Advanced simulation of CO2 storage in saline aquifers
Machine learning and multiscale approach

Zhao, M.

DOI
10.4233/uuid:70fc9610-2ab3-477b-82ba-dd865488789f

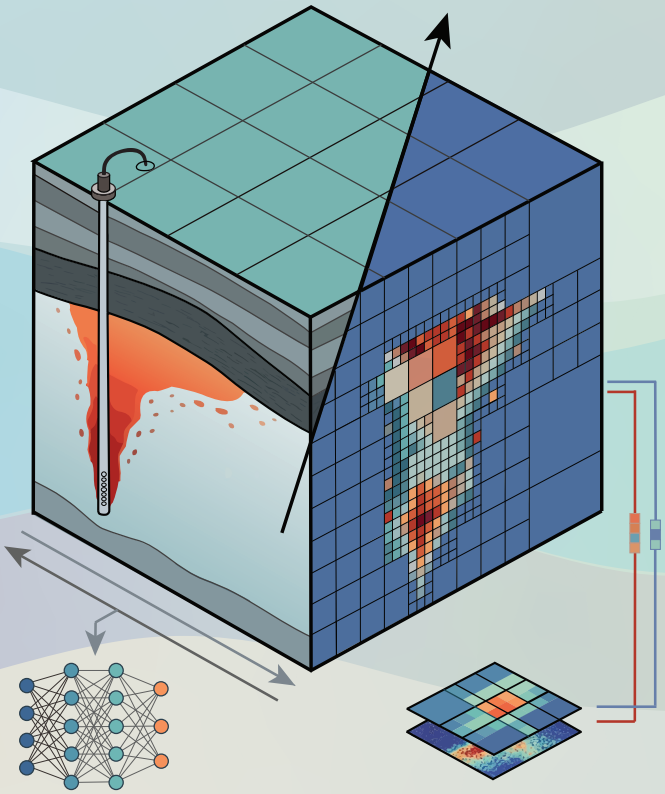
Publication date
2026

Document Version
Final published version

Citation (APA)
Zhao, M. (2026). *Advanced simulation of CO2 storage in saline aquifers: Machine learning and multiscale approach*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:70fc9610-2ab3-477b-82ba-dd865488789f

Advanced Simulation of CO_2 Storage in Saline Aquifers

Machine Learning and Multiscale Approach



Mengjie Zhao
赵孟杰

Advanced Simulation of CO_2 Storage in Saline Aquifers

Machine Learning and Multiscale Approach

ADVANCED SIMULATION OF CO_2 STORAGE IN SALINE AQUIFERS

MACHINE LEARNING AND MULTISCALE APPROACH

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus, prof. dr. ir. H. Bijl,
chair of the Board for Doctorates
to be defended publicly on
Monday 19 January 2026 at 17:30

by

Mengjie ZHAO

Master of Science in Oil and Gas Field Development Engineering,
China University of Petroleum (East China), Qingdao, China

born in Henan, China.

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Dr. ir. M.I. Gerritsma,	Delft University of Technology, promotor
Prof. dr. H. Hajibeygi,	Delft University of Technology, promotor
Independent members:	
Prof. dr. S. Geiger,	Delft University of Technology
Prof. dr. S. Krevor,	Imperial College London
Prof. dr. ir. C. Vuik,	Delft University of Technology
Prof. dr. A. Busch,	Heriot-Watt University
Dr. Y. Wang,	China University of Geosciences (Wuhan)
Dr. D.V. Voskov,	Delft University of Technology, reserve member

This research was carried out within the Flow Physics and Technology Department and the Delft Advanced Reservoir Simulation group (DARSim) at Delft University of Technology, and was sponsored by the Chinese Scholarship Council (CSC).

Keywords:	Geologic carbon storage, saline aquifers, multiphase flow in porous media, deep learning method, multiscale simulation
Printed by:	Ipskamp Printing
Front & Back:	Design by the author

ISBN 978-94-6518-220-9

An electronic copy of this dissertation is available at
https://repository.tudelft.nl/.

To every action there is always opposed an equal reaction.

Isaac Newton

CONTENTS

SUMMARY

The secure and efficient storage of carbon dioxide (CO_2) in deep saline aquifers is widely recognized as a critical component in the global strategy for mitigating anthropogenic climate change. Accurate and computationally efficient modeling of CO_2 migration, phase behavior, and long-term trapping mechanisms in geologically heterogeneous formations remains an open and pressing challenge. This dissertation aims to advance modeling techniques for multiphase multicomponent flow in porous media, with a focus on two complementary directions: physics-constrained deep learning and multiscale numerical solvers. Both approaches seek to provide computational efficiency while delivering accurate results on a desired level, thereby enabling predictive simulations across a range of spatial and temporal scales.

Chapter 1 starts with the scientific context and motivation for geological CO_2 storage, and introduces the key challenges that exist along the way. Moreover, the contributions from the scientific community in various aspects are highlighted, including surrogate modeling and numerical simulation.

Chapter 2 covers the governing equations for compositional multiphase flow in porous media, along with the associated thermodynamic formulations governing phase behavior and equilibrium. The chapter further attempts to provide detailed explanation of the primary CO_2 trapping mechanisms, including structural, residual, dissolution, and mineral trapping. Following this, chapter 3 introduces the numerical discretization strategies employed to solve the governing equations. It provides an overview of the fine-scale simulation framework, comprising spatial and temporal discretization, coupling and linearization techniques, and the implementation of flash calculations for phase equilibrium.

Leveraging these theoretical foundations, chapter 4 presents a multi-task learning with consistency (MTLC) surrogate model. This deep learning architecture jointly predicts the evolution of key state variables by exploiting shared representations and enforcing task-level consistency, thereby capturing the mutual dependencies between them. Chapter 5 builds upon this approach by presenting a physics-constrained neural network (CO_2PCNet), which incorporates the residuals of the governing partial differential equations into the training loss to enforce physical consistency. The model features a spatiotemporal architecture that includes spatial encoders, ConvLSTM layers for temporal sequence modeling, and an auto-regressive structure to extrapolate plume dynamics beyond the training window.

In the previous chapters, the focus is on surrogate modeling based on fine-scale simulation frameworks. However, due to the large size of the real field-scale domains,

performing fully resolved simulations using fine-scale methods remains computationally impractical. From this point onward, the focus shifts towards the multilevel multiscale methods. Chapter 6 introduces the Algebraic Dynamic Multilevel (ADM) method as a compositional flow simulator. ADM constructs a hierarchy of computational grids and employs algebraic prolongation and restriction operators to dynamically refine regions exhibiting strong compositional gradients while coarsening smoother regions. Localized multiscale basis functions enable accurate representation of fine-scale behavior on coarse levels, while constant basis functions are used for hyperbolic parameters. By addressing the trade-off between accuracy and efficiency, the methods developed advance predictive modeling for geological CO_2 storage. Building upon this framework, chapter 7 extends the ADM approach to fractured aquifers by integrating it with a projection-based Embedded Discrete Fracture Model (pEDFM). This combination enables the explicit representation of fractures and allows for independent grid adaptation in the fracture and matrix domains. The method is used to investigate the impact of flow barriers and highly conductive fractures on plume evolution and trapping efficiency. Numerical results demonstrate that accurate predictions can be achieved using only a fraction of the fine-scale resolution, significantly reducing the size of the resulting linear systems. For large-scale simulations, this combined pEDFM-ADM framework offers scalability and computational savings while maintaining physical fidelity.

This dissertation is concluded in chapter 8 and references used in this work follow afterwards.

In summary, this thesis contributes to the understanding of CO_2 storage in deep saline aquifers by developing and evaluating a range of modeling strategies. The subsequent chapters examine specific aspects of this research in detail, offering in-depth analyses and insights that advance the state of knowledge in CO_2 sequestration modeling.

PREFACE

Dear reader,

It gives me great pleasure to see your interest in my PhD dissertation. What you are about to read is the result of a four-year research journey I carried out at TU Delft. While I will not fully capture every detail of this experience, I would like to highlight the key moments that have shaped my path.

It was during the final year of my master's studies in China that I made the decision to pursue a PhD abroad. I was driven by a growing passion for scientific research and a deep curiosity to see the world beyond what I had known. With excitement and uncertainty, I applied to Prof. Hadi Hajibeygi, whose work deeply inspired me. When I received their offer, I knew I was about to begin an entirely new chapter of life.

I arrived in the Netherlands in September 2021, while the world was still under the shadow of the pandemic. It was my first time stepping outside of my home country. Everything was unfamiliar, yet endlessly fascinating. The diversity of cultures, the rhythm of a new academic environment, the cold winter winds in Delft, all of it left a lasting impression.

From the very beginning, Dr. Marc Gerritsma offered me more than academic guidance. He introduced me to the section, patiently helped me navigate complex derivations, and created a supportive and friendly atmosphere that made me feel immediately welcomed. Whenever I needed help, whether academic or personal, he was always there, offering his time and support without hesitation. Around the same time, Prof. Hadi Hajibeygi helped me explore a research direction. With his guidance, and with the thoughtful support of Dr. Yuhang Wang, we found a way to merge my background with a new and exciting topic: machine learning for carbon dioxide storage.

Because of the pandemic, courses and meetings were still held online, and much of my early research was carried out in quiet isolation. The first publication always takes longer than expected, but the joy that comes with seeing your work published makes the wait worthwhile. That joy is, I believe, one of the most powerful reasons to love scientific research.

Later on, under Hadi's mentorship, I began exploring multiscale numerical methods. It was then that I truly discovered the beauty of numerical simulation. With each iteration, each corrected error, and each tested idea, I felt abstract concepts take shape, and invisible systems become concrete. His constant support and trust gave me the confidence to go deeper, not just in solving problems, but in thinking more

critically and creatively. He always believed in me unconditionally and generously provided me with opportunities to grow.

This dissertation is the result of that journey, a path shaped by uncertainty and perseverance, setbacks and breakthroughs, and, above all, the support of those who believed in me. I am excited to share the outcomes of this work and the experiences behind it in the pages that follow.

Yours Sincerely,

Mengjie Zhao
Delft, July 2025

1

INTRODUCTION

Well begun is half done.

Aristotle

1

1.1. OVERVIEW OF CO_2 GEOLOGICAL STORAGE

Carbon Capture and Storage (CCS) is an increasingly critical component of climate change mitigation strategies [1–5]. This technology involves capturing carbon dioxide (CO_2) from significant industrial and energy-related sources, compressing and subsequently injecting it into suitable geological formations deep underground. This approach aims to substantially decrease atmospheric greenhouse gas concentrations, thus contributing to global climate stabilization. Figure 1.1 presents a comprehensive view of the rise in CO_2 emissions due to fossil fuels and industry, alongside their impacts on atmospheric concentrations. The relevance and urgency of advancing CCS technologies have been magnified by the steady increase in atmospheric CO_2 levels, which have risen by approximately 50% since the start of the Industrial Revolution. According to the latest data from THE National Oceanic and Atmospheric Administration (NOAA, 2025) [6], CO_2 levels at the Mauna Loa Observatory in Hawaii have reached 427 parts per million (ppm), a dramatic rise from the pre-industrial average of 280 ppm. This substantial increase highlights the urgency of exploring and implementing CO_2 storage solutions on a broad scale.

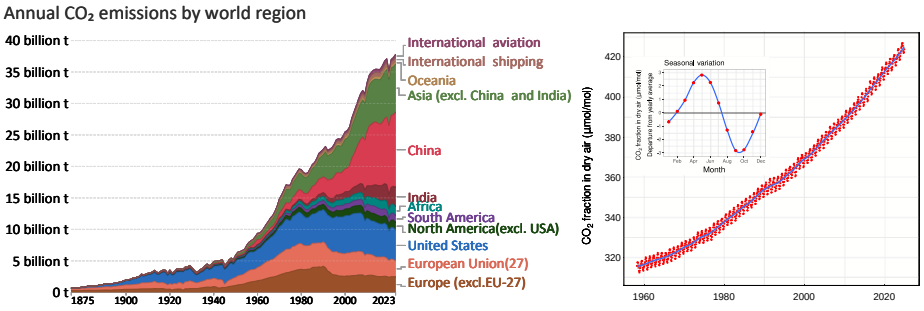


Figure 1.1: Left: Annual CO_2 emissions by world region [7], source: Global Carbon Budget (2024); Right: Atmospheric CO_2 concentration in Hawaii from 1958 to 2024 [6], source: NOAA measurements at the Mauna Loa Observatory.

Geological storage of CO_2 leverages the inherent capacity of subsurface formations to contain fluids over geological timescales, as demonstrated historically by natural reservoirs of hydrocarbons and saline water. Various candidate formations for CO_2 sequestration include depleted oil and gas reservoirs, unmineable coal seams, basalt formations, and particularly deep saline aquifers, owing to their substantial storage capacities and widespread distribution [8]. Deep saline aquifers, typically situated at depths exceeding one kilometer, possess sufficient porosity and permeability to accommodate and retain injected supercritical CO_2 [9–12]. Under appropriate geological conditions, the injected CO_2 is expected to remain securely confined for centuries to millennia [13]. This long-term stability makes geological storage a reliable technology for climate mitigation strategies, offering a durable and large-scale solution for achieving carbon neutrality.

In transitioning toward renewable and sustainable energy systems, CCS stands out as a vital technology in geo-engineering applications. Not only does it address the immediate challenge of reducing the quantity of CO_2 entering the atmosphere, but it also provides a pathway to achieve long-term environmental sustainability goals.

1.2. THE ROLE OF COMPUTATION MODELS

Computational models are fundamental in understanding and managing the CO_2 storage in deep saline aquifers, serving as primary tools in CCS strategies. By offering a robust quantitative framework that simulates complex subsurface phenomena, these models enable accurate prediction of the behavior of injected CO_2 under diverse geological and operational conditions [14]. Reliable modeling of multiphase, multicomponent flows under complex thermophysical conditions is critical for accurately estimating storage capacities, assessing environmental risks [15, 16], and optimizing injection methodologies [17]. To date, high-fidelity numerical simulations of fully physics-coupled flow represent the primary approach for monitoring and predicting the evolution of injected CO_2 plumes [18, 19].

Precise modeling of subsurface pressure distribution and plume migration dynamics is crucial for operational effectiveness and long-term risk mitigation, ensuring secure storage of CO_2. Figure 1.2 schematically presents diverse trapping mechanisms, each occurring across varying temporal scales and collectively contributing to the long-term security of geological sequestration. Comprehensive models incorporating buoyancy-driven flow, solubility-driven dissolution, molecular diffusion, capillary interactions, and reservoir heterogeneity deliver essential insights into the complexities of subsurface CO_2 dynamics [20].

High-resolution geological models, typically constructed using seismic surveys, well log interpretations, and core analysis, facilitate discretized representations of reservoirs. Solutions to mass and momentum conservation equations, calibrated using empirical field data, enhance the fidelity and predictive accuracy of these simulations [21, 22].

The significance of computational models is not limited to CO_2 sequestration. In a broader geo-engineering context, from hydrocarbon production to groundwater management, computational modeling provides the necessary predictive power to analyze interactions among subsurface fluids, rocks, and operational parameters.

1.3. CHALLENGES

Despite extensive progress and application, constructing and running computational models for CO_2 sequestration presents significant challenges. Subsurface environments are inherently complex, characterized by geological uncertainties such as varying porosity and permeability, the presence of faults and fractures, and unpredictable fluid pathways [24]. These geological uncertainties significantly influence the behav-

1

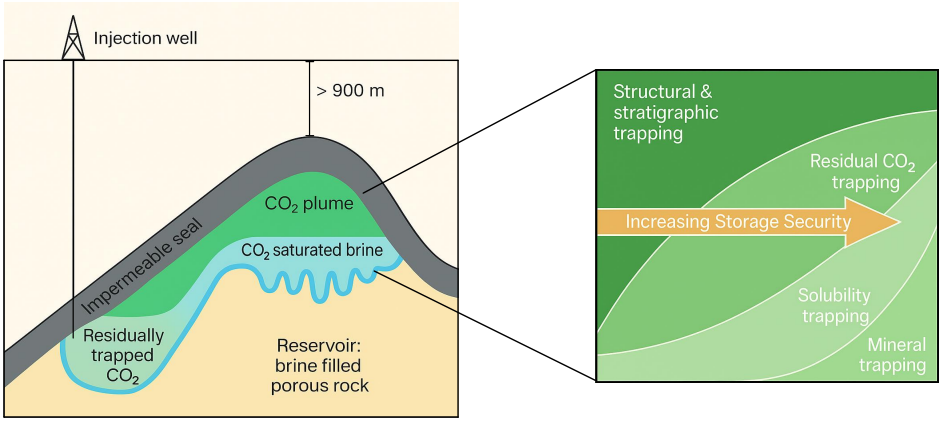


Figure 1.2: Trapping mechanisms governing the immobilization of injected CO_2 in geological formations [13, 23].

ior and migration patterns of CO_2 plumes. Accurately resolving stratigraphic heterogeneity, particularly high-conductivity layers and natural flow barriers, necessitates advanced modeling strategies that maintain computational efficiency without compromising accuracy.

Injected CO_2, existing in a supercritical state due to the elevated pressures and temperatures in deep saline aquifers, exhibits complex thermophysical properties. The immiscibility and lower density of supercritical CO_2 relative to resident brine promote buoyancy-driven upward migration, presenting a substantial modeling challenge. Buoyancy-driven flow significantly influences the spatial and temporal evolution of the plume and can result in preferential flow pathways, plume fingering, and channeling phenomena, each of which requires careful consideration in numerical models [19].

Thermodynamic modeling of CO_2 sequestration systems further complicates simulation process [25]. Accurate phase equilibrium modeling and multiphase flow behavior require precise equations of state and thermodynamic correlations to correctly represent supercritical fluid properties, particularly under variable pressure and temperature conditions. Errors in predicting thermophysical properties, such as viscosity, density, and interfacial tensions, directly influence flow dynamics and trapping efficiency, thereby compromising the reliability of long-term predictions.

Furthermore, realistic field-scale simulations demand high-resolution computational grids that can capture detailed physical and geological heterogeneities, resulting in extremely large, sparse linear systems. Multiple concurrent trapping mechanisms further add modeling complexities. Traditional numerical methods quickly reach computational limitations, motivating the development and implementation of sophisticated multiscale, adaptive discretization, and surrogate modeling techniques [26]. These ad-

vanced methodologies must carefully balance computational efficiency and accuracy, as overly simplified models risk losing critical physical fidelity.

These complexities, coupled with the need for reliable long-term forecasting and robust uncertainty quantification, often result in significant computational demands [27, 28]. Extensive sensitivity analyses and iterative model calibration typically require substantial computational resources, parallel computing frameworks, and advanced modeling techniques to maintain feasible computational costs without sacrificing critical accuracy.

1.4. MODELING TECHNIQUES

To address the aforementioned challenges, two main modeling approaches have been adopted for investigating and managing CO_2 sequestration: surrogate modeling and numerical simulation. Each approach is integral in advancing our understanding and predictive capabilities of CO_2 dynamics within subsurface geological environments.

1.4.1. SURROGATE MODELING

Recent advances in machine learning have significantly enhanced surrogate modeling techniques, providing computationally efficient alternatives to traditional physics-based numerical simulations [29–31]. Given the computational complexity of CCS projects, surrogate models are particularly advantageous for data assimilation and uncertainty quantification, where repeated high-fidelity simulations are impractical [28, 32–34].

Machine learning methodologies such as Gaussian Processes (GP) [35] and Radial Basis Functions (RBF) [36, 37] have been extensively explored. Although certain methods have shown promise in specific contexts, such as the combination of an RBF model with an appropriate sampling strategy reducing computational costs in coastal aquifer management while maintaining accuracy [38], their overall performance is often constrained by the curse of dimensionality [39]. For instance, GP techniques typically work well with up to about 20 parameters [40], but integrating dimensionality-reduction schemes can introduce information loss, potentially affecting model fidelity.

The complexity of problems in our work demands enhanced capabilities from surrogate models, making deep learning (DL) techniques a suitable choice. DL techniques have demonstrated their potential in tackling high-dimensional regression problems and are extensively applied in computer vision (CV) tasks such as object detection and image segmentation with the rapid development of graphics processing units (GPUs) [41, 42]. These DL models utilize statistical modeling to approximate the relationships between inputs and outputs, capturing the underlying physical principles effectively [43–45].

In the general context of fluid flow and transport in porous media, there has been some promising research using deep neural networks (DNN) to predict mappings from

the reservoir properties to the state variables [46, 47]. Figure 1.3 shows a DNN-based surrogate modeling framework. These networks utilize high-fidelity simulation data to construct statistical representations. Under appropriate configuration and training, these surrogates have the potential to accurately approximate even the most complex functions [48]. To be specific, convolutional neural networks (CNN) are powerful tools for image processing, efficiently capturing spatial features while minimizing the number of free parameters through the shared-weight architecture of the convolutional layers or filters [49, 50]. Zhu and Zabaras (2018) [51] first proposed a full CNN-based approach to approximate the single-phase steady state flow in geological models characterized by Gaussian permeability fields, and demonstrated that their neural network, trained with a limited amount of data, was able to predict high-dimensional pressure maps. Building on this, Mo et al. (2019) [52, 53] applied a similar framework to forecast the dynamic response of multiphase flow in heterogeneous media. Meanwhile, Tang et al. (2020) [54] developed a Recurrent residual U-Net for data assimilation in dynamic subsurface flow problems, combining a residual U-Net with a recurrent architecture Long Short-Term Memory (LSTM) to capture both spatial and temporal information. Kadeethum et al. (2021) [55] employed a Generative Adversarial Network (GAN)-based deep learning model to predict the behavior of steady-state flow in heterogeneous porous media.

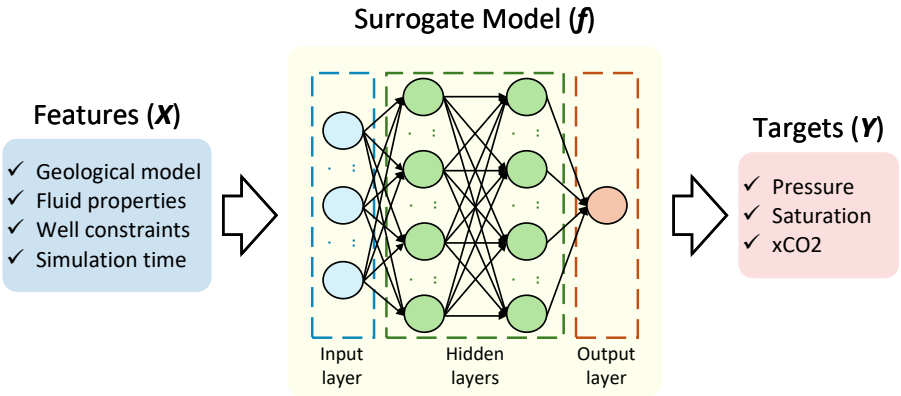


Figure 1.3: Structure of a DNN-based surrogate modeling framework.

Advanced neural architectures have been tailored specifically for CO_2 sequestration challenges. Wen et al. (2021) [56, 57] enhanced the U-Net architecture to translate permeability and injection parameters into CO_2 saturation maps. Zhong et al. (2019) [58] incorporated U-Net with a GAN to capture the temporal dependencies effectively. Additionally, Yan et al. (2022) [59] utilized Fourier Neural Operators (FNO) to predict the spatial and temporal evolution of CO_2 plumes during injection and post-injection periods, respectively. Chu et al. (2023) [60] explored deep learning-based surrogate models in a physics-framed two-phase flow scenario, involving the displacement of

water by CO_2. They compared multiple algorithms, including Multi-layer Perceptrons (MLPs), CNN, LSTM, and Gated Recurrent Unit (GRU) cells, concluding that MLPs exhibited robust performance with most trainable parameters.

Despite the recent advances made with data-driven surrogate models for subsurface CO_2 storage, these models primarily derive insights from data, occasionally overlooking the crucial physical laws governing the systems. This approach, while statistically potent, can lead to predictions that, although accurate, may conflict with established physical principles or intuitive expectations [61]. To mitigate this issue, the concept of physics-informed neural networks (PINNs) has recently received significant attention in the field of computational science [62]. PINNs are designed to solve nonlinear partial differential equations (PDEs) by integrating PDEs, initial conditions, and boundary conditions directly into the loss function of the neural network. This innovative approach allows PINNs to leverage initial condition data to solve PDEs, ensuring that the outputs are not only based on data but also conform to fundamental physical laws. By embedding differential equations directly into the learning process, PINNs provide a powerful tool for modeling complex physical phenomena, particularly in scenarios where traditional data might be limited or sparse. Recent research has demonstrated the potential of combining PINNs with data-driven methods in various fields, from fluid dynamics [63–65] to solid mechanics [66], thus demonstrating their ability to tackle complex, multiscale, and multiphysics phenomena.

1.4.2. NUMERICAL SIMULATION

Numerical simulation remains the primary tool for modeling the CO_2 migration and trapping in deep saline aquifers, by discretizing the governing mass and energy conservation equations and then capturing the dynamics of multiphase, multicomponent flow under buoyancy, dissolution, diffusion, and capillary forces [67, 68]. These high-fidelity simulations provide detailed insights into reservoir performance over both short- and long-term timescales.

Several classical numerical methods have been utilized in reservoir simulations, each offering advantages and trade-offs depending on the complexity of the problem. The finite volume method (FVM) is widely adopted for multiphase flow and reactive transport problems, owing to its local conservation properties and relatively straightforward implementation [69]. The finite element method (FEM) has found broad application in structural analyses and geomechanics [70], but has also been extended to solve coupled flow and deformation problems in reservoir engineering [71]. The lattice Boltzmann method (LBM) is gaining attention for microscale flow phenomena, particularly in pore-scale modeling, where it can directly simulate fluid-rock interactions in complex geometries [72, 73].

For field-scale CO_2 storage problems, FVM-based simulations are frequently employed due to their efficiency and robust handling of geological heterogeneities [14, 74]. They often include fully implicit schemes to handle strong coupling among phases and components, which is especially important for reactive-compositional

flows [75, 76] and processes involving temperature-dependent behavior and capillarity [77]. Such high-resolution grids accurately resolve fine-scale structures, but they inevitably lead to large, sparse linear systems that can become computationally prohibitive when modeling extensive, heterogeneous formations [20].

Multiscale methods have been introduced to overcome these challenges associated with resolving large-scale heterogeneous reservoirs on fine grids. The multiscale finite element (MsFE) method [19, 78, 79] and multiscale finite volume (MsFV) method [80–84] reduce the dimension of the global problem by constructing coarse-scale pressure systems, while incorporating local basis functions or operators to preserve critical fine-scale information. These techniques significantly lower computational costs and memory requirements, making them particularly appealing for large-scale CCS scenarios. Iterative procedures often combine coarse-grid solutions with fine-scale corrections through algebraic restriction and prolongation operators, ensuring that key flow features, such as sharp saturation fronts, are accurately captured without globally refining the entire domain [85, 86].

Building on these multiscale principles, the Algebraic Dynamic Multilevel (ADM) method was specifically developed to handle fully implicit systems on dynamically updated multilevel grids [87, 88]. ADM addresses the multiscale multilevel coexistence of the pressure (elliptic or parabolic) and transport (hyperbolic) unknowns by adaptively refining or coarsening the mesh at each time step based on a front-tracking criterion. Fine-grid resolution is focused where strong gradients occur, such as near injection wells or boundaries of the CO_2 plume. Regions with minimal flow variation remain on a coarser mesh, reducing the number of active cells and thereby lowering computational costs. Finite-volume restriction and prolongation operators are carefully designed to preserve local conservation across each level of refinement. Originally formulated for two-phase immiscible flow and black-oil reservoir models [89, 90], ADM has shown promise in tackling heterogeneous formations typical of CO_2 storage sites. By efficiently managing computational resources, ADM enables large-scale, fully implicit simulations at finer resolutions than traditional uniform-grid approaches can feasibly accommodate.

Many of these numerical methods have been implemented in both commercial and open-source simulators. Commercial platforms such as ECLIPSE (Schlumberger) [91] and CMG-GEM (Computer Modelling Group) integrate advanced physics models for CO_2 storage, while open-source codes like MRST (MATLAB Reservoir Simulation Toolbox) [92, 93], provide flexible environments for experimentation and research. High-performance computing (HPC) resources further enhance scalability, allowing large ensembles of field-scale simulations for history matching or uncertainty quantification [94]. In parallel, advances in solver algorithms and domain-decomposition techniques continue to improve the efficiency of handling large linear systems that arise from implicit multiphase flow models [95].

1.5. RESEARCH GOALS

Building on the challenges and prior research outlined in the previous sections, the aim of this work is to contribute to the field of CO_2 storage in deep saline aquifers. This dissertation focuses on enhancing both the predictive capabilities and computational efficiency of CO_2 sequestration models. The central objective is to enhance the methodologies used to model and manage CO_2 sequestration processes, with a particular focus on improving the efficiency and accuracy of these models under complex geological conditions.

This objective is pursued through two complementary themes. The first is leveraging state-of-the-art machine learning approaches, specifically multi-task deep learning and a physics-constrained neural network to facilitate rapid and reliable predictions of CO_2 plume evolution. The second theme involves developing novel simulation strategies, particularly those that utilize the ADM method, to efficiently capture flow dynamics in large domains with complex geology.

- Develop a comprehensive theoretical framework for CO_2-brine multiphase multicomponent flow, detailing the governing equations of flow dynamics, thermodynamics, and trapping mechanisms;
- Develop and validate multi-task deep learning models that can predict CO_2 migration with high accuracy while ensuring consistency across different computational tasks;
- Design and implement a physics-constraint neural network that integrates physical laws directly into the learning process, aiming to improve the reliability and accuracy of predictions during the full cycle of CO_2 migration;
- Develop the ADM method to handle fully implicit simulations and fully compositional modeling of CO_2 fluid flow;
- Extend the ADM framework to incorporate fractures, enabling accurate modeling of CO_2 storage in fractured media through the pEDFM-ADM method.

1.6. THESIS OUTLINE

The dissertation is organized into eight chapters, beginning with this introduction, and is structured to progressively develop from foundational concepts to advanced modeling and simulation techniques for CO_2 sequestration. Chapter 2 introduces the governing equations for multiphase flow, highlighting the influence of key parameters such as composition and fluid properties, while also discussing fundamental trapping mechanisms and the thermodynamic principles underlying CO_2-brine interactions in saline aquifers. Chapter 3 presents a high-fidelity framework for simulating CO_2 migration at fine scales. This formulation addresses how advanced numerical schemes can capture sharp fronts, heterogeneous formations, and local flow variations critical to CO_2 sequestration. A multi-task deep learning framework is introduced in Chap-

ter 4 to efficiently approximate the temporal and spatial evolution of CO_2 plumes. By emphasizing consistent predictions of multiple physical variables, the method reduces computational costs while preserving predictive accuracy across diverse reservoir scenarios. Chapter 5 advances this approach by incorporating physics-based constraints directly into the neural network architecture. Through embedding the relevant partial differential equations into the training process, the method aims to ensure physically realistic predictions during both injection and post-injection periods. Chapter 6 introduces the Algebraic Dynamic Multilevel (ADM) method as an effective strategy for large-scale simulations on adaptive grids. This chapter details how ADM dynamically refines or coarsens the computational domain in response to flow gradients, thereby reducing computational overhead while preserving fidelity. Chapter 7 extends the ADM approach to fractured media, where flow complexity is significantly heightened by discrete fracture networks. A projection-based embedded approach is employed to represent fractures, enabling accurate modeling of CO_2 migration pathways and trapping in heterogeneous reservoirs. Chapter 8 provides the conclusions and future work. Finally, the references used in this dissertation are listed.

2

FUNDAMENTALS OF CO_2-BRINE MULTIPHASE FLOW

In this chapter, we first present the governing equations for multiphase multicomponent flow specifically related to the CO_2 and brine interactions in saline aquifers. This is followed by a detailed introduction of thermodynamic equilibrium to model the solubility and phase behavior of CO_2 under subsurface conditions. Additionally, the chapter explores various trapping mechanisms — structural, dissolution, residual, and mineral. The primary focus of this chapter is on these fundamental equations and their role in describing the relationships between primary variables and the behavior of CO_2 in subsurface environments.

2.1. MASS CONSERVATION EQUATIONS

The mass conservation equations describing flow and transport phenomena in CO_2 storage systems are presented here in increasing complexity, starting from basic immiscible multiphase flow and concluding with multiphase multicomponent flow in fractured porous media.

2.1.1. IMMISCIBLE MULTIPHASE FLOW

Mass conservation equations governing immiscible multiphase flow through porous media, characterized by n_{ph} distinct fluid phases, are expressed as follows [96]

$$\frac{\partial}{\partial t}\left(\phi \rho_\alpha S_\alpha\right) + \nabla \cdot \left(\rho_\alpha \mathbf{u}_\alpha\right) - q_\alpha = 0, \quad \forall \alpha \in \{1, \ldots, n_{\text{ph}}\}, \tag{2.1}$$

where t is the time, ϕ is the rock porosity, and ρ_α, S_α and q_α are the mass density, saturation and sink/source term of the phase α, respectively. The phase velocity $\mathbf{u}_\alpha$ is defined by Darcy's law

$$\mathbf{u}_\alpha = -\lambda_\alpha \cdot \left(\nabla p_\alpha - \rho_\alpha g \nabla h\right), \tag{2.2}$$

with the phase mobility defined as $\lambda_\alpha = \mathbf{K}\frac{k_{r,\alpha}}{\mu_\alpha}$, where μ_α is the phase viscosity, $\mathbf{K}$ is the rock permeability tensor, and $k_{r,\alpha}$ is the phase relative permeability [97].

Phase pressures p_α are interrelated by capillary pressure P_c, as follows:

$$p_\alpha - p_\beta = (1-\delta_{\alpha,\beta})\, P_{c_{\alpha\beta}}, \quad \forall \alpha,\beta \in \{1,\dots,n_{\text{ph}}\}, \tag{2.3}$$

where $\delta_{\alpha,\beta}$ is the Kronecker delta, equal to 1 if $\alpha = \beta$ and 0 otherwise. Capillary pressure is negative if α is the wetting phase and positive otherwise. Generally, $P_{c_{\alpha,\beta}}$ is a nonlinear function of the wetting-phase saturation. Mass balance in the system is strictly ensured by the saturation constraint

$$\sum_{\alpha=1}^{n_{\text{ph}}} S_\alpha = 1. \tag{2.4}$$

2.1.2. MULTIPHASE MULTICOMPONENT FLOW IN POROUS MEDIA

In the context of CO_2 sequestration in saline aquifers, the governing equation for mass conservation, applicable to an isothermal system comprising n_{c} components distributed among n_{ph} phases, is formulated as [98, 99]

$$\frac{\partial}{\partial t}\left(\phi \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha}\rho_\alpha S_\alpha\right) + \nabla\cdot\sum_{\alpha=1}^{n_{\text{ph}}}\left(x_{c,\alpha}\rho_\alpha \mathbf{u}_\alpha + S_\alpha\rho_\alpha \mathbf{j}_{c,\alpha}\right) - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} q_\alpha = 0, \quad \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{2.5}$$

where subscripts $\alpha \in \{w,n\}$ denote the wetting (brine-rich) and non-wetting (CO_2-rich) phases, respectively. The index c identifies the components (CO_2 and H_2O). $x_{c,\alpha}$ is the mass fraction of component c in phase α and the diffusion flux $\mathbf{j}_{c,\alpha}$ is governed by Fick's law, which accounts for molecular diffusion driven by gradients of the component mass fraction

$$\mathbf{j}_{c,\alpha} = -\phi D_{c,\alpha}\nabla x_{c,\alpha}, \tag{2.6}$$

where $D_{c,\alpha}$ is the mutual diffusion coefficient of component c in phase α. In addition, the model assumes that the pore space is fully occupied by the phases, with CO_2 and H_2O accounting for all component distributions. This is mathematically enforced by the conditions that the sum of phase saturations and component mass fractions in each phase equal to one

$$\sum_{\alpha=w,n} S_\alpha = 1, \quad \text{and} \quad \sum_{c=CO_2,H_2O} x_{c,\alpha} = 1. \tag{2.7}$$

2.1.3. MULTIPHASE MULTICOMPONENT FLOW IN FRACTURED POROUS MEDIA

In geological formations, natural or induced fractures can either act as preferential pathways for CO_2 migration or serve as flow barriers that restrict fluid movement.

The mass conservation equations for the CO_2-brine system in porous media with discrete embedded fractures are formulated as

$$\frac{\partial}{\partial t}\left(\phi^m \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha}\rho_\alpha S_\alpha\right) + \nabla\cdot\sum_{\alpha=1}^{n_{\text{ph}}}\left(x_{c,\alpha}\rho_\alpha \mathbf{u}_\alpha + S_\alpha\rho_\alpha \mathbf{j}_{c,\alpha}\right) - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} q_\alpha - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} Q_\alpha^{mf} = 0, \quad \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{2.8}$$

for the rock matrix m and

$$\frac{\partial}{\partial t}\left(\phi^f \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha}\rho_\alpha S_\alpha\right) + \nabla\cdot\sum_{\alpha=1}^{n_{\text{ph}}}\left(x_{c,\alpha}\rho_\alpha \mathbf{u}_\alpha + S_\alpha\rho_\alpha \mathbf{j}_{c,\alpha}\right) - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} Q_\alpha^{fm} - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} Q_\alpha^{ff} = 0, \quad \forall c \in \{1,\dots,n_{\text{c}}\}. \tag{2.9}$$

for the lower-dimensional fracture f. Q_α^{mf} and Q_α^{fm} are the phase flux exchanges between the rock matrix and the fractures, and Q_α^{ff} represents the influx between the fractures. Note that the mass conservation law enforces: $\iiint_V Q_\alpha^{mf}\, dV = -\iint_{A_i} Q_\alpha^{fm}\, dA$ and $\iint_{A_i} Q_\alpha^{f_i f_j}\, dA = -\iint_{A_j} Q_\alpha^{f_j f_i}\, dA$.

2.2. THERMODYNAMIC EQUILIBRIUM

In deep saline aquifers, injected CO_2 interacts with resident brine, forming a multiphase system typically comprising an aqueous phase and a supercritical or gaseous CO_2-rich phase. Accurate modeling of this system requires a thermodynamic description of phase equilibrium that accounts for pressure, temperature, and salinity. Thermodynamic equilibrium governs the partitioning of chemical species across coexisting phases and provides the foundation for compositional simulation and solubility modeling.

2.2.1. PHASE PARTITIONING AND K-VALUE

For a binary CO_2-H_2O system, thermodynamic equilibrium is achieved when the fugacity of each component is equal in the two phases [100]

$$f_{c,\alpha}\left(p, x_{c,\alpha}\right) - f_{c,\beta}\left(p, x_{c,\beta}\right) = 0, \quad \forall \alpha \neq \beta \in \{1,\dots,n_{\text{ph}}\},\ c \in \{CO_2, H_2O\}, \tag{2.10}$$

where $f_{c,\alpha}$ is the fugacity of component c in phase α. This condition is often reformulated in terms of equilibrium ratios, or K-values

$$K_c = \frac{x_{c,g}}{x_{c,l}}, \tag{2.11}$$

where $x_{c,g}$ and $x_{c,l}$ are the mole fractions of component c in the vapor and aqueous phases, respectively. These K-values are functions of pressure, temperature, and salinity, and provide the basis for equilibrium-based compositional modeling.

2.2.2. EQUILIBRIUM CONSTANTS AND FUGACITY FORMULATION

Equilibrium constants κ_c relate the fugacity of a component in the gas phase to its activity in the aqueous phase [101]

$$\kappa_{H_2O} = \frac{f_{H_2O,g}}{a_{H_2O,l}}, \tag{2.12}$$

and

$$\kappa_{CO_2} = \frac{f_{CO_2,g}}{a_{CO_2,l}}, \tag{2.13}$$

where $f_{c,g}$ is the fugacity of component c in the gas phase, and $a_{c,l}$ is the activity of component c in the aqueous phase. Fugacity is expressed in terms of mole fraction, fugacity coefficient and system pressure [102]

$$f_{c,g} = \varphi_c x_{c,g} p, \tag{2.14}$$

where φ_c is the fugacity coefficient and p is the system pressure.

The equilibrium constants κ_{H_2O} and κ_{CO_2} vary with temperature and pressure. The pressure correction at a given temperature is approximated by

$$\kappa\left(T,p\right) = \kappa^0\left(T,p^0\right)\exp\left(\frac{(p-p^0)\overline{V}_c}{RT}\right), \tag{2.15}$$

where $\overline{V}_c$ is the average partial molar volume of component c over the pressure interval p^0 to p, and p^0 is a reference pressure (typically 1 bar). R is the universal gas constant and T is the temperature. The values of $\kappa^0_{CO_2}$ and $\kappa^0_{H_2O}$ are computed using SUPCRT92 [103] and Harvey et al. [104], respectively.

2.2.3. EQUATION OF STATE

The Redlich–Kwong equation of state (RK-EOS) is used to compute fugacity coefficients φ_c. It is given by [105]

$$p = \frac{RT}{V-b} - \left(\frac{a}{T^{0.5}V(V+b)}\right), \tag{2.16}$$

where parameters a and b represent measures of intermolecular attraction and repulsion, respectively. p corresponds to the total system pressure and V is the molar volume. For a binary mixture, parameters a and b are calculated using standard mixing rules [102]

$$a_{mix} = x^2_{H_2O,g}a_{H_2O} + 2x_{H_2O,g}x_{CO_2,g}a_{H_2O-CO_2} + x^2_{CO_2,g}a_{CO_2}, \tag{2.17}$$

and

$$b_{mix} = x_{H_2O,g}b_{H_2O} + x_{CO_2,g}b_{CO_2}. \tag{2.18}$$

The fugacity coefficient of a component c, φ_c, is then calculated based on these mixing rules and RK-EOS [102]

$$\ln(\varphi_c) = \ln\left(\frac{V}{V-b_{\text{mix}}}\right) + \left(\frac{b_c}{V-b_{\text{mix}}}\right) - \left(\frac{2\sum_i^{n_c} x_{i,g}a_{ic}}{RT^{1.5}b_{\text{mix}}}\right)\ln\left(\frac{V+b_{\text{mix}}}{V}\right) + \left(\frac{a_{\text{mix}}b_c}{RT^{1.5}b_{\text{mix}}^2}\right)\left[\ln\left(\frac{V+b_{\text{mix}}}{V}\right) - \left(\frac{b_{\text{mix}}}{V+b_{\text{mix}}}\right)\right] - \left(\frac{pV}{RT}\right). \tag{2.19}$$

These equations can be solved directly by setting

$$A = \frac{\kappa^0_{H_2O}}{\varphi_{H_2O}p}\exp\left(\frac{(p-p^0)\overline{V}_{H_2O}}{RT}\right), \tag{2.20}$$

and

$$B = \frac{\varphi_{CO_2}p}{55.508\kappa^0_{CO_2}}\exp\left(-\frac{(p-p^0)\overline{V}_{CO_2}}{RT}\right), \tag{2.21}$$

such that

$$x_{H_2O,g} = \frac{1-B}{1/A-B}, \tag{2.22}$$

$$x_{CO_2,l} = B(1-x_{H_2O,g})\,. \tag{2.23}$$

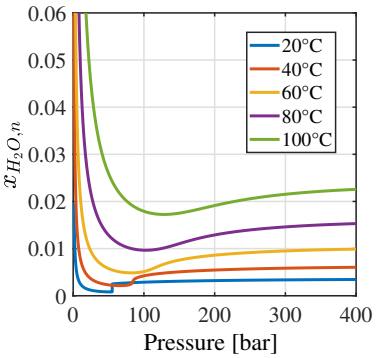
To account for salinity, the aqueous-phase activity is corrected using an activity coefficient γ_{CO_2} [106]. The molality-based mole fraction is computed as

$$x_{CO_2,l} = \frac{m_{CO_2}}{m_{CO_2}+55.508+2m_{Na}}. \tag{2.24}$$

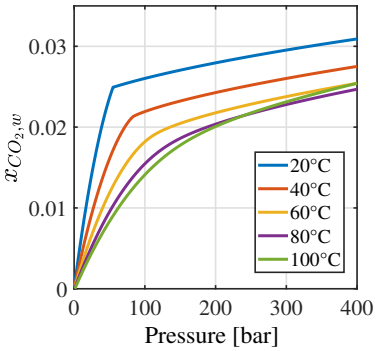
where m_{CO_2} is the molality corrected by γ_{CO_2}, and m_{Na} is molality of Na^+.

By substituting the computed values of $x_{c,l}$ and $x_{c,g}$ into the expressions for K_c, the equilibrium partitioning of CO_2 and H_2O can be determined across a wide range of geological conditions. This framework provides thermodynamically consistent K-values to be used in compositional simulation and data-driven modeling.

Figure 2.1 illustrates representative results derived from this thermodynamic model, showing the mutual solubilities of CO_2 and H_2O as functions of pressure and temperature. At elevated pressures, where CO_2 exists in a dense phase, its solubility in water remains relatively insensitive to further pressure increases. In contrast, under lower-pressure conditions where CO_2 is predominantly in a vapor state, the mass fraction of H_2O in the CO_2-rich phase is significantly higher. As pressure increases within this regime, CO_2 transitions to a denser phase, leading to a marked decline in H_2O solubility. Additionally, increasing temperature enhances H_2O volatility, resulting in higher water content in the CO_2-rich phase, while simultaneously reducing CO_2 solubility in the aqueous phase.



(a) H_2O mole fraction in CO_2-rich phase



(b) CO_2 mole fraction in H_2O-rich phase

Figure 2.1: Mutual solubilities of H_2O and CO_2 from 20 to 100°C and up to a pressure of 400 bar. Data based on the model of Spycher et al. (2003) [101].

2.3. TRAPPING MECHANISMS

Following injection of CO_2 into deep saline aquifers, several trapping mechanisms operate to immobilize and retain CO_2 over different timescales. These mechanisms are commonly categorized into four types: structural and stratigraphic trapping, residual trapping, dissolution trapping, and mineral trapping [107]. An overview of these mechanisms and their associated fluid dynamics is illustrated in figure 2.2. Generally, structural, stratigraphic, and residual trapping dominate in the time scale close to the injection period because of advective and capillary transport, while dissolution becomes increasingly important in the post-injection phase. Mineral trapping, governed by slow geochemical reactions, occurs over centuries to millennia.

2.3.1. STRUCTURAL AND STRATIGRAPHIC TRAPPING

Structural and stratigraphic trapping involve the physical containment of CO_2 beneath impermeable geological seals or in specific reservoir geometries [109]. Structural trapping, typically the earliest and most identifiable mechanism, is based on the buoyant migration of injected CO_2, driven by its lower density compared to resident brine. This upward migration is arrested by overlying caprocks or structural highs. Effective structural trapping requires low-permeability seals with high capillary entry pressure to prevent further vertical migration [110].

Similarly, stratigraphic trapping occurs through lithological variations in the subsurface, such as pinch-outs, facies changes, or unconformities, which create sealed zones capable of containing CO_2. In both mechanisms, the non-wetting CO_2 accumulates beneath the seal, and its buoyancy is counteracted by the capillary entry pressure of the seal, preventing further escape. This process is analogous to natural hydrocarbon accumulation and provides the primary containment immediately after injection, effectively storing a large volume of mobile CO_2 as a separate phase below the seal.

