Physics-informed local-global underground fluid flow modeling with multiple sinks

D. Akhmetov *1 **G. Shutov** *1 **A. Ryabov** 1 **E. Burnaev** 1 2 **V. Vanovskiy** 1

*Equal contribution [1]*Center of Artificial Intelligence, Skolkovo Institute of Science and Technology, Moscow, Russia* [2]*Autonomous Non-Profit Organization Artificial Intelligence Research Institute, Moscow, Russia*. Correspondence to: D. Akhmetov D.Akhmetov@skoltech.ru.

1. Introduction

Conventional method of flow simulation in geological reservoirs is a numerical solution of differential equations [1]. Such methods can provide sufficient prediction accuracy, but the approach is time-consuming and computationally demanding. As an alternative, the application of neural network proxy-models for acceleration of hydrodynamic simulation has been the subject of intensive research nowadays [2]. One of the promising neural network approaches that takes into account the governed differential equations is called 'physics-informed machine learning' (PIML) [3]. The approach limits the solution space of a neural network increasing the reliability of predictions. However, existing works that apply machine learning to simulation of fluid flow in porous media does not take into account the specifics of flow around point sinks (e.g. wells in subsurface reservoir). In this research we develop a physics-informed neural network to solve a local problem of determining the pressure dynamics and flows around a single well. Then the local problem solution is used to solve a global problem of pressure field determination which is formulated as an inverse problem of finding the appropriate unsteady boundary condition for each single-well problem.

2. Materials and methods

2.1 Dataset generation

We consider 2D problem of flow in porous media, where each scenario of numerical simulation has unique permeability spatial distribution $K(X, Y)$ (Brugge dataset [4]), arbitrary number of production wells (point sinks) N_w, random bottomhole pressures p_{bhp} for this wells.

Our local model works on a well-by-well basis, i.e. the model input is not the whole considered reservoir of shape (X, Y), but the section of shape (x, y) with a well placed in the center. Thus we crop each spatial domain (X, Y) into sections around wells (N_w, x, y) as shown in Figure 1.

Note that P, K, X, Y stand for the whole reservoir's pressure, permeability and coordinates and small letters p, k, x, y stand for the same variables for a single section.

The detailed description of the dataset is given in Appendix A.

2.2 Forward problem solution

We tested UNET-like [5] and FNO [6] models for the forward problem solution, as they are often used in physics-informed machine learning tasks on spatial domains [7, 8, 9].

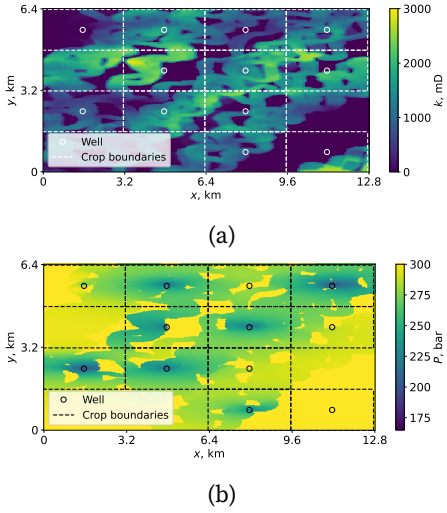

Fig. 1: Random scenario in generated dataset: figure a demonstrates the permeability field $K(X, Y)$ with 12 wells. Figure b shows pressure field in the second simulation step. Dashed lines outline the sections with well in the center that denoted by circles. We use these sections for neural network training.

Model takes as input permeability map k, constant well bottomhole pressure p_{bhp} and unsteady boundary condition $p_{bc}(t)$ which is the pressure at the section boundaries.

We divide the full dataset into three splits: train-validation-test sets in the proportion of 80 %, 10%, 10% for training, hyperparameters selection and validation.

To guide the model toward more realistic solution we utilized physics-informed loss during training as detailed in Appendix B.

2.3 Global problem solution

The global problem of simulating the whole reservoir could be reduced to local ones around each well if we would know the pressure boundary conditions for all the N_w local problems. The unsteady boundary conditions p_{bc} are unknown but should correspond to whole pressure field P, naturally, the inverse problem arises. It could be formulated as finding such pressure boundary conditions p_{bc} for all the wells that correspond to given permeability fields k derived from whole field K, bottomhole pressures p_{bhp}, production rates q and also to the global pressure field P obtained by joining single-wells solutions p. In such a

way global and local problems are linked via pressure for every local single-well problem that should be a corresponding crop of the global pressure field.

We tested two approaches of the boundary conditions p_{bc} optimization on unseen scenario, leveraging the fast inference of the trained neural networks from Section 2.2. The minimization function contains the residual between true production rates from numerical simulator $q(p)$ and those obtained by means of Peaceman approximation [10] $\hat{q}(\hat{p})$ in each optimization step.

