Discrete well affinity data-driven proxy model for production forecast

Tian, X.; Blinovs, A.; Khait, M.; Voskov, D.

DOI
10.2118/205489-PA

Publication date
2021

Document Version
Final published version

Published in
SPE Journal

Citation (APA)
Tian, X., Blinovs, A., Khait, M., & Voskov, D. (2021). Discrete well affinity data-driven proxy model for production forecast. *SPE Journal*, *26*(4), 1876-1892. https://doi.org/10.2118/205489-PA

Discrete Well Affinity Data-Driven Proxy Model for Production Forecast

X. Tian, A. Blinovs, and **M. Khait,** TU Delft; and **D. Voskov*,** TU Delft and Stanford University

Summary

A physics-based data-driven model is proposed for forecasting of subsurface energy production. The model fully relies on production data and does not require any in-depth knowledge of reservoir geology or governing physics. In the proposed approach, we use the Delft Advanced Reservoir Terra Simulator (DARTS) as a workhorse for data-driven simulation. DARTS uses an operator-based linearization technique that exploits an abstract interpretation of physics benefiting computational performance. The physics-based data-driven model is trained to fit data increasing the fidelity of the model forecast and reflecting significant changes in reservoir dynamics or physics over its history. The model is examined and validated for both synthetic and real field production data. We demonstrate that the developed approach is capable of providing accurate and reliable production forecast on a daily basis, even if the exact geological information is not available.

Introduction

Computer technologies are progressing rapidly. Computational capacities that are currently available provide an opportunity for many subsurface applications to perform complex numerical simulations of high-resolution 3D geocellular computer models. Predictions obtained from such models are an important factor governing efficient reservoir management and decision making. The models describe complex geological features through a set of gridblocks and associated rock and fluid properties. However, in many cases, the reliability of geological information is questionable or even not available. Although it is possible to develop a high-fidelity model on a reliable basis of reservoir geology, a high-resolution computer model can exceed a few million blocks and can take hours or even days to simulate. It is still not computationally feasible to perform history matching or reservoir-development optimization at such resolution because it involves a large number of simulation runs.

Different methods have been developed to overcome the issue. Those methods fall into two categories: simplified full-field models or data-driven approaches. Methods such as upscaling, multiscale 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 gridblocks with assigned effective properties aiming to replicate a high-fidelity model response (Durlofsky 2005). Multiscale methods are designed to efficiently capture the large-scale behavior of the solution without resolving all the small-scale features (Hou and Wu 1997; Jenny et al. 2003). The streamline simulation method (Batycky et al. 1997; Datta-Gupta and King 2007) is a Eulerian-Lagrangian approach with implicit-pressure/explicit-saturation time approximation. In this approach, a full 3D transport solution is translated into a set of 1D equations solved along streamlines.

All methods in the first category require an underlying geological characterization as a basis for model construction. However, there are many cases when this information is not available or its reliability is questionable. Does it mean we cannot solve optimization or history-matching problems efficiently? Methods from the second category help to resolve this issue. Data-driven methods assume the building of a proxy model with a sufficient amount of degrees of freedom to accurately mimic a realistic reservoir response according to its calibration to the production data. With frequent, sustained, and accurate data being fed into a reliable regression framework, data-driven models can provide an accurate forecast for the given reservoir.

There are many data-driven approaches available in the industry, including the statistical data-driven model proposed by Jansen and Kelkar (1997); reduced-order models (Cardoso et al. 2009); the capacitance/resistance model (Albertoni and Lake 2003; Yousef et al. 2005; Holanda et al. 2018); the flow-network model (Lerlertpakdee et al. 2014; Ren et al. 2019; Borregales et al. 2020; Kiærr et al. 2020) where a complex 3D flow is represented as a set of 1D finite-difference reservoir models; the interwell numerical-simulation model (Zhao et al. 2015, 2020) and the interwell numerical simulation model with front-tracking for 3D multilayer reservoirs (Guo and Reynolds 2019), which applies a new Riemann solver derived from a convex-hull method that helps to solve the Buckley-Leverett problem with gravity and allows for the inclusion of wells with arbitrary trajectories with multiple perforations; and many other alternative methods that rely on artificial intelligence (Mohaghegh 2009) and data fitting (Zubarev 2009). All of these approaches have specific advantages and limitations.

In this study, combining advantages of methods from both categories, we develop and validate a framework capable of performing an accurate forecast using the historical field data while respecting underlying physical processes at the same time. Moreover, the exact grid parameters or detailed reservoir geology are no longer the prerequisites of reservoir simulation and prediction using the data-driven model. We focus our study on general applicability and validations of the proposed data-driven model methodology for realistic reservoir-simulation problems with limited heterogeneity and gravitational effects. It is achieved through the utilization of the operator-based linearization (OBL) technique (Voskov 2017) as implemented in the DARTS framework.

Methodology

In this section, we describe the main components of the proposed data-driven physics-based simulation framework. It includes the generation of a connectivity graph, governing equations, nonlinear solution, and training of the model.

Connectivity Graph for Proxy Model. To connect spatial well locations with production data, we need to represent the domain of interest in a discrete form. The geometrical discretization of the reservoir is typically performed using the control-volume partitioning. For our proxy model, we use unstructured partitioning and finite-volume discretization suggested by Karimi-Fard et al. (2004). This

---

*Corresponding author; email: d.v.voskov@tudelft.nl

Original SPE manuscript received for review 2 November 2020. Revised manuscript received for review 15 February 2021. Paper (SPE 205489) peer approved 16 March 2021.

results in a spatial connectivity graph that forms a discrete representation of the proxy reservoir model in terms of connections between control volumes and associated transmissibilities (Lim 1995). In the proposed data-driven approach, we adopt this technique for the partitioning of the reservoir domain with a coarse resolution.

The discretized model is defined using boundaries that are gridded utilizing the 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). The automatic open-source meshing software package Gmsh (Geuzaine and Remacle 2009) was used in this work for the model gridding. Exact grid parameters and rock properties (e.g., permeability or exact layer geometry) are considered to be unavailable; therefore, the corresponding initial guesses for control parameters are computed using averaged values derived from initial evaluations of the reservoir. When those values are not known, they can be estimated using an analog field or common sense. Typical unstructured mesh with corresponding well locations are shown in **Fig. 1.**

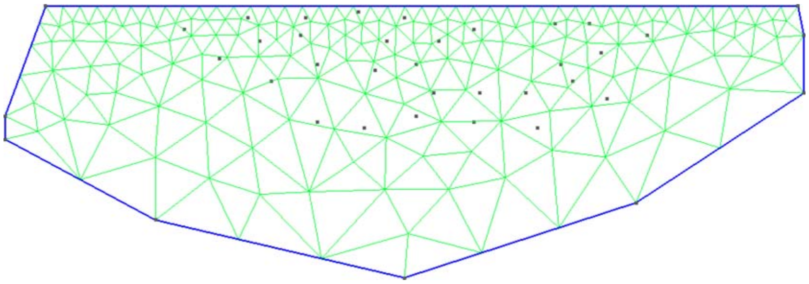


Fig. 1—Typical grid for connectivity graph generated for the proxy model with corresponding border lines and well positions.

Governing Equations. In this subsection, we describe the set of governing equations required for a general compositional numerical simulation. Transport equations for an isothermal system containing n_c components and n_p phases can be written as

$$\frac{\partial}{\partial t}\left(\phi \sum_{j=1}^{n_p} x_{cj}\rho_j S_j\right) + \nabla \sum_{j=1}^{n_p} x_{cj}\rho_j \mathbf{v}_j + \sum_{j=1}^{n_p} x_{cj}\rho_j q_j^{\star} = 0, \quad c = 1, \ldots, n_c, \quad \text{(1)}$$

where phase velocity is described by Darcy's law,

$$\mathbf{v}_j = -\mathbf{K}\frac{k_{rj}}{\mu_j}(\nabla p_j - \gamma_j \nabla d), \quad \text{(2)}$$

where ϕ is rock porosity, x_{cj} is mole fraction of component c in phase j, S_j is phase saturation, ρ_j is phase molar density, $\mathbf{v}_j$ is phase velocity, $q_j^{\star}$ is phase rate per unit volume, $\mathbf{K}$ is permeability tensor, k_{rj} is relative permeability, μ_j is phase viscosity, p_j is pressure of phase j, γ is gravity term, and d is depth (positive downward).

Eq. 1 can be written in a discrete form by applying the finite-volume discretization in space and backward Euler approximation in time,

$$\mathbf{g} = \frac{V}{\Delta t}\left[\phi \sum_j \mathbf{x}_j \rho_j S_j - \left(\phi \sum_j \mathbf{x}_j \rho_j S_j\right)^n\right] - \sum_{l \varepsilon L}\left(\sum_j \mathbf{x}_j^l \rho_j^l T_j^l \Delta\psi^l\right) + \sum_j \rho_p \mathbf{x}_j q_j = 0, \quad \text{(3)}$$

where V is the volume of a control volume, L is the interface connecting a control volume with another gridblock, T is the transmissibility of interface L, and $\Delta\psi^l$ is the pressure potential between two gridblocks. For simplicity, gravity and capillarity are neglected in this study. The fully implicit time approximation requires the flux term to be defined according to the nonlinear unknowns at the new timestep $(n+1)$, which introduces nonlinearity to the system of governing equations. We use the overall molar formulation proposed in Collins et al. (1992). In this formulation, the primary nonlinear unknowns are pressure p and overall composition z_c; therefore, the physical state $\boldsymbol{\omega}$ is completely defined by these variables. The derivatives of all properties in Eq. 3 with respect to nonlinear unknowns can be found by applying several closing assumptions.

Next, the Jacobian and the residual are constructed during the linearization stage. It is required by the Newton-Raphson method, where at each nonlinear iteration, the following linear system of equations is solved,

$$\frac{\partial \mathbf{g}(\boldsymbol{\omega}^k)}{\partial \boldsymbol{\omega}^k}(\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k) = -\mathbf{g}(\boldsymbol{\omega}^k), \quad \text{(4)}$$

where $\frac{\partial \mathbf{g}(\boldsymbol{\omega}^k)}{\partial \boldsymbol{\omega}^k}$ is the Jacobian matrix containing the derivatives with respect to primary unknowns, $\boldsymbol{\omega}$ is the vector of nonlinear unknowns $\boldsymbol{\omega} = \{p, z_c\}$, k is the nonlinear iteration step, and $\mathbf{g}$ is the residual. The conventional nonlinear-solution approach involves evaluation and storage of all properties and their derivatives with respect to the nonlinear unknowns, which can be challenging for analytical derivatives of complex physical problems or can introduce certain performance overhead in the case of numerical or automatic differentiation. A new strategy for linearization was proposed in Voskov (2017) and will be briefly described in the next subsection.

Operator-Based Linearization. Eq. 3 can be written in a compact form as

$$\mathbf{g}(\boldsymbol{\omega}) = \frac{V(\xi)\phi_0(\xi)}{\Delta t}[\alpha_c(\boldsymbol{\omega}) - \alpha_c(\boldsymbol{\omega}_n)] + \sum_l \beta_c^l(\boldsymbol{\omega}) T_{ab}(\boldsymbol{\xi})(p^a - p^b) + \gamma_c(\boldsymbol{\omega}, \boldsymbol{\xi}, \boldsymbol{\theta}) = 0, \quad \text{(5)}$$

where $\boldsymbol{\omega}$ defines the physical state, $\boldsymbol{\xi}$ represents spatial variation, ϕ_0 is initial porosity, and T_{ab} is the geometric part of transmissibility. All involved operators are defined as

$$\alpha_c(\boldsymbol{\omega}) = \left[1 + c_r(p - p_{ref})\right]\sum_j x_{cj}\rho_j S_j, \quad (6)$$

$$\beta_c(\boldsymbol{\omega}) = \sum_p x_{cj}\frac{k_{rj}}{\mu_j}\rho_j, \quad (7)$$

$$\gamma_c(\boldsymbol{\omega}, \boldsymbol{\xi}, \boldsymbol{\theta}) = \sum_j \rho_j x_{cj} q_j(\boldsymbol{\omega}, \boldsymbol{\xi}, \boldsymbol{\theta}), \quad (8)$$

where c_r is the rock compressibility, p_{ref} is initial reservoir pressure, and $\boldsymbol{\theta}$ is the vector of well controls.

