TU Delft Delft University of Technology

Master of Science Thesis

Submitted in Partial Fulfillment of MSc Degree

Physics-Based Data-Driven Model for Short-Term Production Forecast

Author:
Arturs Blinovs

Supervisor:
Dr. Denis Voskov

February 3, 2021

Physics-Based Data-Driven Model for Short-Term Production Forecast

by

Arturs Blinovs

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on Wednesday August 28, 2019 at 14:00 PM

Student number:	4748565	
Project duration:	November 1, 2018 – August 10, 2019	
Thesis Advisor:	Dr. Denis Voskov	TU Delft
Thesis committee:	Dr. Denis Voskov	TU Delft
	Dr. Olwijn Leeuwenburgh	TU Delft/TNO
	Dr. Joep Storms	TU Delft
	Mark Khait	TU Delft
	Dr. Vitaly Elichev	Wintershall Dea

Delft University of Technology

Acknowledgements

My university journey is gradually reaching its logical end and it would be naive not to acknowledge those who supported me thought this joyful adventure. I am very grateful for the chance to meet and interact with such interesting, intelligent and well-mannered people. That was a driving force, support and motivation thought my academic odyssey.

First of all, I would like to say thank you to my advisor *Dr.D.Voskov*, for the encouragement and support during my thesis and moreover, for sharing his wide academic experience with me. Big words of gratitude I would like to dedicate to Ph.D. student *M.Khait*, as he was the person who instilled in me a love for coding and shared good coding practices. I would like to thank my parents *Andrej* and *Galina*, as without their help and endless support, my education and formation as an independent unit of society would not be possible. Also, my older brother *Anton* for the encouragement he gave me during my life abroad. Give thanks to the lovely friends from the House of Culture "At guy's place" for the countless moments of fun and selflessness, as the time we shared was probably the most remarkable in my life.

It was a pleasant journey of building an academic foundation for my professional career. I am extremely excited about how life will turn and to what part of the world it would bring me as I am sure many more achievements are waiting for me to be accomplished. Therefore, for those who want to stay up with them, I would recommend scanning a QR code below. God[1] bless everyone!

Arturs Blinovs
Delft, August 2019

[1] "Nietzsche bless you!" for the atheistic readers

Abstract

Before performing a field production forecast, an inverse problem has to be solved. Resulting in an ensemble of models that include the integration of real data with a complex physical and geological data describing subsurface processes. For large models, this approach can be very time and computationally expensive, therefore we propose an alternative approach for reservoir forecasting. In this work, we develop a physics-based data-driven model that purely relies on production data of the field and does not require any in-depth knowledge of the reservoir geology and physics.

In the proposed approach, we utilize Delft Advanced Reservoir Terra Simulator (DARTS) as a base for our reservoir simulations. DARTS uses an Operator-Based Linearization technique for the approximation of exact physics. It allows us to encounter a more realistic interpretation of physics and is computationally efficient. The physics-based data-driven approach uses sequential regression to the data to increase the fidelity of the model forecast and encounter any significant changes in reservoir dynamics and physics over its history.

The model was examined and validated for synthetic and real field production models. We demonstrate that the developed approach is capable of providing accurate and reliable production forecast on a daily basis.

Contents

List of Tables

List of Figures

1

Introduction

Computer technologies are progressing rapidly, computational capacities that are available nowadays provide an opportunity for many industries to perform more complex numerical simulations of high-resolution three-dimensional (3D), geo-cellular computer models. Prediction from those models is an important factor governing efficient reservoir management and decision making. Such models describe complex geological features through a set of grid blocks and associated rock and fluid properties. A high-resolution computer model can exceed a few million blocks and take hours or days to simulate.

It is still not computationally feasible to perform history matching or reservoir optimization efficiently at such resolution, as it involves a large number of simulations runs. Therefore, engineers develop different methods to speed up this process. Those methods fall into two categories:

- Simplification of high-fidelity geo-cellular models, or
- data-driven approaches.

Methods such as upscaling, multi-scale methods and streamline simulation fall into the first category.

Upscaling is the process of numerical homogenization, where the high-resolution model is represented as a set of coarser grid blocks with effective property replicating high-fidelity model response [10]. Multi-scale methods are somewhat similar to upscaling, where the global flow is computed on a coarse grid, while fine-scale heterogeneity is accounted for thorough basis functions [14]. The streamlined method [3] is the Eulerial-Lagrangian approach with implicit pressure explicit saturation (IMPES) time approximation. In this approach, a full 3D transport solution is translated to a set of one dimensional (1D) equations that are solved along streamlines.

All methods in the first category, require an underlying geological characterization as a basis for decreasing computational load. However, there are many cases when this information is not available. Does it mean we cannot solve optimization or history matching problems efficiently? The second category methods resolve this issue.

The data-driven method is the approach of building a proxy model with a sufficient number of degrees of freedom to accurately mimic the high-fidelity model based on its calibration to the production data. With a sufficient amount of data and reliable regression framework, data-driven models can provide an accurate forecast for the given reservoir and be easily incorporated in a software cycle used by a company.

1.1. Approaches overview

1.1.1. Reduced-order models

One of the data-driven approaches was proposed by Cardoso *et al.* [4] to decrease the computational time of the high-resolution models. The most practical application of this approach is to simulate the high-fidelity problem which is referred to as training. The pressure and saturation states are saved and assembled into matrices donated by $\mathbf{X_p}$ and $\mathbf{X_S}$. Next, singular value decomposition (SVD) is performed on those matrices, where the singular vectors represent the columns of the orthonormal reduced basis Φ, followed by truncation of Φ using energy criteria. Finally, the system of reduced Jacobian and residuals is solved. The main disadvantage of this approach is based on the fact that the high-fidelity (geological) model and exact physics of reservoir fluids is required to generate enough snapshots.

1.1.2. Statistical data-driven model

Property interpolation and estimation traditionally was a stationary process, which ignored any dynamic data. Hence, a stationary approach was applied in traditional geostatistics to interpolate data. The assumption about stationarity was purely based on the limited and usually very uncertain knowledge about the location and structure of a reservoir from static measurements. Dynamic data were often excluded. However, it provides a measure of the in-site flow process and the inter-well relation.

The first data-driven concept that incorporated dynamic data for property estimation was proposed by Jansen et al. [13]. In this approach, the cross-correlations coefficients of each well were generated and used as a tool for enhancing parameter interpolation, and moreover, cross-correlation coefficients indicated the extent of continuity. The main disadvantage of this approach is noise sensitivity. Also, due to the nature of correlation-based methods, future reservoir performance is not quantitatively predictable.

1.1.3. Capacitance-resistance model (CRM)

It was reported by Albertoni and Lake [2] that connectivity between wells only depends on geology and the relative position between them which can be described by a constant value. Also, they found that those constants are independent of the rates. The authors view a reservoir as a system that converts an input signal (injection) into an output/response (production).

Yousef *et al.* [33] improved this concept, by including capacitance (compressibility) and resistivity (transmissibility) effects into the model as well as the option to include bottom hole pressure (BHP) data. In this approach, two coefficients were determined for each well pair: one parameter (the weight) quantifies the connectivity, and another (the time constant) quantifies the degree of fluid storage between the wells. Weights describe the extent of connectivity, and time constant describes the dissipation between injector *i* and producer *j*. This approach showed good results in the identification of geological features of the reservoir. The drawback of this method is that it uses an integrated fraction flow model, which originated from an empirical correlation and can be seen only as an approximation of the reference physics.

1.1.4. Flow-network model

Lerlertpakdee *et al.* [18] proposed a Flow-Network model representing a complex 3D flow as a set of 1D finite-difference reservoir models. Each well in the original model form nodes, which are then connected with other nodes (wells) by the use of a 1D numerical simulation model. Each reservoir is defined by a set of two parameters: absolute permeability and pore volume. The coupling is imposed at the wells that are shared by more than one connection. The advantages of this approach are the construction of this model does not require a high-fidelity geological data and that it follows the physical laws of the realistic multiphase system.

However, the Flow-Network model still requires prior knowledge of the relative permeability curves.

1.1.5. Interwell-numerical-simulation model (INSIM)

INSIM [34] approach is somewhat similar to the CRM [33] and Flow-Network model [18]. In the INSIM assumptions, a reservoir is viewed as a series of units connecting well pairs but instead of discretizing those connections as in the Flow-Network model, INSIM only defines a pair of parameters for each connection. That is a significant reduction in the number of parameters compared to a set of 1D finite-difference reservoir models. Connections between well pairs should be defined in the pre-processing stage and, moreover, the prior knowledge of the relative permeability curves and fluid viscosity is required. Furthermore, INSIM uses only a single connection between injector and producer, which is not sufficient to model complex reservoir dynamics and get an accurate history match.

Later, Guo *et al.* [12] improved this model and called it INSIM-FT. He managed to fix the main problem in the INSIM, which was related to incorrect calculating of water saturation whenever a producer is converted to a water-injection. The new model uses the correct front-tracking procedure to calculate water saturation, hence the name INSIM-FT. Also, imaginary wells are added to provide more flow paths. In INSIM-FT, relative permeability is used as a matching parameter and an option to use separate relative permeability curves for different connections was also introduced.

1.1.6. Alternative approaches

Many other alternative methods rely on Artificial Intelligence (AI) [22] and data fitting [35] principles, but usually they are treated as a black-box approach. Moreover, the physical meaning of the reservoir parameters in those methods is usually lost.

1.2. Objectives and workflow

Many of the mentioned approached for history matching and production forecast either required prior knowledge of reservoir geology and fluid properties or completely refused to take into account underlying physics. Therefore, the main intention of this work is to develop a framework that is capable of producing a reasonably accurate forecast based only on historical field data, respecting underlying physical processes at the same time. This will be achieved through the utilization of the Operator-Based Linearization (OBL) technique [31] and efficient implementation in Delft Advanced Reservoir Terra Simulator (DARTS) [7].

The resulting thesis objectives are the following:

- develop a framework, which is capable to accurately match and predict reservoir rates,
- test and analyze framework on synthetic and real field data,
- understand the sensitivity of regression algorithms to its inputs and constraints, and
- utilise spatial and temporal clustering to enhance framework accuracy and efficiency.

The main steps that were taken during framework development are summarised in the workflow shown in figure 1.1.

1.3. Thesis structure

After a brief overview of approaches used in industry, all necessary theoretical background is thoroughly explained in chapter 2. The overview of the models and essential features related to model data sets is presented in chapter 3. Chapter chapter 4 begins with results obtained from real field example as this thesis is a logical continuation of the internship

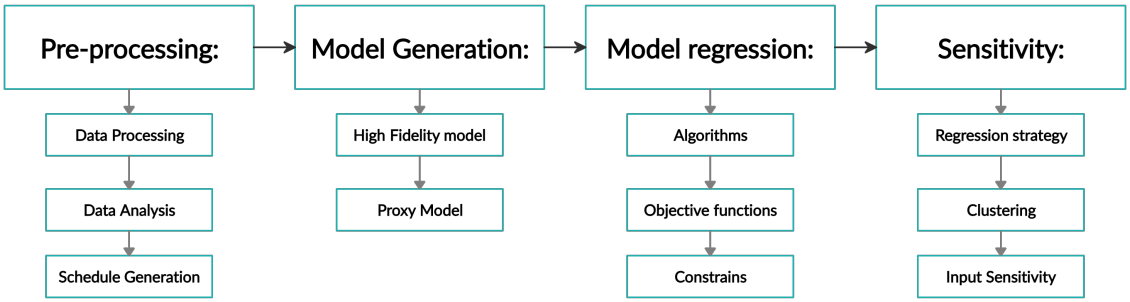


Figure 1.1: Simple visualisation of the framework development steps

project. It continues with sensitivity and quality analysis of synthetic fields: Brugge model and an ensemble of fluvial reservoir models. Conclusion and further work can be found in chapter 5 and chapter 6.

2

Theoretical background

This chapter describes all the necessary theoretical derivations used in the approach. The equations which govern fluid flow in porous media are briefly mentioned, followed by the Operator-Based linearization technique. Methods for model generation and its regression to the historical production data are also present in this chapter. Spatial and temporal clustering approaches used to accelerate and enhance model predictability are explained at the end of this chapter.

2.1. Governing equations for flow and transport in porous media

In this section, we describe the set of governing equations required for a conventional compositional numerical simulation[1], together with their discrete version and a recently proposed linearization technique [31]. The transport equations for an isothermal system containing n_c components and n_p phases can be written as:

$$\frac{\partial}{\partial t}(\phi \sum_{j=1}^{n_p} x_{cj}\rho_j S_j) + \nabla \sum_{j=1}^{n_p} x_{cj}\rho_j \mathbf{v}_j + \sum_{j=1}^{n_p} x_{cj}\rho_j q_j^{\star} = 0, \tag{2.1}$$

where phase velocity is described with a Darcy's Law:

$$\mathbf{v}_j = -\left(\mathbf{K}\frac{k_{rj}}{\mu_j}(\nabla \mathbf{P}_j - \gamma_j \nabla \mathbf{D})\right). \tag{2.2}$$

Here, ϕ - the rock porosity, x_{cj} - the mole fraction of component c in phase j, S_j - the phase saturation, ρ_j - the phase molar density, $\mathbf{v}_j$ - phase velocity, $q_j^{\star}$ - phase rate per unit volume, $\mathbf{K}$ - permeability tensor, k_{rj} - relative permeability, μ_j - phase viscosity, $\mathbf{P}_j$ - vector of pressures in phase j, γ - gravity term, $\mathbf{D}$ is the vector of depths (positive downwards). Equation 2.1 can be written in a discrete form by applying the finite-volume discretization in space and backward Euler approximation in time:

$$V\left((\phi \sum_j x_{cj}\rho_j S_j)^{n+1} + (\phi \sum_j x_{cj}\rho_j S_j)^{n}\right) - \Delta t \sum_{l \epsilon L}\left(\sum_j x_{cj}^l \rho_j^l T_j^l \Delta \psi^l\right) + \Delta t \sum_j \rho_p x_{cj} q_j = 0, \tag{2.3}$$

where ψ^l is the difference between two blocks.

The fully implicit method (FIM) time approximation has been used here that requires that flux term is defined based on the nonlinear unknowns at the new timestep $(n+1)$, which introduce

[1] Several formulations and solution techniques for compositional simulation have been proposed, which usually differ in types of nonlinear unknowns used for the solution.

nonlinearity to the system of equations. The nonlinear formulation used here is the overall molar formulation proposed in [6]. In the molar formulation, the nonlinear unknowns are pressure and overall composition, therefore the physical state ω is completely defined by these variables. A more extensive analysis of the different types of formulations and their applicability has been performed in [32].

The derivatives of all properties in eq. 2.3 with respect to nonlinear unknowns can be found by solving a multiphase flash [21] with subsequent application of the inverse theorem. Next, Jacobian and residual can be constructed that result in the conventional linearization approach. This approach is based on the Newton-Rapson method, where in each nonlinear iteration the following linear system of equations is solved:

$$\mathbf{J}(\boldsymbol{\omega}^k)(\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k) = -\mathbf{r}(\boldsymbol{\omega}^k). \tag{2.4}$$

Here, $\mathbf{J}$ is the Jacobian matrix containing derivatives with respect to primary unknowns, $\boldsymbol{\omega}$ is a vector of non-linear unknown P and z_c, k is the non-linear iteration step and $\mathbf{r}$ is the residual.

Conventional nonlinear solution approach involves evaluation and storage of all properties and its derivatives with respect to the nonlinear unknowns, which is quite challenging. A new strategy for linearization was proposed in [31]. This strategy is successfully utilised in this work.

2.1.1. Operator-Based Linearization (OBL)

Equation 2.3 can be written in a compact form as following:

$$a(\boldsymbol{\omega})(\alpha_c(\boldsymbol{\omega}) - \alpha_c(\boldsymbol{\omega}_n) - \sum_l \beta_c^l(\boldsymbol{\omega}) b^l(\boldsymbol{\omega}, \boldsymbol{\xi}) + \theta_c(\boldsymbol{\omega}, \boldsymbol{\xi}, \mathbf{u}) = 0, \tag{2.5}$$

where ω is a state dependent parameter, ξ is a space dependent parameter. All involved operators are defined as:

$$\alpha_c(\boldsymbol{\omega}) = \left(1 + c_r(p - p_{ref})\right) \sum_j x_{cj} \rho_j S_j \tag{2.6}$$

$$a(\boldsymbol{\xi}) = V\phi_0 \tag{2.7}$$

$$\beta_c(\boldsymbol{\omega}) = \sum_p x_{cj} \frac{K_{rj}}{\mu_j} \rho_j \tag{2.8}$$

$$b(\boldsymbol{\omega}, \boldsymbol{\xi}) = \Delta t T^{ab}(\boldsymbol{\xi})(p^b - p^a) \tag{2.9}$$

$$\theta_c(\boldsymbol{\omega}, \boldsymbol{\xi}, u) = \Delta t \sum_j \rho_j x_{cj} q_j(\xi, \omega, u) \tag{2.10}$$

Here, c_r is the rock compressibility, u is vector of well control variables and T^{ab} is the geometric part of transmissibility.

In this form, the nonlinear system has a simplified description in terms of physical state-dependent operators (α_c, β_c). It is also possible to specify a set of operators for a specific pressure, volume, temperature (PVT), special core analysis laboratory (SCAL) data region if needed. The values of operators are uniquely determined in the parameter-space of the problem with the set of nonlinear unknowns p and z_c. Operators are evaluated at every supporting point in the discrete parameter space during a pre-processing stage and then interpolated (using multi-linear interpolation) during simulation course. Further, this approach was modified to evaluate operators adaptively throughout a simulation run [29].

