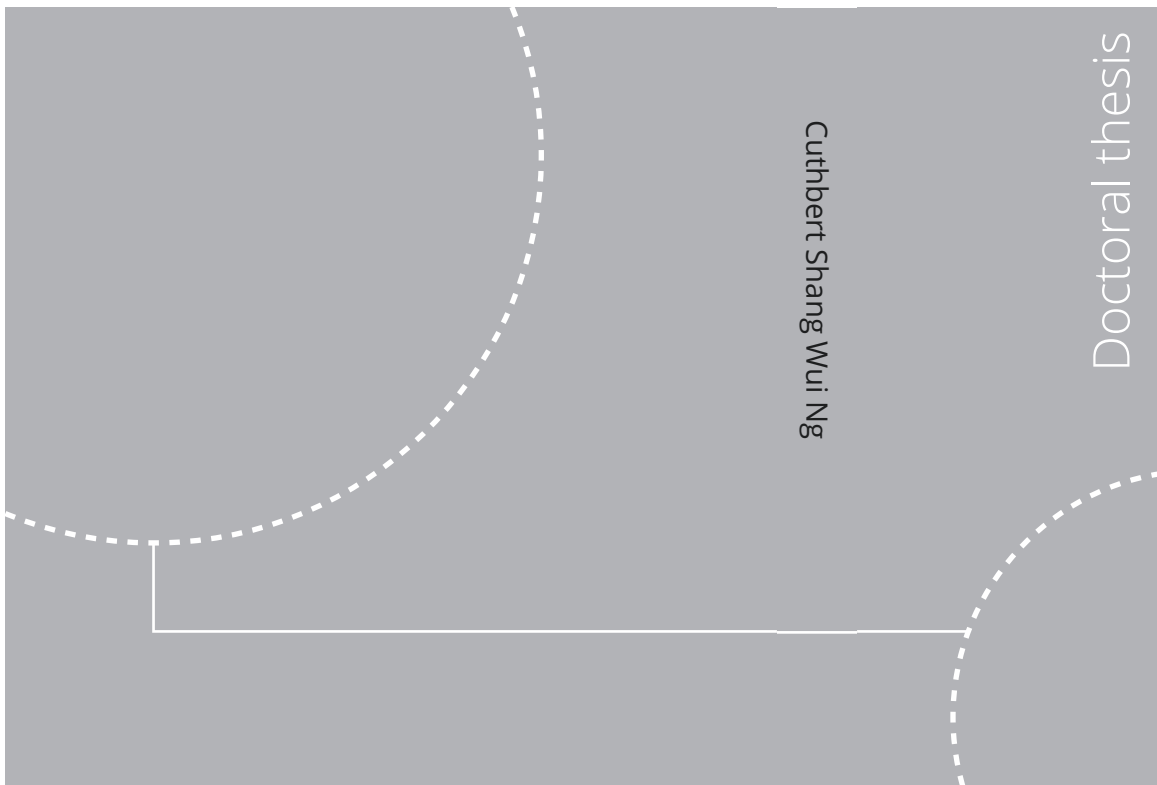


ISBN 978-82-326-7070-3 (printed ver.)  
ISBN 978-82-326-7069-7 (electronic ver.)  
ISSN 1503-8181 (printed ver.)  
ISSN 2703-8084 (electronic ver.)



Doctoral theses at NTNU, 2023:183

Cuthbert Shang Wui Ng

# Data-Driven Reservoir Modeling: Application of Proxy Models in Reservoir Management

Doctoral theses at NTNU, 2023:183

**NTNU**  
Norwegian University of  
Science and Technology  
Thesis for the degree of  
Philosophiae Doctor  
Faculty of Engineering  
Department of Geoscience and Petroleum



Cuthbert Shang Wui Ng

# Data-Driven Reservoir Modeling: Application of Proxy Models in Reservoir Management

Thesis for the degree of Philosophiae Doctor

Trondheim, May 2023

Norwegian University of Science and Technology  
Faculty of Engineering  
Department of Geoscience and Petroleum



Norwegian University of  
Science and Technology

**NTNU**

Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Engineering  
Department of Geoscience and Petroleum

© Cuthbert Shang Wui Ng

ISBN 978-82-326-7070-3 (printed ver.)  
ISBN 978-82-326-7069-7 (electronic ver.)  
ISSN 1503-8181 (printed ver.)  
ISSN 2703-8084 (electronic ver.)

Doctoral theses at NTNU, 2023:183



Printed by Skipnes Kommunikasjon AS

# Preface

This thesis is written to fulfill the requirement for the Ph.D. in Petroleum Engineering at the Department of Geoscience and Petroleum, Norwegian University of Science and Technology (NTNU). The research work discussed in this thesis was conducted for the past 3 years, from November 2019 until November 2022. It is a part of BRU21 – NTNU Research and Innovation Program on Digital Automations for the Oil and Gas Industry under the program area of Reservoir Management and Production Optimization. This research is carried out under the supervision of Associate Professor Dr. Ashkan Jahanbani Ghahfarokhi from the Department of Geoscience and Petroleum and the co-supervision of Professor Dr. Lars Struen Imsland from the Department of Engineering Cybernetics at NTNU. The main goal of this study is to formulate a fundamental methodology that can be implemented to build data-driven models with the aid of machine learning techniques to solve reservoir management issues. Therefore, most of the case studies presented are discussed in the context of petroleum reservoir engineering. This doctorate thesis is prepared in paper-based format in which 8 research journal articles are compiled. It consists of 4 chapters that aim at providing clear ideas about some important concepts to the readers before perusing the journal papers.

# Acknowledgments

This Ph.D. thesis is the result of a collective effort from different individuals and it will not be successfully completed without any of them. First and foremost, I would like to express my utmost gratitude to my supervisor, Dr. Ashkan Jahanbani Ghahfarokhi for his continuous guidance and support throughout my Ph.D. studies. Ashkan has always been passionate to offer me help whenever I need it. He has put a lot of time and effort to clear my doubts, give insightful comments, and keep track of the progress of my work. Also, I would like to thank my co-supervisor, Dr. Lars Struen Imsland for his dedication to discuss my work and providing me with useful suggestions. Many thanks to Dr. Menad Nait Amar too for assisting me along, especially in the beginning phase of this Ph.D. research. It has been a fabulous journey for me to have learned from and worked with three of them.

Besides that, I am grateful to the other two co-authors of our journal articles, Dr. Ole Torsæter and Mr. Wilson Wiranda for their contributions. I would also like to acknowledge the support given by many individuals from the Department of Geoscience and Petroleum at NTNU, especially the Reservoir Engineering and Petrophysics group and the experts from the BRU21 program. Without them, I will not be able to accomplish this milestone. Moreover, I owe my gratefulness to my friends who have been helping, motivating, entertaining, and accompanying me for occasional chats and social activities/during festive seasons. The name list is quite long to be mentioned. Nonetheless, I am certain that you know who you are. Thank you for crossing my paths and inspiring me. Being able to know you is one of my greatest blessings in life.

Certainly, I would like to take this moment to express my special thanks to my parents, siblings, in-laws, and niece. Thank you for your unconditional love and support. You all have been my main source of motivation. To the readers, I appreciate your time in reading this thesis. Apart from the technical details, I would love to share with you one of my favorite quotes that goes as shown below.

“Don’t get carried away when you are in good times.  
But also, don’t give up easily when you are in adversity.”

(顺境时别得意忘形，逆境时别轻言放弃。)

Last but not least,

**정신차려! Semangat! 화이팅! Tusen takk og ha en fin dag!**

Yours faithfully,  
Cuthbert Shang Wui Ng  
Trondheim, 26 January 2023

## Special Dedication

“In nomine Patris, et Filii, et Spiritus Sancti. Amen.  
GLORIA PATRI, et Filio, et Spiritui Sancto. Sicut erat in  
principio, et nunc, et semper, et in saecula saeculorum.  
Amen.”✠

# Table of Contents

<b>Preface</b> .....	i
<b>Acknowledgments</b> .....	ii
<b>Special Dedication</b> .....	iii
<b>Table of Contents</b> .....	iv
<b>List of Papers</b> .....	vi
<b>List of Figures</b> .....	vii
<b>Abbreviation</b> .....	viii
<b>Abstract</b> .....	1
<b>1 Introduction</b> .....	2
<b>2 Background of Concepts</b> .....	4
<b>2.1 Reservoir Management</b> .....	4
<b>2.2 Waterflooding</b> .....	6
<b>2.3 Numerical Reservoir Simulation</b> .....	7
<b>2.4 Data Science</b> .....	8
<b>2.4.1 Data-Driven Modeling Techniques</b> .....	8
<b>2.5 Proxy Models as Replica of Numerical Reservoir Simulation</b> .....	11
<b>2.5.1 Source of Database</b> .....	11
<b>2.5.2 Sampling Technique</b> .....	12
<b>2.5.3 Reservoir Case Study</b> .....	13
<b>2.6 Optimization</b> .....	15
<b>2.7 Decision Analysis</b> .....	17
<b>3 Contributions and Summaries of Papers</b> .....	19
<b>4 Concluding Remarks and Recommendations</b> .....	25
<b>Bibliography</b> .....	28
<b>Collection of Papers</b> .....	35
<b>Paper 1</b> .....	36
<b>Paper 2</b> .....	61
<b>Paper 3</b> .....	94
<b>Paper 4</b> .....	120

<b>Paper 5</b> .....	129
<b>Paper 6</b> .....	143
<b>Paper 7</b> .....	170
<b>Paper 8</b> .....	184



# List of Papers

**Paper 1:** *A Survey on the Application of Machine Learning and Metaheuristic Algorithms for Intelligent Proxy Modeling in Reservoir Simulation.*

Cuthbert Shang Wui Ng, Menad Nait Amar, Ashkan Jahanbani Ghahfarokhi, and Lars Struen Imsland.

Published in **Computers and Chemical Engineering**. Volume 170, February 2023, 108107

**Paper 2:** *Smart Proxy Modeling of a Fractured Reservoir Model for Production Optimization: Implementation of Metaheuristic Algorithm and Probabilistic Application.*

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar, and Ole Torsæter.

Published in **Natural Resources Research**. Volume 30, 2431-2462 (2021)

**Paper 3:** *Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization.*

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, and Menad Nait Amar.

Published in **Journal of Petroleum Exploration and Production Technology**. Volume 11, 3103-3127 (2021)

**Paper 4:** *Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm.*

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, and Menad Nait Amar.

Published in **Petroleum**. Volume 9, Issue 1, March 2023

**Paper 5:** *Adaptive Proxy-based Robust Production Optimization with Multilayer Perceptron.*

Cuthbert Shang Wui Ng and Ashkan Jahanbani Ghahfarokhi.

Published in **Applied Computing and Geosciences**. Volume 16, December 2022, 100103

**Paper 6:** *Fast Well Control Optimization with Two-Stage Proxy Modeling.*

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, and Wilson Wiranda.

Published in **Energies**. Volume 16, Issue 7, 3269, April 2023

**Paper 7:** *Well production forecast in Volve field: Application of rigorous machine learning techniques and metaheuristic algorithm.*

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, and Menad Nait Amar.

Published in **Journal of Petroleum Science and Engineering**. Volume 208, Part B, January 2022, 109468

**Paper 8:** *Optimizing initiation time of waterflooding under geological uncertainties with Value of Information: Application of simulation-regression approach.*

Cuthbert Shang Wui Ng and Ashkan Jahanbani Ghahfarokhi.

Published in **Journal of Petroleum Science and Engineering**. Volume 220, Part A, January 2023, 111166

# List of Figures

Figure 1. Different procedures of Reservoir Development Plan. Adapted from (Satter and Iqbal, 2016b).....	5
Figure 2. Reservoir Management. Adapted from (Satter et al., 1998).....	6
Figure 3. Schematic of Typical Artificial Neural Network. Adapted from (Mohaghegh, 2000).....	10
Figure 4. The architecture of the Egg Model with its well configurations (One of the realizations).....	14
Figure 5. The architecture of the UNISIM-I-D Model with its well configurations..	14
Figure 6. The architecture of the OLYMPUS Model with its well configurations (One of the realizations).....	15

# Abbreviation

Adam	-	Adaptive Moment Estimation
AI	-	Artificial Intelligence
ANN	-	Artificial Neural Network
BRU21	-	Better Resource Utilization in the 21 <sup>st</sup> century
CCS	-	Carbon, Capture, and Storage
CCUS	-	Carbon, Capture, Utilization, and Storage
CF	-	Cash Flow
DA	-	Decision Analysis
DCA	-	Decline Curve Analysis
DM	-	Decision-Making
EOR	-	Enhanced Oil Recovery
FNN	-	Feedforward Neural Network
GA	-	Genetic Algorithm
GBR	-	Gradient Boosting Regressor
GRU	-	Gated Recurrent Unit
GP	-	Genetic Programming
GPR	-	Gaussian Process Regression
GWO	-	Grey Wolf Optimization
HSS	-	Hammersley Sequence Sampling
HS	-	Hydrogen Storage
HPO	-	Hyperparameter Optimization
k-NN	-	k-Nearest Neighbor
LHS	-	Latin Hypercube Sampling
LMA	-	Levenberg-Marquardt Algorithm
LSTM	-	Long Short-Term Memory
LSM	-	Least-Square Monte Carlo
MBE	-	Material Balance Equation
ML	-	Machine Learning
MLP	-	Multilayer Perceptron
MPO	-	Model Parameter Optimization
NRS	-	Numerical Reservoir Simulation
NPV	-	Net Present Value
PSO	-	Particle Swarm Optimization
PVT	-	Pressure, Volume, and Temperature
RF	-	Random Forest
RM	-	Reservoir Management
RMSProp	-	Root Mean Squares Propagation

RNN	-	Recurrent Neural Network
RSM	-	Response Surface Model
SGD	-	Stochastic Gradient Descent
SPM	-	Smart Proxy Models
SRDM	-	Sequential Reservoir Decision-Making
SSS	-	Sobol Sequence Sampling
STEM	-	Science, Technology, Engineering, and Mathematics
SVM	-	Support Vector Machine
SVR	-	Support Vector Regression
TDM	-	Top-Down Models
VOI	-	Value of Information
WAG	-	Water-Alternating-Gas

# Abstract

This Ph.D. thesis consists of 8 papers that summarize the main contents of the research work done over the past 3 years. Due to the ability of machine learning (ML) in capturing high nonlinearity, the thesis mainly touches upon its use in data-driven modeling to provide aids in reservoir management. Data-driven models are referred to as “proxy models” as they act on behalf of the reservoir simulator. Proxy models are deemed practically useful if they can provide fast and desirably accurate solutions.

In this thesis, a survey on the use of ML and metaheuristic algorithms in developing proxy models for reservoir simulation was presented to enlighten the readers. We also explained the methodology of proxy modeling with an associated case study, viz. the waterflooding process. The proxy modeling of a synthetic reservoir model was first formulated on which further works were done as improvements. These improvements, including the integration of sampling techniques and the use of more complex reservoir models, proposed the fundamentals of the proxy modeling methodology in more realistic application cases. Upon the completion of these steps, adaptive sampling and retraining were applied to address the geological uncertainties. Also, two classes of proxy modeling, namely local and global proxy modeling, were implemented to handle optimization problems with higher dimensions.

Furthermore, additional works were illustrated to provide a scaffold for the maturity of the methodology. These works pertain to research on applying ML methods in predictive modeling and a decision analysis framework. One of them illustrated the establishment of ML-based predictive models with splendid predictability. The work also includes a discussion about the steps of predictive modeling for well production forecast based on real field data. The other one displayed coupling of ML with a mathematical algorithm to approximate the Value of Information that was used for optimization under uncertainties. These studies are not only related to those described earlier but also illustrate the robust application of machine learning. In summary, this research project portrayed the establishment of a methodology that could yield proxy models to facilitate the resolution of reservoir management issues with less computational efforts as compared with reservoir simulator without compromising the accuracy.

# Chapter 1

## Introduction

This Ph.D. thesis is a summary of the results obtained from research work done over the past 3 years. This research work is part of the BRU21 program that aims at generating a value chain throughout the oil and gas industry by providing digital and automation solutions. The title of this thesis is *Data-Driven Reservoir Modeling: Application of Proxy Models in Reservoir Management* under the program area of Reservoir Management and Production Optimization of BRU21. As the title implies, the overall goal of the research work is to outline a framework of methodology that offers an alternative solution to reservoir management (RM). This alternative solution is targeted to be fast and within a good level of accuracy. Therefore, this solution (using machine learning) can provide convenience especially if the RM plan needs to be updated quite frequently. Apart from this, this research places a certain degree of emphasis upon the investigation of the use of machine learning in predictive modeling and resolving sequential decision problems. The relevant details will be uncovered later.

With the rapid development of digitalization in STEM (science, technology, engineering, and mathematics), many researchers and engineers have begun exploring and researching machine learning as one of their research domains. This motivates the employment of machine learning, an epitome of data-driven methods, as an alternative approach to resolve any sophisticated engineering problem. In this case, solving optimization problems in reservoir management generally have a high computational footprint. Data-driven modeling was thereafter suggested to provide a computationally cheap and desirably accurate solution. Therefore, a term called “proxy modeling” has been coined to represent these data-driven models. In the context of reservoir simulation, these proxy models act on behalf of the reservoir simulators to yield fast solutions.

In **Paper 1**, a survey of the use of ML and metaheuristic algorithms in building machine learning-based proxy models for reservoir simulation was conducted. Machine learning-based proxy model was termed intelligent proxy model in the paper. Numerous literature that considered the implementation of machine learning and metaheuristic algorithms in intelligent proxy modeling in different applications of reservoir simulations were discussed. For the pertinent details, refer to **Paper 1**. Then, **Paper 2** generally illustrated the proxy modeling of a fractured reservoir model, which was considered a plain-vanilla case, on which further works were done as parts of **Papers 3 to 6** to achieve higher maturity. In these four papers, the sampling techniques were demonstrated to be integrated into proxy modeling to do well control optimization with a relatively higher level of complexity as compared to **Paper 2**. Besides that, in **Papers 3 and 4**, a more sophisticated reservoir model was

## Chapter 1: Introduction

utilized as compared to **Paper 2**. Metaheuristic algorithms were also incorporated in the whole framework to do the well control optimization.

Upon developing a general workflow of methodology that considers the training of proxy models and optimization, small steps of improvements were performed to increase its applicability. One of the refinements done was to consider geological uncertainty in which adaptive sampling was employed to modify the training database and retraining was conducted iteratively. Refer to **Paper 5** for the relevant information. In addition, under the circumstance of a more realistic reservoir model and optimization problem with higher dimensions, two classes of proxy modeling were proposed in which the initially established proxy models were coupled with optimization algorithms to generate a new database to develop new proxy models. **Paper 6** presents the respective details.

It is of great importance to remind the readers that proxy modeling can be used for predictive modeling. This is because proxy models need to possess satisfying prediction performance to be ready for further use, including optimization. So, in this Ph.D. research, we are also motivated to further investigate the use of machine learning to leverage its potential in creating predictive models that can be insightful to the overall methodology of proxy modeling. **Paper 7** is the product of this investigation in which, the developed predictive models were trained based on real field data by using derivative-based and derivative-free algorithms for in-depth comparative studies. Moreover, another intriguing task was done to harvest the potential of ML for the analysis of Value of Information (VOI: an important decision analysis tool to resolve sequential decision problems). VOI served as a guidance to identify the optimal time to initiate waterflooding in different geological settings of a benchmark reservoir model. **Paper 8** consists of the corresponding explanation and details.

In most of the tasks presented in this thesis, considering a practical illustration of the methodology of proxy modeling, we selected waterflooding optimization case study problem that is primarily associated with reservoir management. By doing so, we hope that reservoir engineers and researchers can be inspired to fathom the usefulness of machine learning in reservoir engineering. Despite having refinements throughout this Ph.D. journey, limitations were explained with possible recommendations to offer insights to other engineers and researchers to explore this topic to a greater extent.

After this brief introduction, Chapter 2 discusses the important concepts and theories that the readers must grasp before reading the papers compiled. This discussion aims at providing sufficient fundamentals before diving into the details. Chapter 3 briefs the summaries of each of the papers compiled. This chapter enables the readers to have an entire perspective of the development of this thesis. Additionally, an understanding of the context of each paper can be established by referring to this chapter. Chapter 4 summarizes the main findings and concluding remarks about this thesis. Several proposals are also mentioned in this chapter. Finally, since this thesis is paper collection-based, all the relevant papers that contribute to this thesis are compiled at the end.

# Chapter 2

## Background of Concepts

This chapter aims at briefing the readers on the background of some concepts, which have been implemented to scaffold the framework of the methodology proposed in this work. Having a profound understanding of these concepts enables the readers to have a good rhythm of perusal of this thesis. Generally, this chapter begins with a general introduction to Reservoir Management. Thereafter, there is an explanation about other technological toolboxes, including numerical reservoir simulation and data-driven models, applied to resolve reservoir engineering issues. It also outlines other topics, such as data science, optimization, and decision analysis, that are used to facilitate the foundation of the framework.

### 2.1 Reservoir Management

Reservoir Engineering is a field that implements scientific knowledge to understand fluid flow through porous media and the physical properties of these media (**Dake, 1978**). Understanding the porous media enables the reservoir engineers to formulate a development plan to produce the reservoir fluids more effectively and economically. To accurately decipher the reservoir, we require multidisciplinary knowledge, including (but not limited to) fundamental physics and chemistry, thermodynamics, geology, and applied mathematics (**Craft et al., 1991; Satter and Iqbal, 2016a**).

The combination of these knowledge domains yields different technological toolboxes to be utilized. These toolboxes are transient well test, log analysis, conventional core analysis, computed tomography scan, fluid analysis, reservoir simulation, decline curve analysis, material balance, stream tube model, geo-statistics, enhanced oil recovery (EOR) technology and screening, and so forth (**Satter et al., 1998; Thakur, 1996**). Having these toolboxes facilitates the reservoir engineers to perform their responsibilities, including interpretation and integration of a large amount of reservoir data, characterization of the geological properties, estimation of reserves, forecasting of production, economic analysis, visualization of reservoir fluid flow, and PVT analysis of reservoir fluid samples. The job scope of reservoir engineers is integral to field development planning as cost-effective reservoir depletion schemes can be recommended to optimize the recovery.

Concerning this, the definition of Reservoir Management (RM) is established not only to clearly reflect the responsibilities of reservoir engineers but also to illustrate the general approach that is



## Chapter 2: Background of Concepts

implemented to smoothen the process of managing a reservoir. RM shares different definitions by different authors (**Robertson, 1989; Thakur, 1996; Wiggins and Startzman, 1998**). However, its definition generally gravitates to the application of state-of-the-art technology, economic and labor resources to maximize the profit through the production of fluids from a reservoir and simultaneously minimize the operating and capital costs, starting from discovery phase to abandonment. As discussed in the literature (**Satter and Iqbal, 2016b**), the approach to RM is by formulating a strategy to achieve a purpose. This strategy is then accomplished by developing a plan, implementing, monitoring, and evaluating the results. In this aspect, the details of developing a plan for a reservoir are made up of different procedures as shown in **Figure 1**. As the plan is implemented, monitoring step ensures the plan is performed accordingly. The results of the plan would then be gathered. Upon assessing the results, if the reservoir engineers find them unsatisfactory, the revision of the plan is done (also known as updating step). This process is dynamic and as additional data is acquired, the RM plan is further enhanced with new corresponding changes. RM plan ought to be periodically updated to lead to better results.

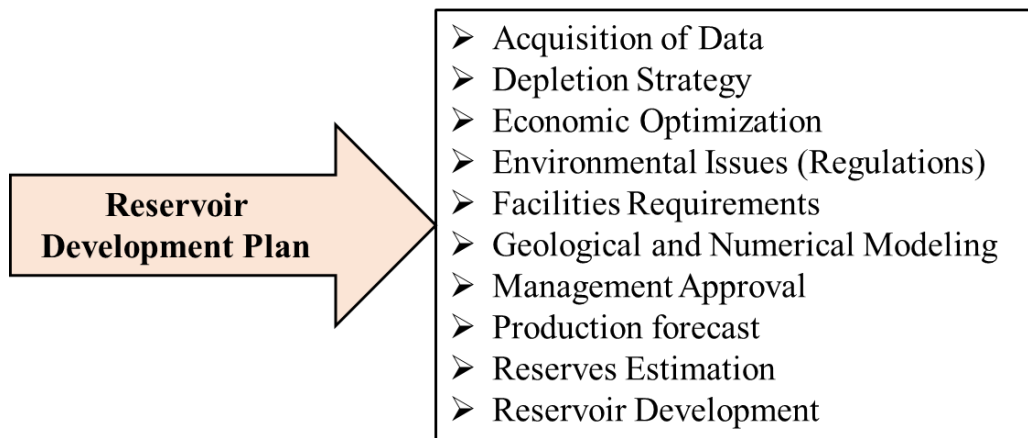


Figure 1. Different procedures of Reservoir Development Plan. Adapted from (**Satter and Iqbal, 2016b**).