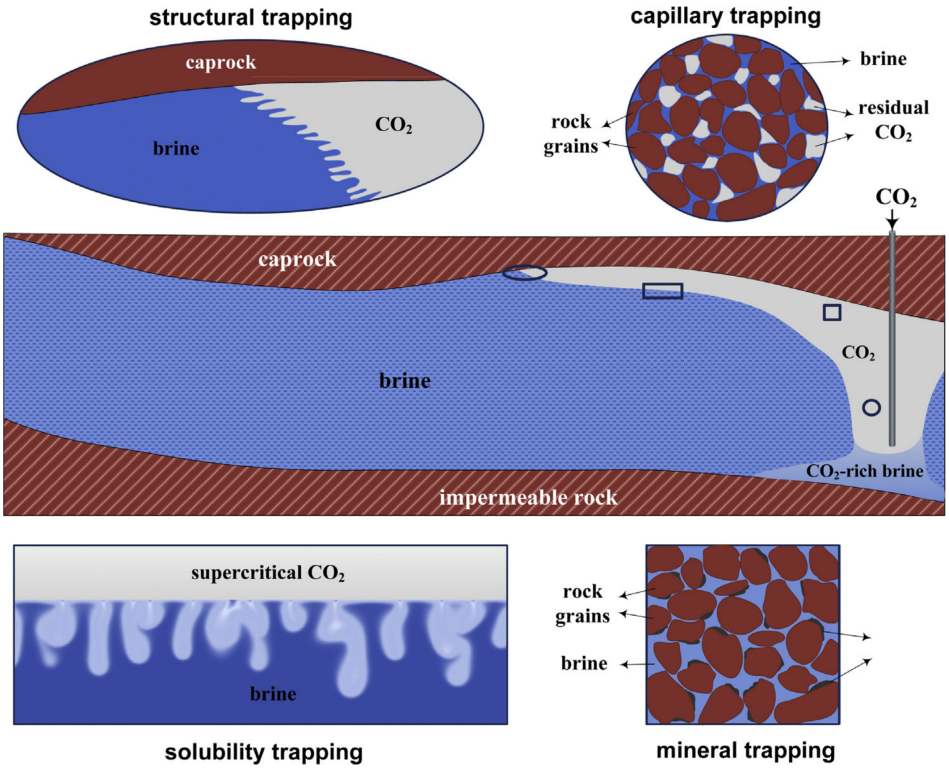


Figure 2.2: Illustration of fluid dynamics and trapping mechanisms associated with CO_2 storage in saline aquifers, including structural trapping, residual trapping, dissolution trapping, and mineral trapping [108]. Source: Adapted from Emami-Meybodi, Hamid, et al., (2015).

These forms of trapping are expected to retain CO_2 for tens to hundreds of years, until slower trapping mechanisms such as dissolution and mineralization progressively secure the CO_2 in immobile forms.

2.3.2. DISSOLUTION TRAPPING

Dissolution trapping, also referred to solubility trapping, occurs when free-phase CO_2 dissolves into the resident brine, becoming part of the aqueous phase. Once dissolved, the CO_2 no longer exists as a separate buoyant plume, thereby eliminating the risk of upward migration driven by buoyancy. Supercritical or gaseous CO_2 in contact with brine will partition into the liquid phase until thermodynamic equilibrium is reached. In deep saline aquifers, CO_2 has significant solubility on the order of tens of kilograms per m^3 of water, depending on pressure, temperature, and salinity.

The dissolution process typically begins at the interface between the CO_2 plume and the underlying brine. As CO_2 spreads laterally beneath the caprock, the surrounding brine immediately becomes saturated with dissolved CO_2. Notably, brine with dissolved CO_2 is slightly denser than the unsaturated brine, as the presence of dissolved CO_2 increases fluid density by several percent, depending on salinity. This density increase leads to gravitational instability, forming a denser layer of CO_2-rich brine beneath lighter, less-saturated brine. The result is a classic density-driven convective mixing process: the CO_2-saturated brine sinks downward and is replaced by fresh undersaturated brine, which in turn absorbs more CO_2, enhancing the overall dissolution rate.

In essence, dissolution trapping evolves in two stages: an initial diffusion-dominated phase, followed by a convective mixing phase once a critical density gradient is established [77]. Ultimately, a substantial fraction of the injected CO_2 can be dissolved into the brine and transported deeper into the formation by these convective currents, securing it as dissolved inorganic carbon.

The efficiency of convective dissolution is governed primarily by the Rayleigh number (Ra), which quantifies the ratio of buoyancy-driven to diffusive forces [108]. Under typical saline aquifer conditions, Ra often exceeds 10^3, indicating strong convective potential. A larger Ra value with higher permeability, thickness, or density contrast leads to faster onset of convection and higher dissolution rate. Additionally, heterogeneity can modulate finger patterns. High-permeability pathways may channel convection, while barriers can localize it.

Without convection, dissolution would rely solely on molecular diffusion, which is exceedingly slow. Complete dissolution could take millions of years. However, once convection begins, typically within years to decades after injection (depending on Ra), the process accelerates dramatically. Convective fingers carrying dissolved CO_2 can descend through the aquifer and spread laterally, continually enlarging the region of solubility trapping. Eventually, as CO_2 becomes more uniformly distributed, the concentration gradient that drives convection diminishes, leading to a gradual decline in mixing, a stage often referred to as “shutdown”, when most of the free-phase CO_2

is gone.

2.3.3. RESIDUAL TRAPPING

Residual trapping, also known as capillary trapping, refers to the immobilization of CO_2 as disconnected ganglia or blobs in the pore space due to capillary forces, after the plume comes into contact with brine. This mechanism typically occurs at the trailing edge of a migrating CO_2 plume or during water invasion into a CO_2-saturated rock [111]. During injection (drainage), CO_2 displaces brine and forms a continuous plume. Once injection stops, or as the plume migrates, brine, acting as the wetting phase, spontaneously imbibes into the regions previously occupied by CO_2, such as along the plume fringes or in regions affected by chase brine injection.

As brine reoccupies the pore space, capillary pressures cause the non-wetting CO_2 to break into isolated droplets or clusters that become trapped in pore throats. Underlying pore-scale phenomena include snap-off, where brine reconnects through narrow throats and isolates CO_2, and mobilization or stranding during interface movement, preventing reconnection. The result is that a certain residual saturation of CO_2 is left behind, effectively locked in place by capillary pressure barriers. This trapped CO_2 is no longer mobile, as the individual ganglia are below the critical size to form a percolating pathway. Residual trapping effectively transforms a portion of the free-phase CO_2 into an immobile phase, analogous to irreducible gas saturation. Notably, this process is irreversible on short timescales unless substantial changes in pressure, saturation, or dissolution occur to reconnect the trapped clusters.

Residual trapping typically acts shortly after injection and can immobilize a significant fraction of the CO_2 plume, especially near its trailing edge. Studies have shown that this mechanism can trap between 10 and 30%, or more, of the injected CO_2, depending on rock properties and saturation history [111, 112]. This makes residual trapping one of the most important mechanisms for short- to intermediate-term storage security in saline aquifers. Compared to dissolution and mineral trapping, residual trapping is immediately more effective because it does not require chemical transformation but instead arises from multiphase flow hysteresis [113].

The presence of a large volume of residually trapped CO_2 significantly reduces the mobility of the remaining free CO_2. As mobile CO_2 is smaller and must bypass trapped pockets, the risk of leakage is reduced. Even under a sustained pressure gradient, the trapped CO_2 remains immobile as long as saturation remains below the remobilization threshold. Over longer timescales, dissolution may gradually shrink the ganglia, and significant pressure depletion may also induce remobilization, although both processes are typically slow.

2.3.4. MINERAL TRAPPING

Mineral trapping is a geochemical mechanism in which dissolved CO_2 reacts with formation minerals to form solid carbonate compounds, permanently locking CO_2 in a

stable solid form. This process is essentially the culmination of a series of geochemical interactions.

Initially, CO_2 dissolves in brine to form carbonic acid (H_2CO^3), which subsequently dissociates into bicarbonate (HCO_3^-) and carbonate (CO_3^{2-}) ions. These reactive species then interact with divalent cations, either already present in the brine or leached from mineral surfaces, leading to precipitation of solid carbonates such as calcite. As a result, CO_2, originally present as a free gas or dissolved phase, is ultimately converted into mineral matter. Once precipitated, the CO_2 becomes part of the solid crystal structure and remains immobile unless subjected to dissolution under significantly different geochemical conditions.

Among the various trapping mechanisms, mineral trapping is generally the slowest. It typically becomes significant over timescales of centuries to millennia. In the early decades following CO_2 injection, its contribution is often limited due to slow reaction kinetics and the dominance of faster processes such as dissolution and residual trapping. Over longer periods, as dissolved CO_2 remains in prolonged contact with reactive minerals, gradual mineralization progressively converts a portion of the CO_2 into solid form.

It is also important to note that many deep saline aquifers are hosted in siliciclastic rocks, such as sandstones, which are predominantly composed of quartz. Quartz is relatively inert and contributes minimally to the mineral trapping process [114]. As a result, the potential for significant mineralization is limited in such formations. Therefore, the following discussion focuses primarily on the first three trapping mechanisms, which are more relevant over practical timescales for geological CO_2 storage. These mechanisms are collectively referred to as hydrodynamic trapping [115].

3

NUMERICAL SIMULATION STRATEGY

To simulate CO_2 sequestration processes in deep saline aquifers with physical and thermodynamic fidelity, it is necessary to translate the governing equations and equilibrium relations introduced in Chapter 2 into a numerical framework that can be applied to realistic geological domains. This chapter develops a framework by outlining the fine-scale compositional formulation, phase equilibrium models, and nonlinear solution strategies required to resolve multiphase, multicomponent flow in both porous and fractured media.

A key component of the modeling strategy is the overall-composition formulation, which uses phase pressure and component-wise overall mole fractions as primary unknowns. This formulation provides a consistent and robust way to represent mass conservation under varying phase conditions, eliminating the need to switch between variable sets during phase appearance or disappearance. The continuous equations are discretized using a finite volume method based on two-point flux approximation (TPFA) for spatial terms and backward Euler time integration. These schemes preserve local mass conservation and offer stable performance for complex transport behavior. The framework is further extended to fractured porous media, where discrete representations of fracture-matrix interactions and fracture-fracture intersections are introduced using transmissibility-based coupling strategies.

Phase equilibrium calculations are tightly integrated into the flow model. Two fluid models are considered for thermodynamic partitioning: a black-oil formulation and a fully compositional approach. Both models employ equilibrium K-values to compute phase compositions and saturations, but differ in the complexity of their assumptions. The black-oil model simplifies the problem by restricting such that only the lighter component can exist in both hydrocarbon phases. In contrast, the fully compositional model accounts for mutual solubilities of CO_2 and H_2O, with equilibrium constraints based on equations of state and activity-based fugacity formulations. Phase stability checks are conducted to determine whether flash calculations are necessary, and when multiple phases are present, the Rachford-Rice equation is solved to determine phase fractions and compositions consistent with the prescribed thermodynamic state.

The algebraic system resulting from temporal and spatial discretization is highly nonlinear due to the dependence of fluxes and thermodynamic quantities on the evolving state variables. Solving this system requires careful linearization and coupling techniques. The fully implicit method treats all governing equations of flow, transport, and thermodynamic equilibrium as a unified system, solved simultaneously at each timestep using Newton-Raphson iterations. This approach ensures strong coupling between variables, enhances convergence properties, and is more robust for complex CO_2 storage problems. Matrix assembly, Jacobian construction, and convergence criteria are described in detail for both porous and fractured domains.

Collectively, the elements presented in this chapter constitute the computational core of the modeling framework. These elements bridge the physical formulations of multiphase multicomponent systems with the modern numerical solution techniques. The presented methods are directly applicable to the field-scale modeling of CO_2 injection, migration, and trapping in heterogeneous geological formations.

3.1. FINE-SCALE FORMULATIONS

The fine-scale governing equations for multiphase, multicomponent flow in porous media are presented in their continuous form, serving as the foundation for numerical modeling of CO_2 storage in deep saline aquifers. It has been shown in the literature that such systems can be solved using as few as n_c primary variables and equations, where n_c is the number of components [116]. Typically, mass conservation equations are selected as the primary equations, while thermodynamic relations are employed to update secondary variables.

In this study, a compositional modeling framework is developed based on the overall-composition formulation. Spatial discretization is performed using the two-point flux approximation finite volume method, chosen for its simplicity and ability to ensure local mass conservation, an essential feature for accurate predictions of CO_2 migration and trapping. Temporal discretization is carried out using a fully implicit backward Euler scheme. The fine-scale formulation is further extended to explicitly represent fracture networks, employing transmissibility-based coupling between matrix and fracture elements, as well as between intersecting fractures, to accurately capture preferential flow paths.

3.1.1. OVERALL-COMPOSITION FORMULATION

Two principal formulations are commonly employed for compositional simulation of multiphase, multicomponent systems: the natural-variable formulation and the overall-composition formulation. In this research, the overall-composition formulation is adopted due to its improved numerical stability and convergence properties, particularly in cases involving phase appearance and disappearance [99]. This approach represents mixtures using component-based variables, specifically the overall composition of each component together with pressure, and retains the same set

of equations and unknowns regardless of the number of phases. By avoiding variable switching, the formulation simplifies numerical implementation and improves robustness during CO_2 dynamic processes, where evolving temperature, pressure, and salinity give rise to strong compositional gradients and transient phase behavior.

Specifically, for modeling CO_2 storage in deep saline aquifers, the injected CO_2 interacts with resident brine to form a two-component (CO_2, H_2O), two-phase (aqueous, CO_2-rich) system. The overall mole (or mass) fraction of component c, denoted z_c, is defined as

$$z_c = \sum_{\alpha=1}^{n_{\text{ph}}} \nu_\alpha x_{c,\alpha}, \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{3.1}$$

and

$$\nu_\alpha = \frac{S_\alpha \rho_\alpha}{\sum_{\alpha=1}^{n_{\text{ph}}} \rho_\alpha S_\alpha}, \tag{3.2}$$

where ν_α is the mass fraction of phase α. The governing mass conservation equations (2.5) for each component, originally expressed in terms of phase variables, are rewritten as

$$\frac{\partial}{\partial t}\left(\phi \rho_T z_c\right) + \nabla \cdot \sum_{\alpha=1}^{n_{\text{ph}}} \left(x_{c,\alpha}\rho_\alpha \mathbf{u}_\alpha + S_\alpha \rho_\alpha \mathbf{j}_{c,\alpha}\right) - \sum_{\alpha=1}^{n_{\text{ph}}} x_{c,\alpha} q_\alpha = 0, \quad \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{3.3}$$

where ρ_T denotes the total density, defined as a function of phase densities and saturations. This system is complemented by thermodynamic constraints that ensure phase equilibrium and compositional consistency,

$$f_{c,\alpha}\left(p, x_{c,\alpha}\right) - f_{c,\beta}\left(p, x_{c,\beta}\right) = 0, \quad \forall \alpha \neq \beta \in \{1,\dots,n_{\text{ph}}\},\ c \in \{1,\dots,n_{\text{c}}\}, \tag{3.4}$$

$$z_c - \sum_{\alpha=1}^{n_{\text{ph}}} \nu_\alpha x_{c,\alpha} = 0, \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{3.5}$$

$$\sum_{c=1}^{n_{\text{c}}} x_{c,\alpha} = 1, \forall \alpha \in \{1,\dots,n_{\text{ph}}\}, \tag{3.6}$$

$$\sum_{\alpha=1}^{n_{\text{ph}}} \nu_\alpha = 1. \tag{3.7}$$

3.1.2. FINE-SCALE DISCRETE SYSTEM IN POROUS MEDIA

The coupled system of non-linear equations (3.3) is discretized in space using the TPFA finite volume method, and in time using the backward Euler scheme.

The convective flux term is approximated using an upwind TPFA scheme. For each component c in phase α, the interfacial convective flux $F_{c,\alpha,ij}$ between each pair of neighboring control volumes i and j is defined as

$$F_{c,\alpha,ij} = x_{c,\alpha} \frac{k_{r,\alpha}}{\mu_\alpha} \rho_\alpha T_{ij}\left(\psi_{\alpha,i} - \psi_{\alpha,j}\right), \tag{3.8}$$

where $\psi_{\alpha,i}$ is the phase potential in cell i and $T_{ij} = \frac{A_{ij}}{d_{ij}} K_{ij}^H$ is the rock transmissibility between cells i and j. Here, A_{ij} denotes the interfacial area, d_{ij} is the distance between the cell centers, absolute permeability K_{ij}^H is harmonically averaged. The phase properties at the interface between the control volumes are evaluated using an first-order upwind scheme. The molecular diffusion flux $J_{c,\alpha,ij}$ of component c in phase α is formulated as

$$J_{c,\alpha,ij} = \rho_\alpha T_{\alpha,ij}^{\text{diff}} \left(x_{c,\alpha,i} - x_{c,\alpha,j} \right), \tag{3.9}$$

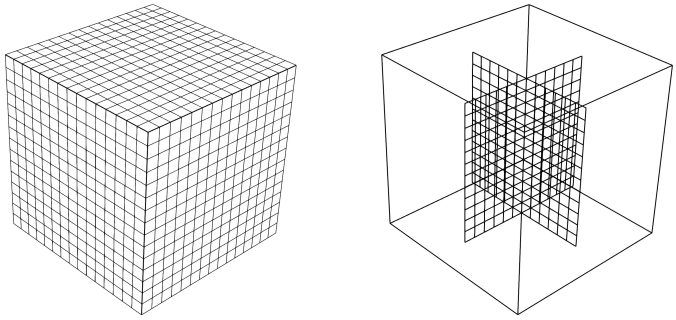
where $T_{\alpha,ij}^{\text{diff}} = \frac{A_{ij}}{d_{ij}} S_\alpha \phi D_{c,\alpha}$ is the diffusive transmissibility. Combining these terms, the discrete form of the conservation equation at each control volume i and time level $t+1$ is

$$\left(\frac{\left(\phi \rho_T z_c\right)_i^{t+1} - \left(\phi \rho_T z_c\right)_i^t}{\Delta t} \right) + \sum_{j \in \mathcal{N}(i)} \sum_{\alpha=1}^{n_{\text{ph}}} \left(F_{c,\alpha,ij} + J_{c,\alpha,ij} \right) - \sum_{\alpha=1}^{n_{\text{ph}}} \left(x_{c,\alpha} q_\alpha \right)_i = 0,$$
$$\forall c \in \{1, \ldots, n_c\}. \tag{3.10}$$

where $\mathcal{N}(i)$ represents the set of neighboring control cells that share a common interface with cell i.

3.1.3. FINE-SCALE DISCRETE SYSTEM IN FRACTURED POROUS MEDIA

The fracture-matrix coupling terms corresponding to equations (2.8)-(2.9) are modeled following the approach of Lee et al. (2000) [117] and Hajibeygi et al. (2011) [118]. Independent structured grids are generated for a three-dimensional (3D) aquifer and 2D fractures. An illustration is presented in figure 3.1.



(a) Matrix Grid (b) Fracture Grid

Figure 3.1: The independent numerical grids: (a) 3D grid for rock matrix and (b) 2D grids for fractures.

For a matrix control volume i with the volume V_i having an embedded fracture

element f with the area A_f, the flux of component c in phase α across the matrix-fracture interface is expressed as

$$F_{c,\alpha,if} = x_{c,\alpha}\rho_\alpha \frac{k_{r,\alpha}}{\mu_\alpha} T_{if}(\psi_{\alpha,f} - \psi_{\alpha,i}), \tag{3.11}$$

and

$$F_{c,\alpha,fi} = x_{c,\alpha}\rho_\alpha \frac{k_{r,\alpha}}{\mu_\alpha} T_{fi}(\psi_{\alpha,i} - \psi_{\alpha,f}), \tag{3.12}$$

where the transmissibility between the matrix and the fracture is defined symmetrically as

$$T_{\alpha,fi} = T_{\alpha,if} = CI_{if} K^H_{if}. \tag{3.13}$$

Here, CI_{if} denotes the connectivity index [118], calculated as

$$CI_{if} = \frac{S_{if}}{\langle d_{if} \rangle}, \tag{3.14}$$

where S_{if} is the surface area of the connection and $\langle d_{if} \rangle$ represents the average distance between the matrix control volume and the fracture surface. The fracture-matrix fluxes satisfy the local conservation property $F_{c,\alpha,if} = -F_{c,\alpha,fi}$.

Similarly, for two intersecting fracture elements i and j, belonging to fractures f_i and f_j, the inter-fracture flux is given by

$$F^{f_i,f_j}_{c,\alpha,ij} = x_{c,\alpha}\rho_\alpha \frac{k_{r,\alpha}}{\mu_\alpha} T^{f_i,f_j}_{ij}(\psi^{f_j}_{\alpha,j} - \psi^{f_i}_{\alpha,i}), \tag{3.15}$$

where $T^{f_i,f_j}_{\alpha,ij}$ is the rock transmissibility between the intersecting fracture elements.

The fully discrete conservation equations are then assembled for both matrix and fracture control volumes. For a matrix cell i belonging to medium m, the balance reads

$$\left(\frac{(\phi\rho_T z_c)^{t+1}_i - (\phi\rho_T z_c)^t_i}{\Delta t}\right)^m + \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j\in\mathcal{N}_m(i)} \left(F_{c,\alpha,ij} + J_{c,\alpha,ij}\right)^m = \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{f\in\mathcal{N}_f(i)} F_{c,\alpha,if} + \sum_{\alpha=1}^{n_{\text{ph}}} (x_{c,\alpha} q_\alpha)_i, \quad \forall c \in \{1,\dots,n_{\text{c}}\}, \tag{3.16}$$

and for a fracture cell i belonging to medium f is

$$\left(\frac{(\phi\rho_T z_c)^{t+1}_i - (\phi\rho_T z_c)^t_i}{\Delta t}\right)^f + \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j\in\mathcal{N}_f(i)} \left(F_{c,\alpha,ij} + J_{c,\alpha,ij}\right)^f = \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{k\in\mathcal{N}_m(i)} F_{c,\alpha,ik} + \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j\in\mathcal{N}_f(i)} F^{f_i,f_j}_{c,\alpha,ij}, \quad \forall c \in \{1,\dots,n_{\text{c}}\}. \tag{3.17}$$

Here, $\mathcal{N}_m(i)$ and $\mathcal{N}_f(i)$ denote the sets of matrix and fracture neighbors of cell i, respectively.

3.2. FLASH CALCULATIONS

Flash calculations are fundamental in modeling multiphase, multicomponent flow during geological CO_2 sequestration in saline aquifers, as they determine the thermodynamic equilibrium state of the fluid mixtures under local pressure, temperature, and composition conditions. It is typically assumed that equilibrium is reached instantaneously, allowing phase behavior to be decoupled from the transport equations. This separation enables independent evaluation of phase compositions and saturations from the governing flow equations. Two equilibrium models are considered: a simplified black-oil model, which captures essential dissolution effects while maintaining computational efficiency, and a fully compositional model, which provides a more physical thermodynamic description based on equations of state. Both models utilize equilibrium K-values to determine phase fractions and compositions through flash calculations, though they differ in complexity and underlying assumptions.

3.2.1. STABILITY CHECK

Before performing flash calculations, it is necessary to determine the number of stable phases present in each control volume. This is achieved through a thermodynamic stability check, which is conducted after updating primary variables during each iteration.

For a binary isothermal system at pressure p, the presence of two phases is indicated by

$$p_{\text{dew}} < p < p_{\text{bub}}, \tag{3.18}$$

where p_{dew} and p_{bub} denote the dew-point and bubble-point pressures, respectively. If $p > p_{bub}$, the system remains entirely in the liquid phase, satisfying [119]

$$\sum_{c=1}^{n_c} z_c K_c - 1 < 0. \tag{3.19}$$

Conversely, if $p < p_{dew}$, the system remains completely in the vapor phase, and the condition becomes

$$\sum_{c=1}^{n_c} \frac{z_c}{K_c} - 1 < 0. \tag{3.20}$$

In the two-phase region, a flash calculation is performed to resolve the phase compositions and saturations based on the local overall composition.

3.2.2. EQUILIBRIUM MODELS

Two phase equilibrium models are implemented to describe fluid behavior during CO_2 storage: the black-oil model and the fully compositional model. Both frameworks rely on K-values and employ the material balance method for phase equilibrium solutions, but differ in the physical assumptions and level of thermodynamic detail involved.

BLACK-OIL MODEL

The black-oil model simplifies phase behavior by assuming the lighter component CO_2 partitions between gas and aqueous phases, while H_2O is confined exclusively to the aqueous phase [14]. Thus, the gas phase is treated effectively as pure CO_2. This assumption significantly reduces computational complexity. Phase compositions are explicitly specified as

$$x_{H_2O,n} = 0, \quad x_{CO_2,n} = 1. \tag{3.21}$$

Under these simplifications, fugacity coefficients in equation (2.19) for both components can be directly calculated without iteration, as mixture parameters, a_{mix} and b_{mix} in equations (2.17)-(2.18), simplify to component-independent values. The quantity of dissolved CO_2 in the aqueous phase is described by the CO_2-brine solubility ratio R_s, defined as the volume ratio of dissolved CO_2 to brine at specified conditions. Following the thermodynamic approach detailed in Section 2.2, R_s relates to CO_2 mole fraction in the aqueous phase as [120]

$$R_s = \frac{\rho_w^{\text{STC}} x_{CO_2,w}}{\rho_{CO_2}^{\text{STC}} \left(1 - x_{CO_2,w}\right)}, \tag{3.22}$$

where $\rho_{\text{b}}^{\text{STC}}$ and $\rho_{CO_2}^{\text{STC}}$ are the brine and CO_2 densities at standard conditions, respectively. The subscript 'b' refers to brine, and the superscript 'STC' denotes properties evaluated at standard temperature and pressure conditions.

The corresponding K-values are given analytically by

$$K_{CO_2} = \frac{\rho_{CO_2} R_s + \rho_w^{\text{STC}}}{\rho_{CO_2} R_s}, \qquad K_{H_2O} = 0. \tag{3.23}$$

Phase compositions are then determined by

$$\begin{bmatrix} x_{CO_2,n} & x_{H_2O,n} \\ x_{CO_2,w} & x_{H_2O,w} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \dfrac{1}{K_{CO_2}} & 1 - \dfrac{1}{K_{CO_2}} \end{bmatrix}. \tag{3.24}$$

The brine-phase density incorporating dissolved CO_2 is updated as

$$\rho_w = \rho_{CO_2}^{\text{STC}} + \rho_w^{\text{STC}} R_s \,. \tag{3.25}$$

If the cell is undersaturated (i.e., only a single liquid phase is present), the amount of CO_2 dissolved in brine is not enough to reach the dissolution limit [121]. R_s is recalculated using the overall mole fraction of CO_2

$$R_s = \frac{\rho_w^{\text{STC}} z_{CO_2}}{\rho_{CO_2}^{\text{STC}} \left(1 - z_{CO_2}\right)}. \tag{3.26}$$

FULLY COMPOSITIONAL MODEL

The fully compositional model accounts for mutual solubility of CO_2 and H_2O in both phases, without restrictive assumptions. The equilibrium K-values are computed from a combination of the Redlich-Kwong equation of state and activity models, as outlined in Section 2.2. In this approach, the fugacity coefficients depend explicitly on mixture composition, pressure, and temperature, necessitating iterative solution of coupled equilibrium equations (2.16)-(2.23).

Once K-values are evaluated, phase partitioning is determined by solving the Rachford-Rice equation for vapor fraction β

$$\sum_{c=1}^{n_c} \frac{z_c(K_c-1)}{1+\beta(K_c-1)} = 0\,. \tag{3.27}$$

This nonlinear equation is robust within the physically meaningful domain and typically solved via bisection methods. Phase compositions are then directly recovered from

$$x_{c,w} = \frac{z_c}{1+\beta(K_c-1)}, \tag{3.28}$$

$$x_{c,n} = \frac{K_c z_c}{1+\beta(K_c-1)}\,. \tag{3.29}$$

The updated phase compositions are subsequently used to evaluate phase mobilities, component fluxes, and local source terms required in the transport equations. The Rachford-Rice procedure is applied efficiently in each active grid cell during the nonlinear iteration process. For cells identified as single-phase through the stability check, the flash calculation is bypassed, and only a single composition is retained. This treatment ensures computational efficiency without compromising thermodynamic consistency. The overall framework accommodates both black-oil and fully compositional models, providing a unified and robust approach under varying flow and storage conditions encountered during CO_2 sequestration in deep saline aquifers.

Furthermore, the effect of CO_2 dissolution on brine density is evaluated using a partial molar volume mixing rule

$$\frac{1}{\rho_w(p_w,T,x_{\mathrm{CO_2},w})} = \frac{1-x_{\mathrm{CO_2},w}}{\rho_w(p_w,T)} + \frac{x_{\mathrm{CO_2},w}}{\rho_{n,\phi}(T)}, \tag{3.30}$$

where $\rho_w(p_w,T,x_{\mathrm{CO_2},w})$ is the density of water containing $x_{\mathrm{CO_2},w}$, while $\rho_w(p_w,T)$ denotes the pure-phase properties, and $\rho_{n,\phi}(T)$ is the apparent density of dissolved CO_2 in water, defined as

$$\rho_{n,\phi}(T) = M_2/V_\phi, \tag{3.31}$$

where $M_2 = 44.01\times10^{-3}$ kg/mol is the molar mass of CO_2, and V_ϕ is the partial molar volume of CO_2 in water, specified in Garcia (2001) [122] and parameterized as

$$V_\phi = 10^{-6}\cdot\left(37.51 - 9.585\times10^{-2}T + 8.74\times10^{-4}T^2 - 5.044\times10^{-7}T^3\right), \tag{3.32}$$

where temperatures are measured in Celsius.

3.3. COUPLING AND LINEARIZATION

The governing equations for multiphase, multicomponent flow presented in the previous sections constitute a strongly nonlinear and coupled system, for which general analytical solutions are typically unavailable. The system is usually solved numerically. Due to the nonlinear interplay among pressure, transport processes, and thermodynamic equilibrium, robust coupling and linearization strategies are essential. The choice of numerical strategy significantly affects the accuracy, stability, and computational efficiency of the simulation.

Two widely adopted strategies are available for solving the coupled nonlinear system: sequential methods and fully implicit methods (FIM). Sequential methods decouple the system into two subproblems, pressure and transport, corresponding to the elliptic/parabolic and hyperbolic part of the governing equations, respectively. The pressure equation is constructed by forming a linear combination of the component mass balance equations, often formulated using either volume- or mass-based approaches. This pressure equation is solved first, where only pressure dependent terms are implicitly treated. The updated pressure field is then used to solve the transport equations [116]. The simplest approach would be the Implicit Pressure Explicit Saturation (IMPES) method [97], where pressure is solved implicitly while saturation is updated explicitly. However, the stability of this approach is limited to weakly coupled systems and suffers from severe restrictions in the allowable time-step size [123].

To improve robustness, the sequential fully implicit (SFI) method was developed. In SFI, additional inner iterations are performed between the pressure and transport steps until convergence is achieved. Originally formulated for immiscible multiphase flow, SFI has been extended in previous studies to both black-oil [82, 124] and fully compositional multiphase systems [125–127]. In presence of strong coupling terms between flow and transport equations, sequential strategies may not be efficient [123].

In contrast, the FIM solves the entire system of governing equations simultaneously, including pressure, transport, and thermodynamic equilibrium. This monolithic approach inherently captures interdependencies between phase behavior, pressure, and compositional changes, offering improved numerical robustness in strongly coupled systems. Although each iteration is computationally more expensive, FIM generally exhibits stable convergence behavior [128] and is widely regarded as the preferred strategy for realistic CO_2 sequestration problems [77].

In the general FIM framework, all equations are first discretized and written in residual form such that a set of discrete equations of the form [129]

$$r(\xi) = 0, \tag{3.33}$$

is obtained. Here, r is the residual vector and ξ is the vector of unknowns. Since $r(\xi)$ is generally non-linear, a global linearization technique is required. The Newton-Raphson method is typically employed, yielding

$$r^{\nu+1} \approx r^{\nu} + \left.\frac{\partial r}{\partial \xi}\right|^{\nu} \delta\xi^{\nu+1} = 0, \tag{3.34}$$

where ν denotes the nonlinear iteration index and $\delta\xi$ is the increment. Thus, at each iteration a sparse large linear system of equations, in the form

$$\mathbf{J}^{\nu}\delta\xi^{\nu+1} = -r^{\nu}, \tag{3.35}$$

which is solved iteratively until convergence is achieved (i.e., $r^{\nu} < \epsilon$, where ϵ is an arbitrary small real number). Here, $\mathbf{J}$ is the so-called Jacobian matrix and contains the partial derivatives of the residuals with respect to the unknowns.

3

3.3.1. SOLUTION STRATEGY IN POROUS MEDIA

In the overall-composition formulation, the wetting-phase pressure (p_w) and the overall mole (mass) fraction of CO_2 (z_{CO_2}) serve as the primary variables. These are stored and updated at each grid cell and timestep, ensuring that both pressure and composition fields are resolved in a fully coupled manner. The governing conservation equations (3.10) are written in residual form for each component c as follows

$$r_{c,i}^{\nu+1} = \frac{V_i\phi_i}{\Delta t}\left(z_c^{\nu+1}\rho_T^{\nu+1} - z_c^{\nu}\rho_T^{\nu}\right)_i + \sum_{j\in\mathcal{N}(i)}\sum_{\alpha=1}^{n_{\text{ph}}}\left(F_{c,\alpha,ij}^{\nu+1} + J_{c,\alpha,ij}^{\nu+1}\right) - \sum_{\alpha=1}^{n_{\text{ph}}}\left(x_{c,\alpha}\,q_{\alpha}\right)_i^{\nu+1} = 0, \quad \forall c \in \{1,\dots,n_c\}. \tag{3.36}$$

where ν and $\nu+1$ are the previous and the new iteration steps. Phase equilibrium equations (3.4) are also enforced at each iteration step

$$E_c^{\nu+1} = K_{c,\alpha,\beta}^{\nu+1}x_{c,\beta}^{\nu+1} - x_{c,\alpha}^{\nu+1} = 0\,. \tag{3.37}$$

The Newton-Raphson method is used to linearize and iteratively solve these equations

$$r_c^{\nu+1} \approx r_c^{\nu} + \frac{\partial r_c}{\partial p_w}|^{\nu}\delta p_w^{\nu+1} + \frac{\partial r_c}{\partial z_{CO_2}}|^{\nu}\delta z_{CO_2}^{\nu+1}\,. \tag{3.38}$$

This leads to a system of linearized equations that can be expressed in a matrix form:

$$\underbrace{\begin{bmatrix} J_{CO_2,p_w} & J_{CO_2,z_{CO_2}} \\ J_{H_2O,p_w} & J_{H_2O,z_{CO_2}} \end{bmatrix}^{\nu}}_{\mathbf{J}^{\nu}} \underbrace{\begin{bmatrix} \delta p_w \\ \delta z_{CO_2} \end{bmatrix}^{\nu+1}}_{\delta\xi^{\nu+1}} = -\underbrace{\begin{bmatrix} r_{CO_2} \\ r_{H_2O} \end{bmatrix}^{\nu}}_{r^{\nu}}. \tag{3.39}$$

Here, $\mathbf{J}^{\nu}$ is the Jacobian matrix of partial derivatives and $\delta\xi^{\nu+1}$ is the vector of solution updates. Nonlinear convergence is reached when the residual norms and variable updates fall below user-specified tolerances [130]

$$\begin{aligned} &\left(\frac{\|r_{CO_2}^{\nu+1}\|_2}{\|r_{CO_2}^{0}\|_2} < \epsilon_{(r_{CO_2})} \vee \frac{\|r_{CO_2}^{\nu+1}\|_2}{\|\text{rhs}_{CO_2}\|_2} < \epsilon_{(r_{CO_2})}\right) \wedge \\ &\left(\frac{\|r_{H_2O}^{\nu+1}\|_2}{\|r_{H_2O}^{0}\|_2} < \epsilon_{(r_{H_2O})} \vee \frac{\|r_{H_2O}^{\nu+1}\|_2}{\|\text{rhs}_{H_2O}\|_2} < \epsilon_{(r_{H_2O})}\right) \wedge \\ &\left(\frac{\|\delta p_w\|_2}{\|p_w\|_2} < \epsilon_{(p_w)} \wedge \frac{\|\delta z_{CO_2}\|_2}{\|z_{CO_2}\|_2} < \epsilon_{(z_{CO_2})}\right). \end{aligned} \tag{3.40}$$

3.3.2. SOLUTION STRATEGY IN FRACTURED POROUS MEDIA

For fractured systems, the residual equations (3.16)–(3.17) are formulated separately for the matrix and fracture domains. For a matrix cell i, the residual is

$$r_{c,i}^{\nu+1} = \frac{V_i \phi_i}{\Delta t} \left(z_c^{\nu+1} \rho_T^{\nu+1} - z_c^{\nu} \rho_T^{\nu} \right)_i^m + \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j \in \mathcal{N}_m(i)} \left(F_{c,\alpha,ij}^{\nu+1} + J_{c,\alpha,ij}^{\nu+1} \right)^m - \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{f \in \mathcal{N}_f(i)} F_{c,\alpha,if}^{\nu+1} - \sum_{\alpha=1}^{n_{\text{ph}}} (x_{c,\alpha} q_\alpha)_i^{\nu+1} = 0, \quad \forall c \in \{1, \dots, n_c\}, \tag{3.41}$$

and for a fracture cell i

$$r_{c,i}^{\nu+1} = \frac{V_i \phi_i}{\Delta t} \left(z_c^{\nu+1} \rho_T^{\nu+1} - z_c^{\nu} \rho_T^{\nu} \right)_i^f + \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j \in \mathcal{N}_f(i)} \left(F_{c,\alpha,ij}^{\nu+1} + J_{c,\alpha,ij}^{\nu+1} \right)^f - \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{k \in \mathcal{N}_m(i)} F_{c,\alpha,ik}^{\nu+1} - \sum_{\alpha=1}^{n_{\text{ph}}} \sum_{j \in \mathcal{N}_f(i)} F_{c,\alpha,ij}^{f_i, f_j, \nu+1} = 0, \quad \forall c \in \{1, \dots, n_c\}. \tag{3.42}$$

Let $r^\nu = \left[(r^m)^\nu, (r^{f_1})^\nu \dots (r^{f_{n_{\text{frac}}}})^\nu \right]^T$ denote the global residual vector where $(r^m)^\nu$ is the residual vector of medium m at iteration ν. Correspondingly, the primary unknowns p_w and z_{CO_2} are defined for each control volume in both matrix and fracture domains. The residual $r_{\nu+1}$ is nonlinear in these variables, and thus, a Newton-Raphson method is employed at each time-step to solve the system iteratively

$$r_c^{\nu+1} \approx r_c^{\nu} + \frac{\partial r_c}{\partial p_w}\Big|^\nu \delta p_w^{\nu+1} + \frac{\partial r_c}{\partial z_{\text{CO}_2}}\Big|^\nu \delta z_{\text{CO}_2}^{\nu+1}. \tag{3.43}$$

The resulting linearized system takes the block structure

$$\underbrace{\begin{bmatrix} \begin{pmatrix} J^{m,m}_{\text{CO}_2,p_w} & J^{m,f}_{\text{CO}_2,p_w} \\ J^{f,m}_{\text{CO}_2,p_w} & J^{f,f}_{\text{CO}_2,p_w} \end{pmatrix} & \begin{pmatrix} J^{m,m}_{\text{CO}_2,z_{\text{CO}_2}} & J^{m,f}_{\text{CO}_2,z_{\text{CO}_2}} \\ J^{f,m}_{\text{CO}_2,z_{\text{CO}_2}} & J^{f,f}_{\text{CO}_2,z_{\text{CO}_2}} \end{pmatrix} \\ \begin{pmatrix} J^{m,m}_{\text{H}_2\text{O},p_w} & J^{m,f}_{\text{H}_2\text{O},p_w} \\ J^{f,m}_{\text{H}_2\text{O},p_w} & J^{f,f}_{\text{H}_2\text{O},p_w} \end{pmatrix} & \begin{pmatrix} J^{m,m}_{\text{H}_2\text{O},z_{\text{CO}_2}} & J^{m,f}_{\text{H}_2\text{O},z_{\text{CO}_2}} \\ J^{f,m}_{\text{H}_2\text{O},z_{\text{CO}_2}} & J^{f,f}_{\text{H}_2\text{O},z_{\text{CO}_2}} \end{pmatrix} \end{bmatrix}^\nu}_{\mathbf{J}^\nu} \underbrace{\begin{bmatrix} \delta p_w^m \\ \delta p_w^f \\ \delta z_{\text{CO}_2}^m \\ \delta z_{\text{CO}_2}^f \end{bmatrix}^{\nu+1}}_{\delta \xi^{\nu+1}} = - \underbrace{\begin{bmatrix} r^m_{\text{CO}_2} \\ r^f_{\text{CO}_2} \\ r^m_{\text{H}_2\text{O}} \\ r^f_{\text{H}_2\text{O}} \end{bmatrix}^\nu}_{r^\nu}, \tag{3.44}$$

where $\mathbf{J}^\nu$ includes all coupling terms between matrix and fracture unknowns across both components. The convergence criteria follow the same equation (3.40) as for the single-continuum case.

4

MULTI-TASK DEEP LEARNING WITH CONSISTENCY FOR CO_2 MIGRATION

Following the previous chapters, which established a comprehensive framework for simulating CO_2 injection and migration in deep saline aquifers using a fully coupled fine-scale compositional flow model, this chapter shifts focus toward data-driven surrogate modeling. This modeling framework integrates the overall-composition formulation, phase equilibrium calculations, and robust solution strategies for nonlinear systems. While such high-fidelity simulations can capture the complex physics of multiphase, multicomponent CO_2-brine systems, their computational cost becomes prohibitive when applied to uncertainty quantification, real-time decision-making, or large-scale field applications involving geological heterogeneity and nonlinear thermodynamic behavior. To address these limitations, we introduce an efficient surrogate modeling framework based on deep learning techniques, aimed at predicting the spatiotemporal evolution of key flow variables, specifically the dissolved CO_2-brine ratio (R_s) and gas saturation S_g in saline aquifers. These two quantities are important for assessing the performance of physical trapping mechanisms, such as dissolution and residual trapping, in long-term CO_2 storage. Rather than treating each variable independently, as in traditional single-task learning approaches, the proposed framework employs a multi-task learning with consistency (MTLC) paradigm that simultaneously predicts both variables while enforcing cross-task consistency informed by the underlying physical relationships.

This chapter presents the design, training, and evaluation of the MTLC model, which leverages an encoder-decoder architecture constructed using residual blocks. The model is trained using datasets generated from high-resolution compositional simulations, as described in the previous chapter, and is guided by domain knowledge of CO_2 trapping physics. Through joint learning and consistency-based regularization, the MTLC framework improves generalization and prediction accuracy, especially in scenarios with limited labeled data. The results demonstrate that the surrogate model captures the dynamic evolution of R_s and S_g with high fidelity while achieving sig-

Parts of this chapter have been published in the Advances in Water Resources, **Vol. 178** (2023) [47].

nificant computational acceleration compared to conventional finite volume solvers.

This chapter marks a transition from traditional numerical simulations to deep learning methods, providing a scalable and data-efficient alternative for modeling CO_2 migration and trapping in deep geological formations.

4.1. PROBLEM DEFINITION

This section defines the problem setting for developing a data-driven surrogate model aimed at predicting the dynamic behavior of CO_2 migration in deep saline aquifers. We first describe the fine-scale physical formulation based on a compositional multiphase flow model. This formulation serves as the foundation for generating high-fidelity simulation data using fully implicit numerical solvers. Subsequently, we formalize the surrogate modeling problem as a supervised learning task. The objective is to approximate the time-dependent response of two key variables, the solution CO_2-brine ratio (R_s) and the gas saturation (S_g), from spatially heterogeneous permeability fields. These quantities characterize the primary hydrodynamic trapping mechanisms of injected CO_2.

4.1.1. FINE-SCALE FORMULATION

For multi-component, multi-phase CO_2-brine flow problems, the mass conservation equation using the overall-composition formulation is presented in Chapter 3 as follows

$$\frac{\partial}{\partial t}(\phi \rho_T z_c) + \nabla \cdot (\sum_{\alpha} x_{c,\alpha} \rho_\alpha \mathbf{u}_\alpha) - \sum_{\alpha} x_{c,\alpha} \rho_\alpha q_\alpha = 0, \tag{4.1}$$

where the first term is the fluid accumulation, the second is the advective term, and the third is the source or sink term. The equation system is closed by enforcing the constraints $S_w + S_g = 1, z_{CO_2} + z_{H_2O} = 1$.

These governing equations are iteratively solved using a fully implicit numerical scheme, where the primary variables are the liquid-phase pressure (p_w) and CO_2 mole fraction (z_{CO_2}). Additionally, to simplify the problem formulation, molecular diffusion and hydrodynamic dispersion effects are not explicitly modeled; a common assumption in CO_2 storage simulations [14].

4.1.2. SURROGATE MODELING FOR CCS

The fully physics-based simulator, DARSim, numerically solves the above governing equations using a finite-volume method (FVM) with a fully implicit scheme [87]. We are concerned with the dynamics of interaction between CO_2 and brine, which is a challenging problem due to the complex interplay between viscous, capillary, and gravitational forces.

Moreover, in order to quantify the impact of different trapping mechanisms and uncertain reservoir properties, we design heterogeneous permeability realizations, and then calculate the amount of CO_2 trapped by dissolution and residual trapping, which are considered to be secure in hydrodynamic trapping. They are calculated based on the solution CO_2-brine ratio R_s and gas phase saturation maps S_g, respectively. In particular, R_s is described as

$$R_s = \frac{\rho^{\mathrm{STC}}_{\mathrm{brine}} x_{\mathrm{CO}_2,w}}{\rho^{\mathrm{STC}}_{\mathrm{CO}_2}(1 - x_{\mathrm{CO}_2,w})}. \tag{4.2}$$

However, solving the discretized version of equation (4.1) repeatedly for uncertainty quantification tasks becomes computationally prohibitive. Performing large ensembles of simulations involving high-dimensional, spatially heterogeneous domains with strong nonlinear interactions and frequent flash calculations is computationally expensive. Suppose the computational simulation is considered as a black-box mapping, a single simulation run can be described as

$$\mathbf{y} = f(\xi), \tag{4.3}$$

where $\xi \in \mathbb{R}^{n_s}$ denotes the high-dimensional realization of random field, n_s is the total number of grid blocks, f indicates the simulation-induced function, $\mathbf{y} \in \mathbb{R}^{2n_s n_t}$ is the dynamic response maps (R_s and S_g) at n_t time steps.

In order to resemble an inexpensive replacement of the numerical simulator, deep learning models with the $\mathbf{y} \approx \hat{\mathbf{y}} = \mathscr{F}(\xi, \theta)$ are trained with a limited simulation dataset $\{(\xi^i, \mathbf{y}^i)\}_{i=1}^N$ to approximate the relationship between the input properties ξ and the corresponding multiple dynamic responses $\mathbf{y}$, where θ are the deep neural network parameters, and N is the number of training simulation-based data. Therefore, the main task is transformed to an image-to-image regression problem which requires pixel-wise predictions as $\mathscr{F} : \mathbb{R}^{n_s} \rightarrow \mathbb{R}^{2n_s n_t}$. Therefore, our goal in this work is to develop a surrogate model to provide the time-dependent states $\hat{\mathbf{y}}$ given a permeability map ξ.