The whole simulation period was split into train one used for the p_{bc} optimization (first 45 months) and prediction one (last 15 months) for validating the approach consistency.

The detailed description of both optimization strategies including pseudo-code is detailed in Appendix C.

3. Results

3.1 Forward problem

We validate the performance of the UNET and FNO models in prediction of pressure field $\hat{p}$ on the test set and obtain mean absolute percentage errors (MAPE) equal to 1.21 %, 1.49 %, respectively. Figure 2 shows the FNO performance, as example.

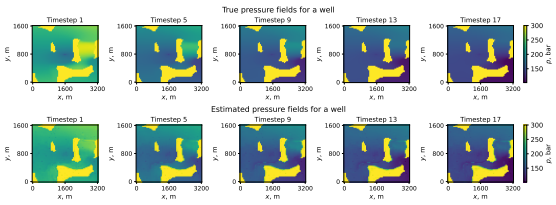

Fig. 2: Pressure fields estimated with FNO $\hat{p}$ (upper row) and the ground truth pressure fields p (lower row) for a random well section from test set.

3.2 Global problem

Two approaches were tested for boundary condition p_{bc} optimization via algorithm 1.

The analysis reveals that FNO model maintains physically-realistic pressure distribution in well sections for both optimization strategies, while in the case of UNET the resulting pressure fields are not realistic. The FNO's success here is due to its structural bias towards smooth, global solutions, which is a powerful form of implicit physics regularization. Thus, the further quantitative results are presented for the FNO architecture.

The first optimization approach shows physically realistic pressure fields producing 8.5 % MAPE. However, the divergence between estimated and true production rates is substantial. Quantitatively, RMSE for the first 45 months (fit period) equals to 12.4 $\frac{m^3}{day}$ and for the last 15 months (prediction period) 5.7 $\frac{m^3}{day}$.

The second approach succeeds not only in maintaining physically plausible pressure fields (6 % MAPE) but also achieves sound accuracy for production rates (0.6 $\frac{m^3}{day}$ and 2 $\frac{m^3}{day}$ RMSE for the fit and prediction periods, accordingly).

Figure 3 demonstrates the resulting production rates from optimization and the ground truth production rates.

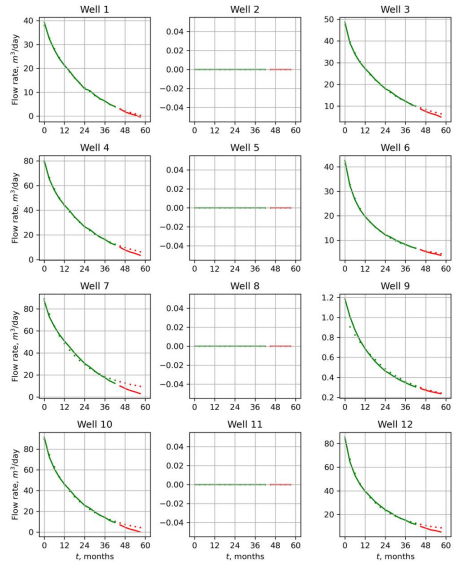

Fig. 3: Production rates from numerical simulator (dots) and obtained by optimization of boundary conditions (line). The green color denotes train (optimization) period and the red one outlines test period.

In appendix D figure A1 outlines comparison of the ground truth pressure fields p_{data} with those obtained as optimization result.

4. Conclusion

We propose a novel local-global approach to estimate pressure field in a geological reservoir with multiple sinks (production wells). For this we solve a local problem of pressure changes around a single well via physics-informed neural network model. We tested UNET and FNO as the forawrd models. The UNET produces mean absolute percentage error (MAPE) of 1.21 %, while FNO achieves 1.64 %. The global problem for the whole reservoir is formulated as to find such unsteady boundary conditions p_{bc} for all single-well problems that will correspond to the given permeability field k, well bottomhole pressures p_{bhp} and production rates q and also will correspond to global pressure solution P. We checked two strategies of boundary condition p_{bc} optimization (non-gradient- and gradient-based) and concluded that the superior configuration is FNO as a forward model and gradient-based optimization with pressure parametrization as a strategy for boundary condition search. The accuracy for pressure field: 6 % MAPE, and for production rates: 0.6 and 2 $\frac{m^3}{day}$ for train and prediction periods, respectively.

Acknowledgments

The study was supported by the Russian Science Foundation grant No. 24-41-02035, https://rscf.ru/en/project/24-41-02035/

References

[1] Laurie P Dake. *The practice of reservoir engineering (revised edition)*. Elsevier, 2001.