Some elements of RM include production optimization, history matching, uncertainty analysis, production prediction, etc. The readers are referred to **Figure 2** for other elements of RM. RM also involves selecting available options, for instance, whether to proceed with an EOR operation or not. This option selection process is simply decision-making (DM) for which the details are explained in **Section 2.7**. In addition, the global effort of carbon emission reduction and energy transition has tweaked the definition of RM in which Carbon, Capture, and Storage (CCS) is considered a “new” element of RM.

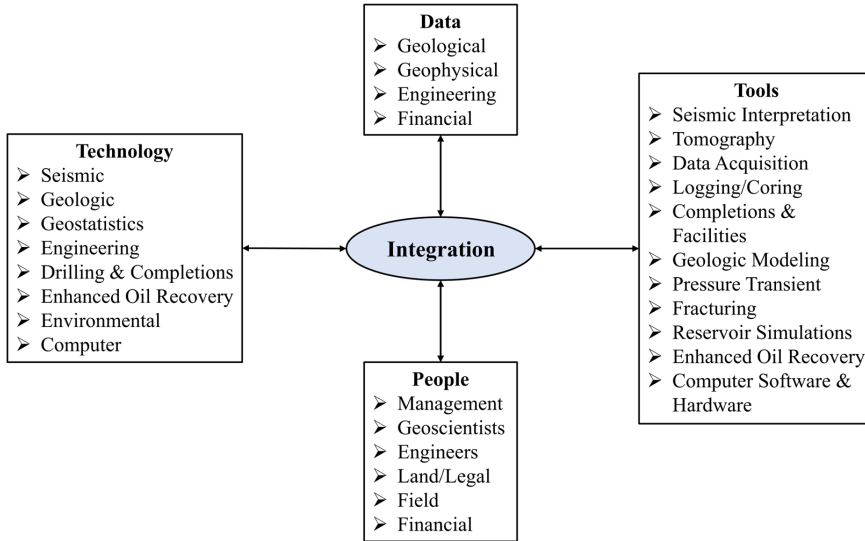


Figure 2. Reservoir Management. Adapted from (Satter et al., 1998).

## 2.2 Waterflooding

In this thesis, waterflooding analysis involves the use of reservoir simulation that is one of the main RM elements that would be focused on. Waterflooding is one of the Improved Oil Recovery methods that has long been used to improve hydrocarbon production. Waterflooding refers to injecting water into reservoirs to increase the recovery of hydrocarbon. The earliest waterflood could be traced back to an accidental incident that occurred due to poorly plugged wells or leaks from casing (Callaway, 1959). The respective advantages harvested have motivated the operators to inject water intentionally. Thereafter, waterflooding has been practiced and standardized as its mechanism is better understood. Apart from maximizing the recovery, the economics of waterflood needs to be considered to make a waterflooding plan successful. Some pieces of literature (Brundred and Brudred Jr., 1955; Muskat and Wyckoff, 1934; Paul Willhite, 1986; Satter and Iqbal, 2016c) discuss the theoretical framework and comprehensive economical assessment of waterflooding.

The main challenge with waterflooding is fundamentally how to optimize it. Such an engineering problem is generally termed “waterflooding optimization”. Different techniques can be applied for waterflooding optimization and examples of these techniques are zonal water injection, changing the direction of water injection, water shut-off, subdivision of the injection-production unit, and cyclic water injection. The readers are encouraged to peruse this reference (Lu and Xu, 2017) for the relevant rich details of these techniques. In addition, finding an optimal set of controls on the injectors (and/or producers) under waterflooding process is another optimization example. This optimal set of controls enables the waterflooding plan to be more cost-effective considering the oil price and the relevant costs of initiating waterflooding. This optimization example is the focus of this thesis.

## Chapter 2: Background of Concepts

To successfully conduct waterflooding optimization, reservoir engineers need useful working tools to forecast hydrocarbon production. Throughout the development of the petroleum engineering, there have been mainly three tools built to predict hydrocarbon production, viz., material balance equation (MBE), decline curve analysis (DCA), and numerical reservoir simulation (NRS). In general, NRS is deemed more robust than both MBE and DCA in capturing the physical system in the reservoir as it can be utilized for one-, two-, and three-phase system (**Odeh, 1969**). Moreover, NRS is more pertinent and useful to be employed to analyze waterflooding plans because it can better describe the reservoir performance under different operating conditions. Hence, NRS is the primary tool applied in this thesis. The details about NRS will be revealed in the following section. Interested readers are referred to the suggested materials for a more comprehensive understanding of MBE (**Craft et al., 1991; Dake, 1978**) and DCA (**Agarwal et al., 1998; Arps, 1945**).

### 2.3 Numerical Reservoir Simulation

Simulation generally means the representation of physical models through salient mathematical equations. In the oil and gas industry, the simulation models are circumscribed to hydrocarbon reservoirs. The term “Reservoir Simulation” hereby has been coined in the past few decades. Reservoir simulation fundamentally applies well-known reservoir engineering equations, which are solved by numerical methods, to model the fluid flow through discretized grid blocks in a subsurface reservoir (**Odeh, 1969**). This tool is alternatively known as NRS. Before proceeding to NRS, a reservoir model needs to be established and reservoir modeling fundamentally pertains to the description of properties (rock and fluid) related to subsurface (**Odeh, 1982**).

On closer scrutiny, a reservoir model is made up of numerous grid blocks in which the modeling highly relies upon static and dynamic data. In retrospect, this reference (**Satter and Iqbal, 2016b**) outlined a good discussion about these static and dynamic aspects. On the static component, the configuration of a reservoir model consists of the number of grid blocks, shapes of grid blocks, number of layers, model geometry, and boundaries. These properties, along with other geological and geophysical characteristics like porosity and permeability, are considered static. Assignments of PVT properties, capillary pressure, and relative permeability to specified regions of the reservoir model are also conducted. These assignments along with the predefined static properties are generally known as “model realization” which serves as part of the input data to the reservoir simulator.

The dynamic component is mainly associated with changes in fluid saturation and pressure in the reservoir. Other dynamic data include well production rate and bottomhole pressure (BHP) over the production period of the reservoir. Therefore, dynamic data is perceived as part of input as well as output for NRS. Besides that, other input data required by the reservoir simulator consists of initial conditions, well location, well constraints, simulation time intervals, and solution convergence criteria. In tandem with “model realization”, this input data contributes to the establishment of a “simulation model”. In the context of reservoir engineering, there are two simulation schemes, namely the black oil model and the compositional model (**Coats et al., 1998**). To have a more profound knowledge of NRS, the readers can refer to these materials (**Aziz and Settari, 1979; Ertekin et al., 2001; Mattax and Dalton, 1990**). In this thesis, E100, which acts as a black oil simulator (**Schlumberger, 2019**), is primarily implemented for the simulation of waterflooding.

### 2.4 Data Science

With the modernization of digital computers, the growth of the field of “Data Science” has extended to the petroleum industry, particularly reservoir engineering. Data Science fundamentally refers to a multidisciplinary study that implements scientific methods and mathematical algorithms to derive useful information and insight from data across a wide range of applications (Cao, 2017; Chen et al., 2018; Cleveland, 2014; Dhar, 2013). Data Science gains much attention in reservoir engineering due to its useful and robust applicability in handling and managing reservoir data. A lot of data can be generated or acquired during the production period. Comprehensive use of the obtained data can generate insights for reservoir engineers to make decisions.

The birth of Data Science contributes to the establishment of other terms, for instance, Data Analytics, Data Mining, and Data Engineering. Albeit these words are occasionally used interchangeably, they have different meanings. These words are formulated under the umbrella of Data Science. Of this, Data Analytics denotes approaches that allow the interpretation of data to retrieve meaningful patterns or relationships for the extraction of knowledge (Cao, 2017) whereas Data Mining implies the respective process of extraction (Han et al., 2012). Besides that, Data Engineering regards the transformation of raw data into usable one for Data Analytics (Reis and Housley, 2022). Further, application of Data Science and Analytics has been demonstrated to help resolve the RM problem in a few references (Mohaghegh, 2018, 2017a, 2017b). In the thesis, a similar illustration will be presented to highlight the robustness of Data Science and Analytics in reservoir engineering.

#### 2.4.1 Data-Driven Modeling Techniques

Using NRS to resolve RM issues can induce computational challenges if the geology of the reservoir model or the nature of the engineering problem (or both) is sophisticated. Hence, to increase the efficiency of computation, numerous solutions have been proposed, including applying high-performance computers, formulating simplifications of physics, having assumptions on the engineering problem, and developing data-driven models. In this thesis, we would mostly shed light on the application of data-driven modeling. Data-driven modeling is a part of Data Science and Analytics. As its name implies, data is the main building block of data-driven modeling. Fundamentally, data-driven modeling is fathomed as building a relationship between input data and output data that aims to reflect a physical system or process. Then, these models are implemented for predictive analysis.

There are two main classes of data-driven modeling, namely mathematics/statistics-based and machine learning-based (ML-based). One of the examples of mathematics/statistics-based techniques is the response surface model (RSM). RSM pertains to the approximate construction of the output yielded (also known as a response) from any process or relationship (to be modeled). Polynomial regression is well-received to build the response surface. For more information about RSM, interested readers can peruse these references (Box and Wilson, 1951; Gunst et al., 1996). RSM has been widely discussed to build data-driven models in different petroleum engineering applications as explained in these papers (Afari et al., 2022; Slotte and Smørgrav, 2008). Despite being convenient to be employed, this method is still subject to difficulty in capturing the highly nonlinear relationships.

## Chapter 2: Background of Concepts

Apart from RSM, kriging is another popular mathematics/statistics-based technique. Kriging is formulated under the context of geo-statistics (Meik and Lawing, 2017). It fundamentally acts as an interpolation method that is based upon the Gaussian process, which is governed by prior mean and covariance (Kleijnen, 2009). Kriging has also illustrated extensive applications in petroleum engineering (Fursov et al., 2020). Nevertheless, it has evident disadvantages that include the need for assumption, e.g., linearity and singularity. This undermines the implementation of mathematics/statistics-based techniques to develop data-driven models in comparison with ML-based ones.

Statistical-based approaches are not the main point of discussion in this thesis. ML-based techniques are in lieu given more emphasis to construct data-driven models. ML is defined as a computer program that is developed to draw inferences from patterns exhibited by data by implementing algorithms (Tom Mitchell, 1997). It is commonplace that ML has been mentioned interchangeably with the words “Artificial Intelligence” (AI). Nevertheless, they are different in that AI refers to the use of technology that enables a machine to emulate human behavior (Russell and Norvig, 2010). So, ML can be thought of as one of the catalysts for the success of AI. In the case of data-driven modeling, as discussed in (Mohaghegh, 2018, 2017a), ML does not require any simplification of physics and assumptions. Besides that, these techniques illustrate the successful implementation in capturing high nonlinearity (Golzari et al., 2015). Under the context of ML, there are three tasks of ML, namely supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning is a task of ML in which the data needs to be properly labeled whereas unsupervised learning pertains to the use of unlabeled data. Supervised learning is perceived as developing a function that can map a relationship between input and output data based on data provided (Kroese et al., 2019). In addition, supervised learning is typically implemented to conduct regression or classification of data. Hence, it is evident that the data used in this ML task needs to be labeled. One of the main differences between regression and classification tasks is that the output data for regression is numerical whereas that for classification is categorical (Kroese et al., 2019). Concerning this, examples of regression problems are the prediction of commodity prices, the forecast of revenue of a company, etc. For classification, one of the typical examples in geoscience pertains to deciding types of lithofacies from well logs. Techniques of supervised learning consist of artificial neural network (ANN), support vector machine (SVM), gradient boosting regressor (GBR), genetic programming (GP), k-nearest neighbor (k-NN) algorithm, and random forest (RF). In this aspect, some intriguing articles (Ozbayoglu et al., 2021; Tian and Horne, 2017) discussed the employment of supervised learning in petroleum engineering.

Unsupervised learning is understood as a type of algorithm that learns the relationship or pattern through unlabeled data (Kroese et al., 2019). Therefore, it is normally utilized to cluster the data provided. In this context, unsupervised learning exhibits the good capability to form different groups of data in which each group shares similar traits. Hierarchical clustering, k-mean clustering, and Gaussian Mixture Model are among the popular techniques of unsupervised learning. ANNs can also be used for unsupervised learning despite their more ubiquitous application in supervised learning. A few relevant examples of ANN-based unsupervised learning are autoencoders (Goodfellow et al., 2016) and restricted Boltzmann machines (Hinton et al., 2006; Tieleman, 2008). Several real-life cases that require unsupervised learning are the segmentation of customers for business strategies, DNA clustering for analysis of biological exploration, and so forth. In the oil and gas industry,

## Chapter 2: Background of Concepts

unsupervised learning has started gaining attention and has been discussed in several research articles (Alakeely and Horne, 2022; Alatrach et al., 2020; Jiang et al., 2022).

Reinforcement learning is comparatively more advanced than both supervised and unsupervised learning. Reinforcement learning is fundamentally developed as a Markov Decision Process, and it involves the use of agent, environment, and reward. The agents will take action in an environment to maximize the relevant reward (Joshi et al., 2021; van Otterlo and Wiering, 2012). Techniques of reinforcement learning are typically Q-learning and Deep Q learning. Reinforcement learning has been exemplary in carrying out various tasks, such as robot control, backgammon, and Alpha Go. However, to the best of my knowledge, its use in the oil and gas industry still does not succumb to wider exploration. Despite this fact, several papers (Hourfar et al., 2019; Ma et al., 2019) discussed the application of reinforcement learning in reservoir engineering to reflect its good potential to be extensively investigated and researched.

Upon understanding the types of ML tasks, readers are to be informed that this thesis will focus on the use of supervised learning. This is mainly because the nature of the engineering problem to be solved resonates better with supervised learning. In this case, the prediction of hydrocarbon production and waterflooding optimization can be perceived as a type of regression problem. Additionally, ANN is the ML technique that has been primarily considered to develop data-driven models. Figure 3 illustrates the schematic of a typical ANN. There are different variants of ANN being implemented in this thesis, such as feedforward neural network (FNN), also known as multilayer perceptron (MLP), Long-Short Term Memory (LSTM), and Gated Recurrent Unit (GRU).

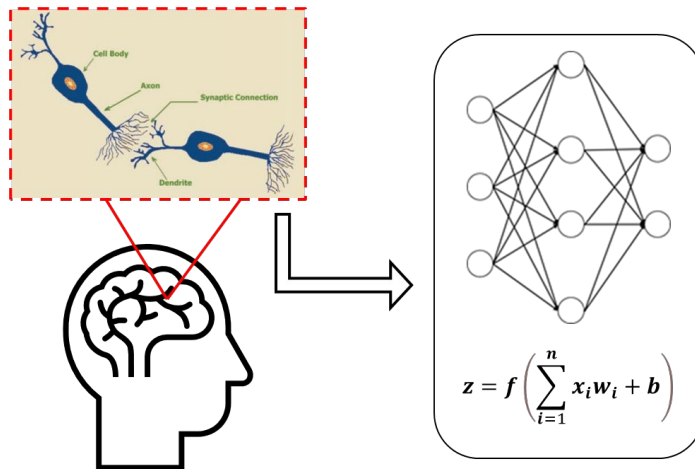


Figure 3. Schematic of Typical Artificial Neural Network. Adapted from (Mohaghegh, 2000).

In general, ML-based data-driven models need to be trained to be able to give a prediction. Training of the ML models pertains to the adjustment of parameters to yield the most optimal prediction. There is a distinct difference between parameters and hyperparameters. By definition, parameters known as “model parameters”, are the configurations that are embedded in the ML whereas hyperparameters are variables that are modified to regulate the training process and optimize the performance (Feurer and Hutter, 2019). Some examples of parameters consist of weights and biases in ANN and support

## Chapter 2: Background of Concepts

vectors in SVM. Hyperparameters include the learning rate, number of epochs, batch size, etc. in ANN and kernel in SVM. It can be baffling to consider hyperparameter optimization (HPO, defined as determining the best hyperparameters) as training. Precisely speaking, model parameter optimization (MPO) is considered training. However, HPO is performed before MPO or training and prevalent methods of HPO are random grid search and Bayesian Optimization. Despite increasing the chance of better predictability, HPO would increase the computational burden in addition to ML training.

### 2.5 Proxy Models as Replica of Numerical Reservoir Simulation

In the context of reservoir simulation, data-driven models can be treated as proxy models (also understood as surrogate models). Proxy models generally act as replica of reservoir simulation models. Aside from data-driven approaches, reduced physics modeling is another type of proxy modeling that requires assumptions and the simplification of physics. So, its applicability might not be considered robust in complex systems. As an example, Capacitance-Resistance Model (CRM) was proposed in the paper (**Bruce, 1943**) according to the idea of capacitors and resistors. Its implementations have been discussed in petroleum engineering, specifically in production optimization (**Hong et al., 2017; Liang et al., 2007; Sayarpour et al., 2009**). In addition, DCA is another option for proxy modeling that is mathematics-based. Nevertheless, DCA is deemed less sensitive to output prediction given changes in parameters (**Mohaghegh, 2017a**).

Based on the discussion in (**Mohaghegh, 2022, 2017a**), the preference for a data-driven ML-based approach has been verified to establish proxy models. Before delving into the details of the data-driven ML-based approach, we note that one of the main advantages of applying such proxy models is generally low computational footprint in tandem with results with high accuracy (compared with reservoir simulation). The objective of the proxy models should also be first identified and hence, the modeler would have a clear direction of how to develop proxy models. In addition, the source of data for proxy modeling originates either from real field data or reservoir simulation data (or a combination of both). Despite the source of data, the implementation of the ML-based approach remains unaltered.

#### 2.5.1 Source of Database

About the source of data, one of the very important reminders is ensuring that the database used for training the ML-based proxy models correctly reflects the physics being modeled. In general, the ML-based proxy models that use the real field data are termed “Top-Down Models” (TDM). This modeling approach marginalizes any simplification of physics or assumption by only leveraging the use of real field data (**Mohaghegh, 2017a**). Thus, everything starts with data and it is usually implemented in brown fields for which data is available. When only simulated data is considered, the proxy models are normally known as “Smart Proxy Models” (SPM) as demonstrated in some articles (**Shahkarami and Mohaghegh, 2020; Vida et al., 2019**). The SPM approach is mainly employed as another alternative for NRS when it comes to field development and planning for lower

## Chapter 2: Background of Concepts

computational footprints. Under the umbrella of TDM and SPM, there are three subcategories of modeling regarding the scale: Field-based (**Matthew, 2021**), Well-based (**Mohaghegh et al., 2012b**), and Grid-based (**Mohaghegh et al., 2012a**).

These three subcategories are associated with the scale of output data that has been used for proxy modeling. Therefore, a grid-based proxy model will yield any output parameter on the grid scale. The common output parameters in the grid scale comprise pressure and fluid saturation in each grid block. Such modeling subcategory has portrayed an extensive application in the domain of carbon, capture, utilization, and storage (CCUS) as being comprehensively discussed in (**Mohaghegh, 2018**). This is because, for in-depth analysis of efficient CO<sub>2</sub> storage, pressure and saturation values in grid-scale are deemed useful. Regarding both well-based and field-based modeling, they are usually employed for EOR or any production optimization process. Data in field-scale or well-scale usually consists of production rate, injection rate, and pressure. The selection of the subcategories is indeed case-dependent and highly relies upon the objective of the proxy models.

The type (or behavior) of data is another important issue that requires attention. Examples of static data include geological properties, such as permeability, porosity, and net-to-gross ratio whereas dynamic data comprise pressure, fluid saturation, production rates, and injection rates. Selection of input and output data can be performed either based upon the domain knowledge of the modelers or by applying the input feature selection method, viz. fuzzy logic (**Mohaghegh, 2017a**) which has been proven efficient in this domain. Besides, the insufficiency of data will impede the application of TDM. However, this can be overcome through the combination of both real field and simulated data. This hybridization approach can considerably be employed as one of the solutions for handling the insufficiency of real field data.

Upon perceiving the database, it is helpful to grasp an overview of the general methodology of proxy modeling. When the database is ready, it will be partitioned into three different datasets, viz., training, validation, and testing. There is no specific rule to set the ratio of partitioning, but it is usually either 7:1.5:1.5 or 8:1:1. After partitioning, the training dataset is primarily used to build the ML-based proxy model whereas the validation dataset is employed to ensure that the overfitting issue is circumvented during the training. Besides that, the testing dataset is applied to justify the predictability of the models. As illustrated in some literature (**Amini and Mohaghegh, 2019; He et al., 2016; Masoudi et al., 2020**), there is an additional step of further verifying the performance of models, which is known as “blind validation”. In practice, an additional database will be created to serve the purpose of blind validation. Unless the result of blind validation is considered good, the whole process of proxy modeling needs to be repeated.

### 2.5.2 Sampling Technique

About the use of simulation data, the key is to apply sampling techniques to generate different simulation scenarios from which a large database can be created. The sampling techniques that have been attempted in this work include Latin Hypercube sampling (LHS), Hammersley Sequence sampling (HSS), and Sobol Sequence sampling (SSS). LHS is considered an example of stratified sampling (**McKay et al., 1979**), formulated to overcome a limitation of Monte Carlo sampling. Such limitation pertains to inadequate sampling from the events with low probability, viz. P1 and P99 (**Bratvold and Begg, 2010**). As briefly highlighted in the cited book, LHS enables the division of



## Chapter 2: Background of Concepts

the Cumulative Distribution Frequency of input variables into a number of strata. This number is equal to the total number of iterations needed in which “sampling without replacement” is practiced. LHS is efficient to enhance the accuracy of Probability Density Function reproduction for a specified number of samples. Also, it decreases the number of samples required for a certain degree of accuracy and hence, it improves computational speed.

Besides that, random samples created from Monte Carlo sampling illustrate clustering of points, which leads to wasteful samples being retrieved. This is due to gaps in sample space. Therefore, low discrepancy sequences have been proposed to leverage the use of more uniformly distributed samples (**Cheng and Druzzel, 2000; Niederreiter, 1992**). Concerning this, discrepancy denotes a measure of nonuniformity of data points (**Wong et al., 1997**). In this context, the employment of low-discrepancy sequences in the creation of samples for Monte Carlo sampling is termed quasi-Monte Carlo. HSS (**Hammersley and Handscomb, 1964**) and SSS (**Sobol', 1967**) are the families of the quasi-Monte Carlo technique. Regarding HSS, as its name implies, it is a sampling technique that performs based on the Hammersley sequence. Concisely speaking, the Hammersley sequence is generated with the aid of prime numbers and radical inverse function. Interested readers are highly encouraged to refer to (**Cheng and Druzzel, 2000; Niederreiter, 1992; Wong et al., 1997**) for a better understanding of the mathematical formulation of HSS.

SSS is another sampling method that has been approached. As briefed in (**Cheng and Druzzel, 2000**), the Sobol sequence is created from a set of binary fractions of length  $w$  in which  $v_i^j$  are known as direction numbers where  $i = 1, \dots, w$  and  $j = 1, \dots, d$ , where  $d$  refers to the dimension of problem. In this case, a more efficient version of Sobol sequence, which was based on Gray code, was introduced in (**Antonov and Saleev, 1979**), and along with its employment was described in (**Bratley and Fox, 1988**).

### 2.5.3 Reservoir Case Study

In this thesis, the focus is placed on the development of SPM (field-based) to optimize the waterflooding process, which is an example of a dynamic problem. The source of data that we utilize is mainly from the simulation of different benchmark models, including the Egg Model (**Jansen et al., 2014**), the UNISIM-I-D model (**Schiozer et al., 2019**), and the OLYMPUS model (**Fonseca et al., 2020**). The details of these models will be correspondingly briefed in the papers compiled. Different reservoir models were attempted in this research work due to the intention of assessing the flexibility and applicability of the methodology proposed. Also, the real field data from Volve (**Equinor, 2018**) has been applied to justify the methodology of proxy modeling proposed here. **Figure 4**, **Figure 5**, and **Figure 6** respectively show Egg Model, UNISIM-I-D, and OLYMPUS. These figures were prepared by using the visualization software called ResInsight (**Ceetron Solution AS, 2020**). The color bar for these three figures denotes the horizontal permeability value in the  $x$ -direction. The warmer color indicates higher permeability.

