# Deep Learning-Based Reservoir Simulation Proxy Models


Alexandre de Souza Jr.[1], Rafael F. V. C. Santos[1], Ramiro B. Willmersdorf[1], Silvana M. B. Afonso[1], Leonardo C. Oliveira[1], Bernardo Horowitz[1], Juan A. R. Tuerosa[2], Mateus G. Machado[2]

[1] *Department of Civil Engineering, Federal University of Pernambuco*
*Av. Prof. Moraes Rego, 1235 - Cidade Universitária, 50670-901, Recife, Pernambuco, Brazil*
*alexandre.souzajunior@ufpe.br, rafael.vcsantos@ufpe.br, ramiro.willmersdorf@ufpe.br, silvana.bastos@ufpe.br, leonardo.oliveira@ufpe.br, bernardo.horowitz@ufpe.br*
[2]*Department of Computer Center, Federal University of Pernambuco*
*Av. Prof. Moraes Rego, 1235 - Cidade Universitária, 50670-901, Recife, Pernambuco, Brazil*
*juan.tueros@ufpe.br, mgm4@cin.ufpe.br*



**Abstract.** The behavior of oil reservoirs, characterized by geophysical, geochemical, and geological properties, can be understood through the simulation of computational models, involving the construction of meshes of finite volume elements with equations derived from fundamental principles. However, the execution of multiple simulations for activities such as optimization and uncertainty assessment results in substantial computational costs. To overcome this challenge, proxy models are proposed, aiming to replace reservoir simulators with adequate precision. This work proposes the implementation of data-based proxies for reservoir simulators using Artificial Neural Networks (ANNs). This approach utilizes time series of well controls as inputs, generating responses for Bottom Hole Pressures (BHPs) and/or flow rates. In recent years, proxy models based on neural networks have been applied to obtain predictions of flows and/or pressures in reservoirs. For example, Recurrent Neural Networks (RNNs), specialized in handling sequential data, were used by [1] to predict water flows in the Xiluodu hydroelectric reservoir in China. Convolutional Neural Networks (CNNs), specialized in pattern recognition in images and videos, were also employed by [2] to predict pressures and flow rates of injector and producer wells, respectively. In the scope of this study, distinct neural network architectures were evaluated to predict outputs of a synthetic two-phase model with partial faults. Given the adoption of mixed controls, where producer wells are controlled by BHP and injector wells by flow rate, the use of Multi-head Neural Networks [3] was also investigated. This approach allows differentiated processing of input data, contributing to more robust and efficient learning. For each considered architecture, the RMSE and number of epochs required during training process was analyzed. The results indicate the CNN architecture proved to be effective with a lower error and reasonable training duration.



**Keywords:** Deep Learning; Neural Networks; Proxy Models; Reservoir Simulation.


## 1 Introduction

Oil reservoirs are underground formations where oil and gas accumulate, and understanding their behavior is crucial for efficient extraction. This is usually done through computational simulations, such as (CMG, 2022), that represent the reservoir in a simplified model, using a grid of finite-volume elements that incorporate petrophysical properties such as porosity and permeability. These simulations involve the numerical solution of equations derived from fundamental principles, such as thermodynamic equilibrium, mass conservation, heat transfer, and fluid flows in porous media.

To optimize oil and gas production, engineers perform various simulations considering different production and injection strategies. However, this process is computationally intensive and time-consuming, as exemplified by the Challenger Olympus case (Fonseca et al., 2018). As an alternative, proxy models based on historical production data have been developed to replace simulators with adequate accuracy. These models can be classified as physical, functional, or data-driven, with neural networks being an important example of the latter category.

In recent years, neural network-based proxy models have been used to predict flow rates and pressures in reservoirs. Zhang et al. (2019) applied recurrent neural networks to forecast water flow rates in hydropower reservoirs, while Alakeely and Horne (2020) used both recurrent and convolutional networks for predicting pressures and flow rates in wells. Jiang et al. (2023) and Kim and Durlofsky (2021) developed recurrent neural network-based models utilizing historical production data. This paper introduces an innovative approach by developing a black-box proxy model that provides time series outputs representing the impact of input controls across different scenarios, offering a comprehensive analysis of reservoir behavior for improved decision-making in reservoir management. The study focuses on proxies for biphasic reservoir simulators using time series data from injector and producer well controls, proposing mixed controls—bottom-hole pressure for producers and flow rates for injectors—to better reflect real-world management. It also explores multi-head neural networks that integrate RNNs and CNNs, creating robust models that capitalize on the strengths of each strategy.

