Computational Geosciences
https://doi.org/10.1007/s10596-023-10237-y

ORIGINAL PAPER

Data-driven modelling with coarse-grid network models

Knut-Andreas Lie[1] · **Stein Krogstad**[1]

Received: 14 November 2022 / Accepted: 10 July 2023

Abstract
We propose to use a conventional simulator, formulated on the topology of a coarse volumetric 3D grid, as a data-driven network model that seeks to reproduce observed and predict future well responses. The conceptual difference from standard history matching is that the tunable network parameters are calibrated freely without regard to the physical interpretation of their calibrated values. The simplest version uses a minimal rectilinear mesh covering the assumed map outline and base/top surface of the reservoir. The resulting CGNet models fit immediately in any standard simulator and are very fast to evaluate because of the low cell count. We show that surprisingly accurate network models can be developed using grids with a few tens or hundreds of cells. Compared with similar interwell network models (e.g., Ren et al., 2019, 10.2118/193855-MS), a typical CGNet model has fewer computational cells but a richer connection graph and more tunable parameters. In our experience, CGNet models therefore calibrate better and are simpler to set up to reflect known fluid contacts, etc. For cases with poor vertical connection or internal fluid contacts, it is advantageous if the model has several horizontal layers in the network topology. We also show that starting with a good ballpark estimate of the reservoir volume is a precursor to a good calibration.

Keywords Data-driven models · Model calibration · Interwell network models

1 Introduction

Ideas from data-driven methods has become increasingly popular for modelling flow in subsurface reservoirs in recent years, and in particular as a means to reduce the computational cost of evaluating the forward model in model-driven production optimization and optimized field development [1–8]. Herein, we consider the somewhat different task of rapidly building a flow model based on well responses but limited geological and petrophysical data.

Data-driven models are often formulated using techniques from machine learning. Such models can give accurate predictions for physical states close to those embedded in the training data, but may also easily predict nonphysical states if no physical laws are incorporated to constrain the model training. Many authors have therefore considered physics-informed models that either try to learn the underlying flow equations directly (in residual form) or penalize predictions that violate fundamental flow physics [1, 2, 6].

Another approach is to start from a set of model equations that represent the flow physics in some discretized form and calibrate the parameters of these equations so that resulting predictions match observed well responses from the reservoir. Herein, we propose to simply use a straightforward finite-volume discretization of the standard flow equations, posed on top of a very coarse, minimal rectilinear grid that covers the assumed map outline and the base and top surfaces of the reservoir, as our data-driven proxy model. The free parameters in this model are the pore volumes, transmissibilities, and well-connection factors of the grid, and possibly also the initial fluid compositions and the parameters of standard Corey-type relative permeability models. These will be calibrated with a Gauss–Newton method to match observed data from the injection and production stream. We will refer to this method as CGNet [9].

The idea of calibrating a coarse-grid model to match observed production responses has been used for decades in history matching; confer, for instance, with [10, 11]. The approach discussed herein is nonetheless different. Modern

✉ Knut-Andreas Lie
knut-andreas.lie@sintef.no

Stein Krogstad
stein.krogstad@sintef.no

[1] Mathematics & Cybernetics, SINTEF Digital, P.O. Box 124 Blindern, N-0314 Oslo, Norway

Published online: 04 August 2023

history matching is based on Bayesian statistics and seeks to sample the uncertainty space, constrained to prior geological information and observed dynamic data. That is, the goal is not only to match observed dynamical data, but also maintain geological correctness and obtain physically interpretable parameters that are representative of the physical system. In our data-driven setting, on the other hand, the parameters of the discrete model equations should only be viewed as *tunable* algebraic coefficients that are calibrated (within reasonable physical limits) to derive a desired, physics-based, input–output relationship between enforced well controls and observed reservoir responses. Put simply, our intention is to achieve a reduced discrepancy with dynamic training data and enhanced prediction accuracy for scenarios closely related to or encompassed by the available data by intentionally omitting the utilization of prior geological information to constrain the model calibration. However, this approach may come at the cost of reduced applicability for scenarios that deviate significantly from the available data. Hence, it is crucial to include pertinent flow regimes, particularly fluid breakthrough at wells, in the dynamic data used for model training to ensure good predictive ability.

CGNet, in essence, shares a similar conceptual framework and motivation with interwell network models, as it represents the reservoir as a network of interconnected flow paths linking injectors and producers. Interwell network models come in various forms. Capacitance-resistance models (CRM) [12–14] are simplified mathematical representations that combine capacitance (fluid storage) and resistance (flow pathways) to predict pressure and fluid movement while preserving the overall material balance. Their validity is somewhat limited by the assumption of constant productivity indices and allocation factors. Another example is interwell networks of stock-tank models [15, 16], used as part of a top-down reservoir modelling framework [17].

The family of interwell numerical simulation models (INSIM) focus on the dynamic behavior and interactions between multiple wells and use analytical or semi-analytical methods to evolve pressures and fluid compositions within each interwell connection [18–22]. A more general approach, referred to by different authors as StellNet/GPSNet/FlowNet, is to use standard finite-volume methods to represent the interwell connections and also allow direct fluid communication among different flow paths without going through the wells [23–29]. Compared with these interwell models, CGNet gives a richer network that has more free parameters per grid cell and a larger set of network paths connecting each pair of wells. In a recent paper [9], we showed that this makes it somewhat simpler to build reduced-order models trained to match simulation data from detailed flow models. Herein, we consider a purely data-driven scenario in which one has not yet developed a detailed 3D simulation model.

For completeness, we mention that Mamonov et al. [30] introduced an alternative method that shares the same spirit as CGNet, focusing on optimizing the coarse mesh to preserve aggregated objective values specifically tailored for production optimization purposes. Likewise, RGNet models [31, 32] provide another example of models with richer connections among reservoir nodes. These models utilize diffusive time-of-flight, representing propagation of pressure fronts, to determine drainage volumes and then employ inter-partition transmissibilities [33] to transform interconnections among these volumes into a comprehensive reservoir network.

2 Model equations and discretization

CGNets can in principle be formed for any standard set of flow equations. To simplify the presentation, we only describe the setup for two-phase, compressible, immiscible flow without capillary pressure, as the formulation for more complex models like black-oil, equation-of-state compositional models, and thermal flow follows the same setup with the obvious modifications. We assume that the reservoir volume has been discretized into a finite set of non-overlapping volumetric grid cells. To specify the associated discrete flow model, we introduce div and grad as the discrete numerical analogues of the standard divergence and gradient operators. These discrete operators are linear maps, represented in terms of sparse matrices, that operate on vectors of quantities associated with each cell or cell interface; see Lie [34] for more details. If we also use backward Euler for the temporal discretization, the resulting fully implicit flow discretization of the flow equations consists of three different parts[1]:

- A discrete continuity equation describing conservation of mass for each phase (or component for multicomponent flows)

$$\frac{1}{\Delta t}\left[\left(\boldsymbol{\Phi}\boldsymbol{S}_\alpha\boldsymbol{\rho}_\alpha\right)^{n+1}-\left(\boldsymbol{\Phi}\boldsymbol{S}_\alpha\boldsymbol{\rho}_\alpha\right)^{n}\right]+\mathsf{div}\left(\boldsymbol{\rho}_\alpha\boldsymbol{v}_\alpha\right)^{n+1}=\boldsymbol{q}_\alpha^{n+1},\quad \alpha\in\{w,o\}. \tag{1}$$

  Here, $\boldsymbol{\Phi}$ is the vector of pore volumes per cell, $\boldsymbol{\rho}_\alpha$ is the vector of cell-averaged phase densities, $\boldsymbol{S}_\alpha$ is the vector of cell-averaged fluid saturations, $\boldsymbol{v}_\alpha$ is the vector of inter-cell phase fluxes, and $\boldsymbol{q}_\alpha$ is the vector of cell-averaged volumetric source terms for phase α. The superscript refers to the time steps of the temporal discretization,

[1] By a slight abuse of notation, we here interpret the product of two vector quantities as the vector of element-wise products.

which we henceforth for brevity will drop for all terms evaluated at time step $n+1$.

- The phase fluxes are given by a discrete version of Darcy's law

$$\boldsymbol{v}_\alpha = -\mathsf{upw}(\boldsymbol{\lambda}_\alpha)\mathbf{T}\Big(\mathsf{grad}(\boldsymbol{p}) - g\,\mathsf{avg}(\boldsymbol{\rho}_\alpha)\mathsf{grad}(\boldsymbol{z})\Big). \quad (2)$$

  Here, $\boldsymbol{p}$ denotes the cell-averaged fluid pressure, g is the gravity constant, and $\boldsymbol{z}$ holds the z-coordinates of the cell centroids. The $\mathsf{avg}(\cdot)$ operator computes the arithmetic average of cell-averaged densities defined on opposite sides of cell faces, whereas $\mathsf{upw}(\cdot)$ picks mobility values λ_α from the upstream side of the face. (The phase mobility is defined as $k_{r\alpha}/\mu_\alpha$, where the relative permeability $k_{r\alpha}(S_\alpha)$ is a known function of S_α and μ_α is the phase viscosity.) Finally, the vector $\mathbf{T}$ holds the inter-cell transmissibilities, which involve geometric terms and harmonic averages of the permeabilities of cells on opposite sides of each interface.
- The volumetric source (or sink) terms from wells perforated inside individual cells are given by an inflow (or outflow) performance relationship,

$$\boldsymbol{q}_\alpha = \boldsymbol{\lambda}_\alpha^{wb}\boldsymbol{J}(\boldsymbol{p}^{wb} - \boldsymbol{p}) \quad (3)$$

  relating flow rates $\boldsymbol{q}_\alpha$ to phase mobilities $\boldsymbol{\lambda}_\alpha^{wb}$ at the wells, well indices $\boldsymbol{J}$, and the difference between the wellbore pressures $\boldsymbol{p}^{wb}$ and the cell-averaged pressure $\boldsymbol{p}$ in the completed cells.