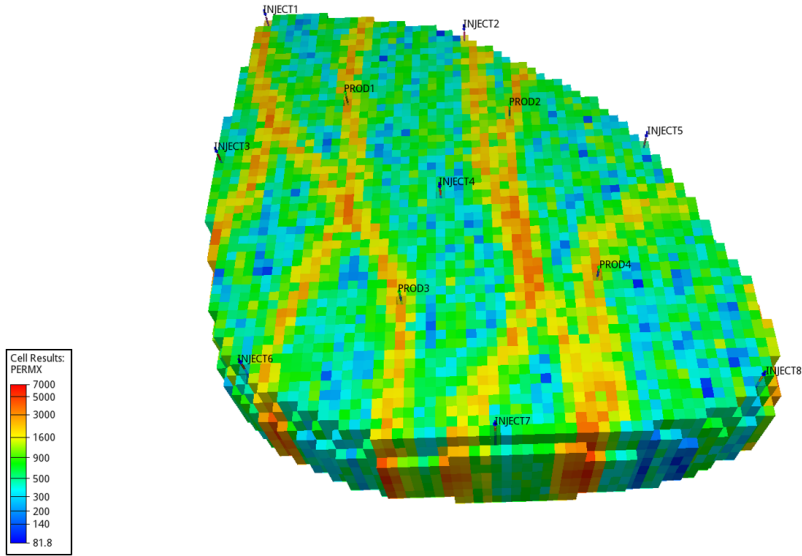


Figure 4. The architecture of the Egg Model with its well configurations (One of the realizations).

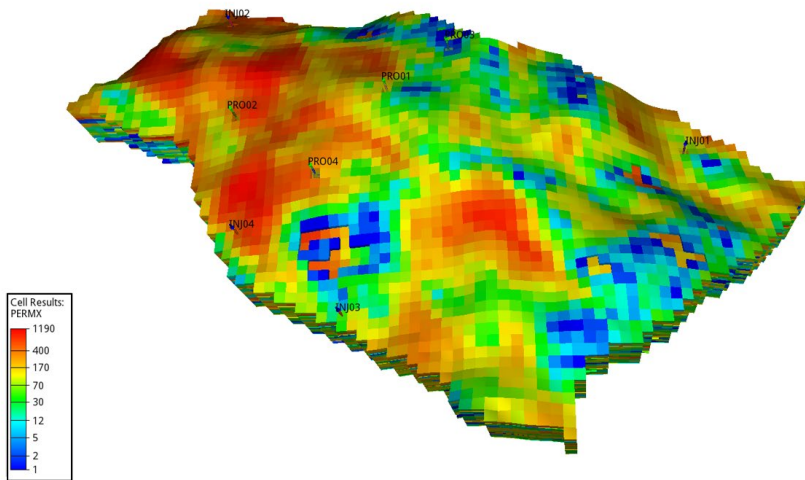


Figure 5. The architecture of the UNISIM-I-D Model with its well configurations.

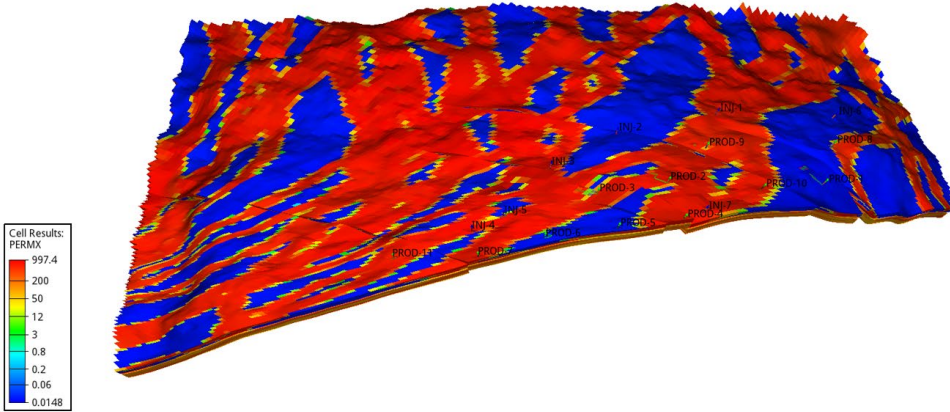


Figure 6. The architecture of the OLYMPUS Model with its well configurations (One of the realizations).

## 2.6 Optimization

Mathematical optimization (simply known as optimization) is defined as finding the best solution that can either minimize or maximize a given function subject to certain conditions. In this aspect, there are three main components in optimization, which refer to objective function (cost function), decision variables (control parameters), and constraints. Therefore, under predefined constraints, optimization is determining the best decision variables that can yield the best result of the objective function. As discussed in (Boyd and Vandenberghe, 2004), an optimization problem is expressed in standard form as follows:

$$\begin{aligned} & \text{minimize } f(x) \\ & \text{subject to } l_i \leq x_i \leq u_i, \quad i = 1, 2, \dots, n \end{aligned} \tag{1}$$

where  $x$  refers to the variables. The constraints are termed variable bounds. The displayed expression is for minimization. In the case of maximization, it can be done by negating the arbitrary function  $f(x)$ .

## Chapter 2: Background of Concepts

In the context of optimization, there are two essential concepts to be perceived: exploration and exploitation. Exploration regards locating the best solution over the solution (search) space whereas exploitation pertains to a more in-depth search for the best solution within a region where the best solution is believed to locate (Yang, 2014). Solutions can be divided into local and global optima. The global optimum denotes the best solution over the entire search space whereas the local optimum refers to the best solution within certain parts of the search space. Perfect optimization is to reach the global optimum. However, in a real-world application, it is nearly impossible to reach the “true” global optima. Thus, the ideal outcome of optimization is via the balance between exploration and exploitation in which convergence (as close as possible) to the global optima can be attained.

Two main types of mathematical algorithms can be employed for optimization, namely derivative-based and derivative-free. About the derivative-based algorithms, examples are the steepest descent (ascent) algorithm, Adaptive Estimation Moment (Adam), Newton-Raphson approximation, Levenberg-Marquardt algorithm (LMA), and conjugate gradient. One of the main challenges of applying derivative-based algorithms is the approximation of the gradient function. When complexity of the objective function increases, the gradient function can be computationally prohibitive to be estimated. Additionally, derivative-based algorithms demonstrate good performance in terms of exploitation. Therefore, this results in their higher tendency to converge to the local optima as compared with derivative-free algorithms.

Derivative-free algorithms are generally population-based and nature-inspired. These algorithms are also known as metaheuristics and comprise (but are not limited to) Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimization (GWO), etc. These algorithms are presented to have good capability to elude premature convergence to local optima because they achieve a good balance between exploration and exploitation (Ezugwu et al., 2020; Yang, 2014). Also, these derivative-free algorithms can create diverse solutions over the search space through the exploration component. Apart from these, the derivative-free algorithms possess practical and convenient implementation (as compared with the derivative-based) because approximation of gradients is not required. This explains the preference for this type of algorithm to perform production optimization.

In the context of reservoir engineering, choices of objective function generally include net present value (NPV), hydrocarbon production (oil or gas), and water cut. In this thesis, NPV is the selected objective function to be maximized by locating the optimal decision variables. The general formula of NPV is expressed in **Equation (2)**. In this equation,  $t$  is the period when the cash flow takes place whereas  $CF_t$  refers to the net cash flow over period  $t$ . Then,  $D$  means the interest rate. The net cash flow is mainly contributed by oil production minus the cost component that is made up of any possible cost that corresponds to conducting waterflooding. This will be explained more comprehensively in the published papers. To achieve production optimization under waterflooding, optimization algorithms are employed. In this work, emphasis is placed on nature-inspired algorithms. Moreover, the decision variables are mainly associated with the control rates of injectors (and the bottomhole pressure of producers). For more details, each paper briefs the background of the optimization problem.

$$NPV = \sum_{t=0}^n \frac{CF_t}{(1 + D)^t} \quad (2)$$

## Chapter 2: Background of Concepts

It is also vital to perceive that there are two different variants of optimization discussed in this thesis, namely ML training and production optimization. To avoid confusion, the term “optimization” used in this thesis (particularly in the published papers) only refers to “production optimization” unless specified. In essence, ML training is considered optimization because it involves determining the best learnable parameters to yield the best outcome of a cost function, and this requires the use of optimization algorithms. One of the pertinent epitomes is the backpropagation process of an artificial neural network to find the best weights and biases. For this, some of the commonly used algorithms for training during the backpropagation process, which are derivative-based, include LMA, SGD (Stochastic Gradient Descent), RMSProp (Root Mean Squares Propagation), and Adam. In this work, derivative-based algorithms, especially LMA and Adam, were primarily implemented to train the ML models as their implementations have been readily embedded in the programming package used. For the details, refer to **(Gavin, 2019)** for LMA and **(Kingma and Ba, 2015)** for Adam. When it comes to optimization tasks, derivative-free algorithms were preferred, including PSO, GWO, and GA. The explanation of these algorithms can be found in the compiled papers. These algorithms were respectively coupled with NRS and proxy models to optimize the waterflooding process.

### 2.7 Decision Analysis

Numerous approaches of RM aim at optimizing the recovery from a hydrocarbon reservoir for higher profits. Hence, engineers cannot be circumvented from making a decision that is considered better under the context of RM. Unfortunately, such a decision-making (DM) process is never easy because it needs an assessment of many sophisticated and uncertain factors. To assist every decision maker in enhancing their DM process, the definition of decision analysis (DA) has been coined and discussed in different resources. Nonetheless, its definition generally gravitates to a process of transforming an opaque decision problem into a more transparent one through a series of transparent steps **(Howard, 1980)**. In the context of DA, it is essential to understand that good decision does not always yield good outcomes.

Uncertainty is an inalienable element of DM. In the domain of RM, we often pursue the idea of uncertainty quantification or reduction to result in decisions with higher quality. However, quantifying or reducing uncertainty does not necessarily increase the quality of a decision. It has thereby been a common misconception among the engineering community that a higher reduction in uncertainty implies better decision outcomes. Such misconception encourages many engineers to include as much information or details as possible in their DM process. Regarding this, uncertainty quantification is only meaningful (or creates values) if it could change a decision that would have been made otherwise. It can be profligate use of resources to further reduce the uncertainty especially when the decision is clear.

To evaluate if the uncertainty quantification is valuable, a DA tool, namely Value-of-Information (VOI) was established. Information or data acquisition is commonplace in RM to quantify uncertainty. It is then important to know if these information acquisition activities will produce any improvement in DM considering their costs. Concerning this, VOI appears to be useful as it has been implemented to assess the benefits of gathering additional data before the data is collected for the DM process. The idea of VOI was first implemented for business decisions as introduced in

## Chapter 2: Background of Concepts

(Schlaifer, 1959). Its application was proposed in the petroleum industry through (Grayson, 1960). Thereafter, a comprehensive review of its use in the petroleum industry was presented in (Bratvold et al., 2009). This provided a good overview of the development of the VOI concept in the oil and gas industry. On closer scrutiny, VOI has started to gain attention and be researched more extensively over the past decade in areas of the industry (Dutta et al., 2019; Eidsvik et al., 2017). VOI framework aids decision makers to embrace uncertainty by valuing the information obtained within a decision context and so, its applicability subsides without a clear decision context (Hong et al., 2018).

Determination of the VOI can be performed by employing the simulation-regression approach. Different insightful references have explained the use of simulation-regression approaches in terms of VOI computation. In hindsight, Least-Squares Monte Carlo (LSM) algorithm is the epitome of simulation-regression approaches. LSM was initiated in (Longstaff and Schwartz, 2001) to value American options in the financial industry. Thanks to its robust application, LSM has begun to be well-received for real option valuation in the petroleum industry. One of them pertained to the valuing of oil and gas options as discussed in (Willigers and Bratvold, 2009). It has also been proven useful to help with the resolution of the sequential DM problems as demonstrated in (Hong et al., 2019; Tadjer et al., 2021), and RM is the epitome of sequential DM problem.

# Chapter 3

## Contributions and Summaries of Papers

This chapter provides brief discussions about contributions and summaries of the 8 manuscripts compiled in this thesis. The papers contributed to the frameworks developed to establish proxy models for the waterflooding process. Each of the papers is overviewed as follows:

### ***Paper 1 – A Survey on Application of Machine Learning and Metaheuristic Algorithms for Intelligent Proxy Modeling in Reservoir Simulation***

Authors: Cuthbert Shang Wui Ng, Menad Nait Amar, Ashkan Jahanbani Ghahfarokhi, Lars Struen Imsland

Status: Published in Computers and Chemical Engineering

In this paper, we conducted a survey on the use of ML and metaheuristic algorithms in the development of intelligent proxy models in the domain of reservoir simulation. The word “intelligent” here implies the involvement of ML techniques to reinforce the predictability of the models built. This paper explained a general workflow of conducting intelligent proxy modeling, which can be a guide for the readers to start exploring the use of ML in reservoir simulation. Besides that, we realized that metaheuristic algorithms have begun playing an important role in formulating proxy models. These algorithms were mainly used to solve optimization problems, but their use in training the proxy models was also discussed. Therefore, we investigated numerous literature which expounded on the application of these algorithms in tandem with intelligent proxies. This survey paper provided insights into the current trend of development of ML-based proxy modeling. Regarding this, the paper offered an overview of how the intelligent proxy models functioned in different aspects of reservoir simulation, namely well placement, monitoring production parameters (e.g., oil and gas production rates), carbon, capture, and storage (CCS), history matching, waterflooding, miscible gas injection, water-alternating-gas (WAG) injection, and other enhanced oil recovery (EOR) techniques. We also outlined discussions and summarized a few opinions of ours on the use of ML and metaheuristic algorithms in reservoir simulation. This survey paper supplied an inspiration for the development and further improvement of the methodology discussed in the next papers.

### Chapter 3: Contributions and Summaries of Papers

#### ***Paper 2 – Smart Proxy Modeling of a Fractured Reservoir Model for Production Optimization: Implementation of Metaheuristic Algorithm and Probabilistic Application***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar, Ole Torsæter

Status: Published in Natural Resources Research

This paper was written as a result of research work done in the Ph.D. course “PG8605 - Dual Porosity Reservoirs” at NTNU. This paper laid out a foundation that helped in the development of a smart proxy model. In this aspect, a synthetic dual porosity / dual permeability model was built and waterflooded as a case study. Also, the prevalent variant of ANN, which is the feedforward neural network, was the selected ML technique in this work. Steps of developing the proxy models, including database generation and training of the models, were holistically discussed. Furthermore, two types of algorithms, viz, backpropagation algorithm and PSO, were investigated and implemented to train the proxy models for comparative studies. The details of these two algorithms were presented to enable the readers to understand how they are related to ML training. SGD and Adam were both used to conduct the backpropagation algorithm. Probabilistic analysis was also incorporated to better perceive the performance of the proxy models established. The work performed in this paper was an important precursor for the rest of the papers. It enabled further improvement to be embedded for applications with closer proximity to real-life cases.

#### ***Paper 3 – Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar

Status: Published in Journal of Petroleum Exploration and Production Technology

This paper displayed how smart proxy modeling, introduced in **Paper 1** and implemented in **Paper 2**, could be extended to a more realistic reservoir model and sophisticated application. Feedforward neural networks were again implemented in this study. Also, in this work, the renowned Egg Model was used as the benchmark, and production optimization was conducted via well control under the waterflooding process. Sampling techniques were incorporated here to generate a database to train the proxy models. This database aimed to cover the solution space in which the optimal well control could be located. With this, the data was partitioned and employed to enable the proxy models to learn the relationship between the input and output data given. When it came to the optimization part, nature-inspired algorithms, viz. PSO and GWO were chosen. To further confirm the accuracy of the results, optimization was also carried out by coupling these algorithms with the NRS. This was to check if the proxy models would be able to yield the optimal result that was close to that of NRS. Upon completing the whole workflow, the methodology was inferred to be practically reliable to resolve the optimization problem discussed.



### Chapter 3: Contributions and Summaries of Papers

#### **Paper 4 – *Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar

Status: Published in Petroleum

This paper is considered a continuation of **Paper 3** in which a different variant of ANN was approached. In this aspect, LSTM, one of the examples of RNN, was selected. This was because, to the best of our knowledge, LSTM has not been much studied in the domain of proxy modeling for the resolution of RM issues. This motivated the formulation of the work presented in this paper. Fundamentally, the methodology discussed in **Paper 3** was implemented to develop the proxy models. Nevertheless, the optimization results attained by having RNN were shown to have slightly higher accuracy as compared with that discussed in **Paper 3**.

#### **Paper 5 – *Adaptive Proxy-based Robust Production Optimization with Multilayer Perceptron***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi

Status: Published in Applied Computing and Geoscience

In this paper, the methodology presented in **Paper 3** and **Paper 4** was further refined to consider geological uncertainty. About this, 10 different geological realizations of the Egg model were embedded in the generation of the database for proxy modeling. Multilayer perceptron (MLP), an alternative term for feedforward neural networks, was applied as the ML technique to conduct the modeling. The refinement done here was to integrate the adaptive sampling into the whole framework. This implied that an additional sample, which was the optimal control obtained from the optimization with the developed proxy models, would be included in the initial database for retraining. Such integration would improve the training database as samples with better quality were added. For this, a criterion check was employed to verify the quality of the samples. After fulfilling the criterion, these samples were considered a new addition to the database. By doing so, the database was able to comprise more diverse samples which enabled proxy models with better performance to be established. Despite having adaptive training in the whole methodology, computational efficiency was ensured considering optimization under geological uncertainty.

### Chapter 3: Contributions and Summaries of Papers

#### **Paper 6 – *Fast Well Control Optimization with Two-Stage Proxy Modeling***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Wilson Wiranda

Status: Published in *Energies*

Complementary work: **EAGE Conference Extended Abstract entitled “*Fast well control optimization using machine learning based proxy models*”.**

This paper provided another viable enhancement to the established methodology. It is worth mentioning that the reservoir model used was the UNISIM-I-D model. In this context, two phases of proxy modeling, viz. global and local proxy modeling, were carried out. Fundamentally, the initially sampled database was used to build the global proxy models. Thereafter, global proxy models were coupled with optimization algorithms to create a new database that was used to train the local proxy models. By comparing the training results, local proxy exhibited an improvement in accuracy. Furthermore, the optimization results of local proxy models were deemed closer to the “ground truth” (or the optimization results obtained by NRS) in comparison with global proxy models. Significant computational efficiency was also attained. Hence, this version of methodology was illustrated to have the ability to solve an optimization problem with higher dimensions involving 200 optimization variables. This paper was inspired by the contemporary study done for a conference abstract that was presented at EAGE Conference on Digital Innovation for a Sustainable Future.

#### **Paper 7 – *Well production forecast in Volve field: Application of rigorous machine learning techniques and metaheuristic algorithm***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar

Status: Published in *Journal of Petroleum Science and Engineering*

In this paper, we used the real well production data from Volve to establish predictive models with the aid of ML. It was basically another extensive illustration of the methodology discussed in **Paper 1**. This work involved the use of different neural networks and SVR. The neural networks included ANN trained by Adam and PSO, simple RNN, LSTM, and GRU. The developed models were implemented to give predictions of well production rate, which serves as one of the important parameters for RM. Conventionally, DCA has been one of the most common methods for this purpose. In this work, it has been showcased that ML-based models could be considered as another alternative. Besides that, comparative studies were done to investigate the performance of each of the models mentioned. Through the investigation, we gained better ideas and insights to establish proxy models.

### Chapter 3: Contributions and Summaries of Papers

#### **Paper 8 – *Optimizing initiation time of waterflooding under geological uncertainties with Value of Information: Application of simulation-regression approach***

Authors: Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi

Status: Published in Journal of Petroleum Science and Engineering

This paper was a demonstration of the coupling between DA tool and ML methods. In another word, it displayed the potential of ML to be used under the framework of DA for RM purposes. In this context, VOI analysis, which is a priori analysis, is the DA tool chosen. This paper presented how VOI could be computed through a simulation-regression approach, namely the modified Least-Square Monte Carlo (LSM) algorithms. It is vital to be cognizant that VOI is a tool that helps decision-makers to improve the quality of a decision by embracing uncertainties instead of reducing uncertainties. The case study used here was the OLYMPUS benchmark model under waterflooding process, in which geological uncertainties were considered. As the name of LSM implies, linear regression is one of its components. In addition, ML techniques, including Gaussian Process Regression (GPR) and Support Vector Regression (SVR), were employed under the paradigm of LSM. The application of LSM in resolving the RM issues is generally termed Sequential Reservoir Decision-Making (SRDM). The incorporation of ML into LSM for the illustration of SRDM portrayed high applicability and usefulness not only in terms of proxy modeling but also in the resolution of the RM problem.

This 3-year doctorate research contributed to the formulation of other research works apart from the journal articles discussed. As a result, I have been able to investigate more about the robust application of ML in other aspects of reservoir engineering, such as modeling of interfacial tension, WAG injection, and wax deposition. The results of these works are published in the following papers which are not considered as elements of this thesis.

Ng, Cuthbert Shang Wui; Djema, Hakim; Nait Amar, Menad; Jahanbani Ghahfarokhi, Ashkan. (2022) **Modeling interfacial tension of the hydrogen-brine system using robust machine learning techniques: Implication for underground hydrogen storage.** International Journal of Hydrogen Energy. Volume 47 (93), 1 December 2022, Pages 39595-39605

Nait Amar, Menad; Jahanbani Ghahfarokhi, Ashkan; Ng, Cuthbert Shang Wui; Zeraibi, Nouredine. (2021) **Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms.** Journal of Petroleum Science and Engineering. Volume 206, November 2021, 109038

Nait Amar, Menad; Jahanbani Ghahfarokhi, Ashkan; Ng, Cuthbert Shang Wui. (2021) **Predicting wax deposition using robust machine learning techniques.** Petroleum. Volume 8 (2), June 2022, Pages 167-173

### Chapter 3: Contributions and Summaries of Papers

Due to COVID-19 Pandemic, I was subject to some travel restrictions for the past 2 years. Hence, my opportunity to physically join conferences only came in my third year of Ph.D. study. With this, I was able to take part in three conferences and present my research works. One of these works is linked with my **Paper 6**. These three conferences are:

Jahanbani Ghahfarokhi, Ashkan; Ng, Cuthbert Shang Wui; Nait Amar, Menad. (2022) **Artificial Intelligence / Machine Learning for Sustainable Utilization of the Subsurface**. EAGE GET. EAGE; The Hague, The Netherlands. 2022-11-07 - 2022-11-09.

Ng, Cuthbert Shang Wui; Jahanbani Ghahfarokhi, Ashkan. (2022) **Fast well control optimization using machine learning based proxy models**. EAGE Conference on Digital Innovation for a Sustainable Future. EAGE; Bangkok, Thailand. 2022-09-13 - 2022-09-15.

Ng, Cuthbert Shang Wui. (2022) **Application of Data-Driven Models in Reservoir Management**. BRU21 Conference. NTNU; Trondheim, Norway. 2022-06-01 - 2022-06-03.

# Chapter 4

## Concluding Remarks and Recommendations

This chapter summarizes the main findings of this Ph.D. research work and discusses the limitations as well as possible extensive works in the future. In general, this research work has achieved its goal in which a fundamental workflow of methodology has been established to develop proxy models. The proxy modeling was performed in the context of reservoir simulation. Also, the proxy models are mainly data-driven in which ML techniques are the primary ingredient. Nevertheless, it has been reckoned that possible future works are still required to further reinforce the maturity of this framework to consider more robust applications. These applications, for instance, include more pertinent uncertainties, specifically for geological properties, relatively more sophisticated optimization problems and reservoir models that are geologically more complex.

Albeit the main research pertains to proxy modeling, it is necessary to understand that having good predictive ability is the initial step to successful proxy modeling. In essence, this thesis also enclosed a framework to yield data-driven models with good prediction performance. This explicitly contributed to the formulation of the fundamental methodology in the aspect of proxy modeling. Overall, the thesis aims at offering a scaffold to the foundation of the proxy modeling framework in reservoir engineering and providing insights into its further reinforcement. This thesis also targets to illustrate a robust embedment of ML in a more systemic context of DA. Despite still being subject to several limitations, the results garnered from this work signify that the milestones have been accomplished.

The main findings and contributions of this thesis are presented as follows:

1. Providing a survey on the application of ML and metaheuristic algorithms in reservoir simulation, particularly in proxy modeling. **Paper 1** overall portrayed the role of ML and metaheuristic algorithms hitherto in facilitating proxy modeling. Through this survey, an in-depth understanding of the potential of ML and metaheuristic algorithms can be obtained.
2. Contributing to a workflow of building proxy models that can help to solve RM issues, particularly for waterflooding. Most of the papers compiled in this thesis illustrated step-by-step explanation of the methodology for better enlightenment about the principles of proxy modeling with ML.