# 2 Problem formulation

Deep Learning is a machine learning technique utilizing artificial neural networks with multiple processing layers. It became prominent in the 2000s due to improved hardware and abundant data, enabling solutions to complex problems like image classification and language models. A key concept in Deep Learning is the loss function, which assesses model quality and tracks training convergence. For regression tasks, the loss function, such as mean squared error (MSE), measures the difference between observed and predicted values by adjusting the neural network's weights and biases, as defined in Equation (1).

$$\underset{W,b}{Minimize\ E} = \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} (E^i) \tag{1}$$

Where $N_{train}$ is the number of samples used in the training process, $E$ is the objective function, $W$ and $b$ are the weights and biases, respectively. The training process involves iteratively adjusting the model weights to minimize the loss function. The classic optimization algorithm is gradient descent, which uses the derivative of the loss function to adjust the weights. The derivatives are obtained through automatic differentiation in TensorFlow Keras (Abadi et al., 2015), chosen for its user-friendly interface, facilitates Deep Learning for beginners. The learning rate is a crucial parameter affecting the step size for optimization, with higher rates speeding up convergence but risking instability, while lower rates ensure stability but slower training. Other hyperparameters, such as batch size and the number of epochs, also impact training. Batch size influences convergence, with options like stochastic or mini-batch methods, and multiple epochs are needed to iterate over the entire dataset, though too many epochs can lead to overfitting. Choosing the right neural network architecture—whether convolutional, recurrent, or fully connected—depends on the problem and may require experimentation. The next section will cover the architectures used in this study.

# 3 Methodology

Recently, the use of Artificial Neural Networks (ANN) in petroleum reservoir engineering has expanded, proving helpful in simulations and proxy creation (Alakeely, 2020; Kim, 2023). This section introduces the architectures applied to our problem, involving both Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), which will be combined to optimize proxy modeling. We will detail how RNNs and CNNs will be integrated into the overall architecture of the implemented networks, exploring their complementary capabilities. It is crucial to highlight that this is not a conventional time series prediction problem. Instead, the proxy acts as a black box, where inputs consist of well controls, and outputs correspond to the desired responses for the respective controls.

Recurrent Neural Networks (RNNs) are designed for sequential data like time series or natural language, maintaining a "memory" of past observations to capture temporal context (Brownlee, 2018). This feature is useful for predicting well flow rates, where changes in control impact both immediate and future flows. Each RNN unit, or recurrent cell, holds and transmits historical input information through hidden states. Long Short-Term Memory (LSTM) networks improve upon RNNs by addressing vanishing gradients with gated recurrent cells and cell states, as shown in Figure 1. LSTMs use forget, input, and output gates to manage memory flow, retaining relevant long-term and short-term information. Despite their robustness and performance, LSTMs are computationally intensive and lead to longer training times.

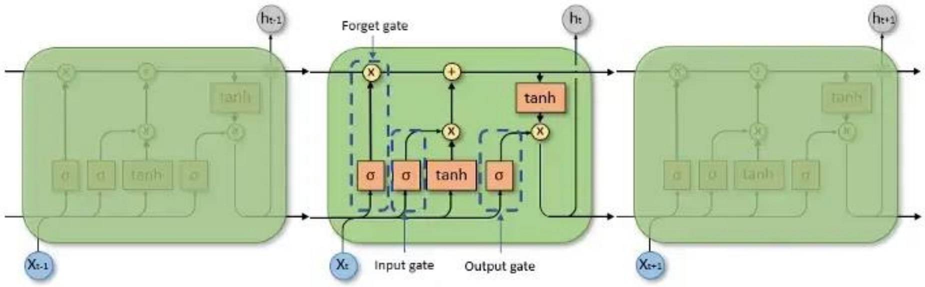



Figure 1. Example of a LSTM layer with three recurrent cells.