Assuming that the two fluids fill pore space completely, the continuity equations (1) and Darcy's law (2) can be reduced to a system of two equations per cell for fluid pressure p and water saturation S_w. In the last equation, the choice of unknown depends upon whether fluid rate or wellbore pressure is specified as a known control, which may vary from one well to the next.

Altogether, we get a nonlinear discrete system $\boldsymbol{F}(\boldsymbol{x}) = \mathbf{0}$, which can be solved by a standard Newton–Raphson method to compute the unknown vector $\boldsymbol{x}$. This vector consists of the primary unknowns $\boldsymbol{p}$ and $\boldsymbol{S}_w$ and complementary subsets of the vectors $\boldsymbol{q}_w$ and $\boldsymbol{p}^{wb}$, all evaluated at time $n+1$. The system has the following primary parameters: the pore volumes $\boldsymbol{\Phi}$, the transmissibilities $\mathbf{T}$, and the well indices $\boldsymbol{J}$; the latter two account for permeability and geometric factors affecting the flow. Once $\mathbf{T}$ and $\boldsymbol{J}$ have been computed, it is common in reservoir simulation to view the resulting finite-volume discretization as a computational graph, in which grid cells (with associated pore volumes $\boldsymbol{\Phi}$) and wellbores form the vertices (nodes) and $\mathbf{T}$ and $\boldsymbol{J}$ define the edges. This view is particularly useful if one wants to incorporate fluid communication among entities that are not geometrically connected (commonly referred to as non-neighboring connections).

In CGNet, we take the graph view one step further and consider the flow model as a general computational graph in which the vectors $\boldsymbol{\Phi}$, $\mathbf{T}$, and $\boldsymbol{J}$ are merely tunable factors we can calibrate to match observed well responses from the reservoir. Fundamental flow physics is incorporated automatically through the flow equations and the network topology derived from a standard grid, and by requiring that $\boldsymbol{\Phi}$ and $\mathbf{T}$ remain nonnegative during calibration. In cases where the fluid contact is not well known, it may be beneficial to extend the set of tunable parameters to include the initial saturations inside each grid cell. It can also be beneficial to include parameters that define $k_{r\alpha}$ as function of S_α.

Herein, we assume that a detailed description of the reservoir geology is not available (or largely uncertain) and that the only information we have is a description of the map outline of the reservoir, given in terms of two surfaces bounding the reservoir from above and below, and a rough estimate of representative permeability and porosity values. To build a CGNet model, we use a fictitious domain approach that starts with a uniform rectilinear mesh covering the bounding box of the reservoir and culls cells that fall outside the map outline and the surfaces bounding the reservoir from above and below as illustrated in Fig. 1. By default, the mesh is populated by uniform permeabilities and porosities, but heterogeneous properties can also be used if available.

The main role of the mesh is to define a network topology that is sufficiently rich to provide multiple network paths between the injectors and producers. Notice that the network topology does not necessarily depend on the well placement, like in GPSNet/StellNet. The mesh is also used to calculate an initial guess for the pore (or bulk) volume and the transmissibilities and well indices. Here, we have used a uniform rectilinear mesh for simplicity, but the method is general and can easily be configured with other mesh types that adapt better to the bounding surfaces and the position of the wells, including curvilinear, Voronoi, and tetrahedral meshes or standard corner-point grids. The bulk volume of the mesh is generally larger than the volume of the reservoir by definition and must be adjusted for when the associated pore volumes are calibrated during the model training. If the surfaces bounding the reservoir are known with confidence, one can obviously refine the initial estimate of $\boldsymbol{\Phi}$ by computing the fractions of each grid cell that is part of the reservoir.

3 Model training

In this study, we utilize a research simulator that incorporates adjoint simulation capabilities, enabling efficient gradient and sensitivity computations. Such advanced tools are generally limited in availability within commercial simulators, contributing to the prevalence of ensemble-based methods in calibration workflows. Methods like the ensemble

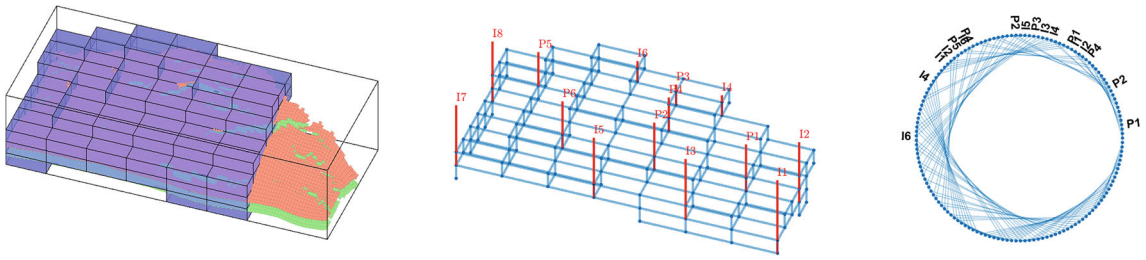

Fig. 1 Illustration of the setup of a CGNet. The left plot shows a rectilinear mesh wrapped around the assumed outline of the reservoir. This mesh is formed by first fitting a 8×8×6 uniform mesh to the bounding box of the reservoir and then removing all cells that lie above or below the top and bottom surface, respectively, or outside of their map outline. This leaves 145 out of the original 384 cells. (Some of these cells are not included in the plot to better show the top and bottom surfaces, displayed in red and green, respectively.) The middle plot shows the 3D network graph formed by connecting the cell centroids and the wells. The right plot shows the network graph as a circular ring. Nodes marked with P or I contain well perforations; the remaining are reservoir nodes

smoother with multiple data assimilation (EsMDA) [35], as employed in studies such as [25], exhibit similar convergence properties to the Gauss–Newton method employed herein but require simulating an ensemble of models, resulting in additional computational costs. Ensemble-based methods, however, offer the advantage of estimating model output uncertainty, assuming proper understanding of model input uncertainties. In our purely data-driven setting, this advantage does not apply as we lack a priori knowledge of model input uncertainties.

3.1 Misfit minimization

Let $\boldsymbol{x}^n$ denote the vector of pressures, saturations, and well rates/pressures at time step n, and let $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ denote the vector containing all our parameters (assumed constant over time). At time step n, we are solving a set of equations $\boldsymbol{F}_n(\boldsymbol{x}^n, \boldsymbol{x}^{n+1}, \boldsymbol{\theta}) = \boldsymbol{0}$, and locally we may regard each $\boldsymbol{x}^n$ as an implicit function $\boldsymbol{x}^n = \boldsymbol{x}^n(\boldsymbol{\theta})$. To train our model, we consider a set of observations $\boldsymbol{y}_{obs} \in \mathbb{R}^{N_y}$ corresponding to model outputs $\boldsymbol{y}$, where $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N)$. For training, we seek to minimize the least squares function

$$J = \frac{1}{2}\sum_{k=1}^{N_y} r_k^2 = \frac{1}{2}\boldsymbol{r}^T\boldsymbol{r}, \tag{4}$$

where $\boldsymbol{r}$ is the vector of residuals, i.e., $r_k = w_k(y_k - y_{k,obs})$. We impose upper and lower bounds for each parameter and further assume that all values within a bound interval are equally likely. As a result, there is no regularization in the objective (4), so for a given set of training data, there will in general not be a unique optimal solution. In the numerical examples (reported in the next section) we consider *perfect* observations obtained from a *high-resolution* reservoir simulation. Accordingly, each residual weight w_k is just a constant representing the reciprocal of a typical value magnitude (e.g., typical rate, pressure range, etc.), introduced for normalization purposes.

For minimization of (4) we mainly utilize Levenberg–Marquardt (see e.g., [36]), but for comparison, we also consider a purely gradient-based quasi-Newton method (L-BFGS-B). Both methods are popular choices within adjoint-based history matching (see e.g., [37]), however since the Levenberg–Marquardt method requires the full *parameter-to-output* sensitivity matrix, it is usually not considered feasible for large-scale problems in its pure form. For our application, however, the state space is so small that all sensitivities can efficiently be computed in a single adjoint simulation. The two approaches are briefly outlined next.

3.2 L-BFGS-B

We consider a version of the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm with bound constraints *L-BFGS-B* [38], where the Hessian approximation at each accepted optimization iteration is updated using L-BFGS and is projected to the active parameter set to obtain search directions. A cubic line-search with acceptance based on strong Wolf conditions is used to obtain step lengths. The gradient $\nabla_\theta J$ of the mismatch function (4) with respect to a given set of parameters $\boldsymbol{\theta}$ is obtained by running an adjoint simulation. In particular, we have that

$$\nabla_\theta J = \sum_{n=1}^{N} \frac{\partial \boldsymbol{F}_n}{\partial \boldsymbol{\theta}}^T \boldsymbol{\lambda}^n,$$

where the Lagrange multipliers $\boldsymbol{\lambda}^n$ are obtained by the linear adjoint equations

$$\frac{\partial \boldsymbol{F}_n}{\partial \boldsymbol{x}^n}^T \boldsymbol{\lambda}^n = -\frac{\partial J}{\partial \boldsymbol{x}^n}^T - \frac{\partial \boldsymbol{F}_{n+1}}{\partial \boldsymbol{x}^n}^T \boldsymbol{\lambda}^{n+1}, \quad n = N, N-1, \ldots, 1. \tag{5}$$

In (5) the last term is omitted for $n = N$.