2.2. Spatial connectivity graph

To perform a numerical evaluation or reservoir simulation we have to go from a continues to a discrete domain. Therefore, discretization of the models, variables, and equations should be performed into discrete counterparts. The geometrical discretization of the reservoir is based on the control volume partitioning [15]. Spatial connectivity graph is a discrete representation of the reservoir model, in terms of the connections between control volumes and associated constants. The discretised model is defined using boundaries which are gridded using hierarchical approach: a volume (convex polyhedra) is bounded by a set of surfaces (convex polygons), a surface is bounded by a series of curves (segments), and a curve is bounded by two endpoints (nodes).

For model gridding, an automatic open-source meshing software package "GMSH" was utilised [11]. It implements several algorithms for mesh optimisation and post-processing [20]. The simple example of meshing is a triangulation with the help of a Voronoï diagram.

First, lets consider a Voronoï cell in figure 2.1(a) of a point S_i, which is a locus of point in a 2D space (R^2) that are closer to S_i than any other points. Then, lets introduce a set of mediators [2] between point S_i and points S_j (blue lines), which result in a Voronoï cell in fig.2.1(b). The set of a Voronoï cell is called Voronoï diagram and can be seen in figure 2.1(c). By linking centroids of Voronoï cells to all its neighbours, we get a triangulation as seen in figure 2.1(d), because Voronoï points are always the meeting points of 3 mediators. With the help of the

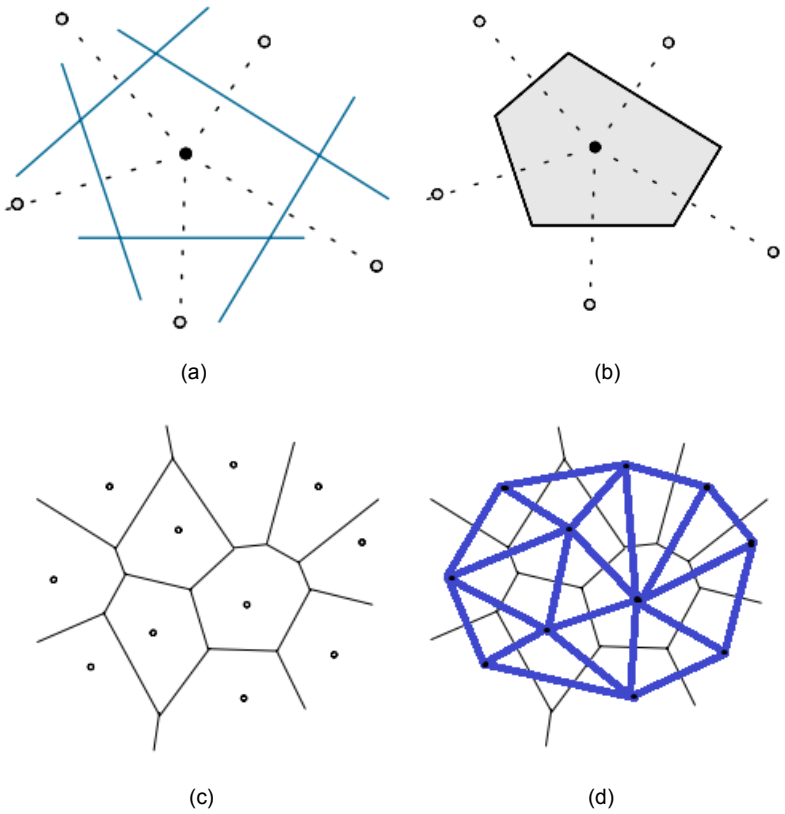


Figure 2.1: Step-by-step approach of a Delaunay triangulation. First we introduce a set of mediators around a given point ***a****, which helps to construct a Voronoï cell* ***b****. Cells can be added together to form a a Voronoï diagram* ***c*** *and then perform a triangulation to produce a mesh seen in* ***d****. Figures were adapted from [11].*

above mentioned procedure, a list of all connections can be generated.

[2] Lets S_1 and S_2 be two points on R^2 , then the mediator $M(S_1, S_2)$ is the locus of all the points which are equidistant to S_1 and S_2.

2.3. Training of the model

By model training, we imply a minimization of the function subject to constraints on its variables. The intention is to minimize an objective function by finding an optimal set of input parameters. In this work, the objective function reflects the difference between the model and the observed data. Water injection, oil production, and water production rates were used as the basis to compute objective function. The minimization problem can be written based on a L_2norm of missfit of phase reactive rates in eq. 2.11 and cumulative phase volumes in eq. 2.12:

$$\min_{x \epsilon R^n} J(x) = \sum_{w=1}^{n_w} \sum_{p=1}^{n_p} \sqrt{\sum_{t=1}^{n_t} [Q^o_{w,p}|_t - Q^r_{w,p}|_t(x)]^2}, \tag{2.11}$$

$$\min_{x \epsilon R^n} J(x) = \sum_{w=1}^{n_w} \sum_{p=1}^{n_p} \sqrt{\sum_{t=1}^{n_t} [V^o_{w,p}|_t - V^r_{w,p}|_t(x)]^2}, \tag{2.12}$$

$$V^o_p = \sum_{t=1}^{n_t} (Q^o_{w,p}|_t * T|_t), \tag{2.13}$$

$$V^r_p(x) = \sum_{t=1}^{n_{ts}} (Q^r_{w,p}|_t(x) * T|_t). \tag{2.14}$$

Here, J is the objective function, n_w is the number of wells, n_p is the number of phases, n is the number of optimisation parameters, x is the vectors of optimisation parameters, depending on the modifiers choice (see section 2.3.2), n_t is the number of observation time steps, depending on selected time coarsening option, $Q^o_p|_t$ - observation reactive rate of phase p at well w and timestep t (comes from production data), $Q^r_p|_t$ - response reactive rate of phase p at well w and timestep t (comes from simulation), $V^o_p|t$ - observed cumulative produced/injected volume of phase p at well w and timestep t, $V^r_p|_t$ - response cumulative produced/injected volume of phase p at well w and timestep t, $T|_t$ - length of time step t, $n_t s$ - the number of simulation time steps.

It is important to point that in the case when wells were controlled by the oil production and water injection rates, the model response and observation data match each other automatically unless a well hits the BHP limit and its rate changes. In conventional data assimilation, it is interpreted as an inconsistency between reservoir parameters and production data. However, this issue usually is eliminated by an optimization algorithm after a few iterations.

Also, to ensure that optimizer stays away from non-physical parameters, a penalty term was imposed. Whenever the nonlinear convergence of simulation is not reached, a large penalty term was returned as an objective function value. The scaling of the objective function is also important for some regression algorithms to perform reliably as well as the scaling of optimization parameters. The final value of the objective function depends on many factors: time coarsening, the closeness of the parameters to the optimum and the length of the training period.

2.3.1. Algorithms

Before start training of our data-driven model, we compare several optimisation algorithms that are applicable to the problem. Four regression algorithms (local, global, derivative-free, gradient-based) have been chosen among dozens of available algorithms. The gradients for gradient-based algorithms are approximated by finite-difference eq. 2.15:

$$f'(x) = \frac{f(x+\epsilon) - f(x)}{\epsilon}. \tag{2.15}$$

The choice of ϵ or perturbation size depends on the optimised parameters range and the problem itself, but in the most of cases, the value of $1e^{-6}$ shows reasonable results. Optimisation has been considered successful when the acceptable relative error f_{tol} in func(x_{opt}) between iteration steps reached the value of $1e^{-6}$, which is a default value for many algorithms. For a fair comparison of the algorithms, the run time was limited by 96 hours, and the final objective function values were compared. Simulations were performed on a cluster with 8 nodes. With each node having 40 Intel Xeon central processing units (CPU) E5-2650 v3 with a 256 megabytes of real memory.

Sequential Least-Square Quadratic Programming (SLSQP)

SLSQP is a local sequential quadratic programming algorithm for non-linearly constrained gradient-based optimisation. It optimises successive second-order approximations of the objective function with a BFGS [3] update. Popular and widely used, but not practical for problems with more than a few thousand parameters. References of this algorithm can be found in Kraft [17].

Method of Moving Asymptotes (MMA)

MMA [27] is a local-gradient based algorithm, which was chosen as an alternative to SLSQP. It is a globally-convergent [4] algorithm, which is very popular for topology optimisation problems. MMA uses a special type of approximation that is both convex and separable. Optimisation of the approximation leads to the new point of x. Then the objective and constraints are evaluated at this point, and if the test is passed, the process is restarted with the new x. If the test is not passed, a penalty term [5] is increased.

Constrained Optimisation by Linear Approximation (COBYLA)

COBYLA is the local derivative-free optimisation algorithm proposed by Powell [24]. It constructs linear polynomial approximations to the objective and constraint functions by interpolation at the vertices of simplices [6]. Calculation of derivatives takes most of the time for two previous algorithms. Therefore, COBYLA is a good alternative if a number of parameters significantly increases.

Multi-Level Signle-Linkage (MLSL)

MLSL [25] is an algorithm for global optimisation by a sequence of local optimisation, either derivative-free or gradient-based, from random starting points. The COBYLA approach was used as a local optimizer in our comparison study.

2.3.2. Modifiers

The methods and associated parameters that are adjusted in a certain way by an optimizer are called **Modifiers** in the description below. Optimization algorithm is changing modifiers to ensure that data-driven proxy model response matches the ”true” response based on either historical recorded data or high-resolution reservoir model response. Following model modifiers are available in DARTS optimization package:

- nonlinear modifier,
- linear modifiers, and
- well modifier.

[3] Broyden–Fletcher–Goldfarb–Shanno (BFGS) is a class of hill-climbing optimisation technique, used to iteratively solve unconstrained nonlinear optimisation problems

[4] Globally convergent means that it guaranteed to converge to a local minimum from the feasible starting point.

[5] Penalty term makes the approximation ”conservative”

[6] Simplex is a generalization of the notion of a triangle or tetrahedron to arbitrary dimensions.

Since regression is performed via constrained optimization, every modifier defines the minimum and maximum values (*i.e.,* bounds) for its parameters. Using the bounds, as some of the regression algorithms are sensitive to the scale of a problem, every modifier automatically performs scaling of parameters to the interval $[0,1]$. For the support of regression of several parameters sets simultaneously, a model modifier aggregator was developed. It wraps one or several model modifiers, consistently splitting the combined vector of optimization parameters between them during regression.

Nonlinear modifiers

Relative permeability has paramount significance for modelling of reservoir fluids. To date, various models have been developed and introduced to calculated relative permeability by Carman-Kozeny, Burdine, Chierici *et al.* Brooks-Corey summarised the work of Burdine and presented a modified Brooks-Corey model or also known as the power-law model, which is the most utilised model in the petroleum industry. The modified Brooks-Corey model may be expressed as:

$$k_{ro} = k_{ro}^{e}\,(1 - S_w^*)^{n_o}\,, \tag{2.16}$$

$$k_{rw} = k_{rw}^{e}\,(S_w^*)^{n_w}\,, \tag{2.17}$$

$$S_w^* = \frac{S_w - S_{wc}}{1 - S_{wc} - S_{or}}. \tag{2.18}$$

where S_w^* is the normalised or effective water saturation, k_{rw} - water relative permeability, k_{rw}^{e} - endpoint water relative permeability, k_{ro}^{e} - endpoint oil relative permeability, n_w, n_o - exponents for water and oil, S_w - water saturation, S_{wc} - residual or connate water saturation and S_{or} is the residual oil saturation.

The relative permeability term is involved in mass balance equation through the Darcy velocity equation, where it gets multiplied by $\frac{\rho_p}{\mu_p}$ term of the corresponding phase p. The set of Brooks-Corey modifier parameters was defined as S_{or}, S_{wc}, n_o, n_w, $k_{rw}^{e}\rho_w/\mu_w$, $k_{ro}^{e}\rho_o/\mu_o$, which were subjected to the following constrains:

- Corey exponents, for both phases, should stay within 0 and 5
- Residual saturation's, for both phases, cant be larger than 0.49
- $k_{rw}^{e}\rho_w/\mu_w$, $k_{ro}^{e}\rho_o/\mu_o$ are constrained within a ranges $(100, 3000)$ and $(10, 2000)$, respectively.

Linear modifiers

The general unstructured grid in reservoir simulation is usually characterized by a spatial connectivity graph represented as connection list [19]. It involves the specification of the connections between grid blocks and associates transmissibility of those connections. Transmissibility directly affects the flow dynamics in the reservoir, as it is involved as a constant multiplier in the Darcy velocity equation. The transmissibility T_{ij} between grid blocks i and j can be defined for a general unstructured grid [15] as:

$$T_{ij} = \frac{\alpha_i \alpha_j}{\sum_n \alpha_n}, \tag{2.19}$$

where α is the sub-transmissibility and can be written as:

$$\alpha = \frac{Ak}{D}. \tag{2.20}$$

Here, A is the interface area between two grid blocks, D is the distance from the pressure node to the interface along the line connecting two pressure nodes and k is the grid block

permeability. For each grid block, we define as many transmissibilities as there are neighbours n. Transmissibility along x axis between two grid blocks in a Cartesian grid can be written as:

$$T_x = \frac{2C_d A}{\frac{d_{x_i}}{k_{x_i}} + \frac{d_{x_j}}{k_{x_j}}}. \tag{2.21}$$

It also involves a Darcy constant C_d that is responsible for unit conversion. It can be clearly seen from eq. 2.21 that using a single transmissibility value as a model regression parameter is more efficient rather than rock permeability fields in x and y directions. First, in this way, transmissibility calculations were excluded from the regression loop. Second, the amount of regression parameters for transmissibility equals the number of connections that increase the number of degrees of freedom. A smaller amount of regression parameters is preferable out of performance considerations, as long as it is not limiting the ability to effectively change the model during regression.

In a way, a single transmissibility parameter connecting two given mesh elements might be even more transparent regression parameter, rather than a single permeability value, which is involved in transmissibility calculations of all connections of the given mesh element. Notice that transmissibility is a linear parameter with respect to the flow rate. Through the regression course, transmissible values were constrained by maximum value of 50000 or that individual connection transsmisibility can be larger than 10 times of its initial value.

Well modifiers

Reservoir grid blocks are typically orders of magnitude larger than wellbore diameter. Consequently, the transmissibility relationship between a well and a reservoir block should be introduced into the simulator in a special way to couple these two different scales. It is done through the use of a well index or productivity index (WI/PI). It was introduced by Coats *et al.,* [5] in the steam-flood simulation to relate the grid block pressure/rate to wellbore flowing pressure/rate. The equation that relates a well and a reservoir grid block under the assumption of a single-phase flow can be written as:

$$q_i^w = \frac{WI_i}{\mu}(P_i - P_i^w) \tag{2.22}$$

where q_i^w is the well rate into (out of) the block i, WI_i is the well index of the grid block i intersected by a well, P_i is the grid block pressure and P_i^w is the well flowing bottom hole pressure. Well index can be also viewed as well-reservoir transmissibility. Once it is determined in single-phase assumptions, it is also applied to a multiphase flow. The equation to calculate WI was provided by Peaceman [23] (eq. 2.23, 2.24) and it is still used by default in modern simulators as:

$$WI_i = \frac{2\pi h\sqrt{k_x k_y}}{\ln r_o/r_w + S}. \tag{2.23}$$

Here, r_o is a well-block or Peaceman radius and is expressed as:

$$r_o = 0.28\frac{\sqrt{\sqrt{k_y/k_x}\Delta x^2 + \sqrt{k_x/k_y}\Delta y^2}}{\sqrt[4]{k_y/k_x} + \sqrt[4]{k_x/k_y}}, \tag{2.24}$$

k_y and k_x are the permeability component in the x and y direction, Δx and Δy are the grid block sizes, h is the thickness of the grid block, r_w is the well radius and S is a skin factor. Consequently, the set of WI modifier will consist of N_{well} parameters. Well index parameters were constrained by minimum value of 1 and maximum of 1000.

2.4. Temporal clustering

Temporal clustering of the production data was done with the most popular unsupervised learning algorithms: k-means. K-means is the method of clustering observation data into the

set of k clusters where each data point is assigned to its closest cluster. The algorithm's aim is to iteratively minimize the euclidian distance between the centroid of the cluster and the given observation. The algorithm was implemented though probably the most popular machine learning library: scikit-learn. The first step is to randomly initialize k cluster centroids. Then the algorithm is followed by a two-step procedure until converged or a maximum number of iterations reached:

- cluster assignment step: where the data points are assigned to the closest centroid based on the computed distance between the data point and the cluster centroid,
- move centroid step: update the centroid position by calculating the mean of points assigned to the cluster.

