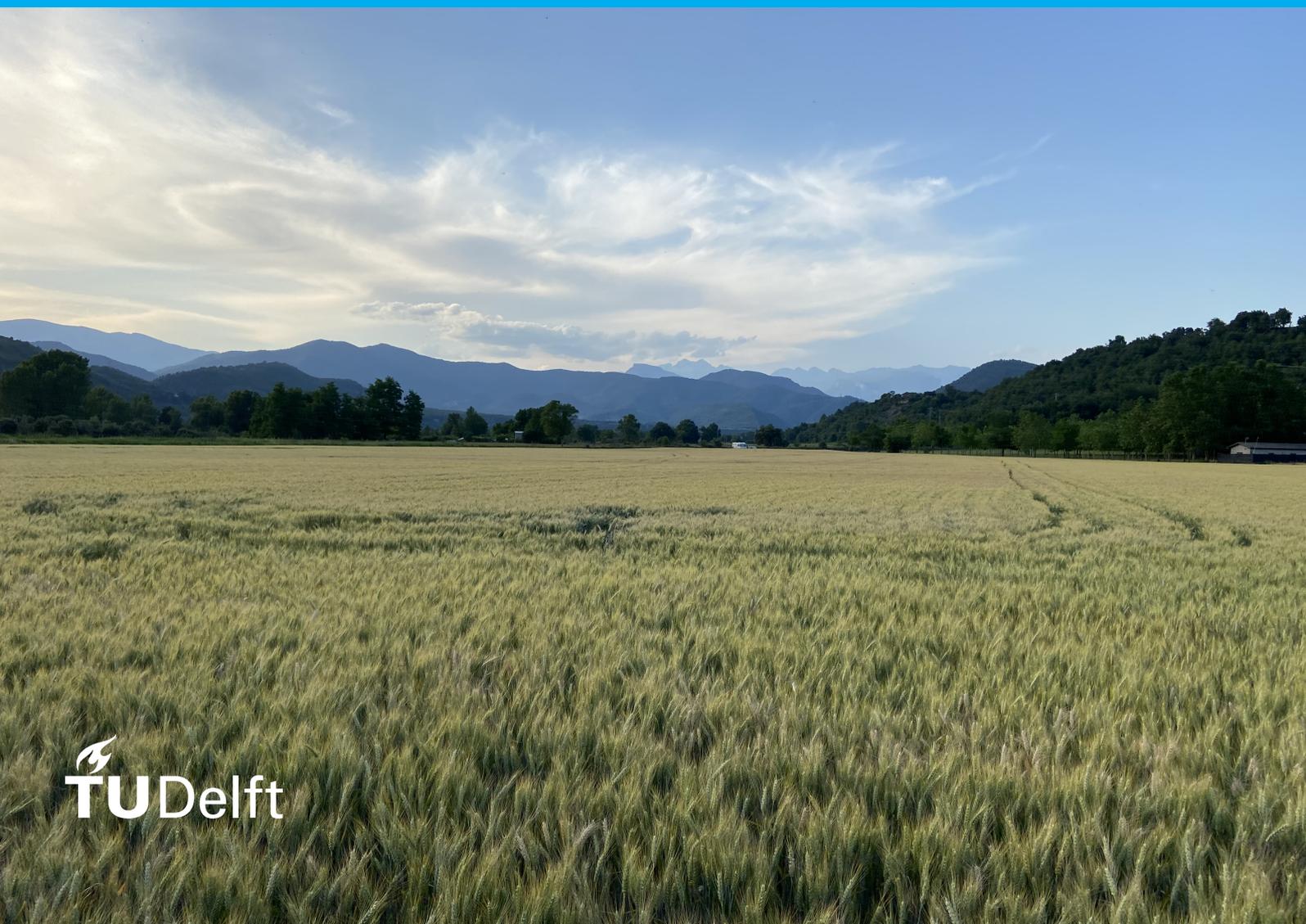


Application of Deep Neural Networks to the Operator Space of Nonlinear PDE for Physics-based Proxy Modeling

George Hadjisotiriou



Application of Deep Neural Networks to the Operator Space of Nonlinear PDE for Physics-based Proxy Modeling

by

George Hadjisotiriou

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday December 15, 2022 at 10:30 AM.

Student number:	4549953
Project duration:	November 8, 2022 – December 15, 2022
Thesis committee:	Dr. D.V. Voskov, TU Delft, chair
	Dr. G. Rongier TU Delft
	K. Mansour Pour TU Delft
	Dr. J. Groenenboom, Shell
	Dr. A. Fadili, Shell

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Preface

I am happy to submit for your consideration this thesis and project report in fulfillment of the Masters in Geo-Energy Engineering at the Delft University of Technology.

Readers particularly interested in improving proxy modeling performance with machine learning methods are referred to chapter 4.1. Those most interested in compositional upscaling are referred to chapter 4.2.

I would like to express my gratitude to Dr. Denis Voskov of TU Delft for his guidance and unwavering support I received at all stages of this project. I am also thankful to my co-supervisor Kiarash Mansour Pour for the insightful conversations regarding physics-informed machine learning.

I was fortunate to complete an internship at the gas flooding team of Shell. My immediate supervisors there, Dr. Jeroen Groenenboom and Dr. Ali Fadili made me feel welcome and gave direction to my research. I am thankful.

I thank my family. You are my sunshine.

George Hadjisotiriou
Delft, December 2022

Summary

Compositional simulation is computationally intensive for high-fidelity models due to thermodynamic equilibrium relations and the coupling of flow, transport and mass transfer. In this report, two methods for accelerated compositional simulation are outlined and demonstrated for a gas vaporization problem. The first method uses a proxy model that reduces the number of components and the second method reduces the number of grid blocks (i.e. upscaling). Both methods are implemented within the operator-based linearization framework of the Delft Advanced Research Terra Simulator.

Lebesgue integration is applied in the loss function of a neural network allowing the neural network to discover the operator space of the reference model in reduced dimensions. Training is carried out for a one-dimensional homogeneous reservoir and minimizes the misfit of the leading and trailing shocks of a compressible pseudo-binary model with respect to observations of the reference model. The operator space of the pseudo-binary model is initially approximated with the method of multiscale reconstruction of physics, a numerical representation of the method of characteristics. Training is carried out in a two-stage transfer learning scheme to increase computational efficiency. In the first stage, neural networks are trained to approximate the analytical reconstruction. In the second stage, a solver is embedded in the loss function of the neural network and the forward solution is used to calculate the Lebesgue integral. The transfer training scheme minimizes the misfit of the leading and trailing shocks for 10 discrete time steps in a one-dimensional homogeneous reservoir. The misfit of the trained model shows a significant improvement in the location of the trailing shock and a modest improvement in the estimation of the leading shock. The trained proxy is applied to the top and bottom 15 layers of the SPE10 model and the estimation of the first and last breakthrough is assessed in conjunction with the error of the phase-state classification. The phase-state classification is significantly improved through time which is also expressed in improvements of the estimation of breakthrough times. The average difference in breakthrough time for the trained and untrained models with respect to the reference model is *293 days* versus *570 days* for the trailing shocks and *15 days* versus *16 days* for the leading shock. The established training framework enables the development of proxies with increased complexity and can be extended to learn directly from field measurements.

Rigorous upscaling defines the upscaled operator space with dynamic and non-equilibrium thermodynamic upscaling functions. These functions combined, define the upscaled operator space for the three-dimensional compositional space (as opposed to single-point upstream weighting) and are inferred from data points gathered from a limited, characteristic portion of the full-size model. Gathered data points are interpreted with an interpolation function or neural networks to construct structured OBL meshes for implementation within DARTS. This upscaled operator space can effectively be used for different boundary conditions without reevaluating the upscaling functions. The method is tested for layered models with and without gravity and/or vertical permeability and consistently outperforms the coarse model where only the absolute permeability is upscaled.

Contents

1	Introduction	1
2	Literature	2
2.1	Accelerated compositional modeling	2
2.1.1	Proxy modeling	2
2.1.2	Multiphase compositional upscaling	2
2.1.3	Thermodynamic equilibrium calculations	3
2.2	Deep learning	3
2.2.1	Deep neural networks	3
2.2.2	Physics-informed machine learning (PIML)	4
3	Methods	6
3.1	Simulation framework	6
3.1.1	Compositional simulation	6
3.1.2	Thermodynamic model	7
3.1.3	Operator-based linearization (OBL)	7
3.2	Pseudo-binary proxy model	8
3.2.1	Multiscale Compositional Transport (MSCT)	8
3.2.2	Multiscale reconstruction in physics for compressible fluids	8
3.2.3	Four-component gas vaporization problem	9
3.2.4	Lebesgue regularisation	11
3.3	Compositional upscaling	12
3.3.1	Conventional upscaling	13
3.3.2	Rigorous upscaling	14
4	Results	16
4.1	Training of pseudo-binary proxy model	16
4.1.1	1D-training	16
4.1.2	Horizontal layers of SPE10	18
4.2	Rigorous upscaling of layered models	20
4.2.1	Four-layer model	21
4.2.2	Upscaling procedure for n_z -layered models	22
4.2.3	Added physics and vertical permeability	22
4.2.4	Neural approximation of operator space	25
5	Conclusion	32
5.1	Discussion/recomendations	32
A	Appendix	36
A.1	Pseudo-binary model	36
A.1.1	Hydrodynamic parameters of pseudo-binary model	36
A.2	Rigorous upscaling	36
A.2.1	Sensitivity to number of layers	36
A.2.2	Histograms of training and test data	37

1

Introduction

Accurate reservoir simulation of gas injection problems for hydrocarbon production and gas storage or sequestration is complex and computationally intensive due to non-linear thermodynamic equilibrium relations and the coupling of flow, transport and mass transfer. This complex nonlinear dynamic often requires a very fine resolution of the computational grid. However, high-fidelity reservoir models contain a large number of grid blocks thus requiring even more computational resources. Developed methods to manage computational resources attempt to reduce the size of the problem by reducing the number of components (i.e. lumping, pseudo-models, etc.) or the number of blocks (i.e. upscaling) while still conserving key features of the full compositional or high fidelity solution. In this report, two methods for accelerated compositional upscaling are utilized and applied to a gas-injection problem within the compositional framework of operator-based linearization. In the first method, the number of components is reduced and in the second the number of grid blocks is reduced. These proxies are useful for optimization or history-matching problems where a large number of runs are required and simulation with the conservative high-fidelity model is prohibitively expensive.

In the first method, a data-driven approach to proxy modeling is proposed and applied to a gas injection problem. Neural networks are embedded in the operator space of a compressible pseudo-binary model and a solver is included in the loss function to improve the model's prediction with respect to the locations of the shocks in space and time. The method trains the operator space with a one-dimensional homogeneous model and improves the estimation of the proxy with regard to the breakthrough time of the injected gas. The effectiveness of the trained model is further validated for two-dimensional heterogeneous cases of the SPE10 model.

In the second method, compositional upscaling of two-dimensional layered models is carried out with a limited portion of the fine-scale model. A known drawback of compositional upscaling, where the global solution of the fine-scale model is used, is that the upscaling functions used in the coarse-scale model must be reevaluated whenever boundary conditions change. This is counterproductive when the coarse model is used for optimization problems where controls or boundary conditions are changed to find an optimal solution. In this work, a rigorous upscaling method is introduced that parameterizes the entire upscaled operator space for different injection conditions. Furthermore, a deep neural network is used to learn the upscaled operator space as a function of layer permeabilities.

The structure of this report is as follows. Chapter 2 contains a short overview of methods for accelerated compositional modeling and further introduces deep machine learning methods including physics-informed machine learning. In chapter 3, the compositional simulation framework is outlined and in the same chapter, the applied methods for proxy modeling and upscaling are explained. In chapter 4, the results of the proxy model training and rigorous upscaling are reported.

2

Literature

This chapter provides an overview of select methods for accelerated compositional modeling and explains the deep learning method of physics-informed machine learning.

2.1. Accelerated compositional modeling

In this section, different methods for improving the performance of compositional simulation are described. Particular attention is paid to methods focusing on reducing the number of components or blocks and methods for accelerated phase behavior calculations.

2.1.1. Proxy modeling

Proxy modeling with a reduced number of components increases the efficiency of compositional simulation by reducing the number of equations that need to be solved. In the lumping technique, the problem is reformulated by presenting the formulation with a limited number of equivalent components and delumping the components at the production stream (Jessen and Moghadam, 2009; Rastegar and Jessen, 2009). Further techniques rely on a numerical representation of the method of characteristics. Tang and Zick, 1993 propose a limited compositional reservoir simulator that solves a four-component problem by solving an equivalent pseudo-ternary problem. Ganapathy and Voskov, 2018 propose a physics-based method for proxy modeling, termed multi-scale reconstruction in physics. Their method uses a pseudo-binary proxy model to solve any transport problem with an arbitrary number of components. Their model is applied to four and eight-component systems with either the equation of state or constant K-value thermodynamics and accurately locates the major features of the compositional displacement profile (Chen and Voskov, 2020).

2.1.2. Multiphase compositional upscaling

Upscaling techniques coarsen the spatial discretization of the reservoir, reducing the number of blocks and cost in exchange for a controlled reduction of accuracy. This is frequently achieved via a global or local analysis of the fine-scale solution where boundary conditions are applied globally or locally and the behavior of the fine-scale model is characterized by defining upscaled quantities in the coarse model. These are the single-phase parameters (absolute permeability, porosity) and the two-phase parameters (relative permeabilities, mobilities).

In the case of multiphase models, with high mobility ratios and large upscaling ratios, upscaling of only single-phase parameters is insufficient. Established methods, therefore, use transport coefficients and/or dynamic pseudo-functions in the coarse-scale conservation equation to capture the dynamic behavior of the fine-scale model in the coarsened model. Dynamic pseudo functions are upscaling functions that represent the behavior of dynamic quantities, such as the relative permeability in the function of the average of the upstream coarse grid block's quantity (saturation or composition). Methods for upscaling two-phase functions generally can be categorized into Kyte and Berry methods and the total mobility upscaling methods (Christie et al., 1995; Kyte and Berry, 1975; Stone, 1991). See Darman et al., 2002 and Barker and Thibeau, 1997 for reviews on (pseudo-)dynamic upscaling methods. Transport coefficients are also functions tabulated with respect to the upstream block but work differently as they are included in the flux term and scale the flux across the coarse grid block's interface (Barker and Fayers, 1994; Christie and Clifford, 1998).

A drawback of upscaling functions is that they need to be re-evaluated for new boundary conditions (Barker and Thibeau, 1997; Durlafsky, 1998). Additionally, it is impractical to evaluate these functions at each coarse grid block interface, so these functions are typically defined for different characteristic rock types (Barker and Thibeau, 1997). Additionally, in the case of compositional upscaling, thermodynamic equilibrium does not hold on the coarse scale (Iranshahr et al., 2014). Therefore, Iranshahr et al., 2014 introduce a methodology where thermodynamic equations are upscaled in conjunction with the single-phase and two-phase parameters (mobility). Their method utilizes non-equilibrium thermodynamic functions to account for differences in phase behavior between fine and coarse scales and achieves better results than upscaling only single- and multiphase parameters. These upscaled thermodynamic functions can be incorporated into the coarse model by including non-equilibrium transport coefficients or adjusted K-values (Salehi et al., 2013; Salehi et al., 2019).

2.1.3. Thermodynamic equilibrium calculations

Further developments in the space of compositional simulation focus on accelerating phase-state or thermodynamic equilibrium calculations. Compositional Space Parameterization (CSP) and the following Compositional Space Adaptive Tabulation (CSAT) techniques parameterize the entire or part of the tie-line space (Voskov and Tchelepi, 2009; Voskov and Tchelepi, 2008). CSP and CSAT are further extended to achieve an equation-of-state-free general-purpose compositional simulation where the governing differential equations are projected on the tie-simplex space (Zaydullin et al., 2013). Additional techniques that reduce the cost of phase behavior computation include the reduced variable method and the shadow region method (Pan and Tchelepi, 2011; Rasmussen et al., 2003).

As a logical extension of parameterization approaches, the Operator-Based Linearization (OBL) method has been introduced in Voskov, 2017. In the OBL approach, the governing equations of the physical problem are linearized and represented as a combination of space- and state-dependent functions. In addition, a uniform mesh is introduced into the physical space of the problem where operators are interpolated. This limited, linear approximation of the physical space provides better nonlinear convergence with controlled error (Lyu et al., 2021; Wang et al., 2020).

2.2. Deep learning

In this section, the physics-informed machine learning method for machine learning is described. First, a general description of the working mechanisms of deep neural networks is given. And secondly, a review of relevant literature on the applications of physics-informed machine learning is given.

2.2.1. Deep neural networks

Deep learning methods are used to directly evaluate how to approximate the output from the input data. The advantage of deep learning versus traditional machine learning is that it does not require feature extraction from the input and instead learns which features are relevant from training data. Deep neural networks are used for various tasks such as regression, classification, transcription and natural language processing. For the task of regression, the neural network is tasked to predict the real value y given a vector of data x containing n number of features. Feedforward deep neural networks, also known as multilayer perceptions, consist of a series of simple mathematical functions that map any input feature x to output y . They consist of n_l number of fully connected layers,

$$f_{\theta}(\mathbf{X}) = z_{n_l}(z_{n_l-1}(\dots(z_2(z_1(\mathbf{X}))))), \quad (2.1)$$

where each hidden layer consists of a stack of artificial neurons which process input feature matrix \mathbf{X} as the weighted sum of weights \mathbf{W}_i and biases \mathbf{b}_i before passing through activation function σ

$$z_i(\mathbf{X}) = \sigma(\mathbf{W}_i\mathbf{X} + \mathbf{b}_i) \quad i = 1, \dots, n_l. \quad (2.2)$$

Consequently function f is parameterized according to the ensemble of weights and biases θ ,

$$\theta = \{W_1, W_2, \dots, W_{n_l}, b_1, b_2, \dots, b_{n_l}\}. \quad (2.3)$$

Training of neural networks is carried out iteratively with the gradient descent optimization algorithm. This method uses two steps commonly referred to as the forward and backward pass. In the forward pass, the loss is calculated, and in the back pass, the weights are updated with the gradient of the loss with respect to the parameters of the neural network,

$$W := W - \alpha \nabla L_W, \quad (2.4)$$

where α is the learning rate. This process is repeated and the loss is minimized, thus training the neural network to approximate y accurately. Modern neural networks use the backpropagation algorithm to compute the gradient. In backpropagation, the neural network is viewed as a mathematical graph. In the forward pass, intermediate values are stored and used in the chain rule to compute the gradient for gradient descent.

When training the neural network, it is important to watch for overfitting or underfitting of the training data. To ensure that the network can perform its intended task well. The quality of the function estimator is expressed by its bias and variance, which need to be balanced. If overfitting occurs, the neural network will not ‘generalize’ well. Consequently, it will poorly estimate y for previously unseen values of x . In the case of overfitting, the bias is small, but the variance is large. Conversely, underfitting occurs if the bias is large, but the variance is low. In the case of underfitting, the neural network cannot fit the data well. Overfitting of data can be handled by increasing the amount of data or adding a regularization term to the loss function. The regularization term expresses a preference for the model weights and helps prevent overfitting by decreasing the neural network’s capacity. The network’s capacity is its ability to fit a wide range of functions (Goodfellow et al., 2016).

