex Reservoirs and Advanced Wells* (Stanford University, 2007).

[56] M. Verschoor and A. C. Jalba, *Analysis and performance estimation of the conjugate gradient method on multiple gpus,* Parallel Comput. **38**, 552 (2012).

[57] K. M. Pour, D. Voskov, and D. Bruhn, *Coupled modeling of well and reservoir for geo-energy applications,* Geoenergy Science and Engineering , 211926 (2023).

[58] D. W. Peaceman, *Interpretation of wellblock pressures in numerical reservoir simulation. part 3. off-center and multiple wells within a wellblock,* SPE Reservoir Engineering (Society of Petroleum Engineers) **5**, 6p 16976 (1990).

[59] J. Holmes, T. Barkve, and O. Lund, *Application of a multisegment well model to simulate flow in advanced wells,* (1998) p. 171 – 181.

[60] Schlumberger, *Eclipse technical description,* (2007).

[61] Y. Jiang, *Techniques for modeling complex reservoirs and advanced wells*, Ph.D. thesis, Stanford University (2008).

[62] S. Livescu, L. Durlofsky, and K. Aziz, *A semianalytical thermal multiphase wellbore flow model for use in reservoir simulation,* (2008) p. 2345 – 2360.

[63] L. Pan and C. M. Oldenburg, *T2well—an integrated wellbore–reservoir simulator,* Computers & Geosciences **65**, 46 (2014), tOUGH Symposium 2012.

[64] S. Livescu, L. Durlofsky, K. Aziz, and J. Ginestra, *A fully-coupled thermal multiphase wellbore flow model for use in reservoir simulation,* Journal of Petroleum Science and Engineering **71**, 138 – 146 (2010).

[65] R. Tonkin, M. O'Sullivan, and J. O'Sullivan, *A review of mathematical models for geothermal wellbore simulation,* Geothermics **97** (2021), 10.1016/j.geothermics.2021.102255.

[66] R. N. Horne, *Design considerations of a down-hole coaxial geothermal heat exchanger.* (1980) p. 569 – 572.

[67] D. Peaceman, *Interpretation of well-block pressures in numerical reservoir simulation,* Society of Petroleum Engineers (1978).

[68] S. GeoQuest, *Schlumberger geoquest. eclipse technical description, multi-segment wells,* (2005).

[69] J. P. Fontanilla, *A mathematical model for the prediction of wellbore heat loss and pressure drop in steam injection wells,* (1980).

[70] M. Prats, *Thermal Recovery*, Society of Petroleum Engineers of AIME (1986).

[71] A. S. Odeh, *Comparison of Solutions to a Three-Dimensional Black-Oil Reservoir Simulation Problem,* Journal of Petroleum Technology **33**, 13 (1981).

[72] B. Li and H. A. Tchelepi, *Nonlinear analysis of multiphase transport in porous media in the presence of viscous, buoyancy, and capillary forces,* Journal of Computational Physics **297**, 104 (2015).

[73] F. P. Hamon, B. T. Mallison, and H. A. Tchelepi, *Implicit hybrid upwinding for two-phase flow in heterogeneous porous media with buoyancy and capillarity,* Computer Methods in Applied Mechanics and Engineering **331**, 701 (2018).

[74] K. H. Coats, *An equation of state compositional model,* SPE Journal **20**, 363 (1980).

[75] G. J. Moridis, M. T. Reagan, and Y. Liu, *Numerical simulations in support of a long-term test of gas production from hydrate accumulations on the alaska north slope: Reservoir response to interruptions of production (shut-ins),* Energy & Fuels **36**, 3496 (2022).

[76] C. Chen, H. Shao, D. Naumov, Y. Kong, K. Tu, and O. Kolditz, *Numerical investigation on the performance, sustainability, and efficiency of the deep borehole heat exchanger system for building heating,* Geothermal Energy **7** (2019), 10.1186/s40517-019-0133-8.

[77] K. H. Coats, *An Equation of State Compositional Model,* Society of Petroleum Engineers Journal **20**, 363 (1980).

[78] G. Acs, S. Doleschall, and E. Farkas, *General purpose compositional model,* Society of Petroleum Engineers Journal **25**, 543 (1985).

[79] P. Deuflhard, *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms* (Springer, Berlin, 2004).

[80] Voskov and Tchelepi, *Compositional nonlinear solver based on trust regions of the flux function along key tie-lines,* Society of Petroleum Engineers (2011).

[81] D. V. Voskov, *Operator-based linearization approach for modeling of multiphase multi-component flow in porous media,* Journal of Computational Physics **337**, 275 (2017).