The optimal number for clusters is determined via the "elbow" method. In the elbow method, we run K-mean clustering for a given range of clusters and find the sum of squared distances of each data point from the centroid of the cluster, which is also called within-cluster sum of squares (WCSS):

$$J(c^{(1)}...c^{(m)}, v_1...v_k) = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - v_{c^{(i)}})^2 \tag{2.25}$$

where c is the index of cluster centroid closest to x^i, v is the cluster centroid, $v_{c^{(i)}}$ is the cluster centroid of cluster to which example x^i has been assigned. The point at which WCSS starts to bend indicates the optimal number of clusters as can be seen in figure 2.2

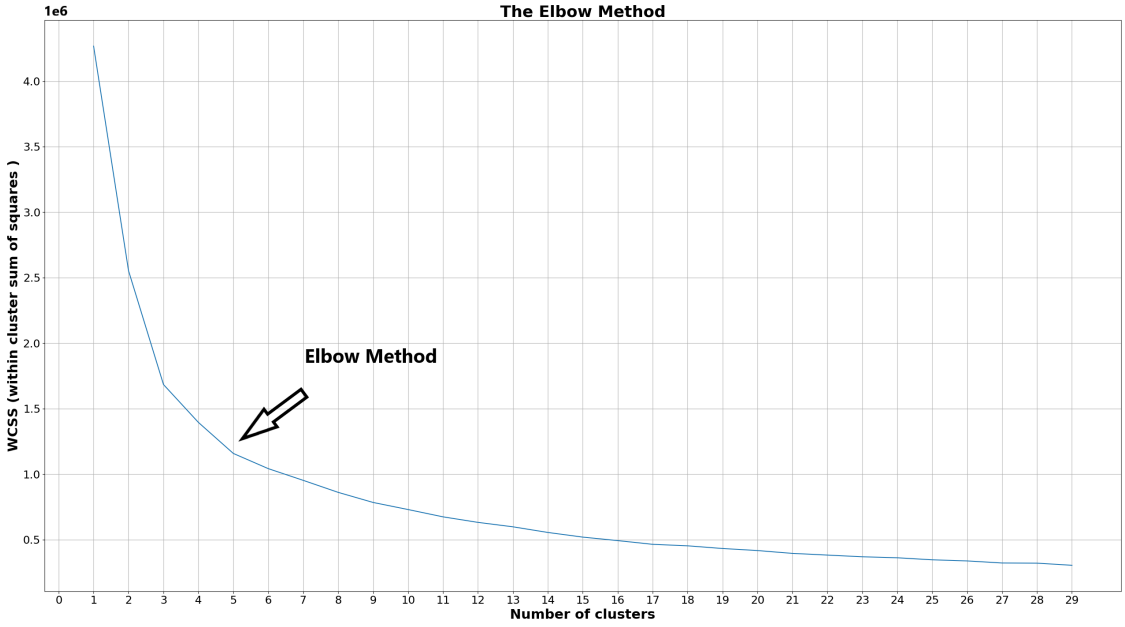


Figure 2.2: Elbow method. The point at which WCSS starts to bend indicates the optimal number of clusters.

3

Models Overview and Construction

In this chapter, the overview of the models used in this study is presented. Historical production data preparation and proxy model construction is also given here. Four model types have been used in this work:

- Real field model: German
- Synthetic model: Brugge
- Synthetic process-based ensemble: Flumy
- Synthetic MPS-based ensemble: MPS

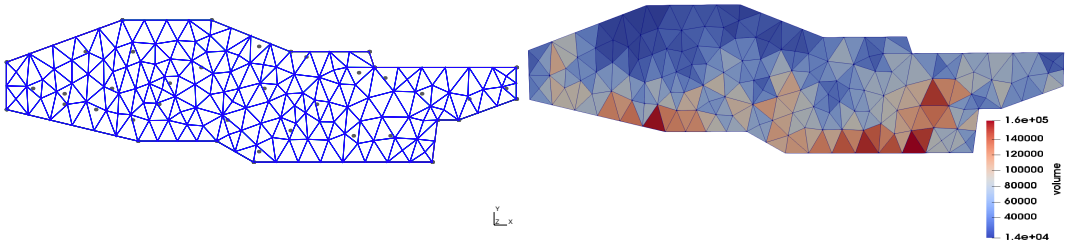
All models used in this work neglect gravity and capillary effects.

3.1. German model

The data-driven model based on the real field data will be referred to as ”German model”. It is a brown field, hence a long history of real production data is available. The production data is used for analysis and evaluation of regression framework robustness and efficiency taking into account all complexities of the data measurements in the real field. German reservoir is an elongated succession of fault blocks heading up-dip to the N-W. For the research purposes, only four up-dip fault blocks are used in this study. Fault transmissibilities and other parameters are barely known for this field. The convectional history-matching studies based on the existing geological model performed at operation company do not produce satisfactory results.

The reservoir has a large variety of depths and reservoir unit thicknesses, which can vary from 2 to 35 meters. Reservoir oil is quite viscous with an average of 20 centipoises. Low reservoir temperature and field specifics yield to in-situ paraffin and scale precipitation. In the data-driven model, the reservoir physics was limited to the two-component compositional formulation, due to negligible field gas rates. However, it is a rough assumption as reservoir physics might have changed several times throughout field exploitation history.

It was decided to use the available geological model for generation of proxy model basic geometry. First, it was necessary to identify reservoir boundary points and well coordinates (i, j). Then, the average thickness was approximated to match reservoir size and volume. An important decision was made on choosing the appropriate grid size for the proxy model: the objective was to place all wells in a separate grid blocks. Model grid generation was done using ”GMSH” open-source software [11] which performs finite element mesh generation. Resulting unstructured grid of the German proxy model can be seen in the figure 3.1(a).



(a) (b)

Figure 3.1: Unstructured proxy model mesh with a 274 elements, dots, excluding the mesh boundary points, represent proximate wells location (a) Volumes assigned to a proxy model grid (b)

3.1.1. Original reservoir volumes

The most important factor for accurate reservoir modeling is the consistency of the simulated model fluid volumes and the evaluated reservoir volumes (*i.e.*, material balance). Here, the geological model is used as a ”true” representation of the subsurface and calculated values are assumed to be correct. Proxy model volume was calculated with a grid pre-processor assuming generated mesh from 3.1(a) and uniform reservoir thickness of one meter. Then, those volumes were corrected by adjusting the following parameters:

- block height: Δz,
- porosity: ϕ,
- initial water saturation: S_w.

The first step was to correctly approximate the heights of proxy model blocks using the geological model. The proxy model elements were associated with correspondent ranges of grid blocks in the fine geological model. Hence, a sum over Δz in the vertical direction for each spatial location within a range can be found (ignoring inactive grid blocks). Then all sums are averaged within that range, representing the approximate height of a coarse block of a proxy model. The alternative option does not require any geological models and purely relies on well logs. It performs simple kriging and extrapolates values across the entire proxy grid. Only with a Δz correction, we have managed to achieve a 16% volume absolute error.

The next step was to approximate porosity and initial water saturation. Iteratively, starting from an initial guess obtained from the geological model we have managed to reduce an absolute volume error to less than 1% as seen in table 3.1.

Table 3.1: Table with the optimal porosity and saturation values that lead to a good match between geological and proxy models. Values were obtained iteratively, starting from the average values of parameters in the geological model

ϕ	0.212	
S_w	0.227	
Phase	Oil	Water
FIIPS geological model $[sm^3]$	$2.84E+06$	$6.56E+05$
FIIPS proxy model $[sm^3]$	$2.85E+06$	$6.56E+05$
Volume abs. error [%]	0.07	0.02

The proxy model grid block volume is visualized in figure 3.1(b). It can be seen that reservoir thickness/volumes gradually go down as we go to the down-dip (y-direction). With the largest blocks located close to the reservoir boundaries.

3.1.2. Well locations

In total, there are 25 wells drilled on the North-West side of the modeled reservoir block. During field lifetime, several wells were used as producers first, and as soon as well started to water-out, it was re-developed to be an injector. It is a normal practice for the economic development of the field. Those switches could be a problem for the nonlinear convergence and can provoke multiple cuts of simulation time steps which leads to the increase of simulation time and inconsistency in gradient evaluation [28]. Therefore, several well configurations were tested for accuracy and simulation efficiency.

- Configuration 1: all wells were duplicated and put in the same spot as original well. With this approach, we mimic wells switch between producer and injector and *vice-versa*; number of wells is 50.
- Configuration 2: well can be either producer or injector. Its type was chosen according to whether it was dominant (larger rates) during filed life time as injector or producer. This approach is associated with a large approximation and impose not 'realistic' flow in the model. Number of wells is 25.
- Configuration 3: wells can switch between controls. This approach is more accurate, however, associated with a small convergence limitation. When well experience switch, pressure cone has to reverse, which produce sharp changes in the function, therefore problems for non-linear convergence. However, it can be solved with an imposing a relaxation period (run model for a short period of 1 day or less). Unfortunately, this configuration required rate smoothing and adjustment as well could have production and injection rates reported for the same month. If it was the case then the minor rate was subtracted from the major to preserve material balance. Number of wells is 25.
- Configuration 4: in this case, only wells that had a switch were duplicated. It allows the most accurate production representation and not subjected to convergence problems. Number of wells is 33.

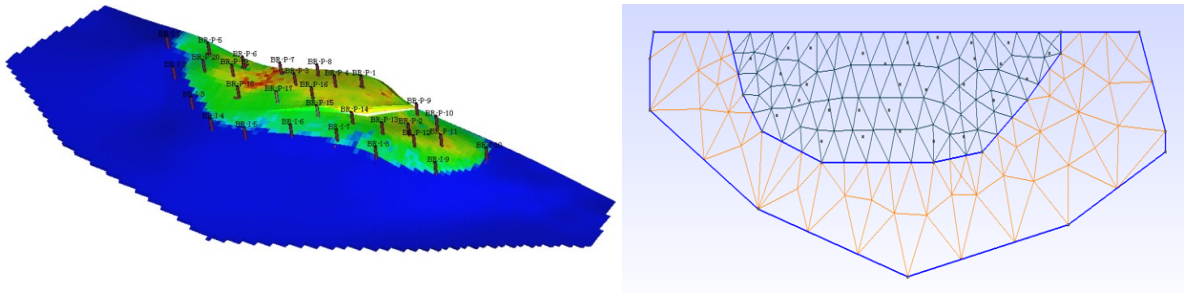
In the table 3.2 the efficiency of all four well configurations is summarised. It was decided to use well configuration 4 as a basis for further simulations as it is most accurate in terms of preserving reported well rates and effective in terms of simulator runtime. During the simulation, wells were controlled by reported oil production and water injection rates, whereas the water production rate was left unconstrained. Controlling rates are based on the real historical rates reported from the field. Injection data can be labeled as reliable since 1975 when all measurement devices were replaced. On the other hand, production data is still subjected to a large uncertainty as it is calculated from a complex allocation procedure (raw data from the three fields). Therefore, mismatches and inconsistencies in data quality are expected.

Table 3.2: Simulation performance analysis on the one month schedule for 4 different wells configurations. Numbers in the circular brackets indicate how many iterations were lost

Well Configuration:	1	2	3	4
Run time [sec]	4.8	2.82	4	3.6
Steps	786(0)	786(0)	810(13)	786(0)
Nr. of non linear iterations	3125(0)	2798(0)	3083(3083)	2952(0)
Nr. of linear iterations	34559(0)	18650(0)	18818(2153)	26137(0)

3.2. Brugge model

Brugge model is a synthetic model developed as a benchmark for optimization of reservoir production. The structure of the Brugge field consists of an East-West elongated half-dome with a large boundary fault at the northern edge as can be seen in figure 3.2(a).



(a) (b)

Figure 3.2: High fidelity Brugge model, used to generate truth model response (139x48x9 grid blocks) (a) Unstructured proxy model mesh with a 283 elements, dots, excluding the mesh boundary points, represent proximate wells location (b)

The model has 30 wells (20 producers and 10 injectors) located in peripheral water drive. Wells are perforated mainly from layer 3 to layer 8. There are more than 100 realizations of this model. However, a single realization encoded as FY-SF-KM-1-1 (Classify facies, Single Shale, Regression per facies) will be used in this study. Grid generation and material balance for this model are performed in the same manner as for German model. The proxy model is constructed in two regions: outer with a coarser meshing and inner with finer meshing, respectively. It is done to preserve accuracy in the area where the main flow happens, whereas outer cells were made larger to reduce the number of cells, therefore save computational time. There will be no significant flow dynamics in that area, hence coarsening if that zone will not significantly affect simulation accuracy.

To construct such a model, only the knowledge of reservoir boundary is required. The first step is to do unstructured discretization of the surface (obtained from the prior knowledge of the reservoir boundary) and extrude it to make a volume. Then, the volume of the proxy model is adjusted with a model thickness (average thickness 12 m of the reservoir pay zone was used), porosity and water saturation in the block to match reported fluid in-place volumes. Connectivity graphs are also obtained from the discretization procedure. Initial guess for transmissibility distribution was calculated from a uniform permeability of 1000 mD, for Corey parameters: $S_{or} = 0.15, S_{wc} = 0.25, n_o = 4, n_w = 3, k^e_{rw}\frac{\rho_w}{\mu_w} = 1800, k^e_{ro}\frac{\rho_o}{\mu_o} = 300$) and WI were initialized with uniform set of 200.

3.3. Fluvial reservoir models

The fluvial reservoir is used for investigation if the proposed approach can be used as an alternative to a conventional model upscaling, under a lack of petrophysical/geological data. High fidelity and upscaled [1] 2D fluvial reservoir model ensembles were generated in [8]. Each ensemble consists of 100 model realizations. High fidelity models have been used to generate truth data for model regression and analyze the reliability of the proxy model response counter upscaled model. Two high fidelity model ensembles were generated by two different modeling approaches:

- Flumy: process-based models using Flumy software, see example in fig. 3.3(a),
- MSP: Multiple Point Statistics (MPS) models, see example in fig. 3.3(c).

This resulted in a completely different model complexity between the ensembles. The main difference between models generated by MPS and Flumy is the main paloeflow orientation ranging from SW-NE to W-E. In addition, Flumy model has a limited statistical variability in comparison to MSP model [8].

[1] Model was upscaled by 100 times using global flow-based upscaling

Each high-fidelity model has a size of 100 by 100 grid cells (cell dimensions are $10 \times 10 \times 10$ m), which was then upscaled horizontally by 100 times. The resulting upscaled transmissibility, porosity and well index was used to initialize the upscaled model. Moreover, the same upscaled porosity was used in the data-driven proxy model to ensure pore volumes match between models. The models use a simple 5-spot vertical well set-up (one injector is located in the reservoir center and surrounded by 4 producers located at reservoir edges). Injection wells are modeled by setting a rate control of 500 m^3/day and production wells are modeled by setting a fixed BHP control of 100 bar.

For the correct comparison of the data-driven and upscaled model response, those parameters should be identical, because upscaling is sensitive to the initial boundary conditions. Initial guess for spatial connectivity and well indexes of the proxy model was chosen to be a uniform distribution of 100 and 200, respectively. However, random initial guesses were also tested with similar results. Reservoir physics was assumed to be known and correct and the same set of parameters was used for all three models.

For "approximate permeability" visualization, a similar approach as for the German field was used. Porosity distribution for the high-fidelity and upscaled/proxy models of realizations 2 from MPS and Flumy ensembles can be seen in figure 3.3. It is clear, that the model generated by MPS is more complex as the phase can flow only through district channels, which are usually smaller sized than the coarse grid block. In contrast, the model generated by Flumy has many overlaying channels that provide multiple possible flow paths, hence easier for capturing of the reservoir dynamics on a larger scale. More details about ensemble generation and simulation properties can be found in [8].

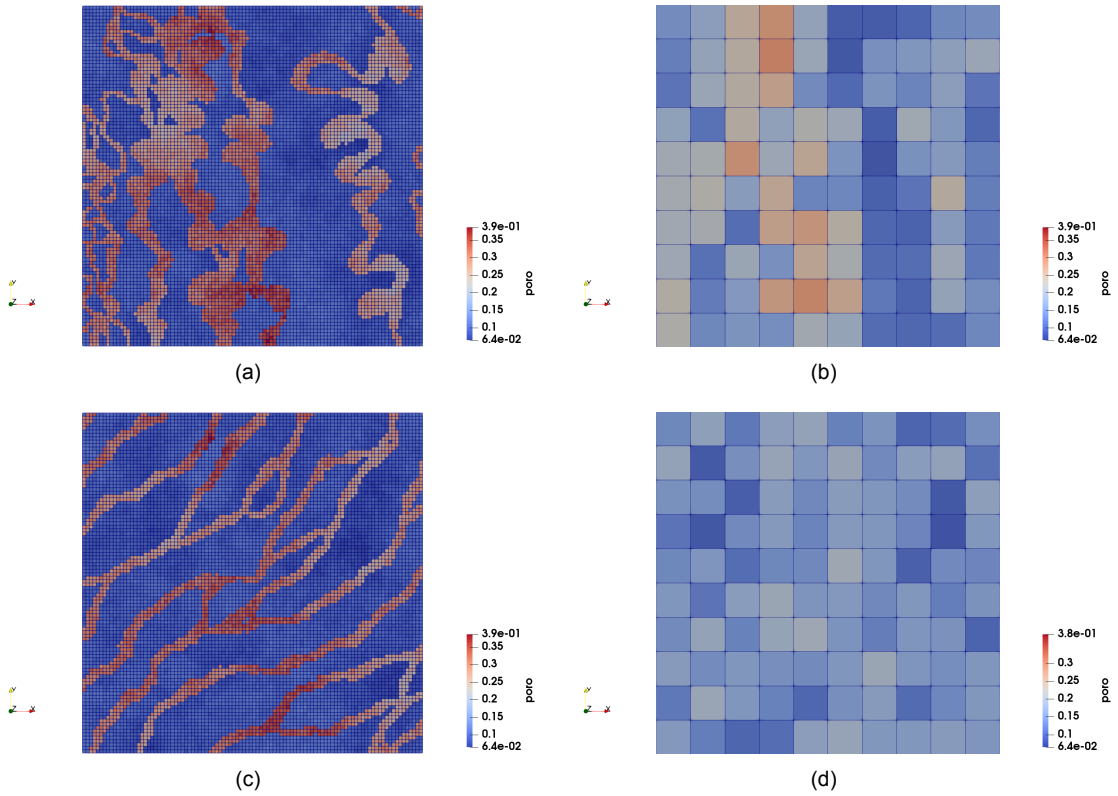


Figure 3.3: Porosity distribution for a high-fidelity (a,c) and upscaled/proxy model (b,d) of the realization Nr.2. Reservoir (a) was generated by a process-based modeling approach utilizing Flumy software. Reservoir (c) was generated by a stochastic modeling approach using Multiple Point Statistics (MPS)

4

Results and Discussion

In this chapter, we discuss the results of the development of the accurate and efficient model regression framework, tested both on real and synthetic data sets. The efficiency and robustness of the proposed framework are examined on the real reservoir model and then on the two synthetic models. Synthetic models allow fast identification of the methodology imperfectness, sensitivity and counterpoising of ”ideal” data with the real-world example.