2.2.2. Physics-informed machine learning (PIML)

Recently, the application of machine learning techniques to solve linear or nonlinear partial differential equations (PDEs) has attracted interest for their ability to solve PDEs without any labeled data of the solution. These machine learning techniques, such as constrained learning and physics-informed machine learning (PIML), utilize artificial neural networks to solve complex equations by encoding the known physics of the problem into the loss function of the neural network as prior-knowledge (Raissi et al., 2019). For PDEs, this is achieved by including the residual form of the equation as a regularization constant in the loss function in addition to a regressive term that uses initial and boundary conditions. PIML is applied to a nonlinear PDE of the form:

$$r := u_t + N[u, \lambda], \quad (2.5)$$

where u_t is a scalar quantity and $N[u, \lambda]$ the nonlinear operator parameterized by λ . The residual form of the equation

$$r_\theta = \frac{\partial u_\theta}{\partial t} + N(u_\theta, \lambda), \quad (2.6)$$

is included in the loss function in addition to a regular regressive loss term. In the first term, the mean squared error (MSE) of a randomly selected set of boundary/initial points (N_u) is computed, and in the second term, the MSE of the residual equation is computed for a random set of collocation points (N_r)

$$L(\theta) = L_u(\theta) + L_f(\theta) = \frac{1}{N_u} \sum_{i=1}^{N_u} |u_\theta^i - u_{bc}^i|^2 + \frac{1}{N_r} \sum_{i=0}^{N_r} |r_\theta^i|^2. \quad (2.7)$$

In the loss functions, the regularization term effectively constrains the hypothesis space of the neural network, thus allowing the physics-informed neural network (PINN) to learn the solution without any labeled data on the interior of the solution domain (Raissi et al., 2019; Stewart and Ermon, 2016).

Raissi et al., 2019 and Fuks and Tchelepi, 2020 demonstrate the application of PIML for nonlinear PDEs. Raissi et al., 2019 use PIML to accurately solve the Burgers equation, a quasi-linear PDE, and find that with a small number of initial and boundary points that their nonlinear behavior is accurately captured. They conclude that if the PDE is well posed, PIML can solve the problem provided that the neural network is expressive and a sufficient number of collocation points are used. Fuks and Tchelepi, 2020 expressly tests the limitations of PIML by application to a hyperbolic two-phase displacement problem. They find that encoding the residual in the loss function does not produce an adequate estimation of saturation, irrespective of the number of collocation points or neural architecture. In particular, the neural network cannot resolve the shock’s location in space and time, which is a defining feature of the solution. Stewart and Ermon, 2016 apply constrained learning to convolutional neural networks to track the movement of objects without any labeled data and demonstrate the usefulness of PIML techniques for small data regimes where data is sparse. An additional advantage of PIML for solving PDEs is that automatic differentiation is used to supply derivatives for the residual and therefore, no mesh generation is required. Consequently, they are presented as an efficient alternative to traditional numerical modeling techniques such as the Galerkin, collocation, or finite volume methods for higher dimensional problems (Blechsmidt and Ernst, 2021). However, for application to hyperbolic problems where an exact solution is required, as opposed to a qualitative approximate solution, PIML remains inadequate.

Besides applying PIML to solve equations, PIML can be used to handle the inverse problem for data-driven discovery of partial differential equations where unknown parameters within the nonlinear operator of the PDE are approximated from observations of the auxiliary quantity. Raissi et al., 2018 use PIML to qualitatively estimate the pressure and accurately determine hidden parameters within Navier-Stokes and Burger's equation.

3

Methods

In this chapter, two methods for accelerated compositional simulation are introduced and implemented within the Delft Advanced Research Terra Simulator (DARTS). The working problem for which these methods are applied is a gas injection problem containing either three or four components. Our main focus is a vaporizing gas drive where gas is injected to vaporize the intermediate component from the oil for an efficient gas displacement (Orr, 2007).

Firstly, a physics-based proxy that reduces the number of components of a compressible four-component problem to two is introduced. Thereafter, a physics-informed machine learning methodology for improving the proxy model's performance is explained. Secondly, a rigorous multiphase upscaling method is introduced that applies established upscaling techniques to a characteristic pattern of the reservoir and defines the up-scaled operator space in the full compositional space. The chapter is divided into three parts, in the first part, the simulation framework and operator-based linearization are outlined. In the second part, the first method of proxy modeling is introduced and in the third part, the upscaling method is explained.

3.1. Simulation framework

In this section, we briefly describe the governing equations of the isothermal compositional problem. The problem is implemented within the operator-based linearization framework of DARTS.

3.1.1. Compositional simulation

Compositional simulation of a gas vaporization problem is carried out in Delft Advanced Research Terra Simulator (DARTS) framework. DARTS utilizes operator-based linearization (OBL) where the mass conservation equation is written in operator form and a mesh is introduced into the physical space. In this approach, operator values are computed for the mesh vertices onto which a multi-linear interpolant is applied for a continuous representation of the physics (Voskov, 2017).

The mass conservation equation (eq.3.1) for n_c components and n_p phases is written in its molar formulation for an isothermal displacement. The mass conservation equation of component i reads:

$$\frac{\partial}{\partial t} \left(\phi \sum_{j=1}^{n_p} x_{ij} \rho_j s_j \right) + \nabla \cdot \sum_{j=1}^{n_p} x_{ij} \rho_j \mathbf{u}_j + \sum_{j=1}^{n_p} x_{ij} \rho_j q_j = 0, \quad i = 1, \dots, n_c, \quad (3.1)$$

where x_{ij} is the mole fraction of component i in phase j , ρ_j is the phase molar density and s_j the phase saturation and ϕ the porosity. Furthermore, \mathbf{u}_j is the Darcy flow velocity and is a function of absolute permeability \mathbf{k} , phase relative permeability $k_{r,j}$ and phase viscosity μ_j . The ratio of relative permeability and viscosity is henceforth referred to as phase mobility λ_j .

$$\mathbf{u}_j = -\mathbf{k} \frac{k_{r,j}}{\mu_j} \nabla p. \quad (3.2)$$

Finite volume discretization is applied to the general mass conservation equation with two-point flux

approximation and backward Euler approximation in time:

$$V\phi\left(\left(\sum_{j=1}^{n_p} x_{i,j}\rho_j s_j\right)^{n+1} - \left(\sum_{j=1}^{n_p} x_{i,j}\rho_j s_j\right)^n\right) - \Delta t \sum_{l \in L} \left(\sum_{j=1}^{n_p} x_{i,j}^l \rho_j^l T_j^l \Delta p\right) + V\Delta t \sum_{j=1}^{n_p} x_{i,j}\rho_j q_j = 0, \quad i = 1, \dots, n_c, \quad (3.3)$$

where V is the control volume, ϕ the porosity and T_j^l the phase transmissibility across the grid block interface L . The phase saturation is calculated as a function of phase density and molar phase fraction v_j :

$$s_j = \frac{v_j / \rho_j}{\sum_{k=1}^{n_p} v_k / \rho_k}. \quad (3.4)$$

3.1.2. Thermodynamic model

The thermodynamic model determines the phase behavior of the compositional problem by determining the quantities and compositions of vapor and liquid at two-phase conditions (McCain, 1990). The stability of the gas-liquid mixture is ensured by the thermodynamic equilibrium condition which states that the fugacities (chemical potentials) $f_{i,j}$ of each component in both phases must be equal:

$$f_{i,o}(p, T, x_{i,o}) = f_{i,g}(p, T, x_{i,g}). \quad (3.5)$$

Further, auxiliary relations are that the sum of mole fractions,

$$\sum_{i=1}^{n_c} x_{i,j}, \quad j = 1, \dots, n_p, \quad (3.6)$$

and phase saturations are equal to one,

$$\sum_{j=1}^{n_p} s_j = 1. \quad (3.7)$$

To determine the phase behavior of the system, flash calculations are carried out with the Rachford-Rice equation and constant equilibrium ratios (K-values) are used. This assumption is valid for cases where the phase behavior is a weak function of the composition (Orr, 2007). In this approach, the phase split is determined by solving the Rachford-Rice equation for the vapor fraction v and further deriving the partitioning coefficients for liquid and vapor x_i and y_i from equation 3.9

$$h(v) = \sum_{i=1}^{n_c} \frac{z_i(K_i - 1)}{v(K_i - 1) + 1} = 0, \quad (3.8)$$

$$z_i = x_i(1 - v) + y_i v. \quad (3.9)$$

3.1.3. Operator-based linearization (OBL)

Operator-based linearization introduces a mesh in the parameter space of the problem such that the operators only need to be assessed at the vertices of the mesh. Subsequent values $\hat{\alpha}(p, z)$ and $\hat{\beta}(p, z)$ in between these points are estimated with multilinear interpolators F_α and F_β (Voskov, 2017)

$$\hat{\alpha}(p, z) = F_\alpha(p, z), \quad \hat{\beta}(p, z) = F_\beta(p, z). \quad (3.10)$$

Following the OBL approach, the discretized equations are rewritten in their algebraic, operator form. In this operator form (eq.3.11), state and spatial operators are defined. State variables (ω) are a function of physical unknowns, pressure p and the overall composition z , whereas spatial variables (ξ) are a function of spatial coordinates

$$r_i(\omega, \xi) = V(\xi)\phi_0(\xi)(\alpha_i(\omega) - \alpha_i(\omega_n)) - \Delta t \sum_l \beta_i(\omega) T^l(\xi) \Lambda(\omega) \Delta p = 0, \quad i = 1, \dots, n_c. \quad (3.11)$$

The residual equation is defined according to the total velocity formulation and contains $2 \times n_c + 1$ operators. These operators are defined according to a set of equations 3.12 where α corresponds to the accumulation operator, β the convective operator and Λ the total mobility which in turn is embedded into the flux operator:

$$\begin{aligned} \alpha_i(\omega) &= (1 + c_r(p - p_{ref})) \sum_{j=1}^{n_p} x_{i,j} \rho_j s_j, \\ \beta_i(\omega) &= \frac{\sum_{j=1}^{n_p} x_{i,j} \lambda_j \rho_j}{\Lambda}, \\ \Lambda(\omega) &= \sum_{j=1}^{n_p} \frac{k_{rj}}{\mu_j}. \end{aligned} \quad (3.12)$$

The equation is approximated using the fully implicit method for time approximation and the resulting non-linear system is solved with the Newton-Raphson method.

3.2. Pseudo-binary proxy model

In this section, we introduce the main ideas behind our physics-based proxy-modeling methodology. The method of Multiscale Reconstruction in Physics is used to develop an analytical approximation of the pseudo-operator space of a compressible gas vaporization problem and reduce the number of components from four to two. Thereafter, a physics-informed machine learning algorithm is proposed that minimizes the difference in the shock estimation in space and time between proxy and reference model.

3.2.1. Multiscale Compositional Transport (MSCT)

Ganapathy and Voskov, 2018 introduce a proxy model to accurately reconstruct the trailing and leading shock of a multicomponent problem by solving a pseudo-binary problem in an incompressible case without mass exchange. Therefore, reducing the number of equations from $n_b \times n_c$ to $n_b \times 2$. Their method utilizes a numerical expression of the established analytical method of characteristics and defines a so-called pseudo-fractional flow curve

$$F_{ri} = \text{conv}(F_i^{ini} \cup F_i^{inj}), \quad (3.13)$$

from the convex union of the fractional flow curves of the initial and injection tie-lines, F_i^{ini} and F_i^{inj} . The fractional flow is a piece-wise function where values are a linear function of the overall composition for a single-phase mixture and a nonlinear function for a two-phase mixture. For a two-phase mixture, the fractional flow curve,

$$F_i = x_i(1 - f_g) + y_i f_g, \quad (3.14)$$

is defined according to phase behavior which in turn is expressed in the partitioning coefficients x_i and y_i for oil and gas. In their work, the reconstruction is applied to an isothermal transport problem and consists of two stages. In the first stage, the pseudo-fractional flow curve is used in the restricted transport equation,

$$\frac{\partial z}{\partial t} + \frac{u_t}{\phi} \frac{\partial F_{ri}}{\partial x} = 0, \quad (n_b \times 2), \quad (3.15)$$

where the total velocity (u_t) is constant. At this stage, the location of the shocks in space and time is accurately reconstructed. Thereafter, in the second stage, the full compositional solution is reconstructed by application of the constant space parameterization technique thus reconstructing the entire compositional solution for an isothermal transport problem.

As mentioned previously, the utilization of compositional simulation within optimization and uncertainty quantification problems, where an adequately large number of simulations/realizations are required, becomes prohibitively expensive due to a large number of unknown variables and the nonlinear nature of the equations (Shirangi and Durlafsky, 2015). However, it is vital that the development of miscibility is accurately captured for gas injection problems. The first stage of the MSCT technique is used with the Peng-Robinson EoS thermodynamic model by Chen and Voskov, 2020 and accurately captures miscibility. They utilize the pseudo-binary model for a CO_2 injection optimization problem and find that the proxy model produces similar Net Present Values to the full multi-components model at a reduced computational cost as the number of degrees of freedom is reduced. Besides, the reduced forward problem also makes the convergence better in optimization due to the better feasibility of the solution (Chen and Voskov, 2020).

3.2.2. Multiscale reconstruction in physics for compressible fluids

In the transport problem, operators are defined for the simplified case where fluid densities are constant and equal. In this case, the total density is constant and equal to the phase density. Consequently, the accumulation operator is equal to the overall composition z , and the flux operator is equal to the fractional flow curve F . In the single-phase regions, β is a linear function of z whereas in the two-phase region the fractional flow curve is computed from the fractional flow of gas f and the partitioning coefficients x and y . The accumulation and flux operators of component i for an incompressible system are defined according to

$$\begin{aligned} \beta_i &= F_i = y_i f + x_i(1 - f) \\ \alpha_i &= z_i = y_i v + x_i(1 - v), \end{aligned} \quad (3.16)$$

and are shown in figure 3.1 for component C1 of a four component gas vaporizing described in table 3.1.

The MSCT method is adapted to include compressibility and flow. In this case, the phase densities are a function of pressure according to the formula for compressibility.

$$\rho(p) = \rho_0(1 + cr(p - p_0)). \quad (3.17)$$

As in the incompressible case, the operator space is computed as the convex hull of the initial and injection curves. Figure 3.2 illustrates the construction of the operator space for a compressible case where the operators are defined according to equations 3.12.

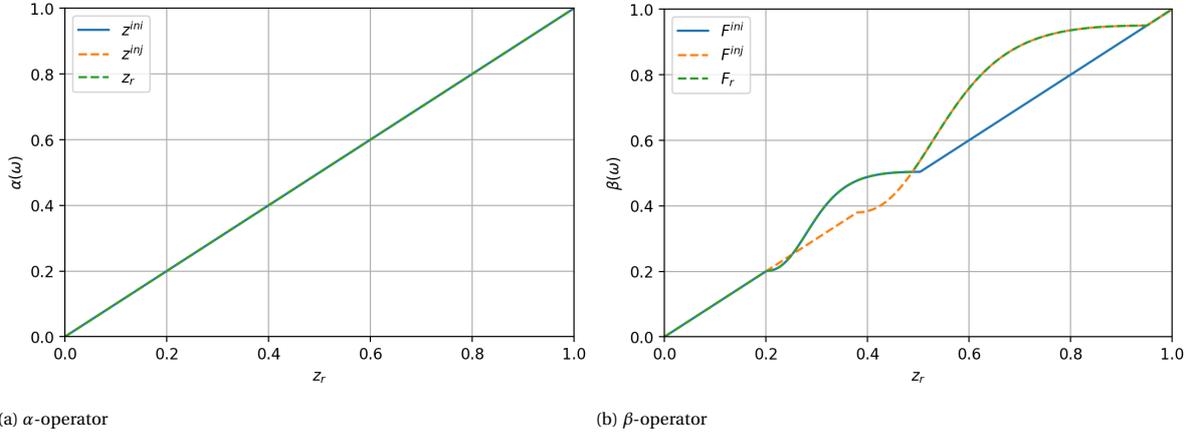


Figure 3.1: α - and β -operators of component C1 for incompressible fluids.

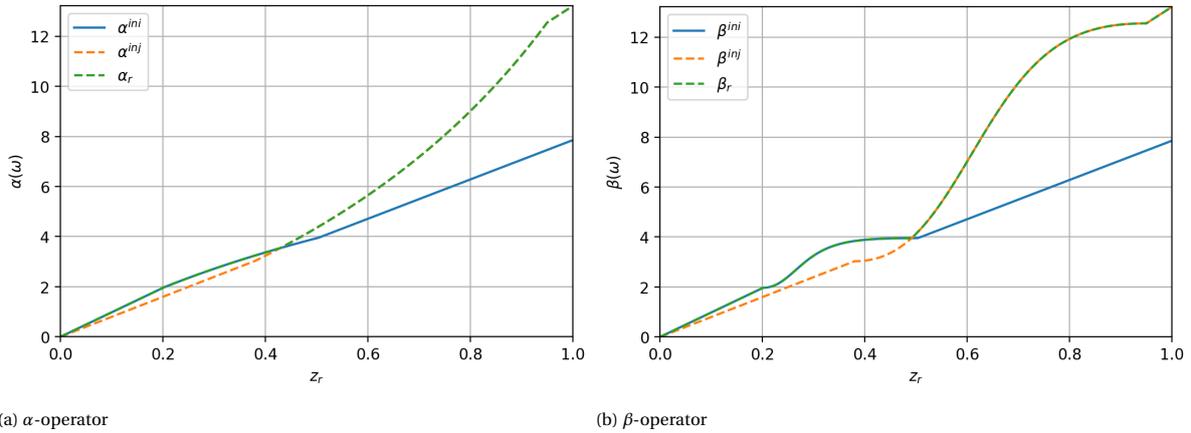


Figure 3.2: α - and β -operators of component C1 for non-equal fluid densities of 200 and 600 kg/cm at $p = 90$ bar. Fluid compressibilities are equal to 10^{-3} bar $^{-1}$ and 10^{-5} bar $^{-1}$ for oil and gas.

3.2.3. Four-component gas vaporization problem

The multi-scale reconstruction of physics (MSRP) method proposed by Ganapathy and Voskov, 2018 is utilized for a four-component gas vaporization problem with injection gas and initial oil compositions as described in table 3.1 below. The reference solution to this problem is shown in figure 3.3. The injection gas consists of 97% of methane while the initial oil composition consists of a mixture of all four components. In a quaternary displacement, the solution is defined according to three key tie-lines, namely the initial, injection and cross-over tie-lines. A tie-line is a line in the compositional space where the liquid and gas component fractions x_i and y_i are fixed and only phase fraction v is changing. For gas injection processes, the compositional path enters and exits the two-phase zone through the extension of the initial and injection tie-lines and is marked by discontinuities in the compositional and saturation profile (i.e. shocks). Consequently, three zones are identified in the compositional profile according to their phase state. These zones are the single-phase vapor, the single-phase liquid and the two-phase zones and are delineated by the leading and

trailing shocks. Downstream of the leading shock, the mixture is liquid at initial composition conditions while upstream of the trailing shock the composition consists purely of C_1 . Consequently, the area between these two shocks is occupied by both phases simultaneously. The compositional profile within the two-phase zone consists of tie-line and non-tie-line rarefactions connecting the tie-lines and following a compositional path.