## Chapter 4: Concluding Remarks and Recommendations

3. Presenting and discussing how the developed proxy models can be coupled with metaheuristic algorithms to handle the optimization problems under the context of RM. To achieve waterflooding optimization, these algorithms were also implemented along with a reservoir simulator for comparison purposes. Via this comparative analysis, the accuracy of proxy models was confirmed.
4. Offering an alternative solution that can provide a fast evaluation for RM and further analyses. Computational efficiency was attained by performing proxy modeling in which much less computational time was required to conduct the optimization.
5. Demonstrating how several extensions can be performed to tackle more sophisticated engineering problems. **Paper 5** and **Paper 6** discussed the approaches taken to enhance the methodology presented in **Paper 2**, **Paper 3**, and **Paper 4**. In this case, **Paper 5** emphasized geological uncertainties whereas **Paper 6** focused on problems with higher dimensionality and a more complex reservoir model.
6. Displaying how ML can play a part in predictive modeling. In this context, proxy modeling can be used for predictive modeling. **Paper 7** briefed about the application of ML techniques to build predictive models for production rate based on real field data. The good prediction performance of these models was highlighted, as a successful application of proxy modeling methodology.
7. Illustrating the potential of ML to be incorporated with DA tools for VOI analysis. **Paper 8** expounded on how some selected ML methods could be integrated into the LSM algorithm for VOI analysis, such as finding the best initiation time of waterflooding.

Some limitations have been discussed thoroughly in the papers. Also, some recommendations have been proposed as possible future works to address these limitations. Other ideas or recommendations are also outlined to enhance the methodology. In general, these recommendations are considered to further tweak the fundamental framework to elevate its maturity. These recommendations are as follows:

1. Integration of more geological uncertainties for proxy modeling: it is of great importance to understand that including as many geological uncertainties as possible is deemed impractical. Thus, a balance between practicality and uncertainty consideration needs to be honored. In this aspect, embedding a clustering technique (**Salehian et al., 2021**) into the proxy modeling methodology can be done to ensure the representativeness of geological realizations and computational efficiency.
2. Consideration of economic uncertainty: waterflooding optimization discussed in this thesis primarily involved constant economic parameters. Stochastic price modeling approaches, such as the Two Factor Price Model (**Jafarizadeh and Bratvold, 2013**), can be embedded as future works. Encapsulating the model of economic uncertainty enables the stochasticity of price to be considered in terms of optimization. This serves a step closer to real-life applications and certainly matures the whole methodology.

## Chapter 4: Concluding Remarks and Recommendations

3. Hyperparameter optimization: an embodiment of hyperparameter optimization would increase the total time of computation. Nonetheless, neglecting it in some cases might produce suboptimal results for ML training. In this work, a trial-and-error approach was employed. In this aspect, including a more robust technique of hyperparameter optimization is certainly applaudable. It is inspiring if a pipeline of an automated workflow (with much higher computational efficiency) that considers hyperparameter optimization and ANN training, as illustrated in this paper (**Olson et al., 2016**) as a Tree-based Pipeline Optimization Tool, can be yielded in future.
4. Dimensionality reduction: the increase in the dimension of data in this work mainly stemmed from the number of input parameters and the number of realizations. To address the former, several existing methods of input parameter selection, such as fuzzy logic and mutual information method (**Thanh et al., 2022**) that is based on Shannon entropy in information theory (**Shannon, 1948**), can be included. Also, regarding the number of realizations, clustering technique that selects useful realizations as explained in (**Salehian et al., 2021**) can be pondered to concisely consider the geological uncertainty.
5. Creation of a better coupling between proxy models and DA tool: to this end, under the context of the simulation-regression approach, a proxy model with high fidelity can act as the source of simulation whereas different ML techniques can “replace” the regression component. This serves as a step forward in better application of DA in reservoir engineering that leverages the use of data-driven approaches, particularly ML.
6. Role of unsupervised and reinforcement learning: the potential of unsupervised and reinforcement learning is worth being researched and studied to explore further possible breakthroughs in proxy modeling and its functionality in the resolution of RM issues.
7. Contributing to the energy transition: upon maturing the methodology discussed here, extending it to the areas in energy transition, viz. CCS, HS and geothermal energy storage is recommended. Achieving energy transition optimally and economically involves different optimization problems and DM processes. Hence, this methodology can play an important part in the future applications.

# Bibliography

- Afari, S., Ling, K., Sennaoui, B., Maxey, D., Oguntade, T., Porlles, J., 2022. Optimization of CO<sub>2</sub> huff-n-puff EOR in the Bakken Formation using numerical simulation and response surface methodology. *J. Pet. Sci. Eng.* 215, 110552. <https://doi.org/10.1016/J.PETROL.2022.110552>
- Agarwal, R.G., Gardner, D.C., Kleinsteiber, S.W., Fussell, D.D., 1998. Analyzing Well Production Data Using Combined Type Curve and Decline Curve Analysis Concepts. *SPE Annu. Tech. Conf. Exhib.* <https://doi.org/10.2118/49222-MS>
- Alakeely, A., Horne, R., 2022. Simulating Oil and Water Production in Reservoirs with Generative Deep Learning. *SPE Reserv. Eval. Eng.* 1–24. <https://doi.org/10.2118/206126-PA>
- Alatrach, Y., Mata, C., Omrani, P.S., Saputelli, L., Narayanan, R., Hamdan, M., 2020. Prediction of well production event using machine learning algorithms, in: *Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference 2020, ADIP 2020.* <https://doi.org/10.2118/202961-ms>
- Amini, S., Mohaghegh, S., 2019. Application of machine learning and artificial intelligence in proxy modeling for fluid flow in porous media. *Fluids.* <https://doi.org/10.3390/fluids4030126>
- Antonov, I.A., Saleev, V.M., 1979. An economic method of computing LPr-sequences. *USSR Comput. Math. Math. Phys.* 19. [https://doi.org/10.1016/0041-5553\(79\)90085-5](https://doi.org/10.1016/0041-5553(79)90085-5)
- Arps, J.J., 1945. Analysis of Decline Curves. *Trans. AIME* 160. <https://doi.org/10.2118/945228-g>
- Aziz, K., Settari, A., 1979. *Petroleum Reservoir Simulation.* Applied Science Publishers.
- Box, G.E.P., Wilson, K.B., 1951. On the Experimental Attainment of Optimum Conditions. *J. R. Stat. Soc. Ser. B* 13. <https://doi.org/10.1111/j.2517-6161.1951.tb00067.x>
- Boyd, S., Vandenberghe, L., 2004. *Convex Optimization, Convex Optimization.* Cambridge University Press.
- Bratley, P., Fox, B.L., 1988. Algorithm 659: Implementing Sobol's Quasirandom Sequence Generator. *ACM Trans. Math. Softw.* 14. <https://doi.org/10.1145/42288.214372>
- Bratvold, R.B., Begg, S.H., 2010. *Making Good Decisions*, first. ed. Society of Petroleum Engineers, Richardson, Texas.
- Bratvold, R.B., Bickel, J.E., Lohne, H.P., 2009. Value of information in the oil and gas industry: Past, present and future. *SPE Reserv. Eval. Eng.* 12. <https://doi.org/10.2118/110378-PA>
- Bruce, W.A., 1943. An Electrical Device for Analyzing Oil-reservoir Behavior. *Trans. AIME.* <https://doi.org/10.2118/943112-g>
- Brundred, L.L., Brudred Jr., L.L., 1955. Economics of Water Flooding. *J. Pet. Technol.* 7, 12–17. <https://doi.org/10.2118/459-G>
- Callaway, F.H., 1959. Evaluation of Waterflood Prospects. *J. Pet. Technol.* 11.



<https://doi.org/10.2118/1258-g>

- Cao, L., 2017. Data science: A comprehensive overview. *ACM Comput. Surv.*  
<https://doi.org/10.1145/3076253>
- Ceetron Solution AS, 2020. ResInsight.
- Chen, J., Ayala, B.R., Alsmadi, D., Wang, G., 2018. Fundamentals of Data Science for Future Data Scientists, in: *Analytics and Knowledge Management*.  
<https://doi.org/10.1201/9781315209555-6>
- Cheng, J., Druzdzel, M.J., 2000. Computational Investigation of Low-Discrepancy Sequences in Simulation Algorithms for Bayesian Networks. *Proc. Sixt. Conf. Uncertain. Artif. Intell.*
- Cleveland, W.S., 2014. Data science: An action plan for expanding the technical areas of the field of statistics. *Stat. Anal. Data Min.* 7. <https://doi.org/10.1002/sam.11239>
- Coats, K.H., Thomas, L.K., Pierson, R.G., 1998. Compositional and Black Oil Reservoir Simulation. *SPE Reserv. Eng. (Society Pet. Eng. 1)*. <https://doi.org/10.2118/50990-pa>
- Craft, B.C., Hawkins, M.F., revised by Terry, R.E., 1991. *Applied Petroleum Reservoir Engineering Second Edition*. Prentice Hall PTR.
- Dake, L.P., 1978. *Fundamentals fo Reservoir Engineering, volume 8, Developments in Petroleum Science*.
- Dhar, V., 2013. Data science and prediction. *Commun. ACM* 56. <https://doi.org/10.1145/2500499>
- Dutta, G., Mukerji, T., Eidsvik, J., 2019. Value of information analysis for subsurface energy resources applications. *Appl. Energy* 252. <https://doi.org/10.1016/j.apenergy.2019.113436>
- Eidsvik, J., Dutta, G., Mukerji, T., Bhattacharjya, D., 2017. Simulation–Regression Approximations for Value of Information Analysis of Geophysical Data. *Math. Geosci.* 49. <https://doi.org/10.1007/s11004-017-9679-9>
- Equinor, 2018. Disclosing all Volve data [WWW Document]. URL <https://www.equinor.com/en/news/14jun2018-disclosing-volve-data.html> (accessed 6.28.21).
- Ertekin, T., Abou-Kassem, J.H., King, G.R., 2001. *Basic Applied Reservoir Simulation, Vol. 7*. ed. Society of Petroleum Engineers (SPE).
- Ezugwu, A.E., Adeleke, O.J., Akinyelu, A.A., Viriri, S., 2020. A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. *Neural Comput. Appl.* 32. <https://doi.org/10.1007/s00521-019-04132-w>
- Feurer, M., Hutter, F., 2019. Hyperparameter Optimization. [https://doi.org/10.1007/978-3-030-05318-5\\_1](https://doi.org/10.1007/978-3-030-05318-5_1)
- Fonseca, R.M., Rossa, E. Della, Emerick, A.A., Hanea, R.G., Jansen, J.D., 2020. Introduction to the special issue: Overview of OLYMPUS Optimization Benchmark Challenge. *Comput. Geosci.*  
<https://doi.org/10.1007/s10596-020-10003-4>
- Fursov, I., Christie, M., Lord, G., 2020. Applying kriging proxies for Markov chain Monte Carlo in reservoir simulation. *Comput. Geosci.* 24. <https://doi.org/10.1007/s10596-020-09968-z>
- Gavin, H.P., 2019. *The Levenberg-Marquardt Algorithm For Nonlinear Least Squares Curve-Fitting Problems*. Duke Univ.
- Golzari, A., Haghghat Sefat, M., Jamshidi, S., 2015. Development of an adaptive surrogate model

## Bibliography

- for production optimization. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2015.07.012>
- Goodfellow, I., Bengio, Y., Courville, A., 2016. *Deep learning* An MIT Press Book, Nature.
- Grayson, C.J.J., 1960. *Decisions Under Uncertainty: Drilling Decisions by Oil and Gas Operators*. Harvard Business School, Division of Research, Boston, MA.
- Gunst, R.F., Myers, R.H., Montgomery, D.C., 1996. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. *Technometrics* 38. <https://doi.org/10.2307/1270613>
- Hammersley, J.M., Handscomb, D.C., 1964. *Monte Carlo Methods*, Monte Carlo Methods. <https://doi.org/10.1007/978-94-009-5819-7>
- Han, J., Kamber, M., Pei, J., 2012. *Data Mining: Concepts and Techniques*, *Data Mining: Concepts and Techniques*. <https://doi.org/10.1016/C2009-0-61819-5>
- He, Q., Mohaghegh, S.D., Liu, Z., 2016. Reservoir simulation using smart proxy in SACROC unit - Case study, in: *SPE Eastern Regional Meeting*. <https://doi.org/10.2118/184069-MS>
- Hinton, G.E., Osindero, S., Teh, Y.W., 2006. A fast learning algorithm for deep belief nets. *Neural Comput.* 18. <https://doi.org/10.1162/neco.2006.18.7.1527>
- Hong, A., Bratvold, R.B., Lake, L.W., 2019. Fast analysis of optimal improved-oil-recovery switch time using a two-factor production model and least-squares Monte Carlo algorithm. *SPE Reserv. Eval. Eng.* <https://doi.org/10.2118/191327-PA>
- Hong, A.J., Bratvold, R.B., Nævdal, G., 2017. Robust production optimization with capacitance-resistance model as precursor. *Comput. Geosci.* <https://doi.org/10.1007/s10596-017-9666-8>
- Hong, A.J., Bratvold, R.B., Thomas, P., Hanea, R.G., 2018. Value-of-information for model parameter updating through history matching. *J. Pet. Sci. Eng.* 165. <https://doi.org/10.1016/j.petrol.2018.02.004>
- Hourfar, F., Bidgoly, H.J., Moshiri, B., Salahshoor, K., Elkamel, A., 2019. A reinforcement learning approach for waterflooding optimization in petroleum reservoirs. *Eng. Appl. Artif. Intell.* <https://doi.org/10.1016/j.engappai.2018.09.019>
- Howard, R.A., 1980. An Assessment of Decision Analysis. *Oper. Res.* 28, 4–27. <https://doi.org/10.1287/opre.28.1.4>
- Jafarzadeh, B., Bratvold, R.B., 2013. Sell spot or sell forward? Analysis of oil-trading decisions with the two-factor price model and simulation. *SPE Econ. Manag.* 5. <https://doi.org/10.2118/165581-PA>
- Jansen, J.D., Fonseca, R.M., Kahrobaei, S., Siraj, M.M., Van Essen, G.M., Van den Hof, P.M.J., 2014. The egg model - a geological ensemble for reservoir simulation. *Geosci. Data J.* <https://doi.org/10.1002/gdj3.21>
- Jiang, T., Bonnie, R.J.M., Correa, T.S., Krueger, M.C., Kelly, S.A., Wasson, M.S., 2022. Integrated Reservoir Characterization Using Unsupervised Learning on Nuclear Magnetic Resonance (NMR) T1-T2 Logs. *Petrophysics - SPWLA J. Form. Eval. Reserv. Descr.* 63, 277–289. <https://doi.org/10.30632/PJV63N3-2022a1>
- Joshi, D.J., Kale, I., Gandewar, S., Korate, O., Patwari, D., Patil, S., 2021. Reinforcement Learning: A Survey BT - *Machine Learning and Information Processing*, in: Swain, D., Pattnaik, P.K., Athawale, T. (Eds.), Springer Singapore, Singapore, pp. 297–308.

## Bibliography

- Kingma, D.P., Ba, J.L., 2015. Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.
- Kleijnen, J.P.C., 2009. Kriging metamodeling in simulation: A review. *Eur. J. Oper. Res.* 192, 707–716. <https://doi.org/10.1016/J.EJOR.2007.10.013>
- Kroese, D.P., Botev, Z.I., Taimre, T., Vaisman, R., 2019. Data Science and Machine Learning, Data Science and Machine Learning. <https://doi.org/10.1201/9780367816971>
- Liang, X., Weber, D.B., Edgar, T.F., Lake, L.W., Sayarpour, M., Al-Yousef, A., 2007. Optimization of oil production based on a capacitance model of production and injection rates, in: SPE Hydrocarbon Economics and Evaluation Symposium. <https://doi.org/10.2523/107713-ms>
- Longstaff, F.A., Schwartz, E.S., 2001. Valuing American options by simulation: A simple least-squares approach. *Rev. Financ. Stud.* 14. <https://doi.org/10.1093/rfs/14.1.113>
- Lu, X.G., Xu, J., 2017. Waterflooding Optimization: A Pragmatic and Cost-Effective Approach to Improve Oil Recovery from Mature Fields. SPE/IATMI Asia Pacific Oil Gas Conf. Exhib. <https://doi.org/10.2118/186431-MS>
- Ma, H., Yu, G., She, Y., Gu, Y., 2019. Waterflooding optimization under geological uncertainties by using deep reinforcement learning algorithms, in: Proceedings - SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/196190-ms>
- Masoudi, R., Mohaghegh, S.D., Yingling, D., Ansari, A., Amat, H., Mohamad, N., Sabzabadi, A., Mandel, D., 2020. Subsurface analytics case study: Reservoir simulation and modeling of highly complex offshore field in Malaysia, using artificial intelligent and machine learning, in: Proceedings - SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/201693-ms>
- Mattax, C.C., Dalton, R.L., 1990. Reservoir Simulation, SPE Monogr. ed. Society of Petroleum Engineers. <https://doi.org/https://store.spe.org/Reservoir-Simulation-P70.aspx>
- Matthew, A., 2021. Proxy Modeling for CO<sub>2</sub>-EOR Design Study: Water Alternating Gas and Storage. Norwegian University of Science and Technology, Trondheim.
- McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*. <https://doi.org/10.2307/1268522>
- Meik, J.M., Lawing, A.M., 2017. Considerations and Pitfalls in the Spatial Analysis of Water Quality Data and Its Association With Hydraulic Fracturing. *Adv. Chem. Pollution, Environ. Manag. Prot.* 1, 227–256. <https://doi.org/10.1016/BS.APMP.2017.08.013>
- Mohaghegh, S., 2018. Data-Driven Analytics for the Geological Storage of CO<sub>2</sub>. CRC Press, Boca Raton, Florida.
- Mohaghegh, S., 2000. Virtual-intelligence applications in petroleum engineering: Part I - artificial neural networks. *JPT, J. Pet. Technol.* 52. <https://doi.org/10.2118/58046-ms>
- Mohaghegh, S.D., 2022. Smart Proxy Modeling: Artificial Intelligence and Machine Learning in Numerical Simulation. CRC Press, Boca Raton. <https://doi.org/10.1201/9781003242581>
- Mohaghegh, S.D., 2017a. Data-Driven Reservoir Modeling, Society of Petroleum Engineers.
- Mohaghegh, S.D., 2017b. Shale analytics: Data-driven analytics in unconventional resources, Shale Analytics: Data-Driven Analytics in Unconventional Resources. <https://doi.org/10.1007/978->

3-319-48753-3

- Mohaghegh, S.D., Amini, S., Gholami, V., Gaskari, R., Bromhal, G.S., 2012a. Grid-Based Surrogate Reservoir Modeling (SRM) for Fast Track Analysis of Numerical Reservoir Simulation Models at the Gridblock Level, in: SPE Western Regional Meeting. Society of Petroleum Engineers. <https://doi.org/10.2118/153844-MS>
- Mohaghegh, S.D., Liu, J., Gaskari, R., Maysami, M., Olukoko, O., 2012b. Application of well-based Surrogate Reservoir Models (SRMs) to two offshore fields in Saudi Arabia, case study, in: Society of Petroleum Engineers Western Regional Meeting 2012. <https://doi.org/10.2118/153845-ms>
- Muskat, M., Wyckoff, R.D., 1934. A Theoretical Analysis of Water-flooding Networks. *Trans. AIME* 107, 62–76. <https://doi.org/10.2118/934062-G>
- Niederreiter, H., 1992. Random Number Generation and Quasi-Monte Carlo Methods, *Random Number Generation and Quasi-Monte Carlo Methods*. <https://doi.org/10.1137/1.9781611970081>
- Odeh, A.S., 1982. An Overview of Mathematical Modeling of the Behavior of Hydrocarbon Reservoirs. *SIAM Rev.* 24. <https://doi.org/10.1137/1024062>
- Odeh, A.S., 1969. Reservoir Simulation ...What Is It. *J. Pet. Technol.* 21, 1383–1388. <https://doi.org/10.2118/2790-PA>
- Olson, R.S., Bartley, N., Urbanowicz, R.J., Moore, J.H., 2016. Evaluation of a tree-based pipeline optimization tool for automating data science, in: GECCO 2016 - Proceedings of the 2016 Genetic and Evolutionary Computation Conference. <https://doi.org/10.1145/2908812.2908918>
- Ozbayoglu, E., Ozbayoglu, M., Ozdilli, B.G., Erge, O., 2021. Optimization of flow rate and pipe rotation speed considering effective cuttings transport using data-driven models. *Energies* 14. <https://doi.org/10.3390/en14051484>
- Paul Willhite, G., 1986. *Waterflooding*, Vol. 3. ed. Society of Petroleum Engineers (SPE).
- Reis, J., Housley, M., 2022. *Fundamentals of Data Engineering: Plan and Build Robust Data Systems*, 1st ed. O'Reilly.
- Robertson, J.D., 1989. Reservoir Management Using 3D Seismic Data. *J. Pet. Technol.* 41, 663–667. <https://doi.org/10.2118/19887-PA>
- Russell, S., Norvig, P., 2010. *Artificial Intelligence A Modern Approach Third Edition*, Pearson. <https://doi.org/10.1017/S0269888900007724>
- Salehian, M., Sefat, M.H., Muradov, K., 2021. A robust, multi-solution framework for well placement and control optimization. *Comput. Geosci.* <https://doi.org/10.1007/s10596-021-10099-2>
- Satter, A., Iqbal, G.M., 2016a. An introduction to reservoir engineering: Advances in conventional and unconventional recoveries. *Reserv. Eng.* 1–10. <https://doi.org/10.1016/B978-0-12-800219-3.00001-2>
- Satter, A., Iqbal, G.M., 2016b. Petroleum reservoir management processes. *Reserv. Eng.* 137–153. <https://doi.org/10.1016/B978-0-12-800219-3.00008-5>
- Satter, A., Iqbal, G.M., 2016c. Waterflooding and waterflood surveillance. *Reserv. Eng.* 289–312. <https://doi.org/10.1016/B978-0-12-800219-3.00016-4>

## Bibliography

- Satter, A., Varnon, J.E., Hoang, M.T., 1998. Integrated reservoir management. SPE Repr. Ser. <https://doi.org/10.2118/22350-pa>
- Sayarpour, M., Zuluaga, E., Kabir, C.S., Lake, L.W., 2009. The use of capacitance-resistance models for rapid estimation of waterflood performance and optimization. J. Pet. Sci. Eng. 69. <https://doi.org/10.1016/j.petrol.2009.09.006>
- Schiozer, D.J., De Souza Dos Santos, A.A., De Graca Santos, S.M., Von Hohendorff Filho, J.C., 2019. Model-based decision analysis applied to petroleum field development and management. Oil Gas Sci. Technol. 74. <https://doi.org/10.2516/ogst/2019019>
- Schlaifer, R., 1959. Probability and Statistics for Business Decisions: An Introduction to Managerial Economics Under Uncertainty, 1st ed. McGraw-Hill, New York, NY.
- Schlumberger, 2019. Eclipse Reservoir Simulation Software Reference Manual, schlumberger.
- Shahkarami, A., Mohaghegh, S., 2020. Applications of smart proxies for subsurface modeling. Pet. Explor. Dev. [https://doi.org/10.1016/S1876-3804\(20\)60057-X](https://doi.org/10.1016/S1876-3804(20)60057-X)
- Shannon, C.E., 1948. A Mathematical Theory of Communication. Bell Syst. Tech. J. 27. <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>
- Slotte, P.A., Smørgrav, E., 2008. Response surface methodology approach for history matching and uncertainty assessment of reservoir simulation models, in: 70th European Association of Geoscientists and Engineers Conference and Exhibition 2008: Leveraging Technology. Incorporating SPE EUROPEC 2008. <https://doi.org/10.2118/113390-ms>
- Sobol', I.M., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput. Math. Math. Phys. [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9)
- Tadjar, A., Hong, A., Bratvold, R.B., 2021. A sequential decision and data analytics framework for maximizing value and reliability of CO2 storage monitoring. J. Nat. Gas Sci. Eng. 96, 104298. <https://doi.org/10.1016/J.JNGSE.2021.104298>
- Thakur, G.C., 1996. What is reservoir management? JPT, J. Pet. Technol. <https://doi.org/10.2118/26289-JPT>
- Thanh, H.V., Binh, D. Van, Kantoush, S.A., Nourani, V., Saber, M., Lee, K.-K., Sumi, T., 2022. Reconstructing Daily Discharge in a Megadelta Using Machine Learning Techniques. Water Resour. Res. 58, e2021WR031048. <https://doi.org/https://doi.org/10.1029/2021WR031048>
- Tian, C., Horne, R.N., 2017. Recurrent neural networks for permanent downhole gauge data analysis, in: Proceedings - SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/187181-ms>
- Tieleman, T., 2008. Training restricted boltzmann machines using approximations to the likelihood gradient, in: Proceedings of the 25th International Conference on Machine Learning. <https://doi.org/10.1145/1390156.1390290>
- Tom Mitchell, 1997. Machine Learning textbook, McGraw Hill.
- van Otterlo, M., Wiering, M., 2012. Reinforcement Learning and Markov Decision Processes BT - Reinforcement Learning: State-of-the-Art, in: Wiering, M., van Otterlo, M. (Eds.), . Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 3–42. [https://doi.org/10.1007/978-3-642-27645-3\\_1](https://doi.org/10.1007/978-3-642-27645-3_1)
- Vida, G., Shahab, M.D., Mohammad, M., 2019. Smart proxy modeling of SACROC CO2-EOR. Fluids. <https://doi.org/10.3390/fluids4020085>

## Bibliography

- Wiggins, M.L., Startzman, R.A., 1998. An approach to reservoir management. SPE Repr. Ser. <https://doi.org/10.2118/20747-ms>
- Willigers, B.J.A., Bratvold, R.B., 2009. Valuing oil and gas options by least-squares monte carlo simulation, in: SPE Projects, Facilities and Construction. <https://doi.org/10.2118/116026-PA>
- Wong, T.-T., Luk, W.-S., Heng, P.-A., 1997. Sampling with Hammersley and Halton Points. J. Graph. Tools 2. <https://doi.org/10.1080/10867651.1997.10487471>
- Yang, X.-S., 2014. Chapter 1 - Introduction to Algorithms, in: Yang, X.-S. (Ed.), Nature-Inspired Optimization Algorithms. Elsevier, Oxford, pp. 1–21. <https://doi.org/https://doi.org/10.1016/B978-0-12-416743-8.00001-4>

# **Collection of Papers**

# Paper 1

## *A Survey on the Application of Machine Learning and Metaheuristic Algorithms for Intelligent Proxy Modeling in Reservoir Simulation*

Cuthbert Shang Wui Ng, Menad Nait Amar, Ashkan Jahanbani Ghahfarokhi, Lars Struen Imsland





# A Survey on the Application of Machine Learning and Metaheuristic Algorithms for Intelligent Proxy Modeling in Reservoir Simulation

Cuthbert Shang Wui Ng<sup>a,\*</sup>, Menad Nait Amar<sup>b</sup>, Ashkan Jahanbani Ghahfarokhi<sup>a</sup>, Lars Struen Imsland<sup>c</sup>

<sup>a</sup> Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

<sup>b</sup> Département Etudes Thermodynamiques, Division Laboratoires, Sonatrach, Boumerdes, Algeria

<sup>c</sup> Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway

## ARTICLE INFO

### Keywords:

Machine Learning  
Metaheuristic Algorithms  
Data-Driven Modeling  
Intelligent Proxies  
Reservoir Engineering  
Numerical Reservoir Simulation

## ABSTRACT

Machine Learning (ML) has demonstrated its immense contribution to reservoir engineering, particularly reservoir simulation. The coupling of ML and metaheuristic algorithms illustrates huge potential for application in reservoir simulation, specifically in developing proxy models for fast reservoir simulation and optimization studies. This is conveniently termed the coupled ML-metaheuristic paradigm. Generally, proxy modeling has been extensively researched due to the expensive computational effort needed by traditional Numerical Reservoir Simulation (NRS). ML and the abovementioned coupled paradigm are effective in establishing proxy models. We conduct a survey on the employment of ML and the coupled paradigm in proxy modeling of NRS. We present the respective successful applications as reported in the literature. The benefits and limitations of these methods in intelligent proxy modeling are briefly explained. We opine that some study areas, including sampling techniques and dimensionality reduction methods, are worth investigating as part of the future research development of this technology.

