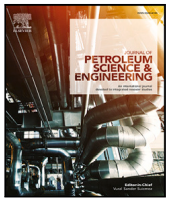
Journal of Petroleum Science and Engineering 221 (2023) 111266

Contents lists available at ScienceDirect

Journal of Petroleum Science and Engineering

journal homepage: www.elsevier.com/locate/petrol



Comparison of two different types of reduced graph-based reservoir models: Interwell networks (GPSNet) versus aggregated coarse-grid networks (CGNet)

Knut-Andreas Lie *, Stein Krogstad

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

ARTICLE INFO

Keywords:
Graph-based models
Reduced-order models
Network models
Grid coarsening
Adjoints
Automatic differentiation
L-BFGS-B

ABSTRACT

Computerized solutions for field management optimization often require reduced-order models to be computationally tractable. The purpose of this paper is to compare two different graph-based approaches for building such models. The first approach represents the reservoir as a graph of 1D numerical flow models that each connects an injector to a producer. One thus builds a network in which the topology is primarily determined by "well nodes" to which "non-well nodes" can be connected if need be. The second approach aims at building richer models so that the connectivity graph mimics the intercell connections in a conventional, coarse 3D grid model. One thus builds a network with topology defined by a mesh-like placement of "non-well nodes", to which wells can be subsequently connected. The two approaches thus can be seen as graph-based analogues of traditional streamline and finite-volume simulation models. Both model types can be trained to match well responses obtained from underlying fine-scale simulations using standard misfit minimization methods; herein we rely on adjoint-based gradient optimization. Our comparisons show that graph models having a connectivity graph that mimics the intercell connectivity in coarse 3D models can represent a wider range of fluid connections and are generally more robust and easier to train than graph models built upon 1D subgridded interwell connections between injectors and producers only.

1. Introduction

The cost of evaluating a forward model is in many cases a limiting factor for model-based uncertainty quantification and field management optimization, in particular when the model evaluation consists of a full simulation run with a traditional reservoir simulator. Various deep-learning methods have been proposed as to build models that are computationally inexpensive to evaluate, see., e.g., Costa et al., 2014, Kim et al., 2020, Tang et al., 2020, Kim and Durlofsky, 2021, Liu and Reynolds, 2021, Zhong et al., 2021, and references therein. Such models can give reliable forecasts of production responses from a given set of input parameters and control strategies. However, a limitation of purely data-driven models is that they cannot guarantee that the predicted results represent physically consistent and meaningful states. This can seriously affect the ability to generalize outside of the range of training data. Models may, for instance, be able to predict the spatial distribution of phase saturations (Maucec and Jalali, 2022; Zhang et al., 2021), without having the ability to provide reliable predictions of consistent well responses. So-called physics-informed methods try to circumvent these problems by incorporating loss functions to penalize predictions that violate fundamental flow physics or by trying to learn the underlying residual equations directly (Navrátil et al., 2019; Jin et al., 2020; Fraces et al., 2020; Wang et al., 2021a; Rodriguez-Torrado et al., 2021). A main drawback with all these machine-learning methods is that they require much observed or simulated data and that the training can be computationally expensive.

An alternative approach is to start more directly from the flow physics and develop simplified/reduced models that can be calibrated to match observed or simulated data. We will, in particular, study the family of interwell numerical simulation models, which represent the reservoir as a flow network connecting injectors and producers. The INSIM versions of this idea use analytical and semi-analytical methods to evolve pressures and saturations for each interwell connection (Zhao et al., 2015; Guo et al., 2018a,b; Guo and Reynolds, 2019) and have proved to be useful in history matching waterflooding scenarios. On the other hand, the INSIM methods require special simulation tools and do not easily extend beyond two-phase flow. The method can be

* Corresponding author.
E-mail addresses: Knut-Andreas.Lie@sintef.no (K.A. Lie), stein.krogstad@sintef.no (S. Krogstad).
URL: http://folk.ntnu.no/andreas (K.A. Lie).

https://doi.org/10.1016/j.petrol.2022.111266
Received 20 December 2021; Received in revised form 16 November 2022; Accepted 16 November 2022
Available online 14 December 2022

generalized to more complex flow physics if one instead uses a standard finite-volume method to solve for the flow within each interwell connection (Lerlertpakdee et al., 2014; Lutidze, 2018; Ren et al., 2019; Kiærr et al., 2020; Wang et al., 2021b; Leeuwenburgh et al., 2022). Among the many possible configurations, we have chosen GPSNet (Ren et al., 2019), which allows multiple flow paths between each injector and producer, but assumes that all connections have the same number of uniform grid cells so that the flow paths can be mapped onto a rectangular grid for simple inclusion in a standard simulator. Another related approach is RGNet (Sankaran and Sun, 2020), which relies on the diffusive time-of-flight to identify drainage regions that are coupled using inter-partition transmissibilities to form a network graph.

If we think of the INSIM/GPSNet family as the graph-model analogue to streamtubes/streamlines, it is easy to also envision graph models having a richer topology that mimics that of a 3D coarse finite-volume grid. In lack of a better name, we will refer to this model as CGNet (coarse-grid network). Setting up a representative 3D volumetric model can be challenging if the reservoir geometry and geology are largely unknown, but preliminary experiments indicate that good matches can be obtained with simple Cartesian grids delimited by the presumed map outline of the reservoir as long as one can provide ballpark estimates of the petrophysical properties (Lie and Krogstad, 2022).

When used for model reduction, the training data for the reduced graph models will come from simulations of an underlying fine-scale model. The training thus consists of tuning the intercell transmissibilities, the pore volumes, and possibly also the parameters of the fluid model to minimize the misfit in the predicted outcomes from the fine-scale and the graph models. One may argue that this setting is no different from traditional history matching of a coarse model (Morteza et al., 2019). In our opinion, however, there are important differences. Firstly, we emphasize that our aim is not to calibrate parameters like in a conventional coarse-grid model, which usually attempt to preserve local characteristics of the reservoir's petrophysics. With CGNet, we instead seek to train physics-based data-driven models that reproduce the same flow rates and pressure as observed in each well, while still adhering to basic laws of flow physics. All petrophysical parameters should thus only be viewed as *tunable* algebraic coefficients, as no attempt will be made to preserve the petrophysical quantities and the geological realism from the prior model, which has traditionally been a major challenge in history matching. (Mamonov et al., 2007 proposed a method that is similar in spirit in the sense that it seeks to optimize the coarse mesh to preserve aggregated objective values used in production optimization.) Secondly, the fine-scale simulations used for training are not set up as representative production histories, but are instead driven by randomly perturbed well controls that attempt to excite a wider span of physical states within the reservoir.

Both model types, CGNet and GPSNet, are realized inside a fully differentiable, open-source reservoir simulator (Lie, 2019) that implements standard discretizations, solvers, and flow models but relies on automatic differentiation for the computation of gradients and Jacobians. This means that the graph models are fully differentiable with respect to all the tunable parameters, so that their gradients can easily be computed by solving adjoint equations. The training can thus be formulated as a misfit minimization problem and solved by a gradient-based quasi-Newton method in complete analogy to the backpropagation methods that power the training of feed-forward neural networks in machine learning. Advantages and disadvantages of using such a method instead of an ensemble smoother with multiple data assimilation (Emerick and Reynolds, 2013), which is very popular within history matching, are discussed in Borregales et al. (2021).

The main purpose of this paper is to compare the ability of the two types of graph models to match simulations from a fine-scale method and compare the predictive ability of the resulting models on production scenarios that are different from those used in training. In Lie and Krogstad (2022), we discuss how one can use similar ideas to produce purely data-driven models that reproduce observed well responses.

2. Flow equations and discretization

The reduced graph-based modeling methods can in principle be applied to both compositional and black-oil type models, but to keep the equation system as simple as possible, we only consider two-phase, compressible, immiscible flow without capillary pressure,

$$\frac{\partial}{\partial t}\left(\phi\rho_\alpha S_\alpha\right) + \nabla\cdot\left(\rho_\alpha\vec{v}_\alpha\right) = q_\alpha, \qquad \vec{v}_\alpha = -\mathbf{K}\lambda_\alpha\left(\nabla p - \rho_\alpha g\nabla z\right),$$
$$q_\alpha = \lambda_\alpha^{wb}\mathrm{J}\left(p^{wb} - p\right), \quad \alpha\in\{w,o\}. \tag{1}$$

In the first equation, which represents conservation of mass, ϕ is porosity and ρ_α is the phase density, S_α is saturation, $\vec{v}_\alpha$ is phase flux, and q_α is the volumetric source of phase α. The second equation, Darcy's law, defines $\vec{v}_\alpha$ in terms of the absolute permeability $\mathbf{K}$, the common fluid pressure p, the gravity constant g, phase density ρ_α, and the phase mobility $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$, where $k_{r\alpha}(S_\alpha)$ is the relative permeability and μ_α is the phase viscosity. The third equation relates the volumetric source q_α to the phase mobility λ_α^{wb} at the well, the productivity index J, and the difference between the wellbore pressure p^{wb} and the reservoir pressure p. If we also specify that the fluids fill pore space completely, the first two equations can be reduced to a system of two equations for the fluid pressure p and the 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.

The system of flow Eqs. (1) will be discretized on three types of grids: (i) on 1D stacks of uniform cells representing a single interwell connection for the GPSNet models; (ii) on the fine-scale 3D reservoir grid, typically a corner-point grid or some other form of stratigraphic grid, which we for generality can assume has unstructured topology and polytopal cell geometries; and (iii) on a coarse partition (or a coarse approximation) of the 3D fine-scale grid for the CGNet models. For a uniform treatment, we follow (Lie, 2019) and introduce `Div` and `Grad` as the discrete numerical analogues of the standard divergence and gradient operators. These operators are linear and can be represented as sparse matrices determined entirely from the grid topology. We now let $\boldsymbol{S}$ be a vector of unknown saturations, one per cell, and $\boldsymbol{v}$ the vector of fluxes across all cell interfaces; vectors are defined analogously for all other physical quantities. If we understand the product of two vector quantities as the vector containing the product of the elements, we can write the discrete flow equations on the form[1]

$$\frac{\boldsymbol{\varphi}^c}{\Delta t}\left[(\boldsymbol{S}_\alpha\boldsymbol{\rho}_\alpha)^{t+\Delta t} - (\boldsymbol{S}_\alpha\boldsymbol{\rho}_\alpha)^{t}\right] + \mathtt{Div}(\boldsymbol{\rho}_\alpha\boldsymbol{v}_\alpha{}^{t+\Delta t}) = \boldsymbol{q}_\alpha,$$
$$\boldsymbol{v}_\alpha = -\boldsymbol{T}\lambda_\alpha\mathtt{Grad}\left(\boldsymbol{p} - g\boldsymbol{\rho}_\alpha\boldsymbol{z}\right), \quad \boldsymbol{q}_\alpha = \lambda_\alpha^{wb}\mathbf{J}(\boldsymbol{p}^{wb} - \boldsymbol{p}). \tag{2}$$

In the fine-scale model, the vector $\boldsymbol{\varphi}$ contains the pore volumes of the cells whereas the transmissibilities $\boldsymbol{T}$ and the productivity indices $\mathbf{J}$ account for permeability and geometric factors affecting the flow. In the graph based models, these three vectors are merely tunable factors that will be calibrated during the training of the models.

3. Reduced graph models

As explained in the introduction, we consider two types of reduced graph models which we can see as the graph model analogue of streamtubes and a conventional finite-volume model, respectively. This section will review their setup briefly and also explain how the models are trained by minimizing the misfit with the training data.