In this form, the nonlinear system has a simplified description in terms of operators α_c and β_c, which depend only on the physical state and are valid at any spatial location of a reservoir. When several regions for pressure/volume/temperature properties or special-core-analysis laboratory properties are introduced in a reservoir, several sets of the operators can be used accordingly. The values of operators are uniquely determined in the parameter space, the dimensionality of which is defined by the set of nonlinear unknowns p and z_c.

The OBL method suggests applying interpolation for the evaluation of both operator values and their derivatives at any point in parameter space instead of conventional direct evaluation. The approach uses a uniformly distributed mesh in parameter space. Then, operator values are only computed at a limited set of supporting points in the parameter space and are saved in a table. When assembling Jacobian and residual, we use the operator values from the table as supporting points. For every state , the multilinear interpolation is applied to calculate the operator values and derivatives using the values of the supporting points. In this case, the expensive property and its derivative evaluations required by the physical model are limited to a few supporting points in parameter space, which significantly reduces the computational time. Moreover, operators are evaluated and added to the table adaptively only at those supporting points in the discrete parameter space, which are required to perform interpolation in the course of simulation (Khait and Voskov 2018a). This approximated physical description allows for increased simulation performance, essential for gradient-optimization problems, while the approximation error remains under control.

An extensive study on applications of OBL for various subsurface problems can be found in Khait and Voskov (2017, 2018b), Kala and Voskov (2020), and Wang et al. (2020), with practical suggestions on the choice of the range and resolution for interpolation tables. We use the solution provided by the OBL approach for solving both high-fidelity and proxy-forward models. In this study, the number of uniformly distributed interpolation points is 64 for both the pressure and the mole fraction of the component. The interpolation ranges of pressure and mole fraction are [0, 200] and [0, 1], respectively.

Training of Proxy Model. The parameters of a model that are changed during the training stage are called control variables. A gradient-based optimization algorithm adjusts the control variables to ensure that the data-driven proxy model response matches the "true" response based on the historical recorded data, although it can have the potential of overfitting. In this study, we take the response of the high-resolution model as the observed "true" response, and the data-driven model will be trained to match this response.

Model training is performed with the objective function J

$$J(\mathbf{u}) = \frac{1}{2}\left[\mathbf{q}(\mathbf{u}) - \mathbf{q}^{\text{obs}}\right]^T \mathbf{C}_D^{-1}\left[\mathbf{q}(\mathbf{u}) - \mathbf{q}^{\text{obs}}\right], \quad (9)$$

where $\mathbf{u}$ is the control-variable vector (e.g., transmissibility, well index, and so forth), $\mathbf{q}(\mathbf{u})$ is the vector of production/injection rates from the proxy model (model response), $\mathbf{q}^{\text{obs}}$ is the vector of observed rates, and $\mathbf{C}_D$ is the covariance matrix. The covariance matrix is derived from the observed rates of the reservoir-model ensemble. The training is performed through the solution of the constrained optimization problem, which can be formulated as

$$\min_{\mathbf{u}\in R^n} J(\mathbf{u}), \quad \mathbf{d}(\mathbf{u}) \le 0, \quad (10)$$

where $\mathbf{d}(\mathbf{u})$ corresponds to the constraints vector. These constraint vectors contain the minimum and maximum of the control variables, which define the lower and upper bounds of this optimization problem. To solve the constrained-optimization problem defined by Eq. 10, a gradient-based optimization with the implementation of the sequential least-square quadratic programming (SLSQP) algorithm was used (Kraft 1988). The reason that we choose the SLSQP approach instead of the family of Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithms is because of our experience. In our extensive tests, the Limited-memory BFGS algorithm for Bound constrained minimization (L-BFGS-B) implementation in the open-source SciPy library (scientific tools for Python) did not demonstrate very stable behavior in proxy model training, while SLSQP performed more robustly. The SLSQP algorithm requires evaluation of the gradients of the objective function with respect to control variables. Next, we will describe two strategies to compute these gradients.

Numerical Gradients. The most generic and easy-to-implement approach for the calculation of objective-function gradients with respect to control variables is the numerical gradients approach. It yields the following form using the forward Euler approximation,

$$\frac{\partial J}{\partial u_k} = \frac{J(\mathbf{u} + \delta_k\varepsilon) - J(\mathbf{u})}{\varepsilon} + O(\varepsilon), \quad (11)$$

where δ_k is the Dirac delta function. The disadvantage of the evaluation of the numerical gradient is that it lacks robustness if the choice of ε is inappropriate. Because each derivative requires a forward simulation run, the numerical gradients method can be computationally expensive with the increase of the degrees of freedom. However, the numerical gradients approach is straightforward and easy to implement because one can directly take the response of the simulator to calculate the gradients without modifying the source code. As for the low efficiency of calculating gradients, it can be tackled to a certain extent with the parallel implementation of numerical-gradient evaluation. This procedure is embarrassingly parallel because it boils down to independent evaluation of multiple (one for each optimization parameter) objective function values.

For parallel implementation of numerical gradients evaluation, we used a Python process pool through the multiprocessing Python package. Each worker process from the pool ran a single simulation, calculated the objective-function value, and returned it to the parent process. Therefore, gradient evaluation speeds up on any system with one or several multicore processors. We also explore another approach using adjoint gradients, explained next.

Adjoint Gradients Formulation. The main idea of the adjoint method is introducing a Lagrange multiplier $\boldsymbol{\lambda}$ to form a new augmented objective function $\overline{\mathcal{J}}$ that shares the identical extrema with the original objective function J. The augmented objective function can be written as

$$\overline{\mathcal{J}}(\boldsymbol{\omega}, \boldsymbol{u}, \boldsymbol{\lambda}) = J(\boldsymbol{\omega}, \boldsymbol{u}) + \boldsymbol{\lambda}^{\mathrm{T}} g(\boldsymbol{\omega}, \boldsymbol{u}), \quad (12)$$

where $\boldsymbol{\lambda}^{\mathrm{T}}$ is the transposed form of Lagrange multipliers. The extrema of Eq. 12 locate either at the boundary of the feasibility region or at stationary points. For the latter case, the first-order derivatives with respect to $\boldsymbol{\lambda}$, $\boldsymbol{\omega}$, and $\boldsymbol{u}$ should be equal to zero, which leads to the set of equations

$$\overline{\mathcal{J}}_{\boldsymbol{\lambda}} = g(\boldsymbol{\omega}, \boldsymbol{u}) = 0, \quad (13)$$

$$\overline{\mathcal{J}}_{\boldsymbol{\omega}} = \boldsymbol{\lambda}^{\mathrm{T}} g_{\boldsymbol{\omega}}(\boldsymbol{\omega}, \boldsymbol{u}) + J_{\boldsymbol{\omega}}(\boldsymbol{\omega}, \boldsymbol{u}) = 0, \quad (14)$$

$$\overline{\mathcal{J}}_{\boldsymbol{u}} = \boldsymbol{\lambda}^{\mathrm{T}} g_{\boldsymbol{u}}(\boldsymbol{\omega}, \boldsymbol{u}) + J_{\boldsymbol{u}}(\boldsymbol{\omega}, \boldsymbol{u}) = 0, \quad (15)$$

where the notations with subscripts $\boldsymbol{\lambda}$, $\boldsymbol{\omega}$, and $\boldsymbol{u}$ mean the derivatives with respect to the corresponding variables. As can be seen from Eq. 13, the derivative of $\overline{\mathcal{J}}$ with respect to $\boldsymbol{\lambda}$ is actually the governing equation of the reservoir, so that this equation is already satisfied. Next, $\boldsymbol{\lambda}$ can be computed using Eq. 14. After applying the values of $\boldsymbol{\lambda}$ in Eq. 15, the adjoint gradient can be obtained and passed to the optimizer to perform history matching.

Comparisons of the accuracy and performance of both methods can be found in Appendix A. To ensure that the gradient optimization process stays in the physical range, a large penalty term was imposed for nonphysical regions. Moreover, the optimization was penalized whenever the nonlinear convergence of the proxy model was not reached because of the nonphysical combination of parameters. Also, the scaling of the objective function and vectors of optimization parameters was implemented in the training procedure to preserve reliable performance.

Governing Relations for Control Variables

Because the training is performed by means of constrained optimization, every control variable is bounded by the minimum and maximum values. Using the bounds, every control variable is scaled to the interval $[0, 1]$ because some of the regression algorithms are sensitive to the scale of a problem. Here, we briefly describe how the initial guess and constrained intervals for control variables can be calculated using the available data and how they are included in the data-driven model.

Rock-Fluid-Property Variables. We used a modified Brooks-Corey model (Brooks and Corey 1964) to derive rock and fluid property variables for a multiphase-flow representation in the data-driven proxy model. The modified Brooks-Corey model can be expressed as

$$k_{ro} = k_{ro}^{e}\left(1 - S_w^*\right)^{n_o}, \quad (16)$$

$$k_{rw} = k_{rw}^{e}\left(S_w^*\right)^{n_w}, \quad (17)$$

$$S_w^* = \frac{S_w - S_{wc}}{1 - S_{wc} - S_{or}}, \quad (18)$$

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

The relative permeability is included in the β_c operator, where it is multiplied by the $\frac{\rho_j}{\mu_j}$ term of the corresponding phase j, as can be seen in Eq. 7. Therefore, the vector of six rock-fluid property variables was defined as

$$v_n = \{S_{or}, S_{wc}, n_o, n_w, k_{rw}^e \rho_w/\mu_w, k_{ro}^e \rho_o/\mu_o\}, \quad (19)$$

subjected to the constraints

$$v_{n,\min} = \{0.0, 0.0, 0.00001, 0.00001, 100, 10\}, \quad (20)$$

$$v_{n,\max} = \{0.49, 0.49, 5, 5, 3000, 2000\}. \quad (21)$$

The first four parameters are unitless; the units of the last two are $[s \times m^{-2}]$. These constraints were obtained from the physical interpretation of control variables.

Reservoir-Property Variables. In reservoir simulation, the general unstructured grid is usually characterized by a spatial connectivity graph represented as a connection list (Lim 1995). It involves the specification of the connections between gridblocks and associated transmissibility of those connections. The transmissibility T_{ij} between gridblocks i and j can be defined for a general unstructured grid (Karimi-Fard et al. 2004) as

$$T_{ij} = \frac{\sigma_i \sigma_j}{\sum_n \sigma_n}, \quad \sigma_i = \frac{A k_i}{D_i}, \quad (22)$$

where A is the interface area between two gridblocks, D_i is the distance from the pressure node to the interface along the line connecting two pressure nodes, and k_i is the gridblock permeability.

Transmissibility directly affects the flow dynamics in the reservoir because it is involved as a constant multiplier in the term T_{ij} of the convection operator in Eq. 5. In our approach, the reservoir property variables effectively change the transmissibilities in the proxy model. As the initial guess, they can be defined with two-point flux finite-volume discretization on an unstructured grid using initial knowledge about the reservoir properties (rock permeability) and approximate geometric parameters (thickness and net-to-gross ratio).

Note that transmissibility is a linear multiplier between the pressure drop and the flow rate. However, it still remains highly nonlinear with respect to the objective function. The reason that we adjust transmissibility instead of permeability is that the high-fidelity model used in this study is defined on a structured grid , while the proxy model uses an unstructured grid. For an unstructured grid, it is relatively easy to make the reservoir volume close to the reservoir volume of the high-fidelity model for the sake of model validation and a fair comparison. Therefore, we take the pore volume and average thickness as the bases of the proxy model, and then adjust the other properties in the course of proxy model training. Nevertheless, the porosity can also be chosen as a control variable and it can be adjusted in the course of training.