3.3 Levenberg–marquardt

Classical Gauss-Newton-type methods use the misfit Jacobian

$$\nabla_\theta \boldsymbol{r} = \left[\nabla_\theta r_1, \nabla_\theta r_2, \ldots, \nabla_\theta r_{N_y}\right]$$

to approximate the Hessian by $\nabla^2 J \approx \nabla_\theta \boldsymbol{r} \nabla_\theta \boldsymbol{r}^T$. Since the term containing the second-order derivatives vanishes for $\boldsymbol{r} = \boldsymbol{0}$, such methods typically perform very well for problems with sufficiently small residuals. The Levenberg–Marquardt method introduces a damping parameter α such that update candidates $\delta\boldsymbol{\theta}$ for the model parameters are obtained by solving

$$\left(\nabla_\theta \boldsymbol{r} \nabla_\theta \boldsymbol{r}^T + \alpha \boldsymbol{I}\right) \delta\boldsymbol{\theta} = -\nabla_\theta J.$$

Our implementation updates the damping parameter between iterations following a trust-region logic. Each column of $\nabla_\theta \boldsymbol{r}$ can be obtained by an adjoint simulation, but since we consider reduced (low-order) models with a limited number of state variables, it is more practical to set up a single adjoint system with multiple right-hand-sides. In particular, we have that

$$\nabla_\theta \boldsymbol{r} = \sum_{n=1}^{N} \frac{\partial \boldsymbol{F}_n}{\partial \boldsymbol{\theta}}^T \boldsymbol{\Lambda}^n,$$

where the matrix of all Lagrange multipliers are obtained by the linear matrix adjoint equations

$$\frac{\partial \boldsymbol{F}_n}{\partial \boldsymbol{x}^n}^T \boldsymbol{\Lambda}^n = -\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{x}^n}^T - \frac{\partial \boldsymbol{F}_{n+1}}{\partial \boldsymbol{x}^n}^T \boldsymbol{\Lambda}^{n+1}, \quad n = N, N-1, \ldots, 1. \tag{6}$$

We here solve (6) using a direct method based on sparse matrix factorization, and as a result, the linear solver time is much less compared to solving N_y individual adjoint equations. We also note that when $N_y \sim N_\theta$, we could have computed $\nabla_\theta \boldsymbol{r}$ from forward linearized equations with similar efficiency.

4 Numerical examples

This section discusses applications of CGNet to three different cases of waterflooding. None of the cases have real well observations, so we will instead use simulated well data from a full 3D model for calibration. The first case is fully conceptual and describes a simple channelized reservoir [39]. The second case uses reservoir geometry and petrophysical properties from a real simulation model of the Norne oil and gas field but describes a synthetic and simplified production scenario. The only reservoir information we use from the simulation models when building the CGNet is the map outline, bounds on the reservoir from above and below, well positions, and field-average values for permeability and porosity. Bounds on the reservoir volume is given as piecewise (bi)linear surfaces, but instead of computing possible intersections with such surfaces, we simply count the number of cell centroids from the simulation model that fall inside each cell in the rectilinear mesh and use this count to cull cells.

All numerical results are obtained using solvers and workflow tools implemented in the open-source MRST toolbox [34, 40, 41].

4.1 The Egg model

The Egg ensemble consists of 101 realizations of a channelized reservoir [39], each described on a $60 \times 60 \times 7$ rectilinear grid with 18 553 active cells. Permeabilities vary two orders of magnitude and are highly correlated across the layers; the porosity is constant. We use the base realization to simulate 123 time steps of production data. Initially, the reservoir contains a mixture of water and oil $[S_w, S_o] = [0.2, 0.8]$. The two-phase system is weakly compressible, has cubic and quartic relative permeabilities curves, and 5 cP and 1 cP viscosities for the oil and water phases, respectively. In the original setup, the reservoir is produced under waterflooding conditions with eight water injectors, operating at a constant rate but with an upper limit on the bottom-hole pressure of 420 bar, and four producers operating at a constant bottom-hole pressure. To make the production history more eventful, we perturb the injection rates randomly every fourth time step in the interval [59.5, 99.5] m^3/day and likewise the bottom-hole controls for the producers in the interval [391, 396] bar.

4.1.1 Ability to match data

To construct a CGNet, we need the map outline of the reservoir, the depth of the surfaces that bound the reservoir from above and below (which here are horizontal), and a rough estimate of the lateral and vertical permeability (we use the field-averaged values). The lateral resolution of the rectilinear mesh is chosen so that no wells end up within the same mesh cell, which is the case, e.g., with a 6×6 lateral mesh, as shown to the left in Fig. 2. As a first start, we simply reuse the fluid model from the fine-scale model without any calibration of parameters. The right plot in Fig. 2 shows that the match obtained after 20 iterations with the Levenberg–Marquardt algorithm is remarkable, given that the model only has 33 cells and 99 tunable parameters. This clearly demonstrates the ability of CGNet to adapt to data.

From the iteration history reported in Fig. 3, we see that the misfit is reduced very fast during the first few iterations, e.g., dropping almost three orders of magnitude over the first three

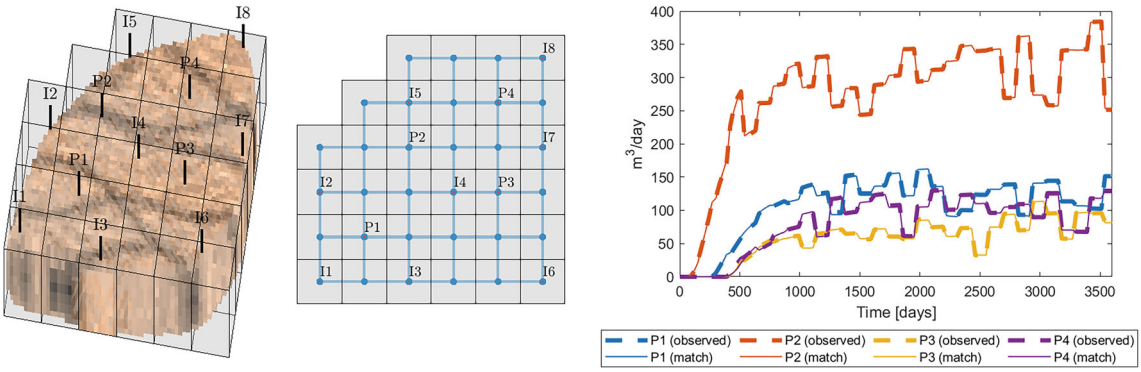

Fig. 2 Calibration of a CGNet model for the Egg case based on a $6\times6\times1$ mesh, which after culling has 33 cells. The left plot shows the rectilinear mesh overlaid on the fine scale model (colors: logarithm of lateral permeability). The right plot shows the match in water rates from the four producers

iterations, demonstrating the utility of the LM method. With only a single vertical layer and matching of rock parameters only (pore volume, transmissibility, and well indices), the misfit reduction levels off after 8–9 iterations.

Increasing the mesh resolution to $6 \times 6 \times 3$ so that the CGNet has 99 nodes and 363 tunable parameters reduces the misfit by a factor 2.7. Alternatively, the fluid model has six parameters,[2] which can be tuned in each reservoir node, giving a total of 297 tunable parameters for the $6\times6\times1$ mesh. This reduces the misfit by a factor 4.0 compared with the base model and is, overall, a better strategy here since it does not increase in the number of network nodes, which is the most important factor that influences the cost of evaluating the network model. Combining the two strategies gives a total of 957 parameters and a reduction with a factor 5.7 compared with the base model.

4.1.2 Predictive power

To assert the predictive power of the CGNet, we only use a subset of the well observations for training. Figure 4 reports the prediction of water rates in well P4 for the CGNet with 297 parameters calibrated using ten LM iterations on well observations from the first 64 time steps only. At this time, all producers have observed water breakthrough and reached a "semi-steady" phase. There is not a significant increase in the misfit during the prediction period, and for all four producers, the cumulative oil and water production are predicted to within an error of 1.5% and 0.6%, respectively.

Figure 5 reports how the error in cumulative oil and water production predicted by CGNet decays as a function of the number of data points (time steps) used for training. To obtain reasonable predictions it seems important that the training data contain representative variations in water rates from the predominant "semi-steady" phase that follows the water breakthrough in all producers. This period starts around day 1000 (step 36) in all producers except P2, which experiences earlier breakthrough.

To further emphasize the significance of incorporating representative data for predominant flow regimes, we conducted a training exercise using a simulation horizon limited to 900 days. This corresponds to the first quarter of our previous time horizon, where water breakthrough has occurred in all producers but a semi-steady plateau for water rates has not yet been established. To align with this reduced timeframe, both the time steps and the period of random perturbations were scaled down by one fourth.

The training data, now consisting of 32 time steps, no longer include any water breakthrough in wells P1, P3, and P4, and only account for the initial increase in water rates for producer P2. Consequently, the predictive capability of CGNet significantly deteriorates. Notably, the largest errors

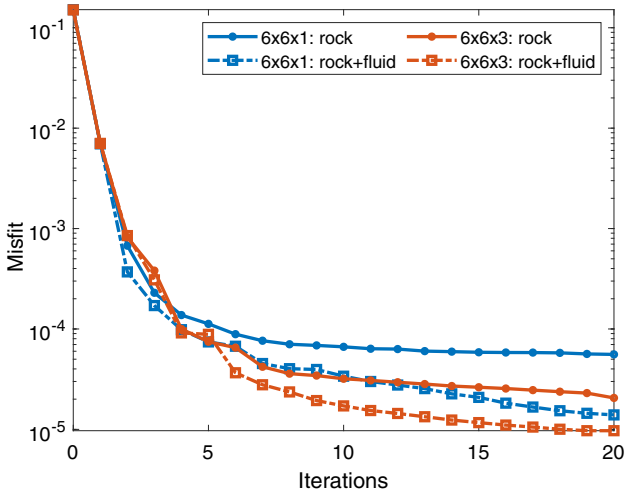

Fig. 3 Comparison of misfit reduction over 20 iterations for the Egg case when calibrating rock parameters or both rock and fluid parameters

[2] In a two-phase Corey-type fluid model, the relative permeability curve for each phase can be uniquely determined by three parameters: the exponent and two values representing the endpoint.