[1] The equations are stated on a conceptual form. The correct equations also contain rock compressibility and averaging operators, including conditional phase upwinding, that are necessary to evaluate mobilities and densities at cell and wellbore interfaces. These do not change the role of the parameters $\boldsymbol{\varphi}$, $\boldsymbol{T}$, and $\boldsymbol{J}$ and have thus been dropped to keep the notation as simple as possible. Full details can be found in Lie (2019).

3.1. Interwell network model (GPSNet type)

In interwell network models, the key idea is to represent the reservoir in terms of a limited set of flow paths that connect injectors and producers. The fluid movement inside each flow path is calculated by solving a 1D system of flow equations, much in the same way as in a streamtube method, except that all the 1D connections are usually coupled through the well nodes, so that the resulting global system of connected flow paths must be solved jointly. We consider the special case of a so-called GPSNet (Ren et al., 2019) interwell model, which in the basic setup has a single flow path for each connected pair of injectors and producers; GPSNet can also be set up with different interwell network topologies and multiple interwell connections per well pair, as we will discuss later. The flow paths terminate in well cells, which offer the possibility for fluid communication among the flow paths without fluids having to go through the wells. Fig. 1 illustrates the setup of a GPSNet model for a simulation case built on top of the reservoir geometry and petrophysics from a public model of the Norne oil field.[2]

There are several ways to determine the set of connected well pairs. The simplest is to just connect all injectors to all producers, like in Fig. 1. With six injectors and five producers, this gives thirty interwell connections. All these well pairs may not be communicating actively in reality, but the advantage of including all possibilities in the model is that it gives us more flexibility in the training of the reduced model. On the other hand, this naive approach can lead to a combinatorial explosion for cases with many injectors and producers or poor model performance because of overparameterization. In INSIM (Zhao et al., 2015) and related approaches like FlowNet (Kiærr et al., 2020; Leeuwenburgh et al., 2022), possible connections are limited based on the lateral proximity of wells (and pseudowells added for increased accuracy). This tacitly assumes that wells that are far apart in physical space are not connected by long-range high-permeability flow channels. To circumvent this potential problem, the RGNet approach, proposed by Sankaran and Sun (2020) uses diffusive time-of-flight instead of Euclidean distance to measure proximity.

In previous research (Borregales et al., 2020), we have shown that if a GPSNet is used as a reduced model for an underlying fine-scale reservoir model, one can minimize the number of interwell connections by initializing according to well allocation factors determined by flow diagnostics (Møyner et al., 2015) of the original simulation model. Minimizing the number of connections based on flow physics unfortunately tends to give too restrictive models that are more difficult to match. Herein, we thus consider the straightforward all-injectors-to-all-producers connection seen in the lower-right plot of Fig. 1 as the basic setup, but will compare it to other types of setups, including all-to-all topologies that also incorporate injector–injector and producer–producer connections.

Once the necessary interwell connections among well and pseudowell nodes have been identified, each connection is assumed to be a volumetric flow tube and subdivided into a finite set of grid cells so that we can use (2) to compute the corresponding fluid flow (Lerlertpakdee et al., 2014). To be able to use the resulting models within standard commercial simulators, Ren et al. (2019) proposed to map the flow paths onto a standard 2D volumetric grid. In their GPSNet, the flow tubes are assumed to be rectangular cuboids, and their discretized versions are mapped and stacked into a rectilinear grid so that each interwell connection becomes a horizontal row of 3D cells, as you can see to the upper left in Fig. 1. Seen from the reservoir simulator, the resulting collection of flow tubes will appear as any conventional 2D$^+$ simulation model, except that the transmissibilities have been manipulated so that only horizontal flow is allowed by setting values corresponding to vertical connections to zero. Each injector or producer is completed in a single well cell and multiple connections are managed through the mechanism of non-neighboring connections, which is present in most standard simulators. Notice that the choice of having the same number of grid cells for each connection was made to simplify the setup and subsequent bookkeeping and is not an inherent limitation of the method.

During training, we match the well indices of individual wells and the *overall* pore volume and transmissibility of each flow tube and not the values associated with individual cells and cell faces in the grid. That is, if n_c denotes the number of interwell connections among the n_w wells, the training parameters will be $\{T_k, \varphi_k\}$ for $k = 1, \dots, n_c$ and $\{\mathrm{J}_k\}$ for $k = 1, \dots, n_w$.

3.2. Graph model with 3D interconnection (CGNet)

A key limitation with the GPSNet type models just described is that they do not include any interaction among the interwell flow paths, excepts through the well cells at each end. In the INSIM family, one can increase the possible flow paths by introducing imaginary "ghost wells" throughout the reservoir (Guo et al., 2018b). These wells do not inject or extract fluids but only redistribute the flow from incoming flow paths to outgoing flow paths and hence serve to enrich the modeling of interconnectivity inside the reservoir. RGNet models (Sankaran and Sun, 2020) achieve much of the same effect through the use of so-called inter-partition transmissibilities (Iino et al., 2020) that turn the interconnection among the dynamic drainage volumes of individual wells into a general reservoir network.

We propose a different idea for creating a richer graph topology than in the interwell network models by specifying a graph that mimics the intercell connectivity of a 3D finite-volume method. In principle, each connection in this graph could be subgridded as in the interwell models, and similar graph topologies could be constructed using the ghost well idea from INSIM or extended versions of the general graph connection of RGNets. However, since we already have an underlying fine-scale model, whose predictions we seek to reproduce as accurately as possible, we can create the graph topology by aggressive coarsening of the underlying 3D simulation model. The coarsening is achieved by grouping cells from the underlying fine-scale model into effective grid blocks in the coarse model. These aggregated blocks are simple to represent using an integer vector, and the approach has the advantage that it preserves the exact geometry of the underlying fine-scale model. The aggregation can be performed in many different ways, including use of systematic graph partitioning methods (Karypis and Kumar, 1998), ad-hoc methods as discussed in Lie et al. (2017) and Chapter 14 of Lie (2019), or just plain "cookie-cutter" partitions.

Fig. 2 illustrates the creation of such a CGNet model for the Norne field, using an index-space partition of the corner-point grid. The resulting model has 36 nodes, each with a tunable pore volume, 58 edges with associated tunable transmissibilities, and 14 well–reservoir connections, each with a tunable well index. This graph model is distinctively different from the GPSNet shown in Fig. 1, which has 11 nodes with associated and tuneable well indices, and 30 edges with two adjustable parameters (pore volume and transmissibility) per edge. The CGNet model has 52% more parameters than the GPSNet model, but maps onto a grid that only has 12% as many cells as the 30×10 grid in the GPSNet model. It is also instructive to compare the interwell communication in the two models. In GPSNet, two wells are either explicitly connected or their communication must go through a subset of the other wells. This limits the number of parameters that can effect the interwell communication. In CGNet, on the other hand, there are many possible paths through the network of non-well nodes that can connect individual well pairs. This means that the communication

[2] The full model is published as an open data set on Github () by the Open Porous Media (OPM) initiative. We use a derived version thereof featured in the example suite of the MATLAB Reservoir Simulation Toolbox (MRST) (Lie, 2019).

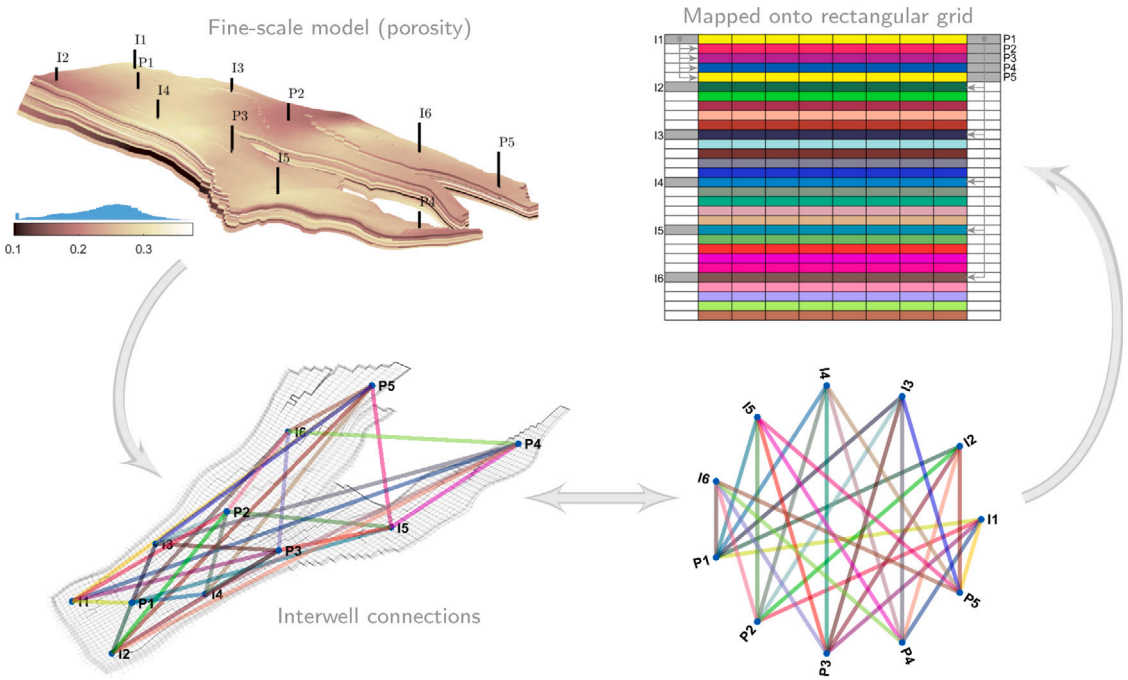


Fig. 1. GPSNet graph model for the Norne field with six injectors and five producers. The interwell connection is simple: each injector is connected to all producers and vice versa. Each interwell connection is discretized into ten uniform cells and mapped onto a uniform rectangular grid. In the upper-right plot, well cells completed by a well are colored dark gray. The white cells are not active in the simulation grid, and to avoid cluttering, arrows indicate the non-neighboring connections have only been included for the well cells connected to I1 and P1.

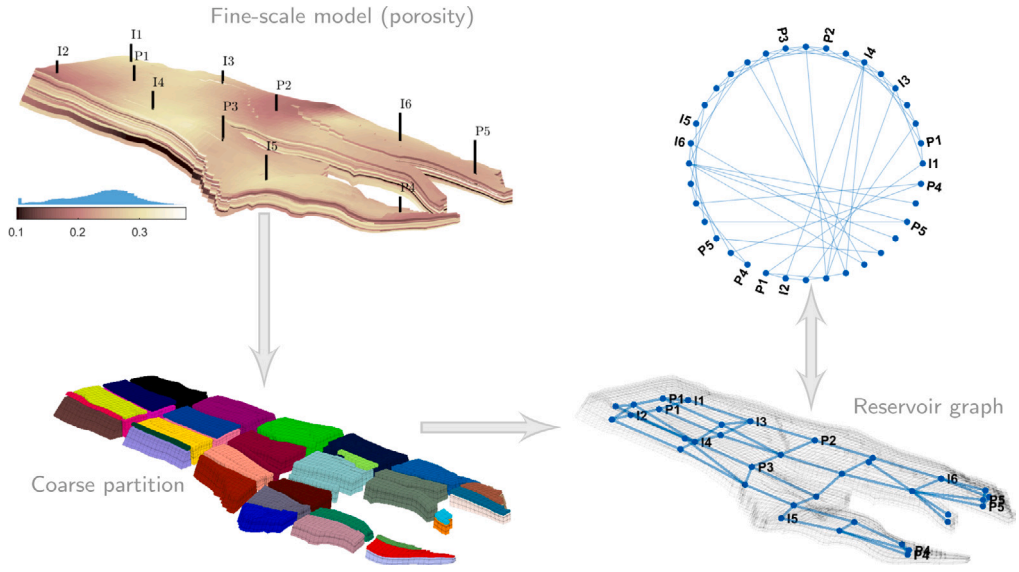


Fig. 2. CGNet graph model built from a coarse partition of the Norne field. The coarse partition is generated in two steps: first a regular $5 \times 6 \times 1$ partition in index space, followed by a split of disconnected blocks, giving a total of 36 coarse blocks. The graph topology is then formed by connecting blocks that share one or more fine-cell faces, i.e., following the transmissibility graph of two-point, coarse-grid, finite-volume scheme.

between two wells is affected by more parameters, so that CGNet will have more flexibility to detect long-range interwell communication.