In our approach, the number of reservoir property parameters equals the number of connections. Through the regression course, the reservoir property variables were constrained by $v_{l,\min} = 1$ cp·m^3/d/bar and $v_{l,\max} = 50,000$ cp·m^3/d/bar.

Well-Property Variables. A well index was proposed by Coats et al. (1974) in the steamflood simulation to relate the gridblock pressure/rate to the wellbore flowing pressure/rate. The equation that relates a well and a reservoir gridblock under the assumption of a single-phase flow can be written as

$$q_i^w = \frac{T_i^w}{\mu}\left(P_i - P_i^w\right), \quad \quad (23)$$

where q_i^w is the well rate into (out of) the block i, T_i^w is the well index of the gridblock i intersected by a well, P_i is the gridblock pressure, and P_i^w is the well bottomhole pressure (BHP).

Once it is determined in single-phase assumptions, it is also applied to a multiphase flow. Instead of using an arbitrary initial guess of the well index, we use the Peaceman (1983) equation (Eq. 24) to calculate the well index using a high-fidelity structured grid and then apply them as the initial guess in the proxy model. Although the well-index values calculated using the Peaceman equation might not be true for the proxy model, they will be adjusted and updated in the course of training.

$$T_i^w = \frac{2\pi\Delta z\sqrt{k_x k_y}}{\ln r_o/r_w + S}, \quad \quad (24)$$

where

$$r_o = 0.28\frac{\left[\left(k_y/k_x\right)^{1/2}\Delta x^2 + \left(k_x/k_y\right)^{1/2}\Delta y^2\right]^{1/2}}{\left(k_y/k_x\right)^{1/4} + \left(k_x/k_y\right)^{1/4}}. \quad \quad (25)$$

Similar to transmissibility, a well index can be seen as a representation of the linear parameter between the pressure difference and the rate. The main difference of a well index is that it is involved in the θ_c operator, which directly relates the reservoir pressure with the wellbore pressure. Besides, the wellbore pressure can be used as either a control or a constraint in a simulation. In our study, the set of well property variables consisted of N_{well} well indices. They were constrained by a minimum value of $v_{w,\min} = 1$ cp·m^3/d/bar and maximum value of $v_{w,\max} = 1,000$ cp·m^3/d/bar in the training stage.

Validation of Proposed Methodology

We validate the proposed data-driven approach using two ensembles of high-fidelity fluvial models. For each model of each ensemble, two reduced-order models are constructed. The first model is built through the conventional flow-based upscaling (Chen and Durlofsky 2006), while the second uses the proposed data-driven approach. The high-fidelity models are considered as reference models to generate the observation true data of the oil rate. The data-driven model is then trained with the true data using constant transmissibilities and well indices as an initial guess. Next, both the results of the upscaled model and the data-driven model are compared with the high-fidelity model for the entire ensemble to check their accuracy. All models are under the assumption that fluid and rock physics are known and fixed except for transmissibility and well indices. The same set of parameters was also used for the rest of the model tests. These parameters are described in Appendix B.

Generation of Ensemble of High-Fidelity Fluvial Models. The stochastic ensemble of fluvial models was used to illustrate the accuracy of the proposed data-driven methodology. High-fidelity-model ensembles were generated by two different modeling approaches:

- Flumy® (Armines, Paris, France) (Grappe et al. 2016), or process-based models using Flumy software **(Fig. 2a)**
- Multipoint statistics (MPS) models (Strebelle and Levy 2008) (Fig. 2b)

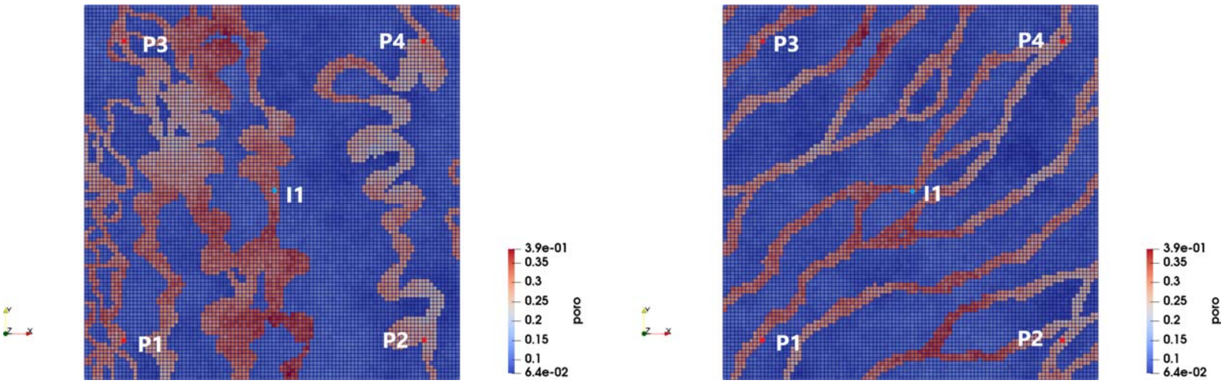


Fig. 2—Porosity distributions of a (a) typical high-fidelity model realization generated by process-based modeling approach with Flumy software and (b) stochastic modeling approach using MPS.

This results in completely different model complexity between the ensembles. Each high-fidelity ensemble model has a size of 100 × 100 grid cells (cell dimensions are 10 × 10 × 10 m). The models use a simple five-spot vertical well setup. One injector (blue dot I1 in Fig. 2) is located in the middle of the reservoir and surrounded by four producers (red dots P1, P2, P3, and P4 in Fig. 2) located at the reservoir edges. Injection wells are modeled by setting a BHP control of 250 bar plus uniformly distributed random perturbations ranging from 1 to 30 for every 200 days, and production wells are set at a fixed BHP control of 100 bar. The simulation was limited by 4,000 days. The main difference between models generated by MPS and Flumy is the main paleoflow orientation ranging from southwest/northeast to west/east. Besides, the Flumy model has a limited statistical variability compared with the MPS model (de Hoop et al. 2018).

A typical porosity distribution of the high-fidelity realizations from MPS and Flumy ensembles can be seen in Fig. 2. It is clear that the model generated by MPS is more complex because the phase can flow only through distinct channels, which are usually smaller-sized than the coarse gridblock. In contrast, the model generated by Flumy has many overlaying channels providing multiple possible flow paths. Once the coarser proxy models are trained to match the historical field data of the Flumy or MPS models, it can be seen that the trained proxy model of Flumy is easier for capturing the reservoir dynamics on a coarser scale. More details regarding the high-fidelity ensemble generation, upscaling, and simulation properties can be found in de Hoop et al. (2018).

Generation of Upscaled Proxy Models. Upscaled proxy models were generated using a global flow-based upscaling technique, which involves solving the fine-scale incompressible single-phase pressure equation and using it to obtain coarse-scale transmissibility,

$$-\nabla \cdot \left[\frac{K}{\mu}\nabla(P - \rho g)\right] = q_{\text{well}}. \tag{26}$$

Under the two-point flux approximation, Eq. 26 can be written in the discrete form, in which the coarse properties can be evaluated,

$$\left(q_x^c\right)_{i+1/2,j} = \left(T_x^c\right)_{i+1/2,j}\left(P_{i,j}^c - P_{i+1,j}^c\right), \tag{27}$$

where $\left(q_x^c\right)_{i+1/2,j}$ is the coarse flux across the interface $i + 1/2, j$, simply defined as the integrated fine-scale fluxes across the coarse interface; $\left(T_x^c\right)_{i+1/2,j}$ is the coarse transmissibility; and ($P_{i,j}^c$ and $P_{i+1,j}^c$) are the coarse pressures obtained by arithmetic averaging of the fine-scale pressures contained in each coarse-scale block, respectively. A similar approach for a flow-based upscaled well index for well in block i can be derived, given by

$$T_i^w = \frac{q_i}{P_{i,j}^c - P^w}. \tag{28}$$

The big advantage of the global flow-based upscaling technique is its computational efficiency and accuracy. However, in highly heterogeneous reservoirs, it has one downfall: The resulting transmissibility values might be large or even negative (Chen and Durlofsky 2006). In this case, we correct the value of transmissibility at the nearest bound (the upper or lower bound), and then assign new pressure values to the neighbor blocks by requiring that Darcy's law is fulfilled. However, the correction of transmissibility can cause new pressure differences between the neighbor blocks. Hence, the upscaling procedure of computing the transmissibility must be applied in an iterative fashion until the criteria are satisfied. This iterative upscaling procedure is generally used to obtain a positive definite transmissibility matrix, which is typically reached within five iterations (Holden and Nielsen 2000).

Each high-fidelity model was scaled up laterally by 100 times (10 × 10) in the *x*- and *y*-direction, which means that the upscaled model has the size of 10 × 10 grid cells (cell dimensions are 100 × 100 × 10 m). The global upscaling with specific well conditions used in the reference model was applied, and details can be found in de Hoop et al. (2018). The resulting upscaled transmissibility, porosity, and well index were used to initialize the upscaled proxy model. Well controls and simulation time were kept identical to the high-fidelity model. Moreover, the same upscaled porosity was used in the data-driven proxy model to ensure that pore volumes match between models.

Generation of Data-Driven Models. Data-driven proxy models were generated with the same grid and well configuration as for upscaled proxy models. The initial guess for transmissibilities and well indices of the data-driven proxy model was chosen as a uniform distribution of 100 and 200 $\text{cp}\cdot\text{m}^3/\text{d/bar}$.

Then, data-driven models are trained using the observation true data. The regression was limited by 100 iterations. Some cases converged before reaching the imposed maximum. The training of a single realization takes from 20 minutes to 1 hour on a single cluster node with two Intel® Xeon® (Intel Corporation, Santa Clara, California, USA) central processing units (CPUs) with E5-2650 v3 processors.

Comparison between Upscaled and Data-Driven Proxy Models. For the correct comparison of the data-driven and upscaled model response, we have ensured consistency between model volume, physical properties, and well controls. Those parameters are identical because the upscaling procedure is sensitive to the boundary conditions. We compare a stochastic response of the trained proxy results for two data-driven models with conventional flow-based upscaled models to validate the accuracy of the proxy modeling methodology.

Fig. 3 illustrates the total water rate of all 100 realizations for high-fidelity, data-driven proxy, and upscaled proxy cases. It can be seen that the stochastic 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. **Fig. 4** shows the P10, P50, and P90 quantile response from Fig. 3. It shows that the data-driven model has a better agreement with the response of the high-fidelity model compared with the response of the upscaled model. The oil and water mean errors for the data-driven and upscaled models are shown in **Table 1.**

Then, the same test was performed for a more complicated model ensemble build with the MPS modeling approach. The water rates and the corresponding quantiles of all 100 realizations for the high-fidelity, data-driven proxy, and upscaled models are shown in **Figs. 5 and 6,** respectively. The oil and water mean errors for the data-driven and upscaled models are shown in **Table 2.** Based on those results, one can conclude that errors for both types of proxy models are higher for the MPS ensemble than the Flumy ensemble.

This result is expected because it is much more difficult to find a value for the effective property on a coarse scale that will accurately represent complex fine-scale features (e.g., small and poorly connected channels, which can be seen in Fig. 2b) typical for an MPS model. 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. The overall accuracy of the data-driven proxy model is still significantly higher than that for the upscaled proxy model. This is because the upscaled model uses the reduced-order approximate properties, while the data-driven model is directly trained by the response of the reservoir. For example, the scaling up of some reservoir properties (e.g., porosity and permeability) is performed using averaging methods. In this case, some accuracy might be lost using more homogenized properties to characterize the original heterogeneity of the model. Besides, we only apply well-established single-phase upscaling, which has a limited accuracy in a multiphase-flow situation. The examples of Flumy and MPS confirm the applicability of the data-driven approach for uncertainty-quantification analysis when a reliable and accurate high-fidelity model is not available.