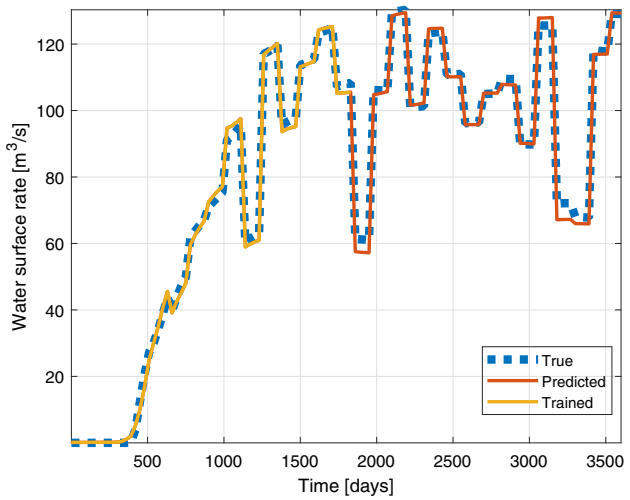

Fig. 4 Prediction of well P4 for a 6×6×1 CGNet with rock and fluid parameters calibrated only for the first 64 out of the 123 time steps

in cumulative oil and water production are observed in well P3, with deviations of 19% and 42% respectively. Since pressure communication is nearly instantaneous in an (almost) incompressible flow model, pressure signals resulting from varying well controls lack sufficient information for calibrating pore volumes and interwell transmissibilities.

To address this limitation, we extend the training data to step 64, ensuring that water breakthroughs are included in all wells. This refinement results in a substantial reduction in the error of predicted cumulative oil production, with deviations below 1% in all wells except for P1, which still exhibits an error of 6.9%. However, the predicted water production remains overestimated by a factor of 15% in all wells except P2, where the training data now cover the complete period from initial water breakthrough to the "semi-steady" phase.

Altogether, this example demonstrates the importance of training CGNet on representative data but also suggests that this type of model may be best suited for scenarios where the predominant dynamics consists of variations around a "semi-steady" phase.

4.2 The Norne field

In our next case, we focus more on the ability to capture the effects of complex geology. To this end, we use a simplified version of a full simulation model of the Norne oil and gas field from the Norwegian Sea,[3] which is one of the standard test cases from the test suite module of MRST. The reservoir is represented on a 46 × 112 × 22 corner-point grid having 44 915 active cells and contains complex geological features like faults, displaced layering, pinched cells, internal gaps, non-neighboring connections, etc. Petrophysical parameters are generated as described by [42] but with no modification of connectivity across faults. Whereas the original simulation model describes a three-phase, water-alternating-gas scenario, the simplified model considers a waterflooding scenario of a reservoir initially filled completely by oil with vertical wells positioned so as to create a dominant five-spot pattern in the main body of the reservoir and much weaker line drives in the two "legs" of the reservoir. In the original setup, the six injectors operate at constant water rates and the producers at constant bottom-hole pressure. We modify this setup by randomly perturbing the imposed water rates in the six injectors and the bottom-hole pressures of the five producers every fourth time step.

The only information we utilize to start building the CGNet model is the map outline of the reservoir, the depth of the surfaces that bound the reservoir from above and below, and field-averaged values for permeability and the product of porosity and net-to-gross. The fluid model, however, is the same as in the true case: two-phase, dead-oil with constant formation-volume factors, quadratic relative permeabilities with zero residual saturations, and constant viscosities with an oil–water viscosity ratio of 5:1. That is, we make no attempt to calibrate the parameters in this model to account for the multiphase upscaling effect one can expect will be present when moving from a fine-scale corner-point model to a (very) coarse rectilinear mesh.

4.2.1 Ability to match data

We chose to start, somewhat arbitrarily, with a 6 × 6 × 1 mesh (as we did for the Egg model), which after culling cells that fall outside the reservoir gave a mesh with 24 cells, as shown in Fig. 6. The mesh is so coarse that wells I1 and P1 fall inside the same cell and are thus connected to the same reservoir node in the network model. Despite this, 20 iterations with Levenberg–Marquardt were able to reduce the mismatch value by a factor 140 so that the calibrated model reproduces qualitatively correct behavior, except in P5, which has the lowest water production among the four wells that experience significant water breakthrough during the simulation.

By increasing the lateral resolution to 8 × 8 (giving a mesh with 35 cells), we are able to reduce the mismatch one order further within 20 iterations, thereby reproducing water breakthrough in P5 and generally obtaining a much better qualitative match for the fluid rates in the producers and the bottom-hole pressures in the injectors. If we increase the lateral resolution further to 12 × 12 or 16 × 16, the misfit minimization does not converge properly and the resulting models fail to reproduce water breakthrough in P2, P3, and P5. One obvious explanation is that the top and bottom surface exhibit significant depth variation that is not possible

[3] The full model is published as an open data set on Github (github.com/OPM/opm-data) by the Open Porous Media (OPM) initiative.

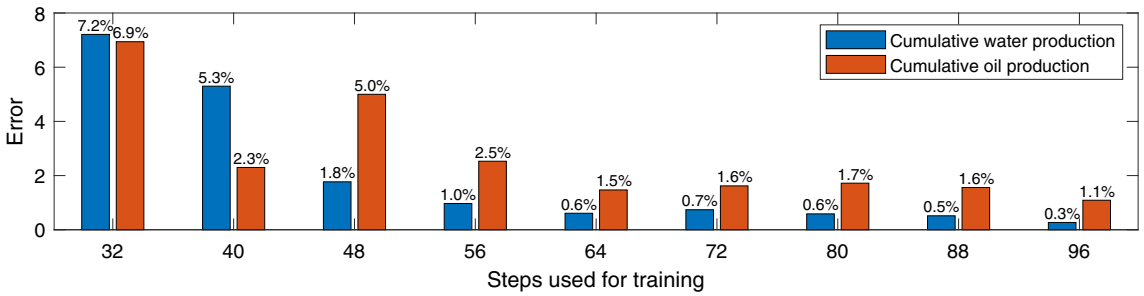

Fig. 5 Error in cumulative oil/water production as function of the number of time steps used for training the 6×6×1 CGNet with 297 tunable parameters using ten iterations of misfit minimization

to represent using a single, fully horizontal layer of cells. Another possible hypothesis could be that the reservoir has significant internal layering, which we are not able to resolve with a single-layered mesh. (This is indeed the case for the Norne reservoir. It exhibits significant variation in lateral and vertical permeability, spanning four and five orders of magnitude, respectively, across different sedimentary layers. Moreover, there is a notable lack of vertical communication between the upper three layers of grid cells and the rest of the model.) To investigate these hypotheses, we increase the vertical resolution from 1 to 4, giving a total of 94 cells in the 4-layered rectilinear mesh after culling. This reduces the mismatch by a factor two and improves the quantitative prediction of production rates, particularly for P3 and P5.

As commented earlier, the rectilinear grid is generally a loose fit with the true model. For the 8 × 8 × 4 model with 94 cells, the initial pore volume of the network graph before calibration overestimates the true pore volume of the reservoir by a factor 4.25. A better match can be obtained if we make a mesh that fits tighter, e.g., by culling all cells that contain less than 70 centroids. This results in a mesh with 68 cells and a reduction in misfit by a factor 1.7.

Figure 7 summarizes the reduction in misfit over 20 iterations for the different initial meshes and compares the match obtained for wells P1 to P3. The relative quality of the match for P1 is much better than for P2 and P3, because deviations in phase rates are weighted equally for all producers and P1 has much higher liquid rates and experiences earlier water breakthrough. In terms of *absolute errors*, the matches for the other wells are (at least) equally good as for P1, so that the overall field oil rate is matched to within a relative error of 2%; see Fig. 8.

4.2.2 Comparison of LM and L-BFGS-B

Continuing the iteration process beyond 15–25 iterations did not lead to significant improvement for any of the CGNet configurations considered so far; see Fig. 9. As an alternative, we also tried to minimize the Lagrangian misfit function using a variant of the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm with bound constraints (L-BFGS-B) [38] with the necessary gradients computed using adjoint equations and automatic differentiation. The Hessian approximations are projected to the active parameter set to obtain search

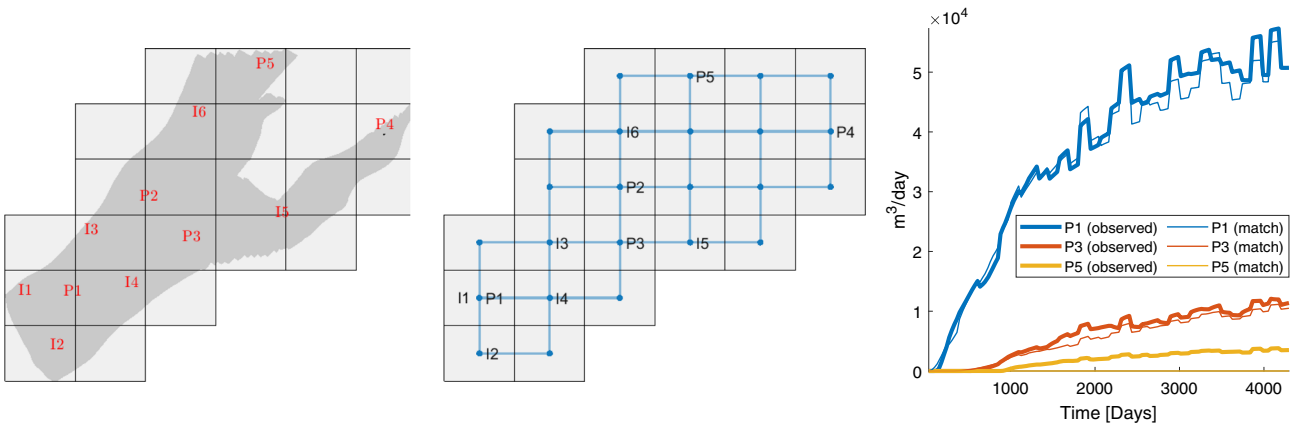

Fig. 6 Calibration of a CGNet for the Norne case based on a 6×6×1 mesh. The right plot shows the observed and predicted water rates for three of the producers