4.2. NEURAL NETWORK ARCHITECTURES

To effectively model the complex spatiotemporal behavior of CO_2 migration and trapping in heterogeneous saline aquifers, a series of deep neural network architectures are introduced in this section. The surrogate models are designed to approximate the high-dimensional, nonlinear mapping between geological input parameters and dynamic flow responses such as the solution CO_2-brine ratio (R_s) and gas saturation (S_g). We first describe the baseline single-task learning (STL) framework, which learns separate models for each output variable using convolutional encoder-decoder structures. To improve training depth and feature extraction capability, residual connections are then introduced through ResNet modules. Next, the multi-task learning (MTL) framework is presented to enable shared representation learning for multiple

outputs, thereby capturing task-relatedness and reducing overfitting. Finally, a multi-task learning architecture with cross-task consistency (MTLC) is proposed to further exploit the inherent physical relationship between R_s and S_g, thereby improving predictive reliability under limited labeled data.

4.2.1. SINGLE-TASK LEARNING (STL)

Given the input-output dataset $(\boldsymbol{\xi}, \mathbf{y})$, standard deep neural networks approximate the mapping $\hat{f}: \boldsymbol{\xi} \rightarrow \mathbf{y}$ using a sequence of fully connected layers, which can be expressed as follows

$$\mathbf{a}^l = h^l(\mathbf{a}^{l-1}) = \sigma^l(\mathbf{W}^l \mathbf{a}^{l-1} + \mathbf{b}^l), \tag{4.4}$$

where $\mathbf{W}^l$ and $\mathbf{b}^l$ are the weights and biases for the lth layer with input $\mathbf{a}^{l-1}$ and output $\mathbf{a}^l$, respectively. σ^l denotes the nonlinear activation function. Therefore, the approximate function is $\hat{f} = h^L \cdot h^{L-1} \cdots h^1$, where L is the depth of the network.

However, a fully connected neural network leads to an extremely large number of trainable parameters when dealing with the high-dimensional problems. Convolutional neural networks (CNNs) are commonly applied to reduce the number of parameters greatly due to the parameter sharing scheme. Therefore, they are widely used for image processing and are able to extract the features of inputs [131]. A convolutional layer consists of a series of convolution kernels which are used to compute the feature maps that are essentially matrices. Suppose that we have some $S \times S$ square neuron layer which is followed by convolutional layer. If we use an $m \times m$ filter $\boldsymbol{\omega}$, the output will be of size $(S-m+1) \times (S-m+1)$, and the feature value $h_{i,j}(\xi_{i,j})$ at location (i, j) is the sum of contributions (weighted by the filter components)

$$h_{i,j}(\xi_{i,j}) = \sigma\left(\sum_{k_i=0}^{m-1} \sum_{k_j=0}^{m-1} \omega_{k_i,k_j} \xi_{i+k_i,j+k_j}\right). \tag{4.5}$$

Therefore, the feature maps of a convolutional layer consisting of N_k filters are $\{\boldsymbol{h}^l, l = 1, ..., N_k\}$.

In contrast to traditional CNNs, which consist of a series of convolutional layers followed by fully connected layers, a popular model design pattern for pixel-wise predictions is the encoder-decoder architecture. It replaces the last fully-connected layers with upsampling or deconvolution layers to recover resolution, exhibiting promising performance in handling mappings between high-dimensional inputs and outputs [52]. The encoder-decoder neural network employs a coarse-refine process, where the encoder reduces spatial dimensions in every layer and increases channels to extract higher-level features at lower spatial resolution, while the decoder increases spatial dimensions and reduces channels to refine the image representation and construct the output. Ultimately, the spatial dimensions are restored to make predictions for each input image pixel [132]. This type of model has been successfully utilized in various fields, including computer vision for image segmentation [133]. Moreover, there are

also successful applications in the general context of multiphase flow [51, 63]. Therefore, a fully convolutional encoder-decoder architecture is employed to formulate our approach.

4.2.2. RESIDUAL NEURAL NETWORK (RESNET)

When dealing with deep CNNs to solve a complicated task, the general operation is to engage in stacking more layers. These additional layers help solve complex problems more efficiently as the different layers could be trained for varying tasks to get highly accurate results. While the number of stacked layers can enrich the feature of the model, a deeper network can suffer from degradation. In other words, as the number of layers of the neural network increases, the accuracy levels may get saturated and slowly degrade after some points. As a result, the performance of the model deteriorates both on the training and testing datasets. This degradation is the result of the problem of vanishing or exploding gradients [134].

In order to solve this problem, the ResNet architecture is introduced with the concept of Residual Blocks. In this network, a technique called shortcut connection is developed. As shown in the left subfigure of Figure 4.1, the shortcut connection connects activations of a layer to further layers by skipping some layers in between. This forms a residual block. ResNets are made by stacking these residual blocks together. The idea behind this is to learn the residual mapping (the difference between the input and output of a layer) instead of the actual mapping, making it easier for the network to learn the identity function and thereby mitigating the vanishing gradient problem in deep networks. Therefore, instead of fitting the original mapping $h(\xi)$, the network is trained to fit the residual mapping:

$$\mathscr{F}(\xi) := h(\xi) - \xi \,. \tag{4.6}$$

The advantage of adding this type of shortcut connection is that if any layer hampers the performance of the architecture, its negative effect can be mitigated by the residual connection. So, this results in training a very deep neural network without the problems caused by vanishing/exploding gradient. Therefore, the ResNet-V2 structure is employed as the network backbone. The approximated relationship is described as

$$\xi^{l+1} = \xi^l + \mathscr{F}(\sigma(\xi^l), \theta^l), \tag{4.7}$$

where σ denotes the pre-activation, ξ^l and θ^l are the input feature and hyperparameters to the l-th Residual Unit, respectively. In this way, the optimization is further eased and meanwhile, the pre-activation scheme enhances regularization effectiveness and helps reduce overfitting [135].

An illustration of the ResNet-V2 block is shown in Figure 4.2. The residual block has two convolutional layers with the same number of output channels. Each convolutional layer is preceded by a batch normalization layer and a ReLU activation function. Then, the input is added directly to the output of the two convolutional layers via a

Figure 4.1: Left: the illustration of a single residual block with a shortcut connection for maintaining feature information from previous layers and enhancing learning capacity. Right: the illustration of the general Multi-Task Learning (MTL) framework using deep neural networks for simultaneously learning multiple tasks. The shared encoder branches out to distinct decoders for each task, leveraging shared knowledge while allowing task-specific learning.

shortcut connection. This implies that the output of the two convolutional layers has to be of the same shape as the input, so that they can be added together. If we want to change the number of channels, we need to introduce an additional 1×1 convolutional layer to transform the input into the desired shape for the addition operation.

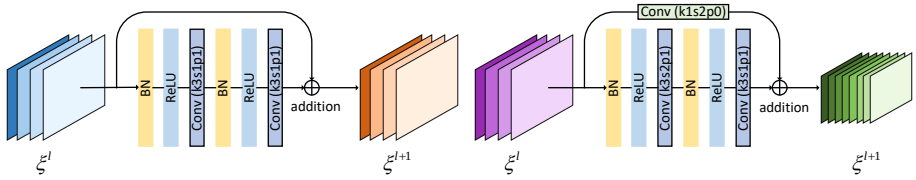


Figure 4.2: The basic residual blocks of ResNet-V2: without (left) and with (right) 1×1 convolution. The 1×1 convolution operation is used to transform the input into the desired shape for the addition operation.

The typical single task learning process of a neural network is described as follows: suppose ξ represents the input domain (i.e., permeability fields) and $\mathbf{y} = \{\mathbf{y}_1, \mathbf{y}_2\}$ is the set of desired prediction domains (i.e., R_s and S_g maps). The aim is to learn the approximate relationship between input and prediction domains:

$$\mathscr{F}_\xi = \{\hat{f}_{\xi\mathbf{y}_1}, \hat{f}_{\xi\mathbf{y}_2}\}, \tag{4.8}$$

where $\hat{f}_{\xi \mathbf{y}_1}$ and $\hat{f}_{\xi \mathbf{y}_2}$ outputs $\mathbf{y}_1$ and $\mathbf{y}_2$ given ξ, respectively. Training $\hat{f}_{\xi \mathbf{y}_1}(\xi)$ involves finding parameters that minimize the loss function, e.g., mean absolute error (MAE). As for $\hat{f}_{\xi \mathbf{y}_2}(\xi)$, the procedure is the same.

As shown in Figure 4.3, ResNet serves as the network backbone and combines Encoder-Decoder architecture to form the single task learning framework in this chapter, and the multi-task framework described later is also developed based on this framework. In particular, in order to capture temporal dynamics, we apply a three dimensional extension of the encoder-decoder architecture which means the convolutional kernels are 3-D that can extract information in both the temporal and spatial dimensions [136].

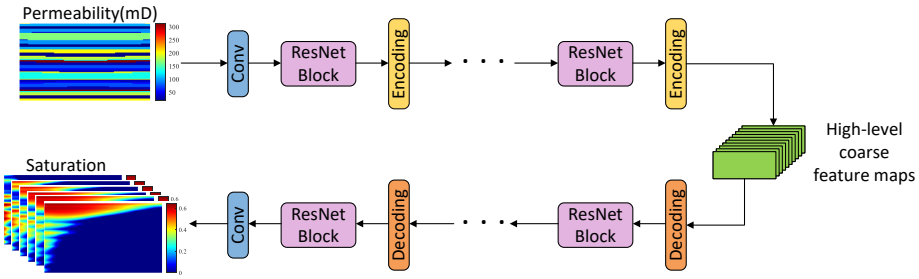


Figure 4.3: Single Task Learning (STL) network architecture, employing an encoder-decoder structure, designed specifically for predicting gas saturation maps. This schematic shows the path of data through the network, highlighting the key components of the architecture.

4.2.3. MULTI-TASK LEARNING (MTL)

Currently, most methods for multiphase flow are focused on only one of these tasks (i.e., saturation or pressure maps), and they also achieve the state-of-the-art performance through the technique of deep learning. However, there may be instances when learning from many related tasks at the same time would lead to better modeling performance [137]. This is addressed in the domain of multi-task learning, a subfield of machine learning in which multiple objectives are trained within the same model simultaneously. Compared to the single-task methods where each individual task is solved separately by its own network, recently, several multi-task learning methods in computer vision have shown a promising direction to improve the predictions by jointly tackling multiple tasks to mutually enhance performance across tasks [138].

Compared to single-task learning, the loss function of typical multi-task learning is described by

$$\mathscr{L} = \left|\hat{f}_{\xi \mathbf{y}_1}(\xi) - y_1\right| + \left|\hat{f}_{\xi \mathbf{y}_2}(\xi) - y_2\right|, \tag{4.9}$$

where $|\cdot|$ denotes the MAE value which is also referred to as L_1 norm. Moreover, the first and the last terms are the standard losses for training $\hat{f}_{\xi \mathbf{y}_1}$ and $\hat{f}_{\xi \mathbf{y}_2}$, respectively.

MTL is used in many fields and joint learning is considered to give an improved representation than other STL since this method can capture the intrinsic association features between tasks [139]. In the context of deep learning, MTL is performed by learning shared representations from multiple supervisory tasks. Classical deep multi-task architectures were hard parameter sharing techniques in which the parameter set is divided into shared and task-specific parameters, as shown in the right subfigure of Figure 4.1. MTL models typically consist of a shared encoder that branches out into task-specific heads. Multi-task learning jointly fits all tasks, which acts as a form of regularization, thereby reducing the risk of overfitting to a single task learning.

4.2.4. MULTI-TASK LEARNING WITH CONSISTENCY (MTLC)

The predictions of R_s and S_g maps are important and challenging for CO_2 trapping in deep saline aquifers. The general objective of multi-task learning is to improve generalization by leveraging the domain-specific information contained in the training signals of related tasks. Common multi-task learning approaches are a shared feature extractor component (encoder) with multiple "heads" (decoder) that perform separate tasks. Common fusion and feature-sharing schemes can leverage the correlations between tasks, but there exist some drawbacks. For example, the integration of different features might result in ambiguity of information; the fusion does not explicitly model the task-level interaction where we do not know what information is transmitted [140].

The prediction tasks for R_s and S_g correspond to distinct but physically related responses governed by the same underlying permeability field. Hence inconsistency among predictions implies contradiction and is inherently undesirable. Furthermore, these two parameters have similar spatiotemporal patterns due to the same CO_2 trajectory. This similarity pattern between the related tasks are informative and can be used to better fit the data. The dynamic processes of R_s and S_g are connected by the underlying physics and consequently enforce some constraints on each other, referred to as consistency constraints [141]. In this work, the consistency constraints are learned from data rather than an a prior given relationship, which makes the method applicable to any pairs of tasks that are not independent, particularly when their analytical relationship is unknown or difficult to formulate.

The independent single-task learning satisfies various ideal properties, if given an infinite amount of labeled data. However, there is only limited amount of expensive labeled data in practice. Thus, a label-efficient multi-task learning method is presented which introduces the concept of cross-task consistency. The difference between single-task network, common MTL network and MTL with consistency (MTLC) is shown in Figure 4.4. The MTLC consists of one shared encoder, two task-specific decoders, and an additional encoder-decoder pair for the transformation between R_s and S_g. The shared encoder is responsible for processing the permeability fields and generating a shared representation for both tasks. Then, the encoder branches into two decoders, one for each task. These decoders receive the shared representation and transform it into a task-specific output. Each decoder is specialized in generating

output for its respective task. Finally, an additional encoder-decoder pair is designed to enforce consistency between tasks. The encoder in this pair takes the output S_g and generates a representation, which is then transformed into R_s by the corresponding decoder. This additional encoder-decoder pair helps in ensuring that the learned representations for S_g and R_s are consistent with each other. The detailed encoder and decoder architecture is shown in Table 4.1. It is worth noting that the number of encoders and decoders varies between the single-task learning (STL), multi-task learning (MTL), and multi-task learning with consistency (MTLC) models. However, the layers within the encoder and decoder remain the same across all three models. This ensures that the architectural differences between the models are primarily related to the handling of tasks, while maintaining consistency in the internal structure of the encoders and decoders.

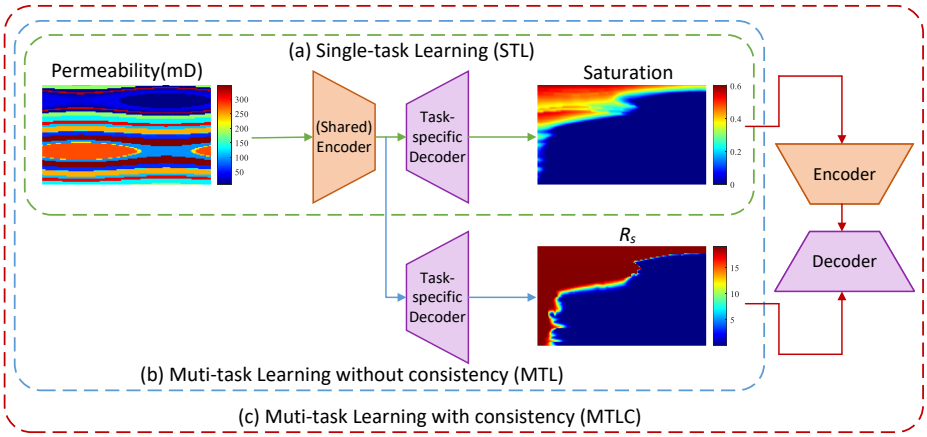


Figure 4.4: The illustration of the differences between STL, MTL and MTLC. MTLC incorporates an additional encoder-decoder architecture to enforce cross-task consistency, encouraging the learning process to better capture and utilize the shared underlying structures between tasks.

Therefore, an additional constraint to guide the training toward cross-task consistency is introduced. In addition to $\hat{f}_{\xi\mathbf{y}_1}$ and $\hat{f}_{\xi\mathbf{y}_2}$, the function $\mathscr{F}_{\mathbf{y}} = \{\hat{f}_{\mathbf{y}_1\mathbf{y}_2}\}$ is also defined which is the set of cross-task functions that map the prediction domains onto each other. The loss function for training the MTLC is defined as

$$\mathscr{L}_{\xi\mathbf{y}_1\mathbf{y}_2} = \left|\hat{f}_{\xi\mathbf{y}_1}(\xi) - y_1\right| + \left|\hat{f}_{\mathbf{y}_1\mathbf{y}_2} \circ \hat{f}_{\xi\mathbf{y}_1}(\xi) - \hat{f}_{\xi\mathbf{y}_2}(\xi)\right| + \left|\hat{f}_{\xi\mathbf{y}_2}(\xi) - y_2\right|, \tag{4.10}$$

where the middle term is the consistency term which enforces that predicting $\mathbf{y}_2$ out of the predicted $\mathbf{y}_1$ yields the same result as directly predicting $\mathbf{y}_2$ out of ξ. This part of the network is aimed at discovering the similar patterns from the data. This consistency constraint helps the model generalize better by focusing on the common aspects of the tasks rather than learning task-specific features. It also helps regularize the model, preventing overfitting, and allowing it to exploit complementary information provided

Table 4.1: Encoder-Decoder architecture details. k denotes the kernel size; s denotes the stride, c denotes the number of kernels; Conv denotes convolutional layer; Residual block is described in figure 4.2.

Net	Layer	Output size
	Input	(96,200,6,1)
Encoder	Conv, k=(3×3×3), s=(2×2×1), c=16	(48,100,6,16)
	Residual block, k=(3×3×3), s=(1×1×2), c=32	(48,100,3,32)
	Residual block, k=(3×3×3), s=(2×2×1), c=64	(24,50,3,64)
	Residual block, k=(3×3×3), s=(1×1×2), c=128	(24,50,2,128)
	Residual block, k=(3×3×3), s=(2×2×1), c=256	(12,25,2,256)
	Residual block, k=(3×3×3), s=(1×1×1), c=256	(12,25,2,256)
Decoder	Upsampling/Residual block, k=(3×3×3), s=(2×2×1), c=128	(24,50,2,128)
	Upsampling/Residual block, k=(3×3×3), s=(1×1×2), c=64	(24,50,3,64)
	Upsampling/Residual block, k=(3×3×3), s=(2×2×1), c=32	(48,100,3,32)
	Upsampling/Residual block, k=(3×3×3), s=(1×1×2), c=16	(48,100,6,16)
	Upsampling/Residual block, k=(3×3×3), s=(2×2×1), c=8	(96,200,6,8)
	Conv, k=(3×3×3), s=(1×1×1), c=1	(96,200,6,1)

by both tasks. Therefore, the learning of predicting $\mathbf{y}_1$ and $\mathbf{y}_2$ is not independent anymore and a single deep learning framework performs simultaneous training of two tasks. In this work, the S_g and R_s maps are considered as $\mathbf{y}_1$ and $\mathbf{y}_2$, respectively.

4.3. NUMERICAL EXPERIMENTS AND RESULTS

This section presents numerical experiments to evaluate the performance of the proposed MTLC framework for predicting CO_2 migration in deep saline aquifers. We begin by describing the generation of simulation datasets based on high-resolution compositional modeling, capturing the effects of permeability heterogeneity and multiphase interactions. The training procedures for different neural architectures are then detailed, followed by performance comparisons between STL, MTL, and MTLC models across various training sizes. Finally, we assess the capability of the models in uncertainty quantification, demonstrating that MTLC offers accurate, efficient, and physically consistent predictions suitable for CCS applications.

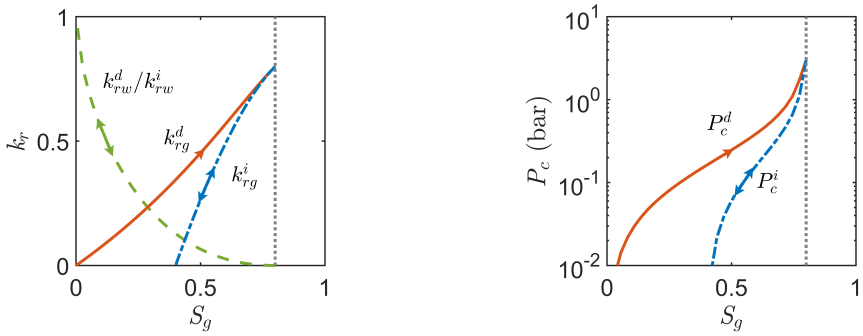
4.3.1. DATASETS GENERATION

In order to illustrate the performance of our method for predicting the CO_2 migration in deep saline aquifers, a 2-D CCS simulation system is considered. From a geological perspective, saline aquifers often have a dominant length in the longitudinal direction compared to their cross-sectional width. Since gravity plays an essential role in the CO_2 trapping process, the 2D xz representation is preferred to xy representation in order to retain important vertical heterogeneity and fluid migration behavior. The

model simulates the migration of CO_2 within a 2-D vertical cross-sectional aquifer domain. The dimensions are 200 m × 96 m, with a grid block size of 1 m × 1 m. The initial pressure is set to be 2.5×10^7 Pa with a constant temperature of 300 K. No flow boundary conditions are imposed on all sides of the aquifer. CO_2 is injected through the bottom 48 m of the domain on the left side at a constant rate of 2×10^{-4} pore volumes per day. A production well is placed on the entire right side. The total simulation time and injection time are 600 days. The nonlinear relative permeability curves and capillary pressure curves are shown in Figure 4.5. Physical parameters and simulation settings are presented in Table 4.2.

Table 4.2: Depiction of the physical parameters and simulation setup utilized in the DARSim Numerical Simulation.

Parameter	Value	Unit
Aquifer length	200	m
Aquifer height	96	m
Porosity	0.2	-
CO_2 injection rate	2×10^{-4}	pore volumes per day
Initial pressure	2.5×10^7	Pa
Bottom hole pressure	2.5×10^7	Pa
Temperature (isothermal)	300	K
CO_2 density at STC	1.98	kg/m^3
Brine density at STC	1060	kg/m^3
Brine salinity	1×10^5	parts per million



(a) Relative permeability curves

(b) Capillary pressure curves

Figure 4.5: (a) Primary drainage and imbibition curves for liquid and gas phases, where the superscripts d and i represent drainage and imbibition, respectively. A single-headed arrow indicates that the process along a given curve is irreversible, while a double-headed arrow signifies that the process is reversible. (b) Illustration of drainage/imbibition capillary pressure curves.

To mimic the geological formations used for CO_2 sequestration, a total of 1400 permeability fields are generated using the open-source package Stanford Geostatistical

Modeling Software (SGeMS) [142] with laterally correlated heterogeneity. The distribution of log-transformed permeability and one random realization of the permeability field are shown in Figure 4.6. The forward simulation is performed using DARSim and we collect the output state maps (R_s and S_g maps) at prescribed time steps respectively. Each simulation takes 4800 seconds on an Intel Core i7-12700K.

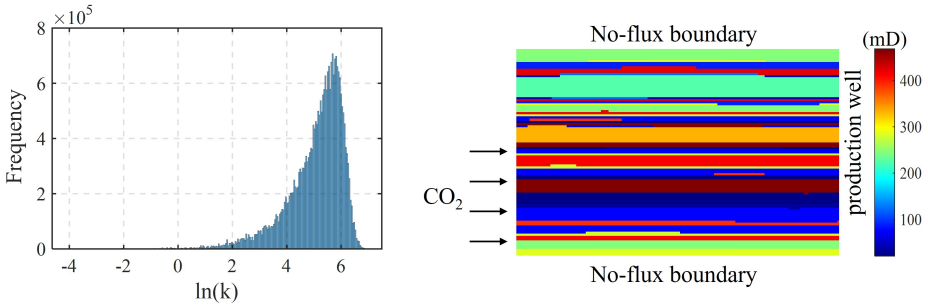


Figure 4.6: Left: the histogram representing the distribution of the 1400 log-permeability fields used in both training and test datasets, showing the variation and spread of the data. Right: visual representation of a permeability realization where CO_2 is injected from cells on the bottom-left boundary, demonstrating the practical application of the field.

We are interested in the spatial-temporal evolution of the R_s and S_g maps during CO_2 injection period. Thus, we collect the output state maps at six uniform time instances to train the networks. The goal is to train networks with the simulation data that yield reliable predictions of R_s and S_g on the hitherto unseen permeability fields in the test set.

4.3.2. TRAINING PROCEDURES

The loss function used for the training models is the Mean Absolute Error (MAE), defined as

$$\mathscr{L}_{\text{MAE}} = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{\mathbf{y}}_i - \mathbf{y}_i \right\|_1, \tag{4.11}$$

where N is the total number of geo-models in the training set. In the training process, the loss function is minimized by tuning the network parameters θ. The gradient of the loss function with respect to θ is automatically computed by back-propagation [143]. In this work, the batch size is set to 16, and the adaptive moment estimation (Adam) optimization algorithm [144] is used with the initial learning rate of 0.001 and a decay factor of 0.1. This has been found to be an effective procedure for the training of many deep neural network architectures. Moreover, we utilize varying sizes of training datasets with 300, 600, 900, and 1200 permeability realizations to train the models. A separate, fixed set of 200 permeability realizations, not included in the

training data, is used for testing the performance of all trained models. Due to the limited size of the dataset, we have not employed a separate validation dataset.

During the training process, the MAE metric is used to monitor the convergence of both the training and test errors. Additionally, to evaluate the quality of the trained models, we also consider two commonly used metrics, the root mean square error (RMSE) metric and the coefficient of determination (R^2) evaluated on 200 predefined test samples. The RMSE is calculated by

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \left\| \hat{\mathbf{y}}_i - \mathbf{y}_i \right\|_2^2}. \tag{4.12}$$

The R^2 is defined as

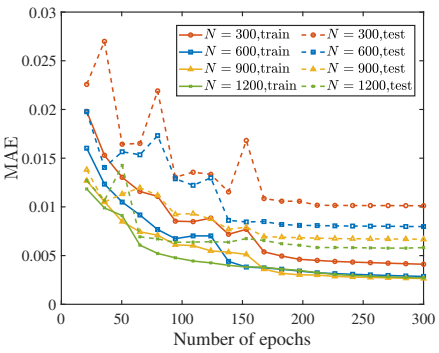
$$R^2 = 1 - \frac{\sum_{i=1}^{N_{\text{test}}} \left\| \hat{\mathbf{y}}_i - \mathbf{y}_i \right\|_2^2}{\sum_{i=1}^{N_{\text{test}}} \left\| \mathbf{y}_i - \bar{\mathbf{y}} \right\|_2^2}, \tag{4.13}$$

where $\bar{\mathbf{y}}$ denotes the average of the reference values. In particular, smaller MAE and RMSE values represent better performance, and an R^2 value closer to 1 indicates better prediction quality.

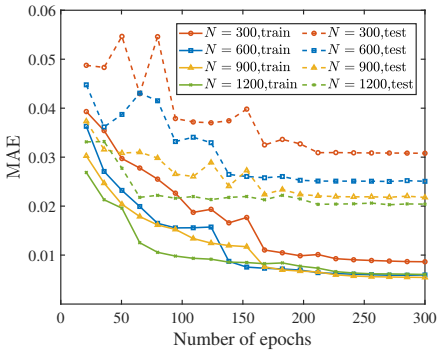
4.3.3. TRAINING AND TESTING RESULTS

Figure 4.7 shows the MAE value of MTLC training and testing loss with the number of epochs and for training ensemble sizes of 300, 600, 900, and 1200 realizations. The models are trained on a NVIDIA GeForce RTX 3080 GPU which takes approximately 2000 to 7000 s to complete 300 training epochs, depending on the number of training samples (ranging from 300 to 1200). The training data generation and training times for the deep learning models are part of the initial setup process and need only be considered once. After training, given a new realization, the MTLC can provide predictions for the state maps, at 6 time steps, in an elapsed time of about 19 ms. This speed-up is 5 orders of magnitude faster than the numerical simulator, which is attributed to the benefits of using a GPU for the deep learning model's execution. The GPU's parallel processing capability and the extensive support provided by modern deep learning frameworks significantly contribute to the efficiency advantage of the MTLC model over the traditional numerical simulator. Therefore, the MTLC model becomes beneficial when a large number of realizations for uncertainty quantification or optimization. It is observed that the MAE starts to stabilize after 200 epochs for both R_s and S_g. For the testing dataset, the MAE for R_s is approximately 0.02 and for S_g is 0.005. The error for R_s is relatively large due to the presence of response discontinuity, which is a well-known challenge for other surrogate models, and this error can be considered acceptable in the context of CCS.

The performance of our MTLC model in approximating the time-dependent multi-output predictions is further demonstrated in Figures 4.8 and 4.9, which depict a comparison of the R_s and S_g fields at various time instances (100, 200, 300, 400,



(a) Loss curves for S_g



(b) Loss curves for R_s

Figure 4.7: (a) Evolution of Mean Absolute Error (MAE) with the number of epochs in both the training and test sets for the task S_g, using different training sample sizes (N = 300, 600, 900 and 1200). (b) Evolution of MAE with the number of epochs in both the training and test sets for the task R_s, using different training sample sizes.

500 and 600 days) predicted by DARSim and our MTLC model using 1200 training samples. As expected, the model achieves high approximation accuracy for both R_s and S_g fields over time.

4.3.4. PERFORMANCE COMPARISON

To demonstrate the efficiency and effectiveness of the MTLC framework, we compare the performance of MTLC, MTL and STL. Each model is trained using different numbers of training samples, 300, 600, 900, 1200 realizations, respectively. After training, the prediction times for STL, MTL, MTLC are 14 ms, 17 ms and 19 ms, respectively, with negligible differences. It is worth noting that for STL, the model evaluation needs to be performed separately for R_s and S_g, while MTL and MTLC only need to be done once.

The R^2 scores for the test dataset for each model with different training data set sizes are shown in Figure 4.10. The figure shows that the model achieves a relatively high R^2 value of 0.9394 for the S_g problem and 0.8807 for R_s with only 300 training samples, even with high input dimensions and the presence of response discontinuity. These values are much higher than those of the STL (0.9140 and 0.8366). When increasing the sample size to 1200, the model achieves R^2 values of 0.9762 and 0.9335 for S_g and R_s, respectively. We also performed the R^2 score test only for the plume grid cells which have non-zero values in the reference maps from the numerical simulator, to demonstrate the significance of the difference. The R^2 scores for the plume area corresponding to R_s are 0.8616, 0.8226, and 0.7876 for MTLC, MTL, and STL, respectively. Meanwhile, the R^2 scores for S_g are 0.9574, 0.9401, and 0.9488, respec-

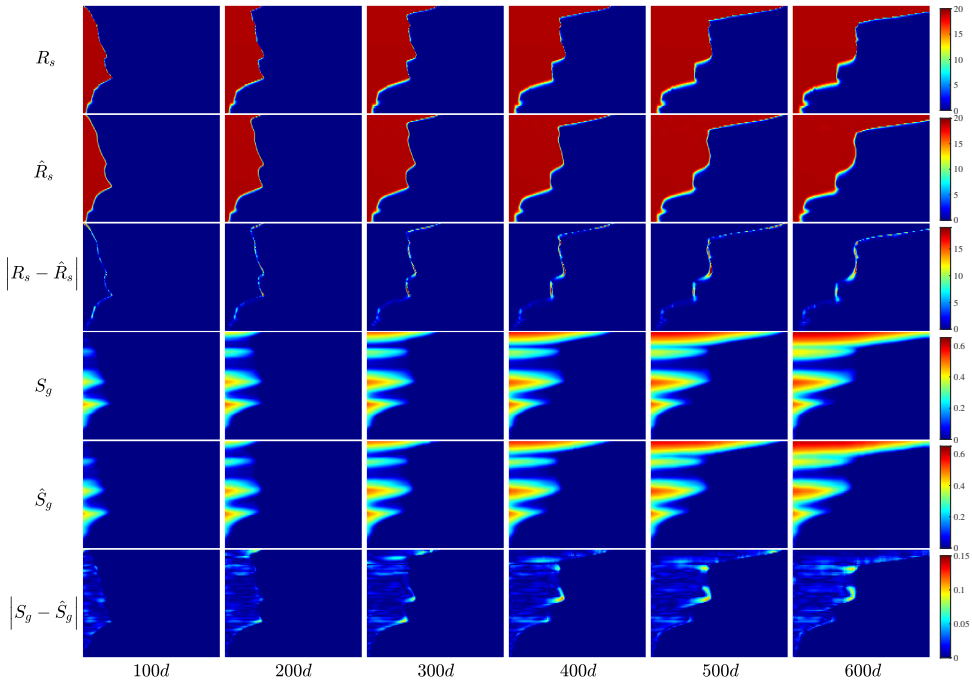


Figure 4.8: Example 1: Snapshots of the R_s and S_g fields at several time instances solved by DARSim and the corresponding $\hat{R}_s$ and $\hat{S}_g$ predicted by the MTLC model trained on 1200 samples, demonstrating the high accuracy of the MTLC model in approximating the dynamic states of these fields.

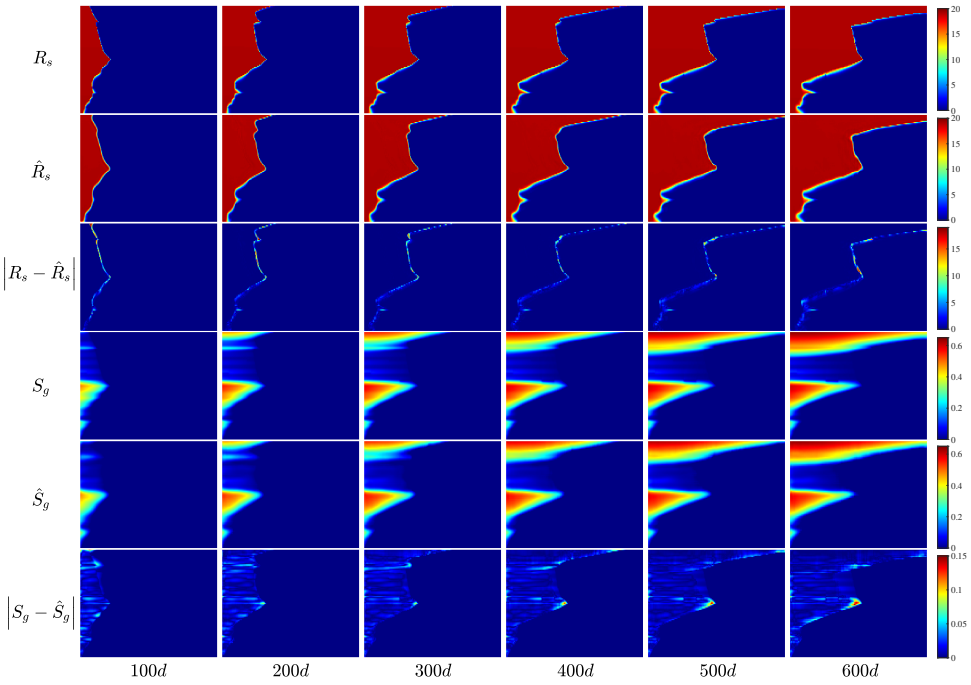
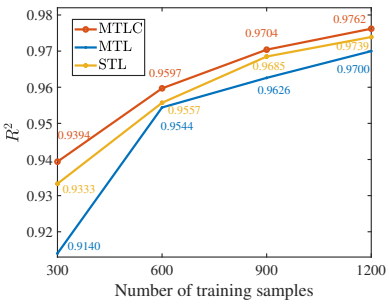
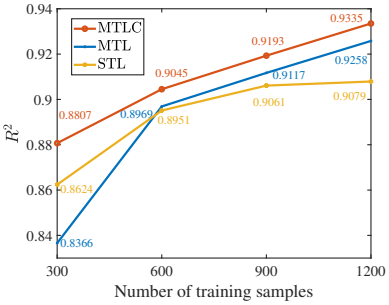


Figure 4.9: Example 2: Another set of snapshots of the R_s and S_g fields at several time instances solved by DARSim and the corresponding $\hat{R}_s$ and $\hat{S}_g$ predicted by the MTLC model using 1200 training samples, further validating the effectiveness of the MTLC model.

tively. It is also observed that the performance of MTL for R_s is better than STL, while that for S_g is worse than STL. This phenomenon proves that common hard parameter sharing scheme could compromise the performance of any task. In contrast, the MTLC framework improves the performance of both tasks due to the consistency constraints.



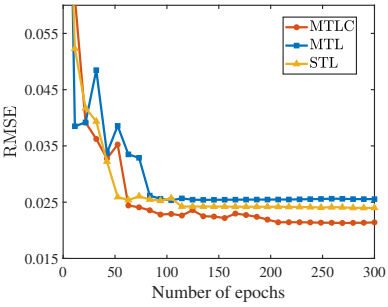
(a) R^2 scores for S_g



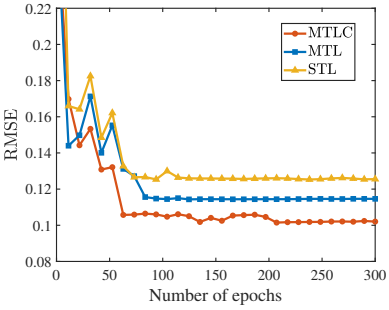
(b) R^2 scores for R_s

Figure 4.10: (a) Comparison of the R^2 scores for S_g for networks trained using different methods (STL, MTL, MTLC) evaluated on 200 test samples with different numbers of training samples. (b) Comparison of the R^2 scores for R_s for networks trained using different methods (STL, MTL, MTLC) evaluated on 200 test samples with different numbers of training samples. These show the relative performance of each method.

The evolutions of testing RMSE for three networks trained with 1200 training samples are shown in Figure 4.11. It is evident that the MTLC has a higher convergence speed for R_s. We also observe that the results of MTLC outperform those of the other models for both tasks.



(a) RMSE for S_g



(b) RMSE for R_s

Figure 4.11: (a) The illustration of RMSE decay with the number of epochs for STL, MTL, and MTLC on the task of predicting S_g using 1200 training samples. (b) The illustration of RMSE decay with the number of epochs for STL, MTL, and MTLC on the task of predicting R_s using 1200 training samples.

In Figure 4.12, we present an example of the predictions in the test dataset at 600 days. The MTLC framework not only achieves the best performance compared to STL and MTL, but also provides the more accurate positions of the R_s and S_g fronts, making it a more reliable model for predicting CO_2 saturation and CO_2-brine ratio in the context of CCS projects. The accuracy of the MTLC framework is further quantified by calculating the quantities of injected CO_2 trapped by dissolution trapping and residual trapping for the test case shown in Figure 4.12. As depicted in Figure 4.13, it is evident that the MTLC framework provides the most consistent predictions for both dissolution and residual trapped CO_2 quantities when compared to the results from the numerical simulator. This highlights the effectiveness of the MTLC framework for predicting CO_2 trapping in CCS projects.

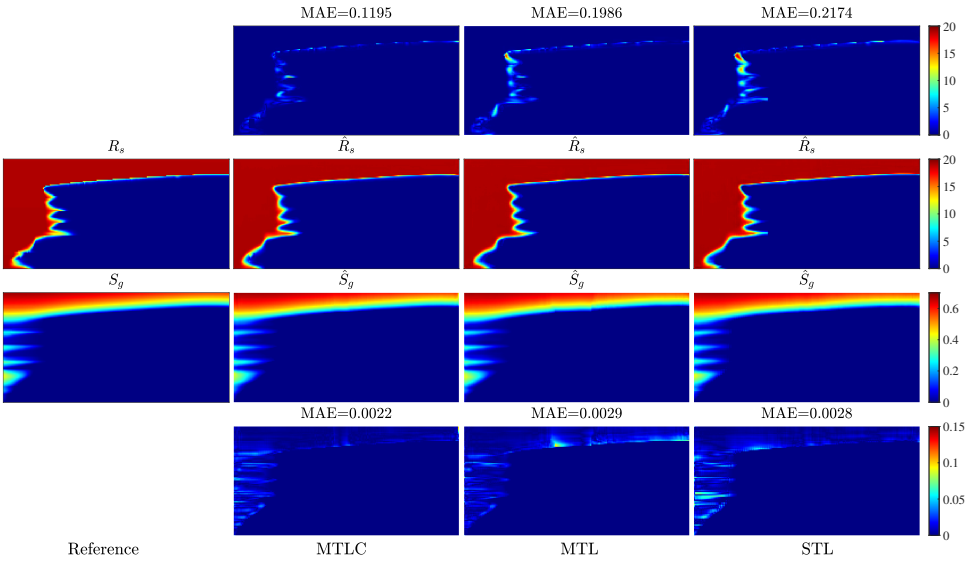
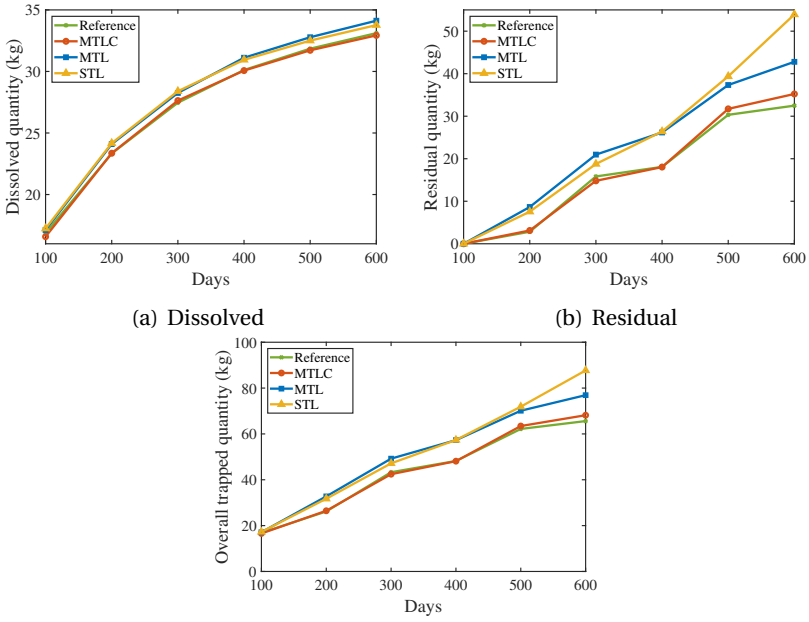


Figure 4.12: Testing results for the three models (STL, MTL, and MTLC) at 600 days. The top and bottom rows present the absolute error with respect to the reference, demonstrating the high accuracy of the MTLC model.

4.3.5. UNCERTAINTY MODELING

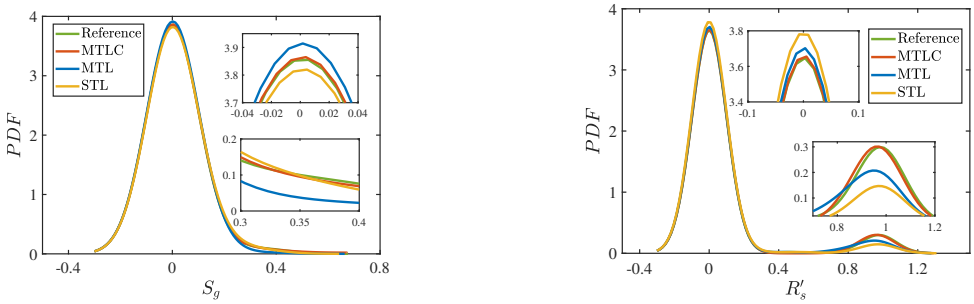
In this study, we compared the performance of three different network structures - STL, MTL, and MTLC - in an uncertainty quantification task focused on estimating R_s and S_g. Figure 4.14 displays the estimated distributions of 200 test permeability realizations for each method at location (48 m, 100 m) after 600 days. The probability density functions (PDFs) of S_g and R_s obtained using the MTLC framework were nearly indistinguishable from those obtained using the numerical simulator. This result highlights the effectiveness of the MTLC model in handling uncertainty quantification tasks within the CCS domain, demonstrating its potential for practical application in similar contexts.



(a) Dissolved (b) Residual

(c) Overall

Figure 4.13: (a) Comparison of dissolution trapping quantities as predicted by the numerical simulator (reference - green lines) and three models - MTLC (red lines), MTL (blue lines), and STL (yellow lines). (b) Comparison of residual trapping quantities between the numerical simulator and different models. (c) Comparison of overall trapping quantities between the numerical simulator and different models.



(a) PDF for S_g (b) PDF for R_s

Figure 4.14: (a) Probability density functions (PDF) of S_g at the specific location (48 m, 100 m) at 600 days, as estimated by numerical simulator and the three model variations (STL, MTL, MTLC). (b) PDFs of R_s at the same location and time point. These comparisons showcase how closely each model can replicate the distributions as generated by the numerical simulator.

The results presented in this study demonstrate that the proposed MTLC method can provide accurate solutions for dynamic CCS simulation in heterogeneous saline aquifers. Therefore, it is able to be regarded as a data augmentation technique, being able to improve the accuracy of predictions with limited data.

4.4. CONCLUSIONS

This chapter presented a deep learning-based surrogate modeling strategy for simulating multiphase, multicomponent CO_2 migration in deep saline aquifers, with a particular focus on improving predictive performance and generalization through multi-task learning with consistency (MTLC). Following the fine-scale compositional flow formulation and high-fidelity numerical simulations established in the previous chapters, we designed an end-to-end surrogate framework that simultaneously predicts key state variables, the CO_2-brine solution ratio (R_s) and gas phase saturation (S_g) from heterogeneous permeability fields. These variables are of primary importance for quantifying the amount of CO_2 immobilized through dissolution and residual trapping mechanisms, which are dominant under hydrodynamic conditions in geological carbon storage.

The central innovation of this study lies in the integration of a multi-task learning paradigm with a cross-task consistency constraint, which leverages the inherent physical coupling between R_s and S_g. Instead of treating each variable independently, as in conventional single-task surrogate models, our MTLC approach enforces coherence between multiple outputs, thereby regularizing the learning process and improving generalization across unseen geological scenarios. The deep neural network is constructed using an encoder-decoder architecture with residual connections (ResNet blocks), which effectively captures spatial correlations within the input permeability fields and allows for high-resolution, time-dependent prediction of the target variables. The model is trained on simulation-generated data produced using a compositional finite-volume solver, ensuring physical consistency and fidelity.

The numerical experiments validate the effectiveness of the MTLC model. Compared with single-task baselines, the MTLC surrogate consistently yields lower prediction errors and better preserves the temporal and spatial characteristics of R_s and S_g fields across test cases. These include scenarios characterized by complex interactions among gravitational forces, viscous flow dynamics, and heterogeneity-induced preferential pathways. Moreover, the target variables reflect physically interrelated migration processes governed by the same permeability structure, the MTLC framework improves sample efficiency, reduces the need for large labeled datasets, and enhances extrapolative capability in data-scarce scenarios. Furthermore, the surrogate model reduces computational costs by several orders of magnitude relative to conventional fully implicit solvers, enabling efficient large-scale evaluations for uncertainty quantification, sensitivity analysis, and optimization under geologic uncertainty.

5

A PHYSICS-CONSTRAINED NEURAL NETWORK FOR FULL-CYCLE MIGRATION

The previous chapter presented a deep learning-based surrogate modeling framework for simulating CO_2 migration in deep saline aquifers. Specifically, a multi-task learning approach was developed to predict both the solution CO_2-brine ratio (R_s) and gas saturation (S_g), which are critical for quantifying hydrodynamic trapping. By leveraging correlations between these two state variables and training on high-fidelity simulation data, the proposed MTLC model achieved improved prediction accuracy and substantial computational speedup, demonstrating its potential for accelerating uncertainty quantification and sensitivity analysis in geological CO_2 storage.