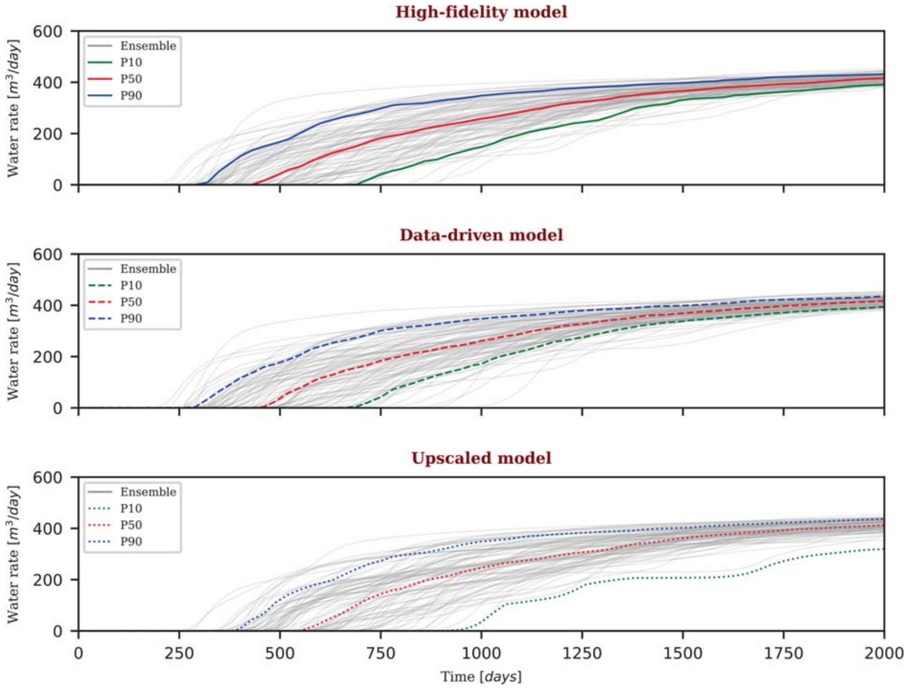


Fig. 3—Total water rate for the high-fidelity (reference) model with the size of the 100 × 100 gridblock for the hundred Flumy realizations, together with the responses of data-driven and upscaled models (10 × 10). Gray lines indicate the rates from a single-model realization, whereas the red, blue, and green lines indicate the quantile response of the ensemble (i.e., the P10, P50, and P90).

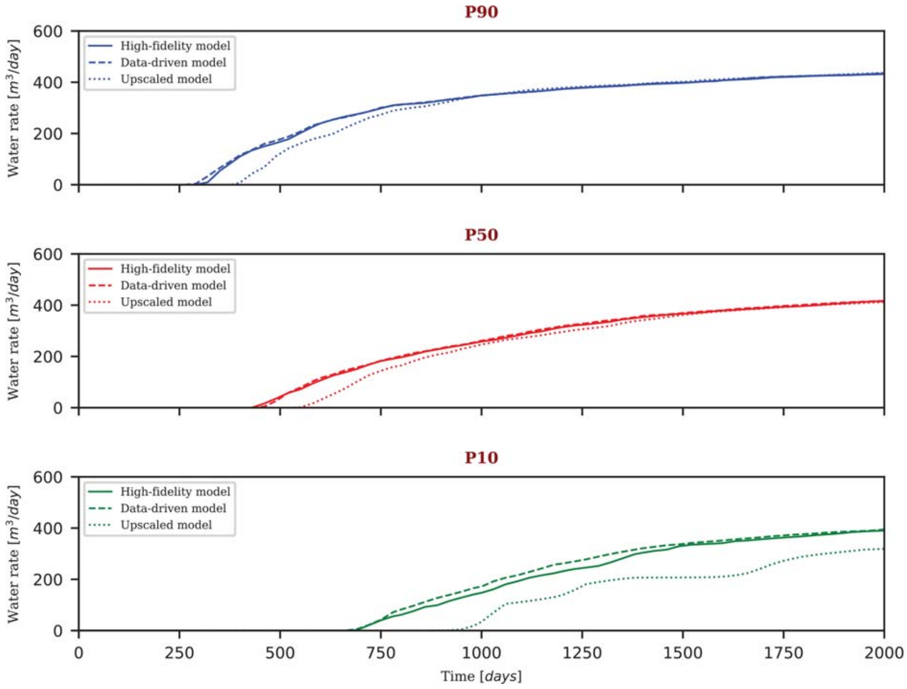


Fig. 4—P10, P50, and P90 quantile response of the high-fidelity (reference), data-driven, and upscaled models for 100 Flumy realizations.

Model	Water Rate (%)	Oil Rate (%)
Data-driven model	1.5	5.8
Upscaled model	13.0	10.6

Table 1—Mean errors of the water and oil rates of the Flumy model.

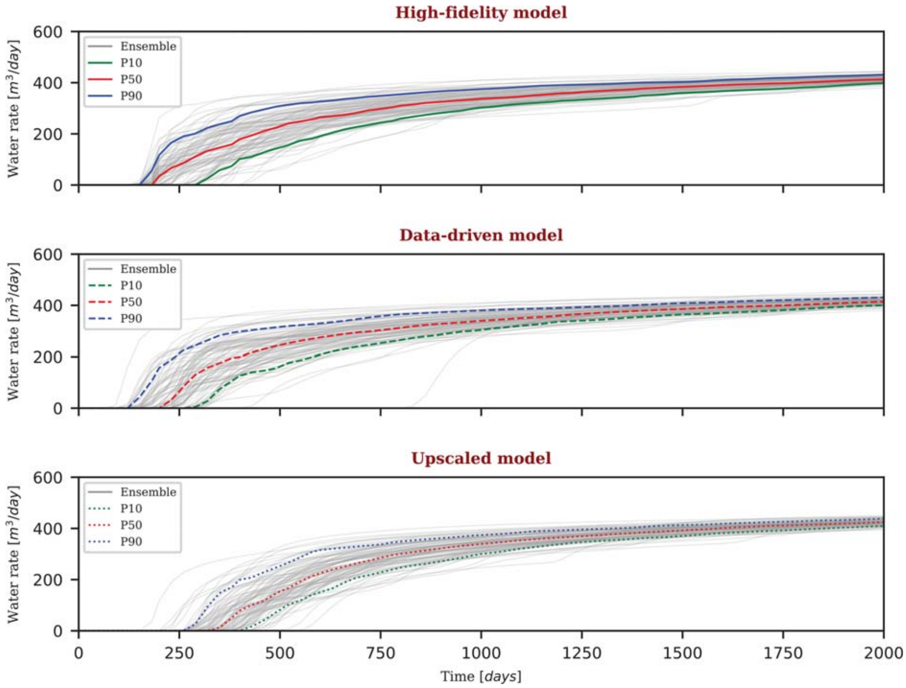


Fig. 5—Total water rate for the high-fidelity (reference) model with the size of the 100 × 100 gridblock for 100 MPS realizations, together with the responses of data-driven and upscaled models (10 × 10).

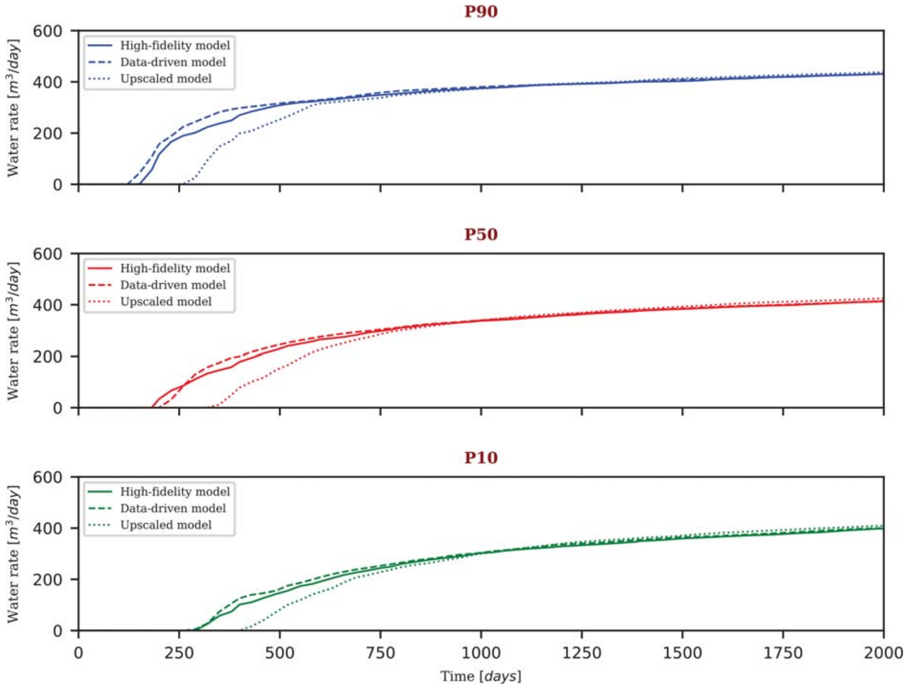


Fig. 6—P10, P50, and P90 quantile response of the high-fidelity (reference), data-driven, and upscaled models for 100 MPS realizations.

Model	Water Rate (%)	Oil Rate (%)
Data-driven model	4.5	5.9
Upscaled model	19.6	12.0

Table 2—Mean errors of the water and oil rates of the MPS ensemble.

Discrete Well Affinity Proxy Model for a Realistic Field

In this section, we provide the description of a realistic field used in this study. Next, we describe a generation of the discrete well affinity (DiWA) proxy model and illustrate the performance of the proposed methodology.

High-Fidelity and Proxy Models. We analyze the performance of the proposed framework using a reservoir model of the Brugge Field (Peters et al. 2010). It 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. The model has 30 wells (20 producers and 10 injectors; see **Fig. 7a**) located in the peripheral waterdrive. There are more than 100 realizations of this model. A single realization encoded as FY-SS-KP-8-73 is used as the true high-fidelity model in this study, and 40 other realizations with the KP code are taken for the high-fidelity ensemble. A covariance matrix can be obtained using these 40 realizations to regularize the objective function during history matching. The true data for history matching were obtained by a 10-year-long simulation with monthly varying BHP well controls.

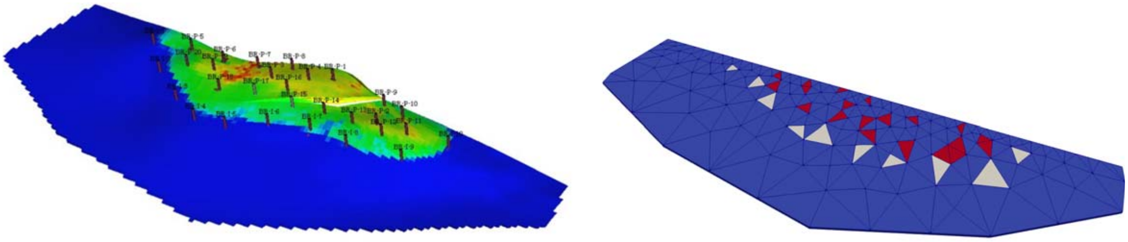


Fig. 7—(a) High-fidelity Brugge model used to generate the true model response (139 × 48×9 gridblocks). (b) Unstructured proxy model mesh with 283 elements: Red cells indicate a producing well, and white cells indicate an injection well.

Only the basic information about the field was used for the DiWA proxy model, and then it wass trained using the observed true data generated from high-fidelity model. For example, the reservoir boundaries were approximated using several piecewise linear segments, as can be seen in Fig. 7b. Next, the unstructured mesh was generated and extruded by an average value of reservoir thickness. The two-point stencil was used for the discretization of an unstructured grid to create the discrete connectivity graph for the proxy model.