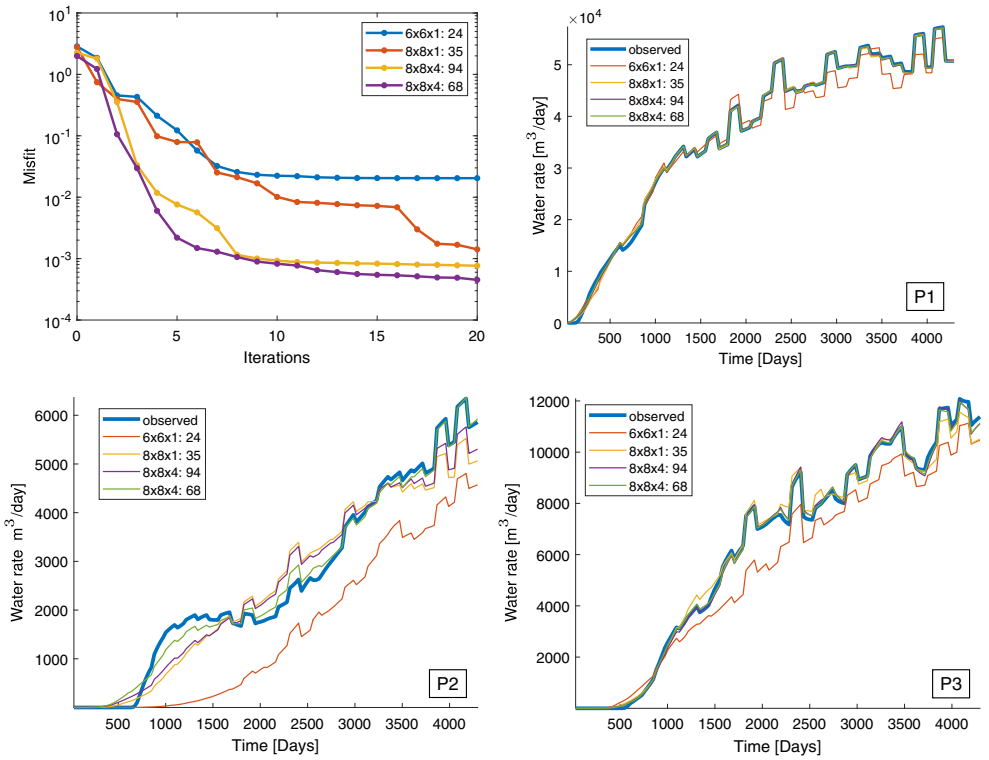

Fig. 7 Match after 20 iterations with Levenberg–Marquardt for the Norne field. The upper-left plot shows reduction in misfit, whereas the upper-right, lower-left, and lower-right plots show water rates for producers P1 to P3

directions, and a cubic line-search with acceptance based on strong Wolf conditions is used to obtain step lengths. Most iteration steps are accepted without the need for further line search, but compared with LM, the initial reduction is much slower and the minimization process stagnates at misfit values that are one order of magnitude higher or more. Superior convergence of LM compared to the quasi-Newton approach is expected when residuals are *small*. The apparent stagnation of the latter could also, at least partly, be due to the fact that the BFGS updates become increasingly sensitive to gradient accuracy for decreasing residuals.

4.2.3 Comparison with GPSNet

To assess the efficiency of the CGNet approach, we compare it with the interwell network model approach, here

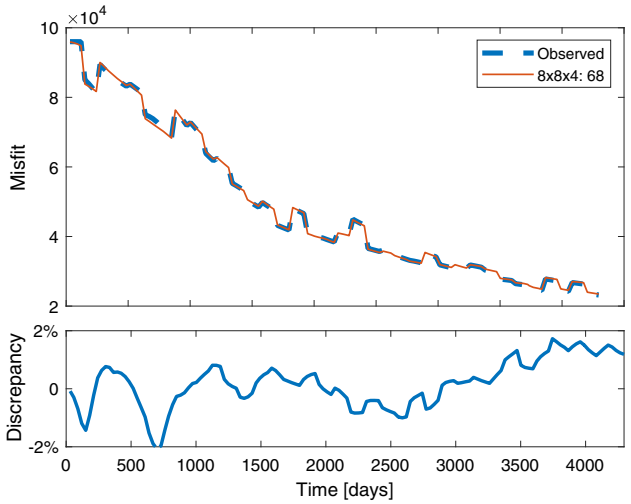

Fig. 8 Observed field oil production rates compared with rates simulated by CGNet derived from a 8×8×4 culled calibrated with 50 LM iterations; relative errors are shown in the lower plot

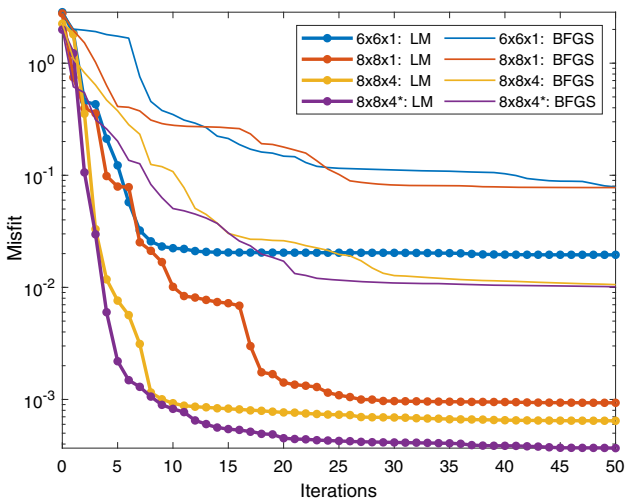

Fig. 9 Comparison of misfit reduction over 50 iterations with the LM and L-BFGS-B methods for Norne. (∗ = tighter culling of the rectilinear mesh.)

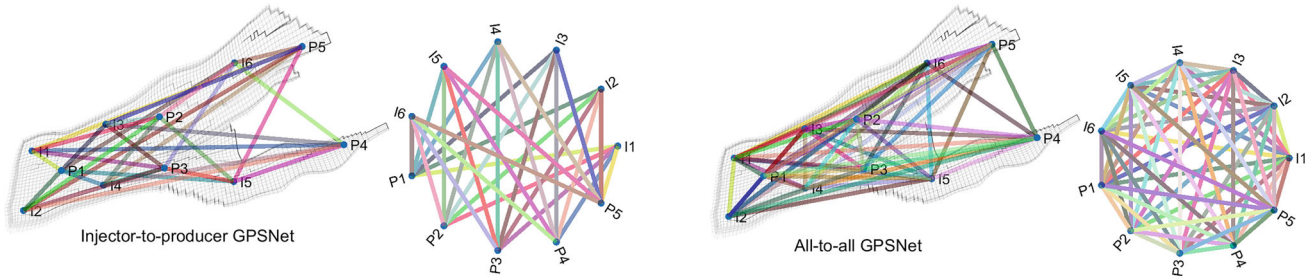

Fig. 10 The two different GPSNet setups for the Norne field case. The network to the left has one flow path connecting each injector–producer pair, whereas the network to the right also has connections among the injectors and among the producers

represented by GPSNet [25] as implemented in the *network-models* module of MRST. These models represent the reservoir in terms of a limited set of flow paths that connect injectors and producers, which are mapped onto a rectangular mesh, so that the resulting flow models can be a set up as a system of non-communicating grid layers in any (standard) simulator. The flow paths terminate in perforated well cells at each end, and this allows fluid communication between different flow paths connected to the same well without having to go through the well itself.

The interwell network can be configured in many different ways. Here, we assume that each interwell connection consists of a single flow path, discretized using 10 cells as suggested in [25], and consider two different network topologies: injector-to-producer and all-to-all communication, as illustrated in Fig. 10, giving 30 and 55 flow paths, respectively. The choice of injector-producer connectivity serves as the default option for GPSNet-type network models and is intuitive from a streamline simulation standpoint. This configuration enables time-varying interactions between wells due to fluctuations in rate or bottom-hole pressure controls, as long as the interwell connections remain relatively stable without significant changes such as well shut-ins or conversions between producers and injectors. However, the inclusion of injector-injector and producer-producer connections in GPSNet provides greater flexibility for fluid movement and has demonstrated its advantages. For a more comprehensive discussion on network topology, please refer to [9].

Regardless of the network topology, every flow path has a transmissibility and a pore volume that can be tuned, which means that altogether, there are 71 adjustable parameters for the injector–producers model: 30 transmissibilities, 30 pore volumes, and 11 well indices. These are calibrated in exactly the same way as for CGNet using the Levenberg–Marquardt method.

To initialize the model parameters, we need to know the well positions and representative permeability and porosity values, like for CGNet. However, unlike for CGNet, we do not need to know the outline of the reservoir; a good estimate of the reservoir volume is sufficient. In lack of more precise information, this is distributed in equal fractions to the individual flow paths.