The graph topology in a CGNet model does not depend on the well placement. This means that a trained model in principle could support conversion of producers to injectors and vice versa. This is generally not possible with interwell graph models since the whole network of injector–producer pairs would change once one or more wells are converted. The exception may be if one uses an all-to-all network topology. For CGNet one can also imagine that the parameters corresponding to transmissibility and pore volume are trained with multiple fine-scale simulations having different placement and number of wells, but it remains to be more thoroughly investigated whether the resulting models will be sufficiently accurate. This *may* also be possible for GPSNet, as long as one can derive a network topology that is representative for all cases, but is conceptually more difficult.

The training process is initialized by upscaling pore volumes, transmissibilities, and well indices from the underlying fine-scale models. For pore volumes and other additive properties, it is sufficient to use standard volumetric averaging. Herein, we upscale the transmissibilities using a well-specific, global approach (Krogstad et al., 2016) that uses a least-squares method to compute the transmissibilities based on $n_w - 1$ independent incompressible flow solutions for n_w wells. The

particular upscaling method is not essential and can easily be replaced by any standard method if the well-specific method is not available or considered too computationally expensive (e.g., for cases with many wells). Initial values for the well indices are computed by averaging the fine-scale well indices over each perforated coarse block.

3.3. Model training

To train our network models to match the output from a full simulation, we seek to determine a set of parameters $\boldsymbol{\theta}$ that minimizes a weighted sum of squares

$$M(\boldsymbol{\theta}) = \boldsymbol{m}(\boldsymbol{\theta})^T \boldsymbol{m}(\boldsymbol{\theta}), \tag{3}$$

where $\boldsymbol{m}$ holds the scaled misfits $m_i = (y_i(\boldsymbol{\theta}) - y_{i,ref})/w_i$ in the quantities of interest. y. Here, $y_{i,ref}$ is the ith reference data point and $y_i(\boldsymbol{\theta})$ is the corresponding output data point computed by the network model. If $y_{i,ref}$ were measurements, the weights w_i would typically be set depending on the measurement error, whereas in our case, in which there is no measurement error, we simply use weights to homogenize the magnitudes of the data points or (de-)emphasize individual data points. It is also common to add a regularization term of the form $w_\theta \|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\|_2^2$ to (3). Here, $\hat{\boldsymbol{\theta}}$ could hold, for instance, initial guesses or the means over parameter groups. Herein, we do not minimize (3) directly but instead work with the Lagrange function ($\boldsymbol{G}$ represents the residual Eqs. (2)):

$$L_\lambda = M(\boldsymbol{\theta}) + \lambda^T \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{\theta}). \tag{4}$$

For the numerical examples presented in the next section, we impose lower and upper bounds on each parameter, and each parameter is scaled to the unit interval. The Lagrange function $L_\lambda(\boldsymbol{\theta})$ is minimized using a variant of the *Limited-memory Broyden—Fletcher—Goldfarb—Shanno algorithm with bound constraints* (L-BFGS-B) (Byrd et al., 1995) and use of the adjoint equations of (2) and automatic differentiation to compute the necessary gradients. The Hessian approximations are projected to the active parameter set to obtain search directions, and a cubic line-search with acceptance based on strong Wolf conditions is used to obtain step lengths.

We typically observe that roughly 90% of the iteration steps are accepted without the need for further line search. On the other hand, after an initial phase that typically reduces the misfit 2–3 orders of magnitude, we observe rather slow convergence for the model training using this approach. Zhang and Reynolds (2002) studied the convergence properties of the L-BFGS method applied to history matching and reported results for different scaling approaches. We experimented with some of these but did not observe consistent improvements for the examples considered here. This could be because our objective is different from the one used by Zhang and Reynolds (2002) and because we are considering a more diverse set of parameters. Accordingly, for the results presented in the following, we use *standard* scaling of the L-BFGS method (see e.g., Nocedal and Wright, 2006).

4. Numerical examples

This section compares CGNet and GPSNet for their ability to match training data simulated from an underlying fine-scale model and then subsequently predict the correct behavior of the same model subject to different well controls. All numerical results are obtained using solvers and workflow tools implemented in the open-source MATLAB Reservoir Simulation Toolbox (MRST, 2021; Lie, 2019; Lie and Møyner, 2021). The implementations of CGNet and GPSNet used herein, as well as tutorial scripts demonstrating their use, have been publicly available as part of the `networks-module` of MRST since the 2022a release.

4.1. Norne field model

In the first example, we investigate how well CGNet and GPSNet work as proxies for the field production from a reservoir model with realistic reservoir geometry and heterogeneity. To this end, we use the reservoir geometry taken from a public simulation model of the Norne oil and gas field from the Norwegian Sea. The reservoir geometry is described by a $46 \times 112 \times 22$ corner-point grid with 44 915 active cells and has several faults, displaced layering, pinched cells, internal gaps, non-neighboring connections, etc. Petrophysical parameters are generated as described by Lorentzen et al. (2019) using their code but without inclusion of some of the random parameters that *perturb* connectivity across faults. The static model and examples of corresponding network topologies for CGNet and GPSNet have already been shown in Figs. 1 and 2.

The Norne field is produced with water drive as the main recovery strategy, but with gas injection to prevent rapid depletion of the gas cap. The corresponding simulation model uses a 3-phase black-oil model with dissolved gas and vaporized oil and involves features such as end-point scaling of relative permeability and capillary pressure, pressure-dependent porosity and transmissibility, and history-matched, time-varying well controls. Altogether, this makes up for a quite complex case, where effects are difficult to disentangle. Herein, our main purpose is to compare and illustrate the behavior of CGNet and GPSNet. We will therefore make several simplifications to make the discussion more clean-cut.

First of all, the reservoir is assumed to be initially completely filled with oil. Our motivation for this simplification is that it is straightforward to specify a stratified fluid distribution in CGNet but more cumbersome with GPSNet; we will come back to a case with fluid stratification in Section 4.2. Second, we simplify the flow physics to a two-phase, waterflooding scenario described by a simple dead-oil model with constant formation-volume factors, quadratic relative permeabilities with zero residual saturations, and constant viscosities with an oil–water viscosity ratio of 5:1, giving an unstable displacement. Likewise, instead of using the true wells and well trajectories, we assume that the reservoir is produced from five vertical producers operating at constant bottom-hole pressure, supported by six vertical water injectors operating at constant rate. The wells are set up to form one dominating five spot and two weaker line drives, which together ensure that fluids flow throughout the whole reservoir volume. (We henceforth refer to this setup as Case 0.) Our motivation for these simplifications is to create a setup with which we can investigate the effect of different network topologies and how robust the calibration process and the calibrated networks are to significant changes in major flow directions inside the reservoir, without also having to consider calibration of initial saturations and fluid parameters determining relative permeabilities, capillary pressures, and PVT behavior. (For completeness, we mention that the full Norne case is available as one of the test cases for FlowNet (Leeuwenburgh et al., 2022) at https://github.com/equinor/flownet-testdata.

The graph models are trained using a fine-scale simulation with oscillatory well controls, created by adding a 5% random variation around the prescribed bottom-hole pressure controls and a 25% variation around the prescribed water injection rates; see Fig. 3. The purpose of the random input controls is to excite the relative interwell dependence inside the reservoir and thereby create training data that span a wider set of possible reservoir states. (Such a perturbation procedure may not be necessary if the simulation schedule contains enough variation in the well controls to excite a wide spectrum of states or if the network models are calibrated to observed well responses in a pure data-driven setting; see, e.g., Lie and Krogstad, 2022.) We thus have two data sets covering the same time horizon: a simulation with oscillatory well controls that will be used for training and a simulation with constant well controls that will be used to measure prediction accuracy. Notice that this setting is different from conventional history

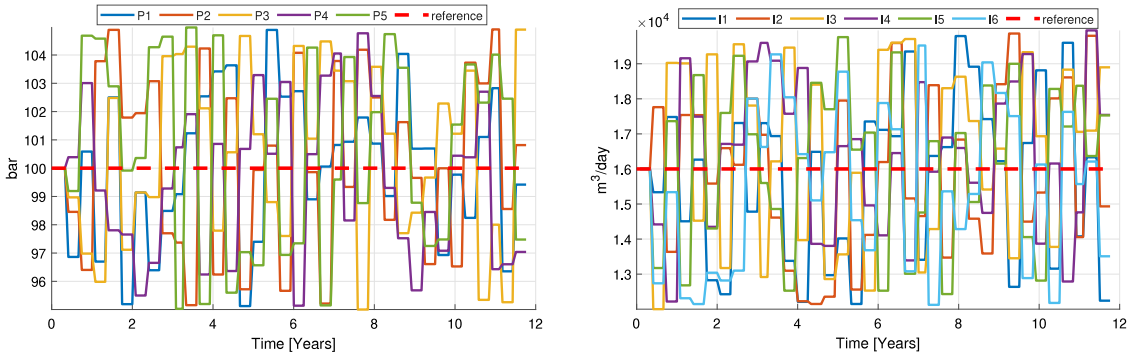


Fig. 3. Stochastic well controls used to create the training data for the Norne field model: bottom-hole pressures for the five producers (left) and water rates for the six injectors (right). The dashed lines show the controls for the reference simulation used for prediction.

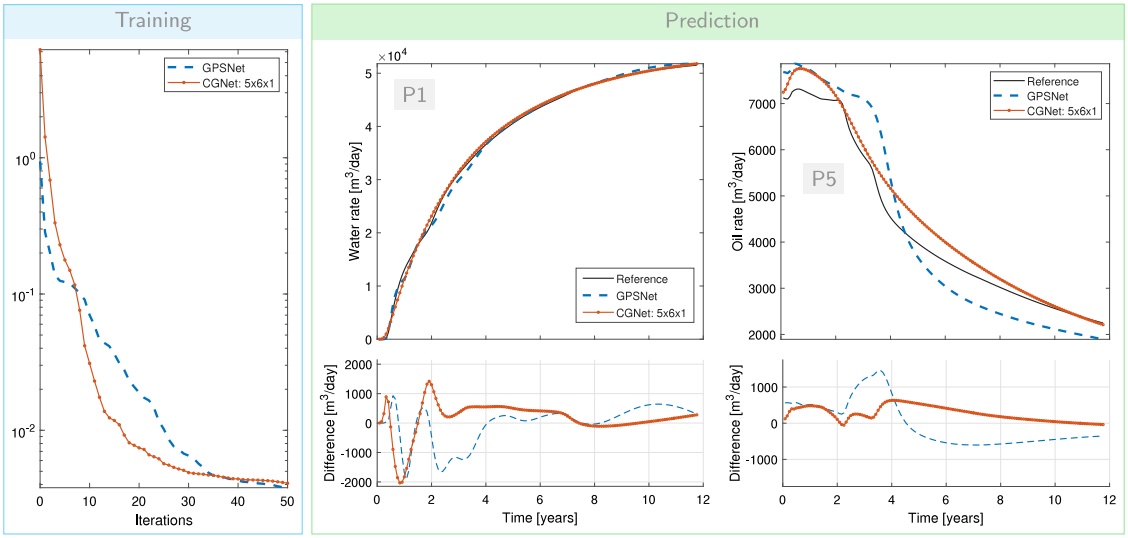


Fig. 4. Training accuracy and prediction quality for GPSNet and CGNet on Norne Case 0. The left plot shows the convergence of the misfit function for the training data. The middle and right columns show deviations in predicted water and oil rates of producers P1 and P5, respectively.

matching, where one usually has a training period followed by a prediction period.