Despite these advantages, the MTLC framework remains fundamentally pure data-driven and does not explicitly encode the governing physics that regulate multiphase flow in porous media. Moreover, it is constrained to make predictions at fixed training times steps and lacks the ability to generalize to future times. This limitation restricts the model's applicability in forward forecasting scenarios, which are essential for evaluating long-term CO_2 migration and storage security.

To address this gap, this chapter introduces a physics-constrained surrogate modeling approach that integrates the governing partial differential equations of mass conservation directly into the training of deep neural networks. We present CO_2PCNet, a physics-constrained neural network for CO_2 storage during injection and post-injection periods. Unlike conventional purely data-driven models, which may yield physically inconsistent results, especially when extrapolating beyond the training domain, CO_2PCNet embeds conservation laws as soft constraints in the loss function. This integration helps guide the network to produce predictions that are not only data-consistent but also follow the fundamental physics governing multiphase flow in porous media.

Parts of this chapter have been published in the Advances in Water Resources, **Vol. 193** (2024) [12].

The architecture of CO_2PCNet combines convolutional encoders for spatial feature extraction from pressure and composition fields, a ConvLSTM module for modeling temporal evolution, and a decoder for reconstructing dynamic spatial fields of interest. Permeability heterogeneity is incorporated as a conditioning input, enabling the network to adapt to varying geological structures. By embedding the physics directly into the learning objective, the model achieves improved generalization and robustness when extrapolating to time intervals and geological conditions beyond those included in the training data.

This chapter therefore marks a further advancement in hybrid modeling strategies for subsurface flow, transitioning from data-driven learning to physics-informed surrogates [62]. The proposed CO_2PCNet framework enhances both the interpretability and reliability of deep learning in CO_2 storage applications, complementing the MTLC model by enabling time-resolved and physically consistent predictions across a broader range of operational scenarios.

5

5.1. PROBLEM DEFINITION

We consider an isothermal, two-component (CO_2-H_2O), two-phase (wetting brine-rich and non-wetting CO_2-rich) flow in saline aquifers. The governing equations are given in Chapter 2. In this chapter, the model is specialized by neglecting molecular diffusion, i.e., the diffusive fluxes are set to zero ($\mathbf{j}_{c,\alpha} = \mathbf{0}$), and only advective transport driven by Darcy flow is included. The liquid phase pressure (P_l) and the CO_2 mass fraction, z_{CO_2}, serve as primary variables for the system.

High-fidelity reference solutions for the scenarios considered in this chapter are generated using DARSim, a fully implicit finite-volume compositional simulator [145].

Given the significant influence of capillary pressure and the dissolution capability of CO_2 in brine, the mechanisms by which CO_2 can be retained in saline aquifers have been outlined by Metz (2005) [13]. In addition, we assume that CO_2 from the gas phase can dissolve in the liquid phase, but dissolution of brine in the gas phase is neglected (i.e. the gas phase is assumed to contain only CO_2).

5.1.1. DISSOLUTION

The solubility of CO_2 in brine is influenced by pressure, temperature and water salinity. For an isothermal system with a constant salinity, the CO_2-brine solution ratio, R_s, quantifies the amount of CO_2 that can be dissolved per unit volume of brine. Importantly, within each simulation cell, the potential quantity of dissolved CO_2 is constrained by the existing amount of CO_2. This highlights the significance of discerning the number of active phases in the cell [121].

If a cell is in a two-phase state, the dissolved CO_2 amount can be extracted from a predefined CO_2-brine solution ratio curve, which is calculated prior to simulation, as shown in Figure 5.1A. Conversely, in an undersaturated state where only the liquid

phase is present, R_s is determined using the mass fractions of the CO_2 component, i.e.,

$$R_s = \frac{\rho_{\mathrm{b}}^{\mathrm{STC}} z_{\mathrm{CO_2}}}{\rho_{\mathrm{CO_2}}^{\mathrm{STC}} (1 - z_{\mathrm{CO_2}})}, \tag{5.1}$$

where the subscript 'b' denotes brine, while the superscript 'STC' signifies properties measured at standard conditions.

The density of the undersaturated aqueous phase is then updated using the solution-ratio relation,

$$\rho_b = \frac{\rho_b^{\mathrm{STC}} + R_s \rho_{\mathrm{CO_2}}^{\mathrm{STC}}}{B_b}, \tag{5.2}$$

where B_b is the brine formation volume factor.

5.1.2. CAPILLARITY

Capillary pressure, which is defined as the pressure difference between the wetting (liquid) and non-wetting (gas) phases, plays a crucial role in CO_2 storage. The interactions of these phases and the transport behavior of the matrix are governed by the relative permeability and capillary pressure functions, which are fundamental for the residual trapping mechanism. For example, rocks with low permeability and finer pore radius tend to display elevated capillary pressures compared to their high-permeability counterparts containing similar fluids. Such characteristics often result in CO_2 accumulating beneath structural and stratigraphic barriers at high saturations, especially when buoyant forces fail to counteract the capillary pressure imposed by the narrow pore throat of the caprock. In many sedimentary contexts, CO_2 typically acts as the non-wetting phase, while brine serves as the wetting phase. This interaction prompts brine to infiltrate the diminishing boundary of the CO_2 plume after injection [14]. Such capillary movements lead to the formation of discrete, immobile CO_2 bubbles and ganglia, representing the residual trapping process. The intricacies of this mechanism are shaped by the rock's inherent pore structure and wettability. This study utilized capillary pressure patterns through the Leverett J-function [146], as illustrated in Figure 5.1B and Figure 5.1C.

5.2. MODEL ARCHITECTURE AND TRAINING FORMULATION

This section introduces the architecture and training formulation of the proposed CO_2PCNet model for simulating CO_2 storage dynamics. To capture the spatial-temporal evolution of state variables, the model integrates convolutional encoders, a ConvLSTM module for temporal learning, and a decoder for reconstructing output sequences. An auto-regressive strategy is used to enable multi-step forecasting beyond the training time steps. The model architecture is further enhanced through conditioning on heterogeneous permeability fields and encoded physical state variables, ensuring spatial

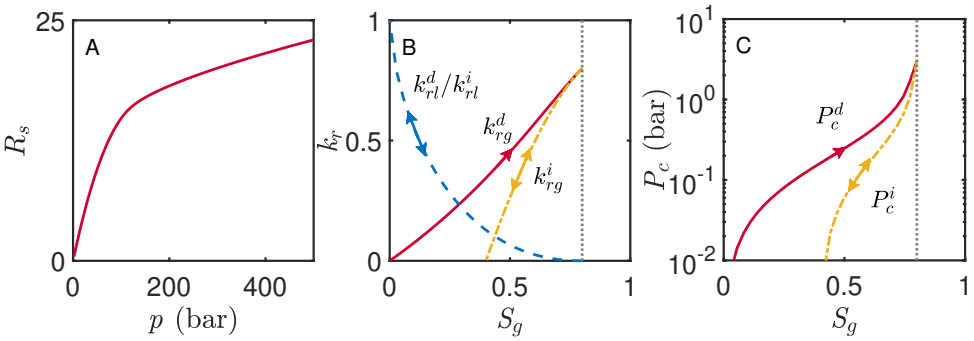


Figure 5.1: Setup of physical models. (A) Predicted solubilities of CO_2 in brine at 60°C. (B) Relative permeability curves: primary drainage and imbibition curves for both liquid and gas phases. Superscripts d and i represent drainage and imbibition, respectively. A single-headed arrow denotes an irreversible process along a given curve, while a double-headed arrow indicates reversibility. (C) Capillary pressure curves: primary drainage and imbibition curves.

sensitivity to geologic variability. To enhance physical reliability, the training loss incorporates residual terms from the governing equations, ensuring that predicted states are consistent with fundamental conservation principles.

5.2.1. ConvLSTM

The Convolutional Long Short-Term Memory (ConvLSTM) introduces a novel paradigm in neural networks, specifically designed for modeling evolving long-term dependencies [147]. By integrating the spatial characteristics of CNNs and the temporal capabilities of Long Short-Term Memory (LSTM) networks [148], ConvLSTM provides a robust solution for multi-dimensional sequential data.

While LSTMs capture time-based patterns, they are less adept with spatial data. The ConvLSTM addresses this by replacing traditional LSTM matrix operations with convolutional operations. This preserves spatial information, such as pixel arrangements, while handling temporal sequences. Fundamentally, ConvLSTM employs memory cells and a gated system to mitigate the vanishing gradient issue seen in standard Recurrent Neural Networks (RNNs). Instead of fully-connected layers used in LSTMs, ConvLSTM incorporates convolutional layers, enhancing its spatial data handling.

The ConvLSTM processes information through a combination of gates and cells. The illustration is presented in Figure 5.2. Let the input tensor at time t be represented as $\mathbf{X}^t$. The corresponding hidden state and cell state at this time are $\mathbf{h}^t$ and $\mathbf{C}^t$, respectively. The forget gate $\mathbf{f}^t$, governed by a sigmoid activation function, determines which segments of the previous memory should be retained or discarded. The input gate $\mathbf{i}^t$ establishes the amount of new data that will be stored into the cell state, while

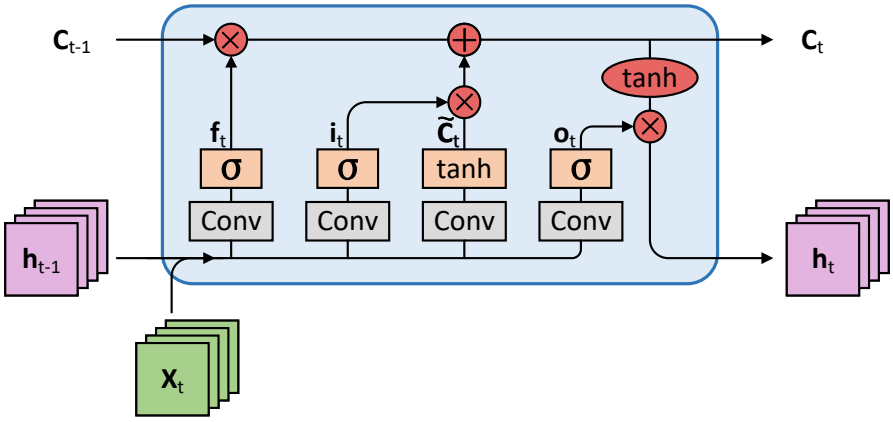


Figure 5.2: Schematic Representation of a ConvLSTM Cell.

the internal cell $\widetilde{\mathbf{C}}^t$, influenced by a hyperbolic tangent activation layer, represents the potential new cell state. This works jointly with the input gate to finalize the information flowing into the actual cell state. Lastly, the output gate $\mathbf{o}^t$ derives the subsequent hidden state based on the current cell state and input. Formally, the ConvLSTM operations can be captured by the following equations

$$
\begin{aligned}
\mathbf{f}^t &= \sigma(\mathbf{W}_f * [\mathbf{h}^{t-1}, \mathbf{X}^t] + \mathbf{b}_f), \\
\mathbf{i}^t &= \sigma(\mathbf{W}_i * [\mathbf{h}^{t-1}, \mathbf{X}^t] + \mathbf{b}_i), \\
\widetilde{\mathbf{C}}^t &= \tanh(\mathbf{W}_c * [\mathbf{h}^{t-1}, \mathbf{X}^t] + \mathbf{b}_c), \\
\mathbf{C}^t &= \mathbf{f}^t \odot \mathbf{C}^{t-1} + \mathbf{i}^t \odot \widetilde{\mathbf{C}}^t, \\
\mathbf{o}^t &= \sigma(\mathbf{W}_o * [\mathbf{h}^{t-1}, \mathbf{X}^t] + \mathbf{b}_o), \\
\mathbf{h}^t &= \mathbf{o}^t \odot \tanh(\mathbf{C}^t).
\end{aligned}
\tag{5.3}
$$

5.2.2. AUTO-REGRESSIVE MODEL FOR CCS

Understanding and forecasting the evolution of critical parameters, the overall mass fraction z_{CO_2} and the liquid phase pressure P_l, is important for optimizing storage strategies and ensuring the safety and sustainability of the reservoir in the context of CO_2 storage. Due to the inherent complexity and non-linear dynamics present in these physical processes, an auto-regressive (AR) model is employed in this work [64]. This model, utilizing past system states to predict future states, offers a robust methodology for providing consistent insights into their evolving states.

The model incorporates the temporal dependencies intrinsic to the CO_2 storage process, casting them into a predictive framework. Enhanced with neural network ca-

pabilities, the AR model predicts the subsequent state at time $t+1$ using the state variables at time t. Given the initial state $\mathbf{Y}^0 = [z^0_{CO_2}, P^0_l]$ and the static reservoir variables $\mathbf{K} = [\mathbf{k}_1, ..., \mathbf{k}_{n_k}]$, a sequence of dynamic variables rollout is computed as: $[\hat{\mathbf{Y}}^1, ..., \hat{\mathbf{Y}}^{n_t}]$. Mathematically, the evolving states of the system can be represented as

$$\hat{\mathbf{Y}}^0 = \mathbf{Y}^0, \tag{5.4}$$

$$\hat{\mathbf{Y}}^{t+1} = \mathscr{F}(\hat{\mathbf{Y}}^t, \mathbf{K}, \theta), \qquad t \in 1, ..., n_t. \tag{5.5}$$

Here, $\hat{\mathbf{Y}}^{t+1}$ represents the predicted state of the system at the next time step $t+1$, $\mathbf{K}$ is the static variables (i.e. the permeability fields), and $\mathscr{F}$ denotes the neural network function with θ as its parameters. The formulation allows the neural network to determine the mapping function $\mathscr{F}$ between the static variables and dynamic state variables, facilitating the computation of the dynamic variables at new time steps.

In essence, this neural network operates as a time integrator, where the output of each sequence is fed into the next layer as input, thereby functioning fundamentally as an AR model.

5

5.2.3. NETWORK ARCHITECTURE: CO_2PCNET

In order to develop robust and accurate modeling approaches for CO_2 storage dynamics, the CO_2PCNet architecture is formulated to learn the complex temporal and spatial dependencies intrinsic to CO_2 storage data sequences. This network consists of three principal components: (1) the encoder, (2) the ConvLSTM, and (3) the decoder, as illustrated in Figure 5.3.

Initially, the encoder, which includes three convolutional layers, adeptly compresses the provided input into a low-dimensional latent space, thereby encapsulating the spatial features crucial for accurate predictive modeling [49]. The Swish activation function is employed for the convolutional layers to introduce non-linearity, fostering more complex representations [149].

Following the encoding phase, the resultant spatial embeddings are channeled into the ConvLSTM layer. This layer, characterized by its recurrent neural network structure, is designed to simultaneously decode spatial patterns and their temporal evolution.

Subsequently, the ConvLSTM layer's outputs are transformed through an upsampling operation, effectively reconstructing the low-resolution latent embeddings into high-resolution predictive outputs [133]. Specifically, this decoder phase is a combination of upsampling and an additional convolution layer, consisting of three stages to refine and project the latent space into detailed output predictions. Consequently, the dynamic variables $z^{t+1}_{CO_2}$, P^{t+1}_l at time instant $t+1$ are concatenated with the static variables, setting the stage for the inputs at the next time step, $t+2$. The detailed structures of the encoder and decoder blocks are depicted in Figure 5.4.

Incorporating an auto-regressive strategy within CO_2PCNet, the network is designed to ensure that each sequential prediction $(t+1, t+2, \ldots)$ is based on the information

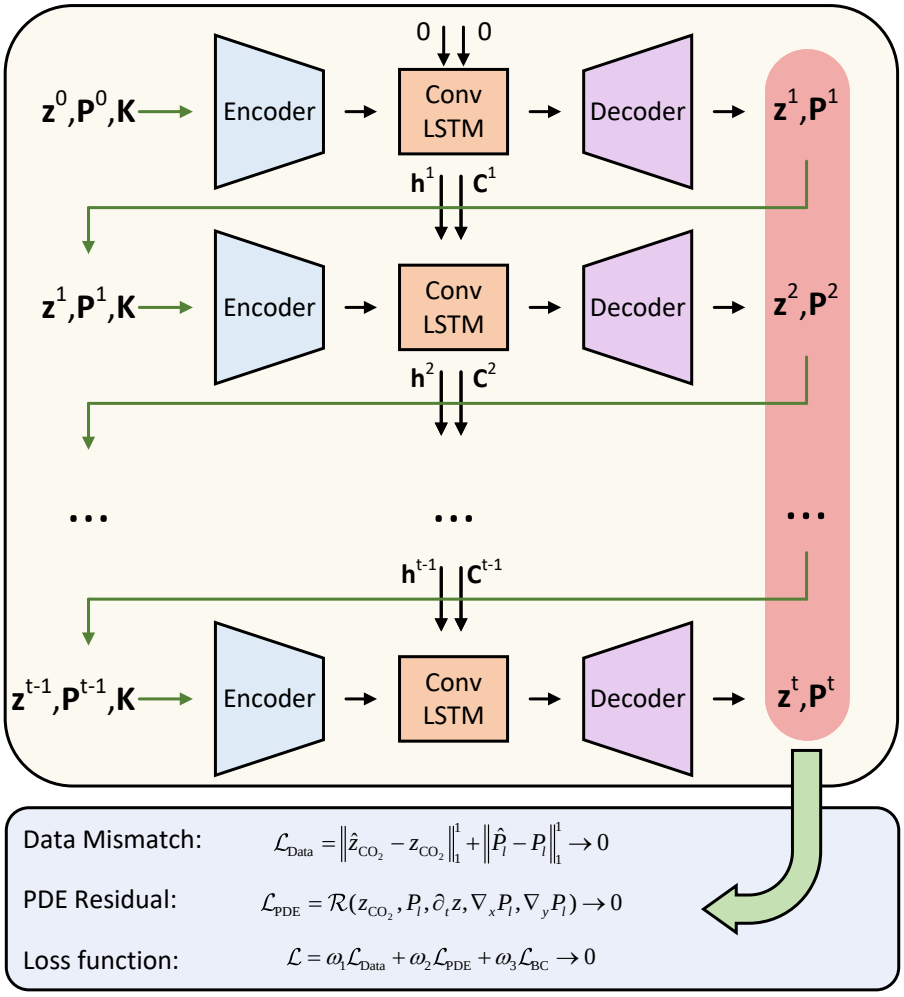


Figure 5.3: The network architecture of CO_2PCNet.

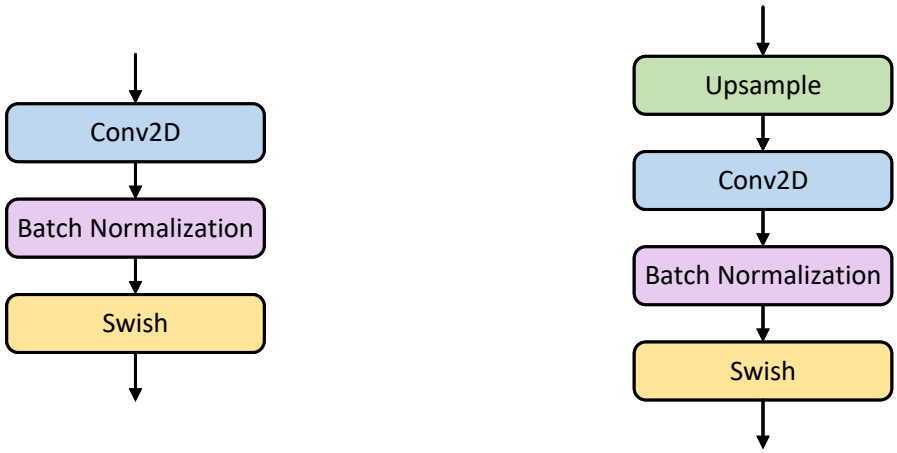


Figure 5.4: Schematic illustration of the Convolutional block and the upsampling convolutional block. Left: encoder structure; right: decoder structure.

of the preceding state. This approach effectively captures the evolving dynamics of the system over time, allowing the network to adaptively refine its predictions based on the continuously updated data. Consequently, this results in forecasts that are progressively informed and accurate, as the model processes more data.

CO_2PCNet employs a consistent architectural framework throughout all phases of CO_2 storage, from injection to post-injection. This approach ensures seamless transitions between simulation phases by using outputs from prior time steps as inputs for subsequent predictions, thus maintaining a coherent and continuous flow of data throughout the model's operation.

The model's capacity for iterative refinement, informed by its auto-regressive framework, offers a robust alternative to traditional simulators for understanding CO_2 storage processes, particularly valuable for tasks such as uncertainty quantification and history matching. Moreover, the integration of an encoder-decoder ConvLSTM within this auto-regressive framework significantly enhances the functionality. This integration enables the efficient extraction and interpretation of spatial features while simultaneously monitoring their temporal development.

5.2.4. LOSS FUNCTION

Another critical aspect of CO_2PCNet's effectiveness lies in its ability to ensure that predicted CO_2 distributions conform closely to the fundamental physical laws governing subsurface flow processes. Our model incorporates these laws directly within

its architecture and loss function to ensure that predictions not only match observational data but also align with physical principles. Therefore, we formulate a loss function that guarantees the outputs of the neural network remain faithful to the physical constraints inherent to the domain of interest.

The training strategy of CO_2PCNet leverages an auto-regressive approach, which processes sequential data by using the prediction from a current time step as the input for the next, facilitating a dynamic modeling of temporal dependencies. This design is important for calculating the discretized PDE residuals of equation (2.5), which form an integral part of the physics-constrained term in our loss function. By embedding these calculations within the network, CO_2PCNet effectively captures the dynamic behavior of CO_2 plume evolution and pressure distribution across the geological formation.

The total loss function, denoted as $\mathscr{L}$, is composed of three primary elements: the data mismatch term $\mathscr{L}_{\text{Data}}$, the physics-constrained term $\mathscr{L}_{\text{PDE}}$, and the boundary condition term $\mathscr{L}_{\text{BC}}$. It is worth noting that, regarding the initial conditions, the state variables are known and thus, there is no specific term for initial conditions in our loss function.

The data mismatch term $\mathscr{L}_{\text{Data}}$ measures the difference between the network predictions and the actual observed data. For our model, we utilize the mean absolute error (MAE) of mass fraction and pressure fields, chosen due to its robustness. This metric is widely used as a loss function

$$\mathscr{L}_{\text{Data}} = \frac{1}{n_t n_s n_g} \sum_{t=1}^{n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_g} (\|(\hat{z}_{\text{CO}_2})_{i,j}^t - (z_{\text{CO}_2})_{i,j}^t\|_1 + \|(\hat{P}_l)_{i,j}^t - (P_l)_{i,j}^t\|_1), \tag{5.6}$$

where n_s is the number of samples, and $n_g = 32 \times 128 = 4096$ represents the number of grid blocks. The terms $(\hat{z}_{\text{CO}_2})_{i,j}^t$ and $(\hat{P}_l)_{i,j}^t$ correspond to the predicted mass fraction and pressure, respectively, for training sample i, within grid block j, at time step t. $(z_{\text{CO}_2})_{i,j}^t$ and $(P_l)_{i,j}^t$ are the actual, or ground truth values of mass fraction and pressure. This loss term ensures the output of the network is closely aligned with the observed data, maintaining the accuracy of the model in capturing CO_2 behavior. In our auto-regressive model, errors at each timestep are calculated and influence the total loss, leading to updates across the entire model. This setup ensures continuous improvement through backpropagation, where early errors adjust model parameters, enhancing predictions across the entire timeline. Thus, the model learns from discrepancies at any point in the sequence, refining its performance iteratively to boost overall accuracy throughout the learning process.

The physics-constrained term $\mathscr{L}_{\text{PDE}}$ plays an important role in ensuring that predictions of CO_2PCNet follow the fundamental physical laws governing the subsurface fluid dynamics. This term is defined as the residuals of the PDEs for the two-component system involving CO_2 and brine, ensuring that mass conservation is represented

$$\mathscr{L}_{\text{PDE}} = \|\mathscr{R}(z_{\text{CO}_2}, P_l, \partial_t z, \nabla_x P_l, \nabla_y P_l)\|_1, \tag{5.7}$$

where $R(\cdot)$ represents the residuals of the mass conservation equations as specified in equation (2.5), aiming for these residuals to approach zero to indicate physical consistency. It encapsulates the mass conservation laws that are critical for modeling flow in porous media.

Derivatives within the PDE constraints are enforced across the entire spatiotemporal domain of our model to ensure that the temporal dynamics and spatial distributions adhere to the physical behavior. This is achieved by discretizing the domain and applying differential equation constraints at each grid point. Specifically, spatial derivatives are computed using finite difference schemes, which are essential for local conservation laws. The discrete temporal derivatives are calculated between time steps, capturing the dynamics of CO_2 behavior over time. For instance, consider the derivative of liquid pressure (P_l) with respect to spatial coordinates, which can be approximated by:

$$(\frac{\partial P}{\partial x})^t_{i+\frac{1}{2},j} = \frac{2(P^t_{i+1,j} - P^t_{i,j})}{\Delta x_i + \Delta x_{i+1}}. \tag{5.8}$$

The boundary condition term, $\mathscr{L}_{\text{BC}}$, is formulated to both constrain and penalize deviations from the pre-defined boundary conditions, defined as:

$$\mathscr{L}_{\text{BC}} = \|\hat{\boldsymbol{u}} - \boldsymbol{u}_{\text{BC}}\|_1. \tag{5.9}$$

The total loss function is a weighted sum of these three terms, i.e.,

$$\mathscr{L} = \omega_1 \mathscr{L}_{\text{Data}} + \omega_2 \mathscr{L}_{\text{PDE}} + \omega_3 \mathscr{L}_{\text{BC}}. \tag{5.10}$$

where ω_1, ω_2 and ω_3 are hyperparameters that balance the influence of each term. The choice of these values is critical and is typically determined through experimentation to find the best model performance. In this work, equal weights are chosen for all terms, thus maintaining a balanced contribution from each aspect of the loss function to the overall learning process. Figure 5.5 illustrates the step-by-step process of our physics enforcement algorithm.

5.3. NUMERICAL EXPERIMENTS AND RESULTS

This section conducts a numerical evaluation of the CO_2PCNet model for simulating multiphase, multicomponent CO_2-brine flow in heterogeneous saline aquifers. The objective is to assess the model's predictive performance across a range of representative scenarios, including both in-sample and out-of-sample permeability fields. Particular emphasis is placed on the model's ability to accurately forecast system states beyond the temporal span of the training data, a critical requirement for post-injection analysis and long-term migration prediction.

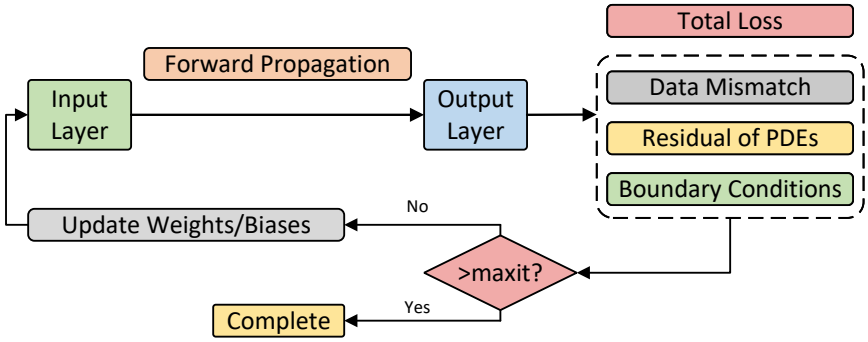


Figure 5.5: Flowchart illustrating the physics enforcement mechanism in CO_2PCNet. The process ensures that each prediction adheres to established physical laws, thereby enhancing the reliability and accuracy of the model outputs.

5

5.3.1. DATASET GENERATION

To facilitate the training and evaluation of the CO_2PCNet in deep saline aquifers, a 2-D CCS simulation system is implemented. From the geological perspective, considering the important role of gravity in the CO_2 trapping process, a 2-D representation in the xz-plane is adopted, because gravity plays a significant role in the migration and trapping of CO_2 in saline aquifers. Aquifers are generally characterized by a longitudinal length compared to their cross-sectional width. Therefore, the dimensions of our simulation domain are set to 128 m × 32 m in the xz-direction and the grid is uniformly discretized into 1 m × 1 m cells. This setup effectively simulates the migration patterns of CO_2 within a 2-D vertical cross-section of an aquifer, providing a conceptual representation of its behavior in such geological contexts. It is worth noting that the architecture of CO_2PCNet is scalable and can be adapted to larger, more realistic reservoir sizes as computational resources allow.

In order to mimic the geological formations for CO_2 storage, a total 1200 permeability realizations were generated using the Stanford Geostatistical Modeling Software (SGeMS), an open-source tool for geostatistical modeling [142]. These permeability fields, which are log-normally distributed with a mean $\ln K$ of 2.5 and a standard deviation of 2, are shown in Figure 5.6, which illustrates three sample realizations. The heterogeneity in permeability is critical for quantifying the impact of uncertain reservoir properties. By incorporating a wide range of permeability scenarios, our simulation framework allows for a detailed analysis of CO_2 migration and trapping behaviors under varying geological permeability conditions.

For the simulation settings, the initial pressure is set at 2.5×10^7 Pa, with a constant temperature of 300 K. No flow boundary conditions are applied to all sides of the aquifer. CO_2 is injected at a consistent rate of 1×10^{-4} pore volumes per day along

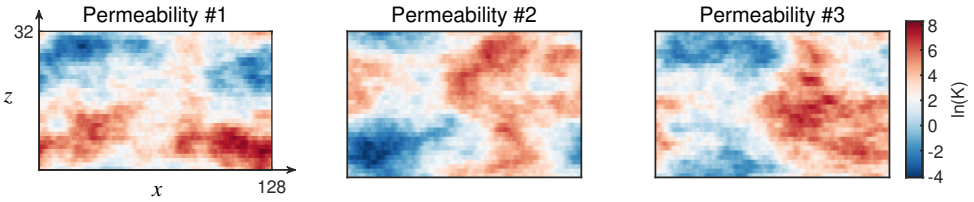


Figure 5.6: Three cases of heterogeneous permeability realizations.

the entire left boundary, with a production well placed along the entire right boundary [18, 150]. The planned simulation duration is 2520 days, including a 600 days injection phase followed by a 1920 days post-injection phase. The total period is evenly divided into 21 intervals, with each lasting 120 days. The forward simulations are performed using DARSim, and Table 5.1 presents the physical parameters and configurations utilized in the DARSim numerical simulation. Our model, which addresses the complex dynamics of multi-component, multi-phase interactions involved in CO_2 storage in deep saline aquifers, employs a mass-conservative numerical formulation for extrapolating to future time periods.

Table 5.1: Physical Parameters and Simulation Setup for the DARSim Numerical Simulation.

Parameter	Value	Unit
Aquifer length	128	m
Aquifer height	32	m
Porosity	0.2	-
CO_2 injection rate	1×10^{-4}	pore volumes per day
Initial pressure	2.5×10^{7}	Pa
Bottom hole pressure	2.5×10^{7}	Pa
Temperature (isothermal)	300	K
CO_2 density at STC	1.98	kg/m^3
Brine density at STC	1060	kg/m^3
Brine salinity	1×10^{5}	parts per million

5.3.2. BASELINE MODEL SETUP

The baseline model is trained using neural architectures similar to those employed in our proposed CO_2PCNet to ensure a fair comparison. We mainly compare the solution snapshots between CO_2PCNet, our physics-constrained network, and CO_2Net, which operates without the incorporation of physics constraints. This comparison aims to discern the impact and efficacy of integrating physical laws directly into the neural network's learning process.

For the setup of neural networks, the encoder of the network features three convolutional layers with channel sizes of 16, 32, and 128 units, using 3 × 3 kernels. The

strides for these layers are set to 2, 1, and 2, respectively, facilitating spatial downsampling and feature extraction. Following the encoder, the network employs a ConvLSTM layer with 128 hidden units, designed to process temporal information using 3 × 3 kernels and a stride of 1. The decoder, tasked with reconstructing the output, mirrors this approach using similar kernels and strides for upsampling.

CO_2PCNet and CO_2Net are trained with data from the first 15 time steps, encompassing 1800 days. The models are trained on 1000 permeability realizations over the first 15 time steps (1800 days), and evaluated on 200 unseen permeability fields for the remaining 6 steps (720 days). During this evaluation phase, a uniform set of 200 permeability fields is employed, ensuring a consistent framework for forecasting. Notably, to standardize the data, min-max normalization is applied to the output values. The network model employs the adaptive moment estimation (Adam) optimizer for training, starting with an initial learning rate of 0.001 [144]. The network is trained for 1000 epochs, and the minibatch size is 20. In our experiments, the training time for CO_2PCNet was around 8 hours, while CO_2Net required approximately 6.5 hours. This additional training duration for CO_2PCNet is primarily due to the integration of physical constraints into the learning process.

In the context of CO_2 storage in deep saline aquifers, two essential parameters - R_s and S_g, representing CO_2 solubility in liquid and gas saturation, respectively, are calculated from the predicted mass fraction z_{CO_2} and pressure P_l. These predictions are also compared to the ground truth to assess the model's accuracy in simulating CO_2 distribution and behavior in these complex geological formations.

5.3.3. EVALUATION METRICS

Quantifying the relative errors in CO_2PCNet predictions for z_{CO_2} and P_l is essential for evaluating the model's accuracy. The relative error for the mass fraction at a given time step t, denoted δ_z^t, across $n_s = 200$ test samples, is calculated as

$$\delta_z^t = \frac{1}{n_s n_g} \sum_{i=1}^{n_s} \sum_{j=1}^{n_g} \frac{|(\hat{z}_{CO_2})_{i,j}^t - (z_{CO_2})_{i,j}^t|}{(z_{CO_2})_{i,\max}^t - (z_{CO_2})_{i,\min}^t}, \tag{5.11}$$

where $(\hat{z}_{CO_2})_{i,j}^t$ and $(z_{CO_2})_{i,j}^t$ represent the predicted and ground truth mass fractions for test sample i at grid block j and time step t, respectively. The normalization factor, $(z_{CO_2})_{i,\max}^t - (z_{CO_2})_{i,\min}^t$, is the difference between the maximum and minimum grid-block mass fraction values for sample i at time step t. The relative error in pressure predictions is computed similarly:

$$\delta_P^t = \frac{1}{n_s n_g} \sum_{i=1}^{n_s} \sum_{j=1}^{n_g} \frac{|(\hat{P}_l)_{i,j}^t - (P_l)_{i,j}^t|}{(P_l)_{i,\max}^t - (P_l)_{i,\min}^t}. \tag{5.12}$$

To obtain a comprehensive view of the model's performance over time, the overall relative mass fraction and pressure errors across n_t time steps, denoted δ_z and δ_z,

are calculated as

$$\delta_z = \frac{1}{n_t} \sum_{t=1}^{n_t} \delta_z^t, \tag{5.13}$$

$$\delta_P = \frac{1}{n_t} \sum_{t=1}^{n_t} \delta_P^t. \tag{5.14}$$

These metrics are employed to evaluate the performance of the proposed network model under various scenarios, offering insights into both accuracy and generalization.

5.3.4. FORECASTING FUTURE RESPONSES

In this case, we first evaluate the performance of CO_2PCNet and CO_2Net in forecasting CO_2 behavior beyond the training period. Figure 5.7 and 5.8 present a comparison between the CO_2PCNet's predictions of the mass fraction (z_{CO_2}) and the liquid pressure (P_l) for Permeability #1 (shown in Figure 5.6). The results are compared with the high-fidelity simulation results and the baseline algorithm CO_2Net at four snapshots (t = 600, 1320 days in the training period, t = 1920, 2520 days beyond the training period). Below each prediction, the mean relative absolute errors δ_z and δ_P are reported below each prediction. These snapshots capture the transition from training phase to the predictive phase, where the model's generalization capability is evaluated against the system's evolving complexity.

Upon CO_2 injection into the saline aquifer, the CO_2 starts to displace the existing brine within the porous media. The CO_2 tends to rise due to its lower density relative to the brine. This migration is governed by a complex interplay of gravity, capillary pressure, and the permeability of the rock. Simultaneously, as CO_2 dissolves in the brine, it increases the density of brine, causing it to descend and thereby promoting further dispersion of CO_2. The pressure distribution observed during the injection phase is typically higher near the injection site, gradually decreasing outward as CO_2 displaces the brine and the system approaches pressure equilibrium. Following injection, the pressure profile demonstrates a gradual decrease to reach a new equilibrium.

Both models exhibit close alignment with the high-fidelity simulations during the training phase, demonstrating their capability to accurately learn and replicate the dynamics of CO_2 distribution. However, as the model progresses into the extrapolation phase, a distinct divergence becomes apparent. CO_2PCNet's predictions preserve a more consistent accuracy, reflected in the tighter error distributions as time advances. In contrast, the errors from CO_2Net, particularly for P_l predictions, appear to grow, which suggests that the integration of physics constraints in CO_2PCNet plays a significant role in ensuring the consistency and accuracy over longer prediction intervals. Although relative error margins naturally widen as the model projects further into the future, CO_2PCNet's outputs remain reasonably accurate, staying within acceptable error margins. Additionally, it is worth noting that the relative errors in

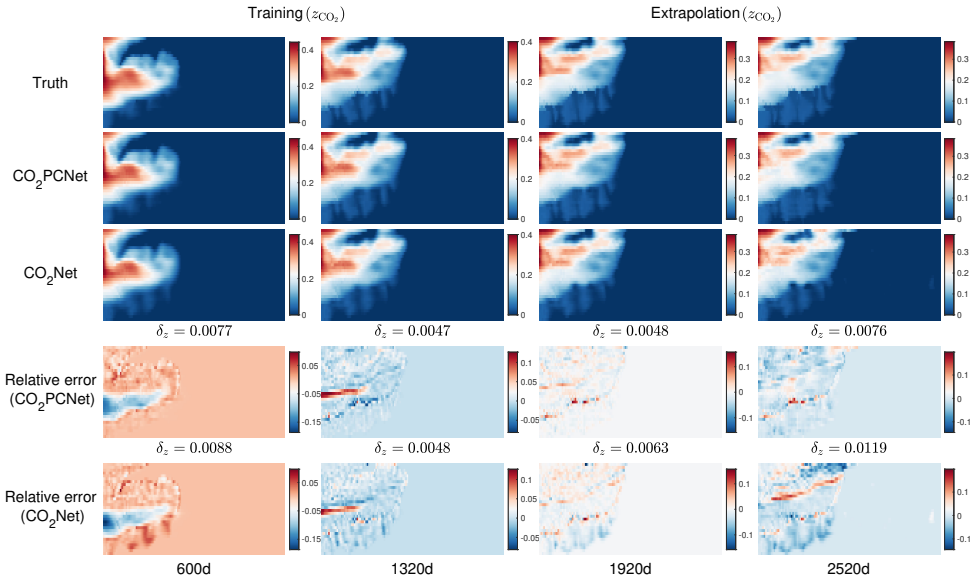


Figure 5.7: Temporal evolution of z_{CO_2}. Numerical simulations (first row), CO_2PCNet predictions (second row), and CO_2Net predictions (third row) are compared at four time steps. Relative errors for CO_2PCNet (fourth row) and CO_2Net (fifth row) are depicted. The left two columns show results within the training period, while the right two columns represent extrapolations.

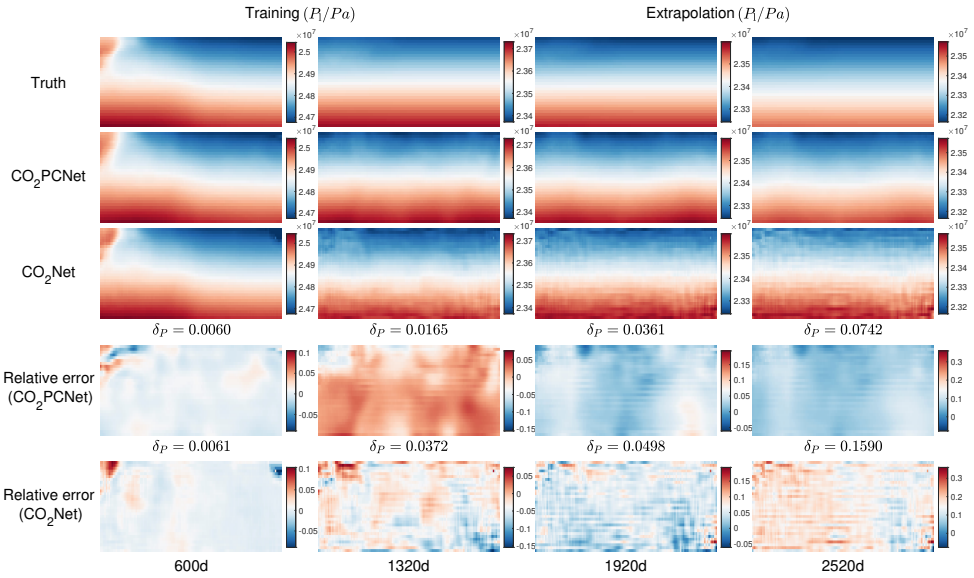


Figure 5.8: Temporal evolution of liquid pressure P_l in pascals (Pa). Numerical simulations (first row), CO_2PCNet predictions (second row), and CO_2Net predictions (third row) are compared at four time steps. Relative errors for CO_2PCNet (fourth row) and CO_2Net (fifth row) are depicted. The left two columns show results within the training period, while the right two columns represent extrapolations.

pressure predictions are distributed across the entire domain due to the elliptic nature of the pressure variable.

To assess the model performance at the final time step, Figure 5.9 provides an insightful comparison of the predictions from CO_2PCNet and CO_2Net against the reference simulations. CO_2PCNet's predictions for both z_{CO_2} and P_l closely align with the reference values, forming a dense cluster around the line of perfect agreement. This indicates high predictive accuracy and a strong correlation between the predicted and actual values. CO_2Net, while still maintaining a reasonable correlation for z_{CO_2}, predictions for P_l show a more significant deviation, indicating a less precise match to the reference data.

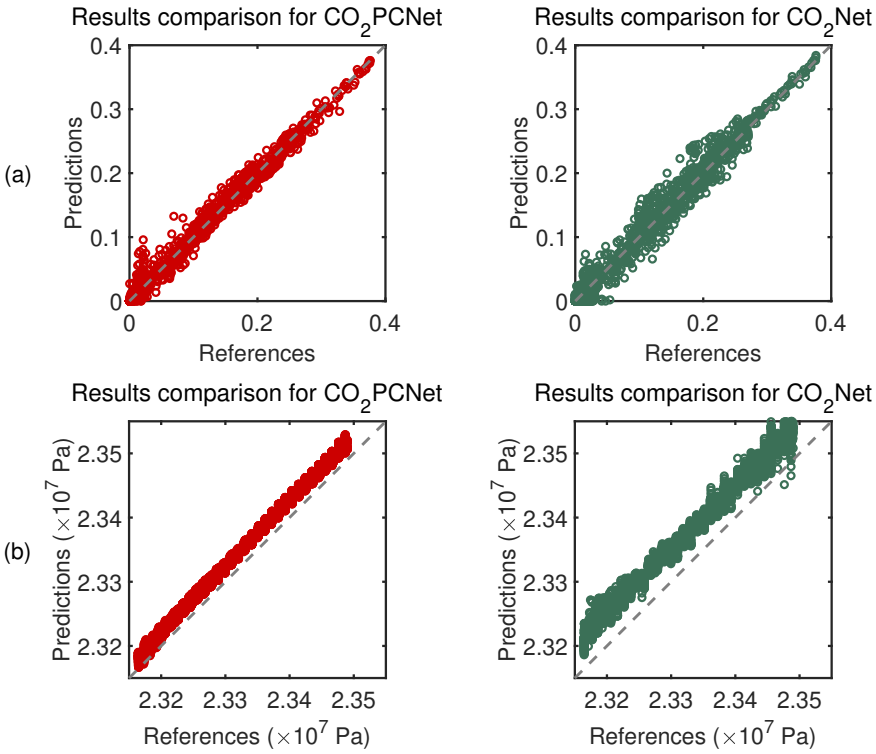


Figure 5.9: Scatter plots of Predictions vs. Reference Values at 2520 days: (a) z_{CO_2} and (b) P_l. Predictions by CO_2PCNet are on the left, while CO_2Net's are on the right, each plotted against the high-fidelity simulation reference. Each point represents a predicted value plotted against its corresponding reference value, with the line of perfect agreement shown as a dashed line for reference.

5.3.5. FORECASTING FUTURE RESPONSES WITH NEW PERMEABILITY FIELDS

In this subsection, the CO_2PCNet model is further tested by predicting CO_2 behavior during both active injection and post-injection phases in scenarios with new permeability fields not encountered during training. This generalization capability is critically important for the applications of neural network models as surrogate models in the context of uncertainty quantification, where the model is required to make accurate predictions for various permeability maps.

Figures 5.10 and 5.11 compare the predictions from CO_2PCNet with those from CO_2Net and high-fidelity numerical simulation results for key variables at the prediction time of 2520 days. The comparisons are made for two distinct permeability scenarios, labeled as Permeability #2 and Permeability #3, which are shown in Figure 5.6. Notably, R_s and S_g are derived from z_{CO_2} and P_l. The mean relative errors for z_{CO_2} are 0.74% and 0.67% for the first and second test cases, respectively. For P_l, CO_2PCNet presents mean errors of 4.01% and 2.92%, suggesting high fidelity in its predictions. In contrast, CO_2Net produces larger errors, especially for P_l, where discrepancies increase significantly (δ_P = 13.91% and 9.57%). Despite encountering some isolated high values due to the response discontinuity, mainly in R_s, which is an intrinsic characteristic of complex systems, the CO_2PCNet model still maintains robust performance.

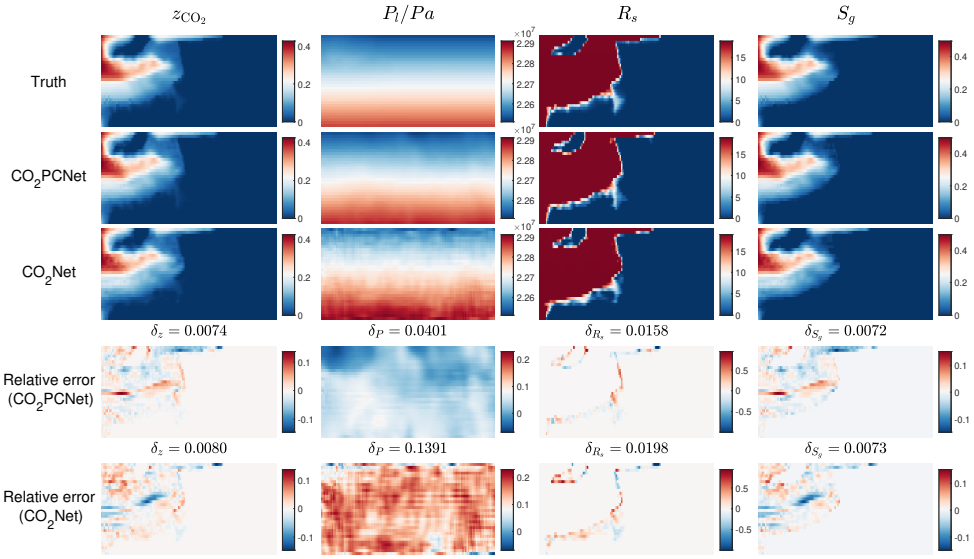


Figure 5.10: Test case 1: Predictions of z_{CO_2}, liquid pressure P_l in pascals (Pa), R_s and S_g at 2520 days. The top row represents the numerical simulation 'Truth', followed by CO_2PCNet and CO_2Net predictions. The bottom two rows quantify the relative errors for CO_2PCNet and CO_2Net, respectively.