[82] B. Li and H. A. Tchelepi, *Nonlinear analysis of multiphase transport in porous media in the presence of viscous, buoyancy, and capillary forces,* Journal of Computational Physics **297**, 104 (2015).

[83] M. Khait and D. V. Voskov, *Adaptive parameterization for solving of thermal/compositional nonlinear flow and transport with buoyancy,* SPEJ (2018), doi:10.2118/182685-PA.

[84] J. Ortega and W. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* Computer Science and Applied Mathematics. A Series of Monographs and Textobooks (Academic Press, New York, 1970).

[85] H. Gilgen, *Univariate Time Series in Geosciences: Theory and Examples* (Springer Berlin Heidelberg, Berlin, 2006).

[86] J. Jansen, R. Fonseca, S. Kahrobaei, M. Siraj, G. Essen, van, and P. Hof, Van den, *The egg model - a geological ensemble for reservoir simulation,* Geoscience Data Journal **1**, 192 (2014).

[87] X. Wang, *TRUST-REGION NEWTON SOLVER FOR MULTIPHASE FLOW AND TRANSPORT IN POROUS MEDIA*, Ph.D. thesis, Stanford University (2012).

[88] F. O. Alpak, *A mimetic finite volume discretization method for reservoir simulation,* SPE Journal **15**, 436 – 453 (2010).

[89] R. Mattey and S. Ghosh, *A novel sequential method to train physics-informed neural networks for allen cahn and cahn hilliard equations,* Computer Methods in Applied Mechanics and Engineering **390** (2022), 10.1016/j.cma.2021.114474.

[90] A. S. Krishnapriyan, A. Gholami, S. Zhe, R. M. Kirby, and M. W. Mahoney, *Characterizing possible failure modes in physics-informed neural networks,* (2021) p. 26548 – 26560.

[91] R. Rastegar and K. Jessen, *Lumping and Delumping for Integrated Compositional Modeling,* in *All Days* (SPE, 2009).

[92] K. Jessen and R. R. Moghadam, *A Flow Based Lumping Approach for Compositional Reservoir Simulation,* in *All Days* (SPE, 2009).

[93] D. Tang and A. Zick, *A New Limited Compositional Reservoir Simulator,* in *SPE Symposium on Reservoir Simulation* (Society of Petroleum Engineers, 1993).

[94] C. Ganapathy and D. Voskov, *Multiscale reconstruction in physics for compositional simulation,* Journal of Computational Physics **375**, 747 (2018).

[95] Y. Chen and D. Voskov, *Optimization of CO2 injection using multi-scale reconstruction of composition transport,* Computational Geosciences **24**, 819 (2020).

[96] M. Raissi, P. Perdikaris, and G. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,* Journal of Computational Physics **378**, 686 (2019).

[97] O. Fuks and H. A. Tchelepi, *LIMITATIONS OF PHYSICS INFORMED MACHINE LEARNING FOR NONLINEAR TWO-PHASE TRANSPORT IN POROUS MEDIA,* Journal of Machine Learning for Modeling and Computing **1**, 19 (2020).

[98] F. Orr, *Theory of Gas Injection Processes* (Tie-Line Publications, Holte, 2007).

[99] M. A. Christie and M. J. Blunt, *Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques,* SPE Reservoir Evaluation & Engineering **4**, 308 (2001).

A

APPENDIX

A.1. ALGORITHM TO TRACK NEWTON'S TRAJECTORY IN AN ARBITRARY DIMENSION OF OBL SPACE

Newton's trajectory passes several cells or hyper cells in a higher dimension of parameter space. In order to detect those cells or hypercells, we implement the iterative algorithm 1. The algorithm is incrementing to the next cell iteratively by finding the minimum distance between the initial point with respect to interfaces and moving on in a gradient direction to find the next point until reaching the last hypercubes. Here's the explanation of the following algorithm steps:

1. *COMPUTESIGMA*: This function calculates σ which is the minimum distance between the initial points and all the interfaces of the OBL cube/ hypercube.

2. *NEXTPOINT*: This function increments to the next hypercubes by moving along the Newton trajectory gradient.

3. *NEWBOX*: This function detects the new vertices of the new cubes/hypercubes.

These steps are repeated until reaching the last points of the OBL and all the interfaces detected. Figure A.1 illustrates in 2-d space the Newton trajectory tracking. The algorithm for tracking is independent of the degree of freedom of the system and is able to track Newton trajectories for an arbitrary number of dimensions.