The mesh was constructed using two regions by Gmsh software: an outer with coarser meshing and an inner with finer meshing, as seen in Fig. 7b. This is done to preserve accuracy in the area where the main flow occurs, whereas the outer cells were made larger to reduce the number of degrees of freedom along with the computational load. There are no significant flow dynamics in that area, hence the coarsening of that zone does not significantly affect simulation accuracy. The resulting performance of the proxy (coarse grid) model vs. the high-fidelity (original fine grid) model is an improvement of approximately two orders of magnitude (0.46 vs. 97 seconds for a single run of forward simulations). This performance is further enhanced using parallel implementation of numerical-gradient calculations discussed above, which brings the proxy model execution to approximately 20 runs per second.

Generation of Proxy Ensemble. Here, we introduce the response of the proxy ensemble to regularize the objective function for the training procedure. The initial guesses for control variables were calculated using a uniform permeability of $k = 1{,}000$ md and governing equation described in Eq. 1. The vector of rock-fluid property parameters $v_n = \{0.15, 0.25, 4, 3, 1800, 300\}$ and v_w was set to 200 cp·m^3/d/bar for all wells. The total volume of the proxy model was adjusted with model thickness, porosity, and initial water saturations of individual cells to match fluid-in-place volumes of the high-fidelity model. In a more general case, these parameters can be adjusted in the training period.

Obtained optimal reservoir property and rock-fluid property parameters are used as the basis for proxy model ensemble generation, where we have applied a random perturbation of parameters using a normal distribution with 20% variance. The results of the training stage are shown in **Fig. 8.** A good data fitting was obtained using the data-driven proxy model. However, some periods were not matched very well, which can be seen between 1,000 and 3,000 days for some wells. This is related to an insufficient number of degrees of freedom in the proxy model, which results in a limited variety of possible flow distribution.

Training and Forecast. Fig. 9 illustrates a 2-year production forecast of the Brugge model using the data-driven prior-ensemble training. For fair analysis of the model, we have generated a new set of well controls, different from the one used during training, with random perturbations within 20% of 100 bar for producers and 120 bar for injectors. The obtained results indicate a relatively accurate rate prediction for all 20 production wells. The mean error for the training period is calculated using Eq. 29 and is equal to 1.83%, whereas the prediction-period error reaches the value of 3.18%.

$$ME = \frac{\sum_{i=1}^{N} \left| \frac{Q^i_{o,\mathrm{opt}} - Q^i_{o,\mathrm{truth}}}{Q^i_{o,\mathrm{truth}}} \right| \times 100}{N}, \qquad (29)$$

where N is the number of timesteps; $Q_{o,\mathrm{opt}}$ is the oil rate of optimized proxy model; and $Q_{o,\mathrm{truth}}$ is the oil rate of true data generated from high-fidelity model.

For comparison, the data-driven model training can be performed using a regularization with a more accurate high-fidelity prior ensemble available for the Brugge model. It can be seen from **Fig. 10** that the high-fidelity prior ensemble has less variation than the proxy model-based prior ensemble. Consequently, this results in a smaller variation in the covariance matrix and is more constraining for the objective function. The mean error for the training period is calculated to be equal to 1.67% using Eq. 29, whereas the prediction-period error came to 4.27%. By comparison, there is no large improvement in terms of model accuracy when trained on the high-fidelity prior ensemble, whereas the time to generate its ensemble is more significant. Besides, multiple realizations of the high-fidelity model are usually not feasible for realistic situations when the detailed geological model is not available.

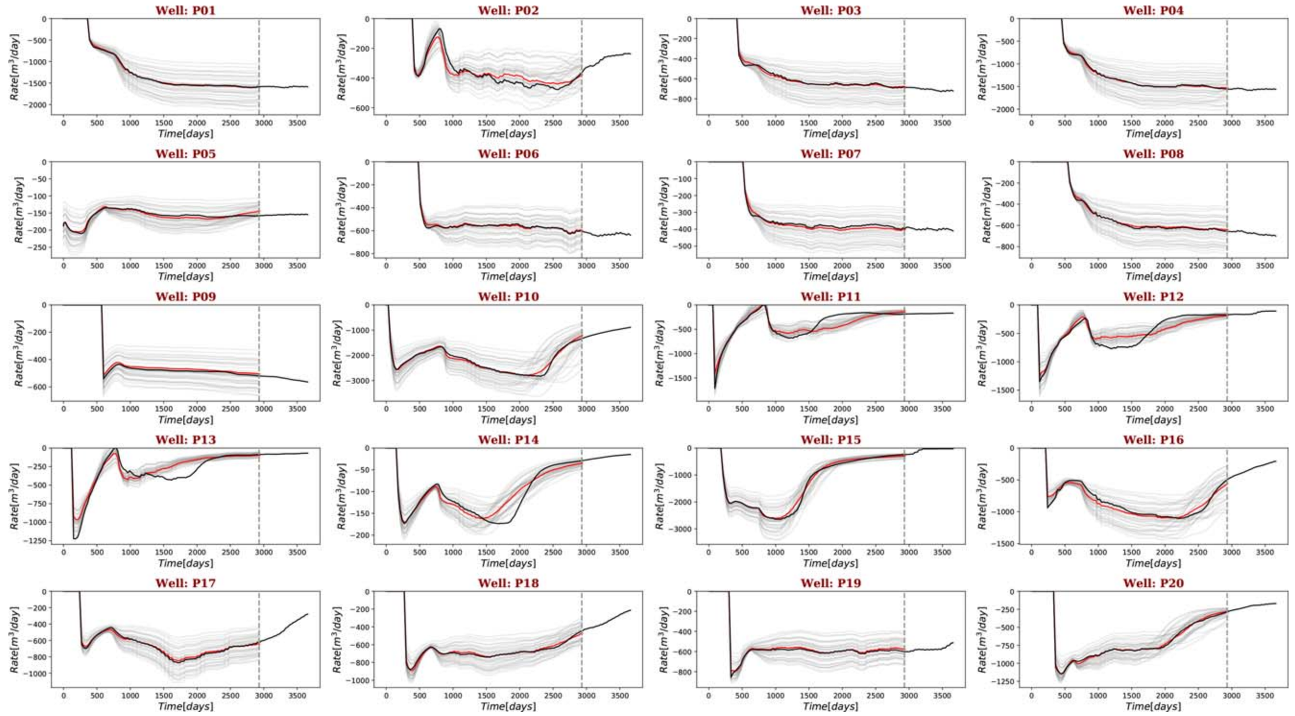


Fig. 8—Oil production rates obtained with the history-matched data-driven proxy model. Gray lines are oil production rates from the prior proxy ensemble, the black line is a true response, and the red line is the history-matched estimated data-driven proxy response.

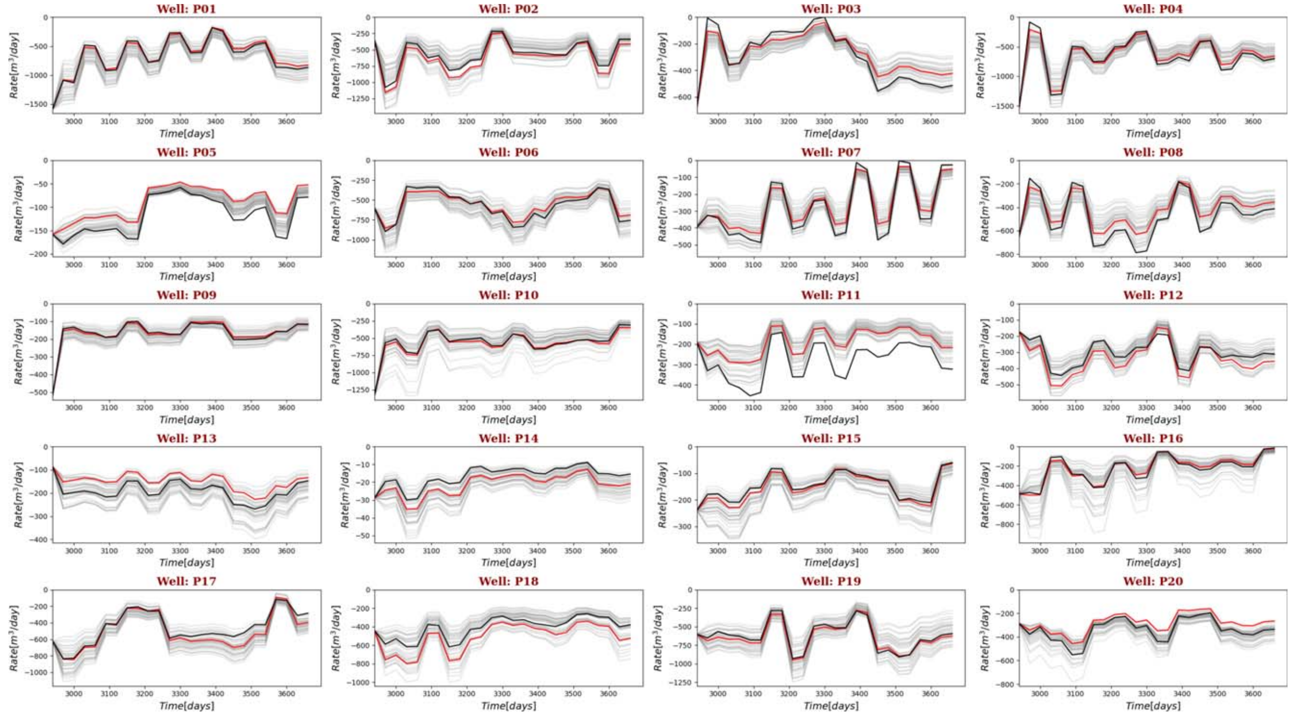


Fig. 9—Oil production rates obtained with the history-matched data-driven proxy model. Gray lines are oil production rates from multiple prior realizations of the high-fidelity model, the black line is a true response, and the red line is the history-matched estimated data-driven proxy response.

It was observed that the accuracy of the proxy model increases with the increase of data density for each well. In cases when the well rate is small compared with the other wells, the output from those wells is less informative for the regression algorithm. Hence, it fails to adjust parameters in the proxy model for those wells with the same quality than for others. Moreover, the number of regression parameters should be chosen wisely because the model is prone to overfitting. A superb training can be achieved in this case, but the model will struggle to give an accurate prediction.

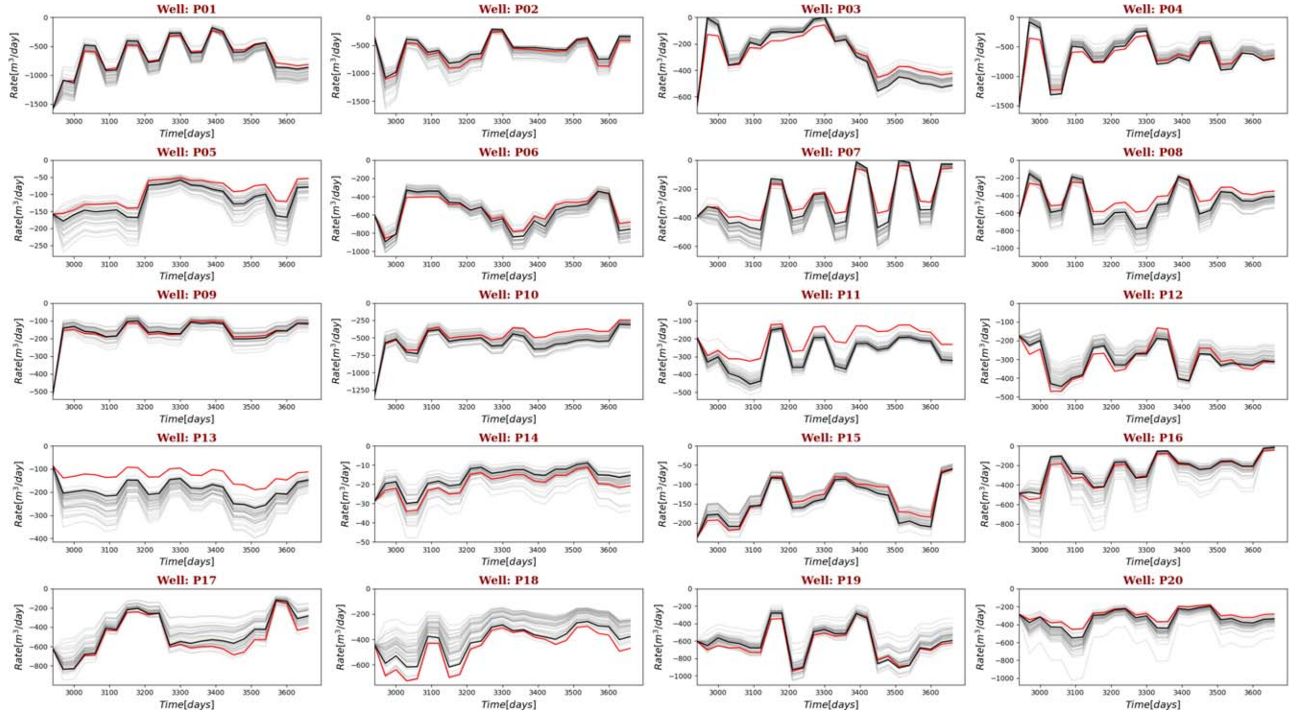


Fig. 10—Oil production rates obtained with the history-matched data-driven proxy model. Gray lines are oil production rates from multiple prior realizations of the proxy model, the black line is a true response, and the red line is the history-matched estimated data-driven proxy response.