The training is set up as a misfit minimization problem between the simulation output from the fine-scale simulation and the predicted outcomes from the graph models, with mismatches in bottom-hole pressure scaled by 500 bar and the oil and water well rates scaled by $2 \cdot 10^4$ and 10^4 m^3/day, respectively. With 144 time steps in each simulation and 11 wells, the number of data points is thus 4752. The gradients required by the L-BFGS-B algorithm are computed by adjoint equations and automatic differentiation. All parameters are scaled to the unit interval. For the pore volumes, we use linear scaling and constraints $[0.001, 4]$ relative to the initial guess, whereas logarithmic scaling and relative limits of $[10^{-3}, 10^2]$ are used for the transmissibilities and well indices. Fluid parameters are not matched in this example.

4.1.1. *Misfit in predicted well curves*

To compare the training ability of the GPSNet and CGNet models from Figs. 1 and 2, we first train both with 50 iterations of the misfit minimization algorithm.[3] The number of calibrated parameters are 108 for CGNet and 71 for GPSNet. Fig. 4 reports the observed reduction in misfit for the training data together with the deviations in the resulting prediction in two of the production wells. GPSNet has a better initial fit to the training data than CGNet and also trains faster for the first eight iterations, but after 50 iterations, the two models give comparable misfits with respect to both training and the true data. With 36 grid cells in total, the CGNet model has very few cells between the wells and will consequently be subject to more numerical smearing and give smoother reservoir responses. It is nonetheless interesting to observe how the larger number of possible paths between injectors and producers in the network graph of the CGNet result in an equally accurate prediction of phase rates as the 300-cell GPSNet model. The slight wiggles in the water-rate curve for GPSNet are a result of breakthrough in the individual interwell connections. (A similar effect can be observed in streamline simulation, but is then less pronounced because each well pair is usually connected by multiple streamlines.)

A cursory look at the production curves in Fig. 4 may give the false impression that the match is much better for the water rate in producer P1 than for the oil rate in producer P5. Our default well configuration gives an unbalanced drainage, in which producer P1 has much higher liquid rates and experiences earlier water breakthrough than the other wells. With equal weighting of deviations in phase rates from all producers, the *relative error* will generally be larger in wells with lower rates, as is clearly seen in the oil rates for producer P5, but the *absolute* errors are of similar magnitude, if not smaller for P5. This is usually acceptable when one is primarily interested in reproducing the correct overall field production. However, one could also argue that it may be advantageous to introduce relative, well-dependent weights

[3] Complete code for setting up and calibrating CGNet and GPSNet can be found in the `norneCGNetAjoint.m` and `norneGPSNetAjoint.m` scripts of the `network-models` module of MRST from release 2022a and onward.

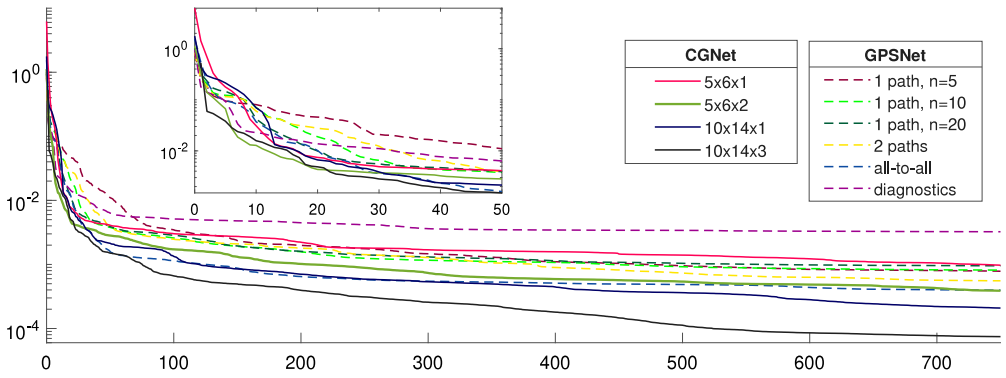


Fig. 5. Reduction in misfit during training of the CGNet and GPSNet models for Norne with training data derived as random perturbations of Case 0.

Table 1
Misfit in the prediction of Case 0 for the Norne waterflood simulation, measured as the scaled sum of the discrepancies in bottom-hole pressure and phase rates.

	GPSNet						CGNet			
specification	1 path injector–producer			2 paths	all-to-all	diagnost	5 × 6 × 1	5 × 6 × 2	10 × 14 × 1	10 × 14 × 3
# parameters	71	71	71	131	121	51	108	248	497	1479
# grid cells	150	300	600	600	550	200	36	70	166	400
50 iterations	1.07e-2	3.00e-3	3.24e-3	3.17e-3	1.01e-3	6.8ee-3	3.41e-3	2.33e-3	1.57e-3	1.17e-3
100 iterations	2.91e-3	2.21e-3	2.61e-3	2.07e-3	8.26e-4	4.69e-3	2.59e-3	1.36e-3	1.03e-3	4.24e-4
200 iterations	1.88e-3	1.69e-3	2.20e-3	1.62e-3	5.44e-4	4.56e-3	2.12e-3	1.00e-3	5.03e-4	2.67e-4
300 iterations	1.23e-3	1.37e-3	2.06e-3	1.16e-3	4.92e-4	3.48e-3	1.67e-3	6.48e-4	3.84e-4	2.24e-4
400 iterations	9.37e-4	1.26e-3	2.01e-3	9.41e-4	4.85e-4	3.48e-3	1.50e-3	4.60e-4	3.40e-4	1.85e-4
500 iterations	9.59e-4	1.15e-3	1.76e-3	8.83e-4	4.79e-4	3.45e-3	1.22e-3	4.24e-4	2.99e-4	1.58e-4
750 iterations	7.11e-4	1.02e-3	1.53e-3	6.08e-4	4.95e-4	3.45e-3	8.74e-4	2.99e-4	2.93e-4	1.24e-4

in the mismatch function that better reflect the observed rates (and pressures) to study individual wells in full detail. Notice that we could also have calibrated the end-point scaling of the relative permeability model (six parameters for a two-phase model) and the initial pressure and saturation, but this proved to not have significant positive effect in this particular case.

4.1.2. Comparison of different network configurations

The two specific networks shown in Figs. 1 and 2 are just examples of how one can set up GPSNet and CGNet models. For a more thorough comparison, we train four additional GPSNet models: two models with five and twenty grid cells per interwell connection, respectively, one having two flow paths connecting each well pair,[4] one with all-to-all communication (i.e., including producer–producer communication), and one with interwell connections determined by flow diagnostics (Borregales et al., 2020, 2021). Likewise, we trained three additional CGNet models derived from partitions having more grid blocks (5 × 6 × 2, 10 × 14 × 1, and 10 × 14 × 3). All models were trained with 750 iterations. Fig. 5 shows the convergence of the training iterations and Table 1 reports the misfits in predictions of models extracted at different stages of the training process.

Increasing the number of network nodes (i.e., coarse blocks) for CGNet improves the misfit with respect to both the training and the prediction data. It is particularly promising that using more fitting parameters does not adversely affect the misfit minimization during training. On the other hand, the convergence of the optimization method is painstakingly slow after the first 20–30 iterations, which indicates that one should look for alternatives to quasi-Newton methods that are better suited for the later stages of the optimization. (In Lie and Krogstad, 2022, we show that the classical Levenberg–Marquardt method Nocedal and Wright, 2006 may be a viable alternative.)

For GPSNet, the initial training process is significantly slower with five cells instead of ten per interwell connection in the default injector–producer setup and only catches up after 400 iterations. On the other hand, the prediction misfit is lower with five cells from iteration 300 and onward. Using twenty cells instead of ten results in larger misfits both with respect to training and prediction data. The extra grid cells are better spent if we instead configure the model with two connections (of ten cells each) for each injector–producer pair. The resulting model has 131 parameters and trains slower up to 330 iterations than the basic 71-parameter model, but the resulting predictions are nonetheless more accurate in all the extracted stages from 100 iterations and onward.

Even better results are obtained if injector–injector and producer–producer connections are added to the basic setup, giving 121 parameters and 550 cells. This may appear counter-intuitive if we view GPSNet as a collection of streamtubes, for which there would not be any flow between two injectors or two producers unless one of them has cross-flow. However, although the streamtube analogy can be useful to motivate the basic setup, the connections in GPSNet are not really streamtubes and should only be considered as edges in a general graph. Fig. 6 shows two snapshots of the water saturation for Case 0. The most distinct feature in the left plot is the water breakthrough from injector I1 to producer P1, but we also clearly see how injector I4 has found a way to connect to the producers by way of I1 (similar observations apply to injectors I2, I3, and I6). Likewise, I1 connects weakly to other wells by way of I5. These observations are more pronounced at the end of the simulation. From this, we conclude that increasing the connection among wells gives the model more freedom to adapt its dynamics to the observed data. Contrary, if we use flow diagnostics to restrict the number of connections to the ones observed in the fine-scale simulations, we deprive the model some of its freedom: The model trains fast for the first 10–20 iterations, but after this, the misfit reduction stagnates quickly, resulting in a worse overall match compared with all the other setups. (In previous research Borregales et al., 2020, we found GPSNet models configured by flow diagnostics to be effective for the much simpler Egg model. These results used fewer training iterations and are thus in agreement with what we observe herein.)

[4] In other words, there are two edges connecting the corresponding well nodes in the network graph and two rows of cells connecting the wells in the rectangular grid.

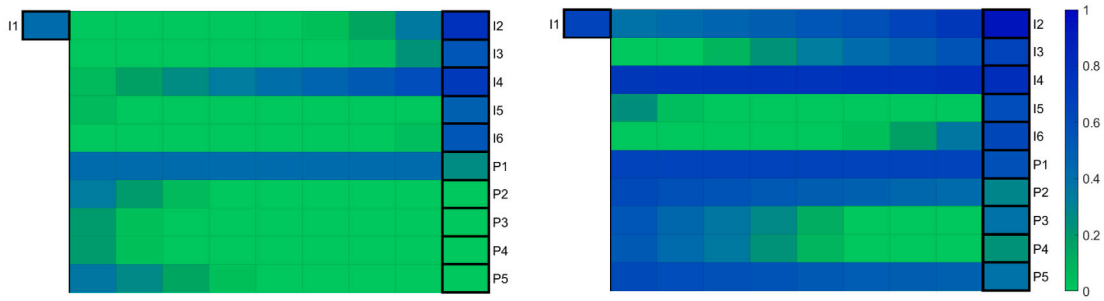


Fig. 6. Water saturation in a GPSNet model with all-to-all communication after 15 time steps (left) and at the end of simulation of Case 0 (right). Out of the 55 interwell connections, the plots only show the connections involving injector I1.

Ranked in terms of accuracy, the GPSNet model with connections derived from flow diagnostics is less accurate than the $5 \times 6 \times 1$ CGNet model, the four models with injector–producer connections fall between the two coarsest CGNet models, while GPSNet with all-to-all connections ranks between the second and third-most accurate CGNet models.

In terms of computational efficiency, a comparison is more complex. When mapped into a reservoir simulator, the computational cost of a forward simulation is generally proportional to the number of grid cells, which, for comparable accuracy in prediction, is significantly lower for the CGNet models. On the other hand, the 1D flow paths in the GPSNet models imply that the resulting discrete systems are sparser and essentially block-triangular, which can be utilized to localize and significantly accelerate the nonlinear solver; see, e.g., Appleyard and Cheshire, 1982 for details. Such solvers are not yet available in standard reservoir simulators, and our conclusion is therefore that the CGNet approach is more computationally efficient.