Table 3.1: Vaporizing gas drive

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

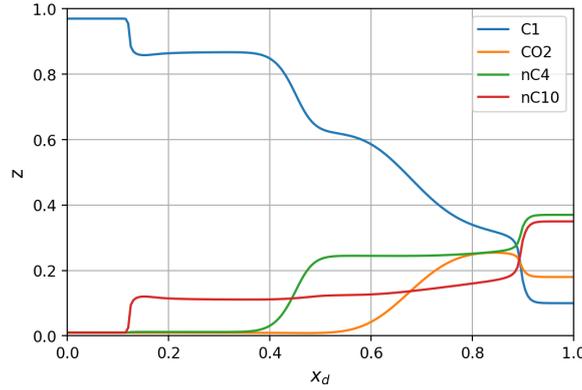


Figure 3.3: Composition (mole fraction) profile for a four-component vaporizing gas drive. The profile is characterized by the leading shock and trailing shock located at approximately $x_d = 0.9$ and $x_d = 0.12$. From left to right, they delineate three zones where single-phase vapor, both phases and single-phase liquid is present.

Since the solution of the compositional problem is characterized by the leading and trailing shocks, the reduced-order proxy model by Ganapathy and Voskov, 2018 is used to partially construct the solution. The displacement path can be constructed according to key tie-lines of the system as the leading- and trailing shocks can only enter and leave the two-phase zone along tie-line extensions (Orr, 2007; Voskov and Entov, 2001). The first stage of the MSRP reconstruction is used with fluid compressibility and includes flow as well as transport.

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

$$z_r = z_i \quad (3.18)$$

$$v = \frac{z_r - x_i}{y_i - x_i} \quad (3.19)$$

Thereafter, the pseudo-operator space $\beta_r(\omega)$ is defined as the convex hull of the union of each of the key tie-lines (eq.3.20). In the resulting operator space, the operator space switches from the initial to the injection tie-lines at their point of intersection in the nonlinear, two-phase zone.

$$\beta_r(\omega) = \text{conv}(\beta_i^{ini} \cup \beta_i^{inj}) \quad (3.20)$$

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

$$\rho_{tot,r} = \text{conv}(\rho_{tot}^{ini} \cup \rho_{tot}^{inj}) \quad (3.21)$$

$$\alpha_r(\omega) = \rho_{tot,r} z_r \quad (3.22)$$

The analytical construction of the pseudo-operator space is illustrated in figures 3.4a and 3.4b.

The total mobility operator is defined by tabulation of values from the reference solution with respect to the state variables. Care is taken to carry out the reference simulation with variable bottom-hole pressure controls in order to ensure that data points are distributed across the range of the entire parameter space. After that, the scattered data points are used to compute a structured OBL mesh by application of a linear interpolant. Figure 3.5 shows the gathered points and the resulting interpolation for the total mobility operator.

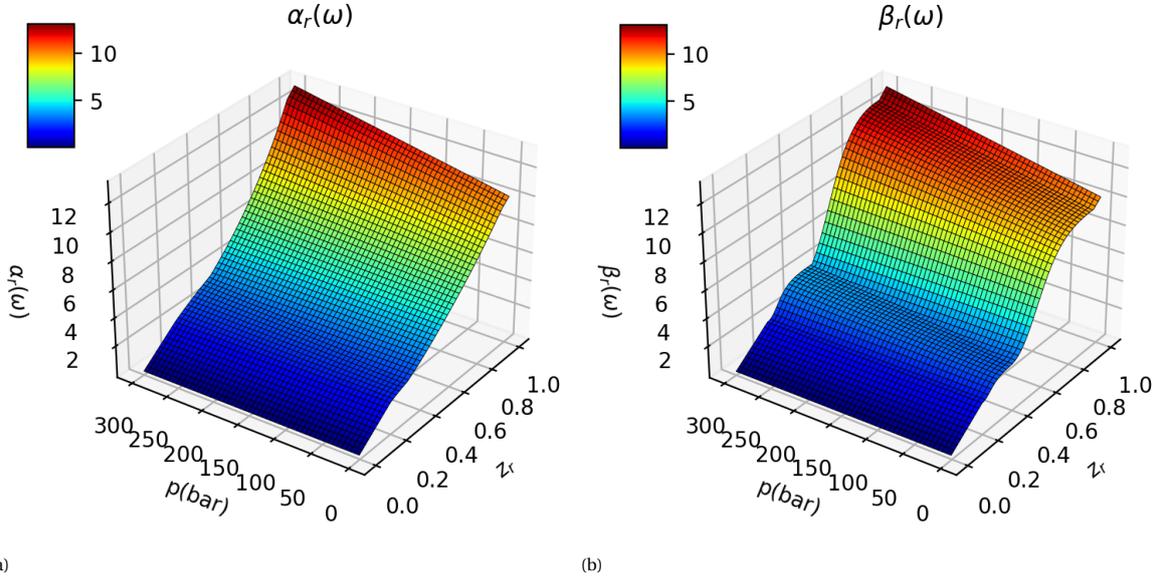
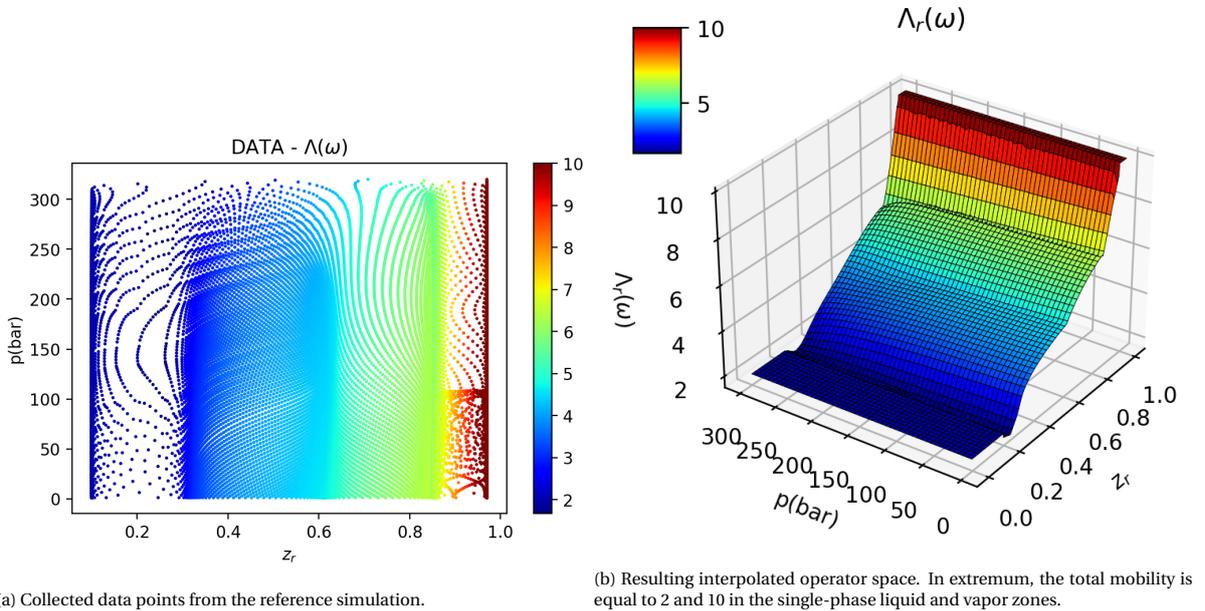


Figure 3.4: Operator space of the binary proxy model.



(a) Collected data points from the reference simulation.

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

Figure 3.5: Total mobility operator $\Lambda_r(\omega)$.

3.2.4. Lebesgue regularisation

The operator space of the aforementioned proxy model is adapted with neural networks to improve its predictive performance. The PIML-inspired method embeds neural networks in the nonlinear operator of the proxy model and uses the Lebesgue integral of the forward solution with respect to the full compositional

model in the loss function to improve the performance of the proxy. Our method is implemented in TensorFlow and utilizes the operator-based linearization approach to compositional simulation. A transfer training approach is employed for increased computational efficiency and solution accuracy Jin et al., 2021.

In this work, a supervised learning scheme applies physics-informed machine learning to adapt the operator space of the aforementioned proxy model and improve its predictive performance. This is achieved by embedding neural networks into the nonlinear operator of the PDE and utilizing the forward solution in a regularization term of the loss function. Our approach integrates the distance between the leading and trailing shocks of the proxy model with respect to the reference model and penalizes the loss function so that the neural network learns the operator space for which the misfit of the shocks is minimal. This can be effectively implemented as an application of the Lebesgue integration in the loss function instead of the Riemann integral.

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

$$L_1 = \frac{1}{N_o^2} \sum_{i=1}^{N_o^2} (f^i(\omega) - f_\theta^i(\omega))^2. \quad (3.23)$$

In the second step, model weights are copied/transferred to a new neural network and a fully implicit solver is included in the loss function. The solver utilizes the predicted operator space and solves for pseudo-composition z_r . Next, the quality of the resulting solution is assessed by calculating the Lebesgue integral between $z_r(x)$ and $z_{ref}(x)$ for N_t number of time steps and adding it to the loss

$$L_2 = \frac{1}{N_t N_z} \sum_{t=1}^{N_t} \sum_{i=1}^{N_z} \sqrt{(x_i^t(z) - x_i^t(z_r))^2}. \quad (3.24)$$

This term within the loss function works to penalize the neural network if the distance between the shocks of the reference and proxy model is large since for any given time-step, Δx

$$\Delta x = -\lambda \left(\int \frac{\partial F_\theta}{\partial z_r} \partial t - \int \frac{\partial F}{\partial z_{ref}} \partial t \right), \quad (3.25)$$

must approach zero. The new total loss function then reads:

$$L = L_1 + L_2. \quad (3.26)$$

After solving the proxy model and computing the loss function, TensorFlow's reverse-mode automatic differentiation is applied to handle back-propagation and update the ensemble of weights and biases θ (Abadi et al., 2016). Effectively, this loads the analytical approximation of the operator space into the neural network. Training in this step is carried out for 600 training iterations and the Adam stochastic optimizer with a constant learning rate of 0.001 is used. Training is carried out for a one-dimensional homogeneous reservoir with constant permeability. Furthermore, the training scheme is described according to neural network architecture, optimizer, and activation function. The architecture of the neural network is identical to that of Raissi et al., 2019 and Fuks and Tchelepi, 2020. It consists of 8 hidden layers with 20 neurons per layer. Standard gradient descent is used for the optimizer with a constant learning rate of 0.0001. A small learning rate is expressly chosen since it is necessary for the operator space to maintain the part of its form that defines the shocks in accordance with the velocity constraint of the method of characteristics. The neural network weights are initialized randomly with Xavier/Glorot initialization. The activation function σ is the hyperbolic tangent function \tanh and introduces nonlinearity to the system.

3.3. Compositional upscaling

Another alternative to the reduction of the degrees of freedom in the reservoir simulation model is upscaling. Conventional upscaling methods for compositional models try to capture the dynamic behavior of the fine-scale model in the coarse model by application of upscaling functions derived from the fine-scale model's solution. However, these upscaling functions need to be redefined whenever the boundary conditions change

which is impractical for optimization problems. Besides, it is desirable to be able to carry out sensitivity analyses of your model without having to recompute the fine-scale model. Furthermore, upscaling functions can be difficult as they typically use single-point upstream weighting and require some degree of post-processing where outliers, non-physical points and noise are removed (Barker and Thibeau, 1997).

In this work, a rigorous ML-based upscaling approach is proposed that defines the upscaled operator space. The upscaled operators are defined for the entire compositional space (i.e. the vector of n_c -compositions) and generalize the conventional concepts of transport coefficients, upscaled multiphase functions or non-equilibrium thermodynamics into a uniform operator-based upscaling in the parameter space which can be directly applied in DARTS. This upscaled operator space is constructed from a limited, characteristic piece of the fine-scale model and is applicable to models with variable injection compositions. Therefore, the proposed approach negates the need for recomputing the fine-scale solution when boundary conditions are changed.

The section is divided into two parts, firstly conventional methods for compositional upscaling are described. These methods employ upscaling functions to adjust for dynamic and non-equilibrium effects on the coarse scale and are phase molar mobilities and non-equilibrium adjusted transport coefficients.

3.3.1. Conventional upscaling

The compositional upscaling framework of Iranshahr et al., 2014 and Salehi et al., 2019 proves its efficiency in many practical problems. In their methodology, thermodynamic equations as well as phase molar mobilities and single-phase parameters are upscaled and used in the coarse-scale model. As an alternative, transport coefficients, $\gamma_{i,j}$ can be used to upscale phase molar mobilities for each component, $(\rho_j \lambda_j)^*$. Transport coefficients tabulate the sum of component fluxes across the fine-scale interfaces between two grid cells with respect to the average composition or saturation of the upstream block. These coefficients are incorporated in the flux term of the conservation equation of the coarse-scale model and effectively function as scalars of the flux. The coarse-scale conservation equations reads,

$$\frac{\partial(\phi^* \rho_{tot}^* z_i^c)}{\partial t} - \nabla \left(\sum_{j=1}^{N_p} \gamma_{i,j} x_{i,j}^* \rho_j^* k^* \lambda_j^* \nabla p^c \right) = 0, \quad i = 1, \dots, n_c, \quad (3.27)$$

where upscaled functions are denoted by $*$ and coarse-scale quantities denoted by c . Note that the form of the coarse-scale conservation equation is identical to that of the fine-scale equation which strictly speaking is not the case due to the nonlinearity of the governing equation (Durlafsky, 1998; Iranshahr et al., 2014). Note that upscaled functions or transport coefficients are calculated at the interface of grid blocks and therefore need to be tabulated for each of the flow directions.

In our study, vertical upscaling is carried out and only upscaled flow functions in the x-direction are required. Furthermore, the transmissibility T^* is upscaled using single-phase flow-based upscaling,

$$T^* = \frac{\sum_l (q^f)_l}{\Delta p}, \quad (3.28)$$

where the fine-scale flux across the fine-scale interface's (l) are summed and divided by the difference of the average fine-scale pressures, Δp .

In non-equilibrium thermodynamic upscaling, transport coefficients $\gamma_{i,j}$ are assumed to be 1 and the upscaled phase molar mobilities have related the sum of fine-scale molar flow rates to the total flow rate computed with the upscaled permeability and averaged fine-scale pressure:

$$(\rho_j \lambda_j)^* = \frac{\overline{\rho_j u_j}}{k^* (\Delta p / \Delta x^c)}. \quad (3.29)$$

The sum of fine-scale molar flow rates across the coarse-scale interface is calculated for blocks containing two-phase (2p) and single-phase (1p) mixtures across all of the fine-scale interfaces of area A :

$$\overline{\rho_j u_j} = \sum_{1p} (\rho_j u_j A_i)_l + \sum_{2p} (\rho_j u_j A_i)_l. \quad (3.30)$$

Further quantities are computed from the molar amounts per component and phase, and are summed over the fine-scale grid blocks:

$$n_{i,j}^* = \sum_{1p} (V \phi \rho_j z_i) + \sum_{2p} (V \phi \rho_j x_{i,j}). \quad (3.31)$$

The molar amounts are used to compute the upscaled partitioning coefficients,

$$x_{i,j}^* = \frac{n_{i,j}^*}{\sum_i^{n_c} n_{i,j}^*} = \frac{n_{i,j}^*}{n_{tot,j}^*}, \quad (3.32)$$

and the total density,

$$\rho_{tot}^* = \frac{\sum_j^{n_p} n_{tot,j}^*}{\sum_j^{n_p} V_j}, \quad (3.33)$$

with phase volumes V_j ,

$$V_j^* = \sum (S_j \phi(\Delta x \Delta y \Delta z))_l. \quad (3.34)$$

As an alternative to non-equilibrium thermodynamic relations, upscaled transport coefficients can be computed per component and phase as the ratio of the sum of fluxes across fine-scale interfaces l and the flux across the coarse-scale interface estimated with Δp^c :

$$\gamma_{i,j} = \frac{\sum_l q_{i,j}}{(\rho_j \lambda_j)^* T^* x_{i,j}^* \Delta p^c} = \frac{\sum_l (\rho_j u_j x_{i,j} A)_l}{(\rho_j \lambda_j)^* T^* x_{i,j}^* \Delta p^c}. \quad (3.35)$$

In this case, phase component fractions $x_{i,j}^*$ and densities ρ_j^* will be computed based on the equation of state.

3.3.2. Rigorous upscaling

The coarse-scale conservation equation can be discretized and written in its operator form as

$$V \phi(\alpha_i^*(\omega_{n+1}) - \alpha_i^*(\omega_n)) - \Delta t \sum_i \beta_i^*(\omega) T^*(\xi) \Lambda^*(\omega) \Delta p^c = 0, \quad i = 1, \dots, n_c. \quad (3.36)$$

In the proposed rigorous upscaling method, observations of the fine-scale $\beta(\omega)$ are used to match the coarse scale operator space to the fine-scale results,

$$\beta_i(\mathbf{z})^* = \frac{\sum_l (\beta_i T \Delta p)_l}{T^* \Delta p^c}, \quad i = 1, \dots, n_c. \quad (3.37)$$

This approach can be seen as a combination of multiphase flow and compositional upscaling. In addition, it also generalizes both non-equilibrium thermodynamics and transport coefficients upscaling approaches which are translated to the upscaled convective operators.

The upscaled operators are parameterized with respect to the complete compositional space. Data points are compiled with a test bench where a smaller but characteristic portion of the full model is placed in between high permeability buffer cells. Figure 3.6 illustrates how the characteristic permeability pattern of a fine-scale model (figure 3.6a) containing $n_x \times n_z$ grid blocks is placed in the test bench (figure 3.6b). In the test bench, uniform pressure boundary conditions are applied and the left boundary cells are initialized at injection composition conditions. The upscaling functions making up the upscaled operator space are computed at the interface between cells 1 and 2. This procedure is repeated for a series of injection compositions and the resulting point cloud of data is interpreted with an interpolation function or regressive neural network to create a structured OBL mesh. An example of the collected data points and the corresponding interpretation for the operator space is shown in figure 3.8.

The compositional problem handled in this report is a gas injection problem where either carbon dioxide, CO_2 , or methane, C_1 , is injected into a reservoir containing a mixture of CO_2 , C_1 and oil nC_{10} . Unless mentioned otherwise, the model neglects compressibility, diffusion, gravity, capillarity and chemical reactions. Furthermore, the thermodynamic model uses constant K-values. The test bench is run for varying injection compositions ranging between 100% of C_1 to 100% of CO_2 . The initial composition remains constant at 15% of C_1 , 29% of CO_2 and 56% of nC_{10} . Figure 3.7 illustrates the various injection points and the initial point used in the test bench. Injection points are included with non-zero amounts of nC_{10} to ensure that the resulting point cloud of data is distributed so that the relevant operator space is well-defined. The compiled operator space is tested for three cases where CO_2 , C_1 or a mixture of both are injected into the reservoir. The exact injection composition per case along with the K-values per component is given in table 3.2. The performance of the model for each of the cases is assessed with the L^2 and L^{inf} norms of the overall composition at the production block.