Enrichment of DiWA Proxy Model

In this section, we extend the proposed data-driven framework to more complex situations. The reservoir models with multiple flow regions and the enrichment around nonmatching wells are investigated and applied in data-driven framework. Besides, more frequent changes of BHP control are introduced in the course of oil recovery to increase the complexity of the training.

We take the same Brugge model as earlier. The initial guess for reservoir property variables is calculated using a uniform permeability of $k = 1{,}000$ md. The initial guess vector of rock-fluid property parameters is $\{0.15, 0.255, 4, 3, 1800, 276\}$, and the well property variables are set as the same with high-fidelity model. The observed true data are generated using the high-fidelity model running for 3,720 days. For every 120 days, the BHP controls of the injection wells are set as 170 bar plus uniformly distributed random perturbations ranging from 1 to 30, and the BHP controls of the production wells are set at 130 bar minus uniformly distributed random perturbations ranging from 1 to 30. The first 3,000 days are chosen as the training period to train the proxy model. The rest of the 720 days are considered as the forecast period to test the forecast accuracy of the trained proxy model.

Data-Driven Model with Multiple Flow Regions. To increase the degrees of freedom of the DiWA proxy model and investigate how the rock-fluid property variable affects the performance of model training, we introduce more flow regions to the Brugge Field model. The rock-fluid property variables vary in different regions, while the grid cells inside each region share the same set of rock-fluid property variables. **Fig. 11** shows the examples in which Brugge Field is divided into two and four regions.

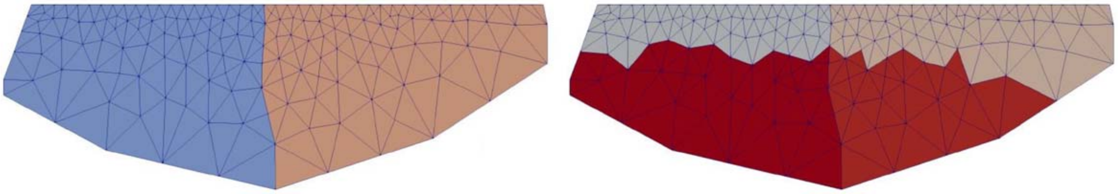


Fig. 11—Region division of Brugge Field model. The cells with the same color share the same set of rock-fluid property variables.

When training the proxy model, the gradients of the reservoir property variable (i.e., transmissibility) and the well property variable (i.e., well index) are calculated using the adjoint method, whereas the gradients of the rock-fluid property variable are calculated using numerical gradients. We were using a desktop station with Intel Core i7-8556U for all the experiments here. The oil rates of the trained proxy models and the observed true data are shown in **Fig. 12.** The mean error, the CPU time of model training, the number of the control variables, and the number of iterations of these three models are shown in **Table 3.**

As can be seen in Fig. 12, some of the wells do not have a good match with the observed true data if a single region is applied in the proxy model. In this case, we introduced more flow regions so that the number of rock-fluid property variables is enriched. However, the mean errors shown in Table 3 indicate that the enrichment of the number of the rock-fluid property variables does not guarantee the improvement of training performances, whereas the CPU time for training the model increases with the increase of the control variables.

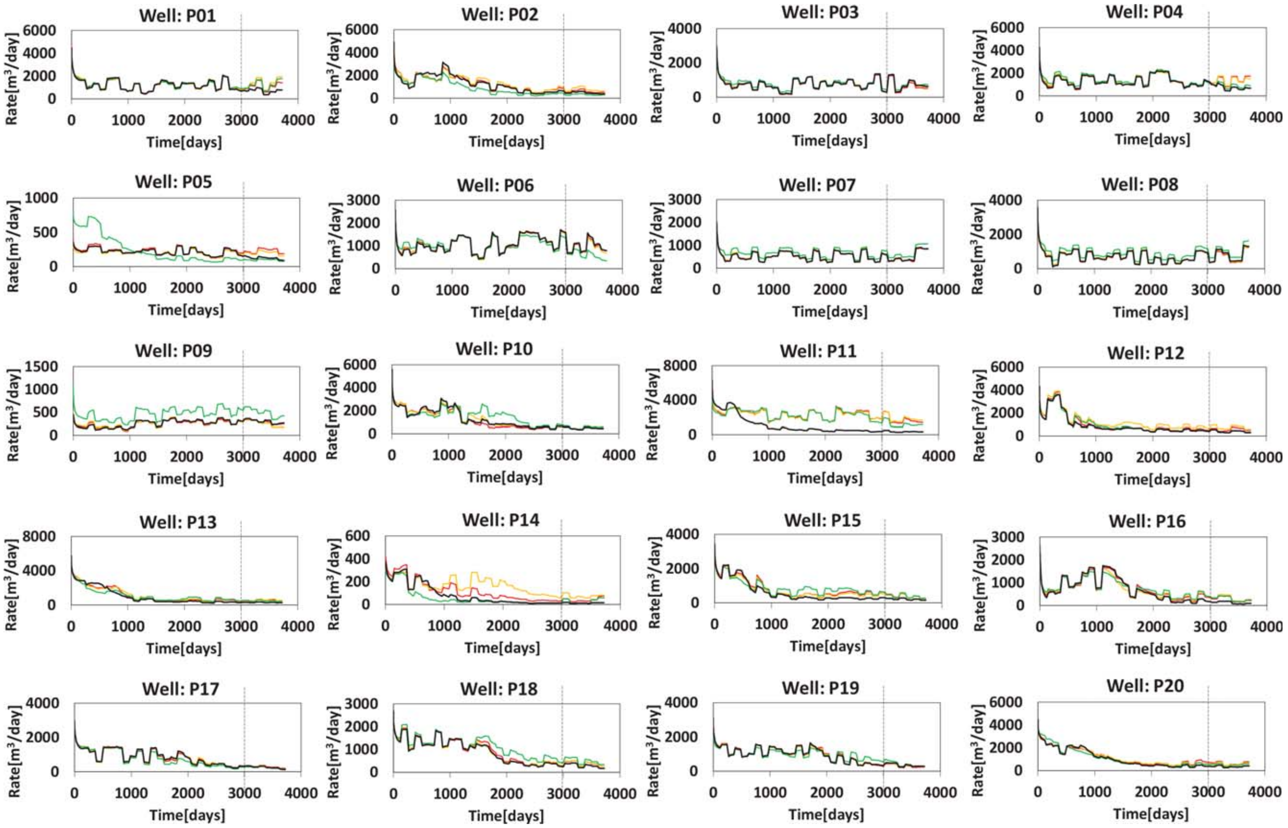
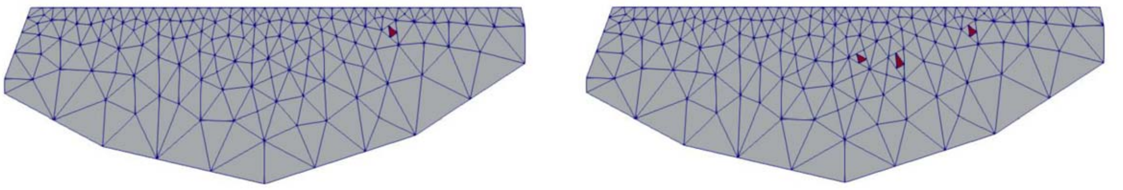


Fig. 12—Oil rates of the trained proxy models and the observed true data. The black curves represent the observed true data. The red, orange, and green curves show the results of the model with one, two, and four regions, respectively. The vertical lines separate the training and forecast periods.

Flow Region	Training (%)	Forecast (%)	CPU Time (minutes)	Number of Control Variables	Number of Iterations
One region	35.9	31.1	48.2	430	100
Two regions	42.9	36.6	54.1	436	100
Four regions	40.2	24.5	52.7	448	45

Table 3—Mean error of the training and forecast periods, the CPU time of model training, the number of control variables, and the number of iterations for the data-driven models with multiple regions.

Enrichment around Nonmatching Wells. It is noticed that some production wells (e.g., Wells P11, P14, P15, and so forth) do not have a good match with the observed true data in Fig. 12. In this subsection, we test an enrichment in the degrees of freedom around those nonmatching wells. In this procedure, we take the midpoint of the edges of the original block where the nonmatching well is located, and then use those midpoints to split the original block into four new triangles. The well is relocated in the central triangle. This procedure increases the degrees of freedom (transmissibilities) of the proxy model. Two cases are introduced in this subsection to investigate the performance of the method of enrichment around nonmatching wells. Models are shown in **Fig. 13.**



Enrichment around well P11

Enrichment around well P11, P14 and P15

Fig. 13—Enrichment around nonmatching wells. The left model contains the enrichment around Well P11. The right model contains the enrichment around Wells P11, P14, and P15. After the enrichment, the new well positions are located in the central red triangles.