For the training process, the forward simulations are a major contributor to the overall computational cost. The second contribution comes from the adjoint equations. These are of the same size as the forward problem but are quicker to solve since they are linear. For the relatively small grid sizes considered here (36 to 600 cells) and number of parameters (51 to 1479), our simulators in MRST are unfortunately so dominated by computational overhead that it is not possible to draw any well-founded conclusions.

4.1.3. Predictive power outside the span of the training data

A next natural question is the predictive power of the GPSNet/CGNet models when applied to cases having reservoir states and well responses outside the data set used to train the model. To investigate this, we consider three different displacement scenarios applied over the same simulation horizon as in Case 0:[5]

Case 1: Fixed injection rates and bottom-hole pressures as in Case 0, but with a 50% random perturbation of the individual water rates and a 10% variation of the individual bottom-hole pressures.

Case 2: Case 0 but with the most dominant well, producer P1, shut in after eight years.

Case 3: Case 0 but with P1 shut in and I1 and I2 converted to producers from the start of the production (time zero).

Table 2 reports misfits in prediction for a subset of the GPSNet and CGNet models just discussed. For Case 1, the fluid displacement follows much the same flow paths as in the training data. All models are therefore able to predict the qualitative behavior of the well responses, but the large difference in how the same total flow volume is distributed amongst the wells compared with the training data causes a slight degradation in the prediction accuracy, as can clearly be seen for P1 in the left column of Fig. 7. Unlike in Case 0, our basic GPSNet model with injector–producer topology is now the least accurate. Results with two paths per injector–producer connection or all-to-all connections are slightly better, but not (significantly) better than the coarsest CGNet model and are hence not reported for brevity. Flow diagnostics on Case 1 identifies one interwell connection not found in Case 0, which immediately suggests that the GPSNet model derived for Case 0 may not be rich enough to give good predictions for Case 1; this is confirmed by our simulations (not reported for brevity).

Case 2 has two distinct production phases: During the first 2/3 of the simulation the displacement is identical to the reference case and well responses are thus predicted well for all the CGNet models. Because the fluid system is incompressible, shutting P1 in after eight years causes an instant redistribution of flow paths and a step increase in the liquid rates in the remaining producers. Such an event is not part of the training data but all models are nonetheless able to capture the qualitative behavior of the liquid rates in all wells, even though the quantitative deviation from the truth case are significant, as shown for the producer with the largest mismatch in the middle column of Fig. 7. It is also interesting to observe a clear example of overfitting for the CGNet with most parameters ($10 \times 14 \times 3$) in the sense that the more we train the model, the worse the prediction mismatch gets. The basic GPSNet model suffers from an incorrect wiggly behavior, as observed earlier, and is generally the least accurate. The results for the other GPSNet configurations are no better and are not reported for brevity.

In Case 3, the displacement inside the main section of the reservoir (the area bounded by injectors I1 to I4) has been converted from a five-spot configuration to a line drive and is thus significantly different from the training simulation. Because we now have a different injector–producer topology, the only GPSNet we can hope will give good results is the one with all-to-all connections. For CGNet, all models should still be applicable without modifications because they do not rely on a specific interwell topology. However, even though all trained models performed better than their untrained (upscaled) counterparts, they cannot be said to give acceptable predictive accuracy. The predictions obtained with GPSNet have similar low accuracy. Altogether, this is disappointing (albeit not surprising) with respect to the generality of the GPSNet and CGNet models and shows the importance of specifying training data with sufficient variation to ensure generality.

4.1.4. Training with multiple simulations

To improve the generality of the models, we will train the networks using the three different production scenarios described in Case 1 to 3. To construct training data, we apply the random-perturbation procedure described earlier to the injection rates and bottom-hole pressures prescribed as controls in Cases 1 to 3. (We do not include

[5] Complete code for this validation test for GPSNet can be found in the `norneGPSNetGeneralityTest.m` script in the `network-models` module of MRST from release 2022a and onward. The code for CGNet is completely analogous and can easily be constructed with the help of the `norneCGNetAdoint.m` script.

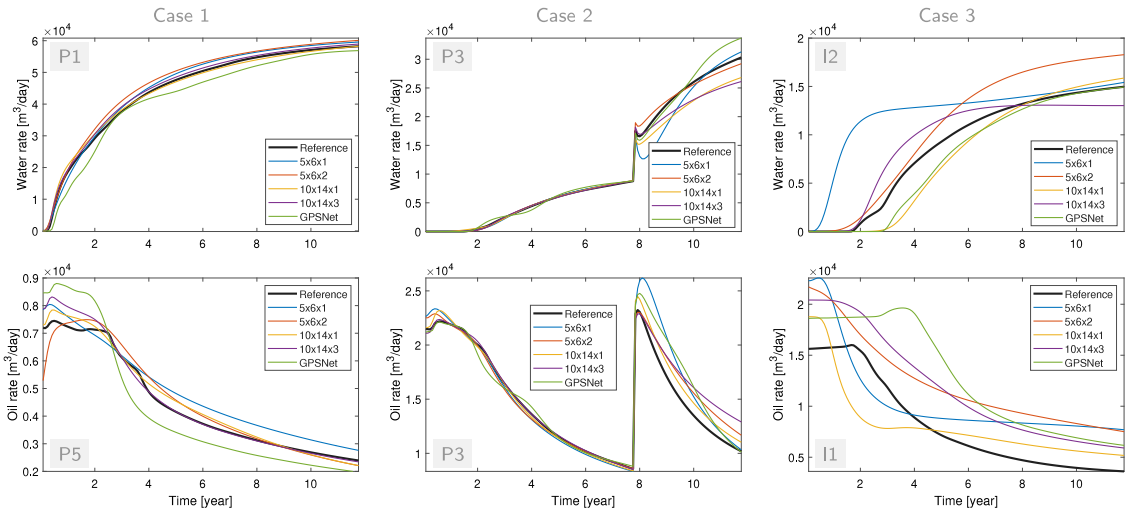


Fig. 7. Well responses predicted on Case 1 to Case 3 by GPSNet and CGNet models trained with 300 iterations on training data derived from the original setup (Case 0). The GPSNet uses injector–producer topology for Cases 1 and 2 but must use all-to-all topology for Case 3 because injectors I1 and I2 have been converted to producers.

Table 2
Misfit in the prediction of three alternative Norne waterflooding scenarios having well responses that (partially) fall outside of the data used to train the GPSNet and CGNet models. As in Table 1, the misfit is measured as the scaled sum of the discrepancies in bottom-hole pressure and phase rates. The first column gives the number of iterations used to train the model, with a zero value indicating pure upscaling without any training. GPSNet uses the default setup with a single interwell connections with ten grid cells for all injector–producer pairs, except for Case 3, which can only be run with the all-to-all network.

Its	Case1				Case 2				Case 3			
	GPSNet	5 × 6 × 1	5 × 6 × 2	10 × 14 × 3	GPSNet	5 × 6 × 1	5 × 6 × 2	10 × 14 × 3	GPSNet	5 × 6 × 1	5 × 6 × 2	10 × 14 × 3
0	3.88e-1	6.43e-0	1.22e-0	5.42e-1	5.31e-1	7.34e-0	1.05e-0	5.16e-1	2.44e-1	3.18e-0	3.93e-1	8.43e-2
50	1.87e-2	1.50e-2	1.24e-2	5.02e-3	3.40e-2	1.47e-2	1.21e-2	8.29e-3	2.57e-2	1.16e-1	1.16e-1	4.03e-2
100	1.33e-2	1.05e-2	1.15e-2	3.00e-3	3.53e-2	1.99e-2	1.91e-2	9.44e-3	4.53e-2	1.33e-1	1.91e-1	3.86e-2
200	1.42e-2	9.99e-3	1.14e-2	2.89e-3	2.22e-2	1.81e-2	1.09e-2	9.52e-3	1.06e-1	6.16e-2	3.71e-2	1.43e-2
300	1.64e-2	6.21e-3	9.75e-3	2.36e-3	3.24e-2	1.29e-2	1.09e-2	1.15e-2	1.38e-1	1.02e-1	3.27e-2	1.41e-2
400	1.66e-2	5.39e-3	8.51e-3	2.34e-3	3.76e-2	1.63e-2	1.08e-2	1.62e-2	1.51e-1	1.21e-1	4.54e-2	1.66e-2
500	1.72e-2	5.07e-3	6.30e-3	1.86e-3	3.67e-2	1.65e-2	1.06e-2	1.93e-2	1.88e-1	1.37e-1	4.09e-2	1.58e-2
750	1.66e-2	4.30e-3	6.19e-3	1.28e-3	2.41e-2	1.68e-2	1.05e-2	1,97e-2	1.15e-1	1.40e-1	4.35e-2	1.53e-2

Table 3
Misfit in the prediction of alternative Norne waterflooding scenarios predicted by three CGNet models and an all-to-all GPSNet, all trained with 200 iterations on random-variation training data derived from the scenarios of Cases 1 to 3.

CGNet	Case 0	Case 1	Case 2	Case 3	Case 4
5 × 6 × 1	3.03e−03	2.36e−03	3.11e−03	3.18e−03	1.48e−02
5 × 6 × 2	2.76e−03	1.89e−03	2.81e−03	2.02e−03	8.78e−03
10 × 14 × 3	1.52e−03	1.48e−03	3.00e−03	5.07e−04	7.18e−03
GPSNet	5.71e−03	3.42e−03	2.82e−03	2.47e−03	2.55e−02

Case 0, as its dynamics is very similar to Case 1.) In each step of the misfit minimization, we will thus perform three forward simulations corresponding to the random perturbed versions of Case 1 to 3, and measure the misfit for each of these simulations. However, the L-BFGS-B method requires a single misfit value and hence we define the overall misfit function as the mean of the misfits from each of the three different data sets, with the gradient used in the optimization algorithm defined analogously. Once each network model has been trained simultaneously for all three scenarios, we go back and evaluate how well it predicts the three individual base cases (Case 1 to 3). Fig. 8 reports the same well responses as in Fig. 7 computed with three different CGNet models and GPSNet with all-to-all connections, all trained with 200 misfit minimization iterations over the three new training data sets; Table 3 reports the corresponding misfits.

With the new training data, the prediction quality improves for Cases 1 to 3, and is now within the variation seen in the training data. For Case 0, the misfits increase slightly to be at the same level of accuracy as obtained for the other cases. The reason for this increase is that the misfit reduction for three sets of training data is somewhat slower from iteration ten and onward compared to training with a single simulation, and for a fixed number of iterations the match is thus slightly worse. In comparing the two network types, we notice again that whereas the misfits largely have similar magnitude, the qualitative behavior of GPSNet is generally less correct due to delayed breakthrough and wiggles caused by breakthrough in individual flow paths.

We end by comparing the predictive power of CGNet, trained on a single simulation derived from Case 0 and on three simulations derived from Case 1 to 3, using a new setup that once again deviates from the training data:

Case 4: Case 0 but with producer P1 shut in the period between four and eight years.

In this case, the well curves of the remaining wells will exhibit two distinct step changes, corresponding to the almost instant redistribution interwell communication when well P1 is shut in and turned back on. Fig. 9 shows that once again, we do not obtain a fully satisfactory match of events that lie outside the span of the training data, but training with a richer data set improves the overall match, as expected.

4.2. Brugge model

In this example, we consider optimization of net-present-value (NPV) for the synthetic Brugge model (Peters et al., 2013) by using a trained CGNet or GPSNet as proxy. The initial saturation and wells of the fine-scale model are depicted in the left plot of Fig. 10. There are 20 producers placed in the oil cap, surrounded by 10 injectors. The grid has 44474 active cells, and the fluid model is two-phase oil–water.