4.1. Real field model: *German*

Accurate choice of regression strategy and careful parameter setup are key for optimal model regression. Proper bounds and thoughtful combination of regressed parameters will ensure relatively fast and accurate convergence. To identify the best regression strategy, we begin with the most simple and fast case with a single nonlinear region and a single training interval. Next, we gradually increase complexity by implementing different levels in time and space discretization. As no bottom hole pressure data is available for this model, wells were modeled with rate controls. Thereby, the well modifier was not used and the consistency of the solution was provided by pressure limit: under the rate control, the pressure limit should never be reached.

4.1.1. Time coarsening

The first thing to make regression procedure more efficient in terms of computational time is to coarsen the model schedule [1]. It was done a script which pre-processes rough historical well data [2] and output a coarsen schedule. In the original (fine) schedule averaged well rates are given for every month. Well up-time was not used for rate correction in this study. It was decided to apply coarsening on the original schedule from month to year average forming coarse schedule.

Training of the model is done on the coarse schedule, whereas quantitative analysis is completed on the fine schedule for all cases presented below (unless otherwise specified). The length of the model training is chosen to be 49 years with the maximum forecast (test) period of 5 years. According to the results that are given in table 4.1, schedule coarsening significantly reduces model training time, whereas almost the same level of accuracy is achieved. Accuracy of the model is quantified by training error, which is the L_2-norm of the difference between ”true” and simulated total rates.

[1] Model schedule is a set of well controls that are used to model wells with a historically reported well rate/BHP
[2] Rough data required well data allocation to correct date, data cleaning and substitution of missing points with zero rates

Schedule	Training Error	Time
Fine	46.96	09 : 33
Coarse	48.824	01 : 37

Table 4.1: The efficiency of a month/fine and a year/coarse schedule. Both cases were trained for 49 years on a single Corey parameter set and a single time interval. Move information about regions and intervals is given further in this chapter. Regression was controlled by a maximum of 50 iterations, however, both cases converged before reaching the maximum at approximately 40. Number of regressed parameters is 392. Training error was obtained by summing a L_2-norm of rate misfit for all wells during imposed period of 49 years

4.1.2. Sensitivity to initial guess for linear regression

Next, we examine the robustness of the modifiers listed in section 2.3.2 by imposing different bounds and initial guesses. Transmissibility is one of the most important parameters that govern flow dynamics in the reservoir, therefore it was analyzed first. Spatial connectivity graph was initialized using the unstructured pre-processor with an assumption of uniform permeability of the formation and uniform thickness of the reservoir, which equals to 2545 mD and 12 m, respectively.

Spatial connectivity visualization is a not trivial task. Since transmissibility is an interface property (i.e., located at the grid block connections), therefore we have to transform it into a cell value for a 2D visualization. It was done using the ”approximate permeability” concept [3]. To obtain ”approximate permeability” we initialize mesh with a uniform permeability of one. After pre-processing, it will allow us to get a geometric part of the spatial connectivity. Then we calculate cell ”approximate permeability” by the division of regressed spatial connectivities with the geometric part for each connection, and then averaging obtained values for every mesh element overall its connections.

The resulting maps can be seen in figure 4.1. Four cases were initialized with different uniform/random initial guesses and surprisingly lead to a similar distribution of an ”approximate permeability” fields with quite distinct patterns. It is obvious that for such a problem there are several local minima and regression results strongly depend on the initial guess and chosen step size, ϵ. However, table 4.2 indicates that starting from a different initial guess, the regressed model converged to a similar parameter distribution with the comparable objective function. It is important to mention that figure 4.1 is prone to distorting a true property distribution image due to the approximate nature of permeability reconstruction.

Case	a	b	c	d
Training Error	12.673	12.468	12.636	12.539

Table 4.2: Training error for different initial guesses for spatial connectivity. Case letters correspond to the permeability distribution in figure 4.1

4.1.3. Sensitivity to initial guess for nonlinear parameters regression

The next modifier that was analyzed is the nonlinear modifier. There are six parameters in this modifiers (three for each phase): Corey exponents n that control function shape, S_c residual saturation that impose bounds and density scaled endpoint mobility $\rho k_{re}/\mu$ that amplifies Corey function. The last parameter is multiplication of endpoint relative permeability with density and viscosity, therefore the effect of individual parameter vanishes and cannot be distinguished. Three cases have been assembled, which then were used to analyze how well a framework can restore an original set of parameters starting from a random guess. In addition, we check how far apart the nonlinear parameters can be modified based on different combination with regression of linear parameters (spatial connectivity). The cases are

[3] Permeability in this work should be not treated as geologically backed up property. It is a purely data-driven work and geologically quantitative judgment of this parameter should be relaxed

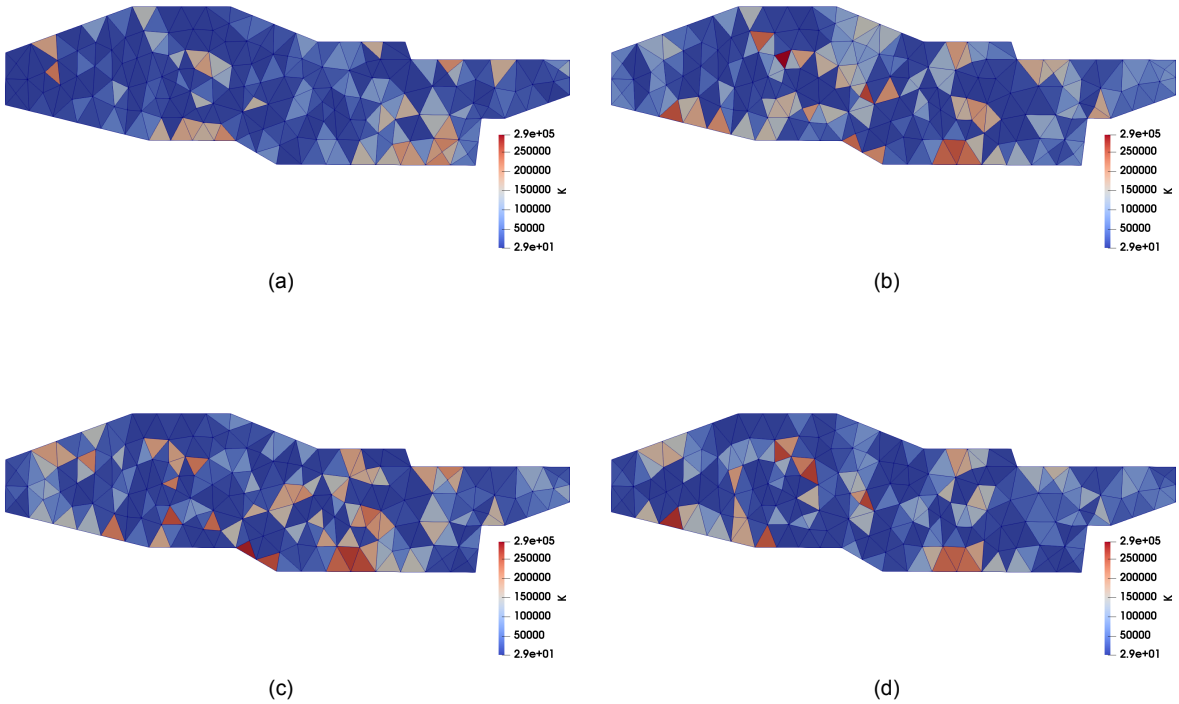


Figure 4.1: Regressed "approximate permeability" maps with different initial guess for spatial connectivity: uniform spatial connectivity of 10 (a), uniform spatial connectivity of 10000 (b), random spatial connectivity in a range of 0.01 to 1000 (c), and spatial connectivity from uniform permeability of 2545 mD (d).

following:

- regress nonlinear parameters prior to spatial connectivity,
- regress nonlinear parameters after spatial connectivity,
- regress nonlinear parameters simultaneously with spatial connectivity.

The reference nonlinear parameters were fitted to historical SCAL data using the power-law Brooks-Corey model and can be seen in table 4.3. For the random initial guess, we have used parameters that are far away from the reference parameters but still within a physically reasonable range. For this analysis, training error is based on the coarse schedule.

	n_w	n_o	S_{wc}	S_{or}	$K_{rwe}\rho_w/\mu_w$	$K_{roe}\rho_o/\mu_o$
Reference	1.58	3.98	0.16	0.14	304	67
Random	2	5	0.255	0.2	1800	250

Table 4.3: Initial guesses used for nonlinear parameters regression

Analysis of the first case suggests that starting from the reference or the random initial guess for nonlinear parameters leads to similar results which lie close to the imposed bounds. This type of solution is called the "bang-bang" solution in the optimal control theory which may indicate that the other parameters of the model (e.g. linear parameters) are not consistent. Hence, the only way to get maximum objective function reduction is to change nonlinear parameters drastically to the limit of physical boundary. From data in table 4.4, it appears that the random initial guess leads to slightly better results than the reference initial guess.

In the second case, we have observed that in both runs regressed parameters are close to

	n_w	n_o	S_{wc}	S_{or}	$K_{rwe}\rho_w/\mu_w$	$K_{roe}\rho_o/\mu_o$	Training Error
Reference guess	0.56	0.50	0.15	0.4	410	200	17.600
Random guess	0.55	0.54	0.23	0.48	917	13	17.494

Table 4.4: Regressed nonlinear parameters before linear parameter regression

the reference initial guess, even when we start from a completely random initial guess. The possible reason for this can be that we have used a regressed linear parameter set that was trained based on reference nonlinear set. Hence, through the regression course, spatial connectivity was modified in such a way that it reflects reference parameters. Hence, it was not hard for the regression algorithm to restore them when they were initialized even with a random guess. The last two parameters (table 4.5) are larger than initial guess. It is expected result as reference values are not related to the proxy model and are the only representative for a high-resolution model with match smaller scale of the blocks.

	n_w	n_o	S_{wc}	S_{or}	$K_{rwe}\rho_w/\mu_w$	$K_{roe}\rho_o/\mu_o$	Training Error
Reference guess	1.79	3.8	0.12	0.17	384	106	12.123
Random guess	1.83	3.9	0.11	0.17	867	261	13.545

Table 4.5: Regressed nonlinear parameters after spatial connectivity regression

In the last case, simultaneous regression of nonlinear parameters and spatial connectivity is evaluated. With reference initial guess, the smallest training error was achieved among all observed cases. However, it can be seen that the regressed parameters are significantly different from the reference set. The possible explanation for this difference is that throughout the simultaneous regression of linear and nonlinear sets, parameters are influencing each other in each iteration step and are adjusted accordingly. In addition, optimization problem has much more degrees of freedom which also affects the convergence.

	n_w	n_o	S_{wc}	S_{or}	$K_{rwe}\rho_w/\mu_w$	$K_{roe}\rho_o/\mu_o$	Training Error
Reference guess	1.00	1.43	0.35	0.46	564	122	11.174
Random guess	0.51	5.00	0.49	0.49	1000	157	16.036

Table 4.6: Regressed nonlinear parameters simultaneously with spatial connectivity regression

Different combinations of optimization strategies have been analyzed and the conclusion was made that the simultaneous optimization of linear and nonlinear parameters starting from physically realistic nonlinear parameters leads to the most accurate results.

4.1.4. Choice of objective function and optimizer

Intending to do an efficient model regression, we have to use an adequate objective function and optimization algorithm. We have tested the objective function based on reactive rates eq. 2.11 and cumulative volumes eq. 2.11. Both of them showed reasonable good results, however, eq. 2.11 showed to be more sensitive to rapid changes and therefore was used as a basis for all further runs. Also, different ways to calculate rate misfit was tested: based on the simple error, squared error, L_2-norm, weighted L_2-norm by a standard deviation of the well error in whole training interval, squared error normalized by N_p, and finally squared error normalized by N_p and weighted by the standard deviation. All of these norms showed nearly the same rate of objective function decrease as seen in Appendix A.2. L_2-norm was used as a basis for the misfit calculation.

Analysis of the optimization algorithm showed that SLSQP is the most efficient algorithm for our problem based on the training interval error. The objective function and algorithm test were performed based on the scenario when nonlinear parameters were assigned to each cell

Algorithm	L_2-norm
COBYLA	40.532
MLSL	40.273
MMA	49.483
SLSQP	31.849

Table 4.7: The objective function of 4 different regression algorithms after 96 hours

individually and the model was trained on the fine schedule. Future studies of the other real data sets are advised to get a full sensitivity of the objective function and algorithms, however, it is not within the main scope of this work.

4.1.5. Regression to the data

We started model regression with the fastest and simplest scenario, which consists of regression of a single nonlinear parameter region and spatial connectivity between all blocks. Model training is done for 49 years on a yearly schedule. Default function tolerance and ϵ was used, as it showed good convergence results during modifier analysis. Figure 4.2 illustrates proxy model convergence to the historical field data and surprisingly, even an un-optimized model response is close to the ”true” data. After the regression course, training error has decreased by two times. An interesting result is observed in the forecast period, there is a distinct peak in optimized response of total water production rate during test interval. It is probably related to the inconstancy of the trained model and forecast data.

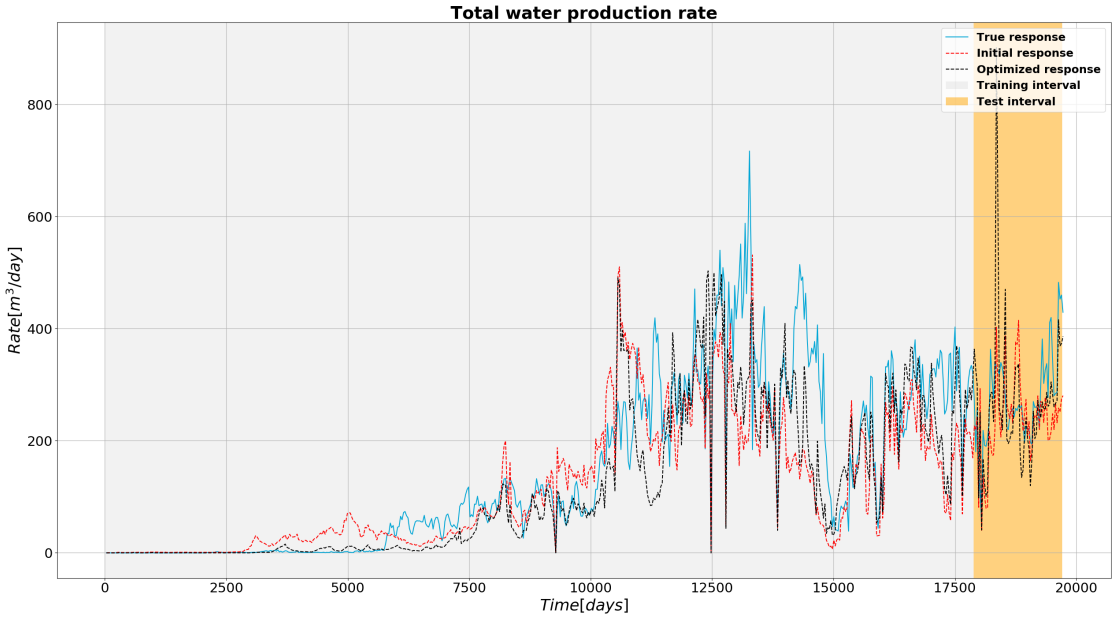


Figure 4.2: shows proxy model response before and after optimization compared to the reported historical rates.

Nonlinear parameters distribution

Generally, we need at least one set of nonlinear parameters to describe and accurately simulate fluid flow in the entire reservoir. However, sometimes for very extensive and heterogeneous reservoirs, we need a separate set of parameters for each physically unified region. We introduce three levels of physical discretization and model separate regions with an individual set of parameters. First, the reservoir is modeled with a single nonlinear parameter set in fig. 4.3(a), hence we call it a single region case. Second, the reservoir was equally

divided into 9 regions as showed in fig. 4.3(b). Third, when each grid block has an individual parameter set as in fig. 4.3(c).

It can be seen from the table 4.8 that as we increase the number of nonlinear regions, we expect a decrease in training and prediction error and an increase in the model regression time. However, when the model is switch from one to nine regions and an additional 54 nonlinear parameters are introduced, the model converged even faster than for a single region with a test error comparably close to the 274 regions. From the figure 4.3 we can see that 9 regions and 274 regions have some similarity in the distribution of the nonlinear parameter, which is a good indication that similar trends were captured with significantly different number of degrees of freedom.