Convolutional Neural Networks (CNNs), introduced by LeCun (1989), are more efficient than traditional neural networks in recognizing patterns in images and videos (Géron, 2023). By replacing dense layers with convolutional layers, CNNs reduce the number of parameters and improve performance in both time and space (Chollet, 2021). They use hierarchical layers where each layer detects progressively complex patterns, with filters or kernels highlighting recognized features. Techniques such as padding, stride, and pooling further optimize the network. Figure 2 illustrates a CNN architecture. Recent studies have applied LSTM and GRU networks to predict reservoir outputs, such as water flow rates in hydropower reservoirs (Zhang et al., 2019) and pressures and flow rates in wells (Alakeely and Horne, 2020). Kim and Durlofsky (2021) developed an LSTM-based proxy model for well-control optimization, using bottom-hole pressures as inputs and water and oil flow rates as outputs, with a suggested architecture of 200 LSTM cells. Jiang et al. (2023) presented an interpretable RNN incorporating material balance for predicting reservoir behavior, integrating physical knowledge into model training.

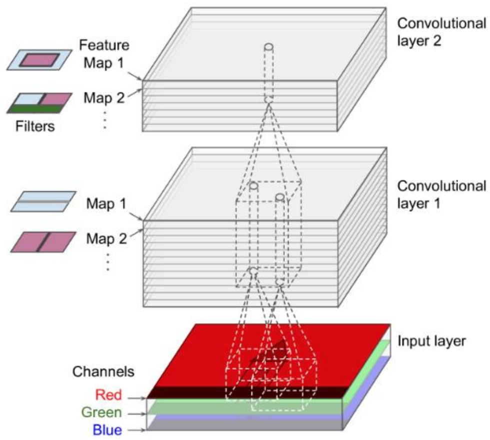



Figure 2. Example of two-layer convolutional architecture applied to an input layer.

In neural network architectures, "multi-head" refers to a structure where several sub-networks process input data individually, and their outputs are subsequently combined and processed by the rest of the network. This approach, introduced by (Ibrahim et al., 2021), allows each sub-network to specialize in different aspects of the input data, enhancing the network's ability to learn complex data representations and improving its generalization capability. In this study, we propose five neural network architectures, two of which are multi-headed to specialize in different input control types: BHP in producer wells and water flow rate in injector wells. The initial architecture consists of two CNN layers for the same input characteristics, followed by dense layers. The second architecture comprises two LSTM layers for time series prediction. Another proposed architecture hybridizes these two models sequentially. Additionally, we explore two more architectures with independent convolutional models for each input type using a multi-head approach: one consisting of two CNN layers only and another with an LSTM layer in sequence. Figure 3 illustrates these architectures.