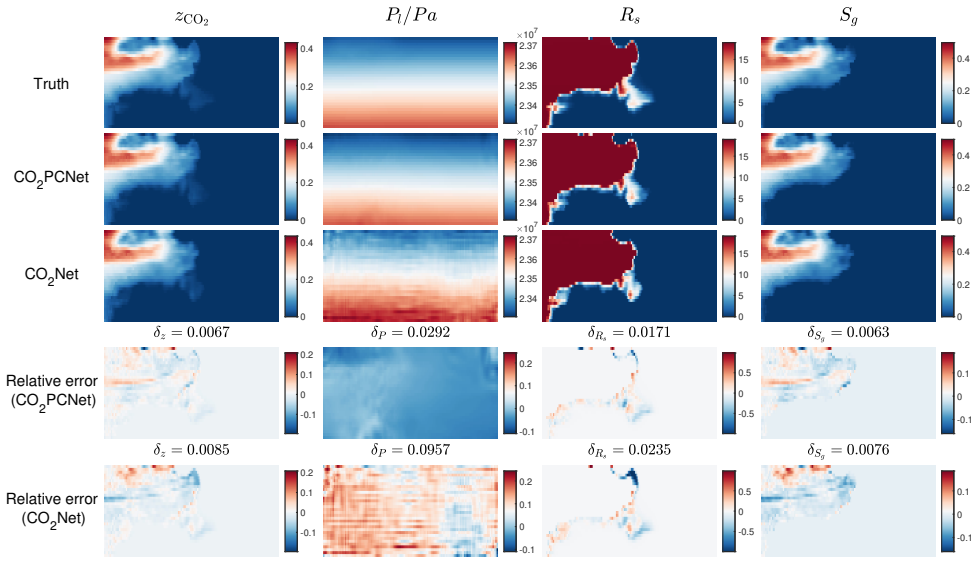


Figure 5.11: Test case 2: Predictions of z_{CO_2}, liquid pressure P_l in pascals (Pa), R_s and S_g at 2520 days. The top row represents the numerical simulation 'Truth', followed by CO_2PCNet and CO_2Net predictions. The bottom two rows quantify the relative errors for CO_2PCNet and CO_2Net, respectively.

Figure 5.12 illustrates the mean relative error trends (equations (5.13) and (5.14)) of CO_2PCNet and CO_2Net over a period of 2520 days. The figure presents evaluation metrics for two individual test cases and results across 200 test cases, providing a perspective on the accuracy of each model in predicting key variables of CO_2 sequestration. In all test cases, CO_2Net can provide satisfactory short-term forecasts. However, CO_2PCNet consistently achieves lower mean relative errors for both CO_2 mass fraction (δ_z) and liquid pressure (δ_P) compared to CO_2Net, reflecting its excellent predictive accuracy.

A statistical comparison of residuals of PDEs, reflecting the mass conservation for CO_2 and brine over time as predicted by CO_2PCNet and CO_2Net across 20 random test cases is presented in Figure 5.13. CO_2PCNet's performance is distinguished by a narrow range of residuals, with medians around 10^{-5} for both CO_2 and brine, demonstrating its consistent precision and adherence to conservation principles. In contrast, CO_2Net produces a wider residual distribution, with median values typically in the 10^{-4} range, suggesting a less precise agreement with mass conservation laws.

5.3.6. PERFORMANCE COMPARISONS

In this section, we evaluate CO_2PCNet's performance against the Auto-regressive Encoder-Decoder (AR-ED), a commonly utilized neural network framework in related studies

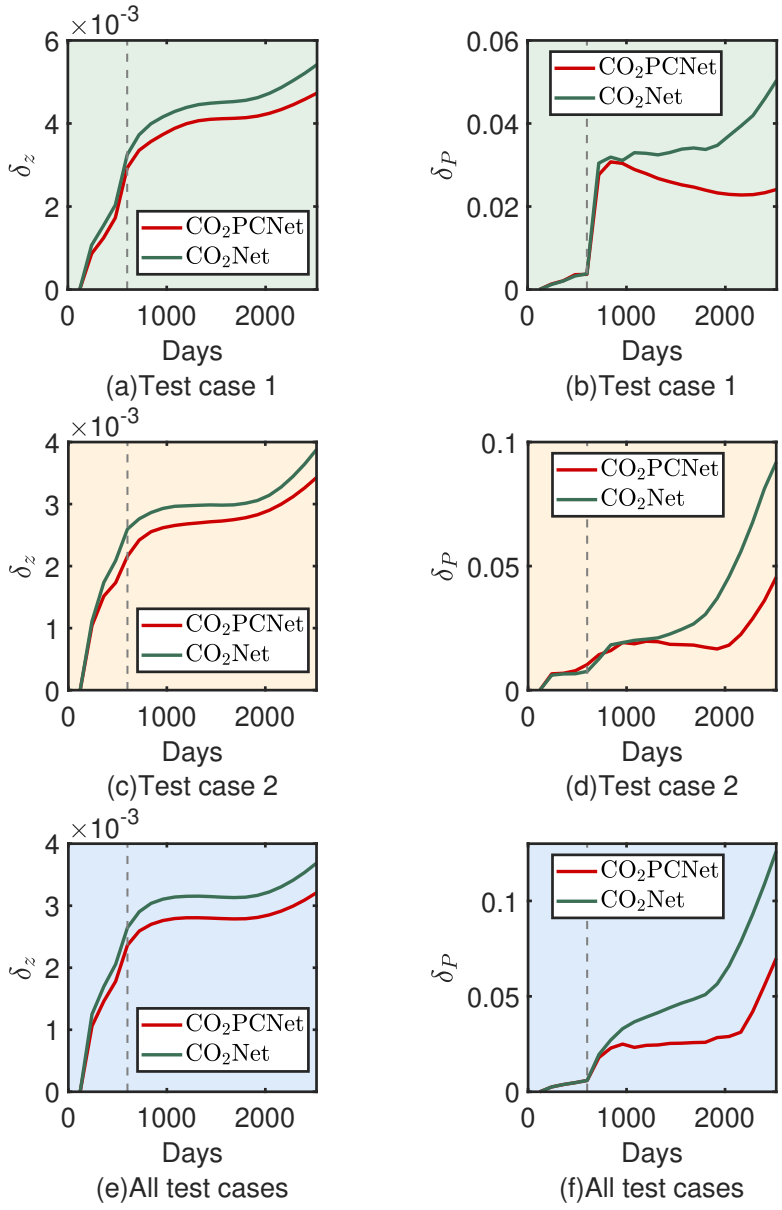


Figure 5.12: Mean relative error trends for CO_2PCNet and CO_2Net over a period of 2520 days across two test cases and all 200 test cases. The vertical dashed lines represent the stop of CO_2 injection.

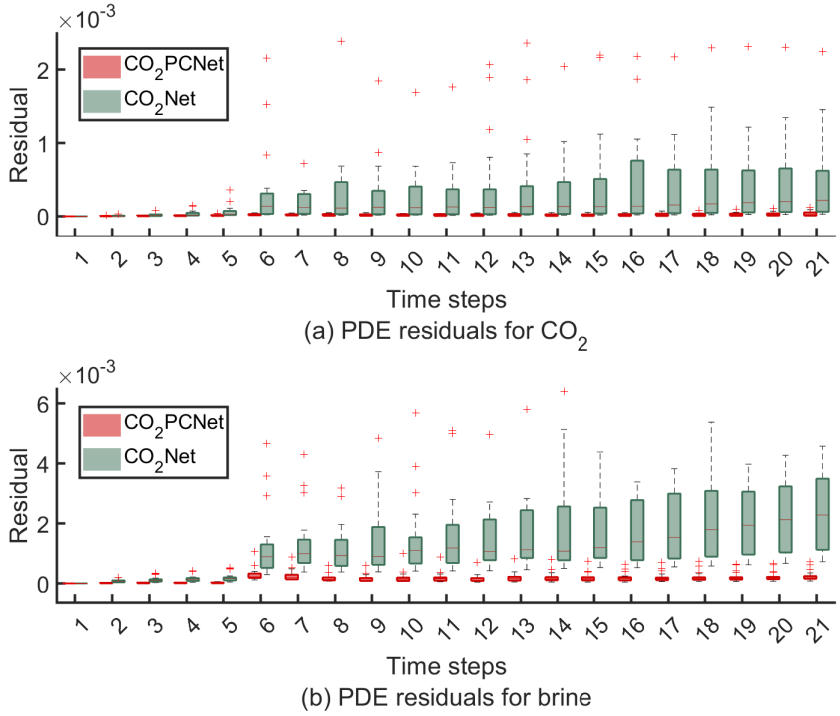


Figure 5.13: Residual analysis for two components CO_2 and brine (equation (2.5)) for 20 random test cases.

[151]. Although AR-ED shares a foundational architecture and physics constraints with CO_2PCNet, it does not incorporate the ConvLSTM component, a crucial element of CO_2PCNet that enriches its ability to process and predict temporal sequences within spatial contexts.

In Figure 5.14 and 5.15, we evaluate the predictive performance of both models against reference values for a random test case at 2520 days, as well as their accumulated error evolution over a period of 2520 days. CO_2PCNet's predictions (in red) demonstrate a high degree of fidelity to the reference data, aligning closely with the ideal prediction line. This is quantitatively supported by a coefficient of determination (R^2) of 0.9944 for z_{CO_2} predictions. In contrast, AR-ED achieves an R^2 of 0.9793, signifying a slightly lower correlation with the simulation data.

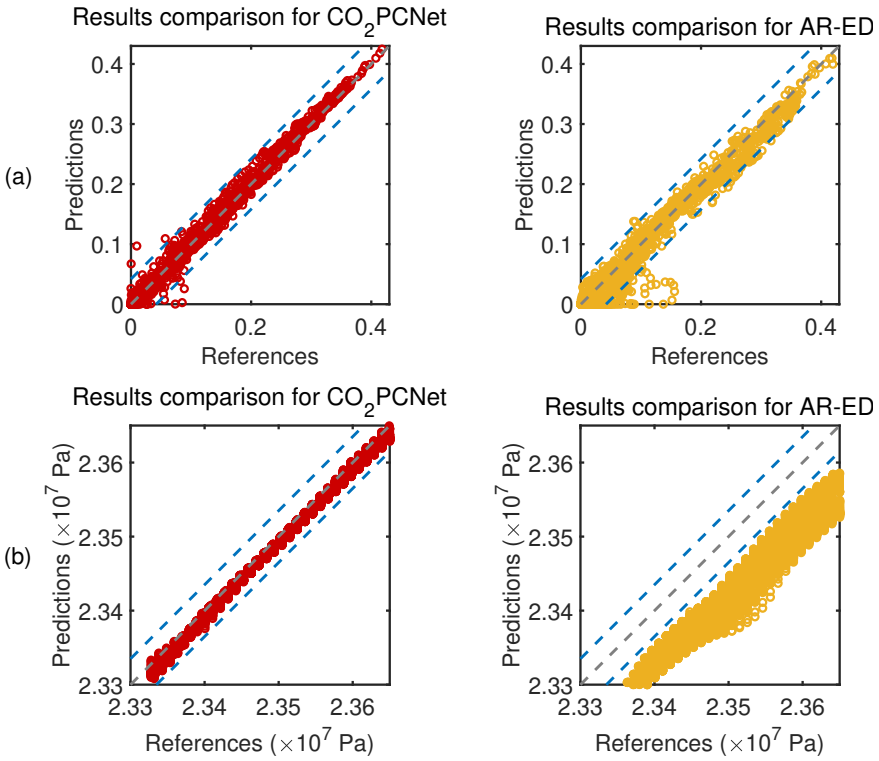


Figure 5.14: Scatter plots of predictions vs. reference values at 2520 days: (a) z_{CO_2} and (b) P_l. CO_2PCNet predictions are shown in red (left), and AR-ED predictions in yellow (right). The dashed gray lines represent perfect predictions, and the blue dashed lines indicate a ±10% error range around the perfect predictions.

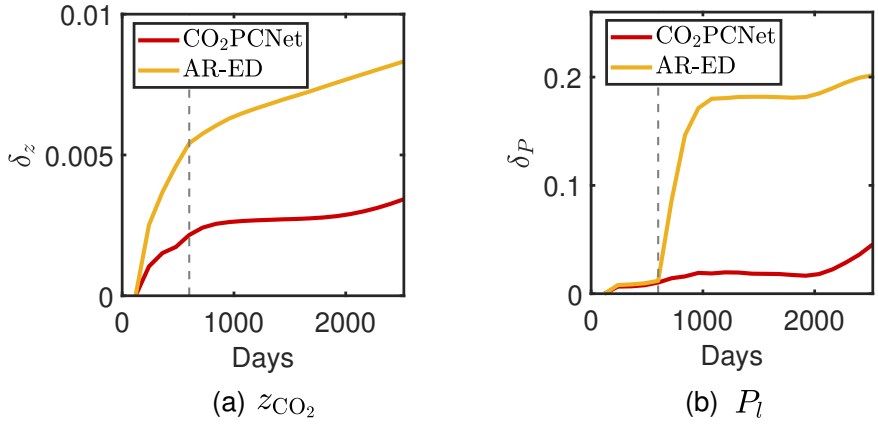


Figure 5.15: Mean relative error trends for CO_2PCNet and AR-ED over a period of 2520 days in a test case. The vertical dashed lines represent the stop of CO_2 injection.

5.4. DISCUSSIONS AND CONCLUSIONS

In this chapter, we proposed and investigated CO_2PCNet, a physics-constrained neural network framework designed for simulating CO_2 injection and migration in deep saline aquifers. By explicitly incorporating the governing partial differential equations for mass conservation into the loss function, CO_2PCNet bridges the gap between purely data-driven surrogate models and physics-based numerical solvers. The network architecture integrates convolutional encoders, ConvLSTM modules, and a decoder to capture the spatiotemporal dynamics of state variables, while respecting the physical constraints governing multiphase flow in heterogeneous porous media.

Through a series of numerical experiments, we have demonstrated that CO_2PCNet provides accurate and stable predictions of key state variables, including overall mass fraction (z_{CO_2}) and liquid pressure (P_l) during both injection and post-injection periods. Compared with purely data-driven models, CO_2PCNet exhibits improved robustness, particularly in extrapolating to future time steps and to geological realizations not seen during training. Importantly, its performance advantage is most pronounced in long-term forecasting tasks, where baseline models tend to accumulate error or produce physically inconsistent results. These findings underscore the benefits of embedding conservation laws directly into the learning objective, rather than relying solely on observed data correlations.

In contrast to earlier deep learning frameworks that predict state variables at fixed time steps, CO_2PCNet enables recursive prediction across the full simulation period. This autoregressive structure, supported by the ConvLSTM component, equips the model with the capacity to capture long-term dependencies and sequential dynamics that are critical in geologic CO_2 storage modeling.

From a broader perspective, this work highlights the promise of hybrid approaches

that leverage the strengths of both physics-based modeling and machine learning. CO_2PCNet offers a computationally efficient and physically consistent alternative to conventional simulators, with implications for uncertainty quantification, optimization, and monitoring design in large-scale CCS projects. It also opens new opportunities for incorporating additional physical processes into future surrogate model designs, such as thermal effects, geomechanics, or reactive transport.

6

ADM METHOD FOR CO_2 STORAGE

The previous chapter introduced a physics-constrained neural network framework that integrates the governing equations into deep learning architectures to simulate CO_2 injection and post-injection dynamics. Although neural surrogates achieve significant computational acceleration and generalization capabilities, their predictive performance remains fundamentally bounded by the representativeness and scope of the training data. In particular, high-resolution simulation of CO_2 plume dynamics under complex geological conditions still requires computational frameworks that can efficiently resolve fine-scale heterogeneities without incurring prohibitive costs. This limitation is especially critical in the context of long-term CO_2 storage, where interactions among buoyancy forces, capillary effects, and compositional gradients evolve over a wide range of spatial and temporal scales.

In this chapter, we explore an alternative simulation strategy based on the Algebraic Dynamic Multilevel (ADM) method. ADM is specifically designed for compositional multiphase flow in heterogeneous saline aquifers, coupling a fully implicit numerical formulation with an adaptive grid hierarchy. Local grid resolution is guided by a front-tracking criterion applied to the gradients of overall component compositions, ensuring enhanced resolution in regions where sharp composition changes occur. By constructing localized multiscale basis functions over a hierarchy of coarse and fine grids, the method effectively captures essential physical processes while preserving computational efficiency. The use of algebraic prolongation and restriction operators enables a consistent projection of primary variables across grid levels, supporting dynamic adjustment of grid resolution throughout the simulation.

By applying ADM to a range of benchmark and synthetic aquifer scenarios, this chapter demonstrates its ability to deliver physically accurate solutions with reduced computational cost compared to globally refined discretizations. These results underscore the ADM method's suitability for large-scale, long-term CO_2 sequestration modeling, where both numerical efficiency and fidelity to complex flow behavior are essential.

Parts of this chapter have been published in the Journal of Computational Physics, **Vol. 539** (2025) [152].

6.1. FINE-SCALE SOLVER SETTINGS

This section specifies the fine-scale compositional simulator used as the reference model for the multilevel strategies. The general governing equations, thermodynamic equilibrium, and the overall-composition Newton framework have been introduced in Chapter 2 and 3. Here we focus on the fine-scale implementation details that are essential for reproducing the reference solutions.

Molecular diffusion is included in the fine-scale reference model through a Fick's flux with prescribed mutual diffusion coefficients $D_{c,\alpha}$. In addition, the impact of mutual solubility on the density of brine is evaluated using:

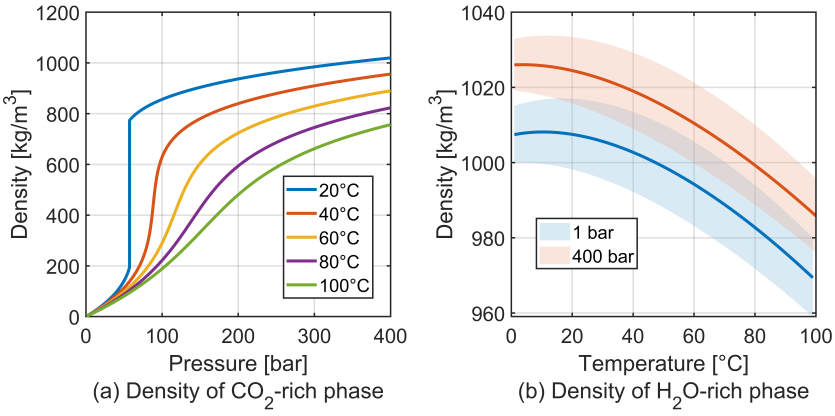
$$\frac{1}{\rho_w(p_w, T, x_{\mathrm{CO_2},w})} = \frac{1 - x_{\mathrm{CO_2},w}}{\rho_w(p_w, T)} + \frac{x_{\mathrm{CO_2},w}}{\rho_{n,\phi}(T)}, \tag{6.1}$$

and

$$\rho_{n,\phi} = \frac{M_{\mathrm{CO_2}}}{10^{-6} \cdot (37.51 - 9.585 \times 10^{-2} T + 8.74 \times 10^{-4} T^2 - 5.044 \times 10^{-7} T^3)}, \tag{6.2}$$

where T is the temperature in Celsius, and $M_{\mathrm{CO_2}}$ is the molar weight of CO_2. $\rho_{n,\phi}$ refers to the apparent density of CO_2 dissolved in water [122]. The densities of the CO_2-rich phase and the H_2O-rich phase, spanning conditions from the surface to the reservoir, are illustrated in Figure 6.1. The density of the gas phase is unaffected by changes in composition. However, changes in phase composition significantly impact the density of H_2O, which in turn influences the migration path of the CO_2 plume. As depicted in Figure 6.1(b), the solid line represents the density curve at a CO_2 mass fraction of 0.05 under corresponding pressures, with the shaded background indicating density variations from a CO_2 mass fraction of 0 to 0.1. This variation in density due to CO_2 solubility in water is a critical factor driving gravity-induced currents, leading to the formation of fingering patterns that significantly contribute to CO_2 trapping through dissolution.

During Newton iterations, because the overall-composition approach tracks the overall mass fraction of CO_2, phase partitioning is determined by thermodynamic equilibrium constraints. If a cell is predicted to be single-phase, certain elements in the Jacobian vanish, simplifying the local flow equations. Conversely, if two phases coexist, the flash calculation accurately allocates the CO_2 between the wetting and non-wetting phases. In this way, the Jacobian automatically accommodates changes in phase presence. Then, the linear system is solved iteratively until the convergence criteria (small changes in pressure and overall mass fraction, or sufficiently low residual norms) are met. In this work, a residual tolerance of 10^{-5} and a solution update tolerance of 10^{-4} are used. Furthermore, the initial and maximum time-step sizes are specified by the user. During the FIM simulation, if the number of Newton iterations exceeds $It_{\max}$, the time-step size is halved and the system solved again. A maximum of 10 time-step chops is allowed.



(a) Density of CO_2-rich phase (b) Density of H_2O-rich phase

Figure 6.1: Density variations in CO_2-rich and H_2O-rich phases under different conditions.

6.2. ADM METHOD FOR CO_2 STORAGE

The ADM enhances computational efficiency in CO_2 storage simulations by dynamically adjusting grid resolutions in response to evolving flow characteristics, such as the sharp CO_2 fronts and well boundaries. This strategy allows for manageable simulations while preserving the fine-scale details accurately.

6.2.1. ADAPTIVE SELECTION OF MULTILEVEL GRIDS

A key feature of ADM is its ability to refine or coarsen the grid at each time step according to a predefined front-tracking criterion. The physical domain is initially discretized into a fine-scale mesh with $N_f = N_{fx} \times N_{fy} \times N_{fz}$ cells, sufficiently refined to capture key CO_2-brine interactions. Although this fully resolved grid can represent the dynamics in detail, running a fully implicit solver across all cells would be impractical for large-scale reservoirs. To address this, ADM constructs a hierarchical series of up to n_l coarser grids, with $l = 0$ denoting the finest mesh and higher l corresponding to progressively coarser resolutions. At grid level l, there are $N_l = N_{lx} \times N_{ly} \times N_{lz}$ control volumes, and the coarsening ratio γ^l is defined as:

$$\gamma^l = (\gamma_x^l, \gamma_y^l, \gamma_z^l) = \left(\frac{N_x^{l-1}}{N_x^l}, \frac{N_y^{l-1}}{N_y^l}, \frac{N_z^{l-1}}{N_z^l} \right), \tag{6.3}$$

with each γ indicating how many fine cells are grouped into one coarse block in the corresponding spatial direction. To maintain smooth transitions, the level difference between neighboring cells is limited to one, preventing sudden changes in resolution that might degrade solution quality.

ADM focuses computational resources on cells that exhibit steep gradients in the

primary unknown z_{CO_2}. Specifically, at each time step t, the grid for the next time step $t+1$ is selected based on the solution at time step t. The strategy compares the maximum difference in z_{CO_2} between two neighboring coarse blocks Ω_l^I and Ω_l^J at coarse level l. Let i and j index the fine-scale cells belonging to the coarse control volumes Ω_l^I and Ω_l^J, respectively. The maximum difference is calculated as:

$$\Delta z_{I,J} = \max_{i\in\Omega_l^I,\ j\in\Omega_l^J} \left|z_i - z_j\right|. \tag{6.4}$$

If $\Delta z_{I,J}$ for any neighboring coarse block exceeds a specified tolerance, the block I is refined from coarse level l to $(l-1)$. Cells surrounding injection wells also remain at the finest resolution to ensure accurate calculations where fluxes are greatest.

6.2.2. SOLUTION STRATEGY

In ADM, the fully-implicit system on the fine grid is transformed to a selected coarser level l by restriction and prolongation operators:

$$\underbrace{\hat{\mathbf{R}}_l^{l-1}\cdots\hat{\mathbf{R}}_1^0\mathbf{J}\hat{\mathbf{P}}_0^1\cdots\hat{\mathbf{P}}_{l-1}^l}_{\mathbf{J}_{\mathrm{ADM}}}\delta\xi_l^{\mathrm{ADM}} = -\underbrace{\hat{\mathbf{R}}_l^{l-1}\cdots\hat{\mathbf{R}}_1^0 r_f}_{r_{\mathrm{ADM}}}, \tag{6.5}$$

where $\mathbf{J}$ and r_f are the Jacobian matrix and residual vector at the fine scale, respectively, and $\delta\xi_l^{\mathrm{ADM}}$ is the solution at grid level l. Restriction operators $\hat{\mathbf{R}}_l^{l-1}$ map residuals of governing equations from level $l-1$ to level l, while the prolongation operators $\hat{\mathbf{P}}_{l-1}^l$ interpolate the solution in the reverse direction. Both $\mathbf{R}$ and $\mathbf{P}$ are block-diagonal matrices:

$$\mathbf{R}_l^{l-1} = \begin{pmatrix} (R_p)_l^{l-1} & \mathbf{0} \\ \mathbf{0} & (R_z)_l^{l-1} \end{pmatrix}_{N_l\times N_{l-1}}, \tag{6.6}$$

and

$$\mathbf{P}_{l-1}^l = \begin{pmatrix} (P_p)_{l-1}^l & \mathbf{0} \\ \mathbf{0} & (P_z)_{l-1}^l \end{pmatrix}_{N_{l-1}\times N_l}. \tag{6.7}$$

Here, the subscript p and z denote the pressure and overall mass fraction sub-blocks, respectively. The same finite-volume restriction operator is typically applied to both variables to ensure the local mass balance, meaning $(R_p)_l^{l-1} = (R_z)_l^{l-1}$. Specifically, each entry of the restriction operator $(R_p)_l^{l-1}$ is either 1 or 0:

$$(R_p)_l^{l-1}(i,j) = \begin{cases} 1 & \text{if cell } j \text{ is inside coarser cell } i, \\ 0 & \text{otherwise.} \end{cases} \tag{6.8}$$

Once the grid hierarchy is established, sequential application of restriction and prolongation transforms the fine-scale FIM system into an ADM system at level l. Solving this coarser system and then prolonging the solution back to the fine grid provides an approximation to the fully resolved solution. This process significantly reduces computational costs compared to solving the original fine-scale system in all cells.

In this work, the linear systems are solved using a direct solver. The final step involves recovering the fine-scale solution $\delta\xi_f$ by iteratively applying all prolongation operators:

$$\delta\xi_f \approx \delta\xi'_f = \hat{\mathbf{P}}^1_f \dots \hat{\mathbf{P}}^l_{l-1} \delta\xi^{\mathrm{ADM}}_l. \tag{6.9}$$

6.2.3. BASIS FUNCTIONS

An important element of accurate ADM simulations is constructing basis functions for pressure and overall mass fraction that can capture fine-scale geological heterogeneity. In this work, constant interpolation is used as the prolongation operator for hyperbolic variables, while multiscale basis functions [153] are employed for pressure to account for variations in transmissibility. The multiscale finite volume approach consists of two sets of overlapping coarse grids, primal and dual coarse grids, superimposed on the fine grid, as illustrated in Figure 6.2. The primal coarse grid comprises N_C control volumes, denoted as Ω^C_i with $i \in \{1,\dots,N_C\}$, while the dual coarse grid consists of N_D local domains Ω^D_j. Multiscale basis functions for pressure are generated by solving a localized numerical problem defined independently within each dual coarse domain Ω^D_j.

Specifically, to localize the flow computation, a reduced boundary condition is implemented for each dual coarse domain, leading to the following localized problem formulation:

$$\begin{cases} -\nabla\cdot\left(\lambda\cdot\nabla\Phi^k_j\right) = 0 & \text{on } \Omega^D_j, \\ -\nabla_\parallel\cdot\left(\lambda\cdot\nabla\Phi^k_j\right)_\parallel = 0 & \text{on } \partial\Omega^D_j, \\ \Phi^k_j(x_i) = \delta_{ki} & \forall\, x_i \in \{1,\dots,N_C\}. \end{cases} \tag{6.10}$$

Here, λ is the mobility, Φ^k_j denotes the basis function associated with coarse node k in dual coarse block Ω^D_j, and δ_{ki} is the Kronecker delta. The subscript $\parallel$ indicates the only the component parallel to the boundary $\partial\Omega^D_j$ is considered. As shown in Figure 6.3, a multiscale pressure basis function is generated by solving localized pressure equations within each coarse block, thereby incorporating the influence of fine-scale transmissibility contrasts.

Figure 6.4 provides a schematic overview of how the ADM method is applied to CO_2 storage. Starting from a fully resolved fine grid, the solver identifies regions of steep solution gradient at each time step using a front-tracking technique and a predefined threshold. Cells with high gradient subdomains remain at fine resolution, along with areas near wells to preserve accuracy. Elsewhere, the domain is discretized at progressively coarser levels, forming a hierarchy of nested grids. Restriction and prolongation operators transfer unknowns and residuals between these levels, enabling an efficient coarse-scale solve before interpolating the solution back to the fine grid. A Newton linearization scheme is employed to iteratively resolve nonlinearities. With this dynamic approach, ADM naturally adapts to evolving CO_2 fronts, capturing the key plume dynamics while optimizing computational resources.

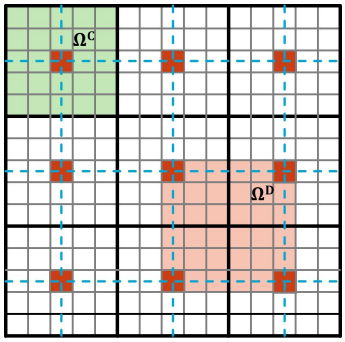


Figure 6.2: Illustration of primal (bold black) and dual (dashed blue) coarse grids at the first coarse level. Fine cells associated with a primal coarse grid (control volume) are shown in green, while those belonging to a dual coarse grid are highlighted in light orange. Coarse nodes are marked in red.

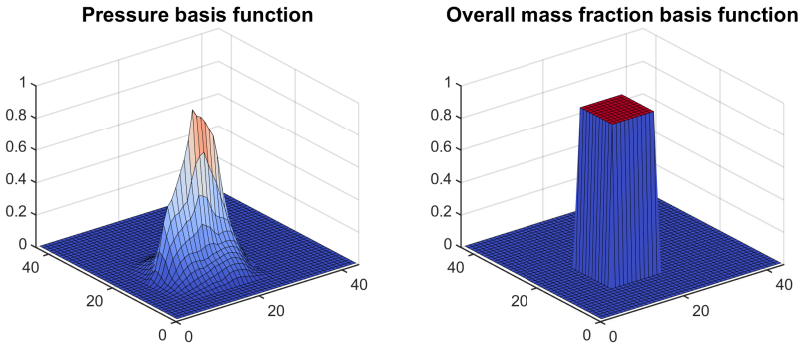
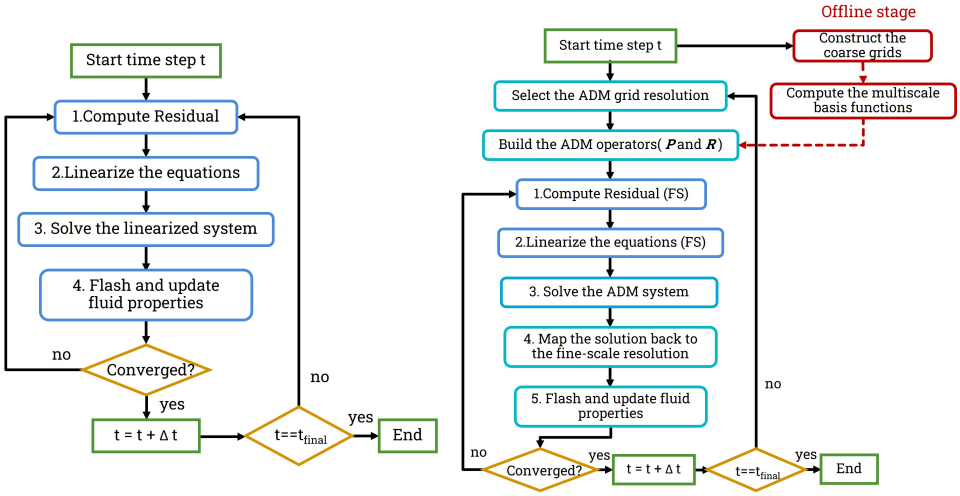


Figure 6.3: Examples of multiscale and constant basis functions for pressure and overall mass fraction in a 2D heterogeneous reservoir.



(a) Fine-scale solution strategy (b) ADM solution strategy

Figure 6.4: Schematic overview of the fine-scale and ADM process for CO_2 storage in deep saline aquifers. ("FS" = fine scale).

6.3. NUMERICAL RESULTS

This section presents a series of numerical results, including a benchmark study, to illustrate the performance of the proposed method. We begin by validating the ADM approach on a two-dimensional synthetic model, then proceed to more complex scenarios to assess the accuracy and efficiency of our framework.

To quantify accuracy for the hyperbolic variables (i.e., x_{CO_2} and z_{CO_2}), the error metric at each time step t is defined as:

$$\epsilon_x(t) = \frac{|x_{CO_2}(t) - x_{CO_2,f}(t)|}{N_f}, \tag{6.11}$$

where $x_{CO_2,f}$ is the fine-scale CO_2 mass fraction distribution, and N_f is the total number of fine cells at time step t. For pressure, we define:

$$\epsilon_P(t) = \frac{\sqrt{\frac{1}{N_f}\sum_{i=1}^{N_f}(P_w^i(t) - P_{w,f}^i(t))^2}}{P_{bc}}, \tag{6.12}$$

where $P_{w,f}$ denotes the fine-scale wetting-phase pressure solution and P_{bc} is a characteristic pressure scale (for instance, a boundary condition value). The average errors over the entire simulation periods are computed as:

$$\epsilon_x = \text{mean}(\epsilon_x(t)), \tag{6.13}$$

$$\epsilon_P = \text{mean}(\epsilon_P(t)). \tag{6.14}$$

6.3.1. TEST CASE 1: VALIDATION OF ADM

To validate the ADM method, we analyze a 2D synthetic model in the $x-z$ plane, which is specifically designed to study both convective mixing and dissolution trapping of CO_2 [154]. The domain extends 100 m horizontally and 50 m vertically, incorporating a static capillary transition zone (CTZ) in the upper 10 m. Within this CTZ, the model maintains a constant CO_2 solubility at $x_{CO_2} = 0.03$, facilitated by large pore volumes. The simulation employs fixed K-values governing the phase partitioning of CO_2 and H_2O. A small amount of water, quantified at $y_{H_2O} = 1.0 \times 10^{-4}$, vaporizes into the CO_2-rich phase, ensuring that the domain primarily consists of a single-phase brine region.

The domain is discretized into a grid of 500 by 250 cells, with each cell measuring 0.2 m × 0.2 m. A diffusion coefficient of $2 \times 10^{-9}\,m^2/s$ applies to both components in both phases. The permeability of rock is set at 100 mD, with a porosity of 0.15. Additional model parameters are consistent with those used in Elenius et al. (2015) [19]. For model validation, simulations are conducted using both fine-scale and ADM strategies, with the ADM model incorporating two coarse levels, each with a coarsening factor of $\gamma = 5 \times 5$. The coarsening criterion based on Δz_{CO_2} differences between neighboring cells is set at thresholds of $\{5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}, 10^{-4}\}$. Both the fine-scale and ADM simulations use the same nonlinear solver tolerances, and the total simulation time is 2000 years. The initial time-step size is set to 0.1 day, and the maximum time-step size is limited to 10 days throughout the simulation.

To quantify the convective mixing, the dissolution rates are calculated, defined as the mass rate of CO_2 transitioning into the single-phase brine region:

$$F = h\phi \frac{\partial \overline{x}_{CO_2}}{\partial t}, \tag{6.15}$$

where h and $\overline{x}_{CO_2}$ are the thickness and the average CO_2 concentration of the single-phase brine region, respectively. Figure 6.5 illustrates the dissolution rate over 2000 years. Initially, there is an increase due to the formation of fingering plumes of dissolved CO_2 which propagate and enhances mass transfer. Around $t_{peel} = 350$ years, the rate transitions toward a shut-down regime, decaying with approximately a $1/t^2$ slope due to the merging of the plumes at the aquifer bottom, reducing the driving force for further dissolution. The behavior of our simulation aligns well with the previous and analytical studies.

We also evaluate the sensitivity of ADM to various refinement criteria, Δz_{CO_2}. Lower thresholds provide finer local resolution in regions with steep composition differences, enhancing the accuracy of convective flow capture. Notably, at the lowest threshold ($\Delta z = 10^{-4}$), the ADM solution curve nearly overlaps with the fully resolved fine-scale reference. Conversely, higher thresholds save computational effort but may overlook some flow details.

Figure 6.6 compares the CO_2 mass fraction profiles after 350 years for the fine-scale solution and the four ADM configurations, while Figure 6.7 presents the error curves over time (left) and the percentage of active grid cells (AGC) in the ADM simulations

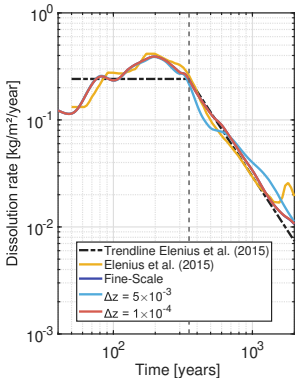


Figure 6.5: Dissolution rate over 2000 years in a 100×50 m domain with a stagnant CTZ.

(right). It is worth noting that lower Δz_{CO_2} thresholds produce closer agreement with the fine-scale solution but require more active grid cells, reflecting a trade-off between computational efficiency and model accuracy. At a threshold of 10^{-3}, the ADM achieves good accuracy while conserving computational resources compared to the fully resolved grid. In this case, the AGC approaches 100% after approximately 200 years, reflecting the fact that the simulation is configured to fully capture the onset and evolution of gravity-driven spreading and convective mixing processes.

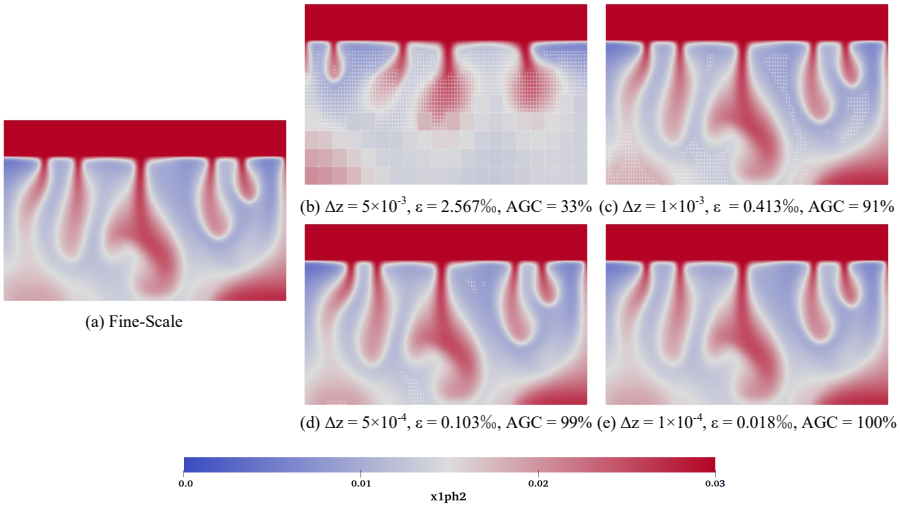


Figure 6.6: CO_2 mass fraction profiles at 350 years. Fine-scale reference is shown alongside ADM solutions at $\Delta z = \{5\times10^{-3}, 10^{-3}, 5\times10^{-4}, 10^{-4}\}$. The overlaid grid lines show the first and second levels of coarsening.

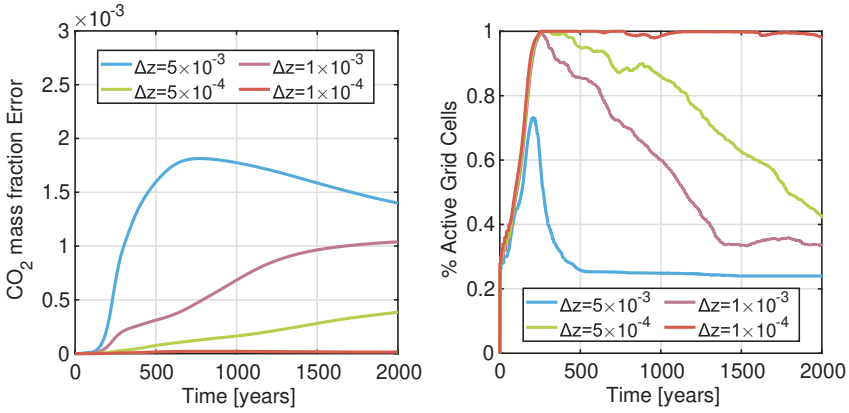


Figure 6.7: CO_2 mass fraction errors (left) and percentage of active grid cells (right) over time for different threshold settings, illustrating the precision-efficiency balance.

6

6.3.2. TEST CASE 2: SCALABILITY EVALUATION IN 3D AQUIFERS

This test case investigates the scalability and efficiency of the ADM method in three-dimensional homogeneous aquifers of varying sizes. First, we consider a 100 m × 28 m × 19 m homogeneous reservoir with uniform porosity $\phi = 0.2$ and absolute permeability $k = 8.0 \times 10^{-14}\,\mathrm{m}^2$. Initial pressure is set at 2.5×10^7 Pa and the temperature is maintained at 65°C. The simulation setup includes an injection well at the lower-left corner with a rate of 4.0×10^{-5} pore volume per day and a production well at the upper-right corner, maintaining a bottom hole pressure of 2.5×10^7 Pa. The reservoir is discretized using a Cartesian mesh with cell dimensions of 1 m in each direction, resulting in a fine-scale grid of 100 × 28 × 19 cells. For the ADM simulation, two coarse levels are employed, each with a coarsening ratio of 3 in the x, y and z directions. A coarsening criterion of 0.01 is used, indicating that cells exceeding this threshold in z_{CO_2} gradient remain refined to capture significant changes accurately. The initial time-step size is set to 0.01 day, and the maximum time-step size is limited to 5 days.

The CO_2 distribution predicted by ADM closely matches that of the fine-scale reference after 200 days, as illustrated in Figure 6.8. The figure highlights the coarse-to-fine transition of grid cells in regions of significant overall CO_2 mass fraction changes, while other areas are coarsened without compromising the accuracy of the simulation. Indeed, error calculations using equation (6.11) show a difference of only 1.5×10^{-6}, demonstrating ADM's effectiveness at reducing computational effort while preserving key flow characteristics.

To further explore the influence of problem size on the ADM algorithm, three ho-

mogeneous reservoirs of different dimensions are considered: 100 m × 28 m × 19 m, 199 m × 55 m× 19 m and 397 m × 109 m × 19 m. Each model is discretized with $\Delta x = \Delta y = \Delta z = 1$ m, and the same permeability, porosity and well configuration are applied. The injection rates are adjusted relative to pore volume, with each simulation extending over 5000 days, including 600 days of injection followed by 4400 days of post-injection monitoring.

Figure 6.9 compares the evolution of active grid cell percentages over the simulation period across varying problem sizes. The results show that ADM efficiency improves with larger domains, especially when injection impacts only a limited reservoir portion. This allows extensive grid coarsening, significantly reducing computational resources needed for long-duration and large-scale simulations. These results highlight ADM's adeptness at dynamically refine critical areas and efficiently coarsen non-critical regions, demonstrating its improved effectiveness in large-scale operations.

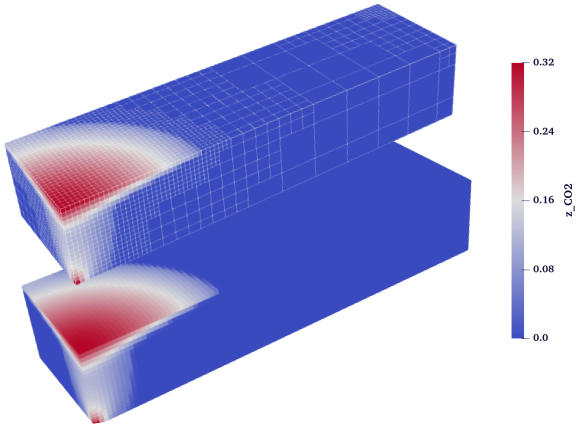


Figure 6.8: Comparative visualization of overall CO_2 mass fraction profiles between ADM (top) and the fine-scale model (bottom) after 200 days.

6.3.3. TEST CASE 3: SPE 11A AT LABORATORY CONDITIONS

Further validation of the ADM framework is conducted through its application to the SPE 11A benchmark model, designed to simulate CO_2 storage under controlled laboratory conditions [155]. This model, which mimics the downscaled characteristics of North Sea storage formations, features a two-dimensional representation encompassing seven distinct facies: one seal (Facies 1), five permeable reservoir sands (Facies 2-6), and one impermeable (Facies 7). These properties are inspired by extensive laboratory experiments and illustrated in Figure 6.10. The properties of these unconsolidated sands are consistent with recent studies from Nordbotten et al. (2022) [156], as summarized in Table 6.1.

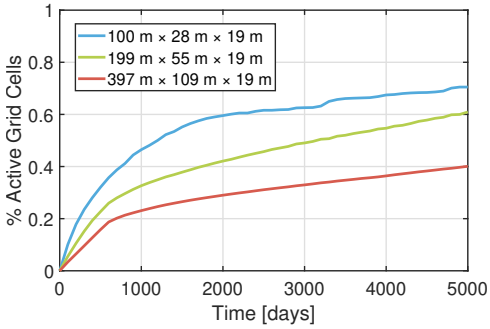


Figure 6.9: Evolution of active grid cell percentages employed by ADM during simulation with the threshold value of 0.01.

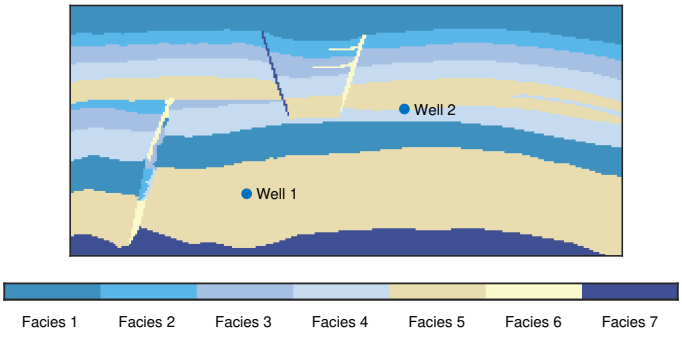


Figure 6.10: Illustration of the discretized two-dimensional geometry of the SPE 11A model, showcasing the arrangement of facies and injection wells locations.

Table 6.1: Physical Parameters and Simulation Setup for the SPE 11A Numerical Simulation.