Regions	Training Error	Test Error	Time
1	48.733	26.322	02 : 10
9	48.824	20.113	01 : 31
274	42.8921	19.250	23 : 09

Table 4.8: Comparison of total reactive rate error between different levels of physical domain discretization. Training interval length is 49 years, whereas prediction length is 5 years

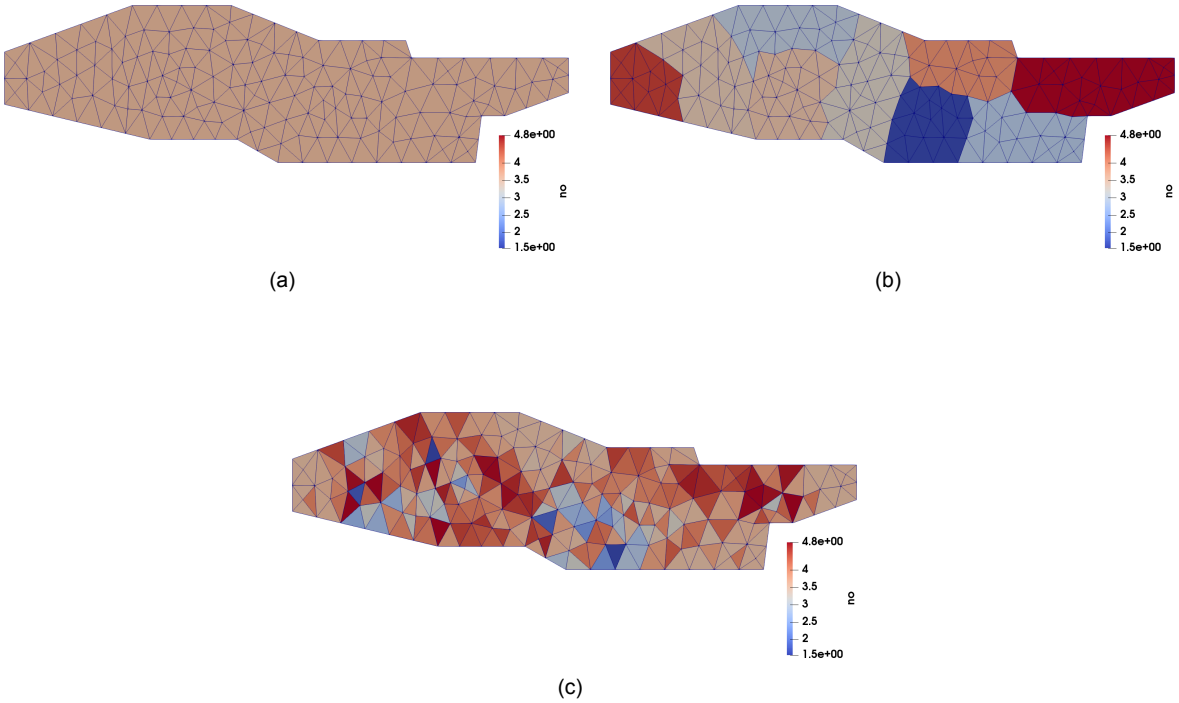


Figure 4.3: Exponential constant for oil phase at different levels of physical domain discretization: single set of nonlinear parameters (a), nine regions with individual set of nonlinear parameters (b), and a unique set of nonlinear parameter is assigned to each individual cell (c).

Sequential in time regression

It is of general knowledge that reservoir physics can change over its field life. It involves changes in the state of the fluid, precipitation of scales, waxes, microbial effect, wettability change, and many other things that are typical during field development. To account for such changes, it was decided to do a sequential regression in pre-defined time intervals. It

ensures that we adjust a separate set of parameters that describes the specific behavior of the reservoir better within each interval. Moreover, this approach enhances regression accuracy as intervals with minor changes in the reservoir dynamics, as usually observed in early field life, do not need many iterations to converge to the optimum solution. Hence, more time can be spent on resolving more complex time intervals.

The first interval is initialized with some random guess, specified by the user. After the regression is complete, the regressed parameters from the previous interval were used as an initial guess for the next interval. Also, each following interval should be initialized with the correct values of nonlinear variables in simulation (from the last time step of the previous interval). The important question is how many intervals should be used and how they should be distributed.

The first attempt was to divide training courses into equally spaced intervals. Then, manual analysis of the reservoir dynamics (mainly investigation of well events: rapid rate change, well openings/closings, water breakthrough, *etc.*) was performed. Finally, k-means clustering was utilized on the data set containing oil and water production rates of each well to support manual time discretization. As a result training course has been discretized into 4, 5, 7 and 16 intervals (interval spacing can be found in A.1).

Several attempts of time interval-based regression were made. First, sequential regression to the data was examined in the simplest case with a single nonlinear region and coarse training schedule. The results were not satisfactory as many intervals failed to converge or regressed results were inconsistent with model boundary conditions. Then, the same approach was tested with 274 nonlinear regions with no significant improvements observed. Those results can be related to inadequate data or its insufficiency due to schedule coarsening and intervals imposing. Finally, the sequential regression of the model using a fine schedule reduced convergence problems and the positive effect of using intervals, shown in table 4.9, was achieved. The analysis showed that the discretization of training course into 5 and 7 intervals was the most successful in terms of both training and forecast errors. There was no improvement when training course was discretized equally into 4 intervals: actually, it even decreased model accuracy. Inconsistent results were obtained with 16 intervals, which is probably due to the limited training course data within a single interval.

Intervals	Training error	Test Error
1	31.84	16.485
4	40.586	15.728
5	28.69	14.361
7	29.742	15.3
16	32.8209	18.51

Table 4.9: Comparison of total reactive rate error for different levels of time refinement. Here, training interval length is 49 *years, whereas prediction length is* 5 *years.*

Refinement in discretization

Some optimization algorithms lose their efficiency when the number of parameters exceeds a few thousand, but on the other hand, extra variables can have significant enhancement on the model accuracy. Hence, there is always a trade-off between accuracy and efficiency. To test the effect of an increase in the number of degrees of freedom, the spatial model refinement was performed. One way to do a simple refinement is to take a coarse model mesh and refine it by uniformly splitting all elements. The new set of nodes are inserted, by linear interpolation, to produce new refined mesh as can be seen in figure 4.5. It is an automated method that can be found as ”refinement by splitting” in ”ParaView” software [1].

An alternative option exists where one can manipulate element size factor or prescribe different mesh sizes at boundary points. Furthermore, there is an option to do a local refinement of the grid. Table 4.10 presents results obtained from a regression of the coarse and refined

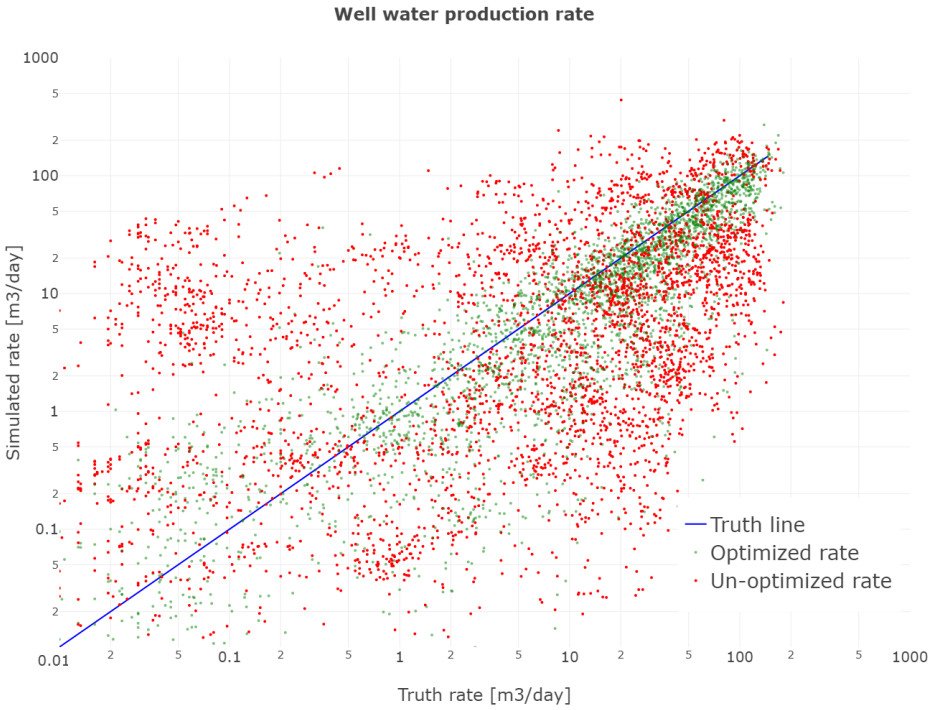


Figure 4.4: Water production rate for all wells throughout the training period; the clear improvement can be seen as the green could is closer to the "true" diagonal.

models under the same regression strategy. It can be seen that refinement with the same initial guess as for the coarse-scale model has no improvement in model accuracy, despite a nearly quadrupled number of regressed variables. To improve these results, we have re-sampled regressed coarse-scale model parameters into a fine-scale mesh and used it as an initial guess.

To get a re-sampled parameter field, which is used to build a fine model spatial connectivity graph, Visual Toolkit (VTK) filter [26] was utilized. "Re-sample with data set" filter samples the points of one dataset on to the points of another dataset. The output parameter field has the same structure as the source field (fine mesh), and its point data contains the re-sampled values from the input set (regressed coarse model parameters). Since data is sampled into a point and not a cell itself, it has to be averaged by simple arithmetic mean. Finally, if the geometry of the connection between cells is known, a re-sampled spatial connectivity graph can be constructed for fine model regression. As table 4.10 shows, the initialization of the refined model with re-sampled data has a positive effect on both training error and model regression time.

Together, these results provide important insight into the development of the optimum model regression strategy. With the increased number of variables, we impose a more local solution to the given problem and the model convergence depends on the starting point (i.e., initial guess), which sometimes can lead to poor results. Therefore, it is advised to run the coarse-scale model (on all levels) first and use regressed parameters for the fine model. The same approach should be used for schedule and possible nonlinear regions.

Optimal training performance

Based on the analysis of the regression for the model, we introduce two best case scenarios (table 4.11): slower but more accurate and faster but less precise.

The cumulative water volume accumulated through the forecast period for the most accurate case is shown in figure 4.6.

Mesh	Training Error	Time
Coarse	48.83	01:37
Fine	52.04	35:35
Fine (re-sampled)	46.50	29:07

Table 4.10: Comparison between coarse scale and refined models. Cases were trained for 49 years on a single nonlinear region and single time interval.

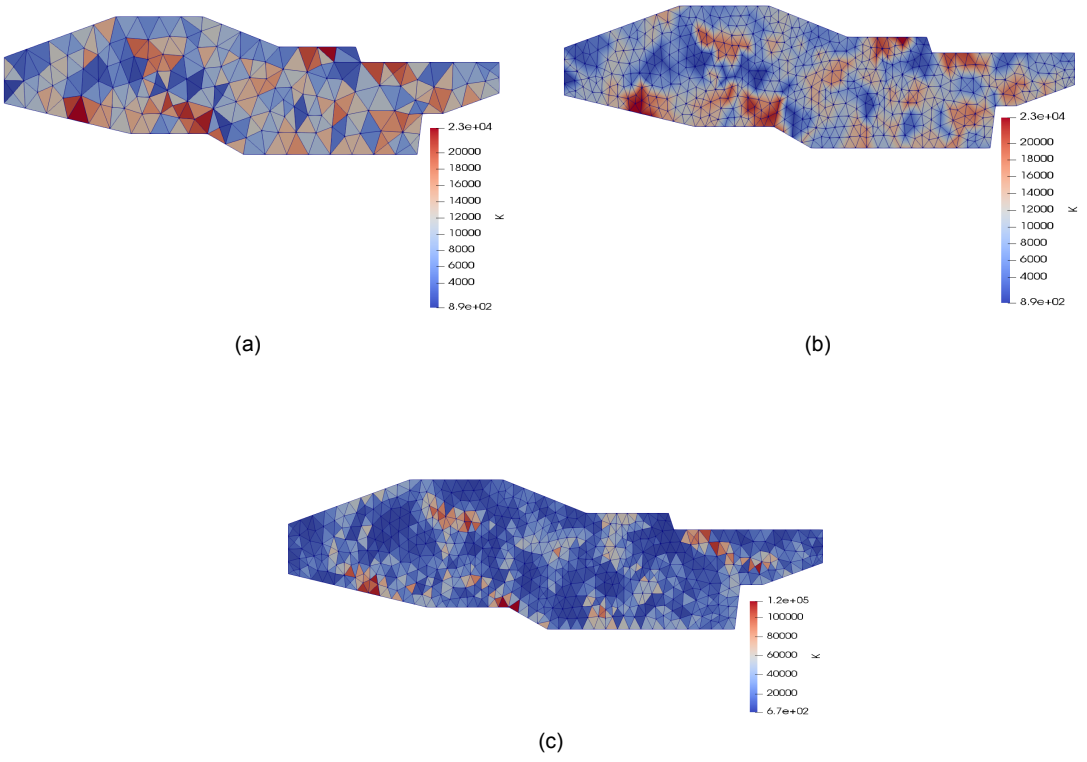


Figure 4.5: Coarse scale mesh with 274 elements with regressed average "permeability" assigned to each cell (a), refined coarse scale model with 1094 elements, where each cell "permeability" is re-sampled from the regressed coarse scale result (b), and regressed average "permeability" (c)

	Fast	Slow
Regions:	9	274
Intervals:	1	5
Mesh Size:	274	274
Schedule:	Coarse	Fine
Regression time:	01:31	84:08
Train error:	48.824	28.69
Test error:	20.113	14.361

Table 4.11: Two best regression scenarios for the German model

Based on the results obtained from the regression to real field data, it is hard to evaluate the efficiency and robustness of the proposed regression framework. For all observed cases, no significant reduction of the training and test error was achieved. This can be either associated with the regression framework, or with poor data quality and lack of BHP data. Consequently, to do an appropriate evaluation of the framework, the observed regression strategies were tested on synthetic data sets, where artificially generated observed data is more complete and accurate.

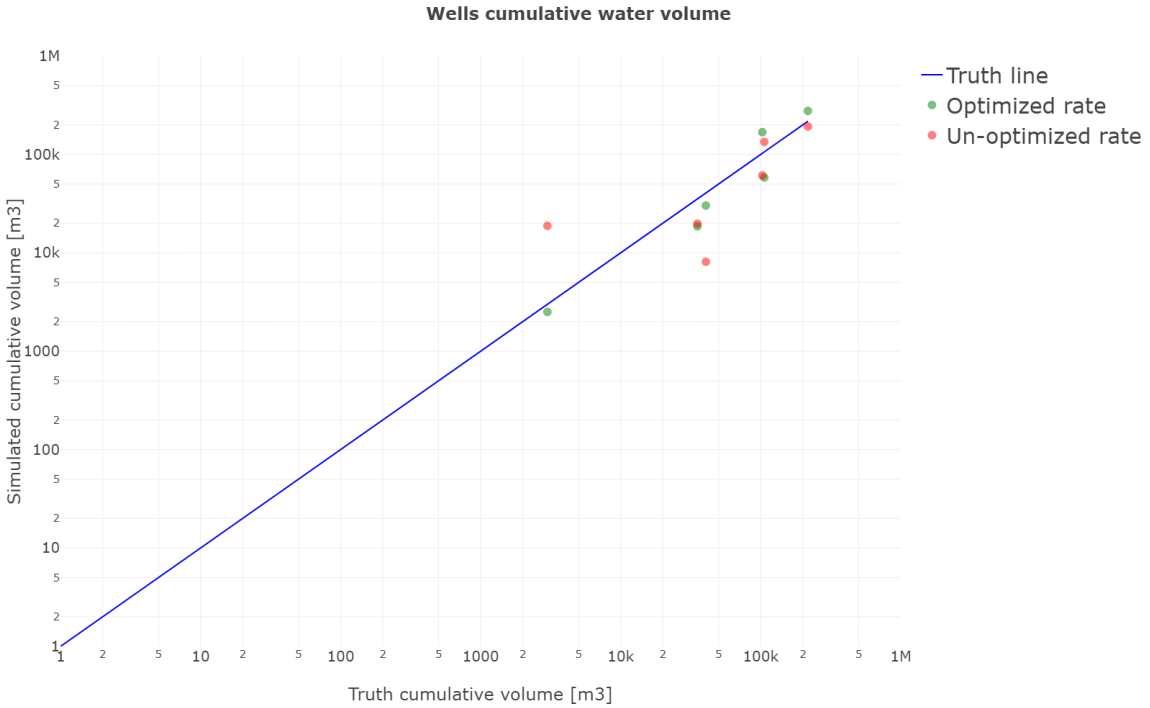


Figure 4.6: Cumulative water volume accumulated through a reservoir forecasting period of 5 years after 49 years of model training; the un-optimized well rate can be found by an associated red point.

4.2. Synthetic model: *Brugge*

The first synthetic model used in this thesis is a single realization (Nr. 77) from the well-documented Brugge benchmark. This model is used to generate synthetic production data, which is then utilized as a ”true” response for data-driven model regression. It ensures that the accuracy of model regression is not limited by the data quality. Moreover, it allows us to incorporate bottom hole pressure data as one of the well controls and increase the number of matching parameters [4], which was not usually available in the real field model.