A

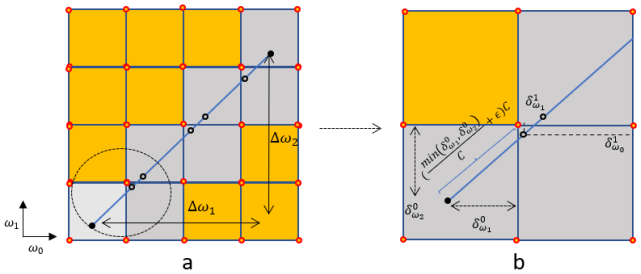


Figure A.1: (a) Newton trajectory passing several OBL cells (b) Zoomed-in view of detecting the point inside the next cell of parameter space

Algorithm 1 Iterative tracking

procedure ITERATIVE TRACKING($initpoint, lastbox$)
  Input:
  $initpoint \leftarrow (p_i, z_{1i}, z_{2i}...)$
  $endpoint \leftarrow (p_e, z_{1e}, z_{2e}...)$
  $C \leftarrow initpoint - endpoint$
  while box != lastbox **do**
    $Sigma \leftarrow ComputeSigma(initpoint, box, C)$
    $initpoint \leftarrow nextPoint(initPoint, C, Sigma)$
    $box \leftarrow newbox(initpoint, z_{vec}, p_{vec})$ ▷ p_{vec}, z_{vec} are uniformly mesh in parameter space
  end while
  return box
end procedure

function COMPUTESIGMA($initpoint, box, C$)
  for $i \leftarrow 1, NC$ **do** ▷ NC = Number of components
    $\Lambda_i \leftarrow max(\frac{rightboundary - initpoint(i)}{C(i)}, \frac{leftboundary - initpoint(i)}{C(i)})$
  end for
  $\sigma \leftarrow min(\Lambda)$
  return σ
end function

function NEXTPOINT($initpoint, Sigma, C$)
  for $i \leftarrow 1, NC$ **do** ▷ NC = Number of components
    $nextPoint(i) \leftarrow initpoint(i) + (\sigma + \epsilon)C(i)$ ▷ $+\epsilon$ to make sure passing the interface
  end for
  return $nextPoint$
end function

function NEWBOX($initpoint, z_{vec}, p_{vec}$)
  for $i \leftarrow 1, NC$ **do** ▷ NC = Number of components
    if $i == 1$ **then**
      $newbox(i) \leftarrow [\frac{initpoint(i)-p_{vec(1)}}{p_{vec(2)}-p_{vec(1)}}]$ ▷ [] Rounding to upper integer
    else
      $newbox(i) \leftarrow [\frac{initpoint(i)-z_{vec(1)}}{z_{vec(2)}-z_{vec(1)}}]$
    end if
  end for ▷ We find one of the new vertices of the new box
end function

A.2. SENSITIVITY ANALYSIS TO THE OBL RESOLUTION

In this section, we investigate the effect of the OBL resolution on the nonlinear solver performance. Table A.1 summarizes the result of the coarsening of OBL resolution of the fracture test for the binary kernel on the immiscible test case for the single control timestep $\Delta t = 100$ days. We observe that by coarsening the resolution, the number of Newton iterations decreases. In general, decreasing the resolution of OBL relaxes the nonlinearity of the problem and fewer Newton iterations. Figure A.2 illustrates the solution for different OBL resolutions. Notice that decreasing the resolution of the OBL significantly can degrade the solution. The rigorous error analysis has been carried out on the original paper [81].

Resolution	Newton iteration	Inflection points
4 x 4	5	1e-8
8 x 8	6	0.285
16 x 16	8	0.40
32 x 32	10	0.44
64 x 64	12	0.476
120 x 120	14	0.487

Table A.1: Coarsening of OBL resolution

A.3. SENSITIVITY OF PRECONDITIONING TO TRANSPORT PARAMETERS

Next, we report the performance of the advanced nonlinear solver after running the simulation with the small timestep ($\Delta t = 0.001$) days until ($T = 0.3$) days and save the solution. Next, we restart from the solution for one large timestep equals to 0.2 days with different preconditioning strategies. Considering the initial guess the inflection point and initial guess equals to the initial condition and initial guess the solution of the sat-

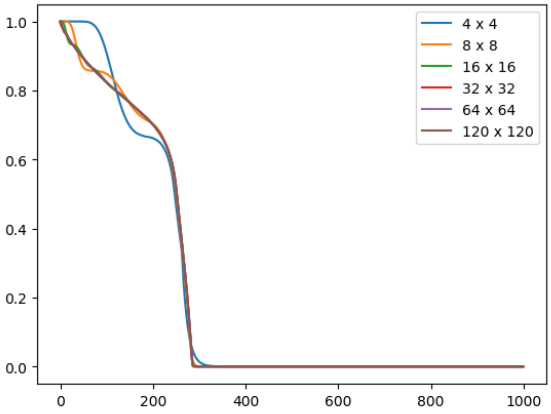


Figure A.2: Solution of front propagation for different OBL resolutions

uration after running the simulation until $T = 0.3d$. We test the transport problems for four types of relative permeability curves.