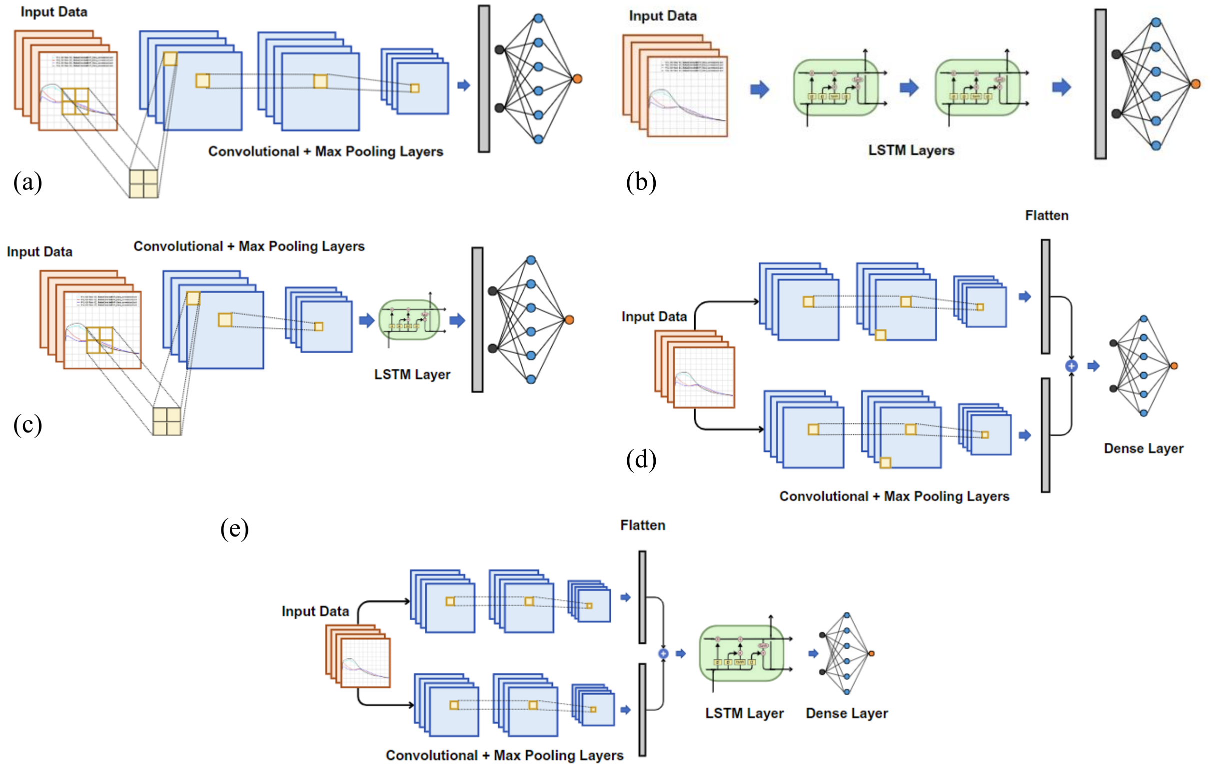



Figure 3. Proposed neural network architecture models: (a) CNN, (b) LSTM, (c) CNN-LSTM, (d) Multi-head CNN, and (e) Multi-head CNN-LSTM.

# 4 Application

This study utilized a synthetic biphasic model (Cao et al., 2014) to validate various proxy strategies using RNNs and CNNs. The reservoir model, with two internal partial faults, features 5 layers of 33x33 active cells each. Horizontal permeability is constant at 270 mD, while vertical permeability is 10% of horizontal permeability. The average reservoir pressure is 500 psi, with an initial water saturation of 0.1 (Figure 4(a)) and porosity of 0.2. The model includes 4 producers and 5 injectors, with fault cells having zero porosity, transmissibility, and permeability (Figure 4(b)).

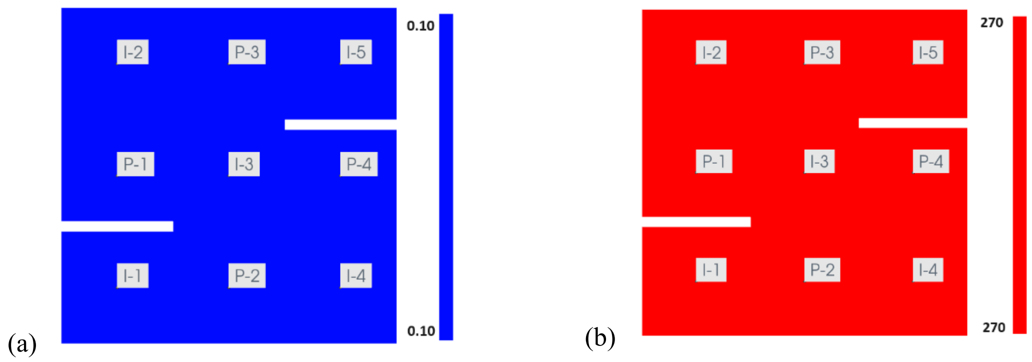
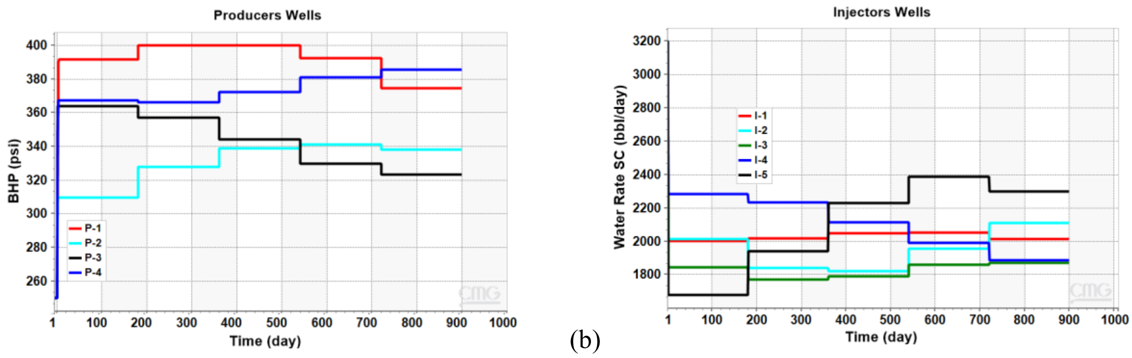