4.2.1. Simulation performance

First, the Brugge model was used to examine the efficiency of (DARTS) [7] and to what extent the proxy model can reduce computational time compared to a full (high fidelity) model. For this purpose, an Automatic Differentiation General Purpose Research Simulator (AD-GPRS) [30] was utilized. Both models were simulated for 2400 days with random perturbations [5] in production and injection wells every 120 days.

- Injection wells: 160 bar plus random number sampled from a uniform distribution in the range from 0 to 30
- Production wells: 100 bar minus random number sampled from a uniform distribution in the range from 0 to 30

Table 4.12 compares the simulation run time of the two models launched on DARTS and AD-GPRS simulators.

It can be seen that DARTS has managed to reduce the computational time of the high fidelity model more than 6 times. It can be explained by improvements in DARTS performance based on OBL techniques as well as a significant reduction in I/O operations since DARTS can pass

[4] If the rate controls are used, water injection and oil production rates are matched automatically
[5] To ensure consistency of the wells controls between simulators the same ”seed” for random number generator was used

all the data to Python directly [6]. The even larger performance gain was achieved for proxy models, see results in table 4.12.

	Full model [sec]	Proxy model [sec]
DARTS	97	**0.46**
AD-GPRS	645	8.3

Table 4.12: Comparison of a run time for DARTS and AD-GPRS on high-fidelity and proxy Brugge models

4.2.2. Training strategies

Investigating the reliability of the proposed framework, we have applied the same model regression strategy as for the German model mimicking similar production regimes. First, the effect of additional data (BHP measurements) to the regression course was analyzed. The pressure controls of the regressed German model were used to mimic the realistic pressure data set. BHP data for each well was averaged over a year period, scaled to fit Brugge model reservoir pressure and then used to model injection well using pressure control. Similar random perturbation was added to injection BHP control as for simulator efficiency analysis above. Production well control was fixed to 100 bar which ensures that any change in production well rate will be purely associated with the perturbations in the injection wells.

The ”true” response was generated for 54 years as for real field example, where 49 years are used for training and 5 years for the forecast (test). An opportunity to use pressure data in wells control allows us to move away from the automatic matching of water injection and oil production rates, which was an inherited principle in rate control scenario. Therefore, the mismatch between those rates can be utilized in the objective function evaluation. Moreover, well index (WI) can be added to the regression parameter vector as it influences the correspondence between well rates and BHP values. Therefore, three modifiers (linear, nonlinear and well) can be used simultaneously.

The results of the case where all three modifiers were used with a single nonlinear set are shown in figure 4.7. The closer the green points are to the ”truth line”, the more accurate is the model prediction. The farthest points from diagonal mainly clustered on the left bottom corner, representing wells with low cumulative volume. The lower volume corresponds to a smaller contribution to the objective function, hence the regression algorithm was less sensitive to those wells as seen if fig.A.3.

It can be seen from the data that there is a significant improvement in the forecast accuracy when BHP data is available. A good match in total model reactive rates and wells cumulative water volume was found between regressed proxy model and high fidelity model response. Nearly all points in figure 4.7 lie on or close to the diagonal that can indicate a reliable restoration of water breakthrough.

It is important to mention that using a well modifier separately from any other modifier affects the final results, yet not so dramatically. Therefore, major improvement of the model regression results compared to the real field model can be associated with the accurate, sufficient amount of data and additional inputs to an objective function (injection rate and oil production, which before that was matched automatically). Moreover, this model set-up was tested in the case of extensive data availability scenario, where data points were generated every month. The results of this case showed marginal improvement over the case in fig. 4.7, but the regression time of the model was tripled. Therefore, even higher observation data frequency does not significantly contribute to regression accuracy in this case.

Next, the model was trained on the same set up for 10 years to limit the amount of observation data and evaluate regression quality under such conditions. The training interval was

[6] With appropriate modifications of the standard I/O capabilities, the reduction in the runtime of AD-GPRS can be achieved as well

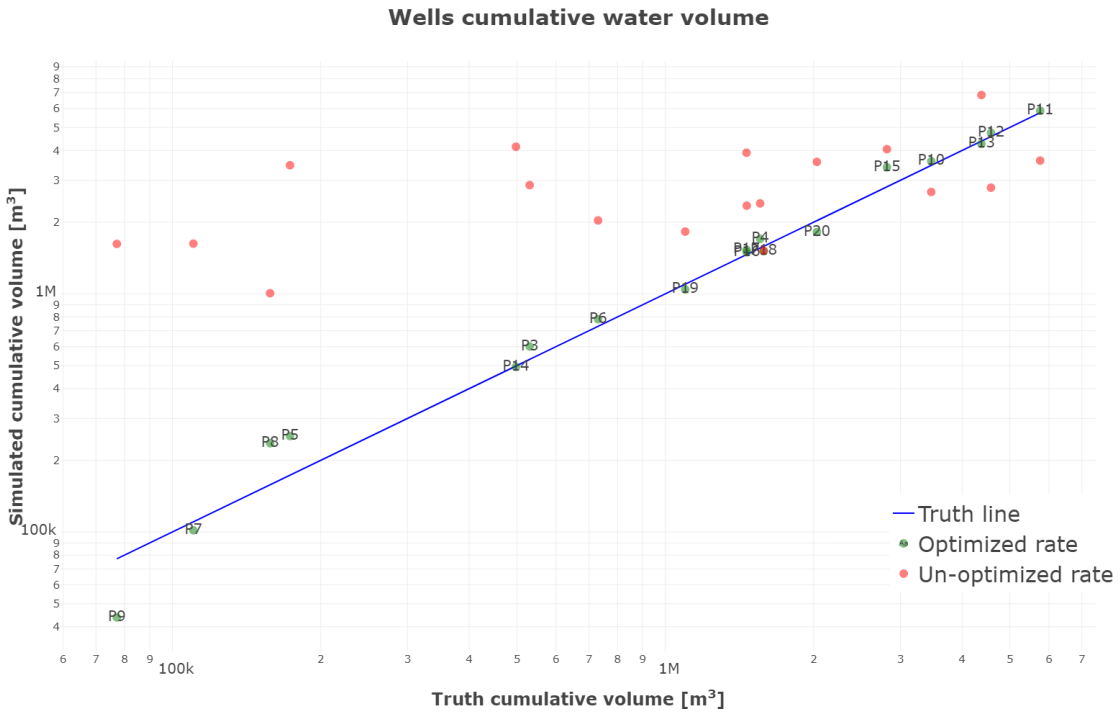


Figure 4.7: Cumulative water volume accumulated through a reservoir forecasting period of 5 years after 49 years of model training; the un-optimized well rate can be found by an associated red point

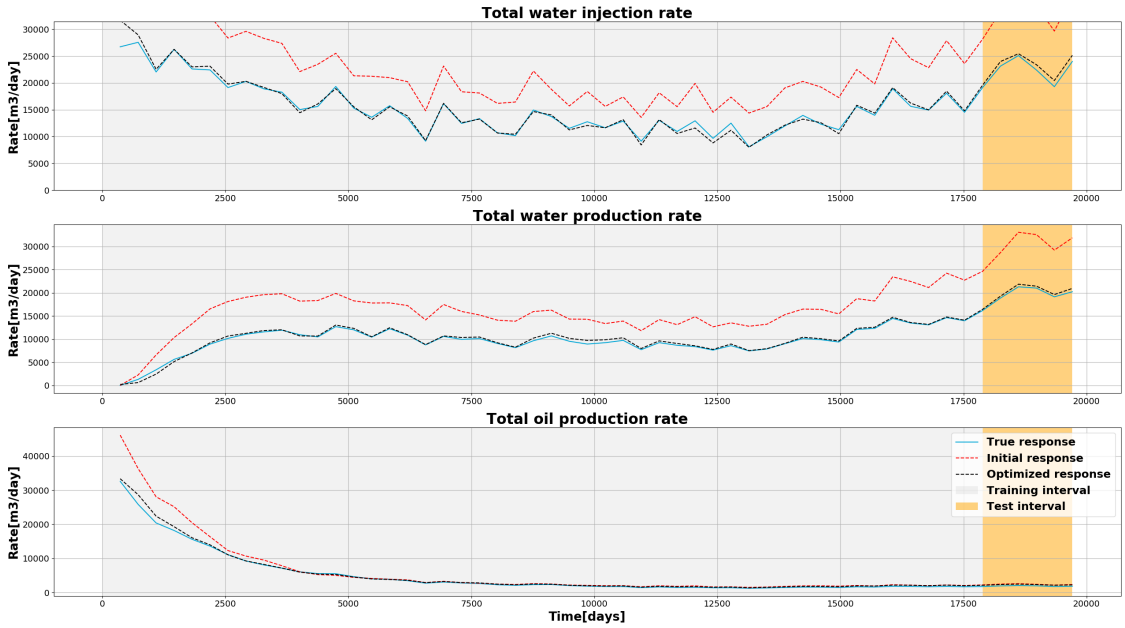


Figure 4.8: Total reactive phase rates for both, training and forecast intervals. Model in both cases was controlled by BHP and phase rates were used as a matching criterion

reduced by 5 times compared to fig. 4.7. Figure 4.9 shows the results obtained from this regression course. It can be seen that there are more outliers, especially wells P9, P3, P7, and P8 (three of them are the same as in the previous case - P7, P8, P9). Further analysis showed that those wells are located far from injectors obscured by other producers, as can be seen in fig. 3.2(a). Consequently, the waterfront reaches those wells later. We believe that it makes rate values from those wells less informative for the regression algorithm. Hence,

it fails to adjust parameters in the reservoir for those wells with the same quality than for others.

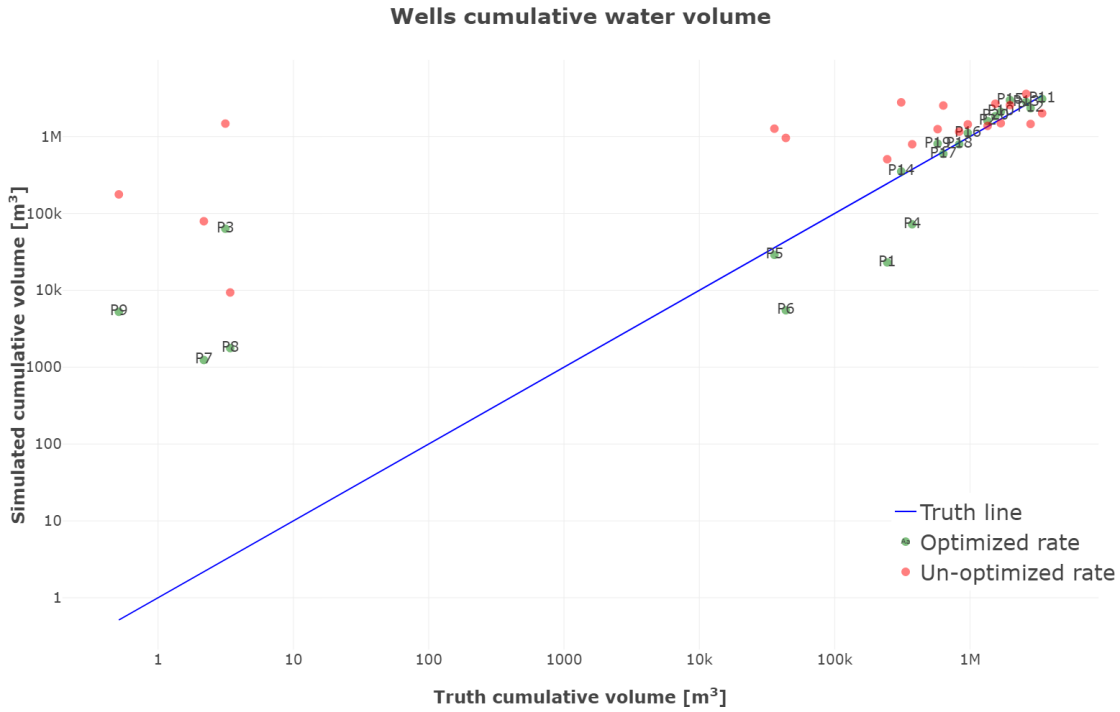


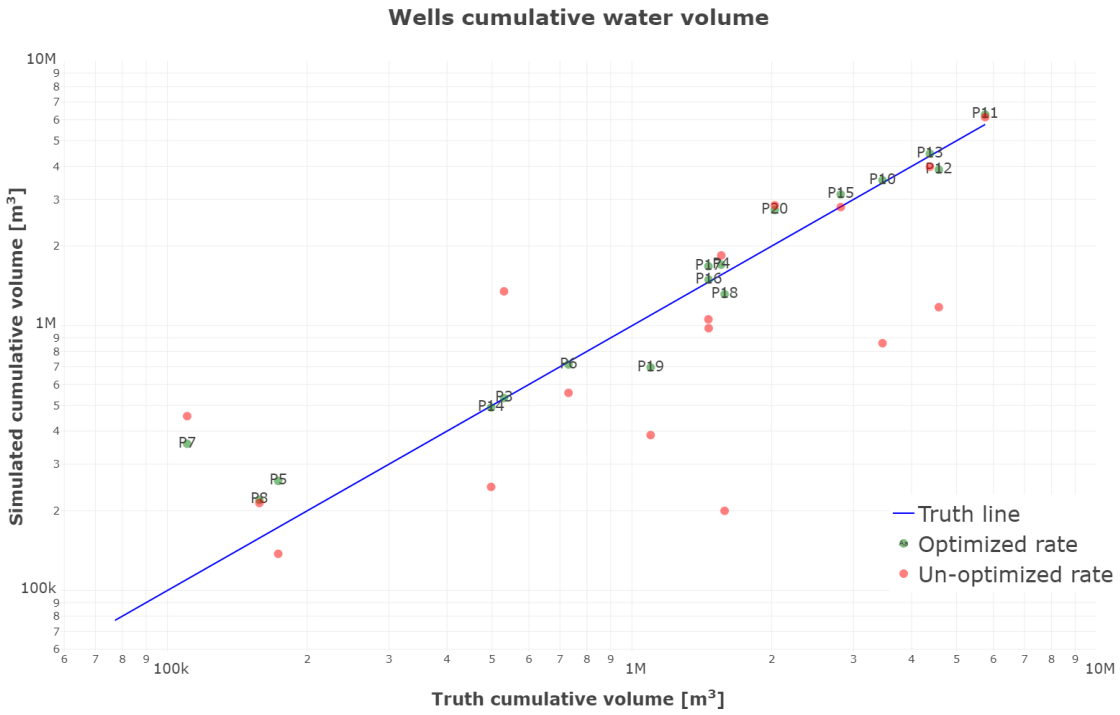
Figure 4.9: Cumulative water volume accumulated through a reservoir forecasting period of 5 years after 10 years of model training. There are more outlying points, comparing to the longer training case. Those outliers are purely associated with the lack of a strong well response.

We can conclude that versatile and accurate data should be available for a long period of production history for efficient regression and accurate long term prediction. Scenario with rate well controls, similar to the German model, was also tested to separate contributions of data accuracy and completeness to successful regression. Wells were modeled using water injection and oil production rate controls based on high fidelity model response. Maximum injection and minimum production bottom hole pressure constraints were also imposed. Still, pressure limits should never be reached during the regression, otherwise well rates will not correspond to the true data as can be seen in the water injection rate in fig. 4.10(b). Interestingly, the results, as shown in figure 4.10, indicate that even with well rates as controls, we can manage to get a reasonably accurate production forecast, which is negligibly less accurate than BHP controlled case.