## 1. Introduction

As global technology advances, the energy demand continues to evolve exponentially (Tillerson, 2008). This noticeable demand has made fossil fuels the dominant link in the energy subject area despite the continuous efforts made by the industrial sector to promote the vision and importance of renewable energies (British Petroleum, 2021). This source of energy, i.e., fossil fuels mainly from oil and gas reservoirs, goes through a step-by-step process to achieve the most desirable recovery factors. As a result, exploitation and development methods have been distinguished and classified into three categories, namely primary, secondary and tertiary recovery techniques (Ahmed, 2018). Fundamentally, these two latter techniques are designed to ensure the continuous production of hydrocarbons given the inefficacy of primary recovery. During primary recovery, the driving mechanism of hydrocarbon production originates from the natural source of energy associated with the rock and fluids in the reservoir. The mechanisms include expansion of liquids and reservoir rock, natural energy from aquifers and gas caps, expansion of dissolved gas, and gravity drainage. Secondary recovery

processes are often implemented by injecting water into the aquifer or injecting gas into the gas cap, to maintain the reservoir pressure. Recovery factors after primary drainage mechanisms and the implementation of secondary recovery techniques are generally moderate (Enick et al., 2012), hence there is a need for tertiary recovery techniques (Enhanced Oil Recovery, EOR) (Ahmadi et al., 2018). The latter aim to improve the recovery by acting on fluids and reservoir rock. Some of the most successful tertiary recovery techniques include water alternating gas injection, miscible CO<sub>2</sub> injection, polymer and surfactant injection, etc. (Afzali et al., 2018; Ahmadi et al., 2016; Dai et al., 2014; Ghriga et al., 2019; Vahdanikia et al., 2020; Xu, 1998). In addition to these three famous recovery stages of hydrocarbon reservoirs, other intervention strategies can be considered during the lifecycle of oil and gas reservoirs, mainly by infill drilling as well as the conversion of wells (e.g. producers into injectors, or vertical into horizontal) (Ding et al., 2020; Jesmani et al., 2020; Redouane et al., 2019).

The optimization of the recovery processes during the different recovery stages is crucial to optimize the techno-economic parameters such as Net Present Value (NPV) while taking into account the different

\* Corresponding author.

E-mail address: [cuthbert.s.w.ng@ntnu.no](mailto:cuthbert.s.w.ng@ntnu.no) (C.S.W. Ng).

<https://doi.org/10.1016/j.compchemeng.2022.108107>

Received 25 May 2022; Received in revised form 12 December 2022; Accepted 15 December 2022

Available online 17 December 2022

0098-1354/© 2022 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

constraints linked to production systems (pressure types such as Minimum Miscibility Pressure (MMP) and Bottom Hole Pressure (BHP) in miscible gas injection, as well as other production parameters such as water cut and Gas Oil Ratio (GOR), etc.) and the cost of the operation (cost of water injection, gas injection, well intervention operations, etc.) (Dai et al., 2014; Nait Amar et al., 2020c; Nait Amar and Zeraibi, 2019; You et al., 2020a, 2020b). Given the non-linearity of differential equations and thermodynamic models describing the different recovery processes, as well as the irregularity and heterogeneity of geometry (computational domain), the description and prediction of the evolution of key design parameters are commonly done numerically by using very powerful computing tools (Nait Amar et al., 2018a; Shahkarami et al., 2014a, 2014b). In this context, several commercial software such as Eclipse™ and CMG™ have been developed in the petroleum industry to allow a rigid optimization of the different tasks related to development strategies of reservoirs and production, while integrating advanced computing paradigms such as black oil, compositional, and streamline approaches. However, the optimization of a process described by a highly non-linear model with non-linear constraints and dependent on a significant number of parameters is very complex even using these advanced simulation tools (Panjalizadeh et al., 2015). Carrying out a direct simulation scenario with the latter for cases close to reality takes time and very efficient computing means (multiprocessors, parallel computing, etc.).

All the aforementioned technical constraints have led a great part of the petroleum community to investigate new alternatives which enable the same problems to be solved with considerable precision but with means that are not binding in terms of calculation time (Ertekin and Sun, 2019; Mohammadi and Ameli, 2019). Among these alternatives, Data-Driven Modeling (DDM) has gained increasing interest in the field of reservoir simulation. Approaches to DDM are generally statistics-based (or mathematics-based), e.g., the surface response method, and Machine Learning (ML) based. DDM may alternatively be known as proxy modeling while proxy model development englobes other approaches such as reduced-order modeling which mainly involve simplification of problems and are not purely data-driven.

The word proxy means to act on behalf of another. This definition has a projection on the technical or numerical sense of proxy models (also known as surrogate models). These are models built from data exploited from numerical simulations, capable of reproducing the simulator's responses with very high precision at a speed of execution that is in the order of a few seconds (Zubarev, 2009). These models have had vast use since the beginning of the 21st century in various areas. The use of proxy models has quickly been proposed in the field of reservoir engineering where there is a wide application of proxy models as substitutions for commercial software in various vital tasks such as well placement optimization (Hassani and Sarkheil, 2011; Sayyafzadeh, 2015a; Zarei et al., 2008), history matching (Sayyafzadeh, 2015b; Shahkarami et al., 2014b), and uncertainty studies (Mohaghegh et al., 2012a, 2006).

As proxy modeling can be regarded as a kind of pattern recognition and functionality identification, the model construction can be done with interpolation methods and Artificial Intelligence (AI) and ML methods. In this aspect, AI can be perceived as technology or tools that simulate the human brain and logic to perform analysis or any assigned task whereas ML denotes the application of computer algorithms to enable learning through data (Mohaghegh, 2018, 2017a, 2017b). Thus, ML is the subset of AI. The effectiveness of a proxy is very dependent on the robustness of the technique used for its elaboration (Na-udom and Rungtattanaubol, 2015; Zubarev, 2009). The robustness of an ML technique can touch upon various aspects, specifically the training procedure including the evolved relevant model parameters to improve the training and the considered mathematical operators (e.g., back-propagation process) in the calculation process. Artificial Neural Networks (ANNs), Support Vector Machines (SVM), kriging, and Response Surface Models (RSM) are among the widely applied techniques for building proxy models in the oil industry. In general, the first two

approaches are ML-based whereas the last two are statistics-based. In this paper, our focus is on the ML-based proxy, also known as an intelligent or smart proxy<sup>1</sup>. It is worth mentioning that before proceeding to the building stage of the proxy, a primordial step consisting of generating a set of points or a database should be done properly. The judicious choice for sampling of the points will bring precision and generalization to the built model because the chosen sampling method tries to capture a wide variety of information about the inputs/responses of the simulators (Yeten et al., 2005). Design of Experiments (DoE) is the statistics branch assembled with proxy models through its methods (Crombecq, 2011; Forrester et al., 2008; Zubarev, 2009). Several works comparing different DoE methods have been published (Crombecq, 2011; Viana, 2016; Yeten et al., 2005). The main conclusion that can be retrieved from applying DoE in the building phase of proxy models is that space-filling techniques, such as Latin Hypercube Design (LHD), are one of the most efficient methods for building rigorous proxy paradigms. The details of the paradigm of intelligent proxy will be delineated later.

The optimization of different complex processes in the oil industry, such as EOR techniques, is a crucial step in reservoir management that significantly affects the efficiency and production strategy (Yazdanpanah and Hashemi, 2012). Several time-dependent parameters and the management procedure should be optimized in such projects (Yazdanpanah and Hashemi, 2012). Thus, traditionally the optimization methods evaluate hundreds or even thousands of potential scenarios to search for the optimal solution, using time-consuming numerical simulations. To deal with this issue which includes the significant calculation time and the considerable number of simulation runs, coupling metaheuristic algorithms with a powerful clustering-based proxy model is generally considered a better alternative for non-linear and multidimensional problems (Onwunali et al., 2008). Metaheuristic algorithms are population-based optimization techniques that consider a predefined criterion (fitness function) to distinguish between the performance of the individuals mimicking the scenarios of the problem. The gain of this kind of coupling is ensured by the exploitation of the advantages of the two approaches, namely the reduced calculation time of the proxy models, and the oriented and targeted runs to perform based on the fitness function of the metaheuristic algorithms. As discussed in this reference (Onwunali et al., 2008), a proxy model is employed to approximate the objective function values of different scenarios. When the estimated values exceed a certain threshold, the respective scenario will be chosen for simulation and optimization. Besides, it is worth mentioning that a smart proxy that is built using a significant number of numerical simulations can be used for dealing with uncertainties as the generated information is generally widespread and it involves an extensive number of interactions between the main parameters of the model for covering this kind of tasks.

Metaheuristic algorithms are the optimization algorithms we would like to emphasize in this work. Metaheuristics algorithms can be defined as mathematical frameworks with advanced searching mechanisms in the solution space (Gogna and Tayal, 2013; Wong and Ming, 2019; Yang et al., 2014). The advanced searching mechanisms of metaheuristic algorithms consist of the exploration and exploitation steps which involve specific operators that help the orientation of the optimization process towards regions of interest within the search space (Hemmati-Sarpardeh et al., 2020b). Exploration refers to inspecting the unexplored parts of the search space, while exploitation corresponds to the search of the neighborhood of the promising area (Tilahun, 2019). In general, these algorithms are derivative-free and nature-inspired. Examples of these algorithms include Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), Firefly Algorithm (FA), Imperialist Competitive Algorithm (ICA), Simulated Annealing (SA), Gray Wolf

<sup>1</sup> To avoid confusion, "intelligent proxy" (or intelligent model) and "smart proxy" models share the same definition in this paper.

Optimization (GWO), Cuckoo Optimization Algorithm (COA), etc. These algorithms have demonstrated their robustness in many areas of application, including prediction of stocks, image processing, bioinformatics, etc. (Gogna and Tayal, 2013).

In terms of reservoir simulation, metaheuristic algorithms have been extensively and successfully employed not only to train different types of proxies but also to solve optimization problems (coupled with either numerical models or proxies). For clearer perusal, implementation of metaheuristic algorithms in the establishment of ML-based proxies and resolution of optimization problem is conveniently termed the coupled ML-metaheuristic paradigm. Based on our studies (Nait Amar et al., 2021, 2020c; Ng et al., 2021a), the paradigm illustrated excellent results of implementation in developing ML-based proxy models where the metaheuristic algorithms were used for training. Additionally, optimization problems can be handled efficiently by applying the coupled ML-metaheuristic paradigm where this paradigm achieves optimum results within reasonable calculation time. Therefore, it is important to have a survey of how useful ML methods are to establish intelligent proxies when being solely employed or coupled with metaheuristic algorithms. Moreover, we opine that there is a necessity to provide this survey since there is not much available discussing these domains together.

This survey paper covers a wide range of research studies related to the application of ML techniques and the coupled ML-metaheuristic paradigm in intelligent proxy modeling. This work will contribute to the research and development related to various reservoir simulation applications mainly by shedding light on the smart schemes and intelligent methods based on ML and metaheuristic algorithms that were implemented for reducing the calculability efforts associated.

The rest of the paper is formulated as follows: Section 2 provides a brief discussion regarding some of the previous literature and reviews on the relevant topics. Section 3 demonstrates the general framework that can be employed to develop an intelligent model. Thereafter, Section 4 briefs several examples of the application of intelligent proxies and the coupled ML-metaheuristic paradigm in the context of reservoir simulation. Section 5 outlines the benefits and limitations of these paradigms as well as the associated challenges in the research domain before ending this survey paper with concluding remarks.

## 2. Previous Works

As briefly mentioned, Data-Driven Modeling (DDM) is considered another modeling approach aside from traditional physics-based modeling. The availability of a large database in petroleum engineering (Mohammadpoor and Torabi, 2020) has, to a certain extent, contributed to the prevalence of data-driven models as data is one of the main building blocks for the use of ML (Mohaghegh, 2022). Explicitly speaking, these data are applied to develop a model that can provide useful insights to petroleum engineers to do some engineering judgments. In the domain of reservoir engineering, DDM has provided a fast and efficient alternative for reservoir simulation (Mohaghegh, 2017a). More intriguingly, the coupling of the metaheuristic algorithms with ML-based data-driven models is another topic that is worth a discussion. To have a better outlook on the development of ML and metaheuristic algorithms<sup>2</sup> in the oil and gas industry, we will briefly discuss some relevant previous works and review papers.

<sup>2</sup> Based upon our survey of the literature, there are not many papers that solely discuss the coupling of metaheuristic algorithms with ML in the petroleum industry. Thus, in this survey paper, apart from explaining the use of ML, one of our discussions is intended to focus on how metaheuristic algorithms can be effectively implemented along with ML mostly in the context of reservoir simulation.

### 2.1. Proxy Modeling

DDM is considered proxy modeling in the aspect of reservoir simulation. Using the proxy model as the substitute for Numerical Reservoir Simulation (NRS) has been applauded due to its quick computation and satisfactory accuracy of results (Mohaghegh, 2022; Nait Amar et al., 2021; Ng et al., 2022a). A simple illustration is displayed in Fig. 1 to outline the relationship between proxy modeling and other terminologies that would be expounded on in the following subsections. The terminologies, such as Subsurface Data Analytics, Top-Down Modeling (TDM), and Smart Proxy Modeling (SPM), will be explained in detail in Section 2.3. ML is one of the approaches to proxy modeling. Zubarev (2009) provided a comparative analysis regarding the effectiveness of four different techniques of proxy modeling as the substitute for complete reservoir simulations. These methods included polynomial regression, multivariate kriging, thin-plate splines, and ANNs. He inferred that in history matching, the proxy models could perform reasonably well in a deterministic case but not in a probabilistic fashion. In the optimization of infill-drilling, the proxy models also illustrated reasonable performance, but the solutions were not optimal. Nevertheless, these models demonstrated excellent performance in terms of prediction of initial hydrocarbons in-place and oil recovery. In general, he stated that kriging models outperformed the others but induced the highest computational footprint. There was another constructive comment that the proxy modeling methods heavily relied upon the sophistication of the model, size of the design space, and quality of input data. This gives us a very well-established cognizance of the limitations or constraints that proxy modeling methods are subject to (Zubarev, 2009). He also opined that the option of proxy modeling methods was problem-dependent and quantifying the errors induced by proxy modeling techniques was needed for quality assurance.

Moreover, Jaber et al. (2019a) conducted a detailed review of the application of proxy modeling in NRS. They summarized that there were two general approaches employed to develop proxy models, which included virtual intelligence and statistical method. Fundamentally, the proxy models were aimed at simplifying the complexity of the physical process regarding uncertain variables and assessing the responses rapidly with reasonable accuracy (Jaber et al., 2019a). The authors expounded that ANN, Fuzzy Logic, and GA were among the prevalent virtual intelligence methods used to build proxy models whereas RSM was the common statistical method in this context. In addition, they discussed several pieces of literature that illustrated the successful applications of virtual intelligence-based proxy models in assisted history matching and forecasting reservoir performance, and statistics-based proxy models in uncertainty analysis and prediction of reservoir response. They also outlined the proper step for validating and evaluating the quality of models. They further argued that virtual intelligence methods coupled with NRS were unable to simultaneously capture the effect of interactions among different uncertain variables. Hence, they opined that statistics-based proxies in general outperformed virtual intelligence-based proxies. More rivetingly, they shared the same opinion with Zubarev (2009) that understanding the use of a proxy was essential in choosing the right method, and evaluating the quality of proxies was highly recommended.

### 2.2. Implementation of ML

Apart from review papers about proxy modeling, several works expound on the general trend of the implementation of ML in the oil and gas industry. Li et al. (2020) provided an interesting insight into how rapidly the transition from digital oilfield to AI oilfield has taken place. Concerning this, they further outlined the pros and cons of different ML algorithms, including ANN, PSO, Fuzzy Logic, SVM, and GA. Thereafter, they discussed the efficient employment of AI in different aspects of the petroleum industry, e.g., history matching, dynamic prediction of production, optimization of a development plan, identification of oilfield

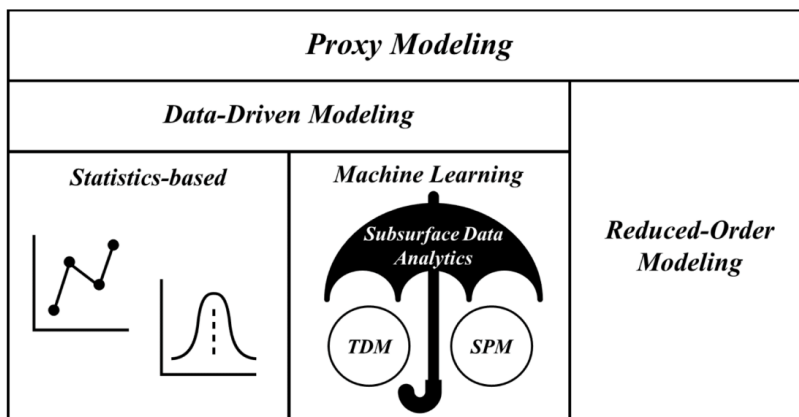


Fig. 1. Schematic of the relationship between proxy modeling and other terminologies.

development, detection of fracture, and EOR. In general, they inferred that compared to the other AI algorithms, ANN was the most prevalently used in the petroleum industry. Appropriate selection of the algorithm was also the solution to certain limitations of the algorithms. They further added that AI algorithms were too data-oriented and marginalized the physics of the process. More importantly, they pointed out that having the capacity to use and integrate big data of the oilfield with intelligent models at different phases was pivotal to ensuring the success of the AI oilfield.

Moreover, Ertekin and Sun (2019) conducted a painstaking status check on the implementation of AI in reservoir engineering. They presented different reservoir engineering-related research works, for instance, proxy modeling, AI-assisted history matching, and optimization of project design, which highlighted the robustness of the AI system. From this, they opined that the formulation of AI models could be divided into two distinct categories: forward and inverse-looking models. Additionally, data could be categorized into three groups, namely reservoir characteristics, project design parameters, and field responses. Perceiving the types of formulation and the associated data could provide a clearer understanding to the reservoir engineers in applying the AI approaches. Nonetheless, they arose the lack of astuteness of AI methods in completely replacing the traditional reservoir engineering models. Thus, they encouraged the hand-shaking protocol between the traditional modeling and the intelligent paradigm to fully exploit the respective advantages of each method and produce a more robust solution to reservoir engineering problems.

Furthermore, Balaji et al. (2018) evaluated the status and implementation of data-driven approaches, including ML, in the oil and gas industry. They first explained different data-driven techniques: linear regression, principal component analysis (PCA), decision tree, SVM, ANN, Fuzzy rule-based systems, GA, and Bayesian Belief Networks. Then, they showed how these methods were used in cases like subsurface characterization and petrophysics, drilling, production, reservoir studies and EOR, facilities, remediation and management, and pipelines. Pros and cons in tandem with the reasons for acceptance (as well as rejection) of these methods in the industry were also touched upon. More specifically, Alkinani et al. (2019) provided a review of the employment of ANN in the industry. They showed the basic steps in ANN modeling: collection and selection of input data, partitioning of data, normalization of data, and determination of the number of hidden layers and training algorithm. Also, they discussed the successful application of ANN in exploration, drilling, production, and reservoir engineering. In addition, Hanga and Kovalchuk (2019) thoroughly discussed the

applications of ML and Multi-Agent Systems (MAS) in the petroleum industry. ML was proven to be effective in production, anomaly detection, and price detection while MAS was applied successfully in production, safety and maintenance, and supply chain management. They also stated how ML and MAS could be used interchangeably in various petroleum industry tasks and discussed the hybridization of both for better implementation. Apart from these, Otchere et al. (2021) did a detailed review of different pieces of literature to compare the application of ANN and SVM models in the forecasting of properties of petroleum reservoirs (mainly seismic and well log applications). They inferred that in the domain of reservoir characterization with limited data and in terms of coupling with other algorithms, SVM was found to outperform ANN.

### 2.3. Subsurface Data Analytics

Despite still having a lack of astuteness, the application of AI in petroleum engineering, especially for reservoir engineering, has gradually achieved enviable breakthroughs and maturity thanks to the contribution of the research group led by Dr. Shahab Mohaghegh. In this aspect, Mohaghegh (2011) explained the complete workflow that has been formulated to exploit the pattern recognition capabilities of AI in building an AI-based model that could act as a substitution for NRS. In this work, a constructive comment that was different from that of Li et al. (2020) regarding the use of physics was presented. He articulated that the use of physics was preserved through the generation of a spatio-temporal database. In simpler terms, it was denoted that the physics of the system was represented by the data. Hence, applying data with the help of AI to develop a model does not ignore physics. He further stated that the existing physical models (and statistical approaches) involved a lot of underlying assumptions which could have simplified the physics of the real problems. He has been consistently championing the utilization of data and AI because of his strong belief that the oil industry is heading toward the fourth paradigm of science, which is data-intensive science (Mohaghegh, 2020, 2011). Thus, he has systematized the whole idea of employing petroleum-related data in the establishment of models and coined it "Subsurface Data Analytics". In general, the benefits of Subsurface Data Analytics over NRS, including circumvention of preconceived notions, biases, and simplifications of problems, have been highlighted. Mohaghegh (2020) expounded a deep concern regarding the hybrid models (combination of physics-based and AI-based approaches) and opined that hybrid modeling was the conventional statistical approach. The reasons for building hybrid models

were assumed the lack of ability in developing good models by only applying ML techniques, employing it as a marketing tool, lack of ability in explaining the results produced by the ML-based models, and lack of ability in responding to the challenges imposed by the conventionalists in the industry.