$$K_{rw} = S^{n_w}; K_{ro} = (1-S)^{n_o}, \tag{A.1}$$

Table A.2: Different preconditioning strategy

K_{rp}	M	Initial guess from		
		without preconditioning	new timestep	inflection point
$n_o = n_p = 2$	0.5	60	13	9
	1	56	14	9
	10	1	4	10
$n_o = n_p = 3$	0.5	142	22	11
	1	149	25	10
	10	111	5	10
$n_o = n_p = 10$	0.5	198	27	10
	1	204	27	12
	10	229	30	11
$n_o = 2, n_p = 10$	0.5	142	20	11
	1	149	21	10
	10	111	22	10

It is noticed that:

(i) With a preconditioning strategy, we accelerate the convergence for all the cases. As expected, choosing the inflection point as an initial guess always works better since the derivative **B** is maximized which maximizes the propagation of the compositional front downstream.

(ii) By increasing the mobility ratio, the number of iteration generally increase for the same time step and the same exponents. It can be explained by the fractional flow theory when by increasing the M (unfavorable displacement), the shock speed increases. The preconditioning strategy helps in all cases.

A

A.4. DEAD-OIL PROPERTIES

Table A.3: Rock-Fluid parameters

Parameter	Value	Description
C_r	1.00E-09	Rock compressibility
S_{pr}	0.01	Phase residual saturation
S_{or}	0.01	Oil residual saturation
n_o	2	Oil exponents

Table A.4: Dead oil properties

Parameter	Oil	Description
ρ	1000	Surface density
μ	1-5 cp	Viscosity range
B_o	1	Formation Volume Factor

Parameter	Water	Description
ρ	1000 kg/cm$\hat{3}$	surface density
c_p	10^{-9}	Compressibility
μ	1cp	Viscosity

CURRICULUM VITÆ

Kiarash MANSOUR POUR

19-08-1992 Born in Tehran, Iran.

EDUCATION

2011–2015 Bachelor Degree, Chemical Engineering

University of Zanjan

2015–2018 Master Degree, Energy Engineering

Politecnico di Milano

2018–2023 Doctor of Philosophy, Reservoir Engineering

Delft University of Technology

Thesis: Modeling of coupled reservoir and wellbore for energy transition

Promotors: Dr. D.V. Voskov

Prof. dr. D. F Bruhn

LIST OF PUBLICATIONS

JOURNAL ARTICLES

8. **K. Mansour Pour, D.V. Voskov, D.F. Bruhn**, *Coupled modelling of well and reservoir for geo-energy application*, Geoenergy science and technology (2023): **1-18**.

7. **K. Mansour Pour, D.V. Voskov, D.F. Bruhn**, *Nonlinear solver based on trust region approximation for* CO_2 *utilization and storage in subsurface reservoir*, Geoenergy science and technology (2023)): **1-18**.

6. **K. Mansour Pour, D.V. Voskov**, *Physics-informed neural networks based on sequential training for* CO_2 *utilization and storage in subsurface reservoir* Journal of Machine Learning for Modeling and Computing): **1-18**.

5. **K. Mansour Pour, D.V. Voskov**, *Physics Informed Neural Networks Based on Sequential Training for CO2 Utilization and Storage in Subsurface Reservoir*, presented at EAGE GET 2022, The Hague, The Netherlands.

4. **G. Hadjisotiriou, K. Mansour Pour, D.V. Voskov**, *Application of Deep Neural Networks to the Operator Space of Nonlinear PDE for Physics-Based Proxy Modelling*, presented at SPE Reservoir Simulation Conference, Galveston, Texas, USA, March 2023.

3. **K. Mansour Pour, D.V. Voskov**, *Modelling Single Well Closed Loop for Dry Rock Geothermal System*, presented at 47^{th} PROCEEDINGS, 47th Workshop on Geothermal Reservoir Engineering (online).

2. **K. Mansour Pour, D.V. Voskov**, *Decoupled velocity formulation for geothermal well and reservoir simulation*, presented at EAGE GET 2021, Utrecht, The Netherlands (online).

1. **K. Mansour Pour, D.V. Voskov**, *Adaptive Nonlinear Solver for a Discrete Fracture Model in Operator-Based Linearization Framework*, presented at 17^{th} European Conference on the Mathematics of Oil Recovery, 2020 (online).