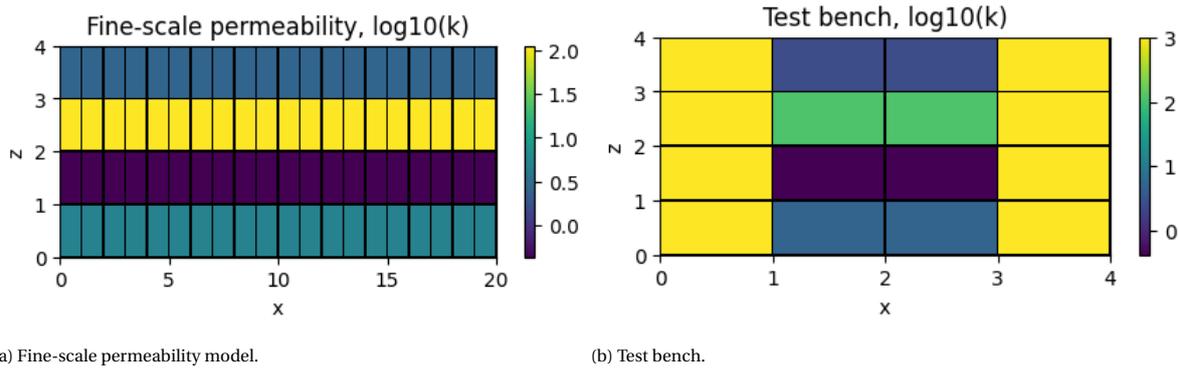


Figure 3.6: A portion of the fine-scale model is supplanted into the test bench and subjected to varying injection compositions. Uniform pressure boundaries are imposed where the left boundary is set at 180bar and the right boundary at 20bar. Injection compositions range between 100mol% C_1 to 100mol% CO_2 .

Table 3.2: Injection compositions for each of the three test cases.

Component	C_1	CO_2	nC_{10}
case 1	0.95	0.05	0.00
case 2	0.05	0.95	0.00
case 3	0.50	0.50	0.00

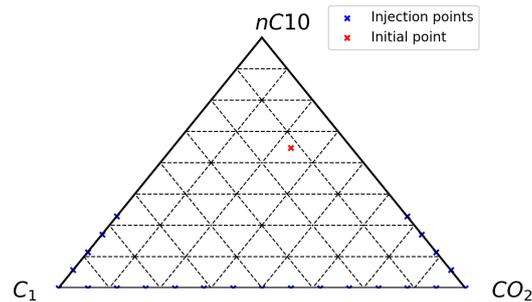


Figure 3.7: Injection composition points are distributed along the axis between 100mol% C_1 to 100mol% CO_2 . Additional injection points are included with a non-zero amount of nC_{10} in order to ensure that the operator space is well sampled.

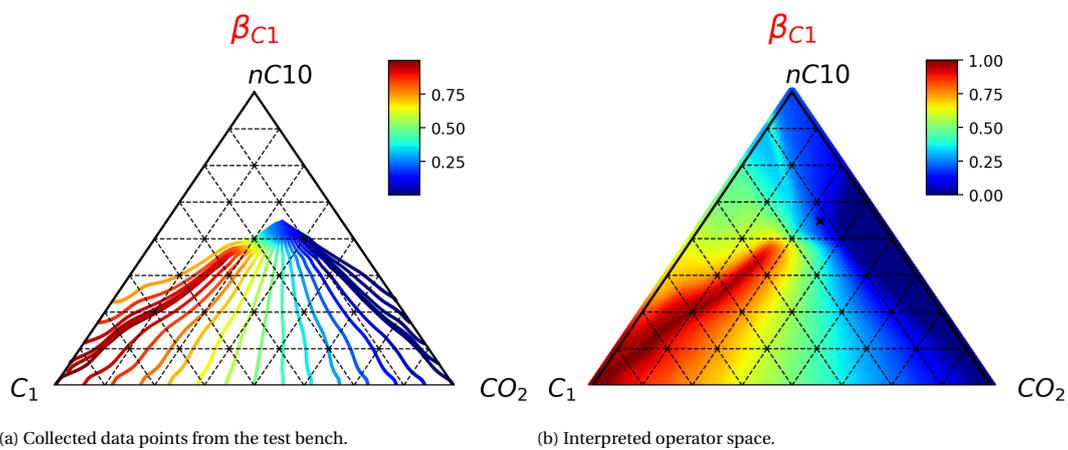


Figure 3.8: Example of collected data points and corresponding operator space

4

Results

4.1. Training of pseudo-binary proxy model

This section describes the main results achieved with the transfer learning approach. Three models are compared, the reference model with n_c components and the binary model before ('base ML') and after transfer training ('transfer ML'). Training is carried out with a one-dimensional reservoir model and the trained model is further validated with two-dimensional heterogeneous layers of the SPE10 model. The trained proxy model shows a marked improvement in the estimation of the trailing shock while estimations of the leading shock remain the same as the base ML model already makes an accurate estimation of the leading shock. Additionally, the error of the phase-state classification of the proxy model is consistently reduced. The average difference in breakthrough time for the transfer ML and base ML models with respect to the reference model is 293days versus 570days for the trailing shocks and 15days versus 16days for the leading shock.

4.1.1. 1D-training

The operator space of the total mobility operator $\Lambda(\omega)$ and the flux operator $\beta(\omega)$ is adapted with our transfer training approach. In the first step, the neural network is exclusively trained to approximate the OBL mesh with labeled data from the analytical construction. At this stage, it is found that the neural network has a smoothing effect on the operator space which can be seen in figure 4.1. This figure compares the operator space at base ML training with the labels. Additionally, due to the structured nature of the training data, it is found that the neural network is able to accurately approximate OBL grids of different densities.

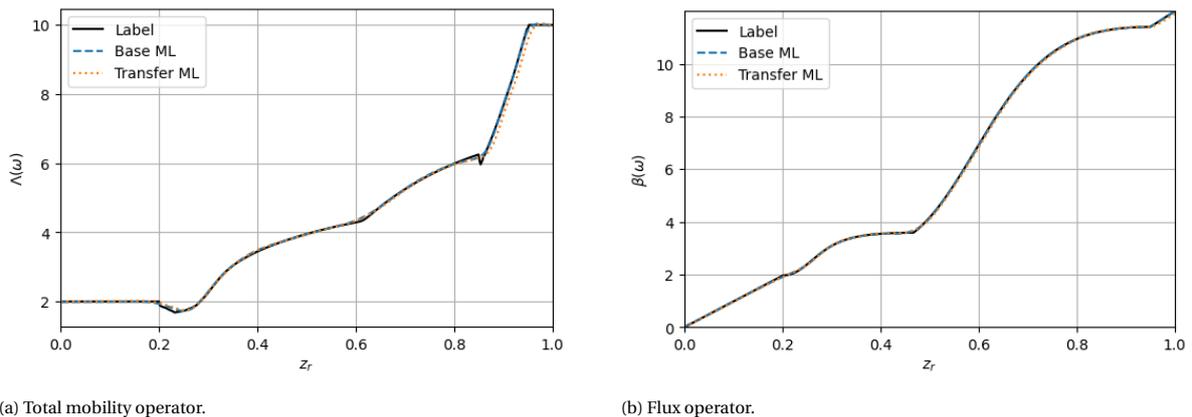


Figure 4.1: Comparison of the operator space before (base ML) and after transfer training (transfer ML) to the labels at $p = 180\text{bar}$.

In the second step, the estimated operator space is used within a fully implicit solver embedded in the loss function of the neural network as described previously. The solver utilizes the OBL approach as it interpolates required values from the vertices of the OBL mesh. Training is carried out for a one-dimensional homogeneous reservoir containing 200 grid blocks where the permeability is equal to 100mD and the length

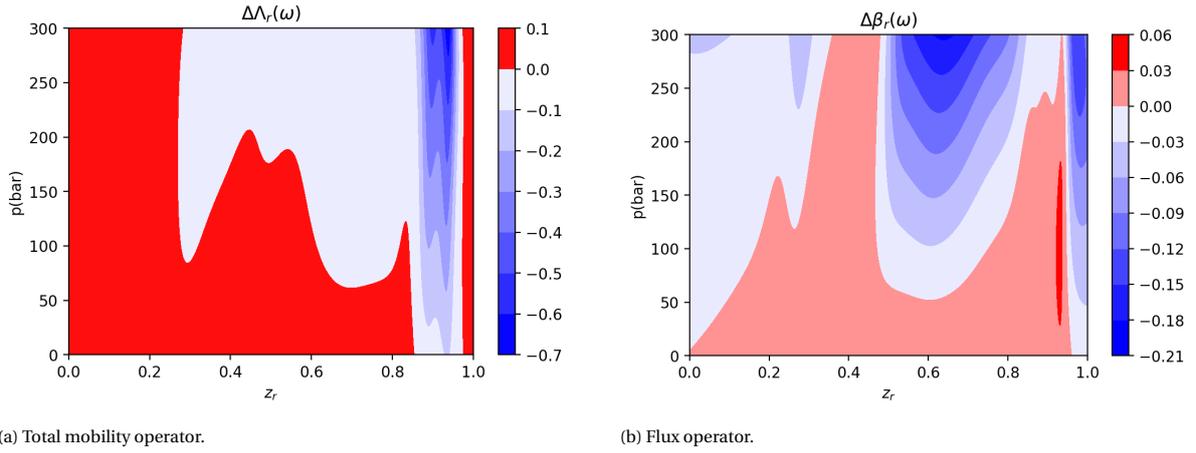


Figure 4.2: Difference of the operator space before and after training. The greatest changes in the operator space are concentrated in the area relevant to the trailing shock.

of the reservoir is 200m. During the training period, the bottom-hole pressures are fixed at the limits of the OBL mesh, namely 1bar and 300bar . The Lebesgue integral of the leading and trailing shocks is computed at 10 discrete time steps ranging from $t = 1\text{day}$ to 10days . The range at which the Lebesgue integral of misfit is calculated lies between $z_r = 0.90$ and 0.96 for the trailing shock and between $z_r = 0.11$ and 0.30 for the leading shock.

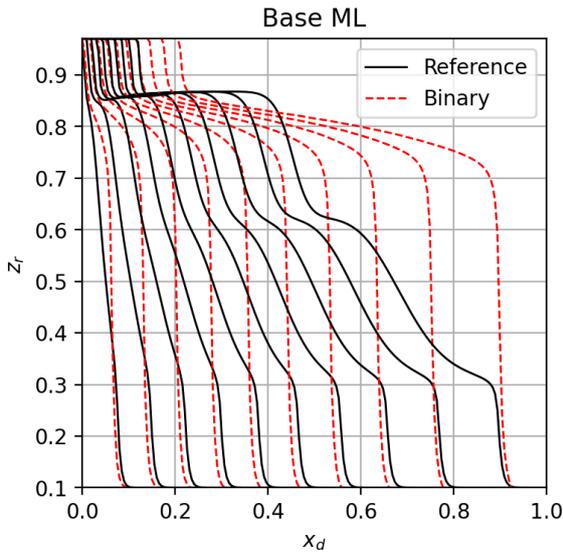


Figure 4.3: Forward solution of the binary model versus the reference solution before transfer training (base ML).

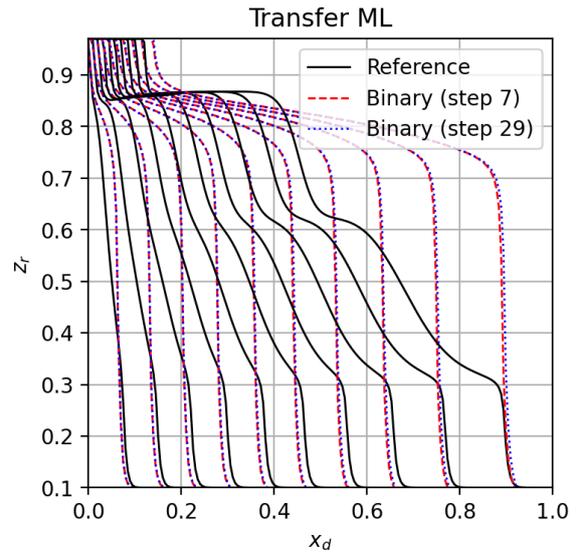


Figure 4.4: Forward solution of the binary model versus the reference solution after transfer training (transfer ML).

The evolution of the loss function in the second step is plotted in figure 4.5 and the evolution of the misfit of the trailing and leading shocks at every time step ts and training iteration is given in figure 4.6. A global minimum is reached at training step 7 after which improvements remain negligible. The misfit of the leading and trailing shock decreases and the estimation of the proxy model is improved with respect to the conservative reference solution of n_c components. In particular, the estimation of the trailing shocks is visibly improved as can be seen in figure 4.4 while changes in the position of the leading shock remain modest. Correspondingly, changes in the operator space for both the total mobility and flux operators are generally greatest in the area relevant to the trailing shock where the overall composition is changing the most. Changes in the operator space before and after training are visualized in figure 4.2.

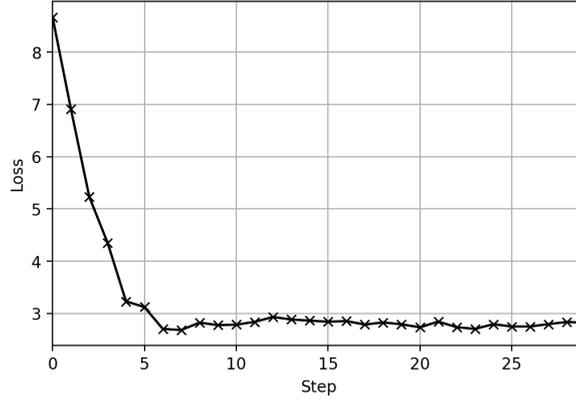
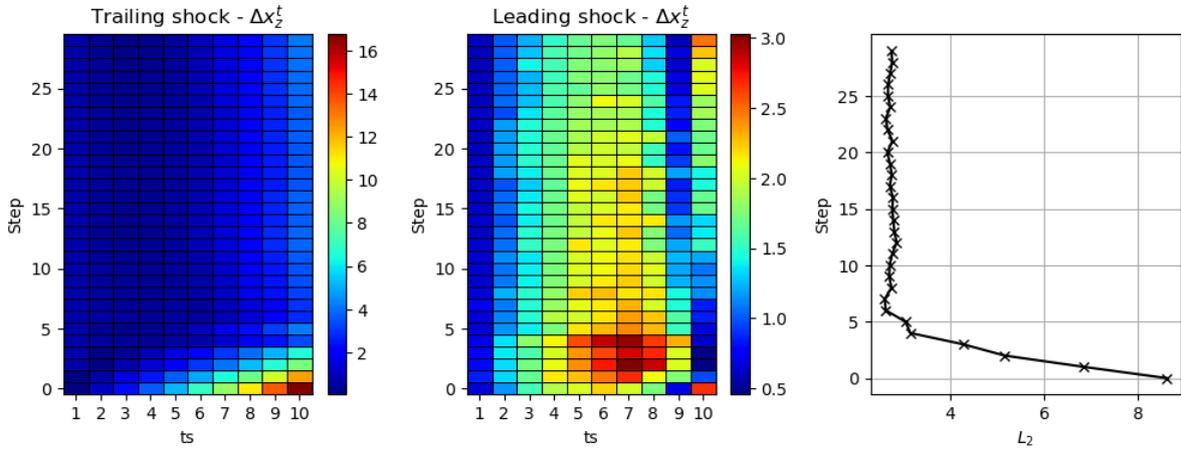


Figure 4.5: Evolution of the loss function.

Figure 4.6: Evolution of the Lebesgue integral of the trailing and leading shocks that make up the L_2 term of the loss function.

4.1.2. Horizontal layers of SPE10

The trained model is applied to two-dimensional heterogeneous layers from the SPE10 geological model. The two-dimensional models consist of (220×60) cells. The SPE10 geological model is divided into two parts, namely the Tarbert formation (top 35 layers) and the Upper Ness formation (bottom 50 layers), which are representations of prograding near shore and fluvial environments (Christie and Blunt, 2001). The cells are of size $\Delta x = 3m$, $\Delta y = 6m$ and $\Delta z = 0.6m$. A 5-spot well pattern is used with an injection well in the middle at fixed bottom hole pressure of 300 bars injecting at 97% of methane and 4 production wells in the corners at 1 bar. The breakthrough times of the first and last breakthroughs are reported for each of the models. The first and last breakthrough coincides with the breakthrough of the leading and trailing shock at any of the four wells. In addition, the quality of the prediction of the locations of the shocks in space and time is quantified by analysis of the phase state of each grid cell.

The performance of each model is evaluated by quantifying the error of the phase-state classification of the resulting compositional profile. The phase-state maps identify three zones within the reservoir corresponding to single- and two-phase zones where gas (2), oil and gas (1), or oil (0) are present. The state of each cell is classified according to the tie-line end points of the injection and initial tie-lines and the error of the model is quantified as the ratio of the number of misclassified grid blocks with respect to the reference model and the total number of grid blocks.

$$T(S_r(x, y, t) - S_{ref}(x, y, t)) = 1 \quad \text{for} \quad S_r - S_{ref} \neq 0 \quad (4.1)$$

$$Error(t) = \frac{\sum_i^{n_b} T(S_{r,i} - S_{ref,i})}{n_b} \quad (4.2)$$

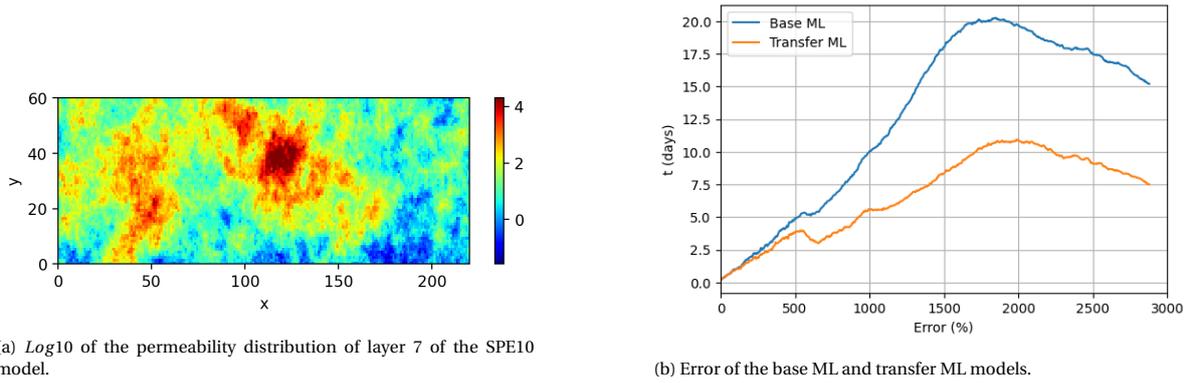


Figure 4.7: Evolution of the error for layer 7 of the SPE10 model before (base ML) and after transfer training (transfer ML).