The adjoint method is also implemented for the training of the enriched models in this subsection to improve the computational efficiency of gradient evaluation. The oil rates of the trained proxy models and the observed true data are shown in **Fig. 14.** The mean error, the CPU time of model training, the number of the control variables, and the number of iterations are shown in **Table 4.**

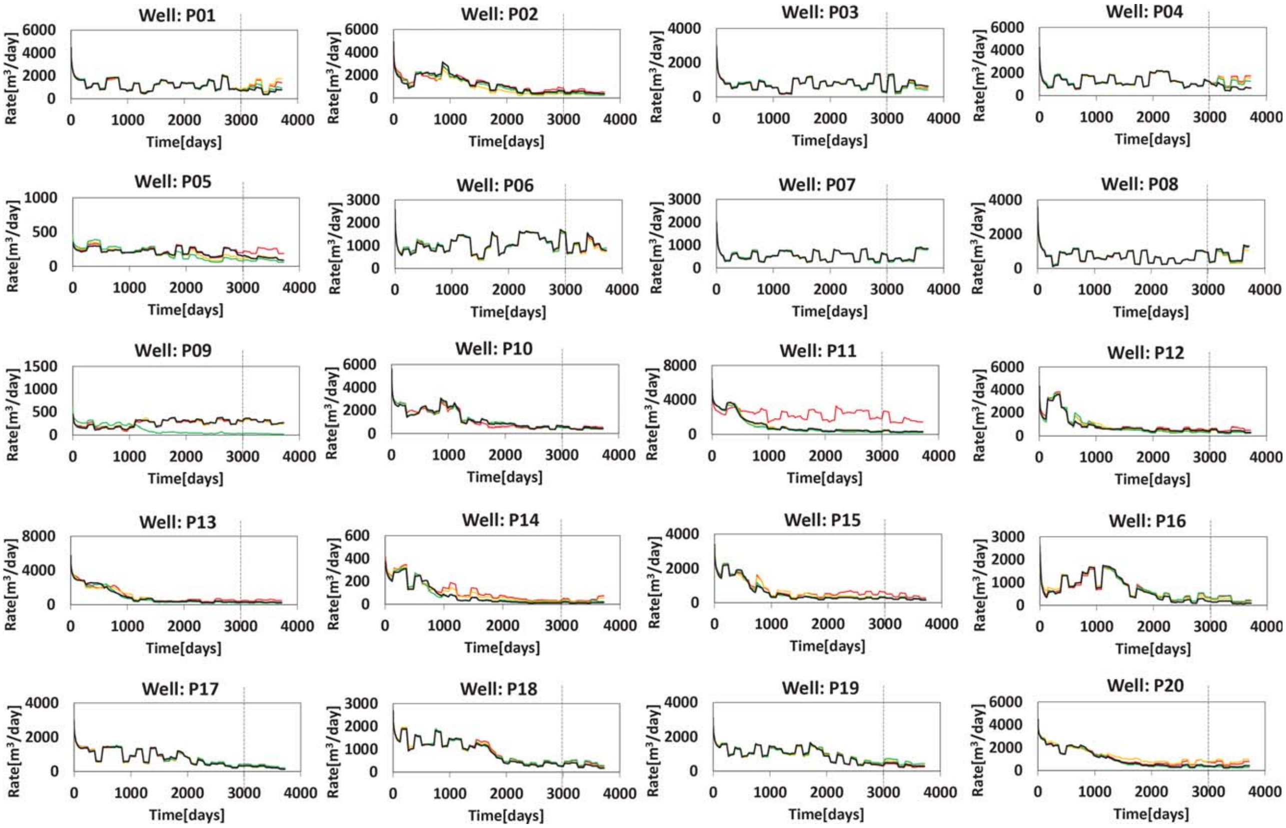


Fig. 14—Oil rates of the trained proxy models and the observed true data. The black curves represent the observed true data. The red, orange, and green curves show the results of the model without enrichment, the model with enrichment around Well P11, and the model with enrichment around Wells P11, P14, and P15, respectively. The vertical lines separate the training and forecast periods.

Enrichment	Training (%)	Forecast (%)	CPU Time (minutes)	Number of Control Variables	Number of Iterations
None	35.9	31.1	48.2	430	100
P11	9.3	6.6	38.7	436	100
P11, P14, P15	6.8	5.0	65.8	448	100

Table 4—Mean error of the training and forecast periods, the CPU time of model training, the number of control variables, and the number of iterations for data-driven models with enrichment around nonmatching wells.

As can be seen in Fig. 14 and Table 4, the oil-rate curves of the enriched models (orange and green curves) almost overlap with the true data (black curves). The mean errors of both the training and forecast period decreased. The results indicate that the enrichment around nonmatching wells can improve the accuracy of the DiWA model.

Conclusions

In this work, a physics-based data-driven framework was developed using the DARTS platform. The resulting DiWA strategy was evaluated on two synthetic data ensembles and showed good prediction accuracy for a significantly reduced model size. Both training and prediction accuracy are within a satisfactory level, with good modeling of all well production rates. In addition, the data-driven proxy methodology was compared with a conventional flow-based upscaling technique and demonstrated an improved accuracy within both stochastic ensembles.

The DiWA framework was examined on a more realistic Brugge Field data set. The proposed data-driven methodology demonstrates good predictive performance for training using synthetic data generated from the high-resolution Brugge Field. We compared the results of optimization with two types of covariance matrices using a high-fidelity prior ensemble and a data-driven proxy prior ensemble. Both approaches behave equally accurately, while the proxy-based prior ensemble is more feasible in practical applications. To investigate how the enrichment of the number of control variables will affect the performance of history matching, we tested the same proxy models with multiple flow regions and others with enrichment around nonmatching wells. The results show that both types of enrichment help to reduce the mean errors of both the training and forecast periods and therefore improve the performance of history matching. This framework is also capable of handling a more complex 3D model with multiple layers, which will be investigated in our future research.

The proposed data-driven DiWA method offers a great opportunity to obtain a fast and reliable framework for solving many subsurface engineering problems, as was demonstrated in this work. The rising popularity of those techniques indicates their 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. The main advantage of the proposed approach is that it is capable of providing accurate and reliable production forecast on a daily basis even though the exact geological information is not available. Besides, it also has the potential for a natural extension of the proxy model for more complex production scenarios and physical processes. For example, in the last section of this study, we have shown that this framework is capable of extending to the cases with multiple flow regions, more frequent changes of BHP control, enrichment around nonmatching wells. Apart from these, more components and phases can be introduced in this framework. Finally, with appropriate geological information available, the proposed proxy methodology can be used for better characterization of petrophysical data.

Nomenclature

A = interface area between neighboring gridblocks, m^2
c_r = rock compressibility, bar^{-1}
C_D = covariance matrix
d = depth (positive downward), m
D = distance from the pressure node to the interface along the line connecting two pressure nodes, m
$\mathbf{g}$ = residual of governing equation
J = objective function
$\overline{\mathcal{J}}$ = augmented objective function
k_{rj} = relative permeability of phase j
k_{ro} = oil relative permeability
k_{ro}^e = endpoint oil relative permeability
k_{rw} = water relative permeability
k_{rw}^e = endpoint water relative permeability
k_x = permeability in x-direction, md
k_y = permeability in y-direction, md
K = permeability tensor, md
L = interface connecting two neighboring gridblocks
n_c = number of component
n_o = exponent for oil
n_p = number of phase
n_w = exponent for water
N_{well} = number of wells
p = phase pressure, bar
P^c = coarse pressures obtained by arithmetic averaging the fine-scale pressures, bar
P^w = well bottomhole pressure, bar
$q*$ = phase rate per unit volume, 1/d
q^c = coarse flux across the interface of two neighboring gridblocks, m^3/d
q^w = well flow rate into (out of) the block, m^3/d
q = phase rate, m^3/d
q^{obs} = observed phase rates, m^3/d
r_o = equivalent well-block radius, m
r_w = well radius, m
S = phase saturation
S_{or} = residual oil saturation
S_w = water saturation
S_w^* = normalized or effective water saturation
S_{wc} = residual or connate water saturation
t = time, day
T_{ij} = transmissibility, $\mathrm{cp} \cdot \mathrm{m}^3$/d/bar.
T_i^w = well index, $\mathrm{cp} \cdot \mathrm{m}^3$/d/bar
u = control variables
v = phase velocity, m/d
V = volume of a control volume, m^3
x_c = mole fraction of component c in phase j
z_c = overall composition
α = operator about accumulation term
β = operator about flux term
γ = gravity term
δ = Dirac's delta function
$\Delta\psi$ = pressure potential, bar
Δx = block size in x-direction, m
Δy = block size in y-direction, m
Δz = block size in z-direction, m
ε = perturbation to control variables
ξ = spatial variation
θ = well controls
λ = Lagrange multiplier
μ = phase viscosity, cp
ρ = phase molar density
ϕ = porosity
ϕ_0 = initial porosity
ω = physical state

Acknowledgments

We acknowledge the financial support from the Wintershall Dea GmbH. We would like to thank Vitaly Elichev for his valuable suggestions during the course of the project and Stephan de Hoop for the help with upscaling ensembles.

References

Albertoni, A. and Lake, L. 2003. Inferring Interwell Connectivity Only from Well-Rate Fluctuations in Waterfloods. *SPE Res Eval & Eng* **6** (1): 6–16. SPE-83381-PA. https://doi.org/10.2118/83381-PA.

Batycky, R., Blunt, M., and Thiele, M. 1997. A 3D Field-Scale Streamline-Based Reservoir Simulator. *SPE Res Eng* **12** (4): 246–254. SPE-36726-PA. https://doi.org/10.2118/36726-PA.

Borregales, M., Møyner, O., Krogstad, S. et al. 2020. Data-Driven Models Based on Flow Diagnostics. Paper presented at ECMOR XVII, Virtual, 14–17 September. https://doi.org/10.3997/2214-4609.202035122.

Brooks, R. and Corey, A. 1964. *Hydraulic Properties of Porous Media*, Vol. 3. Fort Collins, Colorado, USA: Hydrology Papers Series, Colorado State University.

Cardoso, M., Durlofsky, L., and Sarma, P. 2009. Development and Application of Reduced-Order Modeling Procedures for Subsurface Flow Simulation. *Int J Numer Methods Eng* **77** (9): 1322–1350. https://doi.org/10.1002/nme.2453.

Chen, Y. and Durlofsky, L. J. 2006. Adaptive Local–Global Upscaling for General Flow Scenarios in Heterogeneous Formations. *Transp Porous Media* **62** (2): 157–185. https://doi.org/10.1007/s11242-005-0619-7.

Coats, K., George, W., Marcum, B. et al. 1974. Three-Dimensional Simulation of Steamflooding. *SPE J*. **14** (6): 573–592. SPE-4500-PA. https://doi.org/10.2118/4500-PA.

Collins, D., Nghiem, L., Li, Y.-K. et al. 1992. Efficient Approach to Adaptive-Implicit Compositional Simulation with an Equation of State. *SPE Res Eng* **7** (2): 259–264. SPE-15133-PA. https://doi.org/10.2118/15133-PA.

Datta-Gupta, A. and King, M. J. 2007. *Streamline Simulation: Theory and Practice*, Vol. 11. Richardson, Texas, USA: Textbook Series, Society of Petroleum Engineers.

de Hoop, S., Voskov, D., Vossepoel, F. et al. 2018. Quantification of Coarsening Effect on Response Uncertainty in Reservoir Simulation. Paper presented at ECMOR XVI, 16th European Conference on the Mathematics of Oil Recovery, Barcelona, Spain, 3–6 September. https://doi.org/10.3997/2214-4609.201802223.

Durlofsky, L. 2005. Upscaling and Gridding of Fine Scale Geological Models for Flow Simulation. Paper presented at the 8th International Forum on Reservoir Simulation, Stresa, Italy, 20–24 June.

Geuzaine, C. and Remacle, J.-F. 2009. Gmsh: A 3-D Finite Element Mesh Generator with Built-in Pre- and Post-Processing Facilities. *Int J Numer Methods Eng* **79** (11): 1309–1331. https://doi.org/10.1002/nme.2579.

Grappe, B., Cojan, I., Ors, F. et al. 2016. Dynamic Modelling of Meandering Fluvial Systems at the Reservoir Scale, Flumy Software. Paper presented at the Second Conference on Forward Modelling of Sedimentary Systems, Trondheim, Norway, 25–28 April. https://doi.org/10.3997/2214-4609.201600370.

Guo, Z. and Reynolds, A. C. 2019. INSIM-FT in Three-Dimensions with Gravity. *J Comput Phys* **380** (1 March): 143–169. https://doi.org/10.1016/j.jcp.2018.12.016.

Holanda, R., Gildin, E., Jensen, J. et al. 2018. A State-of-the-Art Literature Review on Capacitance Resistance Models for Reservoir Characterization and Performance Forecasting. *Energies* **11** (12): 3368. https://doi.org/10.3390/en11123368.

Holden, L. and Nielsen, B. F. 2000. Global Upscaling of Permeability in Heterogeneous Reservoirs; the Output Least Squares (OLS) Method. *Transp Porous Media* **40** (2): 115–143. https://doi.org/10.1023/A:1006657515753.

Hou, T. Y. and Wu, X.-H. 1997. A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media. *J Comput Phys* **134** (1): 169–189. https://doi.org/10.1006/jcph.1997.5682.

Jansen, F. and Kelkar, M. 1997. Non-Stationary Estimation of Reservoir Properties Using Production Data. Paper presented at the SPE Annual Technical Conference and Exhibition, San Antonio, Texas, USA, 5–8 October. SPE-38729-MS. https://doi.org/10.2118/38729-MS.