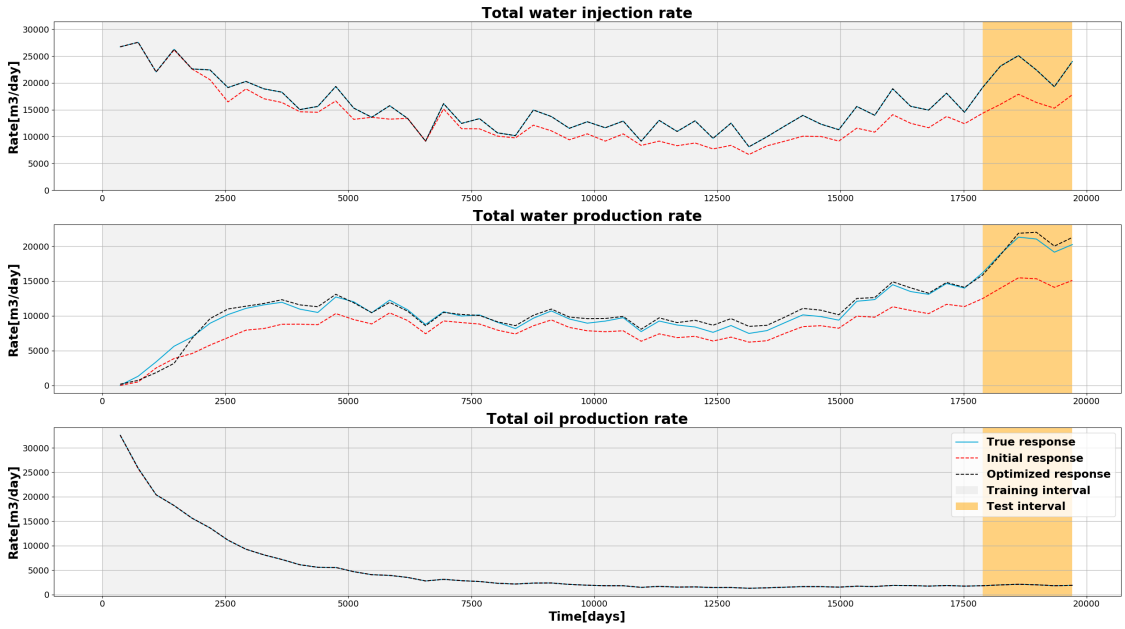
4.2.3. Nonlinear modifiers

We also tried to divide the model into physical regions [7] and see if that can significantly improve the regression accuracy. The idea behind it was to simply increase the number of degrees of freedom while keeping a relatively fast model regression compared to a single region case. The model was clustered into 2, 4, 8 and 283 regions as seen in figure 4.11, where each region was modeled with an individual set of nonlinear parameters. The results of this approach are shown in table 4.13. It can be seen that a small increase in the number of nonlinear regions decreases the training error by up to 10% without a significant increase in optimization time. As expected, the case with 283 regions is the most accurate, however, it takes 6 times longer to converge. Another interesting observation can be made from figure 4.11(d). The variation of nonlinear parameters is concentrated in the area where the main

[7] Physical regions are the model zones that can be modeled with a single nonlinear parameter set



(a)



(b)

Figure 4.10: Cumulative water volume accumulated during forecast interval of 5 years (a) and total reactive phase rates for both, training and forecast (b) Wells were modeled with rate controls

flow is happening, which is a good confirmation that algorithms are sensing dominant model dynamics. Typically, historical production data prone to large uncertainty and measurement noise. We tried to simulate this situation by adding white noise to the ”true” response of the training interval, while the response during the forecast was not changed. Noise addition was achieved by adding a random number sampled from a Gaussian distribution with zero mean and a standard deviation of 5% to the original well rate values. Such noise represents

Regions	Training error	Time
1	0.312	02 : 22
2	0.299	02 : 12
4	0.282	02 : 41
8	0.298	04 : 21
283	0.272	13 : 19

Table 4.13: Error and time comparison between different levels of nonlinear region clustering

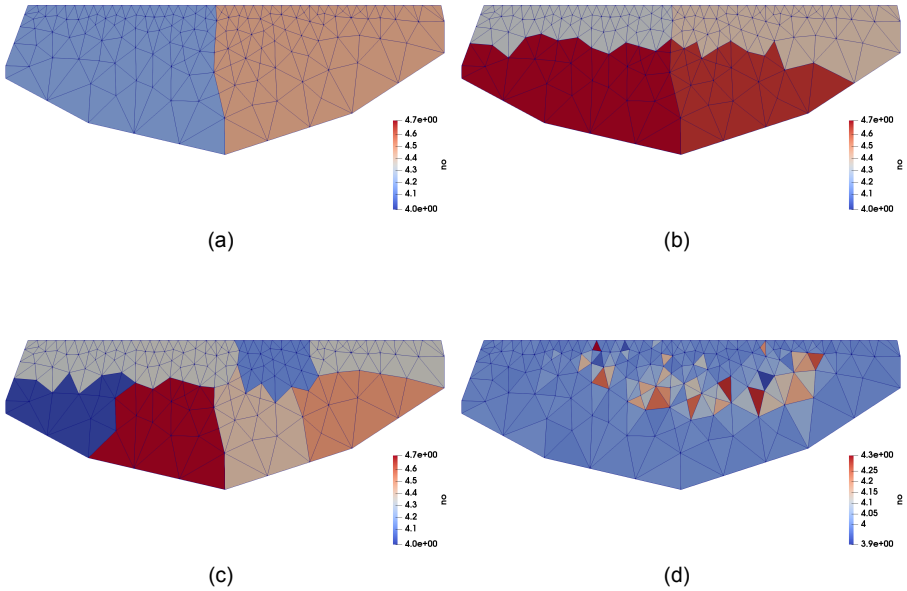


Figure 4.11: Corey exponents for oil phase with 2 (a), 4 (b),8 (c) and 283 nonlinear regions (d)

an extreme case, which mimics a failure in all measurement devices. The resulting match can be seen in figure 4.12. There is a deviation from the true solution for the injection rate in the forecast period, however, the water rate match has improved. Similar results were obtained with a 20% of measurement error.

4.3. Synthetic models: *fluvial reservoirs*

Upscaling reservoir properties is a crucial step in the reservoir simulation workflow. It a necessary procedure, which allows the utilization of a coarser model to perform reservoir simulation in a feasible time. At the same time, the upscaled effective parameters should still capture all important properties of the high fidelity model. To date, various methods have been developed and introduced to perform accurate and efficient model upscaling [16]. Each has its advantages and drawbacks. The flow-based method is one of the most widely used upscaling techniques in the petroleum industry. In general, the global flow-based upscaling approach gives a reliable and consistent representation of the fluid flow in original reservoir heterogeneity.

The data-driven approach proposed in this thesis can be an alternative to classical upscaling techniques when an accurate high-fidelity geo-cellular model is not available, but there is an excess of historical production data. Moreover, this method can be used even when PVT, SCAL data are also unavailable. In our study, we an ensemble of upscaled models only for validation purposes.

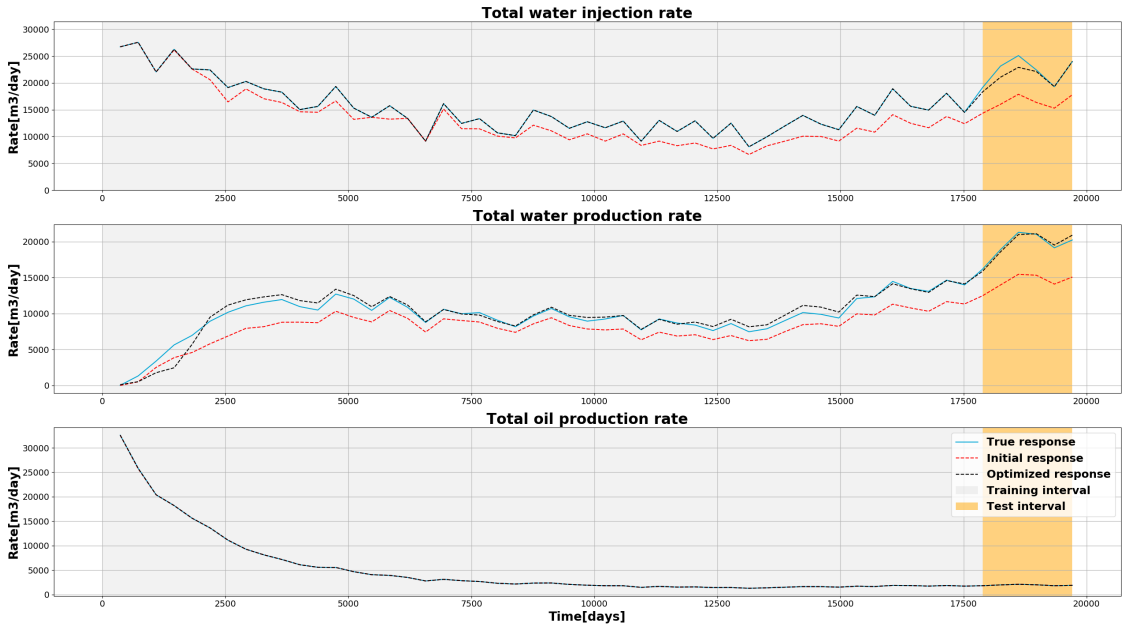


Figure 4.12: Cumulative water volume accumulated during forecast interval of 5 years (a) and total reactive phase rates for both, training and forecast (b) Training data has 5% *measurement noise imposed to it.*

Here, we verify the accuracy of the data-driven proxy model against its upscaled counterpart in comparable conditions. Throughout the upscaling procedure, spatial connectivity parameters (i.e., transmissibilities and well indexes) are evaluated to represent the fine-scale model on a coarser grid. Hence, the same set of parameters was obtained in the data-driven model through its training stage. Besides, the identical PVT and SCAL[8] properties were used for both high-fidelity and proxy models. However, in a more general case, they can also be upscaled or regressed from the data. To account for data uncertainty and validate our approach for a range of models, the analysis was done based on two model ensembles with 100 high-fidelity members each.

The first ensemble was created using Flumy process-based modeling software package and used for uncertainty quantification in [9]. For each high-fidelity model, an upscaled and a data-driven proxy models were constructed. Each data-driven model regression was limited by 100 iterations, however, most cases converged before reaching the imposed maximum. Regression of a single realization took from 20 minutes to one hour on four Intel Xeon CPU E5-2650 v3 processors.

Figure 4.13 illustrates the total water rate of all 100 models for high-fidelity, data-driven proxy and upscaled proxy cases. It can be seen that the response of the high-fidelity and data-driven models have a reasonably good agreement for both mean and individual realization water rates. Whereas, the upscaled model rates matched worse, with a distinct delay in the water-breakthrough [9]. The average error between data-driven and reference water rate throughout the simulation of 20 00 days is 3.4%, while error for the upscaled model is 14.9%.

Next, we complicated the problem for the data-driven proxy model by removing the prior knowledge of relative permeability curves. Instead, a single region of nonlinear modifiers was added to the vector of regression parameters. The initial guess for nonlinear parameters was fixed to values substantially different from the reference physics.

The results for this case are shown in figure 4.14. Regression of some model realizations was

[8] SCAL stands for special core analysis laboratory

[9] Delay in the water-breakthrough can be seen in Appendix A.4. The sharp peaks that show a 100% error indicating that water-breakthrough was not represented correctly

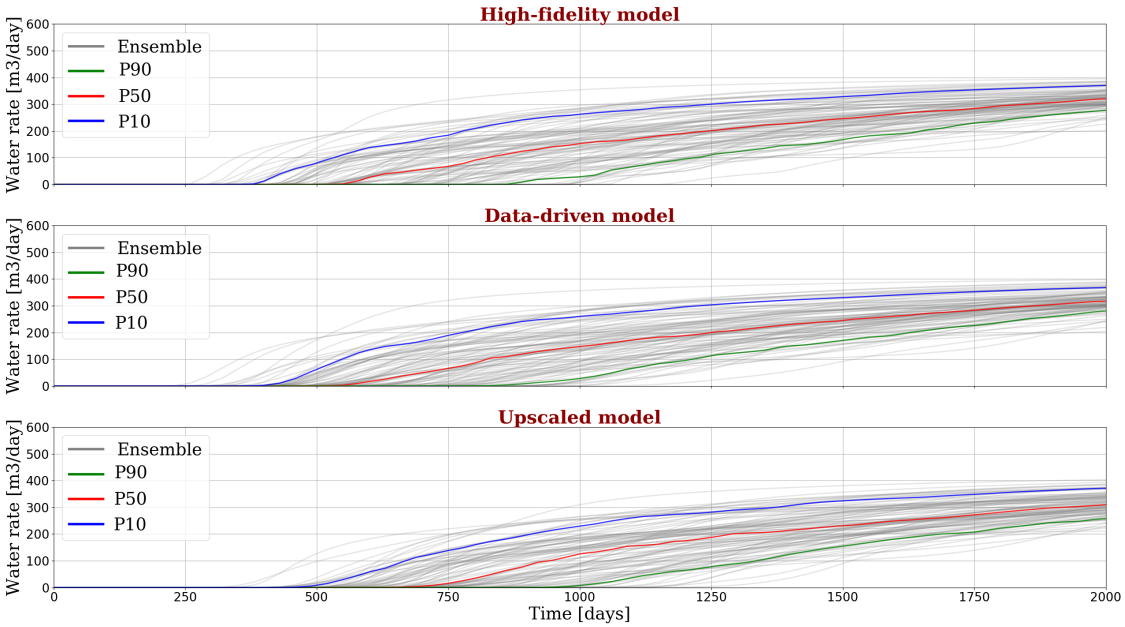


Figure 4.13: The total water rate for the high-fidelity/reference model with the size of the 100x100 grid block for the hundred Flumy realizations, together with data-driven, and upscaled models (10x10) response. Modifiers: linear and well. Grey line indicates a rate from the single model realization, whereas the red, blue and green lines indicates quantile response of the ensemble i.e the P10,P50 and P90

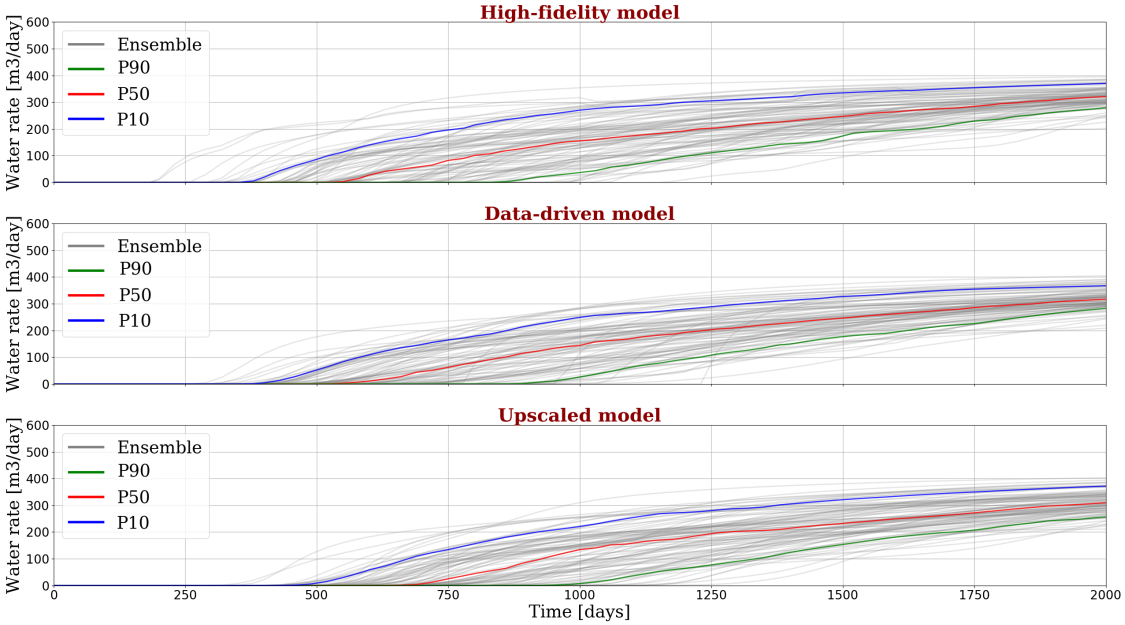


Figure 4.14: The total water rate for the high-fidelity/reference model with the size of the 100x100 grid block for the hundred Flumy realizations, together with data-driven, and upscaled models (10x10) response. Modifiers: linear, nonlinear and well

not so successful in this case (those were removed from the comparison), however, in general, the data-driven model still gives more accurate results compared to conventional upscaling. The mean error of the data-driven approach increased to 6.6%, which is still significantly lower than that for the upscaled model.

Then, the same test was performed for a more complicated model ensemble build with MPS

stochastic modeling approach. Results are shown in figure 4.15. The mean error between

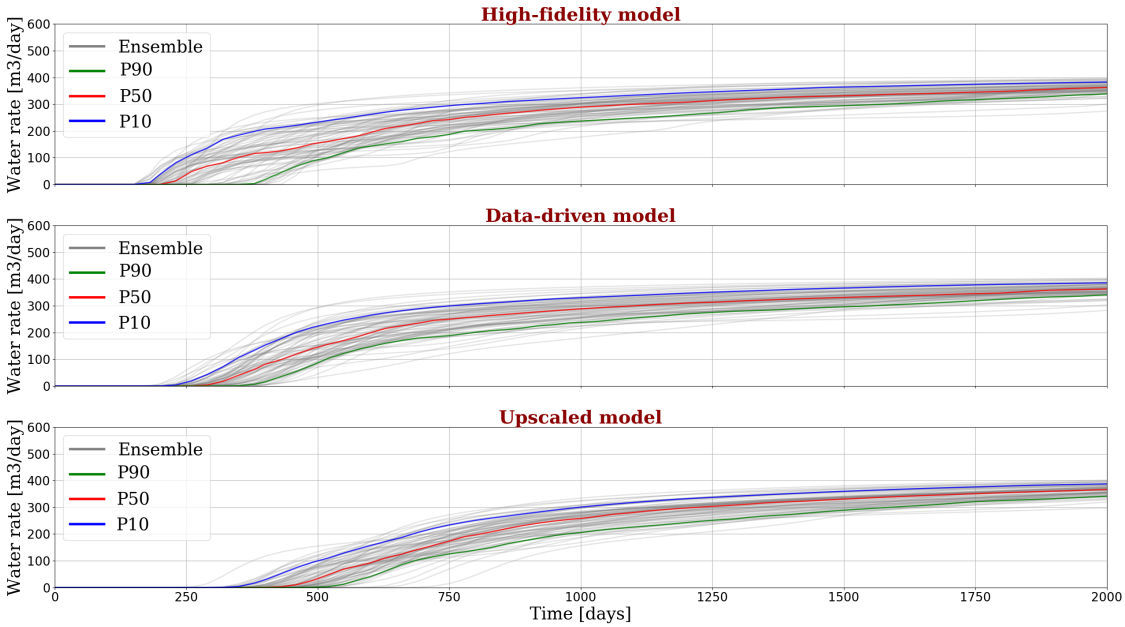


Figure 4.15: the total water rate for the high-fidelity/reference model with the size of the 100x100 grid block for the hundred MPS realizations, together with data-driven, and upscaled models (10x10) response. Modifiers: linear and well

total water rates for both upscaled and data-driven models increased to 19.7 % and 7.4 % respectively. It is an expected result as it is much more difficult to find a value for the effective property on a coarse scale that will accurately represent fine-scale features (*e.g.*, small and poorly connected channels can be seen in figure 3.3 (c). On the contrary, the channels in the Flumy model overlap each other creating more distinct and rough flow paths, which are easier to capture on a coarse scale. However, the overall accuracy of the data-driven proxy model is still significantly higher than that for the upscaled proxy model. It confirms the applicability of the data-driven approach for uncertainty quantification analysis when a reliable and accurate high-fidelity model is not available.