Under the umbrella of Subsurface Data Analytics, there are two main classes of modeling, which are Smart Proxy Modeling (SPM) and Top-Down Modeling (TDM). According to Mohaghegh (2018, 2017a, 2017b), the formulations of both TDM and SPM share the same fundamental idea and methodology. Both models are defined as an ensemble of Neuro-Fuzzy systems that can learn and recognize the hidden pattern of the data provided. The only subtle difference is the source of the data used. For SPM, the data come from the spatio-temporal database generated by NRS whereas the spatio-temporal database for TDM originates either from the field data or the combination of field and simulation data. Regarding the functionalities of these two types of proxies, the smart proxy model is mainly implemented to reduce the computational effort induced by NRS while producing outputs within a satisfactory level of accuracy (Mohaghegh, 2018, 2022). This rapid and accurate assessment can help reservoir engineers to elude wasting extra time in making some reservoir management-related decisions. Besides that, the relevant details of TDM have been outlined in this literature (Mohaghegh, 2017a). It is a completely different method of modeling a subsurface as compared to NRS using a bottom-up approach. In general, TDM is applied to develop a model that can better decipher the behavior of the reservoir system. Both SPM and TDM are useful in different reservoir engineering tasks, including history matching (He et al., 2016; Shahkarami et al., 2018), CO<sub>2</sub> storage and sequestration (Mohaghegh, 2018), CO<sub>2</sub>-EOR (Shahkarami and Mohaghegh, 2020; Vida et al., 2019), and shale analytics (Mohaghegh, 2013). There is also an associated challenge with both TDM and SPM in which the curse of dimensionality will happen as the size of the spatio-temporal database increases. In this case, Mohaghegh (2018, 2017a, 2017b) initiated the use of fuzzy pattern recognition to determine the degree of influence of each possible parameter on the output in terms of Key Performance Index (KPI). The ranked KPIs aid in selecting the input variables. Using fuzzy logic is preferred when calculating the KPIs of input variables because it can model uncertainties associated with vagueness or lack of information as discussed in these references (Mohaghegh, 2018, 2017a, 2017b; Ross, 2010).

The generation of massive data, fathomed as “Big Data”, in the upstream and downstream petroleum industry has also played an integral part in the emerging trend of the use of ML in the industry. In this case, Mohammadpoor and Torabi (2020) illustrated a comprehensive review of how Big Data analytics has been effectively utilized in the industry. They expounded on six characteristics of Big Data that included volume, velocity, variety, veracity, value, and complexity. They outlined the general methodology of Big Data and explained the tools that could be used to perform Big Data analytics. They also presented different examples to demonstrate how it was implemented in different aspects of upstream, such as exploration, drilling, reservoir engineering, and production engineering. Examples of downstream were also provided, e.g., refining, oil and gas transportation, and health and safety execution. Besides that, Temizel et al. (2016) explained the general steps involved in Data Mining and the development of data-driven models through the illustration of a synthetic case. Apart from briefly explaining the use of statistics-based and ML-based methods in Data Mining, they also conveyed the fundamental thought of how data could be useful in terms of modeling if being systematically used. More intriguingly, Ani et al. (2016) discussed the importance of applying uncertainty analysis (probabilistic approaches) in reservoir modeling compared with the deterministic approach. In this context, they added that the use of ML would have a significantly positive impact on the future trend of uncertainty analysis.

#### 2.4. Application of Metaheuristic Algorithms

Based on our investigation, the literature comprehensively reviewed the successful use of metaheuristic algorithms in different domains. However, there are only a handful of studies that examined their application along with ML, especially in the field of petroleum engineering. The metaheuristic algorithms discussed in this paper are mainly nature-inspired. We opine that these algorithms are robust in terms of implementation. They are not only widely used in optimizing the hyperparameters of the intelligent models (Hemmati-Sarapardeh et al., 2020a; Nait Amar et al., 2018b; Nait Amar and Zeraibi, 2018; Ng et al., 2022b, 2021c), but also in solving petroleum engineering-related optimization problems (Nait Amar et al., 2021, 2018a; Ng et al., 2021b; Wang et al., 2021). Hemmati-Sarapardeh et al. (2020b) included an extensive explanation of the mechanism of different metaheuristic algorithms, such as GA, PSO, ACO, ABC, FA, and GWO. They also illustrated how these algorithms could be coupled with different intelligent models and employed in different domains like reservoir and production engineering, drilling engineering, and exploration. Moreover, Plaksina (2019) performed a similar review but with more emphasis on evolutionary computation, swarm intelligence, fuzzy logic, different types of ML, and ANN. She included a lot of petroleum-related applications concerning the abovementioned areas to illustrate the robustness of AI approaches. Also, Rahmanifard and Plaksina (2019) reviewed and explained different optimization approaches, such as GA, DE, and PSO in tandem with ANN and fuzzy logic. They also provided some discussions to outline the applications of these methods in the petroleum industry.

### 3. Paradigm of Intelligent Proxy Development

In this section, we will brief the general framework used in establishing intelligent proxy models in the context of reservoir simulation. This framework is a product of assimilating different workflows proposed in several pieces of literature (Hemmati-Sarapardeh et al., 2020b; Mohaghegh, 2017a; Russell and Norvig, 2010). In this aspect, when ML methods are implemented to perform proxy modeling, it can be termed as either “smart” or “intelligent”. The word “smart” or “intelligent” indicates the capability of the model to learn and decipher the hidden pattern or relationship between the input and output data provided using the ML methods. Metaheuristic algorithms can act as training algorithms to help the models learn better. Their robustness is demonstrated as they can conveniently be coupled with the built intelligent models to solve optimization problems. As mentioned earlier, data act as the most essential element required to build the intelligent proxy model. Hence, it is of paramount importance that the data provided to the proxy correctly capture and represent the physics of the system being modeled. Besides that, we need to understand that the intelligent proxy is never a one-size-fits-all model. The fundamental paradigm of building an intelligent proxy is summarized in Fig. 2.

The first step of the paradigm is to identify the purpose of the proxy and carefully *formulate the problem*. This is important because it provides a clear idea regarding the type of database that needs to be generated or extracted. Having defined the optimization problem clearly, the reservoir engineer would have a better perception of the data needed to develop the corresponding proxy model. It is also vital to emphasize that the number of proxy models required depends upon the complexity of the formulated problem. The important takeaway of this step is that one should be cognizant of the problem to be solved, define it clearly, and ensure the proper variables or parameters needed to build the proxy. Besides that, selecting the appropriate AI methods is another consideration in this step. Such appropriateness can be determined by the capability of the selected method to mathematize the relevant engineering problem as a functional relationship.

There are two main categories of input variables for reservoir simulation, namely static and dynamic input data. Examples of static data include porosity, permeability, and thickness of the formation

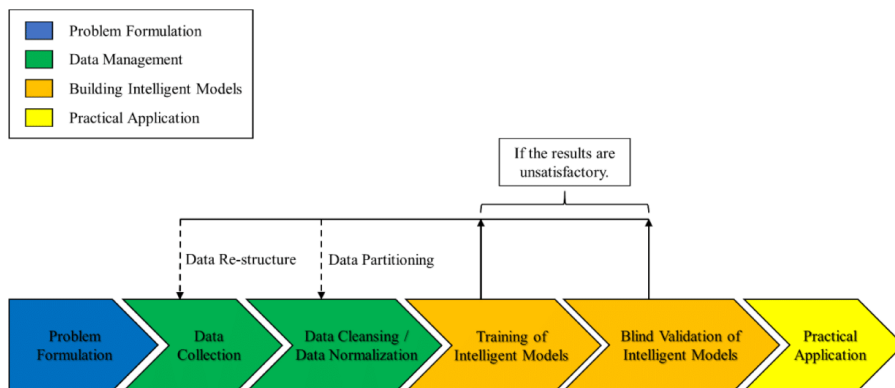


Fig. 2. Paradigm of Intelligent Proxy Development.

layer. Dynamic data consist of production rate, well bottom hole pressure, and saturation. It is important to understand that if a dynamic parameter is considered one of the input variables, one might require to develop a proxy that can forecast this dynamic variable. Thereafter, the predicted dynamic parameter should be fed into the initial proxy to reduce the dependency on the use of NRS. This type of proxy design is termed “cascading design” (Mohaghegh, 2017a). Nonetheless, one should be aware of the possibility of accumulation of prediction error when the “cascading design” is employed. Therefore, these points of discussion ought to be pondered ahead during the phase of problem formulation to ensure a smooth process of proxy development in later stages.

Then, as we proceed to **Data Management**, we need to understand the types of data that should be obtained and identify the sources of data to retrieve the database (NRS, field measurements, or both). In this paper, our discussion concentrates on the use of data generated by NRS. To generate the database from NRS, we need to design several scenarios of simulation runs. Thus, we implement a sampling strategy to extract several samples (of for example rates) within the predefined operational range and define them as simulation scenarios. Each scenario is equivalent to one simulation run. Based on our survey, there is no specified number of runs required to create the database. Theoretically, the higher the number of simulation scenarios, the higher the chances that the solution space of the optimization problem is covered. Nonetheless, this will cause the curse of dimensionality. So, the choice of sampling strategy plays a vital role in ensuring the success of proxy modeling. Examples of renowned sampling methods include Latin Hypercube sampling (McKay et al., 1979), Halton sequence (Halton, 1960), Sobol sequence (Sobol', 1967), and Hammersley sequence (Hammersley and Handscomb, 1964). The selection of input data (also termed feature selection) is another consideration in this step. During problem formulation, we would have known the output data that our developed proxy models can generate. It is important to identify the input variables with a larger degree of influence on the output. In terms of NRS, the selection of useful input variables is indeed essential because including too many of them will induce the curse of dimensionality. There are three approaches to this selection, namely empirical selection, statistical methods, and AI-based methods. The first selection relies upon common knowledge of reservoir engineering. Moreover, several statistical metrics, e.g., percentile of the highest score, k highest score, and chi-squared test are employed to select the useful input parameters. AI-based approaches such as fuzzy pattern recognition have shown successful and robust applications in choosing the input variables (Mohaghegh, 2018, 2017a, 2017b, 2011). According to our investigation, any of these three

methods can contribute to the successful development of proxy models. However, Mohaghegh (2018, 2017a, 2017b) opined that fuzzy pattern recognition outperformed the statistics-based approaches in this context.

Before feeding the database into the model, data cleansing occasionally might be needed to remove any noisy data or outliers which can affect the learning of the models. This is normally done on real field data. For NRS, data cleansing is not needed. Data normalization is another highly recommended step before proceeding to the training of intelligent models, where the values of the database will be rescaled within a smaller range of values, generally either  $[0, 1]$  or  $[-1, 1]$ . Our survey based upon numerous papers (Hemmati-Sarapardeh et al., 2020b; Nait Amar et al., 2019, 2018b) confirms that data normalization is very common to ensure that the intelligent models can capture the pattern induced by the database. In this case, we highly suggest conducting “categorical normalization”. For instance, when there are several columns of input data indicating the same category of data such as porosity, the maximum and minimum values of the datapoint should be chosen from the same category for normalization. After the completion of this phase, the database is deemed ready to be implemented for training the intelligent proxy models.

In the step of **building intelligent models**, the fundamental idea is to enable the intelligent models<sup>3</sup> to learn and capture the physics of the system. Concerning this, it is important to perceive the definitions of model parameters and model hyperparameters (Yang and Shami, 2020). Model parameters refer to the ones that can be initialized and updated through the training process (viz. weights and biases for ANN). Model hyperparameters must be initialized before training and are related to the architecture of ANN, for instance, the number of hidden layers and nodes, learning rate, and dropout rate (Yang and Shami, 2020). Searching for the optimal model hyperparameters, alternatively known as hyperparameter optimization, can be performed to ensure better learning ability of an intelligent model during training. The algorithm selected to perform such optimizations will iteratively tune the model parameters and model hyperparameters to minimize a predefined loss function until a stopping criterion is met. Examples of the loss function can be the Mean Squared Error (MSE), Mean Absolute Error (MAE), Average Percent Relative Error (APRE), and Average Absolute Percent Relative Error (AAPRE). In general, the algorithms used are categorized into two groups, including derivative-based and derivative-free. Examples of derivative-based algorithms include stochastic gradient descent,

<sup>3</sup> An example of intelligent models of interest here is ANN. However, the methodology also applies to other ML-based models.

scaled conjugate gradient, Levenberg Marquardt, and Adaptive Moment Estimation whereas derivative-free algorithms are mainly nature-inspired, such as Genetic Algorithms and Particle Swarm Optimization. Also, combining both can be another option (Nait Amar et al., 2018c, 2018b).

The database needs to be partitioned into for instance three different sets, namely training, validation, and testing<sup>4</sup>. Albeit there is no rule for partitioning ratio, most of the literature (He et al., 2016; Mohaghegh, 2017a; Ng et al., 2022b, 2021c; Shahkarami and Mohaghegh, 2020) used either the ratio of 7:1.5:1.5 or 8:1:1. After the partitioning is done, the training data should be fed into the intelligent model to undergo the training phase. During this phase, for every iteration, the performance metrics of validation are evaluated to check if the overfitting issue occurs. Regarding this, we can infer that the overfitting issue is eluded if decreasing trends of loss function for both training and validation data are observed. If such a trend is not noted, training needs to be repeated. Refer to Shahkarami and Mohaghegh (2020) for the pertinent details. Nonetheless, before repeating the training, the dataset can be re-partitioned to evaluate if better training results can be yielded. However, such re-partitioning is regarded as bad practice by Russell and Norvig (2010). Thus, the whole training process can be performed by either adding new data points or using a completely new set of data (termed data re-structure)<sup>5</sup>. When the overfitting issue is assured to be prevented, we can deduce that the trained intelligent model has passed the first stage of quality assessment.

Training and validation performance can be assessed using the metrics used as the loss function in addition to the coefficient of determination,  $R^2$ . The use of APRE and AAPRE needs attention, especially during the establishment of a proxy model that predicts the water production rate. This is because the water production rate is zero just before the water breakthrough, given the initial water saturation is equal to the immobile water saturation. Thus, it can be cumbersome to implement APRE and AAPRE to evaluate the performance of proxy models before the water breakthrough. In this case, the testing data is fed into the model to evaluate the testing phase performance. This phase is to ensure that the trained model portrays a good level of predictability before being blind-validated, which is the last stage of quality evaluation. In blind validation, it is important to note that the blind data should not have been part of the training, validation, and testing data. Additionally, it is highly recommended to ensure that the blind validation dataset falls within the range of the previously generated database. This is because according to some literature (Barnard and Wessels, 1992; Haley and Soloway, 2003; Xu et al., 2020), intelligent models generally perform well in interpolation but not in extrapolation. If the result of blind validation is excellent, then it denotes that the model has good predictability to serve its purpose and is ready for practical application. Nevertheless, if the blind validation results are not satisfactory, data re-partitioning or data re-structuring can be considered. Generally, these three phases of the quality assessment provide insights to confirm that the model can serve its objective.

For the case of hyperparameter optimization, based on our study (Ng et al., 2021a), using the weighted sum of the training, validation, and testing errors are recommended. The respective weighting factors can be treated as additional parameters to be optimized. Also, one needs to understand that performing such optimization tasks will require additional time, proportional to the size of the database (Shahkarami and Mohaghegh, 2020). Therefore, there is a trade-off to consider when it comes to conducting the optimization. It is also important to know that the models can be divided into static and dynamic types. Static proxy

models are usually built to predict specific variables over a whole period. For instance, a model that forecasts the NPV of a certain production period considering several input variables. This type of proxy is not robust in terms of application despite the ability to speed up the computation. Dynamic proxy models are established to forecast variables at certain timesteps. Albeit building them can be more laborious than static proxies, dynamic proxies offer higher flexibility in terms of application, including prediction of specified output and optimization (Nait Amar et al., 2018a). It is, therefore, necessary to highlight the distinction between these two types of proxies that helps one to have a better perception at the beginning of proxy modeling. Some examples of **practical applications** will be discussed in Section 4.

## 4. Survey of Applications

Applications of ML and coupled ML-metaheuristic paradigm in different domains of reservoir engineering, mainly in the areas that implement reservoir simulation, will be discussed here. Fig. 3 illustrates the examples of domains that are surveyed in this section. Due to the limited use of coupled ML-metaheuristic paradigm, emphasis is on ML in several application examples. A few interesting works (discussing only the use of metaheuristic algorithms or their applications with other variants of proxy models, e.g., reduced-order modeling) have also been included in this section. The summary of the collected literature is demonstrated at the end of each subsection along with the methods used as well as the assumptions and limitations discussed in each work. Refer correspondingly to Table 1 to 8 for the summary of the literature on each subsection.

### 4.1. Well Placement

Optimizing well placement is one of the most challenging tasks in field development planning. This is because multiple scenarios of NRS need to be run to determine the best location to place the wells. The computational efforts will increase when the geological uncertainty of the reservoir being modeled is considered for better decision-making. The optimization task can be cost-effective if the computational time can be shortened. Several pieces of the literature suggest the application of ML approaches as the potential solution. Additionally, the coupling of the simulation models or the respective proxy models (built using ML) with the metaheuristic algorithms has shown some promising results.

Nwachukwu et al. (2018a) performed a handful of NRS to generate training data and implemented the Extreme Gradient Boost (XGBoost) approach to establish a model that could provide a fast forecast of the responses of a reservoir based upon the locations of injectors. In addition, they employed the Fast-Marching Method (FMM) to introduce the well-to-well connectivity to the model and this enhanced the results significantly. The methodology was used in the cases of waterflooding and CO<sub>2</sub> flooding. Thereafter, Nwachukwu et al. (2018b) extended this ML approach to optimize the location of wells and the parameters of WAG injection by coupling the model with a novel optimization algorithm, namely Mesh Adaptive Direct Search (MADS). Xiong and Lee (2020) applied the ANN modeling to build a model to estimate the production of fluids based on reservoir heterogeneity and well locations. Then, they used this model to determine the optimal location of injectors in the case of waterflooding. Chu et al. (2020) discussed the use of Convolutional Neural Network (CNN) to develop three different models, single-, dual-, and multi-modal CNNs, in the optimization of infill well locations. They also compared these models with a Feedforward Neural Network (FNN). Jang et al. (2018) proposed the sequential employment of ANNs to determine the optimal well location in a coalbed methane (CBM) reservoir. They inferred that the sequential ANNs computationally outperformed the direct use of PSO algorithms in the same optimization problem.

Sayyafzadeh (2015a) presented a self-adaptive surrogate-assisted evolutionary algorithm to determine the optimal location of wells. This

<sup>4</sup> Alternatively, Mohaghegh (2017a) uses the terms calibration and validation datasets for validation and testing dataset, respectively.

<sup>5</sup> It relies upon personal preference if re-partitioning of data should be attempted. In this work, our objective is to outline a general workflow that helps the readers to apply the approaches.

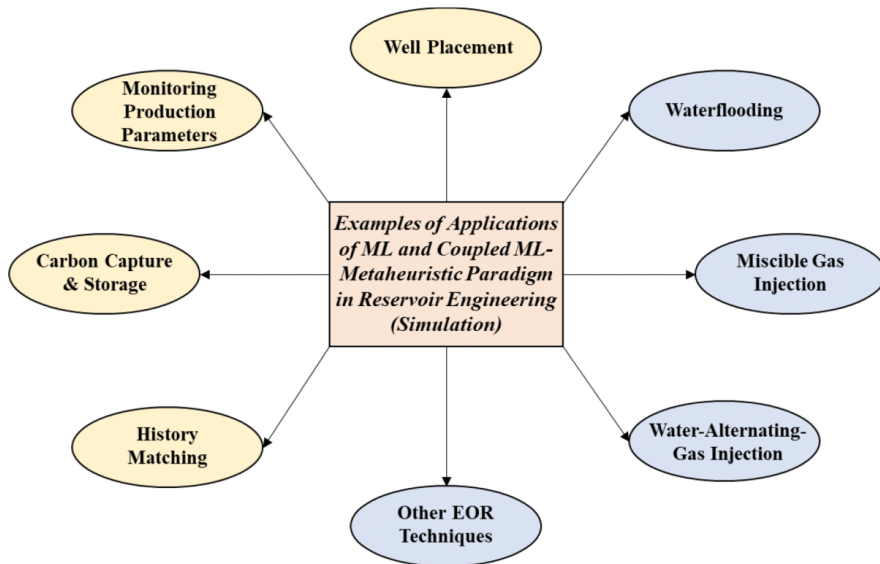


Fig. 3. Examples of Application of ML and Coupled ML-Metaheuristic Paradigm.

algorithm was established by partially or fully replacing the original fitness function (OFF) with the approximate function (AF), which was represented by ANN. Then, two surrogates were used to stochastically decide whether OFF or AF would be applied. This methodology performed well on GA for the problem of optimizing well placements. Redouane et al. (2019) successfully suggested a newly enhanced intelligent framework that involved GA, design of sampling, and proxy model to achieve optimization of well placement in a fractured unconventional reservoir. Busby et al. (2017) illustrated the use of K-medoid algorithm to select the features to run the corresponding simulation and applied the data to train the ML algorithms such as neural networks, gradient boosting, and random forest. This data analytics workflow was successfully applied to a synthetic green field and showed that the location of wells could be optimized under uncertainty. In the work of Mousavi et al. (2020), XGBoost was shown to outperform the central composite design (CCD) method in determining the best location of wells under different reservoir scenarios. In other words, XGBoost could converge to the optimal solution compared with CCD. Kristoffersen et al. (2020) discussed how the methodology of Automatic Well Planner (AWP) could be employed in a specific type of neural network, known as Neuro Evolution of Augmenting Topologies (NEAT). By coupling the neural network model with a derivative-free algorithm, namely Asynchronous Parallel Pattern Search (APPS), the well placement decision was made optimally.

The potential implementation of metaheuristic algorithms is not limited to the above-mentioned pieces of literature. Pouladi et al. (2017) suggested the use of Fast Marching Method (FMM) to develop a proxy model and coupled the proxy with PSO to optimize multiple production well placements. Hassani et al. (2011) developed three different proxy models, such as quadratic model, multiplicative model, and radial basis function of a fractured reservoir in the west of Iran, and coupled the proxies with GA to optimize the horizontal well placement. Morales et al. (2010) also performed horizontal well placement optimization in gas condensate reservoirs with a modified genetic algorithm. They extended the use of the algorithm by considering a similar optimization problem under geological uncertainties (Morales et al., 2011). The

literature on Well Placement is summarized in Table 1.

#### 4.2. Monitoring Production Parameters

In reservoir engineering, hydrocarbon and water productions play a pivotal role in determining the economic feasibility of a field development project. In this context, hydrocarbon production parameters such as oil and gas production rates must be monitored carefully to ensure substantial financial returns for the plan. Water production needs to be monitored to avoid unnecessary handling costs. Therefore, it is essential to develop a model that can monitor and predict these production parameters. However, solely applying the conventional physical and mathematical approaches to build the model is indeed challenging. The reason is that the complexity of the system has been simplified by some assumptions to justify the validity of the physical model. This is where ML methods can be applied to elude the use of these simplifications. Some literature have illustrated the successful applications of ML in monitoring and forecasting production parameters. Some applications also highlighted the development of the models by coupling the ML methods with metaheuristic algorithms.