Jenny, P., Lee, S., and Tchelepi, H. 2003. Multi-Scale Finite-Volume Method for Elliptic Problems in Subsurface Flow Simulation. *J Comput Phys* **187** (1): 47–67. https://doi.org/10.1016/S0021-9991(03)00075-5.

Kala, K. and Voskov, D. 2020. Element Balance Formulation in Reactive Compositional Flow and Transport with Parameterization Technique. *Computat Geosci* **24** (2): 609–624. https://doi.org/10.1007/s10596-019-9828-y.

Karimi-Fard, M., Durlofsky, L., and Aziz, K. 2004. An Efficient Discrete-Fracture Model Applicable for General-Purpose Reservoir Simulators. *SPE J*. **9** (2): 227–236. SPE-88812-PA. https://doi.org/10.2118/88812-PA.

Khait, M. and Voskov, D. V. 2017. Operator-Based Linearization for General Purpose Reservoir Simulation. *J Pet Sci Eng* **157** (August): 990–998. https://doi.org/10.1016/j.petrol.2017.08.009.

Khait, M. and Voskov, D. 2018a. Adaptive Parameterization for Solving of Thermal/Compositional Nonlinear Flow and Transport with Buoyancy. *SPE J*. **23** (2): 522 –534. SPE-182685-PA. https://doi.org/10.2118/182685-PA.

Khait, M. and Voskov, D. 2018b. Operator-Based Linearization for Efficient Modeling of Geothermal Processes. *Geothermics* **74** (July): 7–18. https://doi.org/10.1016/j.geothermics.2018.01.012.

Kiærr, A., Lødøen, O., Bruin, W. D. et al. 2020. Evaluation of a Data-Driven Flow Network Model (Flownet) for Reservoir Prediction and Optimization. Paper presented at ECMOR XVII, Virtual, 14–17 September. https://doi.org/10.3997/2214-4609.202035099.

Kraft, D. 1988. *A Software Package for Sequential Quadratic Programming*. Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme Oberpfaffenhofen, Oberpfaffenhofen, Germany, July 1988.

Lerlertpakdee, P., Jafarpour, B., and Gildin, E. 2014. Efficient Production Optimization with Flow-Network Models. *SPE J*. **19** (6): 1083–1095. SPE-170241-PA. https://doi.org/10.2118/170241-PA.

Lim, K.-T. 1995. A New Approach for Residual and Jacobian Arrays Construction in Reservoir Simulators. *SPE Comp App* **7** (4): 93–96. SPE-28248-PA. https://doi.org/10.2118/28248-PA.

Mohaghegh, S. 2009. Artificial Intelligence and Data Mining: Enabling Technology for Smart Fields. *The Way Ahead* **5** (3): 14–19. SPE-0309-014-TWA. https://doi.org/10.2118/0309-014-TWA.

Peaceman, D. W. 1983. Interpretation of Well-Block Pressures in Numerical Reservoir Simulation with Nonsquare Grid Blocks and Anisotropic Permeability. *SPE J*. **23** (3): 531–543. SPE-10528-PA. https://doi.org/10.2118/10528-PA.

Peters, E., Arts, R., Brouwer, G. et al. 2010. Results of the Brugge Benchmark Study for Flooding Optimization and History Matching. *SPE Res Eval & Eng* **13** (3): 391–405. SPE-119094-PA. https://doi.org/10.2118/119094-PA.

Ren, G., He, J., Wang, Z. et al. 2019. Implementation of Physics-Based Data-Driven Models with a Commercial Simulator. Paper presented at the SPE Reservoir Simulation Conference, Galveston, Texas, USA, 10–11 April. SPE-193855-MS. https://doi.org/10.2118/193855-MS.

Strebelle, S. and Levy, M. 2008. Using Multiple-Point Statistics To Build Geologically Realistic Reservoir Models: The MPS/FDM Workflow. *Geol Soc Spec Publ* **309** (1): 67–74. https://doi.org/10.1144/SP309.5.

Voskov, D. 2017. Operator-Based Linearization Approach for Modeling of Multiphase Multi-Component Flow in Porous Media. *J Comput Phys* **337** (15 May): 275–288. https://doi.org/10.1016/j.jcp.2017.02.041.

Wang, Y., Voskov, D., Khait, M. et al. 2020. An Efficient Numerical Simulator for Geothermal Simulation: A Benchmark Study. *Appl Energy* **264** (15 April): 114693. https://doi.org/10.1016/j.apenergy.2020.114693.

Yousef, A., Gentil, P., Jensery, J. et al. 2005. A Capacitance Model To Infer Interwell Connectivity from Production and Injection Rate Fluctuations. Paper presented at the SPE Annual Technical Conference and Exhibition, Dallas, Texas, USA, 9–12 October. SPE-95322-MS. https://doi.org/10.2118/95322-MS.

Zhao, H., Kang, Z., Zhang, X. et al. 2015. Insim: A Data-Driven Model for History Matching and Prediction for Waterflooding Monitoring and Management with a Field Application. Paper presented at the SPE Reservoir Simulation Symposium, Houston, Texas, USA, 23–25 February. SPE-173213-MS. https://doi.org/10.2118/173213-MS.

Zhao, H., Xu, L., Guo, Z. et al. 2020. Flow-Path Tracking Strategy in a Data-Driven Interwell Numerical Simulation Model for Waterflooding History Matching and Performance Prediction with Infill Wells. *SPE J*. **25** (2): 1007–1025. SPE-199361-PA. https://doi.org/10.2118/199361-PA.

Zubarev, D. 2009. Pros and Cons of Applying Proxy-Models as a Substitute for Full Reservoir Simulations. Paper presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 4–7 October. SPE-124815-MS. https://doi.org/10.2118/124815-MS.

Appendix A—Comparison of Numerical and Adjoint Gradients

To validate the adjoint method and investigate the improvement of the gradient-calculation efficiency, the adjoint gradients are compared with the numerical gradients using five models with different grid resolutions. The models are shown in **Fig. A-1.** The angles θ between adjoint gradient and numerical gradient are calculated using

$$\theta = \arccos \frac{\mathrm{grad}_a \cdot \mathrm{grad}_{\mathrm{num}}}{\|\mathrm{grad}_a\| \cdot \|\mathrm{grad}_{\mathrm{num}}\|}, \qquad \text{(A-1)}$$

where grad_a and $\mathrm{grad}_{\mathrm{num}}$ are the vectors of the adjoint gradient and numerical gradient, respectively.

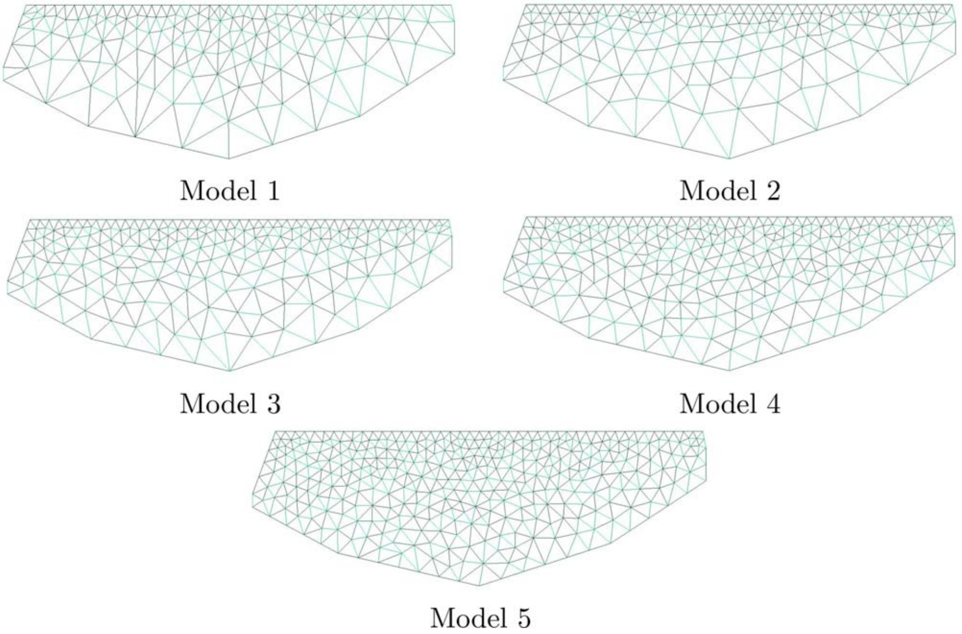


Fig. A-1—Five models with different grid resolutions: Model 1, Model 2, Model 3, Model 4, and Model 5.

For each model in Fig. A-1, 100 different realizations are generated by uniformly sampling the permeability of cells from 1 to 100,000 md. Later, the adjoint gradients and numerical gradients are computed and compared for all 100 realizations of each model. The mean values of the angle and the CPU time for computing the adjoint gradients and numerical gradients for five models are shown in **Table A-1.**

Model	Angle θ (degrees)	Number of Transmissibility and Well Index	$t_{\mathrm{grad}_{\mathrm{num}}}$ (seconds)	t_{grad_a} (seconds)	$t_{\mathrm{grad}_{\mathrm{num}}}/t_{\mathrm{grad}_a}$
1	3.49	424	1,711.28	3.75	456
2	3.33	478	2,214.77	4.23	524
3	3.60	599	3,483.75	5.43	642
4	3.24	741	5,376.97	6.79	791
5	4.34	983	10,312.31	9.69	1,064

Table A-1—Angles between adjoint gradients and numerical gradients and the CPU time of computing gradients.

The third column of Table A-1 shows the total number of control variables of each model, which are transmissibility and well index. $t_{\text{grad}_{\text{num}}}$ and t_{grad_a} are the mean CPU times of computing the gradients by using the adjoint method and the numerical method, respectively. As can be seen in Table A-1, the angle θ for the five models is approximately 3.6°, which is small and therefore validates the values of the adjoint gradient. This also indicates that the adjoint gradient can be used to test if the choice of ε for calculating the numerical gradient as mentioned in Eq. 11 is appropriate. This is because the adjoint method avoids introducing an extrasmall perturbation ε to calculate the gradient. Plotting the number of control variables vs. CPU time from Table A-1 is shown in **Fig. A-2.** Fig. A-2 indicates that if more control variables are introduced, a higher efficiency of gradient calculation using the adjoint method can be obtained compared with the numerical method.

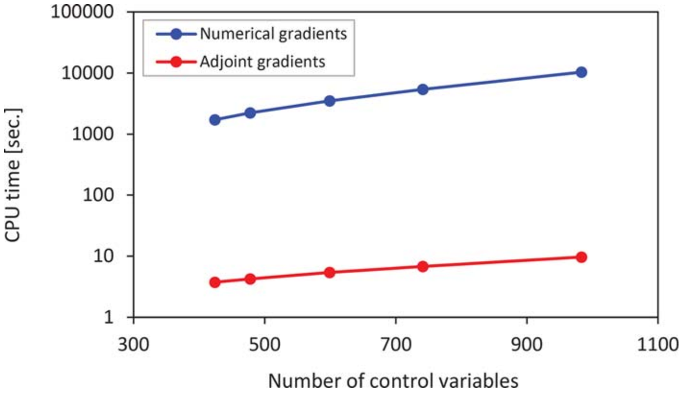


Fig. A-2—CPU time for computing the adjoint method and the numerical method.

Appendix B—Physical Modeling Properties

Table B-1 introduces the main physical properties used for comparison among high-fidelity, upscaled, and data-driven models.

Phase	Oil	Water
Fluid compressibility, c_j (1/bar)	1.34×10^{-4}	4.35×10^{-5}
Fluid densities, ρ_j (kg/m^3)	897.0	1002.8
Residual saturation, S_{jr}	0.15	0.225
Endpoint relative permeability, K_{rje}	0.4	1.0
Saturation exponent, n_j	3.0	3.0
Viscosity, μ_j (cp)	1.294	0.320

Table B-1—Physical properties.