Figures 4.12a and 4.12b illustrate the distribution of the phases and the error of the proxy model before and after training for layer 7 of the SPE10 model. The boundary between the zones delineates the location of the leading and trailing shock within the reservoir and thus the error gives an approximation of how well the location of the shock is approximated in space and time. From the error maps, it is observed that the estimation of the location of the trailing shock is visibly improved for the trained model whereas the estimation of the leading shock remains the same or is slightly better. The evolution of the error as the function of time is reported in figure 4.7b and the permeability of layer 7 is shown in figure 4.7a. The error increases as the displacement front grow and the region of displacement increases.

The performance before and after training of the proxy model is compared for the top 15 layers of the SPE10 model. The breakthrough times of the leading and trailing shocks are reported per layer in figures 4.8 and 4.9 together with the corresponding error of the phase-state classification. The trained model consistently outperforms the model trained at the first stage in terms of error and breakthrough time of the trailing shock while the breakthrough time of the leading shock effectively remains the same as the base ML model already makes a good estimation. The error of the trained model at breakthrough remains below 7.5%. The average difference in breakthrough time for the trained and untrained models with respect to the reference model is 293days versus 570days for the trailing shocks and 15days versus 16days for the leading shock.

The first and last breakthrough times per layer of the bottom 15 layers of the SPE10 model are reported in figures 4.10 and 4.11. The estimation of the leading shock remains virtually the same while the estimation of the trailing shock is consistently improved. Note that in the case where production wells are located outside of sand channels, breakthrough takes a very long time for both the trailing and leading shocks. Consequently, breakthrough times are reported in log form.

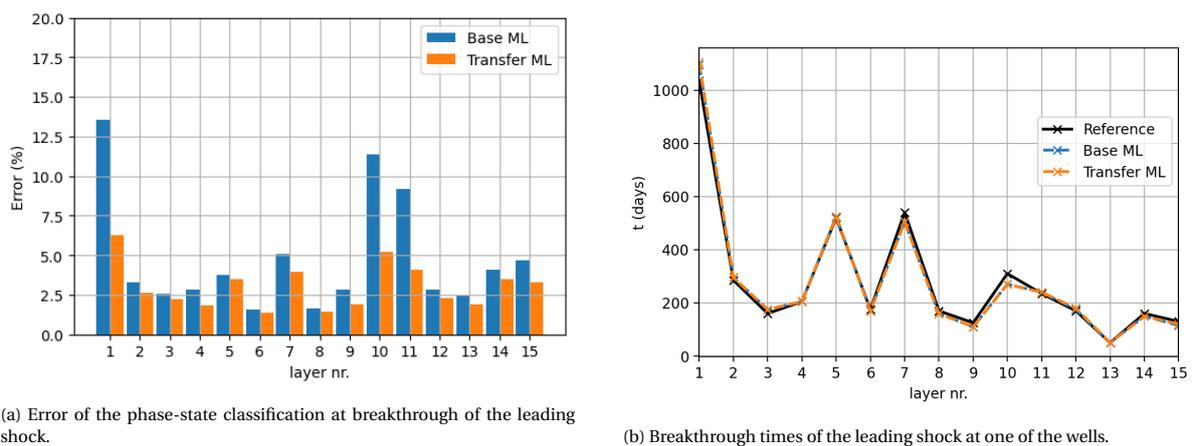
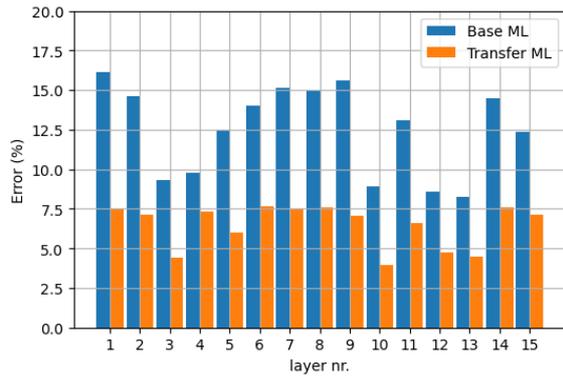
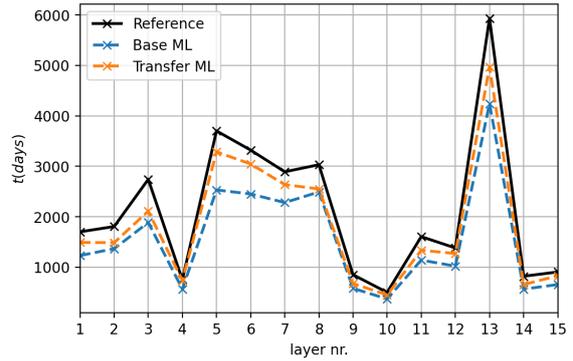


Figure 4.8: The breakthrough time of the leading shock at one of the wells and the corresponding error of the phase-state maps is assessed for the top 15 layers of the SPE10 model.

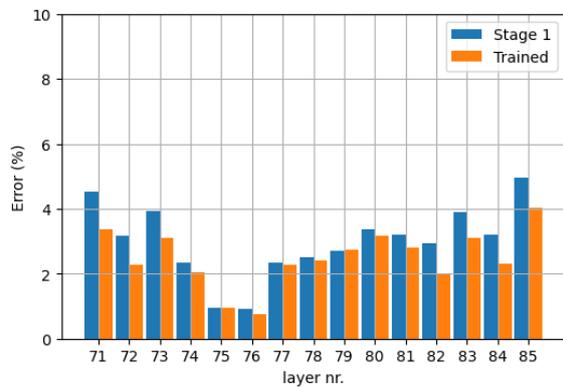


(a) Error of the phase-state classification at breakthrough of the trailing shock.

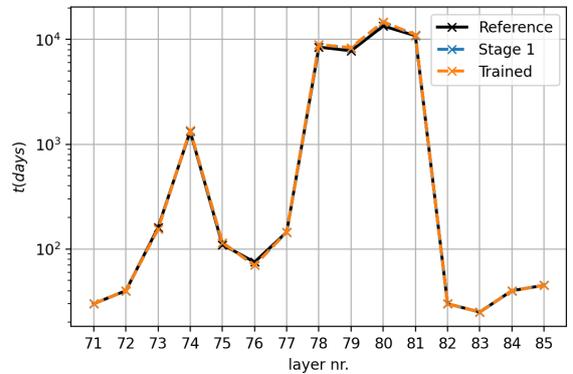


(b) Breakthrough times of the trailing shock at one of the wells.

Figure 4.9: The breakthrough time of the trailing shock at one of the wells and the corresponding error of the phase-state maps is assessed for the top 15 layers of the SPE10 model.

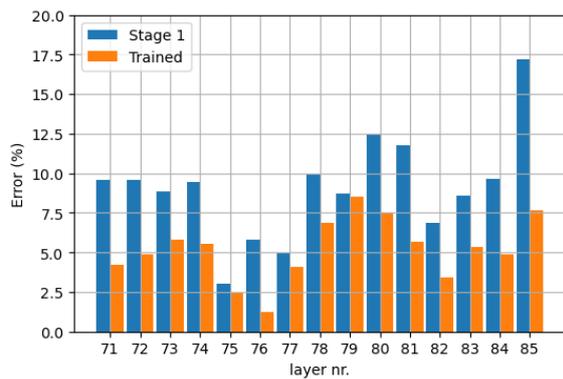


(a) Error of the phase-state classification at breakthrough of the leading shock.

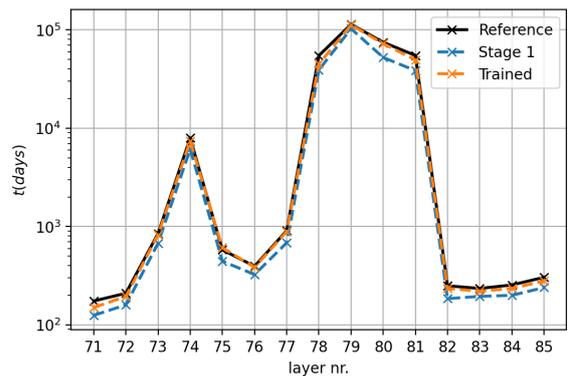


(b) Breakthrough times of the leading shock at one of the wells.

Figure 4.10: The breakthrough time of the leading shock at one of the wells and the corresponding error of the phase-state maps is assessed for the bottom 15 layers of the SPE10 model.



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



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

Figure 4.11: The breakthrough time of the trailing shock at one of the wells and the corresponding error of the phase-state maps is assessed for the bottom 15 layers of the SPE10 model.

4.2. Rigorous upscaling of layered models

This section contains the results of rigorous upscaling. To start, an initial demonstrative example consisting of four layers is given. Thereafter, the ability of the method to handle n_z number of layers and added physics

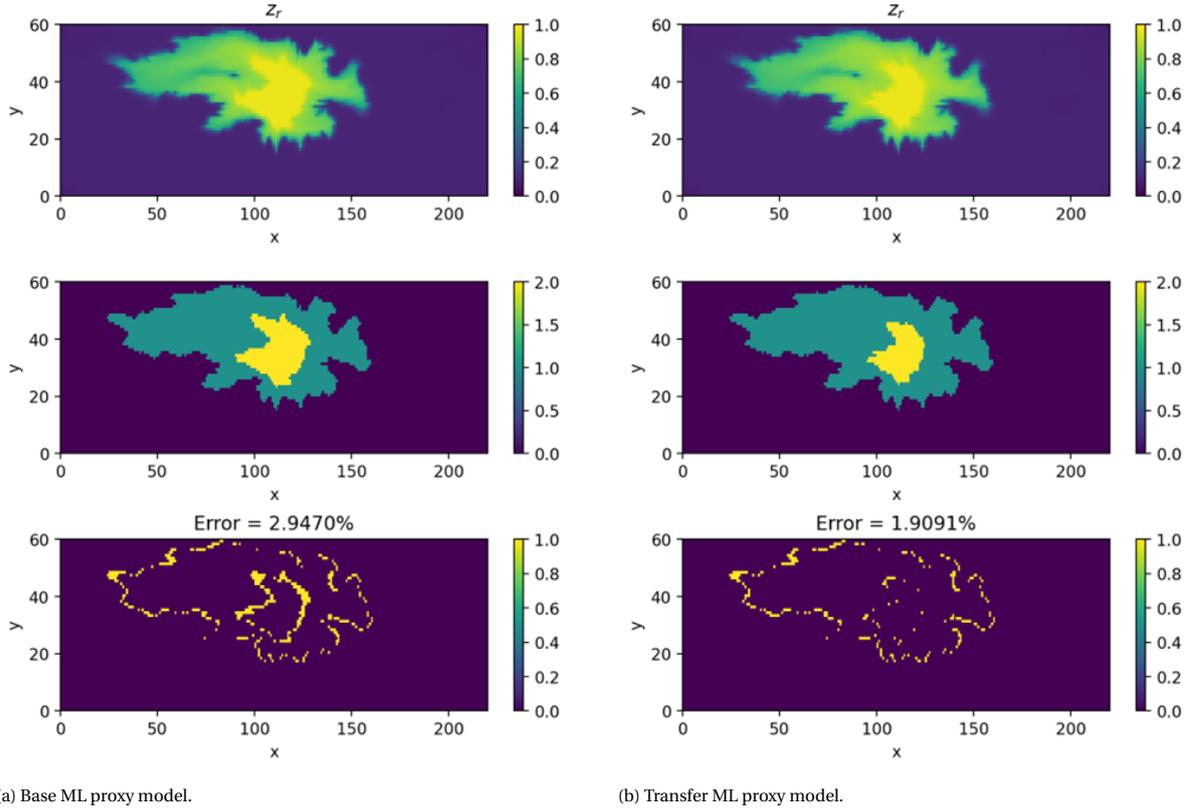


Figure 4.12: Distribution of the phase-states and the corresponding error at $t = 150$ days for layer 7 of the SPE10 model. The phase-states correspond to single- and two-phase zones where gas (2), oil and gas (1), or oil (0) occupy the grid block. The transfer ML model indicates a marked improvement in the estimation of the trailing shock.

is assessed. In the final part, a neural network is used to learn the operator space for a model containing three layers. It is found that, the upscaling procedure provides a reliable method for upscaling compositional models with different injection compositions.

4.2.1. Four-layer model

In the initial demonstrative example, a baseline idea of the performance of the upscaling procedure is developed. The four-layer model contains $n_x = 20$ cells. Permeabilities from top to bottom are 2.6, 113, 0.4 and 5 mDarcy. Grid cell dimensions of the model are uniform and equal to 1 m in every direction. The operator space of the flux and total mobility operators is approximated with an interpolation function instead of a neural network and a baseline idea of the performance is established. In this model fluid is incompressible and phase densities are equal:

$$\rho_{tot} = \sum_j^{n_p} \rho_j s_j = \rho \sum_j^{n_p} s_j, \quad (4.3)$$

therefore the total density is constant and equal to the phase density ($\rho_g = \rho_o = \rho_{tot}$). Further assumptions include zero vertical permeability and constant K-values.

The quality of the upscaled model's prediction is compared to that of the coarse-scale model, where only the absolute permeability is upscaled. The cumulative production of oil and gas for each of the three cases is plotted in figure 4.13. The upscaled model accurately captures the behavior of the fine-scale model while the coarse model does not. In the case of injection of C_1 the cumulative oil and gas production of the upscaled model precisely matches that of the fine-scale model. Correspondingly, the predicted evolution of the molar composition at the production well by the upscaled model for each of the cases is consistently better than that of the coarse model. The evolution of the molar composition at the production block is reported per case and model in figure 4.14. In figures 4.16 and 4.15, the L_2 and L_{inf} norms are compared for each of the models and cases. For cases 1 and 2, the injection component has the greater error however for case 3, where both components are injected, the error is greatest for the production component $n_{C_{10}}$.

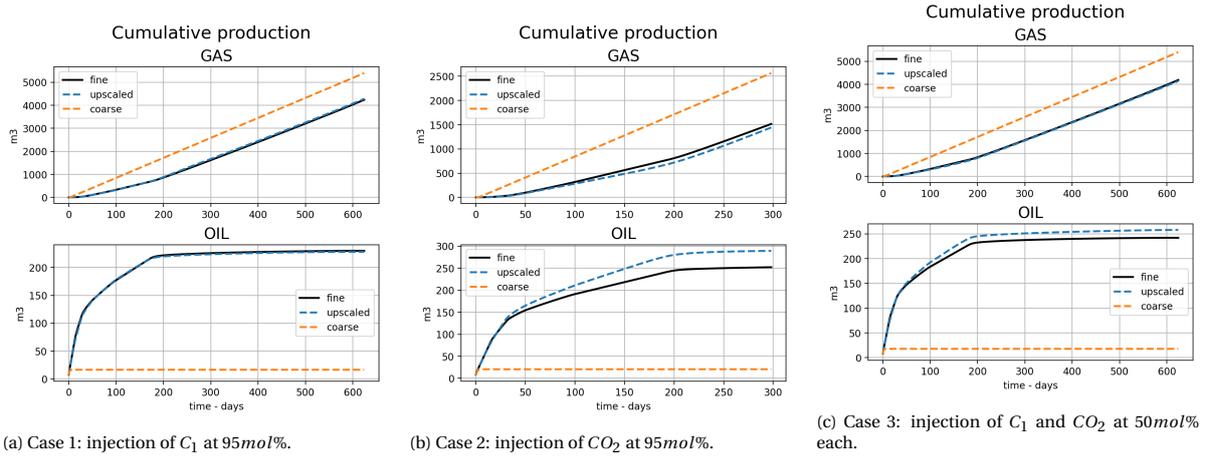


Figure 4.13: Cumulative production of oil and gas for the three different cases is compared to that of the coarse model fine-scale models.

4.2.2. Upscaling procedure for n_z -layered models

The procedure for upscaling of $n_z \times n_x$ layered models is tested for models containing 4, 8 and 16 layers. The upscaled operator space is applied to models containing $n_x = 20$ cells for the three cases described in table 3.2 and the error is assessed for the production component nC_{10} . Layer permeabilities for the models are randomly sampled from a log-normal distribution of permeability where the reservoir is split into two parts. The distribution from which the layer permeabilities are sampled is shown in figure 4.17. For the 4-layer, figure A.1, 8-layer, figure A.2 and 16-layer, figure A.3, models that the L_2 and $Linf$ norm is much greater for the coarse model. Furthermore, the L_2 norm remains below 1.0 and the $Linf$ norm remains below 0.15 for all three sets of models indicating that the procedure can be used for a larger number of layers. However, it is important to point out that the vertical permeability is equal to zero. In the following section, the procedure is tested for a model containing 24 layers and non-zero vertical permeability.

4.2.3. Added physics and vertical permeability

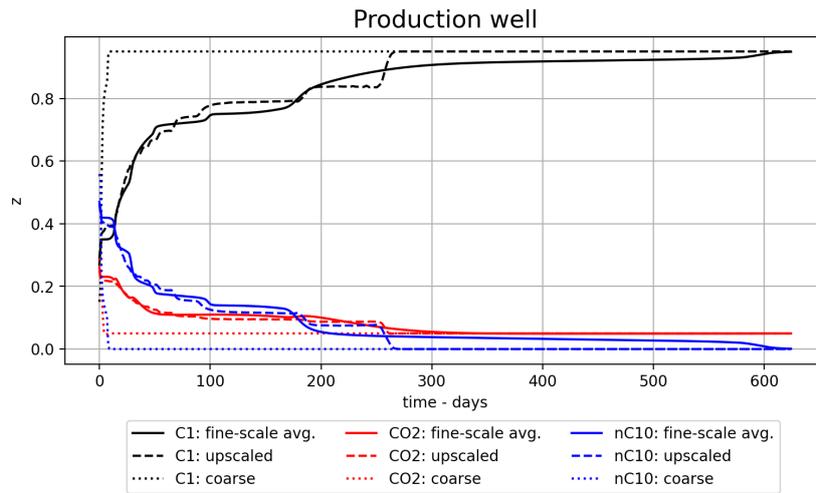
The upscaling procedure is tested for a layered model with permeabilities from a real-world example and non-zero vertical permeability. The vertical permeability of the model is constant and equal to $1D$. Additionally, the physics of the model is extended to include gravity and fluid compressibility. Fluid parameters for the fine-scale simulation are described in table 4.1 below. The permeabilities of the model are shown in figure 4.18a and the distribution of z_{C_1} at $t = 0.1days$ is shown in figure 4.18b. Two zones can be identified in the permeability model. The lower part generally contains higher permeability layers while the upper part has a relatively lower permeability. Between $z = 10$ and $z = 15$, a high permeability streak is present through which C_1 (gas) passes immediately. Due to the vertical communication of the layers and the buoyancy of gas, C_1 concentrates under low permeability layers and further facilitates the spread of C_1 downstream of the injection well into lower permeability layers.