One of the traditional approaches in production forecasting is decline curve analysis (DCA). However, Mohaghegh (2017a) explained that DCA might be insensitive to some changes in operational conditions during implementation. Therefore, ML has been preferred as an alternative for monitoring and production forecast. Sun et al. (2018) implemented the Long Short-Term Memory (LSTM) algorithm to develop a data-driven model to predict the production rate of multiple wells by only employing the production history and tubing head pressure as the input variables. Thereafter, they compared the yield of the data-driven model with three DCA models, which are Duong model, Power Law Exponential Decline (PLE), and Stretched Exponential Decline (SEPD). The comparison illustrated that the LSTM model produced the production forecast with higher accuracy. Alkhalaf et al. (2019) successfully demonstrated the application of ANN in well production forecasting by feeding the real-time data into the model. They also performed the grid search method to optimize the architecture and



**Table 1**  
Summary of Literature in the Domain of Well Placement.

Literature	Methods	Remarks	Assumptions / Limitations
Nwachukwu et al. (2018a)	XGBoost	With different well configurations, ML models with connectivities were built to predict different responses, viz. total profit, cumulative oil/gas production, or net CO <sub>2</sub> stored with less computational effort.	Augmentation of predictor variables due to the sophistication of response surface. The proposed methodology requires further verification in terms of optimization.
Nwachukwu et al. (2018b)	XGBoost / MADS Algorithm	An extended work of Nwachukwu et al. (2018a) in which ML models were made to offer reservoir responses corresponding to well locations and control during WAG under geological uncertainty. MADS was then used for joint optimization.	Augmentation of predictor variables due to the sophistication of response surface. Case-sensitive application. The proposed methodology was implemented on a synthetic case.
Xiong and Lee (2020)	ANN	ML models were built to forecast fluid production as a function of heterogeneity and the location of the injector with an improvement of prediction accuracy by using data from injectors and producers. The selection of optimized injection well placement was done with the aid of P90 and P50.	Updating of models is needed when new data is available in the case of actual field data. Verification of the suggested methodology with other strategies is required.
Chu et al. (2020)	FNN/ CNN	Multi-modal CNN outperformed FNN in terms of finding the optimal infill well placement.	The study was only focused on a single vertical infill well. Dynamic properties utilized as input data were obtained at the time of infill drilling. The exponential increase of the size of search space if horizontal drilling is considered.
Jang et al. (2018)	Sequential ANN/ PSO	Sequential ANN modeling was implemented to refine the model developed. It outperformed the coupled paradigm between the simulator and PSO in terms of the number of simulation runs.	The study was only conducted on a coalbed methane (CBM) reservoir. The performance of the sequential ANN is influenced by its parameters which are meant to be tuned.
Sayyafzadeh (2015a)	FNN/ GA	A self-adaptive surrogate-assisted evolutionary algorithm was introduced to solve	The study was only conducted on the PUNQ-3S reservoir. For this

**Table 1 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
		the well placement optimization problem with an improvement in accuracy.	optimization problem, infill wells were located at an equal distance. The methodology is yet subject to verification of real-life cases.
Redouane et al. (2019)	Gaussian Process/ GA	An adaptive surrogate reservoir modeling was displayed to manage well placement problems in a real-life fractured reservoir model.	Fixed cost of drilling and location independent costs. The methodology is yet to be tested for other field development problems. Formulation of different constraints, including well length, inter-well distance, reservoir bound, and well orientation.
Busby et al. (2017)	Neural Networks, Random Forests, Gradient Boosting/ K-medoid algorithms	Data analytics workflow was shown to determine the locations of wells for a green field.	Limited interaction between the wells to reduce the number of combinations. Limited application to real field cases.
Mousavi et al. (2020)	XGBoost	An ML model was established to predict the NPV of a well placement problem through different scenarios for optimization purposes.	Operational constraints of field development strategies were considered for the reservoir scenarios. Only three scenarios were implemented.
Kristoffersen et al. (2020)	ANN/ APPS, PSO	Automatic Well Planner (AWP) was developed to increase the efficiency of well placement optimization under geological uncertainty.	Formulation of constraints, such as length, dog-leg severity, and deviation of well. The spatial distribution of self-selected properties is assumed to be defined within the reservoir model as property maps. Wells were drilled at the beginning and at the same cost.
Pouladi et al. (2017)	FMM/ PSO	<i>*Although FMM is not considered ML, this paper showed the potential implementation of PSO in terms of coupling with any type of proxy model, which is worth reading.</i>	Fixed prices. Darcy flux is assumed negligible for the volumetric pressure drop estimation by FMM. It appeared to be impractical to illustrate the final pressure map for problems with more than one well.
Hassani et al. (2011)	Quadratic, multiplicative, and	A proxy modeling approach was	Models (to estimate

(continued on next page)

Table 1 (continued)

Literature	Methods	Remarks	Assumptions / Limitations
	Radial basis function / GA	employed to enable the optimization of horizontal well placement to be handled more quickly.	cumulative oil) are assumed to be a function of the location, direction, and length of a new horizontal well. The proposed methodology is yet to be tested for multiple geological realizations.
Morales et al. (2010)	GA	A modified GA was employed to optimize a horizontal well placement in a Gas Condensate reservoir. The Minimal Variation (MiniVar) was modified in this case.	The wellbore was set as eight grids in length. Deterministic approach. Published data of the field is limited.
Morales et al. (2011)	GA	A slight extension of Morales et al. (2010) in which geological uncertainty was considered.	Published data of the field is limited. Assumption of the probability of success and weights assignments to each realization.

hyper-parameters in modeling the ANN. Masini et al. (2019) showed the successful use of XGBoost to build a data-driven model to replace DCA. In their work, clustering techniques such as Random Forest and Density-based clustering had been used to cluster the data points with close operational conditions before training the model to conduct DCA. More intriguingly, Omrani et al. (2019) applied the hybrid approach, which was the combination of a physical model (nodal analysis) and ANN, to predict well production. They inferred that the hybrid approach performed better for long-term production forecasts (production of several years).

The use of ML approaches is extended to other domains of production engineering. Khan et al. (2019) employed ANN, SVM, and Artificial Neuro-Fuzzy Inference Systems (ANFIS) to estimate the oil rate in the artificial gas lift wells. They observed that ANN yielded much better results compared with SVM, ANFIS, and other empirical models. Furthermore, ML methods can be implemented to forecast measurements obtained from virtual flow metering and permanent downhole gauges. Bikmukhametov and Jäschke (2020) examined different approaches to hybridizing ML with first principles models of process engineering to successfully predict the volumetric flow from Virtual Flow Meter. Additionally, Tian and Horne (2017) utilized the information from permanent downhole gauges to develop a data-driven model to forecast reservoir performance via the application of recurrent neural network (RNN). Alakeely and Horne (2020) showed the potential of RNN by employing it to simulate the behavior of reservoir model. CNN was also implemented and demonstrated good results. Yang et al. (2019) illustrated a novel method in which advanced mud gas data was used to develop an ML model to estimate GOR effectively. The model comprised a combination of different techniques such as Gaussian Process, Universal Kriging, Random Forest, K-Means Regressor, and Elastic Net-regularized linear regression model. Chen et al. (2019) proved the excellent integration of ANN modeling with conventional reservoir analog studies to conduct recovery forecasts. The unsupervised ML method, autoencoders (AE), was shown useful by Alatrach et al. (2020) in predicting well production events. In this work, a 6-layered AE-NN model demonstrated positive results and could detect the deviation

from the expected behavior of a well.

There are also some literature discussing the use of ML techniques in unconventional resources. Rahmanifard et al. (2020) performed a design of experiment to develop an ANN model that accurately approximated the well production in Montney Formation, a shale gas formation. Cross et al. (2020) successfully built a decision tree-based ML model to forecast the water production of a well in Williston Basin. Another ML technique, which was the partial least square (PLS) algorithm, was employed by Al-Alwani et al. (2019) to predict the production performance in Marcellus shale based on parameters obtained from stimulation and completion. ANN modeling was employed by Cao et al. (2016) to develop data-driven models for two different scenarios, namely prediction of future production of an existing well and production forecast of a new well. They demonstrated that by incorporating the geological features, the production forecast of new wells produced excellent results. Amaechi et al. (2019) applied ANN and Generalized Linear Model (GLM) to estimate the initial gas production rate from tight gas reservoirs in Ordos basin. They implemented Garson Algorithm in ANN and Variable Importance in GLM to identify the KPI of each feature used in the development of models. The robustness of ML techniques in unconventional resources has been further validated when Urban-Rascon and Aguilera (2020) used ML to build models to achieve optimization in stimulated reservoir volume (SRV) characterization, discretization of fracture systems, and production prediction. In their work (Urban-Rascon and Aguilera, 2020), a self-organizing map (SOM) was utilized to map the hydraulic fracturing stages with microseismic data. Chaikine and Gates (2021) used a hybrid model of convolution-recurrent neural network (c-RNN) to forecast the production from multi-stage horizontal well whereas Hassan et al. (2019) employed ANN to estimate the well productivity of fishbone wells. This literature highlighted the wide applicability of ML in production engineering. These ML methods can also be coupled with metaheuristic algorithms to be more fruitful. Han and Bian (2018) developed a hybrid model of SVM and PSO to estimate the oil recovery factor in a tight reservoir. Panja et al. (2018) applied PSO to optimize the hyper-parameters of SVM and the weights and biases of ANN, which were used to predict the production from shale plays including Eagle Ford, Niobrara, and Bakken in United States. Refer to Table 2 for the summary of the literature on Monitoring Production Parameters.

#### 4.3. Waterflooding

Waterflooding is a common secondary recovery method because of its low cost of implementation. It involves injecting water into the reservoir to increase the production of oil. It is important to carefully design a waterflooding project to ensure that the oil recovery is achieved economically and optimally. Thus, designing a waterflooding project can be formulated as an optimization problem. In this aspect, one of the common practices of optimizing the waterflooding design is to adjust the well control rate or BHP over some time to achieve the targeted oil production that maximizes the objective function, e.g., NPV. Employing different types of algorithms to optimize waterflooding has been extensively researched in reservoir engineering. Optimization of waterflooding can induce high computational footprints especially when the investigated reservoir models are geologically complex. This is where the ML techniques have flourished as they could alleviate this computational challenge as discussed in several pieces of literature.

Mohaghegh (2011) showed that surrogate reservoir model (SRM) or smart proxy model (SPM), which represents a Neuro-Fuzzy system developed by using the database of an oil field, could be used to investigate which wells should undergo the rate constraint relaxation to ensure low water cut from waterflooding. Mohaghegh et al. (2012c) also applied SRM to a waterflooded onshore green field in Saudi Arabia to perform uncertainty quantification. Mohaghegh et al. (2012b) further extended the methodology to build well-based SRM and implemented it in two waterflooded offshore fields in Saudi Arabia for uncertainty

**Table 2**  
Summary of Literature in the Domain of Monitoring Production Parameters.

Literature	Methods	Remarks	Assumptions / Limitations
Sun et al. (2018)	RNN-LSTM	Comparing the production forecast of multiple wells between DCA and RNN-LSTM.	Assumption of constant tubing head pressure. Assumption of initial production for a few years in a few wells.
Alkhalaf et al. (2019)	ANN	Using ANN to predict the flow rates.	The process of retraining is limited to a predefined threshold or every ten new real-time measurements.
Masini et al. (2019)	Random Forest, XGBoost	Demonstrating automated DCA by using ML methods.	Requiring the specification of parameters for every new data set. Limitation of data set: only choke data available.
Omrani et al. (2019)	ANN	Hybridizing the first principle model and ANN to predict the short-, mid-, and long-term production.	Limited training sets. Assumption of production and operational conditions.
Khan et al. (2019)	ANFIS, ANN, SVM	Applying ML to predict the oil rate in the artificial gas lift.	Limitation of the number of epochs to 400. Limited data sets.
Bikmukhametov and Jäschke (2020)	Gradient Boosting, ANN, LSTM	Combining the ML models with the physics of process engineering to forecast the multiphase flow rates.	Simplification of the first principle models. Assumption of steady-state flow and negligible effect of the acoustic wave.
Tian and Horne (2017)	RNN	Employing RNN for the data analysis of permanent downhole gage.	Assumption of model parameterization. Models were for case-specific applications.
Alakeely and Horne (2020)	CNN, RNN	Using CNN and RNN to simulate the reservoir responses.	Limited amount of data. Models were for case-specific applications.
Yang et al. (2019)	Gaussian Process, Kriging, Random Forest, K-Mean, Elastic Net	Implementing machine learning to predict gas oil ratio based on advanced mud gas data.	Limited gas input data. Limited data collection. Limited application to formation-wise model.
Chen et al. (2019)	ANN	Forecasting the reservoir recovery by using ANN based on the analog study.	The number of ANN hidden layers was limited to 3. Assumption of the development of reservoir database through a large number of well patterns.
Alatrach et al. (2020)	Autoencoders	Predicting the event of well production by using autoencoders.	Data from limited wells. Occurrence of false positive prediction (training was conducted on some missed events of production).

**Table 2 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
Rahmanifard et al. (2020)	ANN	Forecasting the well performance in Montney Formation.	Models were for case-specific applications.
Cross et al. (2020)	Decision tree-based model	Prediction of water, gas, and oil production at a timestep of 30 days for the first two years in the Williston Basin.	Lacking information about water-related geology features for more robust modeling. Models were for case-specific applications.
Al-Alwani et al. (2019)	Partial Least Squares (PLS)	Estimating the performance of production in Marcellus Shale from stimulation and completion parameters.	Limitations in the database, including percentage parameters exceeding 100%. Limited use of P10, P50, and P90 production forecast.
Cao et al. (2016)	ANN	Production forecast using ML in unconventional reservoirs.	Data consisting of operational constraints. Production history of the well was needed as a starting point in the case of ANN.
Amaechi et al. (2019)	ANN, GLM	Estimating the initial gas production rate from tight reservoirs.	Models were for case-specific applications.
Urban-Rascon and Aguilera (2020)	SOM	Production prediction in low permeability reservoirs.	Assumed that earthquake showing self-similar behavior in fracture scaling. Models were for case-specific applications.
Chaikine and Gates (2021)	c-RNN	Using c-RNN to forecast the production from multi-stage hydraulically fractured horizontal wells.	Limiting the number of variables used. Limited sample sizes.
Hassan et al. (2019)	ANN, Fuzzy Logic, RBF-NN	Well productivity forecast from fishbone wells using ML methods.	Assumed input parameters. Limits were imposed on the maximum and minimum values of parameters.
Han and Bian (2018)	SVM, ANN/ PSO	Estimating the oil recovery factor of a low permeability reservoir by using the SVM-PSO model.	Models were for case-specific applications.
Panja et al. (2018)	ANN, LSSVM	Determining the production from shales using ML methods.	Homogeneity in reservoir properties. A limited number of iterations due to time constraints.

analysis. Alenezi and Mohaghegh (2017) also successfully developed an SPM for the numerical simulation model of the waterflooded SACROC unit that accurately predicted the pressure and oil saturation values at the grid block level. To estimate the production under waterflooding, Negash and Yaw (2020) used Bayesian regularization algorithm as the

training algorithm to develop an artificial neural network (ANN)-based proxy of a reservoir in Malay basin. Moreover, [Zhong et al. \(2020\)](#) used a more advanced ML method, conditional deep convolutional generative neural network (cDC-GAN), to build a proxy of a 2D oil-water system reservoir to forecast the field production rates under waterflooding. They also used this proxy to conduct optimization and uncertainty quantification.

[Artun \(2017\)](#) did a comparative study between ANN model and Capacitance Resistance Model (CRM) for the determination of interwell connectivity in waterflooded reservoirs. He stated that ANN has better flexibility in terms of modeling and data requirements since CRM is a reduced-physics model. [Kalam et al. \(2020\)](#) employed three approaches

including non-linear regression (NLR), ANN, and adaptive neuro-fuzzy to forecast the performance of waterflooding of a stratified reservoir. They concluded that ANN yielded the best prediction. [Deng and Pan \(2020\)](#) also demonstrated the development of a proxy that consisted of Echo State Network (ESN) coupled with an empirical relationship of water fractional flow. This model was then used for production optimization in a closed-loop manner. SVR was also effectively employed to predict the production of a reservoir under different geostatistical realizations ([da Silva et al., 2020](#)). In another work, [Bai and Tahmasebi \(2020\)](#) built four different models using ANN, RNN, deep gated recurrent unit (GRU), and LSTM to predict the water coning, which has been an important issue to be handled in waterflooding. [Jia and Deng \(2018\)](#)

**Table 3**  
Summary of Literature in the Domain of Waterflooding.

Literature	Methods	Remarks	Assumptions / Limitations
<a href="#">Mohaghegh (2011)</a>	ANN/ GA/ Fuzzy Logic	Introducing AI-based modeling by using a case study of waterflooding.	Models developed were case-specific.
<a href="#">Mohaghegh et al. (2012c)</a>		Using AI technique to develop a Surrogate Reservoir Model (SPM) for an Onshore Green Field under waterflooding in Saudi Arabia.	
<a href="#">Mohaghegh et al. (2012b)</a>		Extending the methodology to well-based SRM to two offshore fields in Saudi Arabia.	
<a href="#">Alenezi and Mohaghegh (2017)</a>		Building a smart proxy model for the waterflooded SACROC unit.	
<a href="#">Negash and Yaw (2020)</a>	ANN	Production prediction of the waterflooding process by using ANN.	Existence of noise in the data collected. Models built were case-specific.
<a href="#">Zhong et al. (2020)</a>	Conditional deep convolutional generative neural network (cDC-GAN), adversarial neural network	Forecasting the field production rates of three waterflooding cases by using the neural network models.	Limitations caused by material balance and difficulty of splitting production among producers increased the uncertainty of final results.
<a href="#">Artun (2017)</a>	ANN	Implementing ANN and reduced physics model to characterize the inter-well connectivity in a waterflooded reservoir.	The synthetic reservoir was set at a maximum BHP of 5000 psia.
<a href="#">Kalam et al. (2020)</a>	ANN/ Adaptive neuro-fuzzy	Estimating the oil recovery of waterflood by using AI methods in four cases: two real field cases, analytical and semi-analytical models.	Communication between layers was assumed to be valid for the first category but not for the second. Immiscible and piston-like displacement without gravity effects. In this methodology, the produced water contained water coning from the injector.
<a href="#">Deng and Pan (2020)</a>	Echo State Network	Embedding ML technique in Closed-Loop Reservoir Management (CLRm) for a waterflooded mature field.	All producers were under BHP control whereas all injectors were under rate control. The reservoir model was assumed to undergo 5 years of production before the start of the workflow. Assumption of data acquisition frequency.
<a href="#">da Silva et al. (2020)</a>	SVR	Predicting production from reservoir considering geostatistical realizations.	The use of the dimensionality reduction method might be needed in the proposed work for much more complex cases.
<a href="#">Bai and Tahmasebi (2020)</a>	LSTM	Forecasting the water breakthrough by using LSTM.	A large variance of the training dataset.
<a href="#">Jia and Deng (2018)</a>	Clustering technique	Employing streamline clustering technique to identify waterflooding flowing area in oil reservoirs.	The flow of reservoir fluids was assumed to be along the streamline at a particular timestep.
<a href="#">Guo and Reynolds (2018)</a>	SVR	Performing waterflooding optimization by using SVR-based proxy models.	Limited total number of simulation runs for training. The constraint of well control by simple bounds.
<a href="#">Hourfar et al. (2019)</a>	RL	Applying RL to conduct waterflooding optimization.	Voidage replacement assumption. Operational constraints, like minimum and maximum injection rate, and an upper limit of the cumulative injection at each time step. Limitation of RL: delayed reward assignment, a trade-off between exploration-exploitation, and curse of dimensionality.
<a href="#">Ma et al. (2019)</a>	RL		Assumption of production period of 1080 days. The maximum production rate was 1500 STB/day and the minimum BHP of the producer was 1000 psi.
<a href="#">Chen et al. (2020)</a>	RBF Network / DE	Conducting Global and Local surrogate modeling to optimize waterflooding with DE.	A limited number of training data points.
<a href="#">Jia et al. (2020)</a>	Machine learning algorithm/ PSO	Illustrating the combined use of ML and PSO to perform data-driven optimization of water injection plans.	A limited number of injection plans were used.

used the streamline clustering AI method to identify the flowing area of waterflood in an oil field. In this work, having a reasonable number of clusters was important to have accurate clustering results. To achieve this, density peak clustering was used.

Production optimization under waterflooding of a reservoir has been frequently done with different algorithms. Guo and Reynolds (2018) developed a proxy model of a channelized reservoir by considering different geological scenarios and performed the optimization by using the stochastic simplex approximate gradient (StoSAG). Hourfar et al. (2019) employed reinforcement learning (RL) method to optimize production through waterflooding. About this, Ma et al. (2019) used deeper RL algorithms to conduct a similar optimization under geological uncertainties. They considered deep Q-network (DQN), double DQN, dueling DDQN, and deep deterministic policy gradient (DDPG). They inferred that in terms of maximization of NPV, DQN was able to perform better than the rest and as well as PSO. Furthermore, other works highlighted the useful application of metaheuristic algorithms in optimizing waterflooding. Chen et al. (2020) introduced a new methodology that was global and local surrogate-model-assisted differential evolution (GLSADE) to optimize waterflooding production. GLSADE was shown to be able to attain higher NPV than the conventional evolutionary algorithm based on three different models, such as two 100-dimensional benchmark functions, a three-channel model, and Egg model. Jia et al. (2020) suggested a data-driven optimization that included ML clustering technique and PSO for waterflooding in a complex reservoir in eastern China. ML clustering algorithm was used to identify the efficiency of waterflood performance at different layers. Then, PSO was used to conduct the optimization of the water injection plan. Peruse Table 3 for the summary of the literature on Waterflooding.

#### 4.4. Water-Alternating-Gas (WAG)

WAG injection is one of the most prevalent EOR techniques. It involves injecting water and gas alternately (in a cyclic manner) over a period to increase sweep efficiency to contribute to higher oil recovery. The injected gas can be CO<sub>2</sub> or a mixture of CO<sub>2</sub> and hydrocarbon gas. Optimization of WAG parameters has been widely researched because it is essential to ensure a high economic return. As stated by Mohagheghian et al. (2018), the WAG parameters generally include water and gas injection rates, BHP of producers, cycle time, cycle ratio, composition of the injected gas, total time of WAG, etc. In this context, they illustrated the successful use of metaheuristics algorithms like GA and PSO to tune the WAG parameters in Norne field to maximize the NPV and incremental recovery factor (IRF). In addition, other literature recommended the implementation of ML methods to provide fast analysis of WAG injection.

Regarding the employment of ML and metaheuristic algorithms in optimizing the WAG process, Nait Amar et al. (2018a) illustrated the development of dynamic proxy using time-dependent multi-ANN to predict the total field oil production. Then, this dynamic proxy was coupled with GA and ACO to determine the optimal WAG parameters. In addition to this, Nait Amar et al. (2020c) successfully applied SVR to build the dynamic proxy of a field in Algeria and coupled it with GA to optimize the water-alternating CO<sub>2</sub> gas parameters. More interestingly, the hyperparameters of SVR were optimally adjusted by GA before being used (Nait Amar et al., 2020c). Nait Amar et al. (2021) implemented two different proxies of Gullfaks field, namely Multilayer Perceptron (MLP) and Radial Basis Function Neural Network (RBFNN). Thereafter, GA and ACO were used along with these proxies to optimize the WAG process. Nwachukwu et al. (2018b) employed XGBoost to establish a proxy of a reservoir model under different geological realizations. This proxy was coupled with MADS to not only optimize the well locations but also find the optimal WAG parameters.

Belazreg et al. (2020) applied a random forest algorithm to build a model based on a database from 28 WAG pilot projects worldwide to forecast the IRF during the WAG process. Belazreg et al. (2019) also

efficiently attempted the use of GMDH to develop the IRF predictive model, which was a function of horizontal and vertical permeabilities, fluid properties, mobility of fluids, WAG injection scenario, residual oil saturation to gas, trapped gas saturation, injected gas volume, and reservoir pressure. Moreover, in the work of Belazreg and Mahmood (2020), GMDH was employed to predict WAG IRF based on the data from 33 WAG projects from 28 fields in the world. Furthermore, the methodology of top-down modeling (TDM) was used by Yousef et al. (2020) to build a model to estimate the reservoir performance of a mature oil field in Middle East under WAG injection. This model also provided a rapid medium for the optimization of WAG parameters. Jaber et al. (2019b) implemented Central Composite Design (CCD) to establish a proxy of a reservoir in Subba oilfield to approximate the incremental oil recovery during the miscible CO<sub>2</sub>-WAG process.

Nait Amar and Zeraibi (2019) established three different MLPs trained by LMA, BR, and SCG. After that, these MLPs were coupled with Non-Dominated Sorting Genetic Algorithm version II (NSGA-II) to conduct multi-objective optimization of the CO<sub>2</sub>-WAG process. Enab and Ertekin (2020) also demonstrated how ANN could be built and used for the screening and optimization of the CO<sub>2</sub>-WAG process and the structures of fish-bone well in low permeability reservoirs. The case study presented was a reservoir from Sirri A field. Read Table 4 for the summary of the literature on WAG.

#### 4.5. Miscible Gas Injection

Miscible gas flooding has been one of the well-known EOR methods applied in the petroleum industry. Examples of gasses usually applied in miscible gas flooding include carbon dioxide (CO<sub>2</sub>), nitrogen (N<sub>2</sub>), natural gas, etc. CO<sub>2</sub> has been preferred over other gasses because implementing miscible CO<sub>2</sub> gas injection not only increases oil recovery but also reduces greenhouse gas emissions. Therefore, the literature survey in this section will focus mainly on miscible CO<sub>2</sub> gas flooding. In miscible gas injection, minimum miscibility pressure (MMP) is one of the most significant parameters that can affect the efficiency of the injection process. Accurate modeling of MMP thus has been extensively researched and application of ML in this context has also been proven successful.