ACKNOWLEDGEMENTS

As I sit down to write the acknowledgments, I can't help but feel a tad bit anxious. It's not just because I know it's the most widely read part of any thesis, maybe even the first one people skim through. It's mostly because I want to make sure I do justice in expressing my deep gratitude to all those incredible individuals who played a part in shaping my journey and getting me to where I am today.

First of all, **Denis**, I am immensely grateful to you for your unwavering guidance, expertise, and continuous support throughout this research journey. Beyond the campus, you've been a great friend who consistently shares your valuable expertise in every aspect of your knowledge, including academic subjects. I truly appreciate your and **Kate's** hospitality when you occasionally arrange barbecues. **David**, I would like to thank you for giving me the opportunity and funding to work on this project. I would like to express my gratitude for your valuable insights and thorough review of our paper. Your stoic demeanor is truly admirable, and it is something I greatly appreciate and aspire to learn from. To all the members of the **DARTS** group **Xiaocong, Mark, Yang, Xiaoming, Stephan, Aleks, Michiel, Yuan, George**, thank you for the great work atmosphere we created together. The weekly meetings held during the COVID-19 period provided us with a sense of purpose and companionship amidst the pandemic. They quietly reassured us that we were not only engaged in meaningful work but also not alone in navigating through these challenging times. **Phil**, I really appreciated it when you invited me to present at a symposium in Aula TU Delft after COVID-19. That symposium right after the pandemic was my first and only experience during my Ph.D. to finally present for a real audience and not online.

I want to say thank you to the Reservoir Engineering department at TU Delft. I have worked as a teaching assistant for a course on multiphase flow in porous media for the past three years. I had the honor of working with **Bill** for two of those years. I really enjoy working with you and your passion for teaching. **Hadi** Thanks for the Darsim meeting and all the pragmatic questions you asked. I have not had a chance to work with you but I always get energized when I attend your presentation. My gratitude goes also to **Prof. Zitha, Prof. Demyanov, Prof.Vardon, Dr. Rongier, Dr Møyner** for being part of my Ph.D. committee.

I would like to express my gratitude to all my office mates to create a friendly environment. **Sian** I just feel like this acknowledgment is not going to be complete without you. Thanks for all the finest cakes you baked and brought almost weekly to share with us and hey thanks for all the tips and techniques on growing long curly hair. **Mohsen, Kishan, Willemijn, Laura, Herminio, Mohsan, Siamak, Artur, Thejas** and all other members of the lunch team group, thanks for all of the lunch that we had together. Thanks to all the Geo-Runners, especially thanks to **Chris, Samantha** not only for all the kudos you gave me on Strava but also for being there and cheering me up during the hot deadly Leiden marathon.

Beautiful **Alina**, you have been one of the big parts of my life in the past years. I can write ten full dissertations about you. We have been through ups and downs, happy and sad moments together and we grow and get stronger. I am forever thankful for your love and companionship and I hope only the best for you and your future.

"Happiness only real when shared", What I found the most valuable during my Ph.D. is the great friends with whom I can share both moments of pure bliss and the depths of sorrow. **Lotte, Diogo, Claudio**, now that we are all apart in different parts of the world, I realize how amazing was the time we had together. They say the time you enjoy wasting is not wasted. **Fabio, Vincenzo, Alberto, Claudia**, Thursday night would be just another monotonic day without you all. **Miguel**, Thanks for all the discussion during our dinner time. Hopefully, you are not going to overcook pasta when you are going to start living away from me. **Jonathan, Ombretta**, that flight cancellation near Christmas that we all could not wait to see our families initially brought disappointment and yet led to a great friendship. **Nuno**, "Spinning, body, mind, and soul, Namaste!" Man, your vibrant energy and untamed Latin spirit during the spinning classes are infectious. **Francesco**, boy, you are one hell of a guitarist. Stay strong with your new life in Amsterdam, I am sure you will be fine. **Mohammad, Mimmo**, when I first met you 6 years ago in Milan, I never thought I was gonna put you in my acknowledgment. Thanks for being a good friend.

Finally, I want to express my gratitude to my family. I deeply appreciate my parents, **Behzad** and **Afsaneh**, for their unwavering love and support. I'm thankful to my sister, **Kiana**, for her constant presence whenever I needed assistance. I also extend my thanks to **Vincent, Claire, Laurence, Pierre, Lolo**, and the entire Lolo team for their support, affection, and exceptional hospitality.

Kiarash Mansour Pour
Delft, June 2023