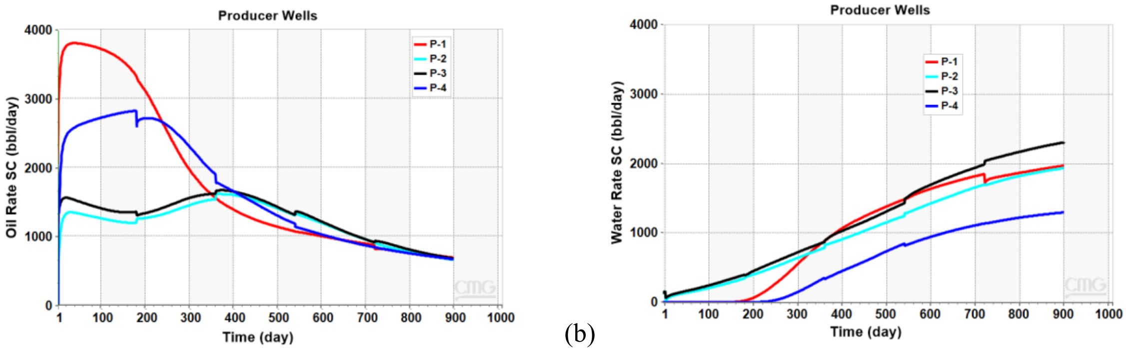

Figure 4. Model of the first case study: (a) initial water saturation and (b) permeability.

Production history data were generated using the IMEX reservoir simulator (CMG, 2022), yielding 366 samples. Well controls, changing every 180 days over 900 days with 5 cycles per well (Figure 5), varied from 300 to 400 psi for producers and 505 to 600 psi for injectors. A correlated controls strategy with a Gaussian covariance function ensured smooth, consistent perturbations. Simulator responses, such as oil and water production rates and water injection or pressure, were recorded every 30 days (Figures 6 and 7). The data were divided into 70% training, 20% validation, and 10% testing. Table 1 shows the primary hyperparameters for RNN and CNN architectures. Training involved up to 500 epochs, a batch size of 32, and the Adam optimizer (Kingma and Ba, 2015) with a learning rate of $10^{-3}$. Early stopping with a patience of 10 epochs and a minimum error variation of $10^{-4}$ was also used.




(a) (b)

Figure 5. Correlated BHP controls for (a) producing wells and (b) injection wells.




(a) (b)

Figure 6. Flow rates in producing wells, with correlated controls: (a) oil rate and (b) water rate.