	$\mathbf{k}$ [D]	ϕ [-]	S_{wi} [-]	P_{entry} [Pa]	D_w [m^2s^{-1}]	D_n [m^2s^{-1}]
Facies 1	44.53	0.44	0.32	1500	10^{-9}	1.6×10^{-5}
Facies 2	506.63	0.43	0.14	300	10^{-9}	1.6×10^{-5}
Facies 3	1013.25	0.44	0.12	100	10^{-9}	1.6×10^{-5}
Facies 4	2026.50	0.45	0.12	25	10^{-9}	1.6×10^{-5}
Facies 5	4953.00	0.43	0.12	10	10^{-9}	1.6×10^{-5}
Facies 6	10132.50	0.46	0.10	1	10^{-9}	1.6×10^{-5}
Facies 7	10^{-8}	10^{-5}	0	0	0	0

The domain geometry measures 2.8 m in length and 1.2 m in height, with a uniform thickness of 0.01 m. The domain is discretized into a grid of 280 × 127 cells, yielding a total of 35560 control volumes. The isothermal conditions are maintained at 20 °C, corresponding to atmospheric pressure. The left, right, and bottom boundaries are treated as no-flow boundaries, while the top boundary is maintained at a constant pressure of 1.1×10^5 Pa and is in direct contact with pure water. The initial time-step size is set to 10^{-6} minute, and the maximum time-step size is limited to 10 minutes throughout the simulation. The simulation monitors CO_2 flow and transport over a total duration of 5 days. Initially, the medium is saturated with water, and CO_2 is injected through two injection wells, as shown in Figure 6.10. The first well operates continuously for the initial 5 hours, injecting CO_2 at a rate of 1.7×10^{-7} kg/s. Subsequently, the second well begins injection at 2.5 hours, and continues until 5 hours, maintaining the same injection rate and thereby overlapping with the first well for half the duration. This setup facilitates detailed observation of CO_2 distribution and phase interactions within the porous medium.

Relative permeability and capillary pressure curves are modeled using the Brooks-Corey equations, adapted to each sand type according to their respective residual saturation values. Specifically, the relative permeability k_r is given by:

$$k_{r,\alpha} = \left(\max\left(\frac{S_\alpha - S_{\alpha,\text{imm}}}{1 - S_{\alpha,\text{imm}}}, 0\right)\right)^2, \tag{6.16}$$

where $S_{\alpha,\text{imm}}$ is the residual saturation of phase α, below which the phase becomes immobile. A residual saturation of 0.1 is specified for the non-wetting phase across all facies. The extended Brooks-Corey function for all saturations is:

$$P_c = P_{c,\max} \cdot \text{erf}\left(\frac{P_{\text{entry}} \cdot (S_{w,n})^{-\frac{1}{2}}}{P_{c,\max}} \frac{\sqrt{\pi}}{2}\right), \tag{6.17}$$

where $S_{w,n}$ is the normalized wetting-phase saturation, P_{entry} denotes the rock entry pressure, and $P_{c,\max} = 2500$ Pa specifies the maximum capillary pressure. The thermophysical properties of pure CO_2 and H_2O phases are obtained from the NIST database [157].

To optimize computational resources, the ADM simulations are conducted with two coarse levels, and the coarsening ratio is $\gamma = 3 \times 3$. We explore four different thresholds $\{5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}, 10^{-4}\}$ for the coarsening criterion based on Δz_{CO_2}. Over the full-cycle period, including both injection and post-injection phase, the system captures CO_2 plume evolution, dissolution, and convective transport.

Figure 6.11 displays CO_2 mass fraction profiles after five hours of injection, comparing the fine-scale baseline with various coarsening thresholds. The illustration highlights the rapid upward and lateral movement of CO_2 driven by buoyancy forces and its interaction with the sealing layer. This visualization demonstrates the effectiveness of the ADM in capturing the early stages of CO_2 plume development under dynamic storage conditions. Figure 6.12, on the other hand, presents the CO_2 distribution after five days, illustrating the post-injection phase, during which the CO_2-saturated upper

layer, denser than the underlying brine, begins to form descending fingers at the interface. These structures indicate the onset of gravitational instability, a critical factor for understanding long-term CO_2 storage efficiency. Although the fine-scale simulation offers a comprehensive understanding of these dynamics, the ADM simulations achieve high accuracy throughout the injection period, with minor errors. During the post-injection phase, reducing the coarsening threshold from $\Delta z_{CO_2} = 10^{-3}$ to $\Delta z_{CO_2} = 5 \times 10^{-4}$ results in only a slight increase in the active grid cells but enhances accuracy. The most refined ADM configuration, with $\Delta z_{CO_2} = 10^{-4}$, achieves the lowest error and highest AGC, effectively replicating the fine-scale model's complex dynamics and enabling precise visualization of individual finger migrations.

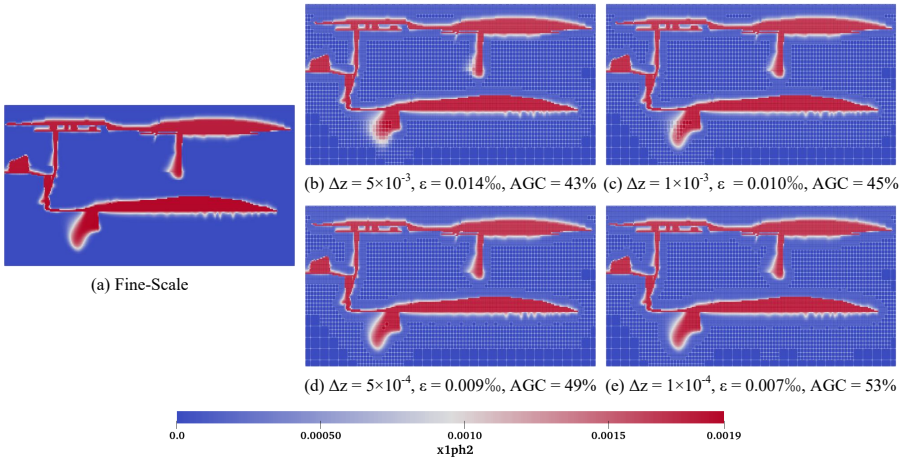


Figure 6.11: CO_2 mass fraction profiles for SPE 11A model after 5 hours injection, showing the finest and coarse grid levels.

The capability of ADM to effectively track these complex features using fewer active grid cells is quantitatively analyzed in Figure 6.13. Figures 6.13(a) and (b) present the ADM error metrics for pressure and overall mass fraction, respectively, throughout the simulation duration. The term "error" refers to the difference between ADM results and the fully resolved fine-scale solution.Figure 6.13(c) details the percentage of active grid cells utilized by ADM and Figure 6.13(d) summarizes the average errors and active cell fractions across all thresholds. This analysis confirms that tighter thresholds, such as 10^{-4}, result in lower errors but require a higher number of active cells, highlighting a trade-off between precision and efficiency. Lastly, Figure 6.14 depicts the fractions of injected CO_2 that remain mobile and that dissolves into the brine over the simulation duration. During injection, most of the CO_2 remains in the mobile phase. Afterward, dissolution and reactivation due to convective transport triggered by density gradients, reactivate a portion of the mobile gas. All ADM configurations closely match the fine-scale reference, indicating that resolving individual fingers in detail is not strictly necessary to accurately capture overall trapping dynamics.

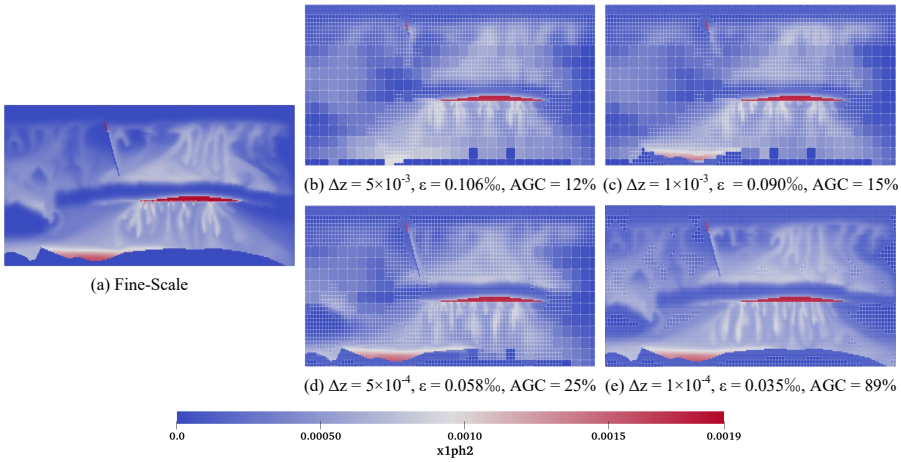


Figure 6.12: CO_2 mass fraction profiles for SPE 11A model after 5 days, illustrating detailed plume behavior across fine and coarse grid levels.

6.3.4. TEST CASE 4: SPE 11B AT RESERVOIR CONDITIONS

Building on the methodologies validated in the laboratory-scale SPE 11A setup, the SPE 11B benchmark extends the simulations to mimic reservoir conditions characteristic of the Norwegian Continental Shelf. While retaining the basic geometric framework of SPE 11A, SPE 11B scales up the model to field dimensions. Specifically, the horizontal dimensions are scaled by a factor of 1:3000 and the vertical dimensions by 1:1000, resulting in a large-scale model measuring 8.4 km in length and 1.2 km in height. Well placements are adjusted accordingly, as shown in Figure 6.15. The model assumes a uniform thickness of 1 m. To accurately reflect reservoir conditions, the facies properties are updated, with details provided in Table 6.2.

Table 6.2: Physical Parameters and Simulation Setup for the SPE 11B Numerical Simulation.

	k_h [mD]	ϕ [-]	S_{wi} [-]	D_w [m^2s^{-1}]	D_n [m^2s^{-1}]
Facies 1	0.10132	0.10	0.32	10^{-9}	2×10^{-8}
Facies 2	101.324	0.20	0.14	10^{-9}	2×10^{-8}
Facies 3	202.65	0.20	0.12	10^{-9}	2×10^{-8}
Facies 4	506.625	0.20	0.12	10^{-9}	2×10^{-8}
Facies 5	1013.25	0.25	0.12	10^{-9}	2×10^{-8}
Facies 6	2026.5	0.35	0.10	10^{-9}	2×10^{-8}
Facies 7	10^{-5}	10^{-5}	0	0	0

While the official SPE 11B benchmark considers thermal effects, this study maintains

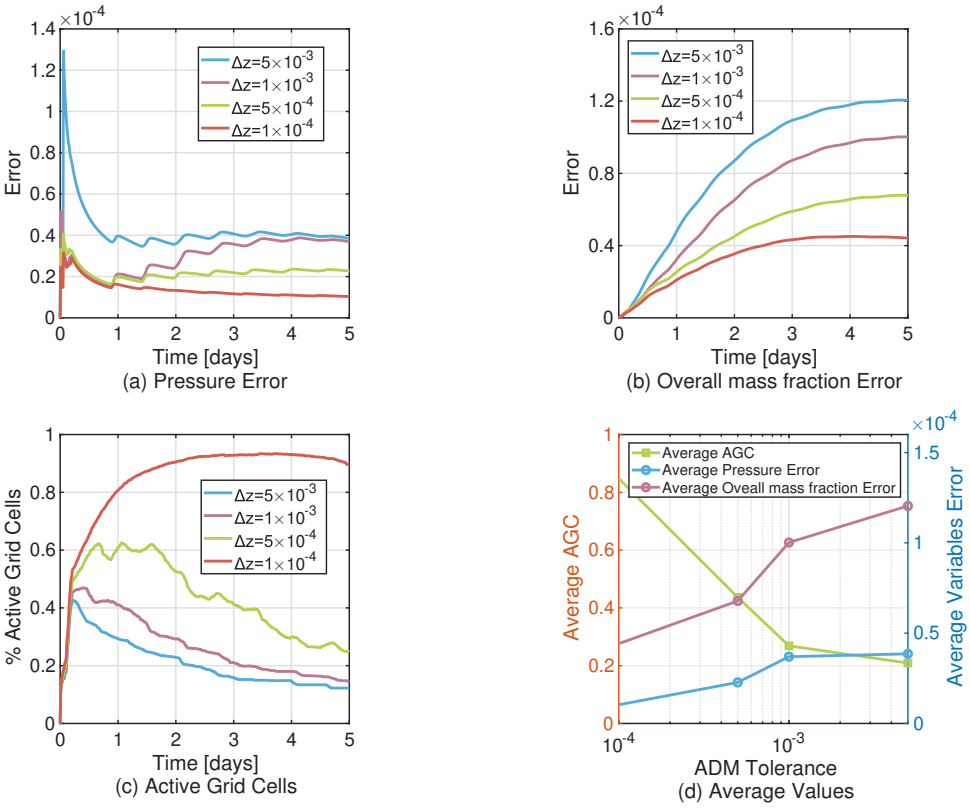


Figure 6.13: SPE 11A error analysis detailing ADM performance across various thresholds, with metrics for pressure, mass fraction errors, and active grid cell usage.

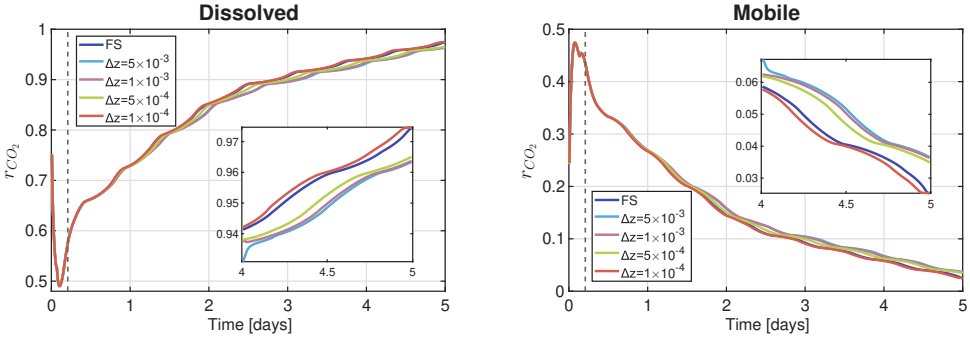


Figure 6.14: Temporal dynamics of CO_2 trapping in the SPE 11A model, illustrated as fractions of injected CO_2 in dissolved and mobile states. The vertical dashed line indicates the transition moment at which injection stops.

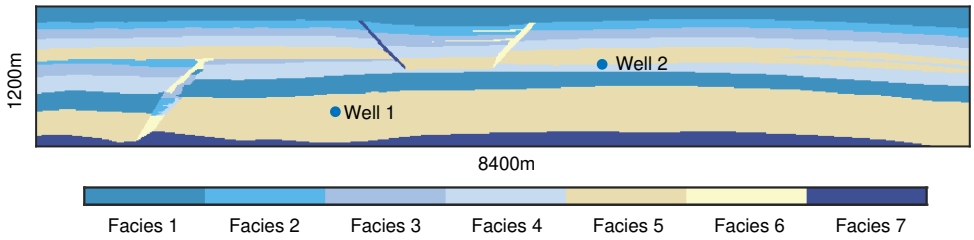


Figure 6.15: Illustration of the SPE 11B model geometry, showing scaled dimensions and well placements in a field scenario.

a focus on mass conservation equations due to the scope of our research. The domain is discretized into a grid of 847×127. The temperature is uniformly maintained at 50°C throughout the simulation. Initially the domain is fully saturated with water, and CO_2 is injected through two injection wells. The first well operates at a constant injection rate of 0.035 kg/s for 50 years. A second well begins injection at the same rate after 25 years, resulting in a 25-year period of overlapping operation. The initial time-step size is set to 1 minute, and the maximum time-step size is limited to 36.5 days throughout the simulation. These simulation setups are crucial for monitoring the long-term migration of the CO_2 plume under these extended temporal and spatial scales.

To prevent unphysical pressure buildup, auxiliary porosity volumes are introduced along the lateral boundaries, facilitating pressure stabilization across the field as detailed in referenced studies [155]. Moreover, permeability is assigned using an anisotropy ratio of 10:1 between horizontal and vertical directions ($k_z = 0.1 k_h$). Differences from SPE 11A include adjustments in capillary pressures, which are defined according to the Leverett J-scaling:

$$P_{\text{entry}} = \sqrt{\frac{\phi}{k_x}} \cdot 6.12 \times 10^{-3} \text{N/m}, \tag{6.18}$$

where 6.12×10^{-3} N/m is derived from Abdoulghafour et al. (2020). [158]. Additionally, shape exponents for relative permeability and capillary forces are adjusted to 1.5, with respective curves for each facies shown in Figure 6.16. The thermodynamic parameters remain consistent with those defined in the SPE 11A model, ensuring physical consistency in the simulation's physical modeling across different test cases. This comprehensive setup allows for an extensive analysis of CO_2 behavior under reservoir conditions, providing valuable insights into its long-term migration patterns.

Figure 6.17 presents the comparative results at the end of the injection phase, utilizing both the ADM method and a fine-scale reference model. The simulations explore varying coarsening thresholds of $\Delta z_{CO_2} \in \{0.1,\ 0.05,\ 0.01,\ 0.001\}$, with a consistent coarsening ratio of $\gamma = 3 \times 3$. In all models, the CO_2 plume rises vertically due to buoyancy, subsequently spreading laterally at sealing layers, illustrating the complex interactions of the plume with geological structures. The fine-scale model provides a baseline for assessing the impact of ADM's varied coarse levels on simulation accu-

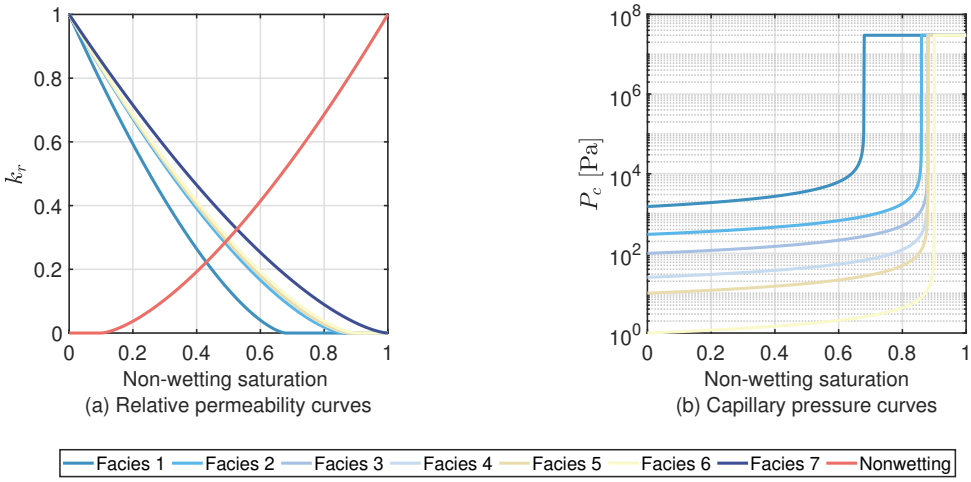


Figure 6.16: Relative permeability and capillary pressure curves for different sand facies as a function of non-wetting saturation.

racy. During the initial injection phase, discrepancies between the ADM simulations remain minimal, largely because the CO_2 plume impacts only a small section of the domain, allowing for a coarser grid in the majority of the domain.

Figure 6.18 displays the long-term CO_2 distribution after 1000 years, highlighting the divergence in simulation accuracy across different coarsening thresholds. As the coarsening threshold decreases from $\Delta z = 0.1$ to $\Delta z = 0.001$, the ADM simulations progressively capture the plume dynamics with increased precision. The lower thresholds, particularly in Figure 6.18(d) and (e), reveal more intricate fingering patterns of CO_2 migration, aligning more closely with those observed in the fine-scale model.

Moreover, Figure 6.19 presents a comprehensive error analysis for SPE 11B simulations across different ADM coarsening thresholds, illustrating how adjustments in Δz impact the accuracy and computational efficiency of the model over a 1000-year simulation period. Figure 6.19(b) depicts the overall mass fraction error, which reveals a distinct trend. Lower Δz values correspond to lower errors, demonstrating improved mass fraction accuracy with finer grid resolutions. This improvement becomes more pronounced over time, underscoring the ADM method's capability to capture detailed physics interactions within the CO_2 plume more effectively as the grid resolution increases. Simultaneously, as Δz decreases, the percentage of active grid cells increases, peaking significantly for the finest threshold ($\Delta z = 0.001$), which reflects the method's dynamic grid adaptation to the evolving simulation demands, particularly in capturing finer details of plume behavior. Figure 6.19(d) integrates these metrics, correlating the average active grid cells, average pressure error, and average overall mass fraction error against the ADM tolerance thresholds, highlighting the trade-offs between computational cost and simulation accuracy.

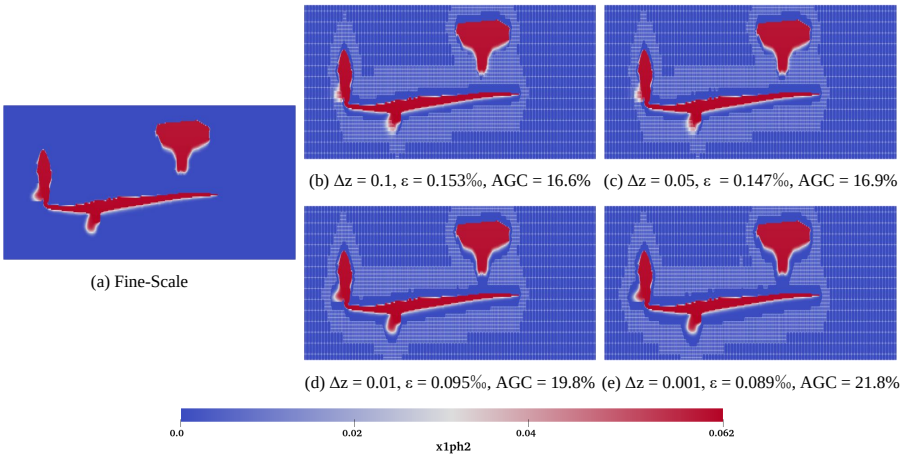


Figure 6.17: CO_2 mass fraction profiles for SPE 11B model after 50 years injection. The overlaid grids represent the first and second levels of coarsening used in the ADM simulations.

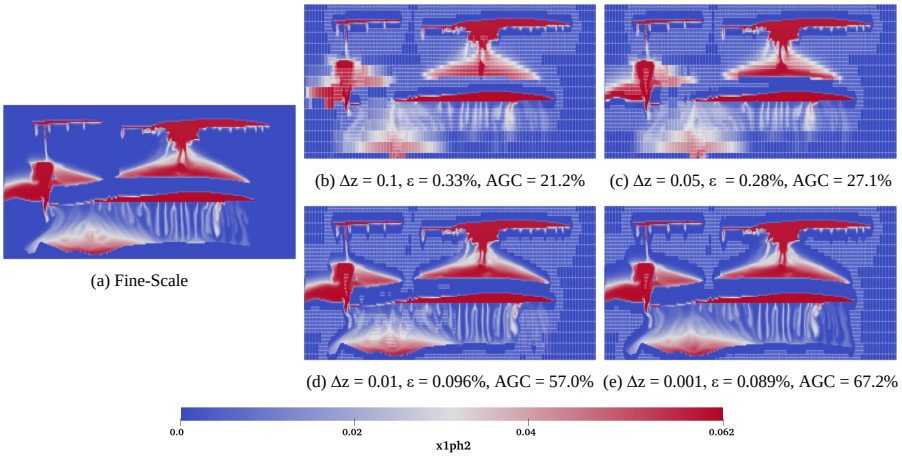


Figure 6.18: CO_2 mass fraction profiles for SPE 11B model after 1000 years. The overlaid grids represent the first and second levels of coarsening used in the ADM simulations.

Lastly, Figure 6.20 shows that varying ADM coarsening thresholds has only a minor impact on the quantification of CO_2 in both dissolved and mobile phases over the 1000-year simulation. This suggests that, even with broader coarsening thresholds, the ADM method still effectively captures the essential dynamics of CO_2 trapping mechanisms in deep saline aquifers. The fractional amounts of injected CO_2 in both dissolved and mobile phases follow similar trajectories across different thresholds, indicating robustness in the ADM's ability to model long-term behavior regardless of the grid coarseness. Even with relatively larger thresholds, the method provides a rapid and efficient quantification approach without significantly compromising the accuracy, making it suitable for extensive parameter studies or scenarios with limited computational resources.

Overall, these findings confirm that the ADM approach validated against fine-scale models retains its ability to capture key CO_2 transport phenomena, including dissolution-driven density changes and extended migration patterns.

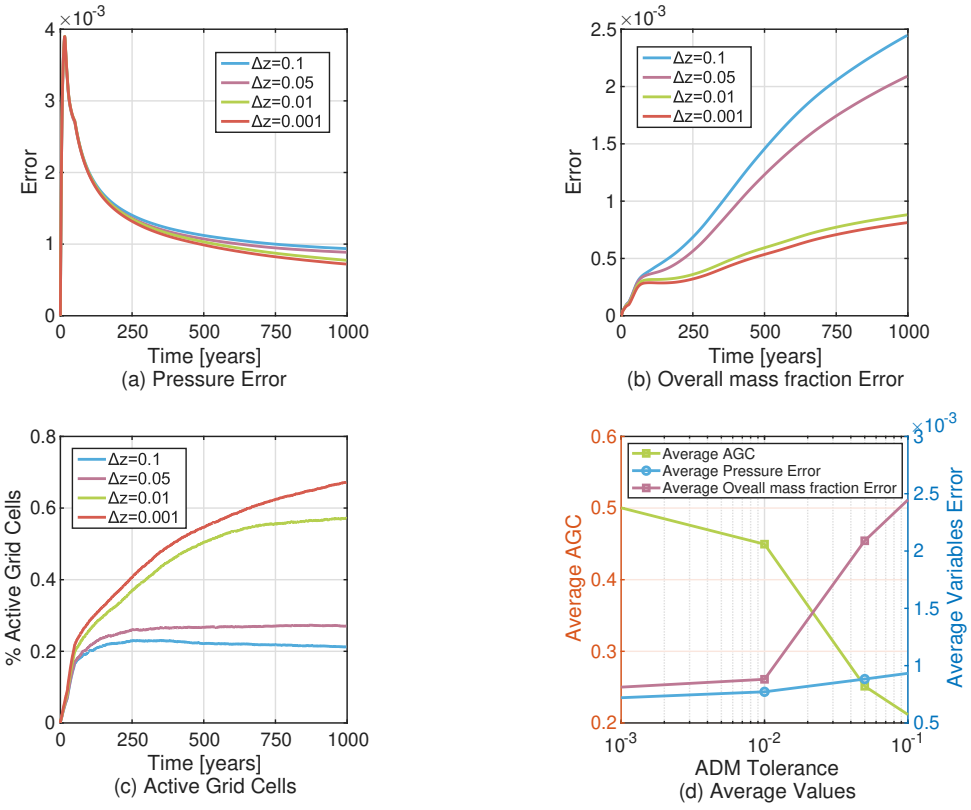


Figure 6.19: Error analysis for the SPE 11B ADM simulations, showing variations in pressure and mass fraction errors alongside the percentage of active grid cells used across different coarsening thresholds.

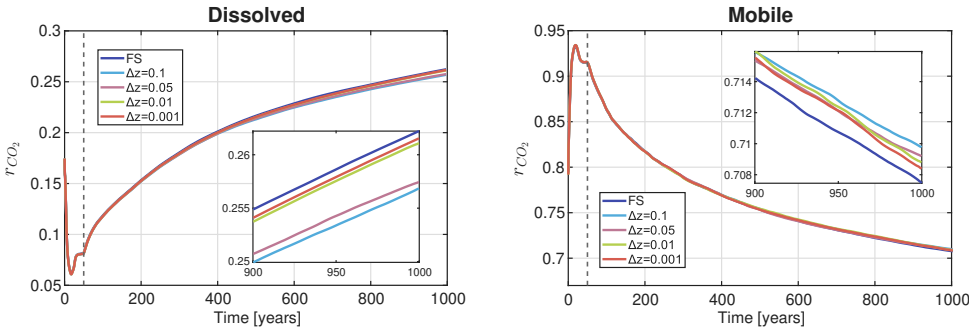


Figure 6.20: Temporal dynamics of CO_2 trapping in the SPE 11B model, illustrated as fractions of injected CO_2 in dissolved and mobile states. The vertical dashed line indicates the transition moment at which injection stops.

6.4. CONCLUSIONS

This chapter presented a detailed fine-scale and multiscale modeling framework for simulating multiphase, multicomponent CO_2-brine flow in deep saline aquifers, with particular focus on the Algebraic Dynamic Multilevel (ADM) method. A fully compositional fine-scale formulation incorporating thermodynamics, capillarity, and molecular diffusion was established as the reference model. Spatial discretization was carried out using the finite volume method, coupled with a fully implicit time integration scheme.

Building on this fine-scale foundation, the ADM framework was introduced as a multilevel algebraic strategy for reducing computational cost while preserving critical physical fidelity. By dynamically adjusting the resolution of the simulation domain based on local gradients of the overall CO_2 composition, ADM efficiently concentrates numerical effort in regions exhibiting sharp fronts, fingering patterns, or near-well transients. The approach employs a hierarchy of nested grids, along with algebraically constructed restriction and prolongation operators, to map residuals and unknowns across levels. Multiscale basis functions for pressure, combined with constant prolongation for composition, further enable accurate flow representation in heterogeneous porous media.

A suite of numerical experiments, ranging from laboratory to field-scale models, was conducted to evaluate the performance and scalability of the ADM method. Across all test cases, ADM consistently reproduced key features of CO_2 migration and trapping with high accuracy while substantially reducing the number of active grid cells. The method captured dissolution-driven convection, plume fingering, and stratigraphic accumulation with minimal error relative to fully resolved simulations. Sensitivity analyses confirmed that appropriately chosen refinement thresholds (Δz_{CO_2}) enable a favorable trade-off between computational efficiency and solution accuracy: tighter thresholds resolve fine-scale features more precisely, whereas looser criteria still preserve global plume dynamics and trapping statistics at reduced cost.

In the context of geological CO_2 storage, these findings underscore the capability of ADM to support long-term simulations of complex flow systems without incurring prohibitive computational burden. Importantly, the method enables reliable quantification of critical storage metrics, such as mobile and dissolved CO_2 fractions, even when small-scale details are only partially resolved. As such, ADM provides a scalable and accurate framework suitable for field-scale sensitivity analyses, and optimization of injection strategies under geological uncertainty. The demonstrated robustness and efficiency of the method make it a valuable computational tool for advancing predictive modeling in CCS applications.

7

MULTISCALE SIMULATION OF CO_2 STORAGE IN FRACTURED AQUIFERS

In the previous chapter, the Algebraic Dynamic Multilevel (ADM) method was introduced as an efficient multiscale simulation strategy for modeling CO_2 injection and migration in geologically heterogeneous saline aquifers. By dynamically adjusting spatial resolution based on composition gradients, the ADM framework enables accurate tracking of plume evolution and phase interactions while substantially reducing computational cost compared to fully resolved fine-scale simulations.

However, many subsurface formations targeted for CO_2 sequestration are naturally fractured or may develop induced fractures during injection operations. These fractures can significantly alter flow dynamics: highly permeable fractures may enhance preferential pathways and elevate leakage risks, while low-permeability or sealing fractures may compartmentalize the reservoir and induce sharp pressure discontinuities. Therefore, reliable modeling of CO_2 storage under such conditions requires explicit representation of fracture–matrix interactions and adaptive control over spatial resolution in both domains.

To this end, the present chapter introduces a multiscale simulation framework that combines the projection-based Embedded Discrete Fracture Model (pEDFM) with the ADM strategy, referred to as pEDFM-ADM. This framework retains the core structure of ADM while enabling explicit yet efficient treatment of discrete fractures. At each simulation time step, the fully implicit, compositional fine-scale system, coupled via the finite-volume method, is mapped onto dynamically constructed multilevel grids in both matrix and fracture domains. The grid adaptivity is governed by front-tracking criteria based on local variations in overall CO_2 mass fraction, and resolution mapping is achieved using algebraic prolongation and restriction operators.

The governing equations, thermodynamic constraints, and numerical solution strategies underlying the fine-scale model have already been established in Chapters 2 and 3, with specific compositional formulation details outlined in Sections 2.2 and 3.1.1. Building on this foundation, the present chapter details the formulation, implemen-

tation, and evaluation of the pEDFM-ADM framework for fractured aquifers. Through a series of numerical experiments, we assess the capability of this approach to resolve flow features with fractures, capture critical trapping mechanisms, and maintain computational scalability in complex geological settings.

7.1. pEDFM-ADM FOR CO_2 STORAGE

The pEDFM-ADM framework integrates pEDFM with the ADM method to efficiently simulate multiphase, CO_2-H_2O flow in fractured saline aquifers. In this approach, the fractured aquifer is discretized on a Cartesian grid, with fractures embedded as lower-dimensional features using pEDFM. pEDFM constructs independent grids for the matrix and fracture domains and systematically adjusts matrix-fracture and matrix-matrix transmissibilities to reflect a wide range of fracture conductivities, from highly conductive pathways to impermeable barriers, while ensuring local mass conservation. The ADM component introduces a dynamic multilevel hierarchy that enables efficient simulation by solving the governing equations on coarser grids and reconstructing fine-scale solutions via multiscale interpolator. This approach allows accurate resolution of key features, such as CO_2 fronts or active fractures, without globally refining the grid.

7.1.1. ADAPTIVE SELECTION OF MULTILEVEL GRIDS

At each time-step, the pEDFM-ADM framework adaptively constructs a multilevel solution grid by combining predefined sets of matrix and fracture grid cells. The grid is refined or coarsened dynamically based on the evolving physical state of the system, particularly the movement of the CO_2 concentration front, which is identified according to a user-defined threshold. For instance, the algorithm tracks the overall mass fraction z_{CO_2} and applies fine resolution in regions with steep gradients, while coarsening areas where CO_2 mass fracture evolves more smoothly. This front-tracking strategy ensures efficient use of computational resources without compromising accuracy.

For a three-dimensional aquifer, the physical domain is initially discretized into a fine-scale grid consisting of N_m cells in the rock matrix and N_{f_i} cells for each fracture i. Although this resolution captures detailed CO_2-brine interactions, solving fully implicit systems across all cells becomes computationally prohibitive for large-scale problems. To address this, pEDFM-ADM constructs a hierarchy of coarser grids, indexed by level l, where $l = 0$ corresponds to the fine-scale grid. Let N_m^l and $N_{f_i}^l$ denote the number of control volumes at level l in the matrix and in fracture i, respectively. The coarsening ratio γ^l at level l is defined as:

$$\gamma^l = (\gamma_m^l, \gamma_{f_1}^l, \cdots, \gamma_{f_{N_f}}^l) = \left(\frac{N_m^l}{N_m^{l-1}}, \frac{N_{f_1}^l}{N_{f_1}^{l-1}}, \cdots, \frac{N_{f_{n_{\text{frac}}}}^l}{N_{f_{n_{\text{frac}}}}^{l-1}}\right). \tag{7.1}$$

This framework allows for independent coarsening strategies in the matrix and individual fractures, offering flexibility. To maintain numerical stability, the level difference between neighboring cells is restricted to one, thereby avoiding abrupt transitions that could degrade solution quality.

The selection of the grid resolution is guided by a threshold-based criterion, specified as an input parameter. This refinement criterion compares the spatial variation of z_{CO_2} between neighboring cells. Let Ω_l^I and Ω_l^J denote two neighboring coarse grid cells at level l, and i and j indicate the indices of fine-scale cells contained in these coarse blocks. The maximum difference of the z_{CO_2} is computed as:

$$\Delta z_{I,J} = \max_{i\in\Omega_l^I,\ j\in\Omega_l^J} \left| z_i - z_j \right|. \tag{7.2}$$

If the difference $\Delta z_{I,J}$ for any pair of neighboring coarse blocks exceeds a specified threshold, the block I is refined from coarse level l to $(l-1)$.

7.1.2. Solution strategy

At each Newton iteration, pEDFM-ADM constructs a reduced multiscale system via algebraic operators based on equation (3.44), defined on a dynamic multilevel grid updated at the beginning of each time-step. In regions where a coarse grid is employed, a fine-scale solution that accounts for subgrid-scale heterogeneity can be efficiently reconstructed from the coarse-scale solution using algebraically defined restriction (**R**) and prolongation (**P**) operators. These operators are constructed to ensure local mass conservation in the reconstructed fine-scale fields. The underlying ADM methodology is described in detail by Cusini et al. [87]. Figure 7.1 provides a schematic overview of how the pEDFM-ADM method is applied to CO_2 storage.

The fully-implicit system on the fine grid is projected to a selected coarser level l by restriction and prolongation operators:

$$\underbrace{\hat{\mathbf{R}}_l^{l-1}\cdots\hat{\mathbf{R}}_1^0\mathbf{J}_0\hat{\mathbf{P}}_0^1\cdots\hat{\mathbf{P}}_{l-1}^l}_{\mathbf{J}_{\mathrm{ADM}}}\delta\xi_l' = -\underbrace{\hat{\mathbf{R}}_l^{l-1}\cdots\hat{\mathbf{R}}_1^0 r_0}_{r_{\mathrm{ADM}}}, \tag{7.3}$$

where $\mathbf{J}_0$ and r_0 are the Jacobian matrix and residual vector at the fine scale, respectively, and $\delta\xi_l'$ is the solution at grid level l. The restriction operators $\hat{\mathbf{R}}_l^{l-1}$ map residuals of governing equations from level $l-1$ to level l, while the prolongation operators $\hat{\mathbf{P}}_{l-1}^l$ interpolate the solution in the opposite direction. Both **R** and **P** are block-diagonal matrices:

$$\mathbf{R}_l^{l-1} = \begin{pmatrix} \left[(R_p)_l^{l-1}\right]_m & 0 & 0 & 0 \\ 0 & \left[(R_p)_l^{l-1}\right]_f & 0 & 0 \\ 0 & 0 & \left[(R_z)_l^{l-1}\right]_m & 0 \\ 0 & 0 & 0 & \left[(R_z)_l^{l-1}\right]_f \end{pmatrix}_{N_l\times N_{l-1}}, \tag{7.4}$$

7

and

$$\mathbf{P}_{l-1}^{l} = \begin{pmatrix} \left[(P_p)_{l-1}^{l}\right]_{mm} & \left[(P_p)_{l-1}^{l}\right]_{mf} & 0 & 0 \\ \left[(P_p)_{l-1}^{l}\right]_{fm} & \left[(P_p)_{l-1}^{l}\right]_{ff} & 0 & 0 \\ 0 & 0 & \left[(P_z)_{l-1}^{l}\right]_{mm} & 0 \\ 0 & 0 & 0 & \left[(P_z)_{l-1}^{l}\right]_{ff} \end{pmatrix}_{N_{l-1}\times N_l}. \tag{7.5}$$

Here, the subscript p and z denote the pressure and overall mass fraction sub-blocks, respectively, and the subscripts 'm' and 'f' refer to the matrix and fractures, respectively. The same finite-volume restriction operator is typically applied to both variables to ensure the local mass balance, meaning $(R_p)_l^{l-1} = (R_z)_l^{l-1}$. Specifically, each entry of the restriction operator $(R_p)_l^{l-1}$ is binary:

$$(R_p)_l^{l-1}(i,j) = \begin{cases} 1 & \text{if cell } j \text{ is inside coarser cell } i, \\ 0 & \text{otherwise.} \end{cases} \tag{7.6}$$

Once the grid hierarchy is established, solving the coarser system and then prolonging the solution back to the fine grid provides an approximation to the fully resolved, fine-scale solution. This process significantly reduces computational costs compared to solving the original fine-scale system in all cells. The final step involves recovering the fine-scale solution $\delta\xi_0$ by iteratively applying all prolongation operators:

$$\delta\xi_0 \approx \delta\xi_0' = \hat{\mathbf{P}}_0^1 \ldots \hat{\mathbf{P}}_{l-1}^{l} \delta\xi_l'. \tag{7.7}$$

7

Previous studies have shown that an iterative procedure is needed for highly-heterogeneous reservoirs [85]. To this end, the pEDFM-ADM method integrates two iterations of a fine-scale smoother to reduce remaining errors to a desired tolerance. This smoother approximates the inverse of the original fine-scale linear operator $\mathbf{A}^{\nu}$, and is implemented using an ILU(0) decomposition.

7.1.3. FRACTURE MULTILEVEL MULTISCALE BASIS FUNCTIONS

In the pEDFM-ADM framework, a non-overlapping primal coarse grid is first constructed on the fine-scale computational domain for both the matrix and fracture cells. By connecting the coarse nodes, an overlapping decomposition is obtained. Local basis functions are then computed for each coarse node within its dual coarse block, following the multilevel grid hierarchy. For hyperbolic variables (e.g., saturation or component mass fraction), constant interpolation is adopted as the prolongation operator, while for the elliptic pressure field, multiscale basis functions following Ţene et al. (2016) [90] are employed to account for heterogeneity in transmissibility.

As with the typical pEDFM method, fractures are discretized as lower-dimensional entities embedded in the matrix grid. Coarse nodes are assigned in the fracture domain similarly to the matrix, ensuring that every fracture contains its own set of coarse

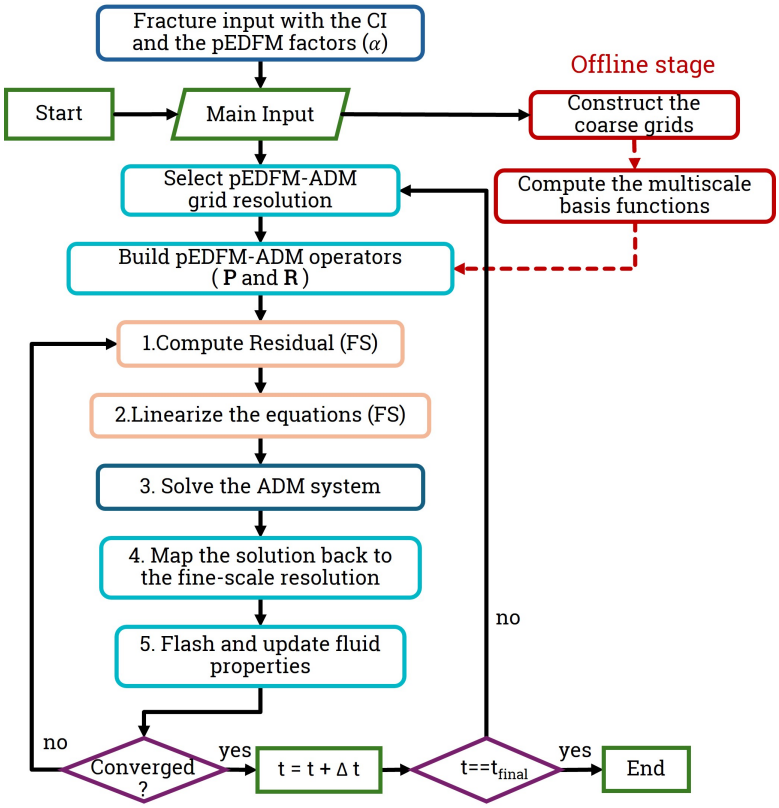


Figure 7.1: Schematic description of the pEDFM-ADM strategy, where 'FS' denotes an evaluation on the fine scale grid.

nodes. As a result, basis functions are constructed not only for matrix coarse nodes but also for fracture coarse nodes. The computation of fracture basis functions follows a similar procedure to that of the matrix; however, the support region of each basis function now extends across both the matrix and fracture media.

This coupling implies that matrix basis functions are influenced by the presence of nearby fractures, and meanwhile, fracture basis functions are also affected by adjacent matrix cells. The resulting basis functions are therefore termed fully coupled basis functions, as they are solved over local dual coarse domains that include both matrix and fracture cells, with all intermediate interactions preserved.

Mathematically, for each coarse node i, the local basis function Φ_i is obtained by solving the following coupled system over its local domain Ω_i:

$$-\nabla\cdot\left(\lambda^*\nabla\Phi_i^{*\bullet}\right)+\sum_{j\in\text{conn}_{mf}^*}\eta_j^*\mathscr{C}\left(\Phi_i^{*\bullet}\right)+\sum_{j\in\text{perf}_w^*}\beta_j^*\left(\Phi_i^{*\bullet}-\Phi_i^{w\bullet}\right)=0, \tag{7.8}$$

which is solved for each basis function $\Phi_i^{*\bullet}$. Here, η_j^* is the matrix-fracture coupling coefficient, and β_j^* is the well index if wells are present. The operator $\mathscr{C}(\cdot)$ denotes the matrix-fracture coupling term. A detailed description of the basis function entries can be found in [90]. The solution Φ_i is then assembled into column i of the prolongation operator P_p, which maps coarse pressures to the fine-scale solution. These multiscale basis functions enable accurate coarse-grid representation of flow in highly heterogeneous porous media with complex matrix-fracture connectivity. Figure 7.2 shows a surface plot of some matrix and fracture basis functions at two different coarsening levels for a 2D homogeneous domain.

7.2. NUMERICAL RESULTS

This section presents a series of numerical simulations designed to evaluate the effectiveness of the proposed framework. We begin by analyzing the influence of fractures on CO_2 sequestration performance, and then proceed to more complex scenarios to assess both the accuracy and computational efficiency of the method.

The relative permeability curves for the matrix domain are modified based on experimental data reported in [159], and are modeled using the van Genuchten formulation [160]. Capillary pressure curves are derived from the Leverett J-function. In this study, we assume that multiphase interactions within the fractures are negligible. Consequently, the relative permeability in fractures follows linear functions [161], and both capillary pressure and hysteresis effects are neglected in the fracture domain. The primary drainage and imbibition curves for the matrix are illustrated in Figure 7.3.

We assess the accuracy of the pEDFM-ADM method by comparing it with a reference fine-scale simulation that employs the pEDFM method without ADM. To quantify the solution accuracy of pEDFM-ADM method for hyperbolic variables (e.g., x_{CO_2} and

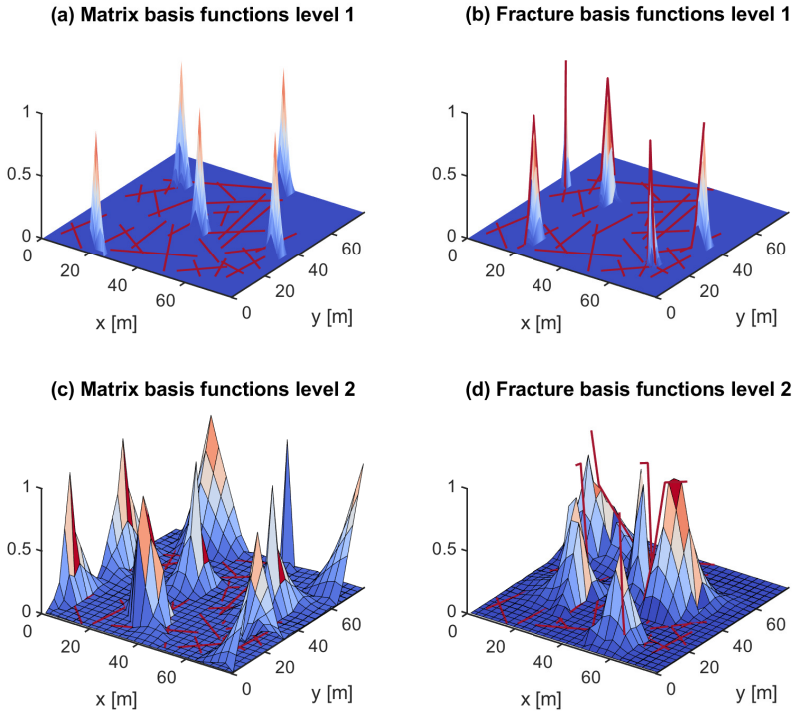


Figure 7.2: Example of multilevel, fully coupled basis functions over multiple dual coarse blocks in a 2D homogeneous fractured domain. Red lines indicate the fracture network, and the basis functions capture both matrix and fracture influences.