[2] Shuyu Sun and Tao Zhang. *Reservoir simulations: Machine learning and modeling*. Gulf Professional Publishing, 2020.

[3] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[4] E Peters, RJ Arts, GK Brouwer, CR Geel, Stan Cullick, Rolf J Lorentzen, Yan Chen, KNB Dunlop, Femke C Vossepoel, Rong Xu, et al. Results of the brugge benchmark study for flooding optimization and history matching. *SPE Reservoir Evaluation & Engineering*, 13(03):391–405, 2010.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[6] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[7] Jiuyu Zhao, Jinsui Wu, Han Wang, Yuxuan Xia, and Jianchao Cai. Single phase flow simulation in porous media by physical-informed unet network based on lattice boltzmann method. *Journal of Hydrology*, 639:131501, 2024.

[8] Navya Jain, Shantanu Roy, Hariprasad Kodamana, and Prapanch Nair. Scaling the predictions of multiphase flow through porous media using operator learning. *Chemical Engineering Journal*, 503:157671, 2025.

[9] Yunpeng Wang, Zhijie Li, Zelong Yuan, Wenhui Peng, Tianyuan Liu, and Jianchun Wang. Prediction of turbulent channel flow using fourier neural operator-based machine-learning strategy. *Physical Review Fluids*, 9(8):084604, 2024.

[10] Donald W Peaceman. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *Society of Petroleum Engineers Journal*, 23(03):531–543, 1983.

Appendix A. Dataset generation details

The permeability maps, utilized for dataset generation, correspond to Brugge benchmark. Each permeability field is highly-heterogeneous having many so-called fluvial channels.

We use a commercial numerical simulator to generate the dataset. The solver utilizes the Peaceman formula for sink/source terms approximation. We use this approximation in the inverse problem setting to calculate production rates.

In each computational scenario, the numerical model is initialized with reservoir pressure P_{init} and an arbitrary number of production wells (point sinks) equal to N_w located in the computational domain of size (X, Y). The wells start to operate at the initial time point with constant bottomhole pressure p_{bhp} chosen randomly from an uniform distribution between p_{bhp}^{min} and p_{bhp}^{max}]. Each simulation continued for 21 time steps T, and at each time step the pressure field P, well production rates q and botomhole pressures are saved. As a result, a dataset of permeability fields K of shape (N, X, Y) and their corresponding pressures P of shape (N, T, X, Y) was generated, where $N = 3450$ is the total number of simulations. Table A1 shows the quantitative properties of the generated dataset.

Table A1: Properties of generated dataset.

Model properties	Values
Number of simulations	3450
Number of well sections	20096
Number of cells (reservoir)	256 × 256 x 1
Number of cells (well section)	64 × 64 x 1
Reservoir size, m	12800 × 6400 x 2
Well section size, m	3200 × 1600 x 1
Number of wells in a scenario	1-16
Initial reservoir pressure, bars	300
Permeability range, mD	0 - 3922.25
Simulation duration, month	60
Number of time steps	21

Our local model works on a well-by-well basis, i.e. the model input is not the whole considered reservoir with shape (X, Y), but the section of shape (x, y) with a well placed in the center. We crop each spatial domain of shape (X, Y) from the generated dataset into sections with shape (N_w, x, y) where N_w is the number of wells in a scenario as shown in Figure 1. Note that P, K, X, Y stand for the whole reservoir's pressure, permeability and coordinates and small letters p, k, x, y stand for the same variables for a single section.

Appendix B. Physics-informed loss

To preserve the physical realism of the neural network solutions, we suggest physics-informed loss function consisting of three components: $\mathcal{L}_{data}$,

$\mathcal{L}_{PDE}$ $\mathcal{L}_{bc}$. $\mathcal{L}_{data}$ is data loss component which can be expressed as:

$$\mathcal{L}_{data} = ||\hat{p}(x, y, t) - p_{data}(x, y, t)||_2^2 , \quad \text{(A1)}$$

here $\hat{p}$ is the pressure field predicted by the neural network with weights θ, and p_{data} is the pressure field calculated in the numerical simulator. The brackets $|| \cdot ||_2^2$ denote the squared second norm. The data loss function is responsible for matching the predicted pressures $\hat{p}$ to the calculated p_{data} pressure fields. The second component of the physics-informed loss function is $\mathcal{L}_{PDE}$, which is the squared residual between the left and right parts of continuity equation for flow in porous media:

$$\mathcal{L}_{PDE} = ||\phi \frac{\partial(\rho(\hat{p}))}{\partial t} - div\,[\rho(\hat{p})u(k, \nabla p)] - q||_2^2 , \quad \text{(A2)}$$