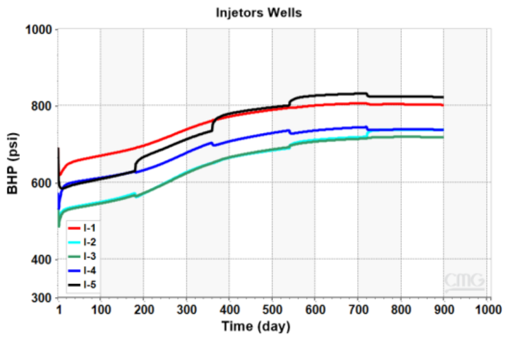


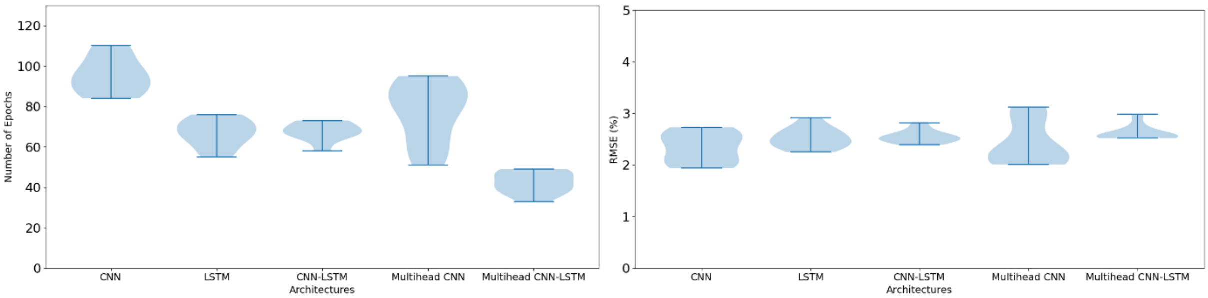

Figure 7. Water flow rate in injection wells, with correlated controls.

Table 1. Hyper parameters used to define the proposed neural network architectures.

| Layer | Hiperparameter | Value |
|---|---|---|
| Convolutional | Filters | 32 |
| | Kernels | 10, 13 |
| | Activation Function | ReLU |
| Recurrent | Number of LSTM Cells | 200 |
| | Activation Function | ReLU |

# 5 Results

The obtained results reveal intriguing insights into the comparative analysis among the different neural network architectures considered in this study. Figure 8 illustrates violin plots for 10 test executions of each architecture, displaying the number of epochs during training and the RMSE (%), respectively. These visualizations provide a comprehensive view of the performance variability and stability across different models.




(a) (b)

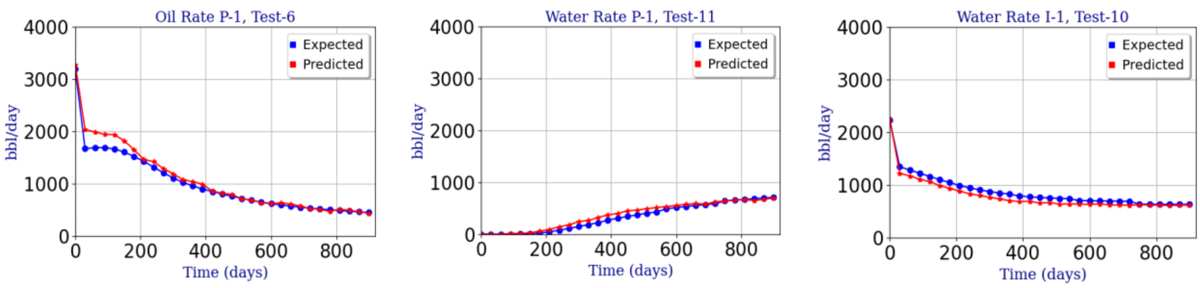
Figure 8. (a) Number of epochs and (b) RMSE (%) during training for 10 executions, with correlated controls.

Table 2 summarizes the results obtained using 30 timesteps across various architectures with a consistent sample size. Among the architectures evaluated, the basic CNN architecture demonstrated notable efficiency with an average RMSE of 2.33% and 95 epochs. The LSTM architecture, while having a higher number of trainable parameters (491,413), required fewer epochs (67) but had a slightly higher average RMSE of 2.52%. The CNN-LSTM hybrid model, with 478,549 trainable parameters, performed similarly to the LSTM with an average RMSE of 2.55% over 68 epochs. The Multi-head CNN showed a moderate performance, with 174,835 trainable parameters and an average RMSE of 2.41% over 77 epochs, balancing complexity and efficiency. Interestingly, the Multi-head CNN combined with LSTM, although having a reduced average epoch count of 42, resulted in the highest average RMSE of 2.62%.