5

Conclusion

The petroleum industry has generated thousands of terabytes of data. Unfortunately, only a limited part of it is used efficiently due to poor data organization and the absence of efficient data-driven approaches. With the advances in data analysis and computing capabilities, data-driven models become more widely used in all aspects of reservoir management, including history matching, optimization, uncertainty quantification, and production forecast. There is a large volume of published studies describing the successful utilization of data-driven models as an analytical tool for the oil and gas industry application. However, many of them significantly relax governing physics or even are treated as black-box approaches, hence sometimes perceived sceptically.

In this work, a physics-based data-driven framework was developed based on the DARTS platform. It showed excellent simulation performance crucial for efficient model optimization. The framework was examined on historical production data from a real brown field, where satisfactory production forecast results were obtained. Various regression strategies have been analyzed laying the foundation for sequential regression and gradual refinement of the model. The resulting strategy was evaluated on the two synthetic data sets and showed exceptional prediction accuracy for a significantly reduced model size. Properly organized multi-process gradient computations can efficiently leverage computing capabilities of modern multi-core architectures and further increase regression performance.

Finally, it was concluded that the reliability of the recorded historical data of the real field is under considerable deliberation. The comparison of regression quality for real and synthetic data suggests the utmost importance of reliable measurements for the successful construction of a proxy model. Besides, the implementation of the machine learning approaches can considerably benefit in temporal and spatial refinement automatization.

To sum up, data-driven methods offer a great opportunity for the industry to get a fast and reliable framework for solving many subsurface engineering problems, as was partly demonstrated in this work. The rising popularity of those techniques indicates their full potential in a modern data-dependent world. There is still a wide range of methods that can be coupled with data-driven approaches to increase prediction capabilities and incorporate data-driven models into widely accepted engineering practice.

6

Future Work

Data-driven problems can have a fascinatingly vast spectrum of possible solutions. Many things were implemented and analyzed throughout this thesis, however, there is still a large potential for further improvement in framework efficiency and accuracy.

To get a full-scale understanding of the framework capabilities and drawbacks, a set of real fields should be examined. Ideally, this set should contain fields with a considerably different extent, physics, and field life. Spatial clustering was done manually and with no qualitative principle. Hence, there is a possibility to implement more advanced automatic spatial and temporal clustering approaches based on the most promising machine learning practices.

All examples in the thesis used binary compositional formulation under the assumption that real field gas rates and advanced compositional effects are negligible to account for. Therefore, it would be interesting to test this framework with more components.

Also, the proposed data-driven framework use gradient-based optimization algorithms, where most time is spent on gradient evaluation. Multi-process gradient computation helps to reduce the time spent on those computations. Alternatively, utilization of the adjoint methods in the evaluation of gradients will provide an accurate and much more computationally efficient optimization.

A

Appendix

A.1. Intervals in temporal discretization

There are a total of four temporal discretization cases analyzed thought this work. In the first case, the schedule was discretized equally into four intervals. Cases with five and seven intervals were based on the manual analysis of the field events and supported by the k-means clustering result. The interval length of the intervals can be seen in table A.1. In the last case, the schedule was equally discretized into a 3-year interval.

Interval Nr.	1	2	3	4	5	6	7
5 intervals [years]	16	12	8	4	9		
7 intervals [years]	16	4	7	8	3	4	7

Table A.1: Distribution of interval lengths for the two cases based on reservoir dynamic analysis. Reservoir started to produce water within the first 16 years of production, therefore it was useful to set the first interval to the same length, as model regression was reasonable simple for that period.

A.2. Objective Function Sensitivity

Here we perform the sensitivity analysis of the optimization results based on different objective functions used in this study.

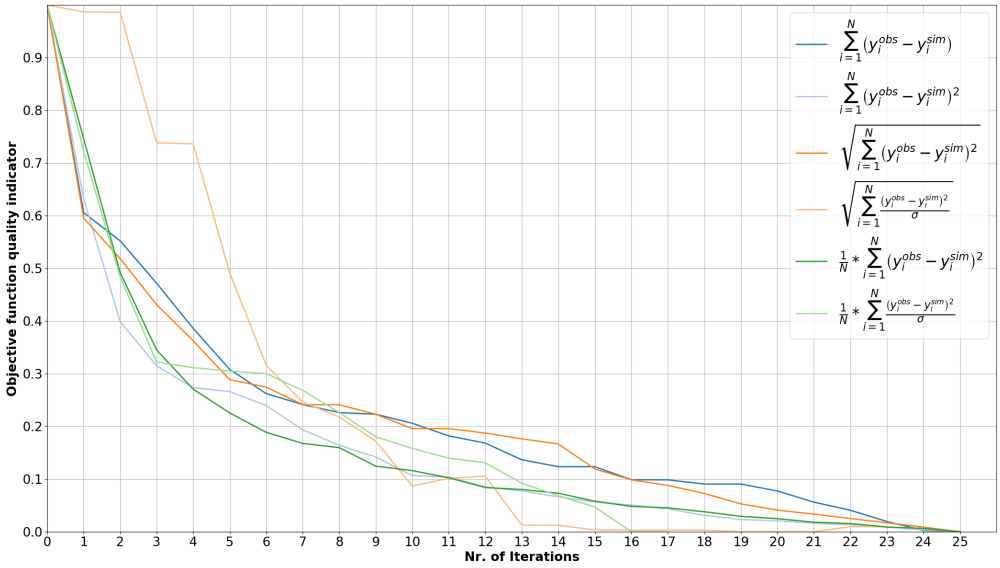


Figure A.1: Effect of six different objective functions on optimisation convergence: 1 - simple error, 2 - squared error, 3 - L_2 norm, 4 - weighted L_2 norm by a standard deviation of well error in whole training interval, 5 - squared error normalised by N, 6 - squared error normalised by N and weighted by standard deviation. The objective function quality indicator is normalized objective function value.

A.3. Brugge model cumulative error

Here we show a cumulative error in production rate evaluation of Brugge model at the end of the training stage.

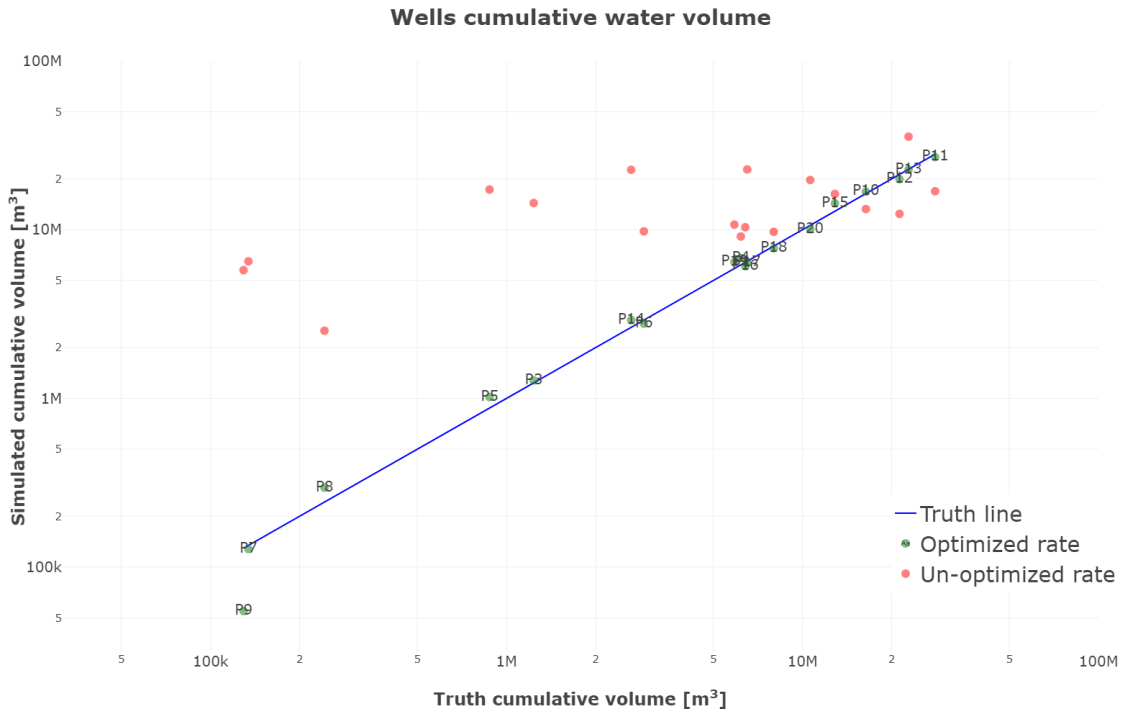


Figure A.2: Brugge model cumulative volume at the end of training period of 49 years. Same outliers are seen as in forecast period.

A.4. Ensemble-based error

Below we present an ensemble errors of upscaling and data-driven approaches for the Flummy-based ensemble by the end of the training stage.

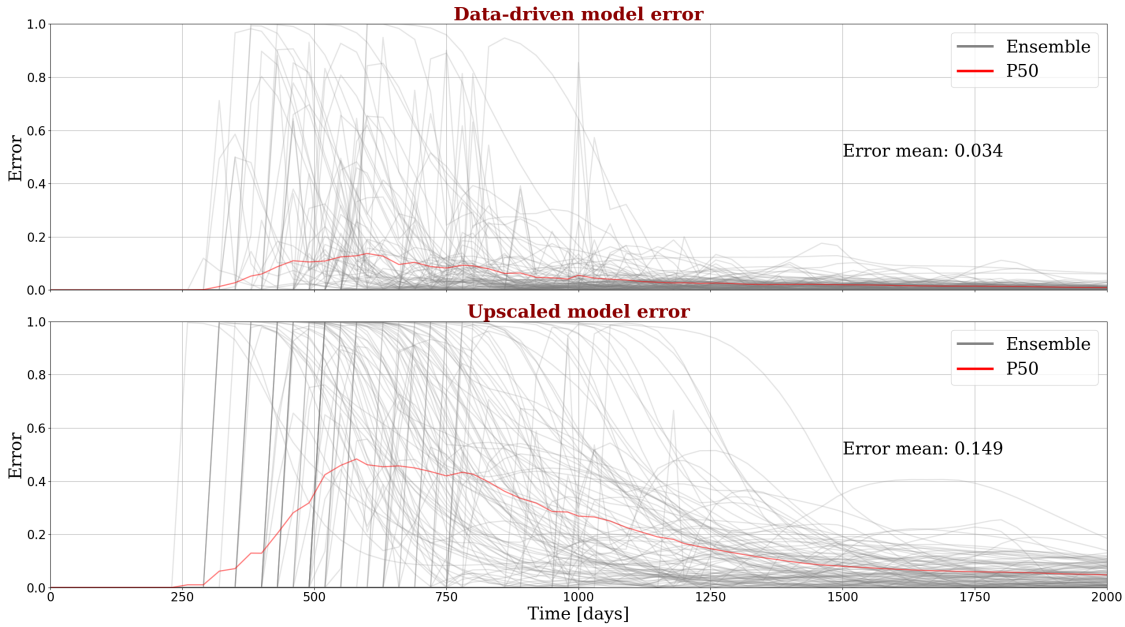


Figure A.3: The error between reference model response and data-driven and upscaled models. Sharp peaks, when the error is 100% indicate water breakthrough timing miss-match.

Bibliography

[1] J. Ahrens, B. Geveci, and C. Law. Paraview: An end-user tool for large data visualization. *The visualization handbook*, 717, 2005.

[2] A. Albertoni and L.W. Lake. Inferring interwell connectivity only from well-rate fluctuations in waterfloods. *SPE Reservoir Evaluation and Engineering*, 6(1):6–15, 2003.

[3] R.P. Batycky, M.J. Blunt, and M.R. Thiele. A 3d field-scale streamline-based reservoir simulator. *SPE Reservoir Engineering (Society of Petroleum Engineers)*, 12(4):246–253, 1997.

[4] M.A. Cardoso, L.J. Durlofsky, and P. Sarma. Development and application of reduced-order modeling procedures for subsurface flow simulation. *International Journal for Numerical Methods in Engineering*, 77(9):1322–1350, 2009.

[5] K.H. Coats, W.D. George, B.E. Marcum, and Chieh Chu. Three-dimensional simulation of steamflooding. *Soc Pet Eng AIME J*, 14(6):573–592, 1974.

[6] D.A. Collins, L.X. Nghiem, Y.-K. Li, and J.E. Grabenstetter. Efficient approach to adaptive-implicit compositional simulation with an equation of state. *SPE Reservoir Engineering (Society of Petroleum Engineers)*, 7(2):259–264, 1992.

[7] DARTS. Delft advanced research terra simulator, 2019. URL `https://darts.citg.tudelft.nl`.

[8] S. De Hoop. Determination of relevant spatial scale in reservoir simulation. 2017.

[9] S. De Hoop, D.V. Voskov, F.C. Vossepoel, and A. Jung. Quantification of coarsening effect on response uncertainty in reservoir simulation. 2018.

[10] L.J Durlofsky. Upscaling and gridding of fine scale geological models for flow simulation. In *8th International Forum on Reservoir Simulation Iles Borromees, Stresa, Italy*, volume 2024, pages 1–59. Citeseer, 2005.

[11] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[12] Z. Guo, A.C. Reynolds, and H. Zhao. A physics-based data-driven model for history matching, prediction, and characterization of waterflooding performance. *SPE Journal*, 23(2):367–395, 2018.

[13] F.E. Jansen and M.G. Kelkar. Non-stationary estimation of reservoir properties using production data. volume Omega, pages 131–138, 1997.

[14] P. Jenny, S.H. Lee, and H.A. Tchelepi. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *Journal of Computational Physics*, 187(1):47–67, 2003.

[15] M. Karimi-Fard, L.J. Durlofsky, and K. Aziz. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE Journal*, 9(2):227–236, 2004.

[16] M.J. King, D.G. MacDonald, S.P. Todd, and H. Leung. Application of novel upscaling approaches to the magnus and andrew reservoirs. volume 2, pages 133–148, 1998.

[17] D. Kraft. A software package for sequential quadratic programming. *Forschungsbericht-Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*, 1988.

[18] P. Lerlertpakdee, B. Jafarpour, and E. Gildin. Efficient production optimization with flow-network models. *SPE Journal*, 19(6):1083–1095, 2014.

[19] Kok-Thye Lim. A new approach for residual and jacobian arrays construction in reservoir simulators. *SPE Computer Applications*, 7(04):93–96, 1995.

[20] G.P Louis, H Borouchaki, F Alauzet, P. Laug, and A Loseille. Mesh generation and mesh adaptivity: theory and techniques. *Encyclopedia of Computational Mechanics Second Edition*, pages 1–51, 2018.

[21] M.L. Michelsen. The isothermal flash problem. part ii. phase-split calculation. *Fluid Phase Equilibria*, 9(1):21–40, 1982.

[22] S. et al. Mohaghegh. Artificial intelligence and data mining: enabling technology for smart fields. *The Way Ahead*, 5(03):14–19, 2009.

[23] D. W. Peaceman. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *Society of Petroleum Engineers journal*, 23(3):531–543, 1983.

[24] M.J.D. Powell. Direct search algorithms for optimization calculations. *Acta Numerica*, 7:287–336, 1998.

[25] A.H.G. Rinnooy Kan and G.T. Timmer. Stochastic global optimization methods part ii: Multi level methods. *Mathematical Programming*, 39(1):57–78, 1987.

[26] P Sujin. Dataset resampling filters, 2016. URL `https://blog.kitware.com/dataset-resampling-filters/`.

[27] K. Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization*, 12(2):555–573, 2002.

[28] O. Volkov and D.V. Voskov. Effect of time stepping strategy on adjoint-based production optimization. *Computational Geosciences*, 20(3):707–722, 2016. doi: 10.1007/s10596-015-9528-1.

[29] D. Voskov and M. Khait. Spe-182685-ms adaptive coarsening in physical representation for the robust thermal-compositional simulation. pages 1610–1625, 2017.

[30] D Voskov, Y Zhou, and O Volkov. Technical description of ad-gprs. *Energy Resources Engineering, Stanford University*, 2012.

[31] D.V. Voskov. Operator-based linearization approach for modeling of multiphase multi-component flow in porous media. *Journal of Computational Physics*, 337:275–288, 2017.

[32] D.V. Voskov and H.A. Tchelepi. Comparison of nonlinear formulations for two-phase multi-component eos based simulation. *Journal of Petroleum Science and Engineering*, 82-83:101–111, 2012.

[33] A.A. Yousef, P. Gentil, J.L. Jensery, and L.W. Lake. A capacitance model to infer interwell connectivity from production and injection rate fluctuations. pages 323–341, 2005.

[34] H. Zhao, Z. Kang, X. Zhang, H. Sun, L. Cao, and A.C. Reynolds. Insim: A data-driven model for history matching and prediction for waterflooding monitoring and management with a field application. volume 1, pages 431–461, 2015.

[35] D.I. Zubarev. Pros and cons of applying proxy-models as a substitute for full reservoir simulations. volume 5, pages 3234–3256, 2009.