Table 1 summarizes the number of adjustable parameters and nodes in the resulting computational networks for the CGNet and GPSNet models we will compare. The first thing to notice is that CGNet has a much higher ratio of parameters to nodes in the compute graph compared to GPSNet, giving the network more freedom for adjustment. Fewer nodes in the computational graph may suggest that a CGNet is faster to evaluate than a GPSNet, and this may indeed be the case using a standard simulator. However, GPSNet's discrete systems have a tridiagonal structure and can, in an optimized implementation, be solved using a highly efficient Gauss–Seidel type nonlinear solver [43] that starts at the injector cells and proceeds towards the producers, solving one cell at the time.

The left plot of Fig. 11 reports misfit reduction for the network models from Table 1. The two GPSNets have almost the same number of parameters as the $8 \times 8 \times 1$ CGNet, and it is interesting to see how much faster they calibrate. This has a simple explanation: In our initial setup of CGNet, we removed mesh cells that are outside of the reservoir but did not adjust the associated bulk volumes (which determine the initial values of Φ) to account for the fact that only a fraction of each cell is inside the reservoir. Hence, whereas the initial volume of the GPSNet models match the true reservoir volume, the $8 \times 8 \times 1$ CGNet model *overestimates* this

Table 1 The number of adjustable parameters and graph nodes in different CGNet and GPSNet models

	CGNet			GPSNet	
setup	$8 \times 8 \times 1$	$8 \times 8 \times 4$	$8 \times 8 \times 4^*$	inj–prod	all-to-all
# parameters	99	305	204	71	121
# nodes	35	94	68	300	550

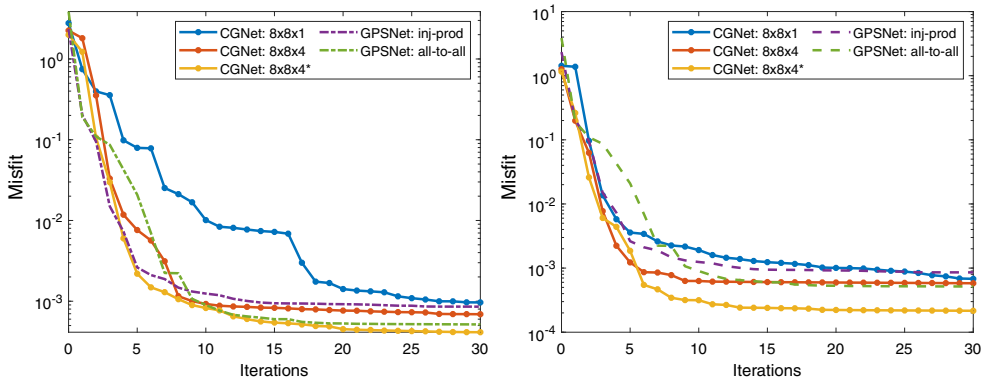

Fig. 11 Comparison of the misfit reduction for CGNet and GPSNet during the Levenberg–Marquardt minimization. In the left plot, CGNet is initialized without using precise information of the reservoir volume. In the right plot, the bulk volumes of the outer nodes have been adjusted to account for the fact that only a fraction of the associated cells in the rectilinear mesh is inside the reservoir

volume by a factor six (and the other two by a factor 4.0 and 2.9, respectively). Fortunately, this is simple to rectify, and if we adjust the initial volumes so that the volumes of nodes never exceed the true volume of the corresponding reservoir section, e.g., by more than a factor 20%, the initial reduction for the $8 \times 8 \times 1$ CGNet is as fast as for the other models. By 30 iterations, this model has same level of accuracy as the two GPSNets and the 4-layer CGNet.

4.3 Sector model with horizontal oil–water contact

In the two examples discussed so far, we assumed the initial fluid distribution to be uniform throughout the whole domain. This is obviously not the case in general, but we made this simplification to discuss one issue at the time. In the last example, we consider a synthetic sector model that contains a horizontal oil–water contact and is produced by a line drive consisting of two injectors and three producers, as shown in Fig. 12. The setup is a modified version of the model discussed in Lie [34, Section 15.4.4].

Unlike in the previous two examples, where we used rectilinear meshes fitted to the bounding box of the reservoir, we now use curvilinear meshes that follow the reservoir's bounding surfaces to set up our CGNet models. Figure 13 shows our three choices of meshes.

To calibrate the CGNets, we consider four different choices of parameters for each mesh (Table 2): (i) rock parameters only, (ii) rock parameters and initial saturation, (iii) rock and fluid parameters, and (iv) rock and fluid parameters and initial saturation. Figure 14 reports the misfit reduction over 21 iterations with LM.

With only one layer in the network topology, CGNet fails to match well responses accurately (left plot in Fig. 15) if we only calibrate pore volume, transmissibilities, and well indices, as these rock parameters do not give any means to represent the fluid segregation in the reservoir. If we also calibrate initial saturation, the misfit improves by more than two orders of magnitude and the calibrated CGNet reproduces the observed production curves accurately; see the right plot in Fig. 15. Adding fluid parameters instead of initial saturation

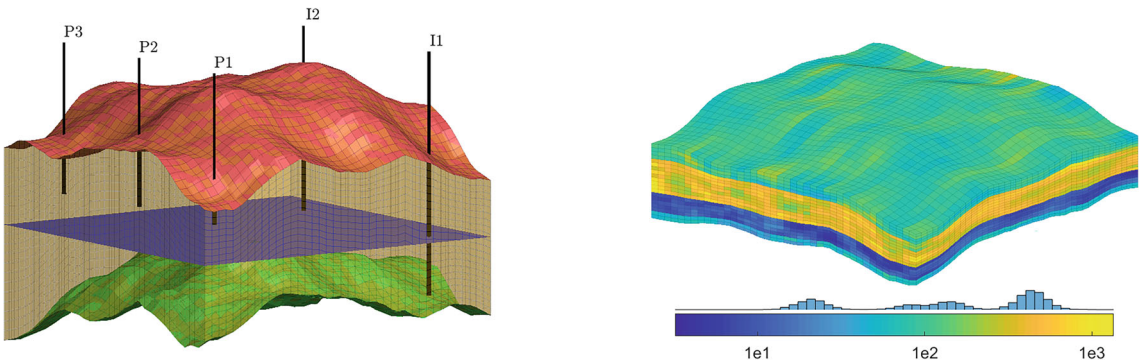

Fig. 12 A sector model with a horizontal oil–water contact modelled on a $40 \times 40 \times 20$ corner-point grid. The left plot shows the upper and lower bounding surfaces of the reservoir, the fluid contact, and the position and perforation of the five wells. The two injectors are perforated through the whole reservoir, whereas the producers are only perforated in the upper 2/3 of the oil zone. The right plot shows the horizontal permeability, which is lognormally distributed within each of the four deposition units

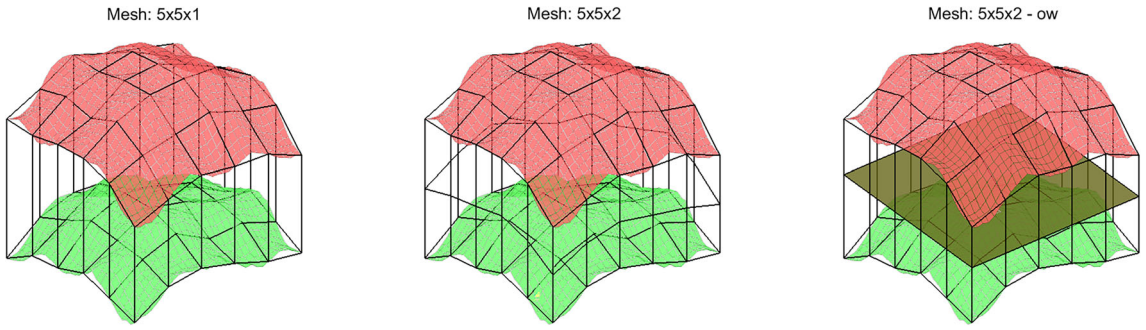

Fig. 13 Three different choices of coarse meshes: a $5 \times 5 \times 1$ and a $5 \times 5 \times 2$ mesh that follow the top and bottom surfaces and a $5 \times 5 \times 2$ mesh that also adapts to the initial oil–water contact

gives a similar but less pronounced effect, whereas calibrating all three parameter types gives the overall best match. This is a good example of a case where the network model reproduces the observed well responses accurately, but where the calibrated parameters should not be interpreted as representative for the associated rock volume in the volumetric mesh used to build the network topology.

With two layers in the topology, the network model can represent vertical fluid movement explicitly, which reduces the overall misfit slightly compared with the best single-layer model, even when only rock parameters are calibrated. On the other hand, adding more calibration parameters reduces the misfit by at most 50% since the match is already very good.

If we adapt the mesh so that the interface between the two cell layers follows the horizontal oil–water contact, the network model will represent the initial fluid distribution correctly in a physical sense, and we hence get a very good match by only calibrating rock parameters. If we continue iterating, the misfit decays slowly and ends up at $1.7 \cdot 10^{-5}$ after one hundred iterations. Adding initial saturation as a calibration parameter accelerates the convergence and improves the mismatch by 15%. If we continue iterating, the mismatch drops to $8.2 \cdot 10^{-6}$ after one hundred iterations. The two CGNets that calibrate fluid parameters have the lowest misfits after 21 iterations, but if we continue iterating, the misfit minimization stagnates after 29 and 33 iterations at values $1.5 \cdot 10^{-5}$ and $2.1 \cdot 10^{-5}$, respectively, because the updates fall below a lower tolerance (here, chosen to be 10^{-10} to force the iteration to continue as long as possible). For misfit values so small, errors in the gradients coming from the numerical solution of the adjoint equations will start to affect the minimization.