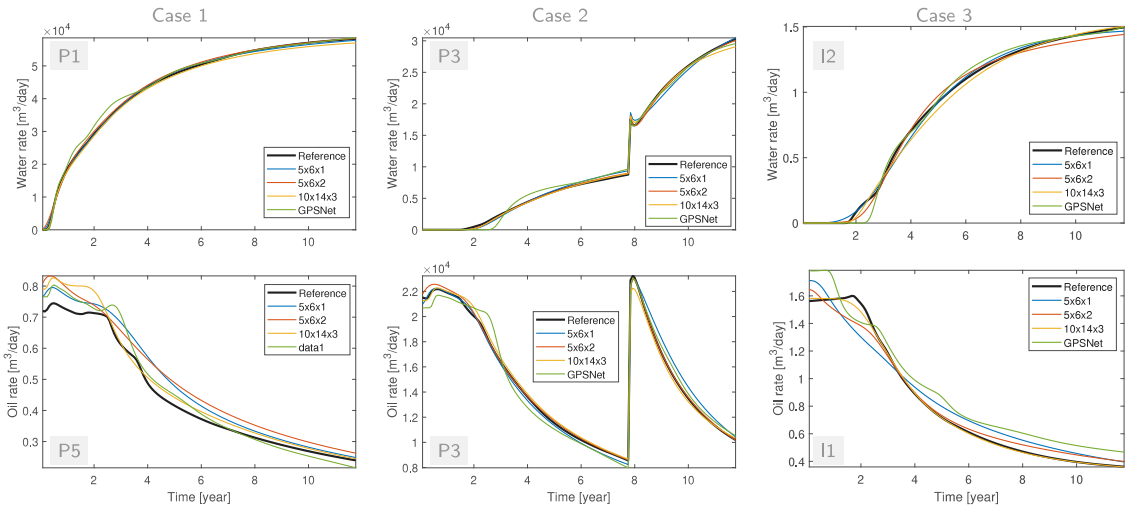


Fig. 8. Well responses predicted on Case 1 to Case 3 by three different CGNet models and GPSNet with all-to-all topology, all trained with 200 iterations on random-variation training data derived from the scenarios of these three cases.

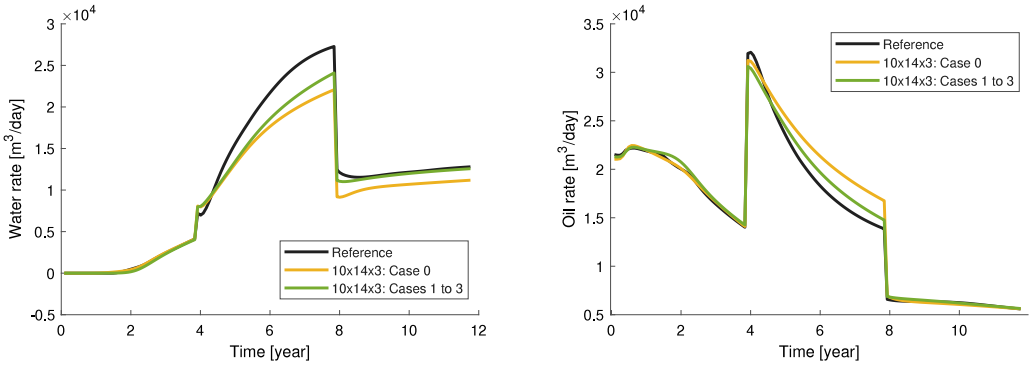


Fig. 9. Water and oil rates for producer P3 predicted on Case 4 by two different 10 × 14 × 3 CGNet models trained with 200 iterations on random-variation training data derived only from Case 0 (yellow curves) and from the three different scenarios in Case 1 to 3 (green curves). The misfits in prediction are 2.32e-2 and 7.18e-3, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

There are seven *saturation*-regions with different relative-permeability and capillary-pressure curves. In the base setup, injectors are set to inject water at 636.95 m^3/day with an upper limit on BHP set to 180 bar. Producers are set to produce at a liquid rate of 317.98 m^3/day with a lower BHP limit of 50 bar. The simulation horizon is 10 years. With these settings, five injectors and three producers hit their BHP-limit during simulation of the (full) reference model.

4.2.1. Model training

To build the CGNet model, we initially partition the grid in 14 × 14 blocks laterally. To obtain more resolution around wells, we also split off each of the 30 groups of cells containing well perforations as individual blocks. The resulting grid blocks are depicted in the second plot of Fig. 10. In total, the resulting CGNet model consist of 196 cells (graph vertices) and 352 interfaces (graph edges). The simple upscaling procedure we use for initialization assumes regularly-shaped blocks, and the initial untrained CGNet-model therefore produces well responses that are far from those of the full model. We introduce additional parameters for scaling the relative permeability curves for each coarse block in the CGNet to reproduce the effect of the multiple fluid regions in the fine-scale fluid model. In particular, we introduce parameters for scaling both the endpoints and the values of the water/oil relative permeability curves. Accordingly, for each block there are 6 parameters for tuning the shape of the fluid functions in addition to the pore-volume parameter. With well connection factors and transmissibilities added, the resulting CGNet model has $7\times196+30+352 = 1754$ parameters. We note that the parameters for, e.g., oil relative permeability in the water zone will have little or no effect on the output. Accordingly, the number of *effective* parameters is probably substantially smaller. We finally note that initial saturations are not included as parameters for this CGNet model but are simply set as the volume average over the coarse blocks.

For GPSnet, we introduce connections for every injector–producer pair, giving a total of 200 connections. Each connection is discretized by ten grid cells, so the GPSNet model contains a total of 2000 cells. We use the same parameters as for the CGNet model, except that the same scaling of transmissibilities and relative permeability scalers are applied to all cells and cell interfaces of each row of cells representing a single interwell connection. In addition, we introduce parameters for initial saturation because reasonable saturation values for the GPSNet model are more difficult to determine. Accordingly, our GPSNet model contains $8 \times 200 + 2000 = 3600$ parameters.

The models are trained in the same way as in the previous example, except that random perturbation are not introduced to excite more states and that we scale the mismatches in liquid rates and bottom-hole pressures by 300 m^3/day and 50 bar, respectively. Fig. 11 depicts bottom-hole pressure and liquid rates for four of the producers. Both the CGNet model and the GPSNet model are obtained by 150 training iterations and show quite good match with the fine reference model (the four wells considered in Fig. 11 constitute a representative sample). Further training could have given us a closer match, but here we wish to test the network models as proxies for optimization of NPV and perform subsequent training based on new fine-scale simulations as we approach an optimum.

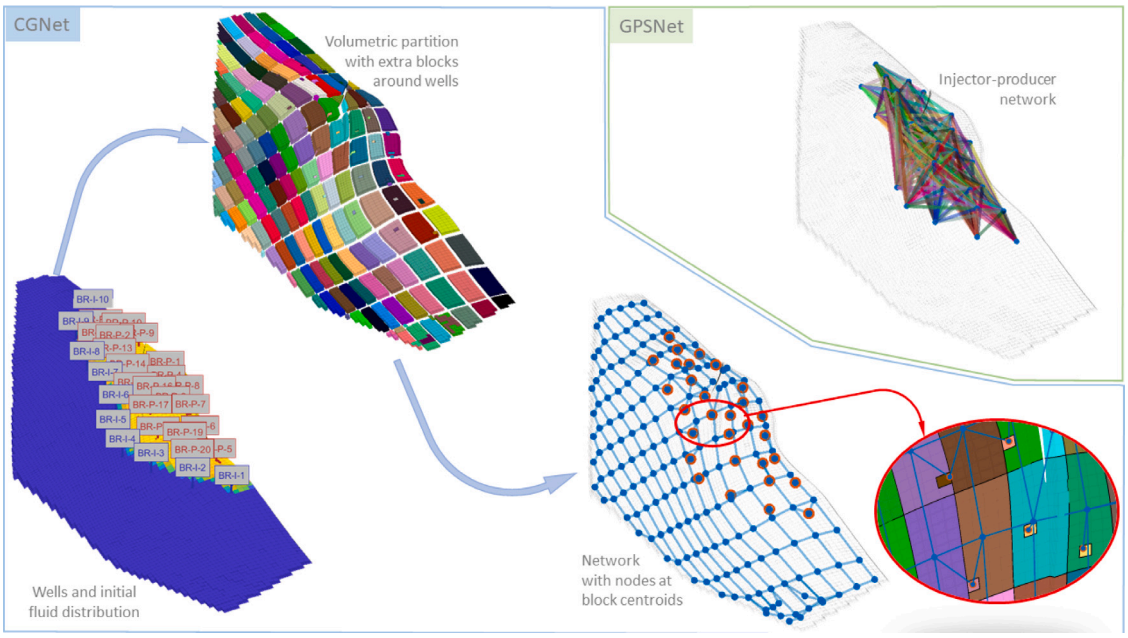


Fig. 10. The Brugge model. In the lower-left plot, injectors are shown with blue labels and producers with red labels. The upper-left plot depicts the coarse partition used to create the CGNet model with corresponding connectivity graph shown in the lower-right plot. Here, nodes that are connected to wells are highlighted using red circles. Because an extra block is created around each well, the well nodes are either connected to a single non-well node if the extra block is contained inside another block, or to two non-well nodes if the extra well block touches the edge of the surrounding block, as shown in the inset to the far right. (More than two connections are only possible for multilayered partitions or if the well block lies at the corner of the surrounding block; this is not the case here.) The upper-right plot depicts the connectivity graph of the GPSNet model in which all injectors are connected to all producers.

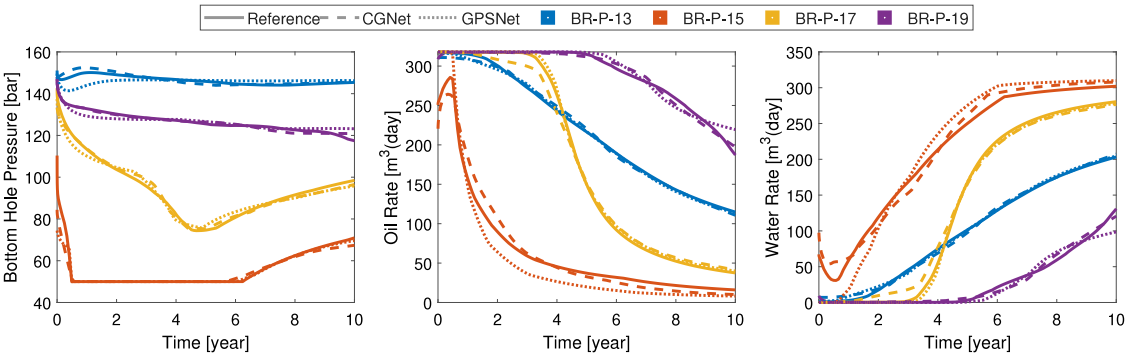


Fig. 11. Bottom-hole pressure and liquid rates for four of the producers for the base-case setup of the Brugge model. Results shown for the reference case (solid lines), trained CGNet model (dashed lines), and trained GPSNet (dotted lines).

Table 4
Target and upper/lower limit values for well controls of the base case and corresponding bounds used in the optimization.

	base case	lower bound	upper bound
Injector target rate [m^3/day]	636.95	10	1000
Injector upper limit BHP [bar]	180.00	160	180
Producer target liquid rate [m^3/day]	317.98	10	500
Producer lower limit BHP [bar]	50.00	50	120

4.2.2. Optimization of net-present value

Next, we employ the two network models to optimize net-present value (NPV). In the NPV-function, oil revenue is set to 50 USD/stb, water injection cost to 3 USD/stb, and water production cost to 3 USD/stb. We set the yearly discount rate to 10 %. The left plot of Fig. 12 depicts how the NPV evolves for the base case, as predicted by the reference model and the two network models. Both the reduced models match the reference closely. At the end of simulation, the NPV is 1.886×10^{10} USD for the reference and 1.879×10^{10} USD for both CGNet and GPSNet (first row of Table 5).