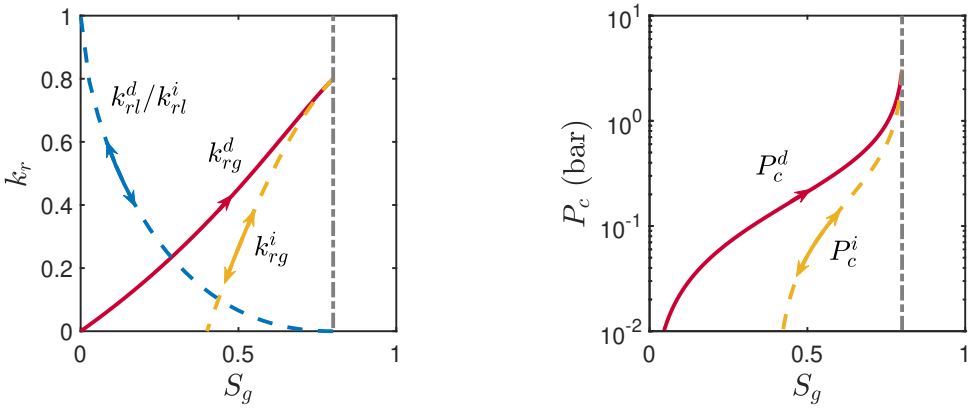


Figure 7.3: Illustration of the constitutive relationships (i.e. relative permeability and capillary pressure curves) used for the matrix domain. Superscripts d and i denote drainage and imbibition, respectively. Single-headed arrows indicate irreversible processes, while double-headed arrows represent reversible ones along the corresponding curves.

z_{CO_2}), the error at each time step t is defined as [152]:

$$\epsilon_x(t) = \frac{\sum_{i=1}^{N} \left| x^i_{CO_2}(t) - x^i_{CO_2,\text{ref}}(t) \right|}{N}, \tag{7.9}$$

where $x_{CO_2,ref}$ is the reference fine-scale solution for the CO_2 mass fraction, and N is the total number of fine cells. For pressure, we define the normalized root-mean-square error as:

$$\epsilon_P(t) = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N} \left(P^i_w(t) - P^i_{w,\text{ref}}(t) \right)^2}}{P_{bc}}, \tag{7.10}$$

where $P_{w,ref}$ denotes the fine-scale solution of the wetting-phase pressure, and P_{bc} is a characteristic pressure scale (for instance, a boundary value). The average errors over the entire simulation period are computed to provide a global measure of solution accuracy:

$$\epsilon_x = \text{mean}(\epsilon_x(t)), \tag{7.11}$$

$$\epsilon_P = \text{mean}(\epsilon_P(t)). \tag{7.12}$$

7.2.1. TEST CASE 1: IMPACTS OF FRACTURES

To systematically investigate the full-cycle behavior of CO_2 storage in a deep fractured saline aquifer, we first consider a two-dimensional conceptual model representing a

reservoir cross-section with a length of 200 m and a height of 50 m. The domain is discretized into a structured grid with 201 cells in the horizontal direction and 51 cells in the vertical direction. The permeability of the rock matrix is set to 8×10^{-14} m^2. A CO_2 injection well is placed on the left boundary, injecting through the bottom 10 m of the formation and terminating after 600 days, while a production well penetrating the full vertical extent of the domain is located at the right boundary. The production well is designed to withdraw formation brine rather than produced CO_2, in order to mitigate pressure perturbations and facilitate CO_2 dispersal throughout the aquifer. Two inclined fractures are embedded in the domain with identical inclination angles but different spatial positions, as shown in Figure 7.4.

To evaluate the impact of fracture conductivity, we vary the fracture permeability while keeping the fracture aperture constant. In each scenario, we consider two logarithmic-scale fracture-to-matrix permeability ratios, defined as $\log(k_f/k_m)$, with values of 4 and -6 representing highly conductive fractures and near-impermeable flow barriers, respectively. The physical parameters and simulation settings used in all scenarios are summarized in Table 7.1.

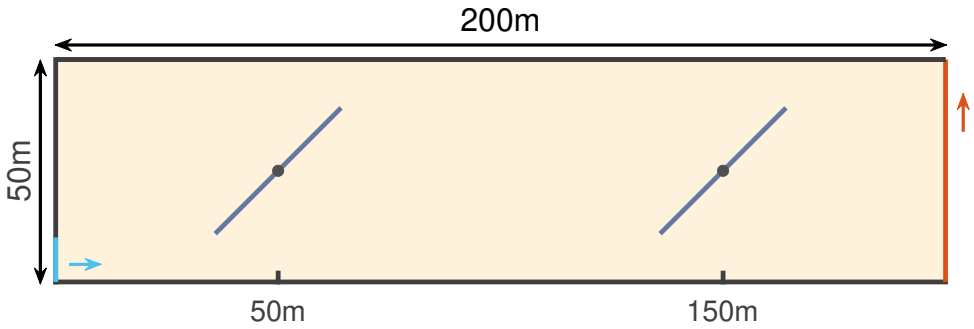


Figure 7.4: Schematic of 200×50 m computational domain and fracture configurations used in Test Case 1. The injector is shown in light blue, the producer in red, and each fracture in dark blue.

Figure 7.5 presents streamline patterns of aqueous phase under three configurations: the base case simulation without fractures, flow barrier fractures, and finally conductive fractures. In Figure 7.5(b), streamlines are visibly diverted around the low-permeability fractures, most clearly near the fracture tips where stagnation zones form. Almost no streamlines pass through the fractures, confirming their role as effective flow barriers. In contrast, in Figure 7.5(c), where the fracture permeability is four orders of magnitude greater than that of the matrix, flow converges into the fractures, accelerates along their length, and re-emerges into the matrix, creating localized acceleration of the brine through the highly permeable pathways of the conductive fractures. The streamline density in the surrounding matrix is noticeably reduced, indicating that transport is dominated by flow through the conductive fractures.

The impact of fracture configuration on the evolution of CO_2 mass fraction is illustrated in Figures 7.6. In the base case without fractures, gravitational instabilities

Table 7.1: Physical Parameters and Simulation Setup for the DARSim Numerical Simulation.

Aquifer length	200	m
Aquifer height	50	m
Fracture length	40	m
Fracture aperture	5.0e-3	m
Fracture inclination angle	45	°
Matrix porosity	0.2	-
Matrix permeability	8×10^{-14}	m^2
Initial pressure	2.5e7	Pa
Bottom hole pressure	2.5e7	Pa
Temperature	323.15	K
Injection rate	1.0e-4	pore volume/day
CO_2 density at STC	1.98	kg/m^3
H_2O density at STC	998	kg/m^3
Simulation time	3.2e4	day
Injection time	6.0e2	day

develop near the top boundary due to the density difference between CO_2-rich and CO_2-poor brine. These instabilities give rise to downward-propagating convective fingers, which enhance dissolution trapping by increasing the contact area between CO_2 and resident brine [99]. In the low-permeability fracture case, some vertical finger development cannot cross the barriers, but the overall convective pattern remains similar to the base case. In the high-permeability case, CO_2-rich fluid rapidly enters the conductive fractures, descends to greater depths, and re-enters the matrix, driving large-scale flow patterns on both sides of the fracture network.

Figure 7.7 further shows the temporal evolution of both dissolved and residual trapping fractions. Dissolution increases progressively as the CO_2 plume interacts with undersaturated brine. Residual trapping, on the other hand, is primarily controlled by capillary hysteresis. After injection stops, brine, as the wetting phase in most sedimentary formations, imbibes into the trailing edge of the non-wetting CO_2 plume. This imbibition process leaves disconnected and immobile CO_2 ganglia behind [113]. In the conductive fracture case, residual trapping is lower because the injected CO_2 is funneled into high-permeability pathways, leaving a smaller fraction of the pore space available to immobilize CO_2. In contrast, dissolution trapping is higher because the fracture-controlled flow locally enhances mixing and interfacial contact between CO_2-rich and CO_2-poor brine.

7.2.2. TEST CASE 2: VALIDATION OF PEDFM-ADM

Building on Test Case 1, we further evaluate the capability of the proposed pEDFM-ADM framework in capturing the dynamics of CO_2 migration and trapping in Test Case 2 by comparing with a fine-scale solution. The pEDFM-ADM simulations em-

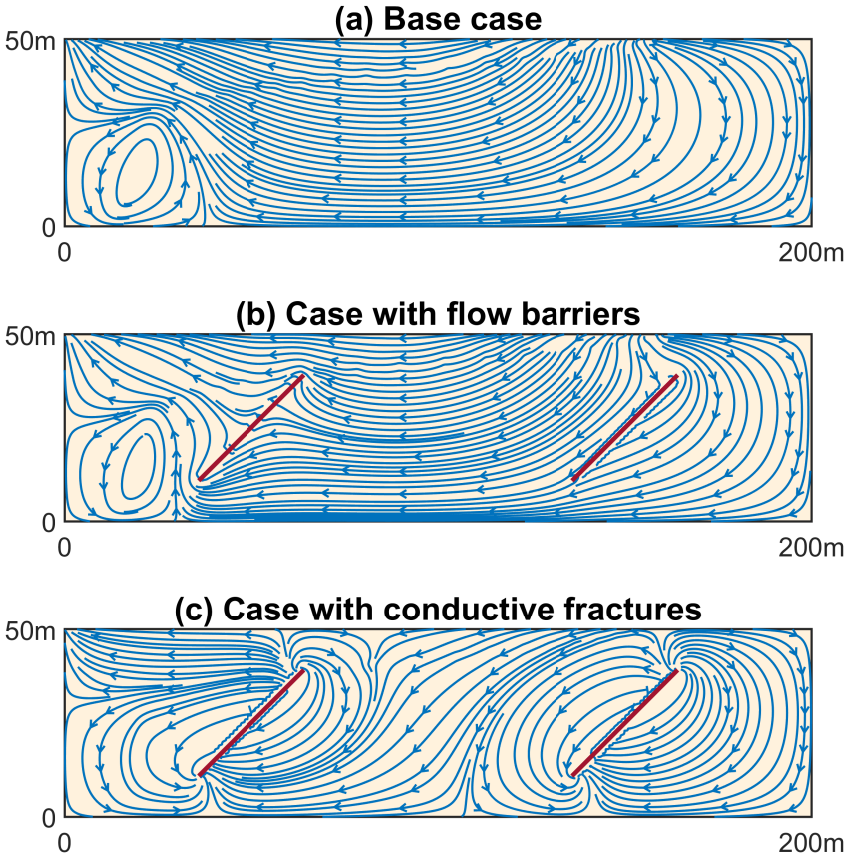


Figure 7.5: Streamline distribution of aqueous phase at 1200 days for Test Case 1.

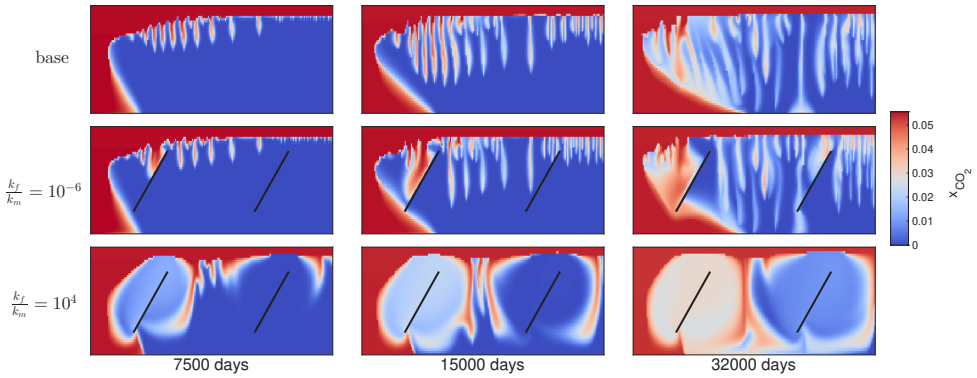


Figure 7.6: Snapshots of CO_2 mass fraction at 7500, 15000, 32000 days for Test Case 1. The aspect ratio of the figures has been adjusted for improved visualization.

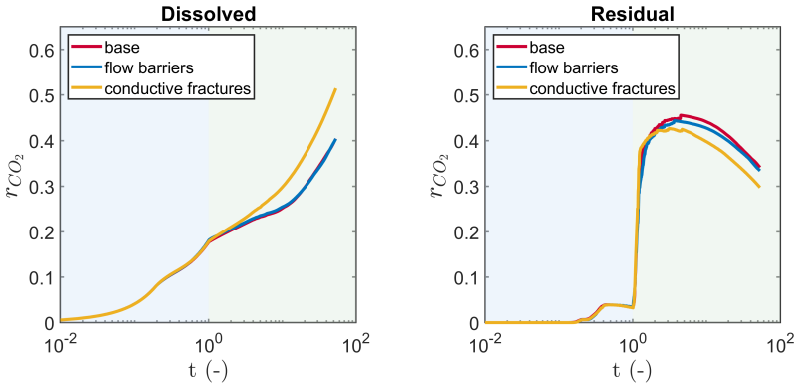


Figure 7.7: Temporal evolution of the fraction of injected CO_2 trapped by different mechanisms. The elapsed time (t) is nondimensionalized with respect to the duration of the injection period. The light-blue shaded region denotes the injection phase, while the light-green shaded region corresponds to the post-injection phase.

ploy two coarse levels, each with a uniform coarsening ratio of 5 in both x- and z-directions. To evaluate the sensitivity of the adaptive resolution control, we test three threshold values for the front-tracking criterion based on local variations in the overall CO_2 mass fraction, $\Delta z_{CO_2} \in \{10^{-1}, 10^{-2}, 10^{-3}\}$. This criterion governs both refinement and coarsening during the simulation, ensuring that high resolution is applied selectively in regions with sharp CO_2 gradients. Over the full simulation cycle, including both injection and post-injection periods, the model captures key processes such as plume propagation, convective mixing, and dissolution, while dynamically adjusting grid resolution in response to the evolving front.

Figures 7.8 and 7.9 present snapshots at $t = 20000$ days for two contrasting fracture scenarios: one with two low-permeability fractures acting as flow barriers ($k_f/k_m = 10^{-6}$), and another with two conductive fractures ($k_f/k_m = 10^4$). For the barrier case, the pEDFM-ADM simulations closely reproduce both the extent and the fine-scale morphology of the plume, including fingering structures. As the refinement threshold tightens, the error drops from 0.43% to 0.14%, accompanied by an increase in the fraction of active grid cells (AGC) from 39% to 91%. Here, active grid cells refer to the fine-scale cells that are explicitly resolved in the simulation at a given time step and higher AGC values mean finer resolution but also greater computational cost. In the conductive-fracture case, even the coarsest setting ($\Delta z_{CO_2} = 10^{-1}$) adequately captures the large-scale convection pattern with only 0.35% error, using merely 37% of the fine-grid resolution.

Figure 7.10 complements these visual comparisons by reporting quantitative accuracy and efficiency metrics. Figures 7.10(a) and (b) show the differences in pressure and overall mass fraction between pEDFM-ADM and the fine-scale reference solution. Both quantities remain consistently small across the simulation, demonstrating that pEDFM-ADM preserves the accuracy of the fine-scale model. The AGC curves in Figure 7.10(c) highlight how the dynamic algorithm selectively refines only the active regions. For the most stringent threshold, the algorithm retains approximately 80% of the fine grid cells on average over the simulation period. As the ADM threshold increases, the number of active cells decreases, which reduces computational cost, but the error in each primary variable correspondingly increases. In this particular test case, the solutions for thresholds of 10^{-2} and 10^{-3} are nearly indistinguishable. These results demonstrate that pEDFM-ADM enables a tunable balance between computational efficiency and accuracy. In both the flow-barrier and conductive-fracture cases, the framework preserves essential features such as solute fingering, and convective mixing, while maintaining acceptable accuracy with reduced computational cost.

7.2.3. TEST CASE 3: FRACTURES WITH COMPLEX GEOMETRIES

To further demonstrate the effectiveness and generality of the proposed pEDFM-ADM framework, we consider a more challenging scenario involving a complex, irregular fracture network, as illustrated in Figure 7.11. The computational domain spans 200 meters in the horizontal direction and 50 meters vertically, and contains 30 fractures with diverse orientations, lengths, and spatial distributions, including both in-

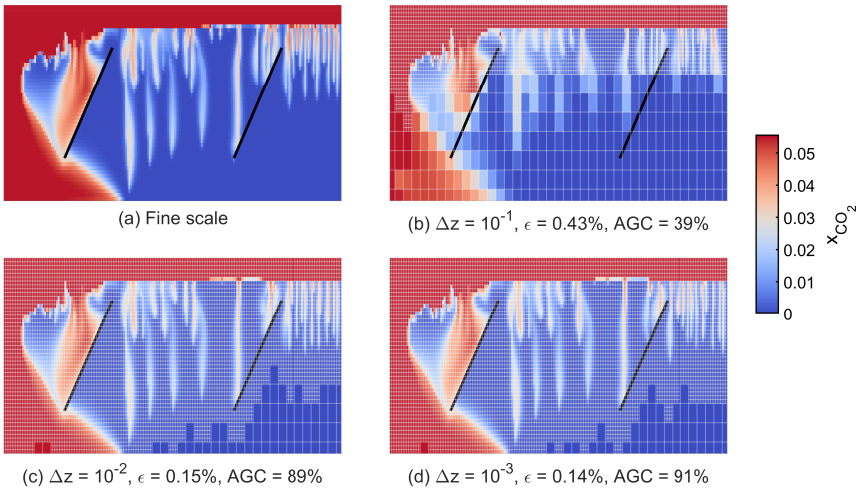


Figure 7.8: CO_2 mass fraction profiles after 20000 days for the case with $\log(k_f/k_m) = -6$, showing the finest and coarse grid levels.

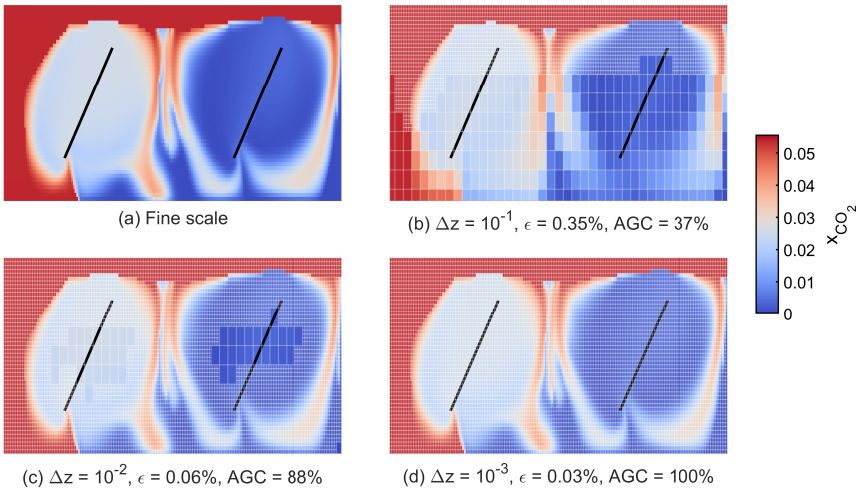


Figure 7.9: CO_2 mass fraction profiles after 20000 days for the case with $\log(k_f/k_m) = 4$, illustrating detailed plume behavior across fine and coarse grid levels.

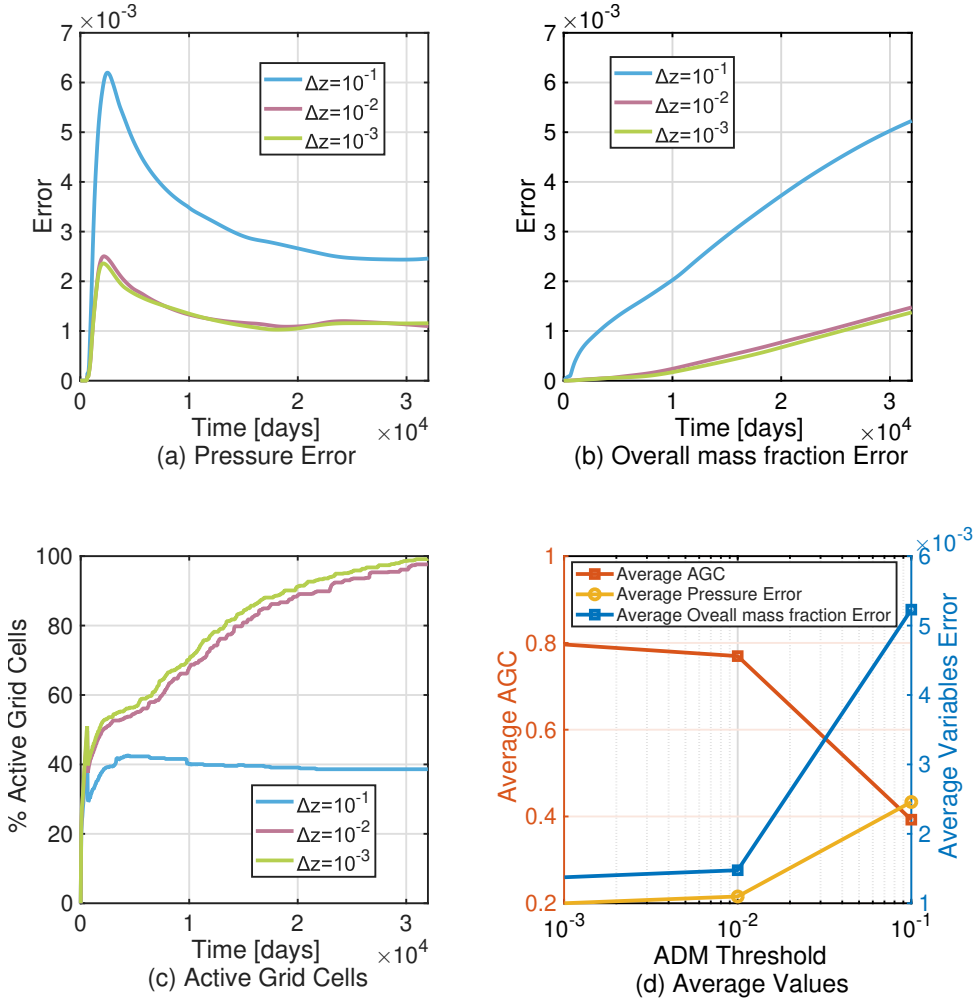


Figure 7.10: Error analysis for case with flow barriers, detailing pEDFM-ADM performance across various thresholds, with metrics for pressure, mass fraction errors, and active grid cell usage.

tersecting and isolated segments.

As with the previous test cases, we simulate two representative scenarios with extreme fracture-matrix permeability contrasts: a low-permeability fracture network with $\log(k_f/k_m) = -6$, and a highly conductive network with $(\log(k_f/k_m) = 4$. For each case, pEDFM-ADM simulations are again performed using three different adaptivity thresholds based on Δz_{CO_2}: $\{10^{-1}, 10^{-2}, 10^{-3}\}$. Boundary conditions and fluid properties are consistent with those listed in Table 7.1.

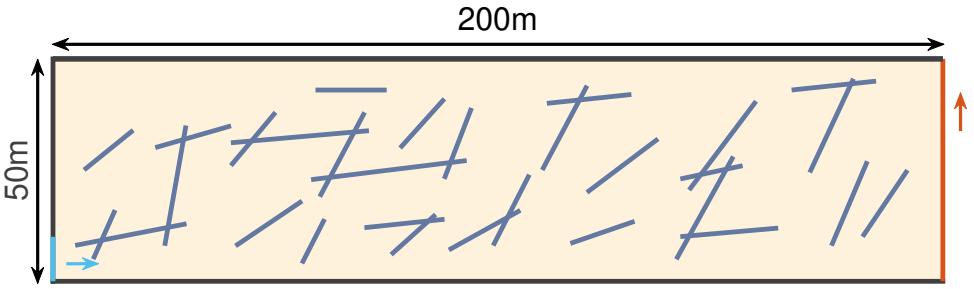


Figure 7.11: Schematic of computational domain and fracture geometries for Test Case 3.

Figures 7.12 and 7.13 present CO_2 mass fraction distributions at intermediate (2500 days) and late (32000 days) times, comparing fine-scale reference solutions with pEDFM-ADM results across varying adaptivity thresholds. For the low-permeability fracture network ($\log(k_f/k_m) = -6$), the fractures act as flow barriers that partition the domain and inhibit vertical plume propagation. At 2500 days, the pEDFM-ADM simulations accurately capture both the leading front and finger morphology with high fidelity compared to the fine-scale solution, with an error of $\epsilon = 0.03\%$ at $\Delta z_{CO_2} = 10^{-1}$, while activating only 48% of the grid cells. At 32000 days, the plume remains largely confined by low-permeability segments, and the pEDFM-ADM results remain consistent with the reference solution, with $\epsilon = 0.32\%$ and 47% active cells at the coarsest threshold.

The high-permeability fracture network ($\log(k_f/k_m) = 4$) functions as a conduit system that channels CO_2 along fracture planes and induces large-scale convective mixing patterns within the matrix. At 2500 days, these transport features are reproduced with $\epsilon = 0.18\%$ using only 37% of the fine grid at $\Delta z_{CO_2} = 10^{-1}$. At 32000 days, the plume geometry becomes increasingly influenced by fracture-controlled mixing. Resolving these structures requires finer adaptivity: the error decreases from $\epsilon = 0.61\%$ at $\Delta z_{CO_2} = 10^{-1}$ to 0.19% at tighter threshold, with corresponding active cell fraction increasing from 30% to 82%. These comparisons illustrate that the pEDFM-ADM concentrates resolution near the front, and that higher fracture conductivity requires finer spatial resolution to accurately represent plume dynamics.

Figure 7.14 quantifies the temporal evolution of errors and the fraction of active grid cells used in the simulation. For the low-permeability fracture network, plume migration is limited by the barrier effect, producing a compact CO_2 plume with rel-

atively sharp z_{CO_2} gradients along its boundary. Since the pEDFM-ADM algorithm refines only in regions of steep gradients, most of the reservoir can be represented on a coarse grid, with fine-scale resolution concentrated along the advancing front. In contrast, for the high-permeability fracture network, the plume rapidly disperses throughout the reservoir, generating large-scale mixing and numerous localized regions of CO_2 gradients. Accurately resolving this transport pattern requires fine-scale resolution across much of the domain, keeping the active cell fraction high for an extended period. Figure 7.15 aggregates the dissolved and residual trapping fractionsto visualize the total trapping occurring in each simulation. The conductive network exhibits much stronger dissolution due to fracture-induced mixing and enhanced interface area. Residual trapping primarily occurs during the post-injection period. Across all thresholds, including $\Delta z_{CO_2} = 10^{-1}$, the pEDFM-ADM solutions closely follow the fine-scale curves, demonstrating the capability of the framework for rapid yet reliable quantification of trapping metrics, even under relatively coarse adaptive thresholds.

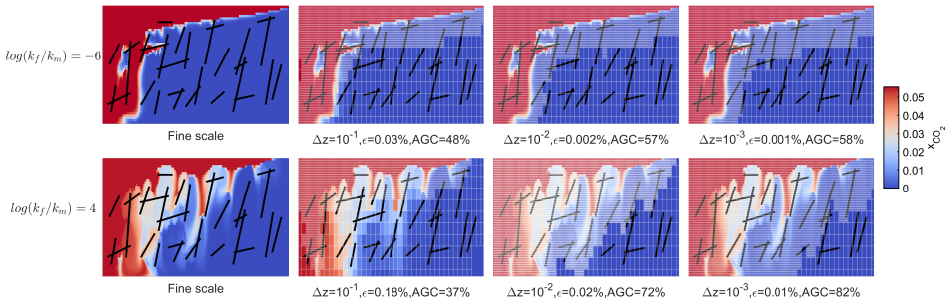


Figure 7.12: CO_2 mass fraction profiles at 2500 days for Test Case 3. Fine-scale results are compared with pEDFM-ADM solutions at thresholds $\{10^{-1}, 10^{-2}, 10^{-3}\}$. Overlaid grid lines show local coarsening levels.

7.2.4. TEST CASE 4: HETEROGENEOUS AQUIFER WITH MIXED-CONDUCTIVITY FRACTURES

In this section, we consider a complex scenario that couples heterogeneous matrix permeability with fractures of contrasting hydraulic roles (i.e. both conductive fractures and flow barriers). The matrix permeability field is derived from the SPE10 benchmark dataset ("SPE10Bottom"), characterized by strong spatial correlations and channelized high-permeability streaks [162]. The same fracture network from Test Case 3 (composed of 30 fractures) is superimposed on this field. As shown in Figure 7.16, white lines denote highly conductive fractures with $k_f = 8.0 \times 10^{-10}$ m^2, while black lines represent low-permeability barriers with $k_f = 8.0 \times 10^{-20}$ m^2. This setup reflects a more geologically realistic subsurface environment where both matrix and fracture heterogeneity influence flow behavior.

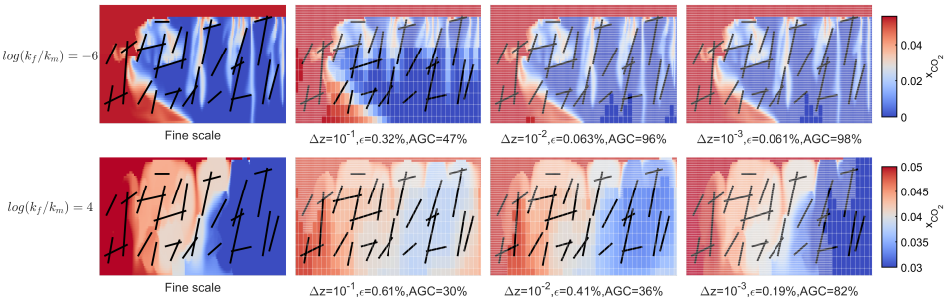


Figure 7.13: CO_2 mass fraction profiles at 32000 days for Test Case 3. Fine-scale results are compared with pEDFM-ADM solutions at thresholds $\{10^{-1}, 10^{-2}, 10^{-3}\}$. Overlaid grid lines show local coarsening levels.

Figures 7.17 and 7.18 show CO_2 mass fraction fields at early time (600 days) and late time (20000 days), respectively, comparing fine-scale reference solutions with pEDFM-ADM results under different refinement thresholds. At 600 days, the plume evolution is primarily governed by the heterogeneous matrix permeability field, with migration strongly aligned along high-permeability pathways and hindered by low-permeability zones. Conductive fractures serve to locally enhance transport where they intersect permeable matrix channels, whereas barrier-type fractures alter flow paths and create compartmentalization. The adaptive pEDFM-ADM approach successfully reproduces these early-stage features: even at the coarsest setting ($\Delta z_{CO_2} = 10^{-1}$), the error remains as low as $\epsilon = 0.10\%$ while activating only 46% of fine grid cells.

By 20000 days, the CO_2 plume exhibits a substantially more complex structure, shaped by long-range interactions between the underlying matrix heterogeneity and fracture connectivity. The presence of conductive fractures facilitates deep plume penetration and sustains extensive fracture-controlled mixing. At the coarsest refinement level ($\Delta zCO2 = 10^{-1}$), pEDFM-ADM captures large-scale plume distribution but fails to resolve finer structural details, with a error of 0.40% and only 26% of the fine grid cells active. Tighter thresholds ($\Delta zCO_2 = 10^{-2}$ and 10^{-3}) recover progressively more fine-scale features, with errors of 0.10% and 0.02% and activate 81% and 99% of the grid, respectively. While this demonstrates the framework's ability to focus resolution in physically significant regions such as fracture intersections and sharp fronts, the trade-off between accuracy and efficiency becomes apparent: the $\Delta z_{CO_2} = 10^{-3}$ case achieves near fine-scale accuracy but offers minimal computational savings due to the high active grid cell count.

As seen in Figure 7.19, both pressure and CO_2 mass fraction errors are reported alongside the temporal evolution of the active grid cell fraction, providing a quantitative assessment of solution accuracy and grid activation for this test case. Errors consistently decrease with tighter refinement thresholds, while temporal variations reflect gradual accumulation as the plume interacts with increasingly complex regions

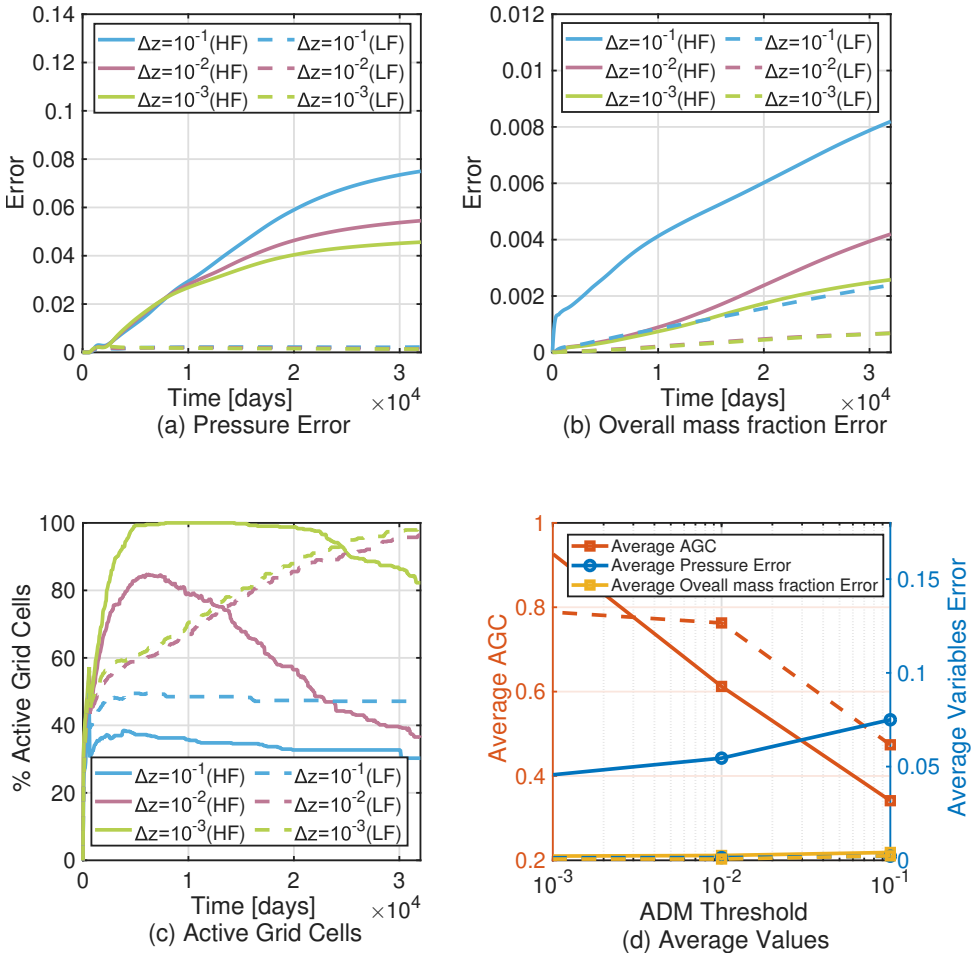


Figure 7.14: Quantitative error analysis of pEDFM-ADM solutions for Test Case 3. Pressure and overall CO_2 mass fraction errors are shown alongside the percentage of active grid cells for each coarsening threshold. (HF: high-permeability fractures; LF: low-permeability fractures)

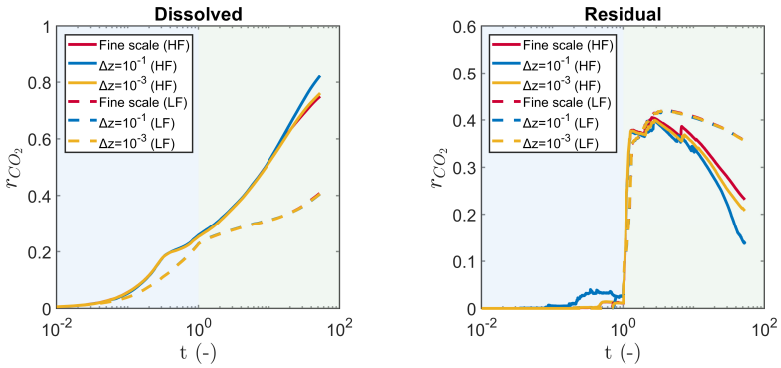


Figure 7.15: Temporal evolution of CO_2 trapping mechanisms in Test Case 3. Results are expressed as fractions of the injected CO_2 retained in immobile and dissolved states over dimensionless time. The light-blue shaded region denotes the injection phase, while the light-green shaded region corresponds to the post-injection phase.

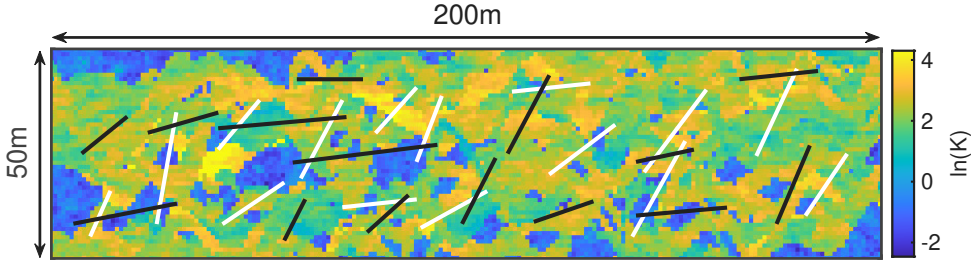


Figure 7.16: Log-permeability field $\ln(K)$ of the matrix and fracture configuration for Test Case 4. White segments represent highly conductive fractures, while black segments denote low-permeability barriers.

of the domain. The averaged metrics in Figure 7.19(d) further confirm the accuracy-efficiency trade-off: loose thresholds greatly reduce active fine cell usage but lose fine-scale detail, whereas the tightest threshold achieves near fine-scale accuracy with minimal computational savings because the active grid fraction remains close to 100% for most of the simulation.

Finally, Figure 7.20 illustrates the temporal evolution of CO_2 storage mechanisms, including dissolved and residual trapping fractions. Across all thresholds, including the coarsest setting, the pEDFM-ADM framework accurately reproduces the evolution of global storage metrics and maintains close agreement with the fine-scale reference. These results confirm the method's robustness and predictive capability for simulating multiphase, multicomponent flow in complex geological domains containing both matrix heterogeneity and mixed fracture types, reinforcing its potential for practical applications in geological CO_2 sequestration.

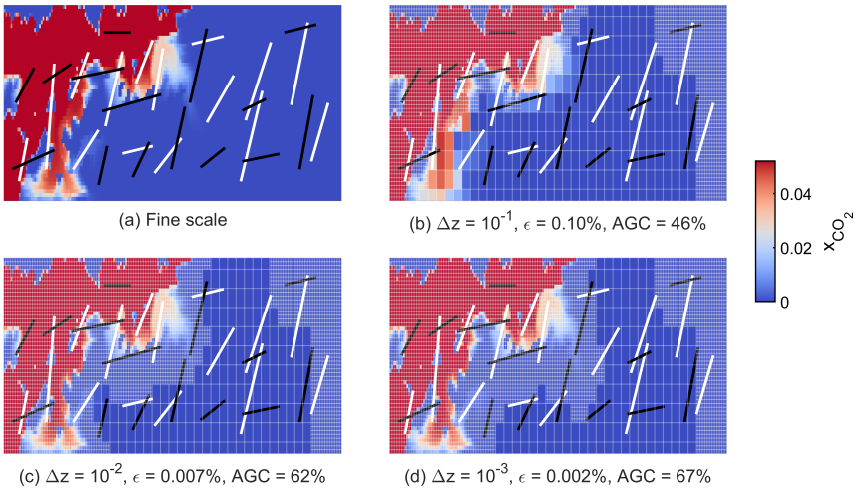


Figure 7.17: CO_2 mass fraction profiles after 600 days for Test Case 4. For the pEDFM-ADM simulations, the dynamically refined grid is overlaid as white lines.

7.3. CONCLUSIONS

This chapter introduced a multiscale simulation framework for modeling multiphase, multicomponent CO_2 flow in fractured saline aquifers. The proposed method couples the projection-based embedded discrete fracture model (pEDFM) with the algebraic dynamic multilevel (ADM) approach, enabling fully implicit simulations of fracture-matrix systems with dynamic resolution control. The fine-scale compositional flow equations are projected onto independently constructed multilevel grids

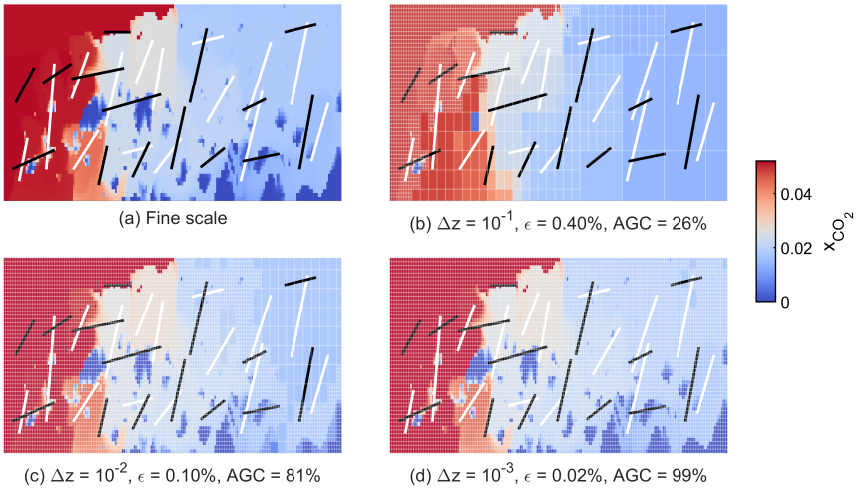


Figure 7.18: CO_2 mass fraction profiles after 20000 days for Test Case 4. For the pEDFM-ADM simulations, the dynamically refined grid is overlaid as white lines.

for the matrix and fracture domains, using localized multiscale basis functions derived from algebraic prolongation operators. These basis functions are assembled following coarse-node selection on both the rock matrix and the fracture sub-domains, allowing the framework to retain fine-scale fidelity in regions with strong gradients while coarsening elsewhere to enhance computational efficiency.

The adaptive mesh strategy is driven by front-tracking criteria based on composition gradients, allowing the model to focus resolution dynamically at physically significant locations, such as CO_2 plume fronts and fracture intersections, while preserving coarse representations in regions of limited flow activity. The resulting multiscale system maintains the coupling between phases and components and remains thermodynamically consistent throughout the simulation.

The method was validated across a series of test cases with increasing complexity, including systems with low-permeability flow barriers, highly conductive fractures. The proposed approach demonstrates its capability to capture multiphase multicomponent flow phenomena features, including fracture-guided transport, barrier-induced compartmentalization, dissolution-driven convection, and residual trapping. In scenarios dominated by low-permeability fractures, the method successfully resolves flow deflection and localized fingering. In contrast, under highly conductive fracture conditions, the model captures preferential transport and large-scale convective mixing. Importantly, the method preserves key storage metrics across all tolerances, including dissolved and residual CO_2 fractions, thereby confirming its reliability in capturing

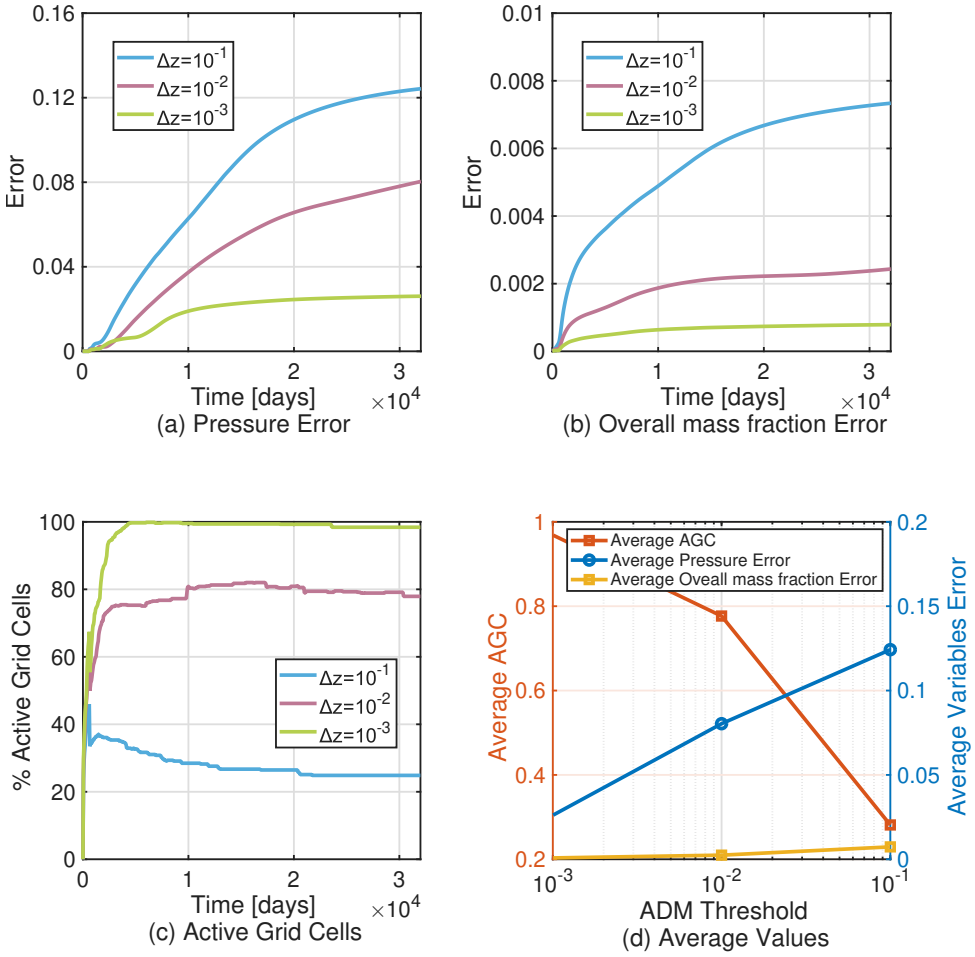


Figure 7.19: Error analysis of pEDFM-ADM solutions for Test Case 4. Pressure and overall CO_2 mass fraction errors are shown alongside the percentage of active grid cells for each coarsening threshold.

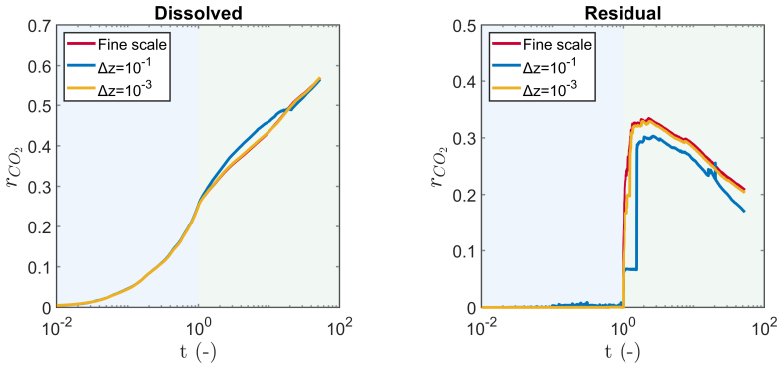


Figure 7.20: Temporal evolution of CO_2 trapping mechanisms in Test Case 4. Results are expressed as fractions of the injected CO_2 retained in immobile and dissolved states over dimensionless time. The light-blue shaded region denotes the injection phase, while the light-green shaded region corresponds to the post-injection phase.

long-term trapping behavior. These results highlight the potential of projection-based multiscale embedded fracture models as a scalable and predictive tool for assessing CO_2 migration and storage performance in geologically realistic formations.