Our overall conclusion is that the parameter spaces for all the $5 \times 5 \times 2$ CGNets are large enough to allow very good matches, particularly when adapting the mesh to the oil–water interface. However, in choosing among the four versions, we prefer the CGNet with rock parameters and initial saturations as free parameters and avoid calibrating fluid parameters as this would effectively result in multiple fluid regions, which may be somewhat cumbersome in certain simulators.

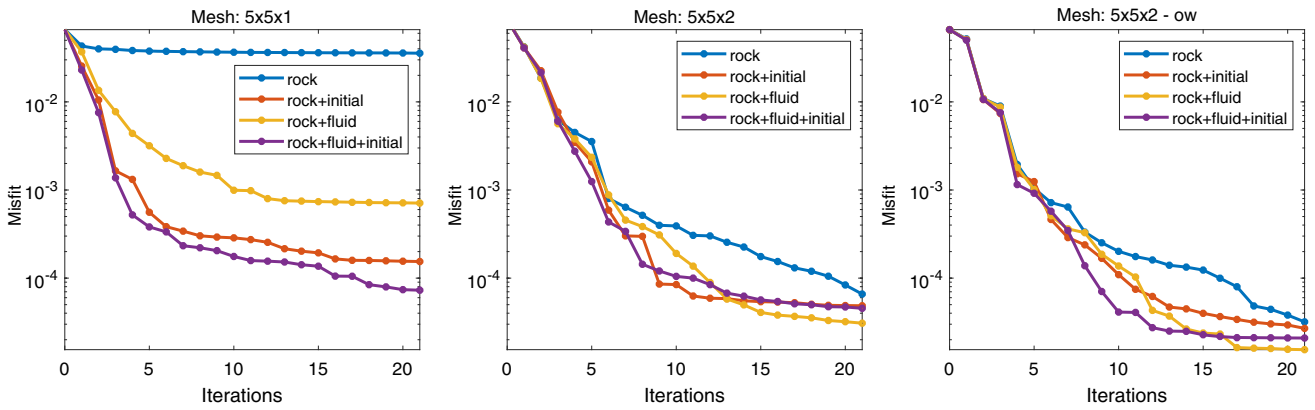

Fig. 14 Misfit reduction over 21 iterations with LM for the sector model with three different meshes and four different choices of calibration parameters

Table 2 The number of tunable parameters in the different CGNets for the sector model

	$5 \times 5 \times 1$	$5 \times 5 \times 2$	$5 \times 5 \times 2$ – ow
rock	70	162	162
rock + initial	95	212	212
rock + fluid	220	462	462
rock + fluid + initial	245	512	512

5 Concluding remarks

We have introduced a simple idea for constructing network models that has proved to work (surprisingly) well on a number of problems; three of these have been reported herein. Our CGNets are easy to set up; all you essentially need is a coarse volumetric mesh that covers the assumed outline of the reservoir, preferably designed so that no wells are included in the same mesh cell. Herein, we have used either a rectilinear mesh or curvilinear mesh, but the same idea applies without changes to other volumetric tessellations (tetrahedral, prismatic, general Voronoi, etc.). In addition, you need a rough estimate of representative petrophysical properties and some idea of a suitable fluid model and potential fluid contacts.

Given these, CGNets calibrate relatively robustly, even when you can only provide very crude initial guesses for transmissibilities, pore volumes, and the overall reservoir outline and bulk volume. The importance of calibrating specific parameters depends on the starting point of the calibration process. If the initial pore volumes and/or transmissibilities deviate significantly, CGNet may fail to predict any fluid breakthrough, rendering the optimizer unable to determine the correct direction based on gradients.

In reservoirs initially filled solely with oil, fluid breakthrough is primarily influenced by pore volumes, transmissibilities, and well indices. Initially, it is crucial to get these parameters roughly right before starting to fine-tune them, along with the ones related to the fluid model, to ensure accurate prediction of fluid behavior and match the intricate details observed in production curves. Similarly, for models with initial fluid stratification, the calibration of initial saturation holds equal significance in achieving reliable predictions.

Sharper initial estimates of the reservoir outline and possible fluid contacts will generally improve the model accuracy and speed up the calibration process, but for models with complex geometry our experience indicates that the misfit minimization is faster if you start with a slightly overestimated bulk volume (up to 20–50%). For network models with a few hundred network nodes and up to a few thousand parameters, our experience is that the Levenberg–Marquardt method by far outperforms quasi-Newton methods like L-BFGS-B. For models with many hundred wells, or with simulators that are incapable of computing gradients, other optimization methods like ES-MDA may be necessary.

Compared with interwell network models, CGNets have a richer network topology and have more tunable parameters per network node. As a result, a calibrated CGNet will in a typical setup therefore have fewer network nodes than an interwell network model of comparable accuracy and hence be faster to evaluate in a forward simulation by a standard simulator. Likewise, because the CGNets are essentially a very coarse standard simulation model, it is more straightforward to include information about fluid contacts and petrophysical heterogeneity.

Our motivation for developing CGNet has been to provide compact models that are quick to calibrate and fast to evaluate

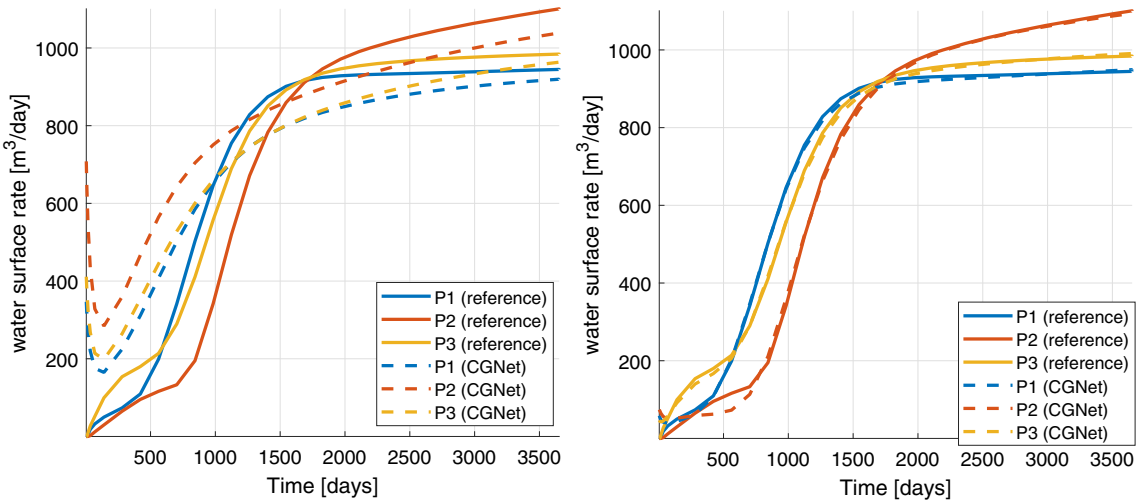

Fig. 15 Observed water production rates compared with rates simulated by CGNet with network topology corresponding to a $5 \times 5 \times 1$ mesh. In the left plot, only rock parameters have been calibrated using 21 LM iterations, whereas the right plot also includes calibration of initial saturation

for optimization purposes in settings where you have access to a significant amount of production data. (One example would be short-term rate optimization, as a more physics based alternative to pure data-driven models.) Given the limited amount of information necessary to set up a CGNet, it may be tempting to try to use them for early-stage history matching, too. However, as the Egg example illustrated, this will likely be futile for cases with low compressibility, since observed production data will not contain much useful information to determine effective interwell transmissibilities and communication volumes before the breakthrough of displacing fluids. Whether CGNets can prove useful for early-stage history matching of more compressible systems remains to be investigated.

Acknowledgements The authors acknowledge funding from the Research Council of Norway through grant no. 280950 with co-funding from Equinor Energy AS, TotalEnergies EP Norge AS, and Wintershall DEA Norge AS.

Declarations

Conflict of Interests/Competing Interests The research leading to the results reported in this manuscript received funding from the Research Council of Norway through grant no. 280950 which includes co-funding from Equinor Energy AS, TotalEnergies EP Norge AS, and Wintershall DEA Norge AS. Apart from this funding, the authors have no financial or competing interests to declare that are relevant to the content of this ar