For optimization, we split the ten-year simulation horizon into ten equal control periods. During these control periods, each injector can be controlled either by rate, BHP, or a mix of the two, as our adjoint implementation calculates the derivative of whichever control mode is active; see, e.g., Kraaijevanger et al. (2007) and Krogstad et al. (2018). Similarly, producers are controlled by liquid rate and/or BHP. With 30 wells having 10 control steps each, this adds up to a control vector with $2\times30\times10 = 600$ entries. The control bounds used in the optimization are listed in Table 4. The NPV-optimization converges in approximately 40 iterations. The corresponding curves for the trained CGNet and GPSNet models are shown in the middle plot of Fig. 12; see the right plot for magnification of the upper-right corner.

Based on the results from the Norne example, we cannot necessarily expect that the network models trained against the base-case scenario will match well with the fine-scale simulation for the new and optimized well settings. Accordingly, we retrained the network models based on the fine-scale simulation with the new and optimized well setting and then subsequently performed NPV-optimization from the new start points we obtained by simulating the previously optimized settings with the new retrained models. These start points will not

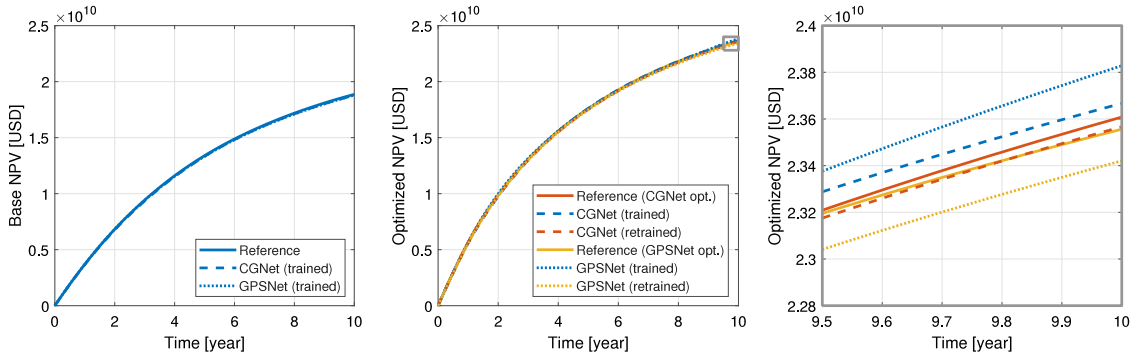


Fig. 12. The left plot shows how NPV evolves for the Brugge base case; curves for the reference model and CGNet/GPSNet overlay. The middle plot shows optimized NPV predicted before and after retraining and re-optimization; curves for the reference model are computed using the well settings suggested after the second optimizations with the retrained network models. The right plots shows an enlargement of the upper-right corner of the middle plot.

Table 5
Net-present values at the end of simulation (see Fig. 12) computed by the fine-scale and the two network models on three different scenarios: the base case, after optimization with CGNet, and after optimization with GPSNet. For the fine-scale simulations, we use well settings obtained by the retrained and reoptimized network models.

		CGNet		GPSNet	
	fine-scale	trained	retrained	trained	retrained
Base case [10^{10}\$]	1.886	1.879	–	1.879	–
CGNet optimized [10^{10}\$]	2.361	2.367	2.357	–	–
GPSNet optimized [10^{10}\$]	2.356	–	–	2.383	2.342

necessarily be optima for the retrained network models. However, the improvements in NPV, as predicted by the fine-scale model after this second optimization step, were very limited because both network models matched the fine-scale simulation well, even before the retraining. The middle and right plots of Fig. 12 show NPV curves for GCNet and GPSNet after the first optimizations (trained) and after the second optimizations (retrained), as well as NPV values computed by the fine-scale model with well controls proposed by the second optimizations. Table 5 displays the corresponding NPV-values at the end of simulation (10 years).

Figs. 13 and 14 compare the mismatch in predicted bottom-hole pressures and liquid rates for the CGNet and GPSNet models, respectively, applied to the optimized well schedules. The producers are the same as in Fig. 11. The optimized well settings are distinctly different for the two network models, even though similar trends can be observed. It has previously been observed (see e.g., van Essen et al., 2011) that the response surfaces of well-control optimization problems do not exhibit distinct extrema but rather *ridges* and *valleys*. It is therefore to be expected that the two models give different optimized controls. Because no attempt was made to regularize the controls in time, sharp jumps between control steps are also anticipated. For CGNet, we observe close agreement with the fine-scale simulation in both pressure and rates. For GPSNet, on the other hand, larger discrepancies can be observed for pressure. This could be a result of convergence problems we observed in the forward simulation of the GPSNet model (due to extensive well crossflow) during the training procedure, which in turn could hamper the gradient accuracy.

5. Concluding remarks

We have demonstrated how GPSNet and our new CGNet can be used to derive reduced models by training the model against simulation responses from a fine-scale model. Although the models are guaranteed to give physical results, their predictions will only be (highly) accurate in a (small) parameter domain surrounding the training data. To improve generality, we therefore proposed to train against simulations with randomly perturbed well controls that excite a larger variation in well responses and reservoir states; more systematic and targeted perturbations will likely produce similar results. For simulation studies encompassing large differences in flow patterns and interwell connections, it is recommended to train the models using a selection of representative displacement setups. In this regard, CGNet is more general because the topology of the underlying network is not given by the interwell connectivity, which may vary from one scenario to another.

At their best, and for a comparable number of parameters, CGNet and GPSNet have similar predictive power. However, we observed that CGNet models are easier to configure and initialize, particularly for cases involving nonuniform fluid distributions. In our opinion, it is also simpler to systematically increase the accuracy of CGNet. One explanation for this is that CGNet has richer network topology and more parameters relative to the number of grid cells. For a comparable number of parameters, evaluation of a CGNet model can therefore be expected to be faster. The richness of the network topology can also be increased in GPSNet by adding non-well nodes, e.g., as discussed by Leeuwenburgh et al. (2022), but this quickly increases the cell count of the corresponding 2D+ simulation model, as each new flow path connecting two non-well nodes or a well node and non-well nodes must be represented by a row of cells. Altogether, our conclusion is therefore that CGNet offers a more efficient and robust approach to reduced modeling than GPSNet.

In terms of computational speed, CGNet and GPSNet are expected to have similar performance. A typical configuration of GPSNet models will give larger but sparser systems than CGNet. Which is the more efficient to solve will depend strongly on the linear solver. In our MRST implementation, the network models discussed herein typically run within 20–60 s because the computational overhead of MATLAB and the general prototyping framework is significant for so small models. Preliminary experiments performed with a compiled simulator (written in Julia), show that the same simulations run within a few seconds at most and that the training may take minutes and be significantly faster than in MATLAB.

Herein, we have used the L-BFGS-B method for the misfit minimization, which is quite efficient in reducing the initial mismatch but experiences slow convergence once the mismatch has been reduced a few orders of magnitude. Nonlinear least-square problems are traditionally solved with Gauss–Newton, Levenberg–Marquardt, or variants of these (see e.g., Nocedal and Wright, 2006). Experience from Lie and Krogstad (2022) shows that switching to Levenberg–Marquardt can accelerate the misfit reduction and speed up the training significantly. This confirms previous observations by Li et al. (2003) from history matching. A drawback with Gauss–Newton/Levenberg–Marquardt methods is that they require a representation of the sensitivity matrix $\boldsymbol{J} = d\boldsymbol{y}/d\boldsymbol{\theta}$. In a simulation setting, this implies either n_y (number of data points) linearized forward simulations or n_θ (number of parameters) adjoint simulations. Accordingly, for large-scale models with many

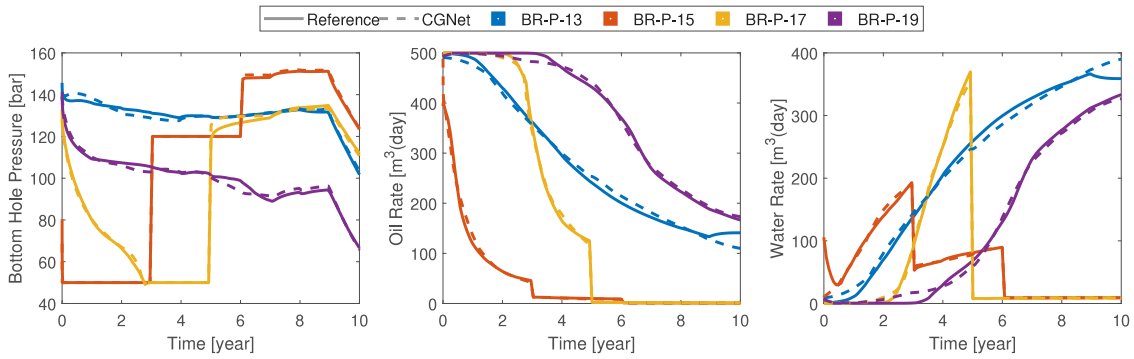


Fig. 13. Bottom-hole pressures and liquid rate for four of the producers for Brugge with well controls optimized by CGNet. Results are shown for the reference case (solid lines), retrained CGNet model (dashed lines).

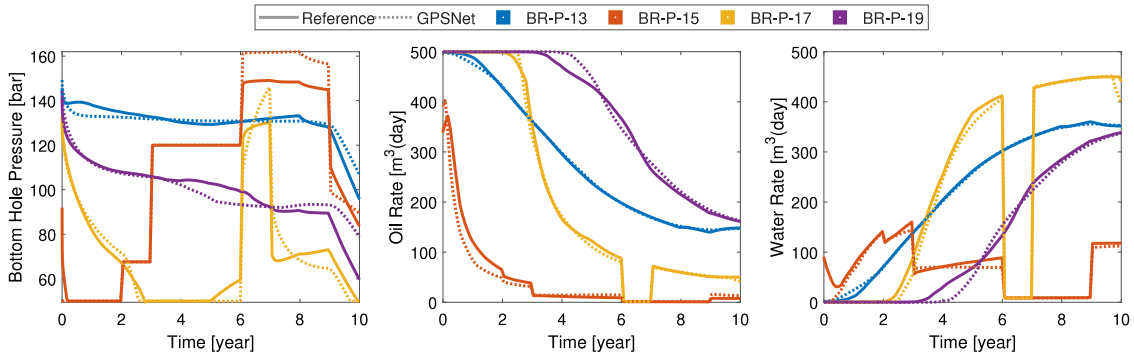


Fig. 14. Bottom-hole pressures and liquid rate for four of the producers for Brugge with well controls optimized by GPSNet. Results are shown for the reference case (solid lines), retrained GPSNet model (dashed lines).

data points and/or parameters, these methods become computationally intractable. For the (low-order) network-type models considered here, however, multiple linearized-forward/adjoint simulations can be executed simultaneously, which ensures their high efficiency. If neither Levenberg–Marquardt nor L-BFGS-B can be used, another alternative could be to use a gradient-free method such as the ensemble smoother with multiple data assimilation (ES-MDA), as compared with L-BFGS-B in Borregales et al. (2021). With ES-MDA, however, one will eventually experience ensemble collapse as the mismatch is driven close to zero (which is the aim for the training we consider here).

CRediT authorship contribution statement

Knut-Andreas Lie: Methodology, Software, Writing. **Stein Krogstad:** Methodology, Software, Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors acknowledge funding from the Research Council of Norway through grant no. 280950 with co-funding from Equinor Energy AS, Norway, Total E&P Norge AS, Norway, and Wintershall DEA Norge AS, Norway and through grant no. 308817 with co-funding from Equinor ASA, Norway.