8

CONCLUSIONS AND FUTURE WORK

This dissertation investigated modeling and simulation strategies for multiphase, multicomponent CO_2-brine flow in deep saline aquifers, with a particular focus on both data-driven surrogates and algebraic multiscale numerical methods. This work addressed challenges in CCS, including the need for physical fidelity, computational efficiency, and the capability to handle complex geological heterogeneity and fracture networks. This chapter summarizes the key scientific contributions and insights developed throughout the research and outlines potential directions for future work.

8.1. CONCLUSIONS

The first part of this dissertation established the scientific motivation, theoretical foundations, and a numerical baseline for CO_2 storage modeling. Chapter 1 situated this work within the broader context of climate mitigation and subsurface carbon management, outlining the critical role of geological storage, the challenges associated with multiphase and multicomponent flow in porous media, and the need for predictive and computationally efficient simulation strategies. It further reviewed the limitations in existing surrogate modeling and numerical simulation techniques, motivating the development pursued in later chapters.

Chapter 2 formulated a compositional multiphase flow that captures for key physical mechanisms, including capillary pressure, dissolution, and phase partitioning. The thermodynamic framework accounted for fugacity-based equilibrium across phases and compositional effects on fluid properties. Chapter 3 complemented this by presenting a fine-scale numerical implementation based on finite-volume discretization, fully implicit time stepping, and Newton-Raphson nonlinear solvers. Flash calculations were embedded to resolve phase behavior under varying thermodynamic conditions, forming a robust simulation reference against which subsequent surrogate modeling and multiscale developments are evaluated.

Chapter 4 introduced a multi-task deep learning surrogate that jointly predicted gas saturation (S_g) and solution CO_2–brine ratio (R_s) through a shared encoder-decoder

structure, exploiting their inherent physical coupling. The resulting model improved accuracy and temporal coherence relative to single-task surrogates while delivering substantial speedups, indicating that architectural consistency translated into more stable spatiotemporal predictions. However, the method remained constrained by the temporal scope and distribution of its training data, motivating the integration of physical constraints.

To move beyond purely data-driven generalization, Chapter 5 proposed a physics-constrained neural network framework (CO_2PCNet) for full-cycle CO_2 migration modeling. The method incorporated the governing equations directly into the training loss via residual-based penalties, guiding the learning process toward physically consistent solutions. The network architecture combined spatial feature encoding, temporal sequence modeling via ConvLSTM, and auto-regressive inference to enable long-term prediction beyond the temporal extent of the training data. By coupling sequence modeling with explicit physical constraints, the approach reduced unphysical drift and improved robustness when forecasting long-term plume evolution, where small local errors could otherwise accumulate into qualitatively incorrect behavior.

Chapter 6 shifted the emphasis to high-fidelity numerical simulation at scale. An Algebraic Dynamic Multilevel (ADM) method was developed to simulate compositional CO_2-brine flow with a hierarchy of multilevel grids. The method combined a fully implicit solution strategy with front-tracking technique guided by overall CO_2 mass fraction gradients. Localized multiscale basis functions were constructed to represent fine-scale heterogeneities within each coarse grid level. The ADM method dynamically refined areas with significant overall CO_2 mass fraction gradients while coarsening smooth regions, thus optimizing computational resources without compromising the accuracy required to capture essential flow and transport characteristics. This dynamic grid adjustment was facilitated by algebraic prolongation and restriction operators, which map the fine-scale system onto a coarser grid suited to the evolving distribution of the CO_2 plume. This feature allowed the ADM to navigate various coarsening thresholds efficiently, striking a trade-off between computational cost and resolution of key trapping and transport processes. Even at relatively higher thresholds, key trapping mechanisms were captured with sufficient detail for quantification. Therefore, the ADM framework preserved essential flow and mixing dynamics with substantially reduced computational expense, making long-term, field-scale simulation more tractable in strongly heterogeneous settings.

Building on this, Chapter 7 extended the ADM framework to fractured formations through the integration of a projection-based Embedded Discrete Fracture Model (pEDFM). At each simulation time step, the fully implicit fine-scale system was mapped onto a dynamically constructed multilevel grid. The dynamic grid hierarchy was determined through a front-tracking criterion based on local variations in the overall CO_2 mass fraction, allowing the model to selectively refine regions of interest while maintaining coarse resolution elsewhere. Prolongation operators, built from localized multiscale basis functions, facilitated the projection between grid levels. These basis functions were defined following the selection of coarse nodes in both matrix and fracture domains, allowing for explicit representation of fractures at that coarse scale.

Through a series of progressively complex test cases, the proposed approach demonstrated its capability to capture multiphase multicomponent flow features, including fracture-guided transport, barrier-induced compartmentalization, dissolution-driven convection, and residual trapping. It resolved both deflection and localized fingering in low-permeability fracture regimes and preferential transport with large-scale convective mixing when fractures were highly conductive, supporting its applicability to realistic fractured aquifers.

Overall, this dissertation advanced a set of complementary modeling strategies for CO_2 storage in deep saline aquifers: from physics-constrained machine learning surrogates for rapid prediction, to multiscale numerical solvers capable of resolving fine-scale dynamics in geologically complex systems. The key findings can be summarized as follows:

- Data-driven surrogates could be made more reliable for CO_2 migration forecasting with task-consistent architectures and physics-based constraints, yielding fast approximations that better preserved coupled flow behavior and mitigated unphysical extrapolation.
- Algebraic dynamic multilevel method enabled efficient and accurate long-term simulation of compositional CO_2–brine systems, including fractured media via pEDFM integration, by dynamically concentrating resolution near evolving plume features while maintaining coarse representations elsewhere.

8.2. RECOMMENDATIONS FOR FUTURE WORK

There exist further scope for research and the remaining of this chapter discuss some possible topics for future research activities. These directions can potentially enhance the devised methods and shed light on their performance and practicality.

EXTENSION TO UNSTRUCTURED GRIDS

The current implementations of both the numerical simulation and surrogate modeling frameworks are primarily based on structured grid discretizations, which may limit their applicability in complex geological domains such as folded formations, faulted structures, or irregular boundaries. Extending the fine-scale and multiscale methods to support unstructured grids, such as those based on tetrahedral or polyhedral elements, would allow more flexible and geologically realistic domain representations.

TRANSITION FROM CPU-BASED TO GPU-ACCELERATED IMPLEMENTATIONS

The ADM method has so far only been implemented in MATLAB. Future work should focus on its reimplementation in a compiled language (e.g., C++), which would enable a comprehensive evaluation of computational performance. Such a framework would

also facilitate the development of parallel and GPU-accelerated strategies, allowing for scalability studies and enhanced efficiency in large-scale simulations.

EXTENSION FROM TWO-COMPONENT TO THREE-COMPONENT SYSTEMS

This dissertation focused on binary CO_2-H_2O mixtures, which adequately represent many storage scenarios but exclude potentially important components such as dissolved salts (e.g., NaCl), impurities (e.g., H_2S, CH_4), or tracers used for monitoring. Incorporating a third component introduces additional complexity in both the thermodynamic modeling (e.g., extended equations of state, multi-component phase equilibrium) and the numerical solution (e.g., increased nonlinearity and coupling).

REFERENCES

[1] F. M. Orr Jr. “Onshore geologic storage of CO_2”. In: *Science* 325.5948 (2009), pp. 1656–1658.

[2] J. C. Pires, F. G. Martins, M. C. Alvim-Ferraz, and M. Simões. “Recent developments on carbon capture and storage: An overview”. In: *Chemical engineering research and design* 89.9 (2011), pp. 1446–1460.

[3] M. E. Boot-Handford, J. C. Abanades, E. J. Anthony, M. J. Blunt, S. Brandani, N. Mac Dowell, J. R. Fernández, M.-C. Ferrari, R. Gross, J. P. Hallett, *et al.* “Carbon capture and storage update”. In: *Energy & Environmental Science* 7.1 (2014), pp. 130–189.

[4] L. Selma, O. Seigo, S. Dohle, and M. Siegrist. “Public perception of carbon capture and storage (CCS): A review”. In: *Renewable and Sustainable Energy Reviews* 38 (2014), pp. 848–863.

[5] S. Krevor, H. De Coninck, S. E. Gasda, N. S. Ghaleigh, V. de Gooyert, H. Hajibeygi, R. Juanes, J. Neufeld, J. J. Roberts, and F. Swennenhuis. “Subsurface carbon dioxide and hydrogen storage for a sustainable energy future”. In: *Nature Reviews Earth & Environment* 4.2 (2023), pp. 102–118.

[6] NASA Global Climate. *Carbon Dioxide Concentration | NASA Global Climate Change*. 2025.

[7] Global Carbon Budget. *Annual CO_2 Emissions by World Region*. 2024.

[8] M. A. Celia, S. Bachu, J. M. Nordbotten, and K. W. Bandilla. “Status of CO_2 storage in deep saline aquifers with emphasis on modeling approaches and practical simulations”. In: *Water Resources Research* 51.9 (2015), pp. 6846–6892.

[9] E. Lindeberg and D. Wessel-Berg. “Vertical convection in an aquifer column under a gas cap of CO_2”. In: *Energy Conversion and management* 38 (1997), S229–S234.

[10] J. Ennis-King and L. Paterson. “Role of convective mixing in the long-term storage of carbon dioxide in deep saline formations”. In: *SPE Annual Technical Conference and Exhibition*. SPE. 2003, SPE–84344.

[11] K. Michael, A. Golab, V. Shulakova, J. Ennis-King, G. Allinson, S. Sharma, and T. Aiken. “Geological storage of CO_2 in saline aquifers—A review of the experience from existing storage operations”. In: *International journal of greenhouse gas control* 4.4 (2010), pp. 659–667.

[12] M. Zhao, Y. Wang, M. Gerritsma, and H. Hajibeygi. “A physics-constraint neural network for CO_2 storage in deep saline aquifers during injection and post-injection periods”. In: *Advances in Water Resources* 193 (2024), p. 104837.

[13] B. Metz, O. Davidson, H. De Coninck, M. Loos, and L. Meyer. *IPCC special report on carbon dioxide capture and storage*. Cambridge: Cambridge University Press, 2005.

[14] Y. Wang, C. Vuik, and H. Hajibeygi. “Analysis of hydrodynamic trapping interactions during full-cycle injection and migration of CO_2 in deep saline aquifers”. In: *Advances in Water Resources* 159 (2022), p. 104073.

[15] A. Mazzoldi, T. Hill, and J. J. Colls. “Assessing the risk for CO_2 transportation within CCS projects, CFD modelling”. In: *International Journal of Greenhouse Gas Control* 5.4 (2011), pp. 816–825.

[16] R. J. Pawar, G. S. Bromhal, J. W. Carey, W. Foxall, A. Korre, P. S. Ringrose, O. Tucker, M. N. Watson, and J. A. White. “Recent advances in risk assessment and risk management of geologic CO_2 storage”. In: *International Journal of Greenhouse Gas Control* 40 (2015), pp. 292–311.

[17] L. Nghiem, V. Shrivastava, B. Kohse, M. Hassam, and C. Yang. “Simulation and optimization of trapping processes for CO_2 storage in saline aquifers”. In: *Journal of Canadian Petroleum Technology* 49.08 (2010), pp. 15–22.

[18] S. T. Ide, K. Jessen, and F. M. Orr Jr. “Storage of CO_2 in saline aquifers: Effects of gravity, viscous, and capillary forces on amount and timing of trapping”. In: *International journal of greenhouse gas control* 1.4 (2007), pp. 481–491.

[19] M. Elenius, D. Voskov, and H. Tchelepi. “Interactions between gravity currents and convective dissolution”. In: *Advances in Water Resources* 83 (2015), pp. 77–88.

[20] R. Iskhakov. “High-resolution numerical simulation of CO_2 sequestration in saline aquifers”. PhD thesis. Stanford University, 2014.

[21] X. Jiang. “A review of physical modelling and numerical simulation of long-term geological storage of CO_2”. In: *Applied energy* 88.11 (2011), pp. 3557–3566.

[22] B. Flemisch, J. M. Nordbotten, M. Fernø, R. Juanes, J. W. Both, H. Class, M. Delshad, F. Doster, J. Ennis-King, J. Franc, *et al.* “The FluidFlower validation benchmark study for the storage of CO_2”. In: *Transport in Porous Media* 151.5 (2024), pp. 865–912.

[23] S. Flude and J. Alcade. “Carbon Capture and Storage Has Stalled Needlessly–Three Reasons Why Fears of CO_2 Leakage are Overblown”. In: *The Conversation* (2020).

[24] N. Gholizadeh Doonechaly, R. R. Abdel Azim, and S. S. Rahman. “A study of permeability changes due to cold fluid circulation in fractured geothermal reservoirs”. In: *Groundwater* 54.3 (2016), pp. 325–335.

[25] Z. Ziabakhsh-Ganji and H. Kooi. “An Equation of State for thermodynamic equilibrium of gas mixtures and brines to allow simulation of the effects of impurities in subsurface CO_2 storage”. In: *International Journal of Greenhouse Gas Control* 11 (2012), S21–S34.

[26] T. Praditia, R. Helmig, and H. Hajibeygi. “Multiscale formulation for coupled flow-heat equations arising from single-phase flow in fractured geothermal reservoirs”. In: *Computational Geosciences* 22 (2018), pp. 1305–1322.

[27] M. Liu and D. Grana. “Petrophysical characterization of deep saline aquifers for CO_2 storage using ensemble smoother and deep convolutional autoencoder”. In: *Advances in Water Resources* 142 (2020), p. 103634.

[28] M. Tang, X. Ju, and L. J. Durlofsky. “Deep-learning-based coupled flow-geomechanics surrogate model for CO_2 sequestration”. In: *International Journal of Greenhouse Gas Control* 118 (2022), p. 103692.

[29] Z. Guo and A. C. Reynolds. “Robust life-cycle production optimization with a support-vector-regression proxy”. In: *Spe Journal* 23.06 (2018), pp. 2409–2427.

[30] M. Zhao, K. Zhang, G. Chen, X. Zhao, J. Yao, C. Yao, L. Zhang, and Y. Yang. “A classification-based surrogate-assisted multiobjective evolutionary algorithm for production optimization under geological uncertainty”. In: *SPE Journal* 25.05 (2020), pp. 2450–2469.

[31] S. Kamrava, M. Sahimi, and P. Tahmasebi. “Simulating fluid flow in complex porous materials by integrating the governing equations with deep-layered machines”. In: *Npj Computational Materials* 7.1 (2021), p. 127.

[32] A. Y. Sun. “Optimal carbon storage reservoir management through deep reinforcement learning”. In: *Applied Energy* 278 (2020), p. 115660.

[33] O. A. Omosebi, C. M. Oldenburg, and M. Reagan. “Development of lean, efficient, and fast physics-framed deep-learning-based proxy models for subsurface carbon storage”. In: *International Journal of Greenhouse Gas Control* 114 (2022), p. 103562.

[34] X. Ju, F. P. Hamon, G. Wen, R. Kanfar, M. Araya-Polo, and H. A. Tchelepi. “Learning CO_2 plume migration in faulted reservoirs with Graph Neural Networks”. In: *Computers & Geosciences* 193 (2024), p. 105711.

[35] H. Hamdi, I. Couckuyt, M. C. Sousa, and T. Dhaene. “Gaussian Processes for history-matching: application to an unconventional gas reservoir”. In: *Computational Geosciences* 21 (2017), pp. 267–287.

[36] M. Babaei and I. Pan. “Performance comparison of several response surface surrogate models and ensemble methods for water injection optimization under uncertainty”. In: *Computers & Geosciences* 91 (2016), pp. 19–32.

[37] G. Chen, K. Zhang, L. Zhang, X. Xue, D. Ji, C. Yao, J. Yao, and Y. Yang. “Global and local surrogate-model-assisted differential evolution for waterflooding production optimization”. In: *SPE Journal* 25.01 (2020), pp. 105–118.

[38] V. Christelis, R. G. Regis, and A. Mantoglou. “Surrogate-based pumping optimization of coastal aquifers under limited computational budgets”. In: *Journal of Hydroinformatics* 20.1 (2018), pp. 164–176.

[39] G. Chen, K. Zhang, X. Xue, L. Zhang, J. Yao, H. Sun, L. Fan, and Y. Yang. “Surrogate-assisted evolutionary algorithm with dimensionality reduction method for water flooding production optimization”. In: *Journal of Petroleum Science and Engineering* 185 (2020), p. 106633.

[40] M. Zhao, K. Zhang, G. Chen, X. Zhao, C. Yao, H. Sun, Z. Huang, and J. Yao. “A surrogate-assisted multi-objective evolutionary algorithm with dimension-reduction for production optimization”. In: *Journal of Petroleum Science and Engineering* 192 (2020), p. 107192.

[41] S. Ren, K. He, R. Girshick, and J. Sun. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015).

[42] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.

[43] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444.

[44] S. Jiang and L. J. Durlofsky. “History matching for geological carbon storage using data-space inversion with spatio-temporal data parameterization”. In: *International Journal of Greenhouse Gas Control* 134 (2024), p. 104124.

[45] Z. Feng, Z. Tariq, X. Shen, B. Yan, X. Tang, and F. Zhang. “An encoder-decoder ConvLSTM surrogate model for simulating geological CO_2 sequestration with dynamic well controls”. In: *Gas Science and Engineering* 125 (2024), p. 205314.

[46] N. Wang, H. Chang, and D. Zhang. “Efficient uncertainty quantification and data assimilation via theory-guided convolutional neural network”. In: *SPE Journal* 26.06 (2021), pp. 4128–4156.

[47] M. Zhao, Y. Wang, M. Gerritsma, and H. Hajibeygi. “Efficient simulation of CO_2 migration dynamics in deep saline aquifers using a multi-task deep learning technique with consistency”. In: *Advances in Water Resources* 178 (2023), p. 104494.

[48] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*. Vol. 1. 2. MIT press Cambridge, 2016.

[49] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).

[50] K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[51] Y. Zhu and N. Zabaras. “Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification”. In: *Journal of Computational Physics* 366 (2018), pp. 415–447.

[52] S. Mo, Y. Zhu, N. Zabaras, X. Shi, and J. Wu. “Deep convolutional encoder-decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media”. In: *Water Resources Research* 55.1 (2019), pp. 703–728.

[53] S. Mo, N. Zabaras, X. Shi, and J. Wu. “Deep autoregressive neural networks for high-dimensional inverse problems in groundwater contaminant source identification”. In: *Water Resources Research* 55.5 (2019), pp. 3856–3881.

[54] M. Tang, Y. Liu, and L. J. Durlofsky. “A deep-learning-based surrogate model for data assimilation in dynamic subsurface flow problems”. In: *Journal of Computational Physics* 413 (2020), p. 109456.

[55] T. Kadeethum, D. O’Malley, J. N. Fuhg, Y. Choi, J. Lee, H. S. Viswanathan, and N. Bouklas. “A framework for data-driven solution and parameter estimation of pdes using conditional generative adversarial networks”. In: *Nature Computational Science* 1.12 (2021), pp. 819–829.

[56] G. Wen, C. Hay, and S. M. Benson. “CCSNet: a deep learning modeling suite for CO_2 storage”. In: *Advances in Water Resources* 155 (2021), p. 104009.

[57] G. Wen, M. Tang, and S. M. Benson. “Towards a predictor for CO_2 plume migration using deep neural networks”. In: *International Journal of Greenhouse Gas Control* 105 (2021), p. 103223.

[58] Z. Zhong, A. Y. Sun, and H. Jeong. “Predicting CO_2 plume migration in heterogeneous formations using conditional deep convolutional generative adversarial network”. In: *Water Resources Research* 55.7 (2019), pp. 5830–5851.

[59] B. Yan, B. Chen, D. R. Harp, W. Jia, and R. J. Pawar. “A robust deep learning workflow to predict multiphase flow behavior during geological CO_2 sequestration injection and Post-Injection periods”. In: *Journal of Hydrology* 607 (2022), p. 127542.

[60] A. K. Chu, S. M. Benson, and G. Wen. “Deep-learning-based flow prediction for CO_2 storage in shale–sandstone formations”. In: *Energies* 16.1 (2022), p. 246.

[61] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. “Physics-informed machine learning”. In: *Nature Reviews Physics* 3.6 (2021), pp. 422–440.

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

[63] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris. “Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data”. In: *Journal of Computational Physics* 394 (2019), pp. 56–81.

[64] N. Geneva and N. Zabaras. “Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks”. In: *Journal of Computational Physics* 403 (2020), p. 109056.

[65] P. Ren, C. Rao, Y. Liu, J.-X. Wang, and H. Sun. “PhyCRNet: Physics-informed convolutional-recurrent network for solving spatiotemporal PDEs”. In: *Computer Methods in Applied Mechanics and Engineering* 389 (2022), p. 114399.

[66] S. Goswami, M. Yin, Y. Yu, and G. E. Karniadakis. “A physics-informed variational DeepONet for predicting crack path in quasi-brittle materials”. In: *Computer Methods in Applied Mechanics and Engineering* 391 (2022), p. 114587.

[67] J. M. Nordbotten, B. Flemisch, S. Gasda, H. Nilsen, Y. Fan, G. E. Pickup, B. Wiese, M. A. Celia, H. Dahle, G. Eigestad, *et al.* “Uncertainties in practical simulation of CO_2 storage”. In: *International Journal of Greenhouse Gas Control* 9 (2012), pp. 234–242.

[68] A. J. Yamaguchi, T. Sato, T. Tobase, X. Wei, L. Huang, J. Zhang, J. Bian, and T.-Y. Liu. “Multiscale numerical simulation of CO_2 hydrate storage using machine learning”. In: *Fuel* 334 (2023), p. 126678.

[69] H. Versteeg and W. Malalasekera. “Computational fluid dynamics: the finite volume method”. In: *Harlow, England: Longman Scientific & Technical* (1995).

[70] J. N. Reddy. “An introduction to the finite element method”. In: *New York* 27.14 (1993).

[71] J. Kim, H. A. Tchelepi, and R. Juanes. “Stability and convergence of sequential methods for coupled flow and geomechanics: Fixed-stress and fixed-strain splits”. In: *Computer Methods in Applied Mechanics and Engineering* 200.13-16 (2011), pp. 1591–1606.

[72] J. Zhu and J. Ma. “An improved gray lattice Boltzmann model for simulating fluid flow in multi-scale porous media”. In: *Advances in Water Resources* 56 (2013), pp. 61–76.

[73] Y. Wang, T. H. Chakrapani, Z. Wen, and H. Hajibeygi. “Pore-scale simulation of H_2-brine system relevant for underground hydrogen storage: A lattice Boltzmann investigation”. In: *Advances in Water Resources* (2024), p. 104756.

[74] Y. Wang, C. Vuik, and H. Hajibeygi. “CO_2 Storage in deep saline aquifers: impacts of fractures on hydrodynamic trapping”. In: *International Journal of Greenhouse Gas Control* 113 (2022), p. 103552.

[75] K. Kala and D. Voskov. “Element balance formulation in reactive compositional flow and transport with parameterization technique”. In: *Computational Geosciences* 24.2 (2020), pp. 609–624.

[76] E. Ahusborde, B. Amaziane, and M. I. Moulay. “High performance computing of 3D reactive multiphase flow in porous media: Application to geological storage of CO_2”. In: *Computational Geosciences* 25.6 (2021), pp. 2131–2147.

[77] X. Lyu and D. Voskov. “Advanced modeling of enhanced CO_2 dissolution trapping in saline aquifers”. In: *International Journal of Greenhouse Gas Control* 127 (2023), p. 103907.

[78] T. Y. Hou and X.-H. Wu. “A multiscale finite element method for elliptic problems in composite materials and porous media”. In: *Journal of computational physics* 134.1 (1997), pp. 169–189.

[79] Y. Efendiev, T. Hou, and T. Strinopoulos. “Multiscale simulations of porous media flows in flow-based coordinate system”. In: *Computational Geosciences* 12 (2008), pp. 257–272.

[80] P. Jenny, S. Lee, and H. A. Tchelepi. “Multi-scale finite-volume method for elliptic problems in subsurface flow simulation”. In: *Journal of computational physics* 187.1 (2003), pp. 47–67.

[81] P. Jenny, S. H. Lee, and H. A. Tchelepi. “Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media”. In: *Journal of Computational Physics* 217.2 (2006), pp. 627–641.

[82] S. Lee, C. Wolfsteiner, and H. Tchelepi. “Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity”. In: *Computational Geosciences* 12 (2008), pp. 351–366.

[83] I. Lunati and S. H. Lee. “An operator formulation of the multiscale finite-volume method with correction function”. In: *Multiscale modeling & simulation* 8.1 (2009), pp. 96–109.

[84] O. Møyner and K.-A. Lie. “A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids”. In: *Journal of Computational Physics* 304 (2016), pp. 46–71.

[85] H. Hajibeygi, G. Bonfigli, M. A. Hesse, and P. Jenny. “Iterative multiscale finite-volume method”. In: *Journal of Computational Physics* 227.19 (2008), pp. 8604–8621.

[86] B. Faigle, R. Helmig, I. Aavatsmark, and B. Flemisch. “Efficient multiphysics modelling with adaptive grid refinement using a MPFA method”. In: *Computational Geosciences* 18 (2014), pp. 625–636.

[87] M. Cusini, C. van Kruijsdijk, and H. Hajibeygi. “Algebraic dynamic multilevel (ADM) method for fully implicit simulations of multiphase flow in porous media”. In: *Journal of Computational Physics* 314 (2016), pp. 60–79.

[88] M. Cusini, B. Fryer, C. van Kruijsdijk, and H. Hajibeygi. “Algebraic dynamic multilevel method for compositional flow in heterogeneous porous media”. In: *Journal of Computational Physics* 354 (2018), pp. 593–612.

[89] M. HosseiniMehr, M. Cusini, C. Vuik, and H. Hajibeygi. “Algebraic dynamic multilevel method for embedded discrete fracture model (F-ADM)”. In: *Journal of Computational Physics* 373 (2018), pp. 324–345.

[90] M. Ţene, M. S. Al Kobaisi, and H. Hajibeygi. “Algebraic multiscale method for flow in heterogeneous porous media with embedded discrete fractures (F-AMS)”. In: *Journal of Computational Physics* 321 (2016), pp. 819–845.

[91] Schlumberger. “ECLIPSE: Technical Description 2013.2”. In: Schlumberger, 2013.

[92] K.-A. Lie. *An introduction to reservoir simulation using MATLAB/GNU Octave: User guide for the MATLAB Reservoir Simulation Toolbox (MRST)*. Cambridge University Press, 2019.

[93] S. Krogstad, K.-.-A. Lie, O. Møyner, H. M. Nilsen, X. Raynaud, and B. Skaflestad. "MRST-AD–an open-source framework for rapid prototyping and evaluation of reservoir simulation problems". In: *SPE Reservoir Simulation Conference*. Spe. 2015, D022S002R004.

[94] H. S. Viswanathan, J. Ajo-Franklin, J. T. Birkholzer, J. W. Carey, Y. Guglielmi, J. Hyman, S. Karra, L. J. Pyrak-Nolte, H. Rajaram, G. Srinivasan, *et al.* "From fluid flow to coupled processes in fractured rock: Recent advances and new frontiers". In: *Reviews of Geophysics* 60.1 (2022), e2021RG000744.

[95] Y. Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[96] M. Cusini, A. A. Lukyanov, J. Natvig, and H. Hajibeygi. "Constrained pressure residual multiscale (CPR-MS) method for fully implicit simulation of multiphase flow in porous media". In: *Journal of Computational Physics* 299 (2015), pp. 472–486.

[97] J. H. Abou-Kassem, S. M. Farouq-Ali, and M. R. Islam. *Petroleum Reservoir Simulations*. Elsevier, 2013.

[98] G. S. Pau, J. B. Bell, K. Pruess, A. S. Almgren, M. J. Lijewski, and K. Zhang. "High-resolution simulation and characterization of density-driven flow in CO_2 storage in saline aquifers". In: *Advances in Water Resources* 33.4 (2010), pp. 443–455.

[99] D. V. Voskov and H. A. Tchelepi. "Comparison of nonlinear formulations for two-phase multi-component EoS based simulation". In: *Journal of Petroleum Science and Engineering* 82 (2012), pp. 101–111.

[100] M. H. Reed. "Calculation of multicomponent chemical equilibria and reaction processes in systems involving minerals, gases and an aqueous phase". In: *Geochimica et Cosmochimica Acta* 46.4 (1982), pp. 513–528.

[101] N. Spycher, K. Pruess, and J. Ennis-King. "CO_2-H_2O mixtures in the geological sequestration of CO_2. I. Assessment and calculation of mutual solubilities from 12 to 100 °C and up to 600 bar". In: *Geochimica et cosmochimica acta* 67.16 (2003), pp. 3015–3031.

[102] J. M. Prausnitz, R. N. Lichtenthaler, and E. G. De Azevedo. *Molecular thermodynamics of fluid-phase equilibria*. Pearson Education, 1998.

[103] J. W. Johnson, E. H. Oelkers, and H. C. Helgeson. "SUPCRT92: A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous species, and reactions from 1 to 5000 bar and 0 to 1000°C". In: *Computers & Geosciences* 18.7 (1992), pp. 899–947.

[104] A. Harvey, A. Perkin, and S. Klein. "NIST Standard Reference Database 10, Version 2.2". In: *International Association for the Properties of Water and Steam (IAPWS)* 95 (2000).

[105] O. Redlich and J. N. Kwong. "On the thermodynamics of solutions. V. An equation of state. Fugacities of gaseous solutions". In: *Chemical reviews* 44.1 (1949), pp. 233–244.

[106] K. G. Denbigh. *The principles of chemical equilibrium: with applications in chemistry and chemical engineering*. Cambridge University Press, 1981.

[107] O. Massarweh and A. S. Abushaikha. "CO_2 sequestration in subsurface geological formations: A review of trapping mechanisms and monitoring techniques". In: *Earth-Science Reviews* (2024), p. 104793.

[108] H. Emami-Meybodi, H. Hassanzadeh, C. P. Green, and J. Ennis-King. "Convective dissolution of CO_2 in saline aquifers: Progress in modeling and experiments". In: *International Journal of Greenhouse Gas Control* 40 (2015), pp. 238–266.

[109] F. Al Hameli, H. Belhaj, and M. Al Dhuhoori. "CO_2 sequestration overview in geological formations: Trapping mechanisms matrix assessment". In: *Energies* 15.20 (2022), p. 7805.

[110] J. Luo, Y. Xie, M. Z. Hou, Y. Xiong, X. Wu, C. T. Lüddeke, and L. Huang. "Advances in subsea carbon dioxide utilization and storage". In: *Energy reviews* 2.1 (2023), p. 100016.

[111] K. Rasmusson, M. Rasmusson, Y. Tsang, S. Benson, F. Hingerl, F. Fagerlund, and A. Niemi. "Residual trapping of carbon dioxide during geological storage-Insight gained through a pore-network modeling approach". In: *International Journal of Greenhouse Gas Control* 74 (2018), pp. 62–78.

[112] S. Krevor, M. J. Blunt, S. M. Benson, C. H. Pentland, C. Reynolds, A. Al-Menhali, and B. Niu. "Capillary trapping for geologic carbon dioxide storage–From pore scale physics to field scale implications". In: *International Journal of Greenhouse Gas Control* 40 (2015), pp. 221–237.

[113] R. Juanes, E. Spiteri, F. Orr Jr, and M. Blunt. "Impact of relative permeability hysteresis on geological CO_2 storage". In: *Water resources research* 42.12 (2006).

[114] W. Gunter, E. Perkins, and I. Hutcheon. "Aquifer disposal of acid gases: modelling of water–rock reactions for trapping of acid wastes". In: *Applied geochemistry* 15.8 (2000), pp. 1085–1095.

[115] S. Bachu, W. Gunter, and E. Perkins. "Aquifer disposal of CO_2: hydrodynamic and mineral trapping". In: *Energy Conversion and management* 35.4 (1994), pp. 269–279.

[116] H. Cao. *Development of techniques for general purpose simulators*. Stanford University, 2002.

[117] L. Li and S. H. Lee. "Efficient field-scale simulation of black oil in a naturally fractured reservoir through discrete fracture networks and homogenized media". In: *SPE Reservoir evaluation & engineering* 11.04 (2008), pp. 750–758.

[118] H. Hajibeygi, D. Karvounis, and P. Jenny. "A hierarchical fracture model for the iterative multiscale finite volume method". In: *Journal of Computational Physics* 230.24 (2011), pp. 8729–8743.

[119] A. Danesh. *PVT and phase behaviour of petroleum reservoir fluids*. Vol. 47. Elsevier, 1998.

[120] H. Hassanzadeh, M. Pooladi-Darvish, A. M. Elsharkawy, D. W. Keith, and Y. Leonenko. “Predicting PVT data for CO_2–brine mixtures for black-oil simulation of CO_2 geological storage”. In: *international journal of greenhouse gas control* 2.1 (2008), pp. 65–77.

[121] H. Hajibeygi and H. A. Tchelepi. “Compositional multiscale finite-volume formulation”. In: *SPE Journal* 19.02 (2014), pp. 316–326.

[122] J. E. Garcia. “Density of aqueous solutions of CO_2”. In: *Technical Report. LBNL* (2001).

[123] K. Coats, W. George, C. Chu, and B. Marcum. “Three-dimensional simulation of steamflooding”. In: *Society of Petroleum Engineers Journal* 14.06 (1974), pp. 573–592.

[124] O. Møyner and K.-A. Lie. “A multiscale restriction-smoothed basis method for compressible black-oil models”. In: *SPE Journal* 21.06 (2016), pp. 2079–2096.

[125] O. Møyner and H. Tchelepi. “A multiscale restriction-smoothed basis method for compositional models”. In: *SPE reservoir simulation conference*. SPE. 2017, D031S009R003.

[126] A. Moncorgé, H. A. Tchelepi, and P. Jenny. “Modified sequential fully implicit scheme for compositional flow simulation”. In: *Journal of Computational Physics* 337 (2017), pp. 98–115.

[127] A. Moncorgé, H. A. Tchelepi, and P. Jenny. “Sequential fully implicit formulation for compositional simulation using natural variables”. In: *Journal of Computational Physics* 371 (2018), pp. 690–711.

[128] R. Younis. “Advances in modern computational methods for nonlinear problems; a generic efficient automatic differentiation framework, and nonlinear solvers that converge all the time”. PhD thesis. Stanford University USA, 2009.

[129] M. Cusini. “Dynamic Multilevel Methods for Simulation of Multiphase Flow in Heterogeneous Porous Media”. PhD thesis. Delft University of Technology, 2019.

[130] S. M. HosseiniMehr. “Scalable simulation models for fractured porous media with complex geometries”. PhD thesis. Delft University of Technology, 2021.

[131] S. Albawi, T. A. Mohammed, and S. Al-Zawi. “Understanding of a convolutional neural network 2017 International Conference on Engineering and Technology (ICET)”. In: *New York. IEEE* (2017), pp. 1–6.

[132] V. Badrinarayanan, A. Kendall, and R. Cipolla. “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[133] O. Ronneberger, P. Fischer, and T. Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[134] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[135] K. He, X. Zhang, S. Ren, and J. Sun. “Identity mappings in deep residual networks”. In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. Springer. 2016, pp. 630–645.

[136] D. Maturana and S. Scherer. “Voxnet: A 3d convolutional neural network for real-time object recognition”. In: *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE. 2015, pp. 922–928.

[137] D. Brüggemann, M. Kanakis, A. Obukhov, S. Georgoulis, and L. Van Gool. “Exploring relational context for multi-task dense prediction”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 15869–15878.

[138] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. “Cross-stitch networks for multi-task learning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 3994–4003.

[139] S. Vandenhende, S. Georgoulis, W. Van Gansbeke, M. Proesmans, D. Dai, and L. Van Gool. “Multi-task learning for dense prediction tasks: A survey”. In: *IEEE transactions on pattern analysis and machine intelligence* 44.7 (2021), pp. 3614–3633.

[140] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and J. Yang. “Pattern-affinitive propagation across depth, surface normal and semantic segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 4106–4115.

[141] A. R. Zamir, A. Sax, N. Cheerla, R. Suri, Z. Cao, J. Malik, and L. J. Guibas. “Robust learning through cross-task consistency”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11197–11206.

[142] N. Remy, A. Boucher, and J. Wu. *Applied geostatistics with SGeMS: A user's guide*. Cambridge University Press, 2009.

[143] R. Hecht-Nielsen. “Theory of the backpropagation neural network”. In: *Neural networks for perception*. Elsevier, 1992, pp. 65–93.

[144] D. P. Kingma and J. Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[145] Z. Zhang, Y. Wang, C. Vuik, and H. Hajibeygi. “An efficient simulation approach for long-term assessment of CO_2 storage in complex geological formations”. In: *SPE Reservoir Characterisation and Simulation Conference and Exhibition*. SPE. 2023, D022S008R007.

[146] C. Ruprecht, R. Pini, R. Falta, S. Benson, and L. Murdoch. “Hysteretic trapping and relative permeability of CO2 in sandstone at reservoir conditions”. In: *International Journal of Greenhouse Gas Control* 27 (2014), pp. 15–27.

[147] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. “Convolutional LSTM network: A machine learning approach for precipitation nowcasting”. In: *Advances in neural information processing systems* 28 (2015).

[148] S. Hochreiter and J. Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[149] P. Ramachandran, B. Zoph, and Q. V. Le. “Searching for activation functions”. In: *arXiv preprint arXiv:1710.05941* (2017).

[150] J. M. Nordbotten and M. A. Celia. “Similarity solutions for fluid injection into confined aquifers”. In: *Journal of Fluid Mechanics* 561 (2006), pp. 307–327.

[151] Z. Zhang, X. Yan, P. Liu, K. Zhang, R. Han, and S. Wang. “A physics-informed convolutional neural network for the simulation and prediction of two-phase Darcy flows in heterogeneous porous media”. In: *Journal of Computational Physics* 477 (2023), p. 111919.

[152] M. Zhao, M. Gerritsma, M. Al Kobaisi, and H. Hajibeygi. “Algebraic dynamic multilevel (ADM) method for CO_2 storage in heterogeneous saline aquifers”. In: *Journal of Computational Physics* 539 (2025), p. 114202.

[153] Y. Wang, H. Hajibeygi, and H. A. Tchelepi. “Algebraic multiscale solver for flow in heterogeneous porous media”. In: *Journal of Computational Physics* 259 (2014), pp. 284–303.

[154] M. T. Elenius, J. M. Nordbotten, and H. Kalisch. “Convective mixing influenced by the capillary transition zone”. In: *Computational Geosciences* 18 (2014), pp. 417–431.

[155] J. M. Nordbotten, M. Fernø, B. Flemisch, A. R. Kovscek, and K.-A. Lie. “The 11th Society of Petroleum Engineers Comparative Solution Project: Problem Definition”. In: *SPE Journal* (2024).

[156] J. Nordbotten, M. Fernø, B. Flemisch, R. Juanes, and M. Jørgensen. “Final benchmark description: FluidFlower international benchmark study”. In: *Zenodo* (2022).

[157] E. W. Lemmon. “Thermophysical properties of fluid systems”. In: *NIST chemistry WebBook* (2010).

[158] H. Abdoulghafour, M. Sarmadivaleh, L. P. Hauge, M. Fernø, and S. Iglauer. “Capillary pressure characteristics of CO_2-brine-sandstone systems”. In: *International Journal of Greenhouse Gas Control* 94 (2020), p. 102876.

[159] M. J. Oak, L. E. Baker, and D. Thomas. “Three-phase relative permeability of Berea sandstone”. In: *Journal of petroleum technology* 42.08 (1990), pp. 1054–1061.

[160] M. T. Van Genuchten. “A closed-form equation for predicting the hydraulic conductivity of unsaturated soils”. In: *Soil science society of America journal* 44.5 (1980), pp. 892–898.

[161] J. R. Fanchi. *Principles of applied reservoir simulation*. Elsevier, 2005.

[162] M. A. Christie and M. J. Blunt. “Tenth SPE comparative solution project: A comparison of upscaling techniques”. In: *SPE Reservoir Evaluation & Engineering* 4.04 (2001), pp. 308–317.

CURRICULUM VITÆ

Mengjie ZHAO

11-04-1996 Born in Henan, China.

EDUCATION

2014–2018 **China University of Petroleum (East China), Qingdao, China**
Bachelor of Petroleum Engineering

2018–2021 **China University of Petroleum (East China), Qingdao, China**
Master of Oil and Gas Field Development Engineering

2021–2025 **Delft University of Technology, Delft, the Netherlands**
Ph.D. in Flow Physics and Technology
Promotor: Dr. ir. M.I. Gerritsma & Prof. dr. H. Hajibeygi

LIST OF PUBLICATIONS

JOURNAL PAPERS

4. **M. Zhao**, R. Haagenson, M. Gerritsma, H. Hajibeygi, Multiscale projection-based embedded discrete fracture modeling approach for CO_2 storage in deep saline aquifers, Advances in Water Resources, **207**, (2026) 105200.

3. **M. Zhao**, M. Gerritsma, M. Al Kobaisi, H. Hajibeygi, Algebraic dynamic multilevel (ADM) method for CO_2 storage in heterogeneous saline aquifers, Journal of Computational Physics, **539**, (2025) 114202.

2. **M. Zhao**, Y. Wang, M. Gerritsma, H. Hajibeygi, A physics-constraint neural network for CO_2 storage in deep saline aquifers during injection and post-injection periods, Advances in Water Resources, **193**, (2024) 104837.

1. **M. Zhao**, Y. Wang, M. Gerritsma, H. Hajibeygi, Efficient simulation of CO_2 migration dynamics in deep saline aquifers using a multi-task deep learning technique with consistency, Advances in Water Resources, **178**, (2023) 104494.

ORAL PRESENTATIONS AND EVENTS

3. A neural network model with physics constraints for simulating CO_2 storage in deep saline aquifers during and after injection. Talk at InterPore conference in Qingdao, China. 13 - 16 May 2024.

2. A robust and efficient deep-learning-based surrogate model for CO_2 storage in deep saline aquifers. Talk at InterPore conference in Edinburgh, Scotland. 22 - 25 May 2023.

1. Fast Modeling CO_2 Migration Dynamics System with Deep Learning Methods. Poster presentation at the Woudschoten Conference in Zeist, The Netherlands. 5 - 7 October 2022.

ACKNOWLEDGEMENTS

I am deeply grateful for the invaluable support, guidance, and encouragement I have received from so many people throughout this remarkable four-year journey. Reflecting on my time at TU Delft, I feel incredibly fortunate for the people I have met, and the experiences I have gained. Writing this section gives me a rare opportunity to pause and express my sincere gratitude to all who made this journey possible.

First and foremost, I would like to thank my promotors, **Dr. Marc Gerritsma** and **Prof. Hadi Hajibeygi**, for giving me the opportunity to come to the Netherlands and pursue my PhD. Thank you for believing in my potential from the very beginning and for offering me this unique chance. Your passion for research and teaching, as well as the commitment to your work, have been a constant source of motivation and inspiration. Dear Hadi, your energy, optimism, and drive have continuously pushed me forward. You helped me believe in my capabilities, and your encouragement has meant more than I can say. Dear Marc, your insightful guidance and dedication have shaped every step of this work. The time you spent discussing ideas with me, helping me find answers, and improving the quality of my work is beyond measure. What an incredible journey we have shared. I truly cannot thank you enough for your patience and all the support you have given me.

My gratitude goes also to **Prof. Cornelis (Kees) Vuik**, **Prof. Samuel Krevor**, **Prof. Andreas Busch**, **Prof. Sebastian Geiger**, and **Dr. Yuhang Wang** for being part of my PhD committee, and **Dr. Denis Voskov** as the reserve member. I also would like to thank **Dr. Matthias Möller** and **Dr. Anh Khoa Doan** for being in the committee of my Go/No-Go assessment in the first year.

To all members of the ADMIRE group, thank you for creating such a welcoming and inspiring work environment. A special thank you to **Dr. Yuhang Wang**, thank you for selecting me from among many candidates and giving me this opportunity. Your support, both in research and daily life, has been instrumental. We've shared many valuable discussions, and your advice on academic and personal, has always been thoughtful and encouraging. **Willemijn**, thank you for going through most of this journey together. We started our PhDs around the same time, and your generous help made many things much easier. I also appreciate your contributions to the group, and I will never forget the joyful time we had at the InterPore conference in Qingdao. **Dr. Fanxiang Xu**, thank you for inviting me to be your paranymph, and for always looking out for me like an older brother. **Dr. Maartje Boon**, thank you for giving me the opportunity to chair a session at InterPore. It was an important step for me and helped boost my confidence. **Ryan**, **Artur**, and **Amin**, it was such a

pleasure working with you in the final year. Thank you for your kindness, patience, and all the enjoyable discussions we shared. **Herminio**, **Kishan**, **Leila**, **Thejas**, the InterPore conference in Edinburgh was of a much special experience for me. **Naomi**, **Mohammad Saeed**, **Mathias**, **Sijmen**, **Filipe**, **Milad**, **Tim**, **Ziliang**, **Sara**, thank you for all the great moments we shared in our group meetings, the Denmark summer school, and other events. It has been a privilege to grow alongside you all.

I would also like to thank my colleagues in the Department of Flow Physics and Technology. Special thanks to our department secretary, **Colette**, for always being responsive and helpful. **Ata**, **Wencan** and **Guanqun**, thank you for walking this research journey together. **Varun**, **Yi**, and **Xiaodong**, I appreciate your early support when I was just beginning my PhD. To **Renzhi**, **Ruiying**, **Yifu**, **Aneek**, **Tyler**, **Thomas**, **Jane**, **Haris**, **Andrea**, **Babak**, **Max**, **Sajad**, **Suyash**, **Kherlen**, **Sara**, **Luuk**, **Adrian**, **Joel**, **Marco**, **Kevin**, and **Brian**, thank you for sharing the office. Even though my research interests are quite different and we did not have many chances to discuss research, you were consistently kind and welcoming. I am especially grateful to **Kherlen**, since our home countries are close, it was always a pleasure to share conversations about familiar places. I would also like to thank **Renzhi** for helping me with remote access whenever I worked from home. Special thanks to **Tyler** for organizing the Thanksgiving dinner every year.

Over these years, I have also been incredibly fortunate to have a close group of friends who supported me in every possible way. **Guofeng**, **Lifei**, **Kai**, **Fanxiang**, **Shihao**, **Bing**, **Wencan**, **Jianxin**, **Ziliang**, and **Jing**, thank you for all the time we spent together, all the activities and laughter we shared. Thank you for always being there for me, especially during the difficult times. Your encouragement, advice, and companionship meant so much. I will forever cherish our carefree nights playing Avalon game those were some of my happiest moments. **Shihao** and **Wencan**, we started this journey together when we came to the Netherlands, and I am truly grateful to have shared this experience with you both. To all my friends, I hope we will continue to stay in touch, even if life takes us to different places. I look forward to the day when we can all look back together and cheer to the wonderful memories we have made.

Hang, my boyfriend, your love has been my greatest source of strength. I feel incredibly lucky to have met you and to have started a new chapter of my life with you. All the adventures we shared, our trips in the Netherlands, and to the United Kingdom, Denmark and especially Norway, were all truly unforgettable. Your love motivates me every day, and I am excited for the future and for wherever life takes us next. Thank you for your patience and support throughout my PhD journey.

Finally, I would like to express my deepest gratitude to my family. To my parents, my sister, and our beloved dogs, thank you for your unconditional love and support. **Mom** and **Dad**, thank you for always giving me the strength to overcome every challenge and for supporting my decisions, no matter how far they took me from home.

Mengjie Zhao
Delft, July 2025


TU Delft