Table 4.1: Fluid parameters

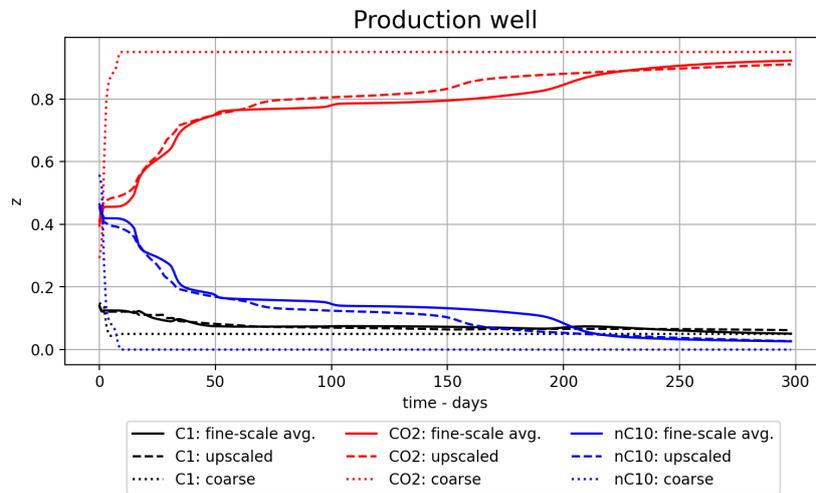
	Gas	Oil
Density (g/m^3)	200	600
Compressibility (bar^{-1})	10^{-3}	10^{-5}
Viscosity (cP)	0.1	0.5

The test bench is used to approximate the upscaled operator space and is shown in figure 4.19. In the case of fluid compressibility and non-equal phase density, the total density has to be defined. Note that the total density is reported in molar densities (mol/m^3). The compositional profile for each of the models and cases is shown in figure 4.21. It is obvious that the proposed upscaling approach demonstrates remarkably good performance. The L_2 and $Linf$ norms of the production component nC_{10} are reported in figure 4.22 and remain below 1.5 and 0.5 for all three cases.

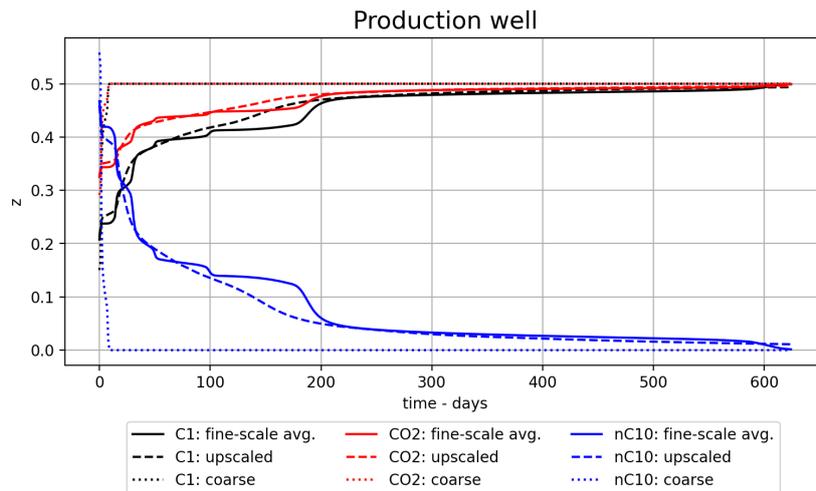
The effect of including vertical permeability is demonstrated in figure 4.20. When vertical permeability is added to the fine-scale model, the average compositional path of the downstream coarse-grid blocks be-



(a) Case 1: injection of C_1 at 95mol%.



(b) Case 2: injection of CO_2 at 95mol%.



(c) Case 3: injection of C_1 and CO_2 at 50mol% each.

Figure 4.14: Evolution of the molar composition at the production well for the coarse-scale model and the upscaled model.

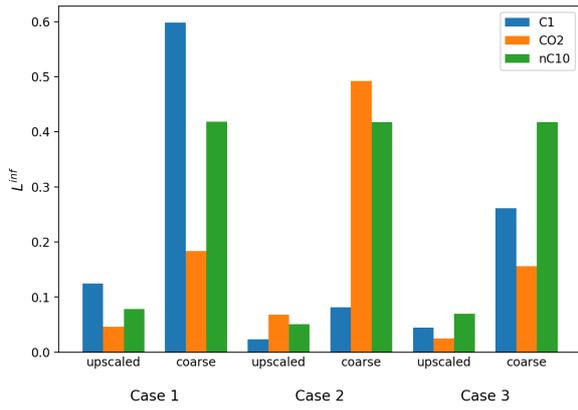


Figure 4.15: L_{∞} norm of $z_i(t)$ at the production well ($n_x = 20$) for each of the three cases.

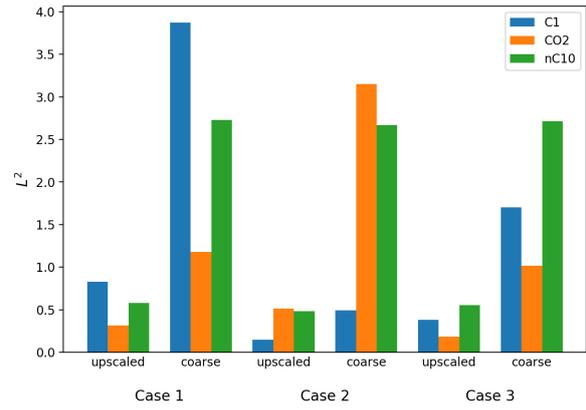


Figure 4.16: L_2 -norm of $z_i(t)$ at the production well ($n_x = 20$) for each of the three cases.

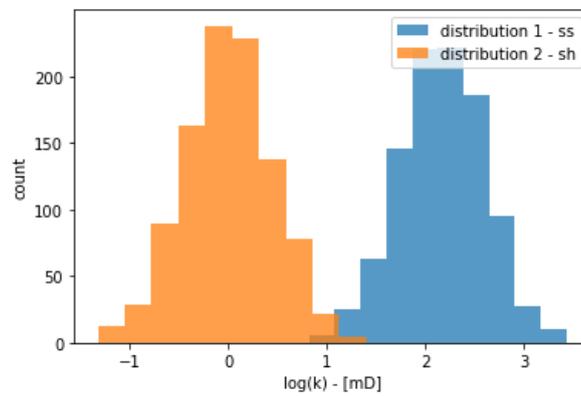
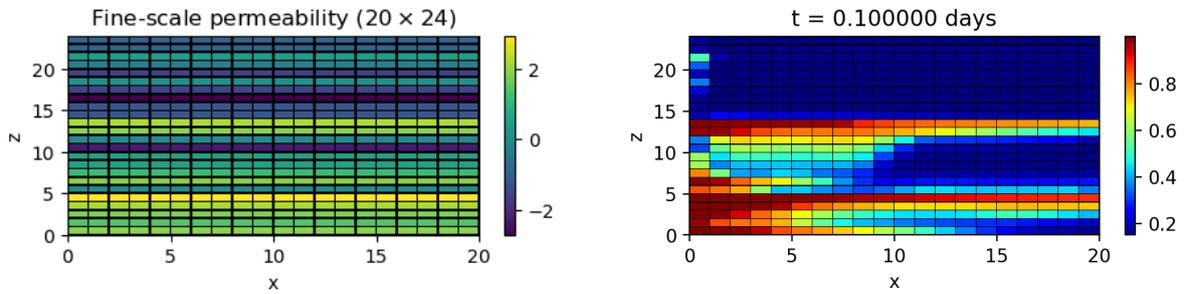


Figure 4.17: $\text{Log}_{10}(k)$ of the distribution of permeabilities.



(a) Log_{10} of the permeability distribution of the fine-scale model.

(b) Fine-scale solution of the overall composition of C_1 at $t = 0.1 \text{ days}$.

Figure 4.18: Fine-scale model with added physics and real permeabilities.

comes straighter and less variable. This is because the vertical communication of layers enables the spreading of the shocks. Consequently, the displacement becomes increasingly uniform. Therefore, data points gathered from the test-bench fall closer to the real compositional path of the downstream coarse grid blocks. This is advantageous for the construction of a quality upscaled operator space.

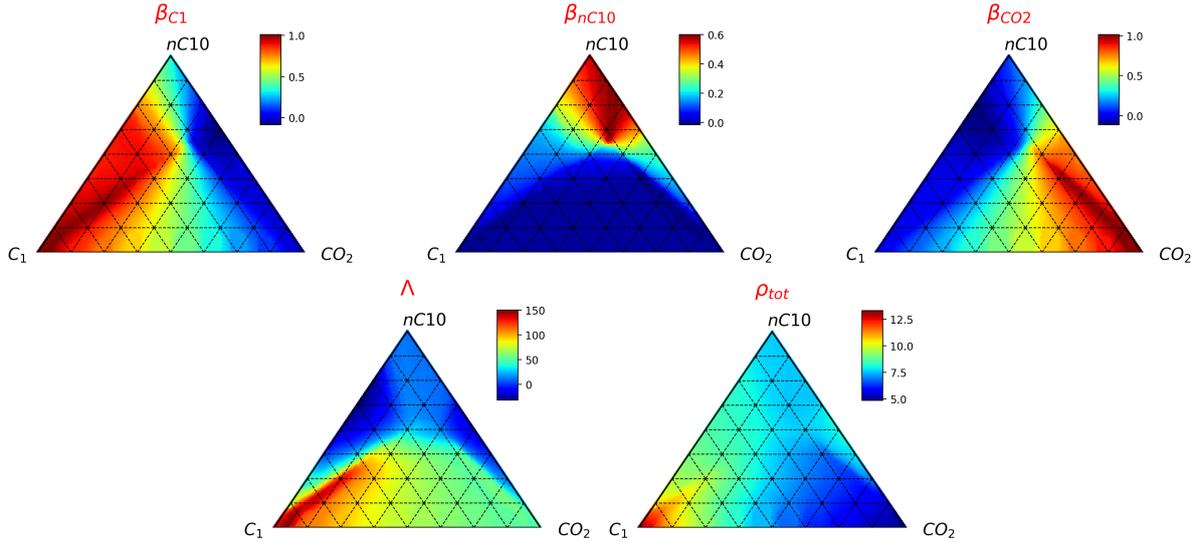


Figure 4.19: Operator space for the upscaling of the well model. The total density is defined in addition to the flux and total mobility operators. The total density is reported in mol/m^3 .

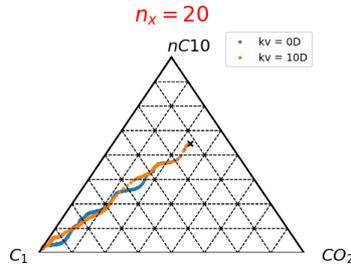


Figure 4.20: The inclusion of vertical permeability has a straightening effect on the compositional path of downstream coarse grid blocks.

4.2.4. Neural approximation of operator space

The operator space for a three-layer model is parameterized according to layer permeabilities and injection composition. Three types of layers are defined by their permeability, namely 2.7, 50 and 0.4mDarcy, and are treated as an analogue of geological facies or hydrological flow units. A data set is created from these permeabilities by taking all possible combinations of the layers and defining a total of 27 models. Note that in this approach some models contain layers with identical permeability. Furthermore, layer models with identical upscaled absolute permeability are filtered from the data set. The resulting data set for training and testing contains 11 models.

Three neural networks are used to learn and approximate the upscaled operator space of the flux operator per component. The complete data set is split into models used for training and testing the neural networks. The training set consists of 10 models and for testing the model containing 2.7, 50 and 0.4mDarcy layer permeabilities from top to bottom are used. Here, the validation data and test data are identical as neural networks are used as black box functions and hyperparameter tuning is not carried out. Therefore the risk of inadvertently overfitting the test data is negated. The complete training set has a design matrix consisting of 29,228 samples and 6 features. These features are the permeabilities per layer, the mean of the layer permeabilities (upscaled absolute permeability) and the overall molar fraction of components C_1 and CO_2 . Histograms of each of the input features and data labels are shown in figures A.4 and A.5 of the appendix.

The neural networks are trained with early stopping where training is halted if the loss of the test model no longer decreases. This prevents overfitting of the training data and poor generalization. Figure 4.23 plots the loss of the training and test sets. The trained neural network is used to approximate the upscaled β operator for the test model. The resulting compositional profile is shown in figure 4.24. The results are compared to the coarse-scale model and an upscaled model that uses an interpolation function to define its operator space. The interpolation is applied directly to the test data points. Figure 4.28 summarizes the results in terms of

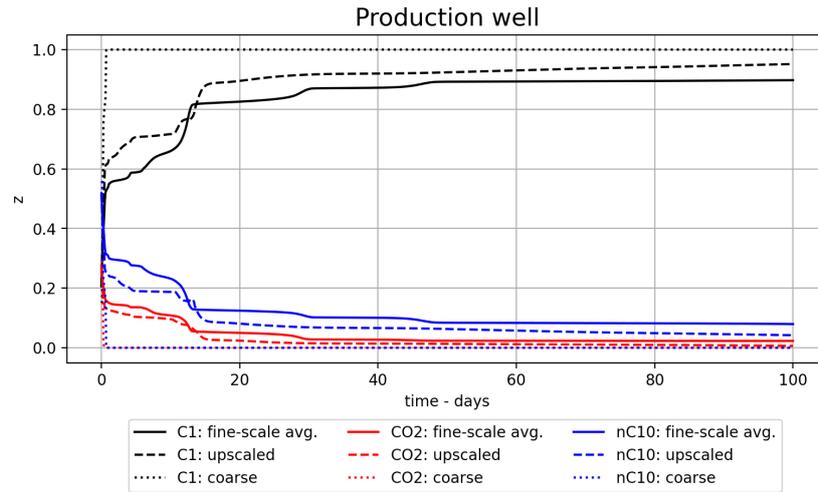
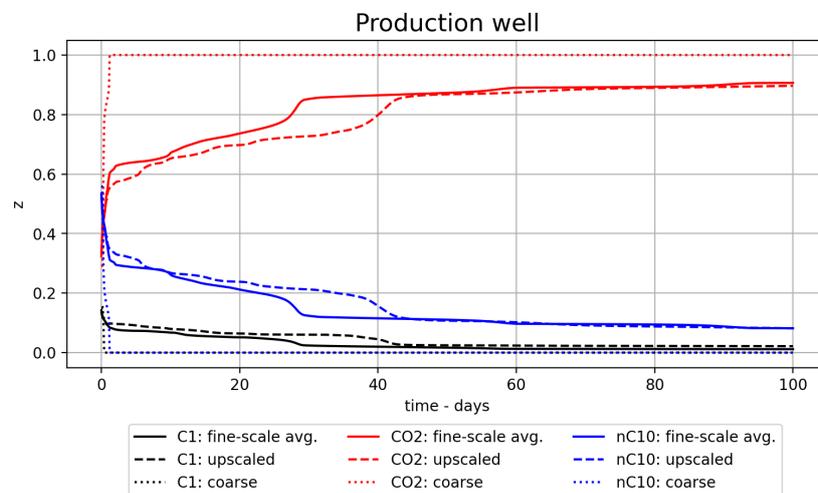
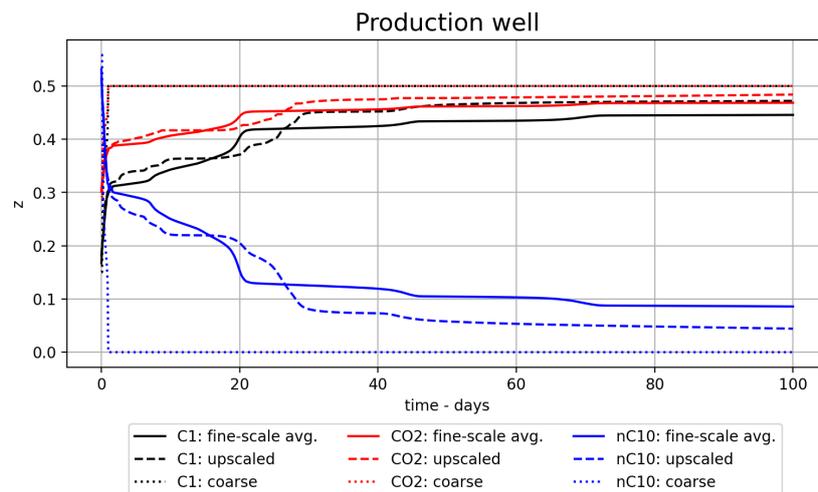
(a) Case 1: injection of C_1 at $100\text{mol}\%$.(b) Case 2: injection of CO_2 at $100\text{mol}\%$.(c) Case 3: injection of C_1 and CO_2 at $50\text{mol}\%$ each.

Figure 4.21: Evolution of the composition profile for the three cases with added physics (fluid compressibility and gravity) real well permeabilities and non-zero vertical permeability.

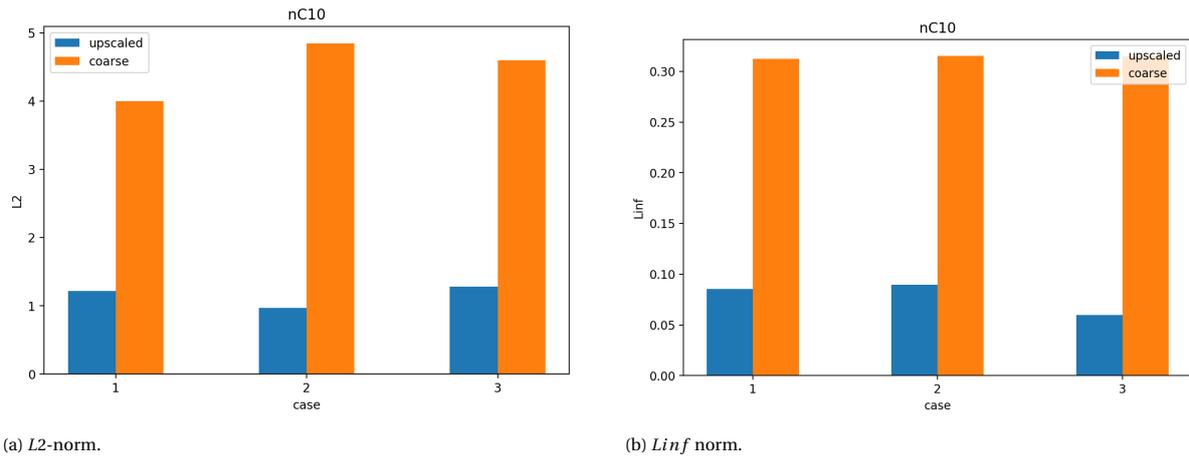


Figure 4.22: Error of $z_{CO_2}(t)$ at the production well for the well model with added physics (fluid compressibility and gravity).

error. The upscaled model outperforms the coarse model but does not make an equally good approximation of the operator space as the interpolated model. The neural approximated operator space is shown in figure 4.25 and the interpolated operator space is shown in figure 4.26. The neural networks make an accurate qualitative approximation of the operator space. The differences between the two approximations of the operator spaces are shown in figure 4.27. Note that both upscaled models use the same total mobility operator which is interpolated from data and shown in figure 4.29.