References

1. Navrátil, J., King, A., Rios, J., Kollias, G., Torrado, R., Codas, A.: Accelerating physics-based simulations using end-to-end neural network proxies: an application in oil reservoir modeling. Frontiers in Big Data **2**, 33 (2019). https://doi.org/10.3389/fdata.2019.00033
2. Jin, Z.L., Liu, Y., Durlofsky, L.J.: Deep-learning-based surrogate model for reservoir simulation with time-varying well controls. J. Petrol. Sci. Eng. **192**, 107273 (2020). https://doi.org/10.1016/j.petrol.2020.107273
3. Kim, J., Yang, H., Choe, J.: Robust optimization of the locations and types of multiple wells using CNN based proxy models. J. Petrol. Sci. Eng. **193**, 107424 (2020). https://doi.org/10.1016/j.petrol.2020.107424
4. Kim, Y.D., Durlofsky, L.J.: A recurrent neural network-based proxy model for well-control optimization with nonlinear output constraints. SPE J. **26**(04), 1837–1857 (2021). https://doi.org/10.2118/203980-PA
5. Liu, Z., Reynolds, A.C.: Gradient-enhanced support vector regression for robust life-cycle production optimization with nonlinear-state constraints. SPE J. **26**(04), 1590–1613 (2021). https://doi.org/10.2118/204236-PA
6. Wang, N., Chang, H., Zhang, D.: Theory-guided auto-encoder for surrogate construction and inverse modeling. Comput. Methods Appl. Mech. Eng. **385**(114037) (2021). https://doi.org/10.1016/j.cma.2021.114037
7. Zhang, K., Wang, Y., Li, G., Ma, X., Cui, S., Luo, Q., Wang, J., Yang, Y., Yao, J.: Prediction of field saturations using a fully convolutional network surrogate. SPE J. **26**(04), 1824–1836 (2021). https://doi.org/10.2118/205485-PA
8. Zhong, Z., Sun, A.Y., Ren, B., Wang, Y.: A deep-learning-based approach for reservoir production forecast under uncertainty. SPE J. **26**(03), 1314–1340 (2021). https://doi.org/10.2118/205000-PA
9. Lie, K.A., Krogstad, S.: Comparison of two different types of reduced graph-based reservoir models: Interwell networks (GPSNet) versus aggregated coarse-grid networks (CGNet). Geoenergy Sci. Eng. **221**, 111266 (2023). https://doi.org/10.1016/j.petrol.2022.111266
10. Khodabakhshi, M., Jafarpour, B., King, M.J.: Field applications of a multi-scale multi-physics history matching approach. In: SPE Reserv.- Simul. Symp. (2015). https://doi.org/10.2118/173239-MS
11. Khodabakhshi, M., Jafarpour, B., King, M.J.: Inference of global reservoir connectivity from static pressure data with fast coarse-scale simulation models. Math. Geosci.**51**(5), 625–648 (2019). https://doi.org/10.1007/s11004-018-9772-8
12. Yousef, A.A., Lake, L.W., Jensen, J.L.: Analysis and interpretation of interwell connectivity from production and injection rate fluctuations using a capacitance model. In: SPE/DOE Symposium on Improved Oil Recovery, Tulsa, Oklahoma, USA (2006). https://doi.org/10.2118/99998-MS
13. Holanda, R., Gildin, E., Jensen, J., Lake, L., Kabir, C.: A state-of-the-art literature review on capacitance resistance models for reservoir characterization and performance forecasting. Energ. **11**(12), 3368 (2018). https://doi.org/10.3390/en11123368
14. Alghamdi, A., Hiba, M., Aly, M., Awotunde, A.: A critical review of capacitance-resistance models. In: SPE Russian Petroleum Technology Conference, Virtual (2021). https://doi.org/10.2118/206555-MS
15. Dobbyn, A., Marsh, M.: Material balance: A powerful tool for understanding the early performance of the Schiehallion field. In: SPE Offshore Europe Oil and Gas Exhibition and Conference, Aberdeen, United Kingdom, September (2001). https://doi.org/10.2118/71819-MS
16. Govan, A., Primmer, T., Douglas, C., Moodie, N., Davies, M., Nieuwland, F.: Reservoir management in a deepwater subsea field–the Schiehallion experience. SPE Reservoir Evalaluation & Engineering **9**(04), 382–390 (2006). https://doi.org/10.2118/96610-PA
17. Williams, G.J.J., Mansfield, M., MacDonald, D.G., Bush, M.D.: Topdown reservoir modelling. In: SPE Annual Technical Conference and Exhibition, Houston, Texas, September 2004 (2004). https://doi.org/10.2118/89974-MS
18. Zhao, H., Kang, Z., Zhang, X., Sun, H., Cao, L., Reynolds, A.C.: INSIM: A data-driven model for history matching and prediction for waterflooding monitoring and management with a field application. In: SPE Reservoir Simulation Symposium, Houston, Texas, USA, February 2015 (2015). https://doi.org/10.2118/173213-MS
19. Guo, Z., Reynolds, A.C., Zhao, H.: A physics-based data-driven model for history matching, prediction, and characterization

of waterflooding performance. SPE J. **23**(02), 367–395 (2017). https://doi.org/10.2118/182660-PA
20. Guo, Z., Reynolds, A.C., Zhao, H.: Waterflooding optimization with the INSIM-FT data-driven model. Comput. Geosci. **22**(3), 745–761 (2018). https://doi.org/10.1007/s10596-018-9723-y
21. Guo, Z., Reynolds, A.C.: INSIM-FT in three-dimensions with gravity. J. Comput. Phys. **380**, 143–169 (2019). https://doi.org/10.1016/j.jcp.2018.12.016
22. Li, Y., Onur, M.: INSIM-BHP: A physics-based data-driven reservoir model for history matching and forecasting with bottomhole pressure and production rate data under waterflooding. J. Comput. Phys. **473**, 111714 (2023). https://doi.org/10.1016/j.jcp.2022.111714
23. Lerlertpakdee, P., Jafarpour, B., Gildin, E.: Efficient production optimization with flow-network models. SPE J. **19**(06), 1083–1095 (2014). https://doi.org/10.2118/170241-PA
24. Lutidze, G.: StellNet – physics-based data-driven general model for closedloop reservoir management. PhD thesis, The University of Tulsa (2018)
25. Ren, G., He, J., Wang, Z., Younis, R.M., Wen, X.-H.: Implementation of physics-based data-driven models with a commercial simulator. In: SPE Reservoir Simulation Conference, Galveston, Texas, USA, April (2019). https://doi.org/10.2118/193855-MS
26. Behm, E., Asimi, M.A., Maskari, S.A., Juna, W., Klie, H., Le, D., Lutidze, G., Rastegar, R., Reynolds, A., Tathed, V., Younis, R., Zhang, Y.: Middle East steamflood field optimization demonstration project. In: Abu Dhabi International Petroleum Exhibition & Conference, Abu Dhabi, UAE, November 2019 (2019). https://doi.org/10.2118/197751-MS
27. Kiærr, A., Lødøen, O.P., Bruin, W.D., Barros, E., Leeuwenburgh, O.: Evaluation of a data-driven flow network model (FlowNet) for reservoir prediction and optimization. In: ECMOR XVII – 17th European Conference on the Mathematics of Oil Recovery (2020). https://doi.org/10.3997/2214-4609.202035099
28. Asimi, M.A., Qasabi, N.A., Le, D., Zhang, Y., Zhu, D., Balushi, K.A.: Expansion of data analytics for optimizing steamflood in Mukhaizna heavy oil field. In: Abu Dhabi International Petroleum Exhibition & Conference, Abu Dhabi, UAE, November (2021). https://doi.org/10.2118/207680-MS
29. Wang, Z., He, J., Milliken, W.J., Wen, X.-H.: Fast history matching and optimization using a novel physics-based data-driven model: an application to a diatomite reservoir. SPE J. **26**(06), 4089–4108 (2021). https://doi.org/10.2118/200772-PA
30. Mamonov, A., Couet, B., Bailey, W.J., Prange, M., Djikpesse, H.A., Druskin, V.: Optimal grid coarsening: A fast proxy for large reservoir simulations. In: SPE Reservoir Characterisation and Simulation Conference and Exhibition, Abu Dhabi, UAE, October 2007 (2007). https://doi.org/10.2118/111378-MS
31. Sankaran, S., Sun, W.: A flow network model based on time of flight for reservoir management. In: Abu DhabiInternational Petroleum Exhibition and Conference (2020). https://doi.org/10.2118/203390-MS
32. Guo, Z., Sankaran, S., Sun, W.: Reservoir modeling, history matching,and characterization with a reservoir graph network model. SPE Reserv. Eval. Eng. 1–13 (2023). https://doi.org/10.2118/209337-PA
33. Iino, A., Jung, H.Y., Onishi, T., Datta-Gupta, A.: Rapid simulation accounting for well interference in unconventional reservoirs using fast marching method. In: SPE/AAPG/SEG Unconv. Res. Technol. Conf. (2020). https://doi.org/10.15530/urtec-2020-2468
34. Lie, K.A.: An Introduction to Reservoir Simulation Using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST). Cambridge University Press, Cambridge, United Kingdom (2019). https://doi.org/10.1017/9781108591416
35. Emerick, A., Reynolds, A.: Ensemble smoother with multiple data assimilation. Comput. Geosci. **55**, 3–15 (2013). https://doi.org/10.1016/j.cageo.2012.03.011
36. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, New York, NY, USA (2006)
37. Oliver, D.S., Reynolds, A.C., Liu, N.: Inverse Theory for Petroleum Reservoir Characterization and History Matching. Cambridge University Press, Cambridge, United Kingdom (2008). https://doi.org/10.1017/CBO9780511535642
38. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing **16**(5), 1190–1208 (1995). https://doi.org/10.1137/0916069
39. Jansen, J.D., Fonseca, R.M., Kahrobaei, S., Siraj, M.M., Essen, G.M.V., den Hof, P.M.J.V.: The egg model - a geological ensemble for reservoir simulation. Geosci. Data J. **1**(2), 192–195 (2014). https://doi.org/10.1002/gdj3.21
40. MRST: The MATLAB Reservoir Simulation Toolbox (MRST), version 2022a. SINTEF, Oslo, Norway. SINTEF Digital (2022). https://doi.org/10.5281/zenodo.6390187.mrst.no
41. Lie, K.A., Møyner, O. (eds.): Advanced Modeling with the MATLAB Reservoir Simulation Toolbox. Cambridge University Press, Cambridge, United Kingdom (2021). https://doi.org/10.1017/9781009019781
42. Lorentzen, R.J., Luo, X., Bhakta, T., Valestrand, R.: History matching the full Norne field model using seismic and production data. SPE J. **24**(4), 1452–1467 (2019). https://doi.org/10.2118/194205-PA
43. Natvig, J.R., Lie, K.A.: Fast computation of multiphase flow in porous media by implicit discontinuous Galerkin schemes with optimal ordering of elements. J. Comput. Phys. **227**(24), 10108–10124 (2008). https://doi.org/10.1016/j.jcp.2008.08.024