where $\rho(\hat{p})$ is the fluid density obtained from the predicted pressure field $\hat{p}$, ϕ is reservoir porosity, $\partial/\partial t$ – time partial derivative, div is divergence operator, $u(k, \nabla p)$ – the flow velocity which is calculated using Darcy equation, q – point sink term. Minimization of this loss component helps to simulate the physics of fluid flow in the neural network predictions. The third component of our loss $\mathcal{L}_{bc}$ stands for the difference between the unsteady boundary condition $\hat{p}_{bc}$ which the network predicts and the boundary condition p_{bc}^{data} from the input:

$$\mathcal{L}_{bc} = ||\hat{p}_{bc} - p_{bc}^{data}||_2^2 , \quad \text{(A3)}$$

The loss component is introduced to have more control on prediction of the boundary condition during training.

The overall error function for the neural network training could be expressed as:

$$\mathcal{L} = \mathcal{L}_{data} + \lambda_1 \mathcal{L}_{PDE} + \lambda_2 \mathcal{L}_{bc} , \quad \text{(A4)}$$

where λ_1 and λ_2 are hyperparameters that were selected using Bayesian optimization.

Appendix C. Optimization of boundary condition

We propose the boundary conditions p_{bc} optimization procedure leveraging the fast inference of the trained neural network from Section 2.2. The optimization problem is minimization of loss function that contains a residual between true production rates from numerical simulator $q(p)$ and those derived from pressure field prediction $\hat{q}(\hat{p})$. To obtain well production rates during optimization $\hat{q}(\hat{p})$ we calculate them by means of Peaceman approximation.

The algorithm 1 shows the pseudocode of the boundary condition optimization.

Algorithm 1 Pressure Boundary Condition Optimization

```
Initialize pressure field guess P_init(X, Y, T) for the full reservoir.
Crop pressure boundary condition p_bc^0 around each well j from P_init(X, Y, T).
Optimize p_bc:
for i in 1, ..., n_iter do
    Predict pressure around wells p̂^i using neural network.
    Calculate production rate q_j^pred using Peaceman approximation for each well j.
    Calculate mean squared error: MSE(q_j^pred, q_j^true) where q_true is production rate from the numerical simulator.
    if |q_j^pred − q_j^true| < ε then
        break
    end if
    Renew p_bc^i using one of the optimization strategy.
end for
```

We tested two boundary condition optimization strategies. The first strategy updates the boundary condition utilizing weighted average of the form:

$$p_{bc}^{i+1} = \frac{p_{bc}^{i} + \alpha \hat{p}_{bc}^{i}}{1 + \alpha} , \quad \text{(A5)}$$

where p_{bc}^{i+1}, p_{bc}^{i} - new and current step boundary conditions, $\hat{p}_{bc}^{i}$ - boundary conditions predicted by neural network, α - learning rate which is hyperparameter, i - iteration step.

The second gradient-based approach leverages differentiability of the neural network. Given that we solve the slightly compressible single-phase flow problem, the second approach assumes that the boundary pressure at each time step could be effectively approximated in the form of:

$$p_{bc}(\hat{R}) = P_e - \frac{(P_e - p_{bhp})}{ln\left(\frac{R_k}{r_w}\right)} \times ln\left(\frac{R_k}{\hat{R}}\right) \times K; , \quad \text{(A6)}$$

where $p_{bc}(\hat{R})$ - pressure at a boundary cell; P_e, R_k - optimization parameters; p_{bhp} - bottomhole well pressure; r_w - well radius (constant for all wells); $\hat{R}$ - distance between the well and considered boundary cell; K - scaled to [0, 1] permeability at a boundary cell.

The boundary pressure parametrization strategy effectively decrease the dimensionality of the problem.

Appendix D. Results of global problem solution

Figure A1 outlines comparison of the ground truth pressure fields p_{data} and the pressure fields obtained by means of the optimization procedure $\hat{p}$. Figure 3 demonstrates the resulting production rates from optimization and the ground truth production rates.

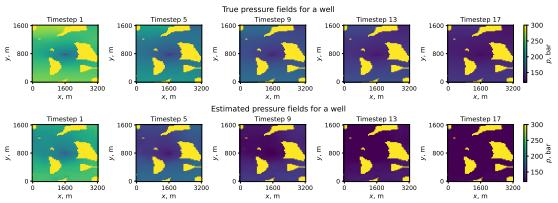

Fig. A1: Resulting pressure fields $\hat{p}$ for a well section using the second optimization strategy (gradient-based) and FNO as forward model. The upper row of pictures - numerical simulator output and the lower row - neural network estimation after boundary condition optimization.