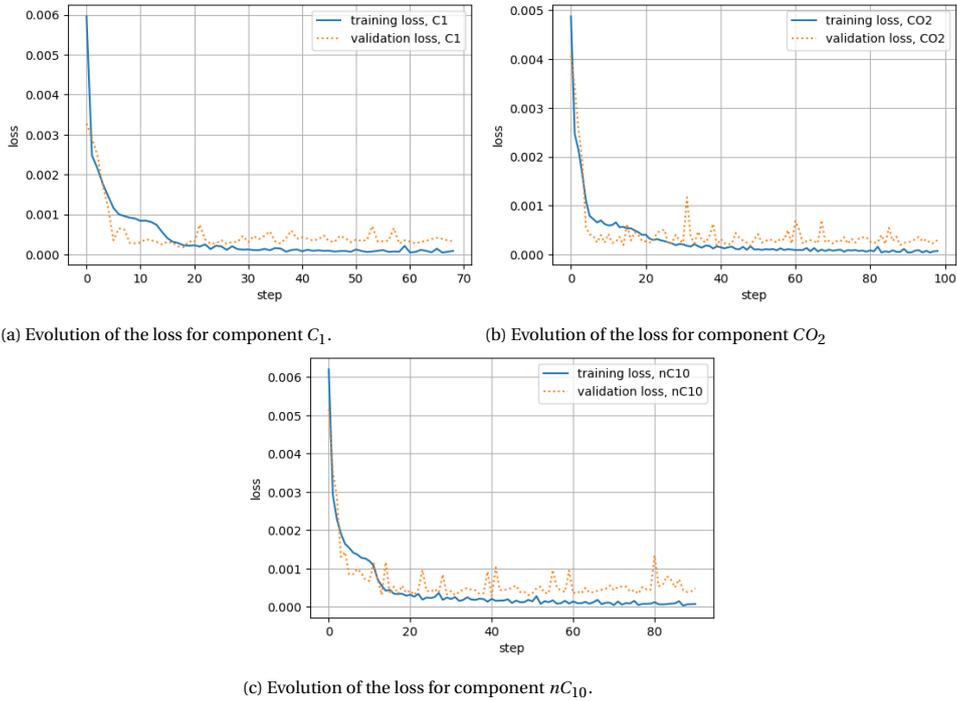
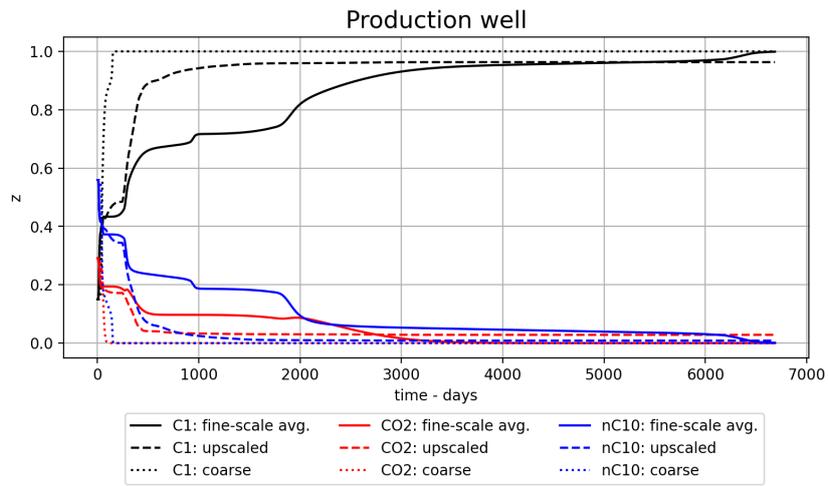
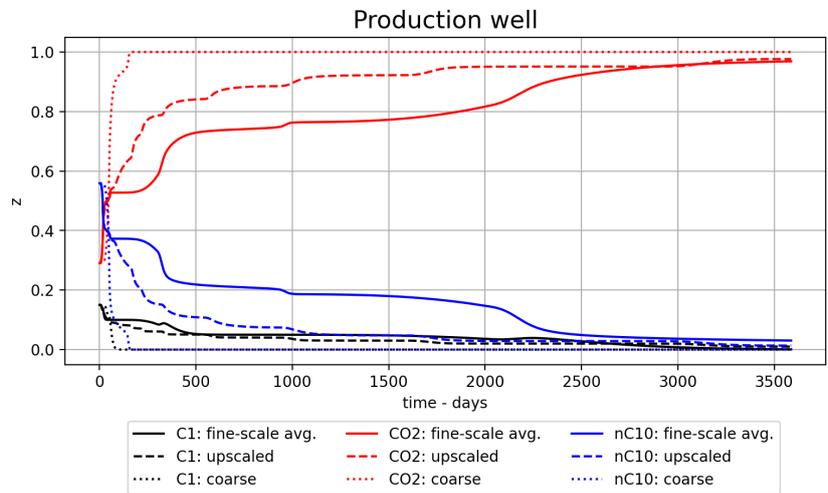


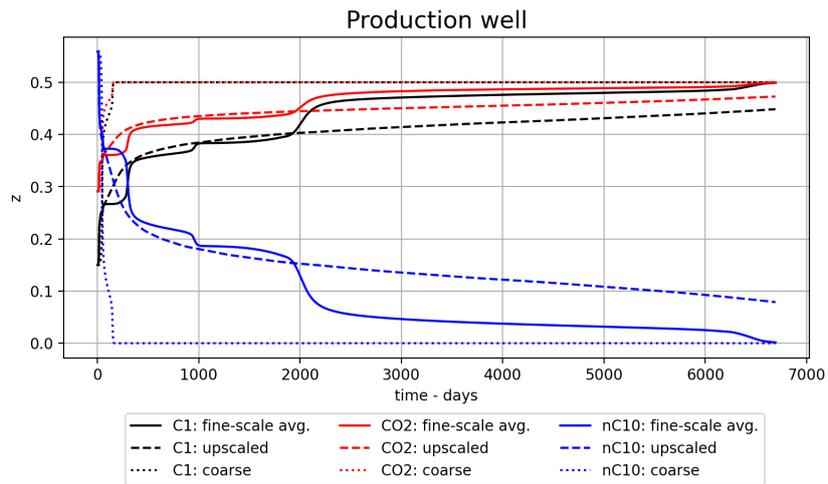
Figure 4.23: Evolution of the loss curves for each of the neural networks used to approximate the flux operators for upscaling.



(a) Case 1: injection of C_1 at 100%.



(b) Case 2: injection of CO_2 at 100%.



(c) Case 3: injection of C_1 and C_2 at 50% each.

Figure 4.24: Evolution of the composition profile for the three cases with operator space approximated by neural networks.

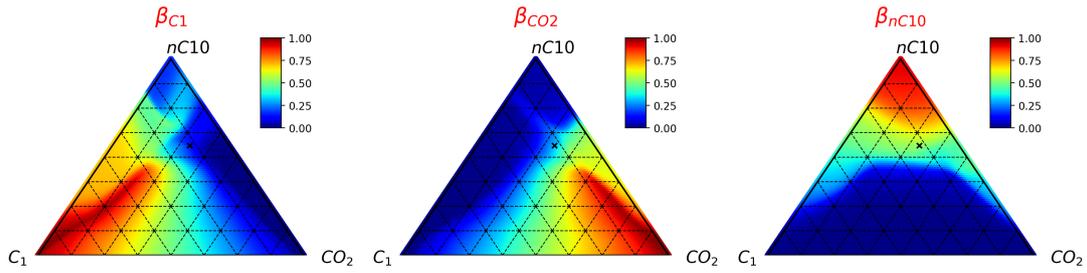


Figure 4.25: Operator space approximated by neural networks. Note that the space above the initial composition point (marked with an x) is non-sensical as no data is gathered in this area.

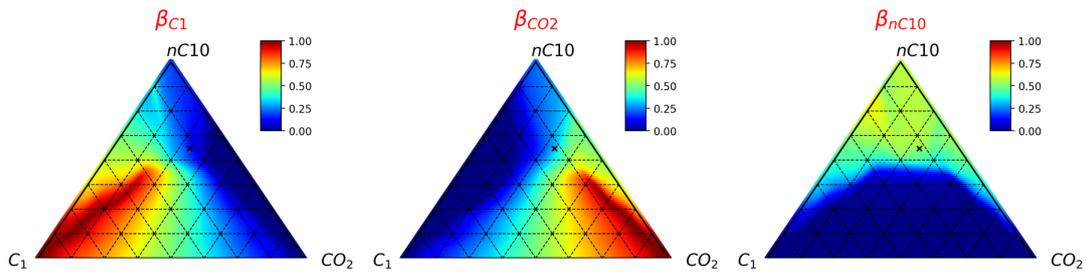


Figure 4.26: Interpolated operator space. Note that the space above the initial composition point (marked with an x) is non-sensical as no data is gathered in this area.

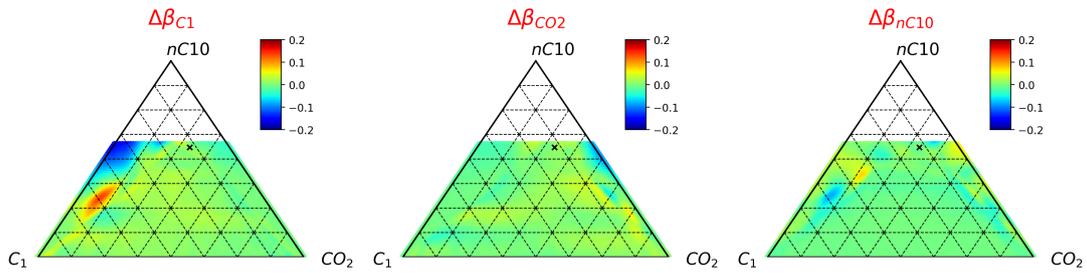
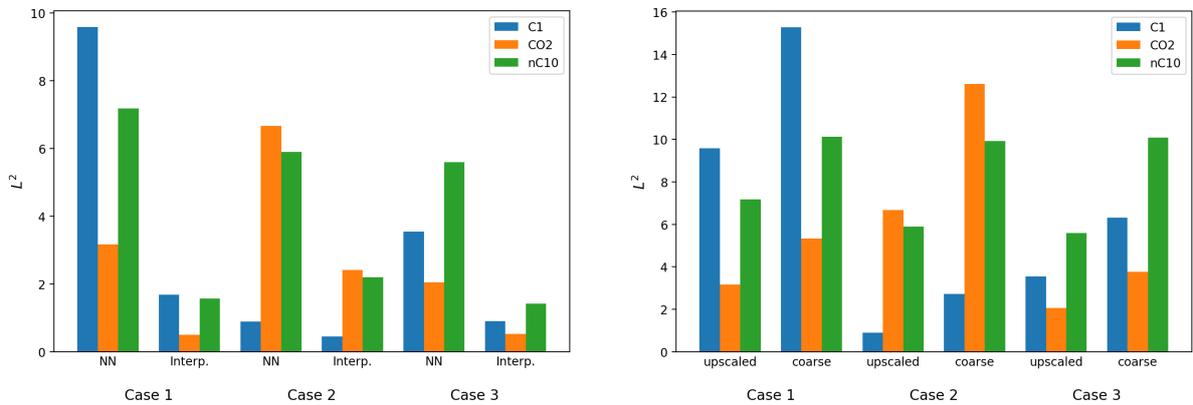


Figure 4.27: Difference in operator space between neural approximated and interpolated operator space. The greatest differences are located at the boundary of single- and two-phase zones.



(a) The L_2 norm of the upscaled model with a neural network is greater than that of the upscaled model using interpolated operator space (Interp.)

(b) The L_2 norm of the upscaled model with neural approximated operator space is smaller than that of the coarse models.

Figure 4.28: L_2 -norm of the upscaled and coarse models.

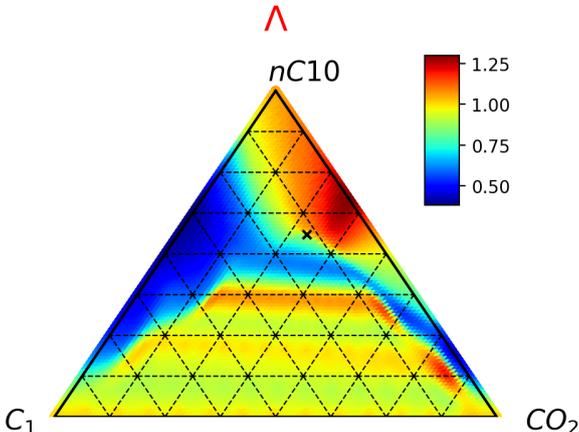


Figure 4.29: For both the neural network approximated and interpolated models the same total mobility operator is used. The total mobility operator is interpolated directly from data points.

5

Conclusion

Compositional simulation can be prohibitively expensive for optimization and history-matching problems where a large number of model runs are required. In this report, two methods for efficient compositional modeling are used namely, proxy modeling and upscaling. These methods are applied to a vaporizing gas injection problem within the operator-based linearization framework of the Delft Advanced Research Terra Simulator (DARTS).

In the first method, a pseudo-binary proxy model based on the Multiscale Reconstruction in Physics is extended to include fluid compressibility. Thereafter, neural networks' ability to function as universal function approximators is leveraged to learn the physical space of the full compositional simulation in the reduced dimension (1D) and improve the proxy model for multi-dimensional simulation. The approach uses an adaptation of PIML by subjecting the operator space of the proxy model to neural networks and including the Lebesgue integration of the forward solution in the loss function. Improvements in the estimation of breakthrough times are realized for the bottom and top fifteen layers of the SPE10 model. This method can be further extended for developing a generic physical proxy model with lower degrees of freedom trained at data points generated by a high-fidelity physical model or based on real-world observations.

In the second method, the upscaled operator space is defined via rigorous upscaling. In this method, the upscaled operator space is defined exhaustively in three-dimensional compositional space and accounts for non-equilibrium and dynamic effects on the coarse scale. The upscaling procedure is carried out on a limited, characteristic portion of the fine-scale model and is applicable to models with variable injection compositions without having to rerun the fine-scale model. The method is found to work for models with n_z -number of layers, and added physics (gravity and compressibility). In addition, a neural network is trained to approximate the operator space of a three-layer model in function of the injection composition and the permeability distribution. The resulting neural network makes a qualitatively good approximation of the operator space, however performance is diminished.

5.1. Discussion/recomendations

In both methods, a constant K-value equilibrium model is used. In order to capture the effect of miscibility it is necessary to implement the proxy model with an equation of state thermodynamic model. This has already been done in previous work by Chen and Voskov, 2020 and is readily implementable in our machine-learning framework. Furthermore, Iranshahr et al., 2014 utilize a Peng-Robinson equation of state in their coarse-scale compositional framework.

The upscaling procedure maintains the same conservation equation on the coarse and fine model scales. Yet, alternative methods include a diffusive term in the coarse-scale conservation equation (Efendiev and Durlofsky, 2003) or use volume averaging of higher moments to define upscaled functions (Durlofsky, 1998).

Rigorous upscaling essentially requires that a portion, albeit limited in size, of the fine-scale model is run several times for various injection conditions. An off-the-cuff timing of the upscaling procedure versus the fine-scale model of the demonstrative example shown in chapter 4.2.1 has the following compute times: 164s for the upscaling procedure versus 68s (case1), 44s(case2) and 56s(case3) for the fine-scale solution. Note that these times depend heavily on the sampling strategy used.

Three-dimensional simulation is the standard for reservoir simulation. Here models are limited to two

dimensions however gains in proxy model performance extend to three dimensions and the upscaling procedure described can be applied for three dimensions as well.

Neural networks are used as black box functions and no hyperparamter tuning of the neural architecture is carried out. Therefore, it is recommended that the method is tested for different neural architectures to see if the solution of the ML trained proxy can further be improved.

Bibliography

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems.
- Barker, J. W., & Fayers, F. J. (1994). Transport Coefficients for Compositional Simulation with Coarse Grids in Heterogeneous Media. *SPE Advanced Technology Series*, 2(02), 103–112. <https://doi.org/10.2118/22591-PA>
- Barker, J. W., & Thibeau, S. (1997). A Critical Review of the Use of Pseudorelative Permeabilities for Upscaling. *SPE Reservoir Engineering*, 12(02), 138–143. <https://doi.org/10.2118/35491-PA>
- Blechschmidt, J., & Ernst, O. G. (2021). Three ways to solve partial differential equations with neural networks — A review. *GAMM-Mitteilungen*, 44(2). <https://doi.org/10.1002/gamm.202100006>
- Chen, Y., & Voskov, D. (2020). Optimization of CO₂ injection using multi-scale reconstruction of composition transport. *Computational Geosciences*, 24(2), 819–835. <https://doi.org/10.1007/s10596-019-09841-8>
- Christie, M. A., & Blunt, M. J. (2001). Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *SPE Reservoir Evaluation & Engineering*, 4(04), 308–317. <https://doi.org/10.2118/72469-PA>
- Christie, M. A., & Clifford, P. J. (1998). Fast Procedure for Upscaling Compositional Simulation. *SPE Journal*, 3(03), 272–278. <https://doi.org/10.2118/50992-PA>
- Christie, M., Mansfield, M., King, P., Barker, J., & Culverwell, I. (1995). A Renormalisation-Based Upscaling Technique for WAG Floods in Heterogeneous Reservoirs. *All Days*. <https://doi.org/10.2118/29127-MS>
- Darman, N., Pickup, G., & Sorbie, K. (2002). A Comparison of Two-Phase Dynamic Upscaling Methods Based on Fluid Potentials. *Computational Geosciences*, 6(1), 5–27. <https://doi.org/10.1023/A:1016572911992>
- Durlofsky, L. J. (1998). Coarse scale models of two phase flow in heterogeneous reservoirs: volume averaged equations and their relationship to existing upscaling techniques. *Computational Geosciences*, 2(2), 73–92. <https://doi.org/10.1023/A:1011593901771>
- Efendiev, Y., & Durlofsky, L. J. (2003). A Generalized Convection-Diffusion Model for Subgrid Transport in Porous Media. *Multiscale Modeling & Simulation*, 1(3), 504–526. <https://doi.org/10.1137/S1540345902413693>
- Fuks, O., & Tchelepi, H. A. (2020). LIMITATIONS OF PHYSICS INFORMED MACHINE LEARNING FOR NON-LINEAR TWO-PHASE TRANSPORT IN POROUS MEDIA. *Journal of Machine Learning for Modeling and Computing*, 1(1), 19–37. <https://doi.org/10.1615/JMachLearnModelComput.2020033905>
- Ganapathy, C., & Voskov, D. (2018). Multiscale reconstruction in physics for compositional simulation. *Journal of Computational Physics*, 375, 747–762. <https://doi.org/10.1016/J.JCP.2018.08.046>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Iranshahr, A., Chen, Y., & Voskov, D. (2014). A coarse-scale compositional model. *Computational Geosciences*, 18(5), 797–815. <https://doi.org/10.1007/s10596-014-9427-x>
- Jessen, K., & Moghadam, R. R. (2009). A Flow Based Lumping Approach for Compositional Reservoir Simulation. *All Days*. <https://doi.org/10.2118/119160-MS>
- Jin, X., Cai, S., Li, H., & Karniadakis, G. E. (2021). NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 426, 109951. <https://doi.org/10.1016/j.jcp.2020.109951>
- Kyte, J., & Berry, D. (1975). New Pseudo Functions To Control Numerical Dispersion. *Society of Petroleum Engineers Journal*, 15(04), 269–276. <https://doi.org/10.2118/5105-PA>
- Lyu, X., Voskov, D., Tang, J., & Rossen, W. R. (2021). Simulation of Foam Enhanced-Oil-Recovery Processes Using Operator-Based Linearization Approach. *SPE Journal*, 26(04), 2287–2304. <https://doi.org/10.2118/205399-PA>
- McCain, D. W. (1990). *The Properties of Petroleum fluids* (2nd ed.). PennWell Publishing Company.
- Orr, F. (2007). *Theory of Gas Injection Processes*. Tie-Line Publications.