References

Appleyard, J.R., Cheshire, I.M., 1982. The Cascade method for accelerated convergence in implicit simulators. In: European Petroleum Conference, London, United Kingdom, October 1982. Society of Petroleum Engineers, http://dx.doi.org/10.2118/12804-ms.

Borregales, M.A., Holm, H.H., Møyner, O., Krogstad, S., Lie, K.-A., 2021. Numerical comparison between ES-MDA and gradient-based optimization for history matching of reduced reservoir models. In: SPE Reservoir Simulation Conference, 4–6 October 2021, on-Demand. http://dx.doi.org/10.2118/203975-MS.

Borregales, M., Møyner, O., Krogstad, S., Lie, K., 2020. Data-driven models based on flow diagnostics. In: ECMOR XVII – 17th European Conference on the Mathematics of Oil Recovery. EAGE, http://dx.doi.org/10.3997/2214-4609.202035122.

Byrd, R.H., Lu, P., Nocedal, J., Zhu, C., 1995. A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. 16 (5), 1190–1208. http://dx.doi.org/10.1137/0916069.

Costa, L.A.N., Maschio, C., Schiozer, D.J., 2014. Application of artificial neural networks in a history matching process. J. Pet. Sci. Eng. 123, 30–45. http://dx.doi.org/10.1016/j.petrol.2014.06.004.

Emerick, A.A., Reynolds, A.C., 2013. Ensemble smoother with multiple data assimilation. Comput. Geosci. 55, 3–15. http://dx.doi.org/10.1016/j.cageo.2012.03.011.

Fraces, C.G., Papaioannou, A., Tchelepi, H., 2020. Physics informed deep learning for transport in porous media. Buckley Leverett problem. arXiv:2001.05172.

Guo, Z., Reynolds, A.C., 2019. INSIM-FT in three-dimensions with gravity. J. Comput. Phys. 380, 143–169. http://dx.doi.org/10.1016/j.jcp.2018.12.016, URL http://www.sciencedirect.com/science/article/pii/S002199911830812X.

Guo, Z., Reynolds, A.C., Zhao, H., 2018a. A physics-based data-driven model for history matching, prediction, and characterization of waterflooding performance. SPE J. 23 (02), 367–395. http://dx.doi.org/10.2118/182660-PA.

Guo, Z., Reynolds, A.C., Zhao, H., 2018b. Waterflooding optimization with the INSIM-FT data-driven model. Comput. Geosci. 22 (3), 745–761. http://dx.doi.org/10.1007/s10596-018-9723-y.

Iino, A., Jung, H.Y., Onishi, T., Datta-Gupta, A., 2020. Rapid simulation accounting for well interference in unconventional reservoirs using fast marching method. In: SPE/AAPG/SEG Unconventional Resources Technology Conference. http://dx.doi.org/10.15530/urtec-2020-2468.

Jin, Z.L., Liu, Y., Durlofsky, L.J., 2020. Deep-learning-based surrogate model for reservoir simulation with time-varying well controls. J. Pet. Sci. Eng. 192, 107273. http://dx.doi.org/10.1016/j.petrol.2020.107273.

Karypis, G., Kumar, V., 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J. Sci. Comput. 20 (1), 359–392. http://dx.doi.org/10.1137/s1064827595287997.

Kiærr, A., Lødøen, O., De Bruin, W., Barros, E., Leeuwenburgh, O., 2020. 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. EAGE, http://dx.doi.org/10.3997/2214-4609.202035099.

Kim, Y.D., Durlofsky, L.J., 2021. A recurrent neural network–based proxy model for well-control optimization with nonlinear output constraints. SPE J. 26 (04), 1837–1857. http://dx.doi.org/10.2118/203980-PA.

Kim, J., Yang, H., Choe, J., 2020. Robust optimization of the locations and types of multiple wells using CNN based proxy models. J. Pet. Sci. Eng. 193, 107424. http://dx.doi.org/10.1016/j.petrol.2020.107424.

Kraaijevanger, J.F.B.M., Egberts, P.J.P., Valstar, J.R., Buurman, H.W., 2007. Optimal waterflood design using the adjoint method. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers, Houston, Texas, U.S.A., http://dx.doi.org/10.2118/105764-MS.

Krogstad, S., Nilsen, H.M., Møyner, O., Rasmussen, A.F., 2018. Well control optimization of the OLYMPUS case using MRST and OPM. In: EAGE/TNO Workshop on OLYMPUS Field Development Optimization. Barcelona, Spain, http://dx.doi.org/10.3997/2214-4609.201802287.

Krogstad, S., Raynaud, X., Nilsen, H.M., 2016. Reservoir management optimization using well-specific upscaling and control switching. Comput. Geosci. 20 (3), 695–706. http://dx.doi.org/10.1007/s10596-015-9497-4.

Leeuwenburgh, O., Egberts, P., Barros, E., Turchan, L., Dilib, F., Lødøen, O., Bruin, W.D., 2022. Application of coupled flow network and machine learning models for data-driven forecasting of reservoir souring. In: ECMOR 2022. European Association of Geoscientists & Engineers, http://dx.doi.org/10.3997/2214-4609.202244046.

Lerlertpakdee, P., Jafarpour, B., Gildin, E., 2014. Efficient production optimization with flow-network models. SPE J. 19 (06), 1083–1095. http://dx.doi.org/10.2118/170241-PA.

Li, R., Reynolds, A.C., Oliver, D.S., 2003. History matching of three-phase flow production data. SPE J. 8 (04), 328–340. http://dx.doi.org/10.2118/87336-PA.

Lie, K.-A., 2019. An Introduction to Reservoir Simulation using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST). Cambridge University Press, http://dx.doi.org/10.1017/9781108591416.

Lie, K., Krogstad, S., 2022. Data-driven modelling with coarse-grid network models. In: ECMOR 2022. European Association of Geoscientists & Engineers, http://dx.doi.org/10.3997/2214-4609.202244065.

Lie, K.-A., Møyner, O. (Eds.), 2021. Advanced Modeling with the MATLAB Reservoir Simulation Toolbox. Cambridge University Press, http://dx.doi.org/10.1017/9781009019781.

Lie, K.-A., Møyner, O., Natvig, J.R., 2017. Use of multiple multiscale operators to accelerate simulation of complex geomodels. SPE J. 22 (6), 1929–1945. http://dx.doi.org/10.2118/182701-pa.

Liu, Z., Reynolds, A.C., 2021. Gradient-enhanced support vector regression for robust life-cycle production optimization with nonlinear-state constraints. SPE J. 26 (04), 1590–1613. http://dx.doi.org/10.2118/204236-PA.

Lorentzen, R.J., Luo, X., Bhakta, T., Valestrand, R., 2019. History matching the full Norne field model using seismic and production data. SPE J. 24 (4), 1452–1467. http://dx.doi.org/10.2118/194205-PA.

Lutidze, G., 2018. Stellnet – Physics-Based Data-Driven General Model for Closed-Loop Reservoir Management (Ph.D. thesis). The University of Tulsa.

Mamonov, A., Couet, B., Bailey, W.J., Prange, M., Djikpesse, H.A., Druskin, V., 2007. Optimal grid coarsening: A fast proxy for large reservoir simulations. In: SPE Reservoir Characterisation and Simulation Conference and Exhibition, Abu Dhabi, UAE, October 2007. http://dx.doi.org/10.2118/111378-MS.

Maucec, M., Jalali, R., 2022. GeoDIN – geoscience-based deep interaction networks for predicting flow dynamics in reservoir simulation models. SPE J. 27 (03), 1671–1689. http://dx.doi.org/10.2118/203952-PA.

Morteza, K., Behnam, J., King, M.J., 2019. Inference of global reservoir connectivity from static pressure data with fast coarse-scale simulation models. Math. Geosci. 51 (5), 625–648. http://dx.doi.org/10.1007/s11004-018-9772-8.

Møyner, O., Krogstad, S., Lie, K.-A., 2015. The application of flow diagnostics for reservoir management. SPE J. 20 (02), 306–323. http://dx.doi.org/10.2118/171557-PA.

MRST, 2021. The MATLAB Reservoir Simulation Toolbox (MRST), Version 2021b. SINTEF, http://dx.doi.org/10.5281/zenodo.5517526, SINTEF Digital. URL www.mrst.no.

Navrátil, A., Rios, J., Kollias, G., Torrado, R., Codas, A., 2019. Accelerating physics-based simulations using end-to-end neural network proxies: An application in oil reservoir modeling. Front. Big Data 2, 33. http://dx.doi.org/10.3389/fdata.2019.00033.

Nocedal, J., Wright, S.J., 2006. Numerical Optimization, second ed. Springer, New York, NY, USA.

Peters, E., Chen, Y., Leeuwenburgh, O., Oliver, D.S., 2013. Extended Brugge benchmark case for history matching and water flooding optimization. Comput. Geosci. 50, 16–24. http://dx.doi.org/10.1016/j.cageo.2012.07.018.

Ren, G., He, J., Wang, Z., Younis, R.M., Wen, X.-H., 2019. Implementation of physics-based data-driven models with a commercial simulator. In: SPE Reservoir Simulation Conference. Society of Petroleum Engineers, http://dx.doi.org/10.2118/193855-MS.

Rodriguez-Torrado, R., Ruiz, P., Cueto-Felgueroso, L., Green, M.C., Friesen, T., Matringe, S., Togelius, J., 2021. Physics-informed attention-based neural network for solving non-linear partial differential equations. arXiv:2105.07898.

Sankaran, S., Sun, W., 2020. A flow network model based on time of flight for reservoir management. In: Abu Dhabi International Petroleum Exhibition and Conference. http://dx.doi.org/10.2118/203390-MS.

Tang, M., Liu, Y., Durlofsky, L.J., 2020. A deep-learning-based surrogate model for data assimilation in dynamic subsurface flow problems. J. Comput. Phys. 413, 109456. http://dx.doi.org/10.1016/j.jcp.2020.109456.

van Essen, G.M., Van den Hof, P.M., Jansen, J.D., 2011. Hierarchical long-term and short-term production optimization. SPE J. 16 (01), 191–199. http://dx.doi.org/10.2118/124332-PA.

Wang, N., Chang, H., Zhang, D., 2021a. Efficient uncertainty quantification and data assimilation via theory-guided convolutional neural network. SPE J. 26 (06), 4128—-4156. http://dx.doi.org/10.2118/203904-PA.

Wang, Z., He, J., Milliken, W.J., Wen, X.H., 2021b. 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. http://dx.doi.org/10.2118/200772-PA.

Zhang, F., Reynolds, A.C., 2002. Optimization algorithms for automatic history matching of production data. In: ECMOR VIII – 8th European Conference on the Mathematics of Oil Recovery. European Association of Geoscientists & Engineers, http://dx.doi.org/10.3997/2214-4609.201405958.

Zhang, K., Wang, Y., Li, G., Ma, X., Cui, S., Luo, Q., Wang, J., Yang, Y., Yao, J., 2021. Prediction of field saturations using a fully convolutional network surrogate. SPE J. 26 (04), 1824–1836. http://dx.doi.org/10.2118/205485-PA.

Zhao, H., Kang, Z., Zhang, X., Sun, H., Cao, L., Reynolds, A.C., 2015. INSIM: A data-driven model for history matching and prediction for waterflooding monitoring and management with a field application. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers, http://dx.doi.org/10.2118/173213-MS.

Zhong, Z., Sun, A.Y., Ren, B., Wang, Y., 2021. A deep-learning-based approach for reservoir production forecast under uncertainty. SPE J. 26 (03), 1314–1340. http://dx.doi.org/10.2118/205000-PA.