Table 2. Results obtained for the synthetic two-phase model proxies.

| Architecture | N. Trainable Params. | Average N. Epochs | Average RMSE (%) |
|---|---|---|---|
| CNN | 204,339 | 95 | 2.33 |
| LSTM | 491,413 | 67 | 2.52 |
| CNN-LSTM | 478,549 | 68 | 2.55 |
| Multi-head CNN | 174,835 | 77 | 2.41 |
| Multi-head CNN + LSTM | 449,045 | 42 | 2.62 |

The CNN architecture excelled in RMSE and training duration, accurately predicting production and injection rates. Figure 10 shows the forecasted oil and water production rates and injection rates, demonstrating the CNN model's strong alignment with actual data and its effectiveness in capturing complex reservoir dynamics.




(a) (b) (c)

Figure 10. Flow forecasts: (a) oil production from producer P-1, (b) water production from producer P-1 and (c) water injected into I-1, using correlated control samples.

# 6 Conclusions

This study introduces a novel black-box proxy model for reservoir simulation, outputting time series data on the impact of input controls under various scenarios. It improves reservoir management by analyzing biphasic reservoirs (oil and water phases) with mixed controls reflecting real-world practices. The model uses multi-head neural networks, including RNNs and CNNs, to enhance robustness. Results show that while CNNs, typically used in computer vision, achieved a lower RMSE and reasonable training time, the Multi-head CNN and CNN with LSTM offered a balance between epochs and model complexity. These insights highlight the need to balance performance metrics and model complexity in selecting reservoir simulation proxy models. Future work may focus on optimizing hyperparameters and applying the models to different reservoirs and control strategies.

**Acknowledgements.** We gratefully acknowledge the support provided by PETROBRAS, ANP, FINEP, PRH, EMBRAPII, FACEPE, CNPq, and CAPES, which has been instrumental in the successful execution of this work.

**Authorship statement.** The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

# References

[1] M. Abadi, A. Agarwal, and P. Barham. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. Disponível em: http://download.tensorflow.org/paper/whitepaper2015.pdf, 2015.
[2] A. Alakeely and R. N. Horne. *Simulating the behavior of reservoirs with convolutional and recurrent neural networks*. SPE Reservoir Evaluation & Engineering, vol. 23, no. 3, SPE-201193-PA, pp. 992-1005, 2020.
[3] J. Brownlee. *Deep learning for time series forecasting*, Machine Learning Mastery, 2018.
[4] M. A. Cardoso. *Development and Application of Reduced-Order Modeling Procedures for Reservoir Simulation*, 2009.
[5] F. Cao. *Development of a two-phase flow coupled capacitance resistance model. PhD Dissertation*. The University of Texas at Austin, USA, 2014.
[6] F. Chollet. *Deep learning with python*. Manning Publications Co., 2021.
[7] Computer Modeling Group LTD. *IMEX: User's Guide*. Calgary, Canada, 2022.
[8] A. Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc., 2023.
[9] D. P. Kingma and J. Ba. *A Method for Stochastic Optimization.* International Conference on Learning Representations (ICLR), 2015.
[10] Y. LeCun et al. *Backpropagation Applied to Handwritten Zip Code Recognition.* Neural Computation, vol. 1, no. 4, pp. 541-551, 1989.
[11] B. Ibrahim and L. A. Rabelo. *Deep Learning Approach for Peak Load Forecasting: A Case Study on Panama*. Energies, vol. 14, p. 3039, 2021.
[12] Y. Jiang, W. Shen, H. Zhang, K. Zhang, J. Wang, and L. Zhang. *An interpretable recurrent neural network for waterflooding reservoir flow disequilibrium analysis*. Water, vol. 15, pp. 623-639, 2023.
[13] Y. D. Kim and L. J. Durlofsky. *A recurrent neural network–based proxy model for well-control optimization with nonlinear output constraints.* SPE Journal, vol. 26, no. 4, pp. 1837-1857, 2021.
[14] Y. D. Kim and L. J. Durlofsky. *Convolutional - recurrent neural network proxy for robust optimization and closed-loop reservoir management*. Computers & Geosciences, vol. 27, no. 2, pp. 179-202, 2023.
[15] D. Zhang, Q. Peng, J. Lin, D. Wang, X. Liu, and J. Zhuang. *Simulating reservoir operation using a recurrent neural network algorithm*. Water, vol. 11, no. 4, p. 865, 2019.