- Pan, H., & Tchelepi, H. A. (2011). Compositional Flow Simulation Using Reduced-Variables and Stability-Analysis Bypassing. All Days. <https://doi.org/10.2118/142189-MS>
- Raissi, M., Perdikaris, P., & Karniadakis, G. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378, 686–707. <https://doi.org/10.1016/j.jcp.2018.10.045>
- Raissi, M., Yazdani, A., & Karniadakis, G. E. (2018). Hidden Fluid Mechanics: A Navier-Stokes Informed Deep Learning Framework for Assimilating Flow Visualization Data.
- Rasmussen, C. P., Krejbjerg, K., Michelsen, M. L., & Bjurstram, K. E. (2003). Increasing Computational Speed of Flash Calculations with Applications for Compositional, Transient Simulations. All Days. <https://doi.org/10.2118/84181-MS>
- Rastegar, R., & Jessen, K. (2009). Lumping and Delumping for Integrated Compositional Modeling. All Days. <https://doi.org/10.2118/125017-MS>
- Salehi, A., Voskovand, D. V., & Tchelepi, H. A. (2013). Thermodynamically Consistent Transport Coefficients for Upscaling of Compositional Processes. All Days. <https://doi.org/10.2118/163576-MS>
- Salehi, A., Voskov, D. V., & Tchelepi, H. A. (2019). *K*-Values-Based Upscaling of Compositional Simulation. *SPE Journal*, 24(02), 579–595. <https://doi.org/10.2118/182725-PA>
- Shirangi, M. G., & Durlofsky, L. J. (2015). Closed-Loop Field Development Under Uncertainty by Use of Optimization With Sample Validation. *SPE Journal*, 20(05), 908–922. <https://doi.org/10.2118/173219-PA>
- Stewart, R., & Ermon, S. (2016). Label-Free Supervision of Neural Networks with Physics and Domain Knowledge.
- Stone, H. (1991). Rigorous Black Oil Pseudo Functions. SPE Symposium on Reservoir Simulation. <https://doi.org/10.2118/21207-MS>
- Tang, D., & Zick, A. (1993). A New Limited Compositional Reservoir Simulator. SPE Symposium on Reservoir Simulation. <https://doi.org/10.2118/25255-MS>
- Voskov, D. V., & Entov, V. M. (2001). Problem of Oil Displacement by Gas Mixtures. *Fluid Dynamics*, 36(2), 269–278. <https://doi.org/10.1023/A:1019290202671>
- Voskov, D., & Tchelepi, H. A. (2009). Compositional Space Parameterization: Theory and Application for Immiscible Displacements. *SPE Journal*, 14(03), 431–440. <https://doi.org/10.2118/106029-PA>
- Voskov, D., & Tchelepi, H. A. (2008). Compositional Space Parametrization for Miscible Displacement Simulation. *Transport in Porous Media*, 75(1), 111–128. <https://doi.org/10.1007/s11242-008-9212-1>
- Voskov, D. (2017). Operator-based linearization approach for modeling of multiphase multi-component flow in porous media. *Journal of Computational Physics*, 337, 275–288. <https://doi.org/10.1016/J.JCP.2017.02.041>
- Wang, Y., Voskov, D., Khait, M., & Bruhn, D. (2020). An efficient numerical simulator for geothermal simulation: A benchmark study. *Applied Energy*, 264, 114693. <https://doi.org/10.1016/j.apenergy.2020.114693>
- Zaydullin, R., Voskov, D. V., & Tchelepi, H. A. (2013). Nonlinear Formulation Based on an Equation-of-State Free Method for Compositional Flow Simulation. *SPE Journal*, 18(02), 264–273. <https://doi.org/10.2118/146989-PA>

A

Appendix

A.1. Pseudo-binary model

A.1.1. Hydrodynamic parameters of pseudo-binary model

Table A.1: Hydrodynamic parameters

	Oil	Gas
Porosity	0.3	0.3
Residual saturation	0	0
End points $k_r(S)$	1.0	1.0
Saturation exponents	2.0	2.0
Viscosity (cP)	0.5	0.1
Density (kg/m^3)	600	200
Compressibility ($1/bar$)	10^{-5}	10^{-3}

Table A.2: Molar weights of each component in g/mol

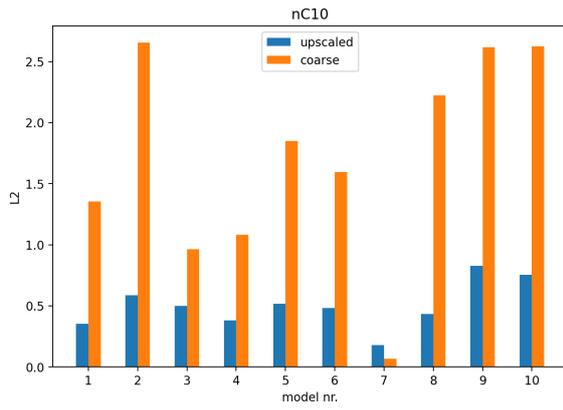
C1	CO2	nC4	nC10
16.043	44.010	58.123	114.520

A.2. Rigorous upscaling

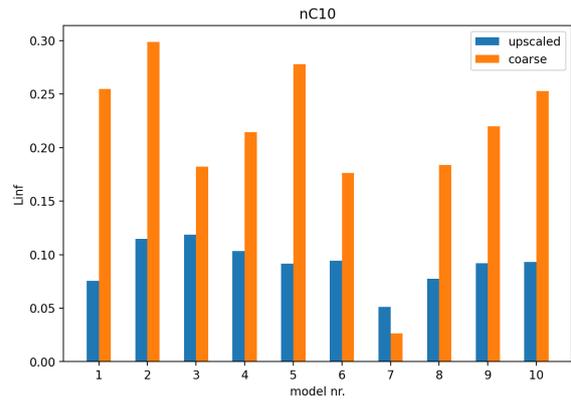
Multiphase upscaling is carried out rigorously in the full compositional space.

A.2.1. Sensitivity to number of layers

The error statistics of the production component are reported for n_z -layered models.

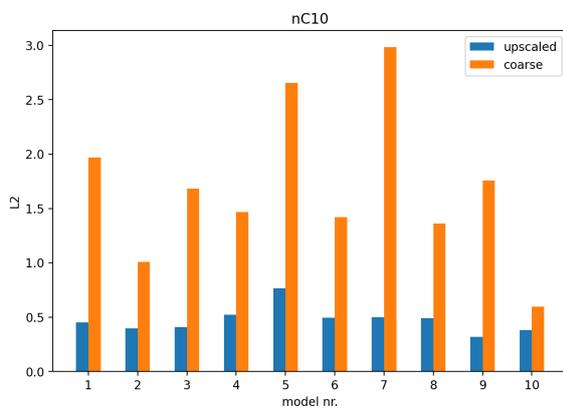


(a) Second norm.

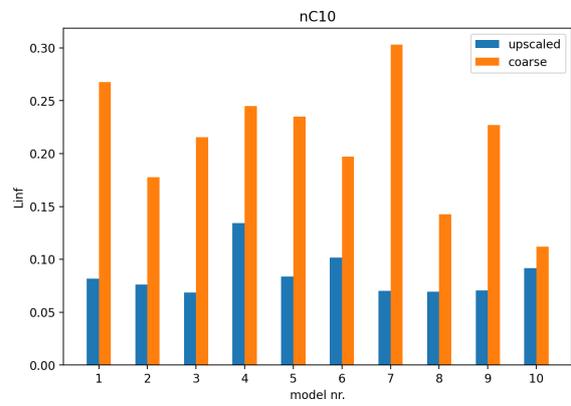


(b) Infinite norm.

Figure A.1: Error of $z_{CO_2}(t)$ at the production well for four-layer models.

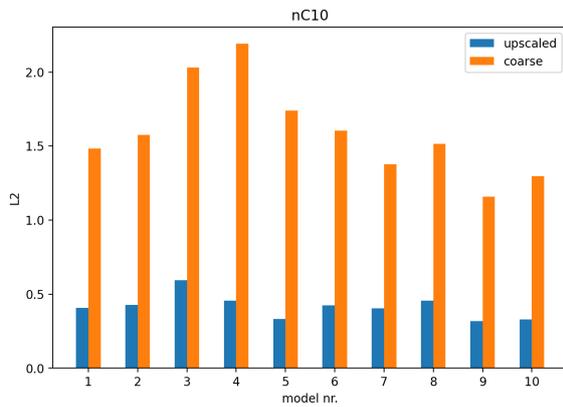


(a) Second norm.

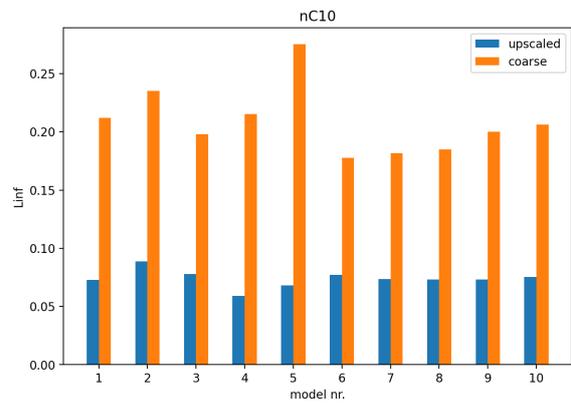


(b) Infinite norm.

Figure A.2: Error of $z_{CO_2}(t)$ at the production well for eight-layer models.



(a) Second norm.



(b) Infinite norm.

Figure A.3: Error of $z_{CO_2}(t)$ at the production well for sixteen-layer models.

A.2.2. Histograms of training and test data

Training and test data set of a three-layer model.

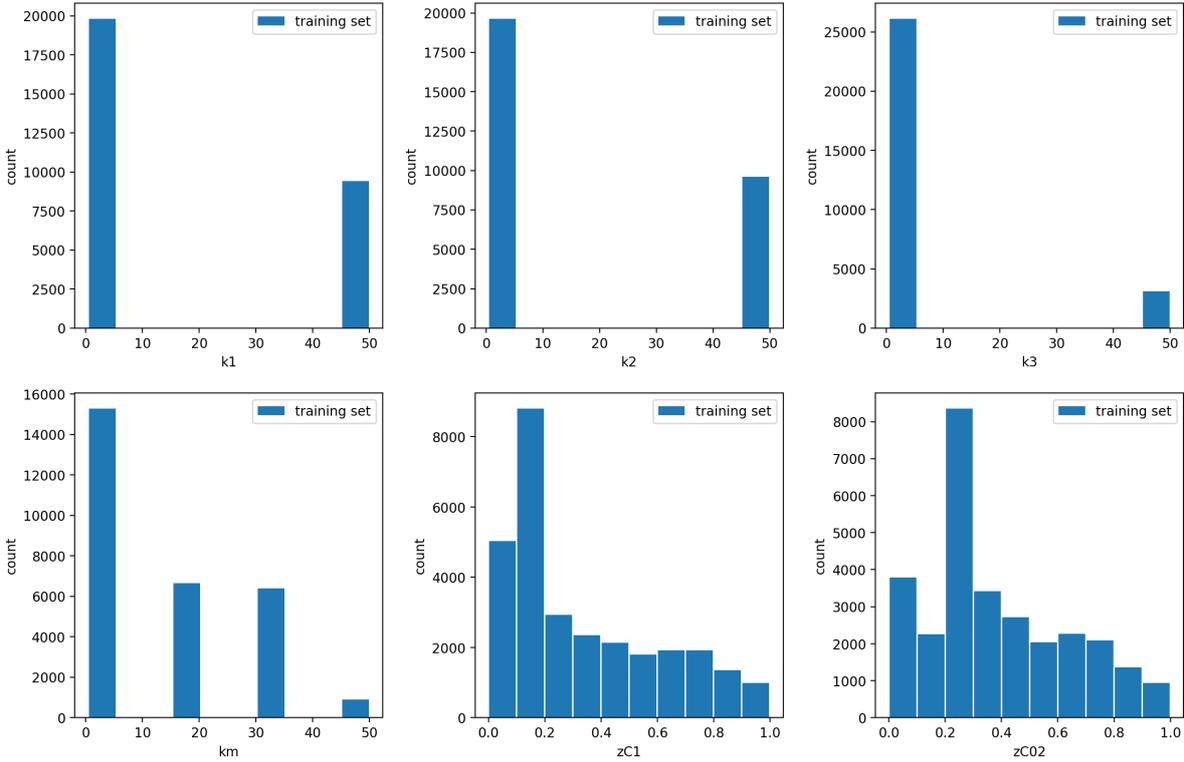


Figure A.4: Histogram of the input features of the neural network.

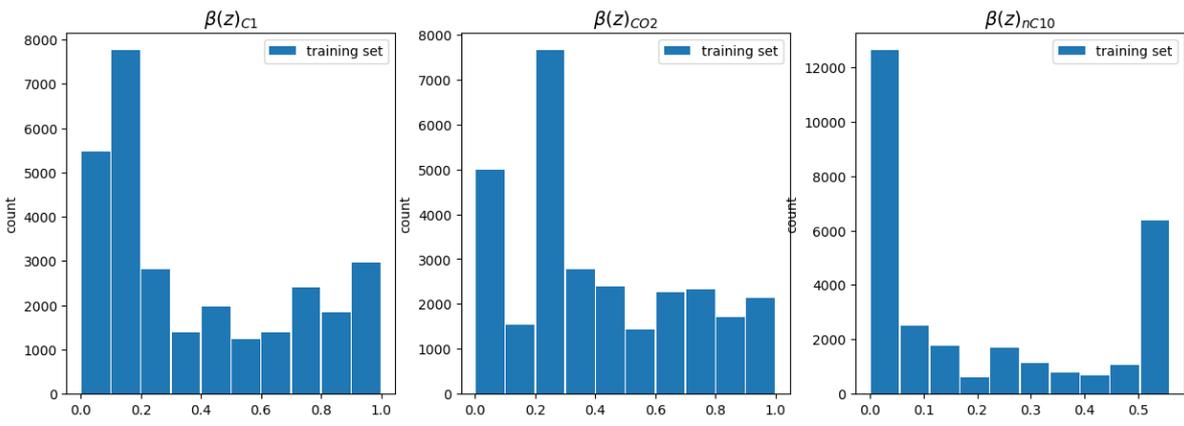


Figure A.5: Histogram of the output features of the neural network.

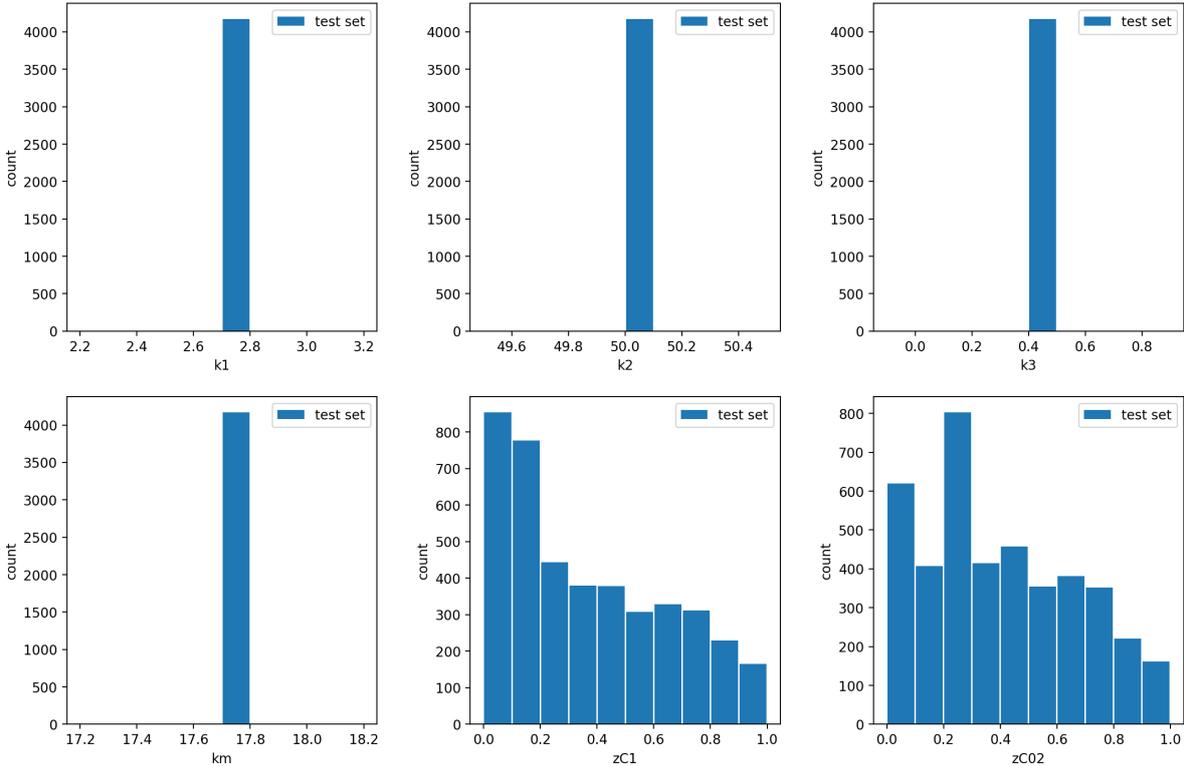


Figure A.6: Histogram of the input features of the neural network.

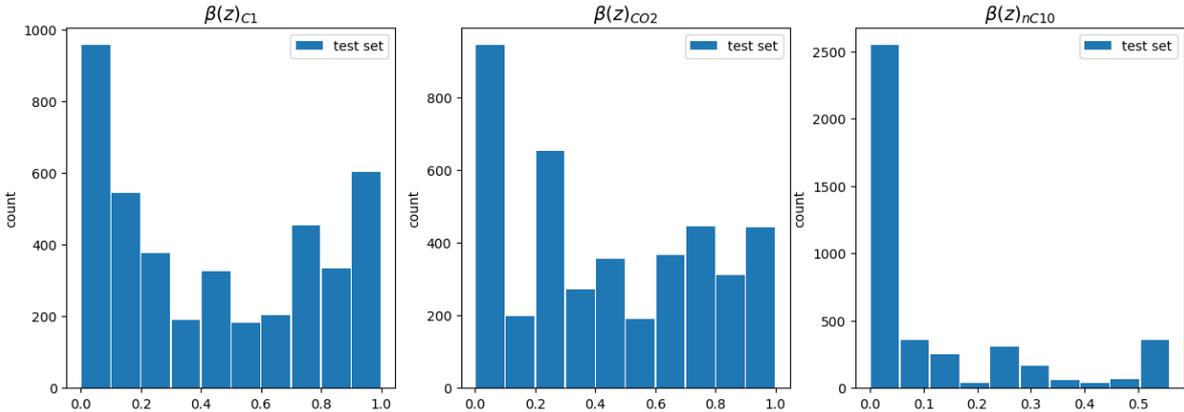


Figure A.7: Histogram of the output features of the neural network.