Tatar et al. (2013) employed the RBFNN to estimate the MMP of pure and impure CO<sub>2</sub>-reservoir oil. 147 data sets from different pieces of literature were used to generate the database for the modeling. Apart from RBFNN, other approaches like GA-based Backpropagation Algorithm Neural Network (GA-BPNN) were also efficiently applied by Chen et al. (2014) to develop the predictive model of MMP in the CO<sub>2</sub>-EOR process. GA-BPNN outperformed other existing correlations as discussed in Chen et al. (2014). In addition to BPNN, Bian et al. (2016) illustrated that GA could be coupled with SVR to develop a model that could determine CO<sub>2</sub>-oil MMP in both pure and impure streams of CO<sub>2</sub>. GA-SVR was demonstrated to yield more accurate results of MMP than other correlations. Karkevandi-Talkhooncheh et al. (2018) used the hybrid models of RBFNN and five different metaheuristic algorithms: GA, PSO, DE, Imperialist Competitive Algorithm (ICA), and ACO. These models were able to forecast the MMP under pure and impure CO<sub>2</sub> injection conditions. In their study (Karkevandi-Talkhooncheh et al., 2018), ICA-RBFNN outperformed other hybrid models.

Furthermore, Nait Amar et al. (2018c) established a hybrid model of ANN and DE to forecast MMP for a pure CO<sub>2</sub>-oil system. The initial best weight and bias parameters of ANN were optimized by employing DE. Then, this DE-optimized ANN undergoes backpropagation training again to be used as a predictive model. Nait Amar and Zeraibi (2018) also successfully tuned the hyperparameters of SVR by using ABC and applied it as a model to predict MMP in CO<sub>2</sub> flooding. SVR-ABC yielded a more accurate result than the SVR optimized via trial and error and other correlations. Dargahi-Zarandi et al. (2020) utilized more ML methods to develop three intelligent models to do the same prediction. These methods include Group Method of Data Handling (GMDH), MLP,

**Table 4**  
Summary of Literature in the Domain of WAG.

Literature	Methods	Remarks	Assumptions / Limitations
Mohagheghian et al. (2018)	GA, PSO	Optimizing WAG in Norne field with evolutionary algorithms.	Economic constraints comprise a lower limit on oil production (10 Sm <sup>3</sup> /day) and upper limits on water cut (0.95) and GOR (500 vol/vol). Variables, apart from cycle ratio, cycle time, and total WAG, were assumed to be continuous.
Nait Amar et al. (2018a)	ANN/ GA, ACO,	Optimizing WAG in a synthetic field with ANN and nature-inspired algorithms.	Imposing different constraints to the design parameters. The database was generated based on multiple runs of the simulation.
Nait Amar et al. (2020c)	SVR/ GA	Optimizing CO <sub>2</sub> -WAG in a synthetic field with ANN and nature-inspired algorithms.	
Nait Amar et al. (2021)	ANN/ GA, ACO	Optimizing WAG in Gullfaks field with ANN and nature-inspired algorithms.	
Nwachukwu et al. (2018b)	XGBoost/ MADS Algorithm	An extended work of Nwachukwu et al. (2018a) in which ML models were built to offer reservoir responses corresponding to well locations and control during WAG under geological uncertainty. MADS was then used for joint optimization.	Augmentation of predictor variables due to the sophistication of response surface. Case-sensitive application. The proposed methodology was implemented on a synthetic case.
Belazreg et al. (2020)	Random Forest	Predictive Modeling of Incremental Recovery Factor of CO <sub>2</sub> -WAG.	Modeling was done based on limited/missing data.
Belazreg et al. (2019)	GDMH, ANN	Predictive Modeling of Recovery Factor of WAG.	WAG was assumed to begin after 10 years of waterflooding.
Belazreg and Mahmood (2020)		Predictive Modeling of WAG Incremental Recovery Factor of WAG through pilot projects.	Modeling was done based on limited data. The recovery factor of the pilot tests ranged from 5 to 10%.
Yousef et al. (2020)	ANN	Implementing ANN for top-down modeling in the prediction of reservoir performance under WAG.	A limited number of pressure tests are available. Reservoir characteristics were slightly modified and assumed to be reasonably accurate for TDM. History data (initial injection rate) was assumed to be the benchmark to assess the efficiency of injection.
Jaber et al. (2019b)	CCD	Employing a data-driven proxy to evaluate the incremental oil recovery of the CO <sub>2</sub> -WAG process.	7 independent variables were assumed in the study. The database was generated based on multiple runs of the simulation.

**Table 4 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
Nait Amar and Zeraibi (2019)	ANN/ NSGA-II	Multiobjective optimization of WAG-CO <sub>2</sub> in a synthetic field.	The daily oil production rate was limited to 8500 Sm <sup>3</sup> /day. Total Field Oil Recovery and Total Field Water Production were assumed as objective functions. The database was generated based on multiple runs of the simulation.
Enab and Ertekin (2020)	ANN	Applying ANN to screen and optimize CO <sub>2</sub> -WAG and the structures of fish-bone wells in reservoirs with low permeability.	Limitations were imposed by defining the range of each variable. Limitations on drilling and completions were not considered.

and Adaptive Boosting SVR (AdaBoost SVR). Sinha et al. (2020b) built four models, linear SVM, K-Nearest Neighbor regression (KNN), Random Forest Regression (RF), and ANN, to determine the MMP. They deduced that RF worked best compared with the other models. Thereafter, they substantially enhanced the RF model to become an ensemble model (hybridization of available correlation and RF) which they termed the super-learner method.

Dong et al. (2019) integrated the use of L2 regularization (which acts as a penalty term to prevent overfitting during the training phase) and dropout as a step in improving the ANN-based model that was employed to forecast MMP. This improvement could prevent the overfitting issue and further strengthen the predictive capability of the model. Other than estimating MMP, the Fuzzy Logic method was shown by Karacan (2020) capable of determining the recovery factor of miscible CO<sub>2</sub> gas flooding. This fuzzy-based model (with the Mamdani-type inference system) was developed by using the data from 24 major USA field projects. You et al. (2019b) also implemented a hybrid method that considered the coupling of ANN with PSO to perform multi-objective optimization of CO<sub>2</sub>-EOR. The objective functions included CO<sub>2</sub> storage, oil recovery factor, and NPV. The literature on Miscible Gas Injection is summarized in Table 5.

#### 4.6. Other EOR techniques

EOR methods can be fathomed as tertiary recovery techniques used to retrieve the remaining oil from hydrocarbon reservoirs. These techniques will be initiated after the exhaustion of both primary and secondary recovery methods. Examples include surfactant flooding, polymer flooding, any other chemical flooding, nitrogen gas injection, in-situ combustion, Steam-Assisted Gravity Drainage (SAGD), cyclic steam injection, fire-flooding, microbial flooding, and so forth. The cost of implementation of these methods is relatively higher than primary and secondary recovery methods. Therefore, careful design and optimization of the tertiary recovery methods are important to elude any unnecessary waste of expenditure and ensure the profitability of the project. Several works have illustrated the application of ML methods in the context of the employment of different tertiary recovery techniques.

Rezaian et al. (2010) applied experimental methods to examine the effect of Poly Vinyl Acetate (PVA) on the rheology of crude oil and water. This was because they wanted to study the effectiveness of PVA to be used in polymer flooding. Thereafter, they demonstrated the successful implementation of ANN in developing a predictive model based on the experimental data. Zerafat et al. (2011) illustrated the use of

**Table 5**  
Summary of Literature in the Domain of Miscible Gas Injection.

Literature	Methods	Remarks	Assumptions / Limitations
Tatar et al. (2013) Chen et al. (2014)	RBFNN Backpropagation Neural Network / GA	Modeling of the CO <sub>2</sub> -reservoir oil minimum miscibility pressure.	Models were developed based on available experimental data.
Bian et al. (2016)	SVR / GA	Modeling of CO <sub>2</sub> -oil minimum miscibility pressure with pure and impure CO <sub>2</sub> .	Built based on available experimental data.
Karkevandi-Talkhooncheh et al. (2018)	Radial Basis Function Networks / GA, PSO, ICA, ACO, DE		Separate models for pure and impure CO <sub>2</sub> . Models were extended on a basis of limited data points.
Nait Amar et al. (2018c)	ANN / DE	Developing the predictive model of minimum miscibility pressure in a pure CO <sub>2</sub> -oil system.	Models were built based on data from a few experiments. Choice of input parameters was assumed.
Nait Amar and Zeraibi (2018)	SVR / ABC	Building the predictive model of minimum miscibility pressure in the CO <sub>2</sub> -EOR process.	
Dargahi-Zarandi et al. (2020)	Adaptive Boosting SVR, GDMH, MLP	Predictive Modeling of MMP of pure and impure CO <sub>2</sub> -crude oil systems.	Predicting the limited range of MMP between 1000 psia and 4900 psia. Dataset limitation.
Sinha et al. (2020b)	Linear SVM/ K-Nearest Neighbor Regression/ Random Forest regression/ ANN	Predictive Modeling of MMP of CO <sub>2</sub> -crude oil systems.	Data set limitation. Further applicability of models was limited.
Dong et al. (2019)	ANN		A limited number of field cases. Input variables were assumed based on the availability of data.
Karacan (2020)	Fuzzy Logic	Forecasting of recovery factor of miscible CO <sub>2</sub> -EOR.	The model was constructed by only using data from 24 U.S. field projects.
You et al. (2019b)	ANN/ MO-PSO	Applying ANN for multi-objective optimization of CO <sub>2</sub> -EOR.	Only 4 input parameters: water cycle, gas cycle, BHP of producer, and

**Table 5 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
			water injection rate, were considered.

Bayesian network analysis to screen for an efficient EOR method. They applied different data sets from seven different EOR methods, like miscible N<sub>2</sub> injection, miscible hydrocarbon injection, miscible and immiscible CO<sub>2</sub> injection, polymer flooding, in-situ combustion, and steam injection. Siena et al. (2016) further built a novel EOR screening tool by using the Bayesian approach. The approach they implemented included Bayesian Hierarchical Clustering (BHC) algorithm and PCA, which could be understood as a two-step algorithm. PCA was used to reduce the dimensionality of data and provide accurate distance metrics regarding the similarity among the projects. The database they used generally comprised thermal EOR, chemical EOR, and gas/WAG injection that were derived from different worldwide projects and literature.

Parada and Ertekin (2012) applied ANN modeling to establish a new screening tool for four different recovery methods including water-flooding, miscible N<sub>2</sub> injection, miscible CO<sub>2</sub> injection, and steam injection. Khazali et al. (2019) presented the use of a fuzzy decision tree in the assessment of EOR screening. They stated that the fuzzy decision tree could perform the simultaneous ranking and classification of different EOR techniques. Hence, an expert system could be designed to generate the EOR rules. In their work, the decision tree was applied to the dataset of 548 observations related to ten different EOR methods. Sun and Ertekin (2020) showed that ANN-based proxies could be established to do the screening of polymer flooding. Then, they coupled the proxies with PSO to optimize the polymer flooding process to maximize the NPV. In the domain of optimization, Ma and Leung (2020) designed a hybrid workflow that integrated multi-objective optimization (MOO) and proxy modeling in the case of injection of warm solvent into heterogeneous heavy oil reservoirs. In their work (Ma and Leung, 2020), NSGA-II was used to perform the MOO.

Regarding recovery performance forecasting, Ehsan et al. (2014) applied PCA to decrease the dimensionality of the input data before modeling the ANN. The ANN was used to estimate the production induced by the SAGD process in heterogeneous reservoirs. Ersahin and Ertekin (2020) also conducted the development of ANN of cyclic steam injection (CSI) in naturally fractured reservoirs. The ANN models developed included a forward model and two inverse models. The forward model was used to estimate the cumulative oil production and changes in viscosity near the wellbore. About the inverse-looking models, the first one was used to find out the ideal design of injection variables whereas the second one was used for the characterization of some reservoir properties. Abdullah et al. (2019) developed five ANN models to be implemented in chemical EOR in a sandstone reservoir. These models were applied to estimate reservoir performance, forecast reservoir properties, determine the design parameters for known performance and properties, and find out the design parameters for a targeted cumulative oil production and project period. Refer to Table 6 for the summary of the literature on other EOR techniques.

#### 4.7. Carbon Capture and Storage (CCS)

The increasing amount of carbon dioxide (CO<sub>2</sub>) gas in the atmosphere is one of the main factors contributing to climate change today. Nevertheless, CO<sub>2</sub> emission is an inevitable consequence of different types of industrial and commercial activities required to fulfill our daily practical needs. Therefore, awareness has arisen among researchers to look for an efficient strategy to reduce CO<sub>2</sub> emissions. One of the proposed strategies to assure that emission of CO<sub>2</sub> will remain at a low level

**Table 6**  
Summary of Literature in the Domain of Other EOR Techniques.

Literature	Methods	Remarks	Assumptions / Limitations
Rezaian et al. (2010)	ANN	ML models were built to predict the effect of Poly Vinyl acetate on the rheology of water and crude oil in EOR.	Data was only from one experiment and this might limit the applicability of the models developed. The experiment was done under predefined conditions.
Zerafat et al. (2011)	Bayesian Network	The model was created as a tool for EOR screening based on data from 10 Iranian southwest reservoirs.	The study was done without considering economic limitations. Models were case-specific.
Siena et al. (2016)	Bayesian Clustering/ PCA	A novel EOR screening tool was established.	Evaluation of probability based upon the fundamental assumption of Bayesian clustering. Identification of analogs is vital to the successful implementation of this methodology.
Parada and Ertekin (2012)	ANN	An ANN-based EOR screening tool was built.	Ability to predict reservoir response to different conditions within certain limits. Four different compositions of hydrocarbon were considered.
Khazali et al. (2019)	Fuzzy Decision Tree	EOR screening evaluation by using a fuzzy decision tree.	The proposed method works best with sufficient data. Economic issues were not concerned.
Sun and Ertekin (2020)	ANN	The ANN-based model was created to screen and optimize polymer flooding.	Salinities of injected and in-situ water were assumed the same. Gravitational forces and capillary pressure were assumed to be negligible. Existence of upper and lower limits of the search space of design parameters.
Ma and Leung (2020)	ANN/ NSGA-II	Hybridization of ANN and NSGA-II for multi-objective optimization of warm solvent injection in heterogeneous heavy oil reservoirs.	Assumption of uniform properties within each facies. Only sand was assumed to exist at the well grid cell. Only bottom-hole pressures were chosen as design parameters. Excessive startup time and slow extraction rate limited the application.
Ehsan et al. (2014)	ANN/ PCA	An integrated approach of ANN and PCA for the prediction of SAGD performance in heterogeneous reservoirs.	The study was limited to the database that was created from the combinations of the attributes of heterogeneous reservoir as input. Separate ANNs were required for better results.
Ersahin and Ertekin (2020)	ANN	Using ANN to model the Cyclic Steam Injection Process in Naturally Fractured Reservoirs.	Oil behaves as Newtonian fluid. A trial-and-error approach was needed to

**Table 6 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
Abdullah et al. (2019)	ANN	Applying ANN to design and model the implementation of chemical EOR.	train the ANN and determine its optimum design. The surfactant was in the aqueous phase. Data available was assumed to be reservoir characteristics, project duration aimed, and cumulative oil volume.

is Carbon Capture and Storage (also known as Carbon Capture and Sequestration) (CCS). Fundamentally, CCS is performed by injecting the captured CO<sub>2</sub> into geological formations and ensuring it is safely trapped underground. Much research has been done on the domain of CCS and one of the most cutting-edge topics is the coupling of ML techniques with CCS. Several pieces of literature also discussed the application of metaheuristic algorithms along with the ML methods in CCS.

Sipöcz et al. (2011) developed two different ANN models to predict the CO<sub>2</sub> capturing processes. The difference between the models was the training algorithm used where one was trained using scaled conjugate gradient (SCG) algorithm whereas the other training algorithm employed was Levenberg Marquardt algorithm (LMA). They deduced that these models could provide results not only much faster than process simulator CO<sub>2</sub>SIM but also within an acceptable level of accuracy. Miscibility of CO<sub>2</sub> in formation fluids is another important aspect of CCS. Mesbah et al. (2018) illustrated the implementation of a multilayer perceptron neural network (MLP-NN) by employing 1386 experimental data points to forecast the miscibility of CO<sub>2</sub> and supercritical CO<sub>2</sub> in ionic liquid. During the development of the model, they performed outlier diagnostics to ensure the quality of data used. Furthermore, Sinha et al. (2020a) used ML methods, like random forest and multilayer feedforward neural network (MFNN), to build models for leakage detection in a carbon sequestration project in Cranfield reservoir, Mississippi, USA. The models were made based on time series signals from the pressure pulse test. Vo Thanh et al. (2020) also successfully showed the use of ANN to estimate the performance of CO<sub>2</sub>-EOR and storage in a residual oil zone located in Permian basin.

Metaheuristic algorithms were also proven to be useful to be coupled with ML techniques in CCS. You et al. (2019a) provided a framework to conduct co-optimization on CO<sub>2</sub> storage, the performance of CO<sub>2</sub>-EOR, and the NPV of the project. In the framework, RBFNN and multilayer neural network modeling were implemented to build the proxies of the reservoir model. Then, PSO was used to do the co-optimization. After that, You et al. (2020c) also developed ANN to establish a proxy of the sandstone reservoir in Pennsylvanian Upper Morrow to estimate the time series of cumulative oil production and CO<sub>2</sub> storage. PSO was again applied to co-optimize CO<sub>2</sub> storage, the performance of CO<sub>2</sub>-EOR, and the NPV of the project. In addition to proxy modeling, other interesting literature have discussed the use of ML to predict important parameters relevant to CCS. The solubility of CO<sub>2</sub> in formation fluid is an essential parameter to be considered in CCS. In this context, Nait Amar et al. (2019) applied MLP and RBFNN to make predictive models of CO<sub>2</sub> solubility in brine. More intriguingly, LMA was employed to train MLP whereas GA, ABC, and PSO were used to train RBFNN. In their study, RBFNN-ABC outperformed the other models. Hemmati-Sarapardeh et al. (2020a) also used four ML techniques, including RBFNN, MLP, Least-Squares Support Vector Machine (LSSVM), and Gene Expression Programming (GEP), to model the solubility of CO<sub>2</sub> in water at high temperature and pressure. During the training phase, four back-propagation algorithms were used in the modeling of MLP whereas four nature-inspired algorithms were used in the modeling of RBFNN and LSSVM. These nature-inspired algorithms included PSO, GA, FA, and



DE.

In addition, [Nait Amar and Jahanbani Ghahfarokhi \(2020\)](#) presented how white-box ML methods could be used to estimate CO<sub>2</sub> diffusivity in brine. These white-box ML techniques were GMDH and GEP. These models could be applied to predict the diffusivity coefficient of CO<sub>2</sub> in brine as functions of temperature, pressure, and viscosity of the solvent. Also, [Nait Amar et al. \(2020a\)](#) utilized MLP, GMDH, and GEP to build predictive models of CO<sub>2</sub> viscosity at high temperature and pressure. Four backpropagation algorithms, LMA, SCG, Bayesian Regularization (BR), and Resilient Backpropagation (BR), were used to train the MLP. The thermal conductivity of carbon dioxide is another important parameter in CCS projects. Regarding this, [Nait Amar et al. \(2020b\)](#) first established some MLP-based models and RBFNN trained by PSO to forecast the thermal conductivity of carbon dioxide. After that, the two best models were coupled with two Committee Machine Intelligent Systems (CMIS) via the weight averaging method and GMDH. Peruse [Table 7](#) for the summary of the literature on CCS.

#### 4.8. History Matching (HM)

History Matching (HM) can be understood as a task that involves tuning or adjustment of any parameter that is used in reservoir modeling to enable a reservoir model to yield results that match the observed real-field data. It can be understood that HM can be very laborious and time-consuming. To mitigate this computational challenge, several works propose the application of ML techniques in establishing the proxies of the numerical reservoir models to be employed in HM. Besides that, HM is considered an optimization problem as it involves the minimization of the error between the predicted data and observed data. In this aspect, metaheuristic algorithms have widely contributed to the successful and efficient deployment of HM. More intriguingly, some literature highlighted the coupling of proxies with metaheuristic algorithms in performing HM. Thus, ML and metaheuristic algorithms show great potential to be further improved in the future implementations of HM.

[Sampaio et al. \(2009\)](#) presented the fundamental use of FNN as the nonlinear proxy model of a numerical and synthetic heterogeneous model. Then, they applied it in HM and showed very positive results. However, they opined that the complexity of the reservoir model could be increased to illustrate the robustness of ML. [Shahkarami et al. \(2014b\)](#) proposed the use of a surrogate reservoir model (SRM), which was represented as a Neuro-Fuzzy system, in the HM phase. They termed it AI-assisted HM (AHM) and successfully showed that it could reduce the computational time induced by the conventional approach of HM using a very heterogeneous model. [Masoudi et al. \(2020\)](#) employed a similar methodology to conduct HM on a very complicated and mature offshore oilfield in Malaysia. However, the SRM used was the deconvolutional neural network. Also, they applied top down modeling (TDM) that included the data from the real field in designing the SRM used for HM. [Illarionov et al. \(2020\)](#) studied different approaches to HM of a real-field model on an FNN-based proxy termed as Neural Differential Equations based Reduced Order Model (NDE-b-ROM). The HM methods considered a variation of reservoir model parameters, an adaptation of neural network architecture, and an adaptation of latent space of model parameters. They inferred that latent space adaptation would yield the best result.

More advanced techniques were also used in proxy modeling along with HM. [Chaki et al. \(2020\)](#) employed deep neural networks (DNN) and RNN to build proxy models of Brugge reservoir and conducted an exhaustive search of HM using the models. [Honorio et al. \(2015\)](#) also included a novel ML method to study the prior information on geology and use pluri-principal-component-analysis (pluri-PCA) to rebuild a model. Fundamentally, they implemented pluri-PCA to transform the geological models to Gaussian PCA coefficients and tuned them in HM. [Rammay et al. \(2020\)](#) examined different algorithms used for HM of imperfect subsurface models. These algorithms included HM without considering model error, HM with an update of total error covariance

**Table 7**  
Summary of Literature in the Domain of CCS.

Literature	Methods	Remarks	Assumptions / Limitations
<a href="#">Sipőcz et al. (2011)</a>	ANN	ML was employed for the modeling and prediction of the CO <sub>2</sub> capture process plant.	Limited to 5000 epochs due to low computational space. Each input parameter was assumed and underwent sensitivity analysis to assess its dependence on the output.
<a href="#">Mesbah et al. (2018)</a>	MLP	ML was used to develop predictive models of miscibility of CO <sub>2</sub> and supercritical CO <sub>2</sub> in ionic liquid.	Input parameters used for modeling were assumed. The methodology is yet subject to the verification of other databases.
<a href="#">Sinha et al. (2020a)</a>	Multilayer FNN, Random Forest, Linear models	ML models were established for leakage detection in a carbon Sequestration project.	Simplistic ML techniques showed limited sufficiency in capturing the details. The window of 1000 samples was not decided through a comprehensive analysis.
<a href="#">Vo Thanh et al. (2020)</a>	ANN/ PSO	ANN was applied to forecast the performance of CO <sub>2</sub> EOR and storage in a residual oil zone.	The model developed is case-specific. The selection of the range of uncertainty parameters requires more attention.
<a href="#">You et al. (2019a)</a>	RBFNN, Multilayer Neural Networks / PSO	An optimization framework, considering ML and PSO, was proposed to co-optimize CO <sub>2</sub> EOR and storage in a sandstone reservoir.	Production pressure is limited to 4000 psia whereas injection pressure is 5000 psia. Three different development strategies were assumed.
<a href="#">You et al. (2020c)</a>	ANN / PSO	A part of the extended work of <a href="#">You et al. (2019a)</a> . An ML-assisted computational workflow was introduced to optimize a CO <sub>2</sub> -WAG injection plan that considers CO <sub>2</sub> sequestration and hydrocarbon recovery.	Production pressure is limited to 4000 psia whereas injection pressure is 5000 psia. Operational cost is primarily influenced by the amount of CO <sub>2</sub> .
<a href="#">Nait Amar et al. (2019)</a>	MLP, RBFNN / GA, PSO, ABC	Different ML models were built to determine the solubility of CO <sub>2</sub> in brine, which is important to the application of CCS.	Limited to the database used for modeling (robustness still needs to be verified). Input data parameters were

(continued on next page)

Table 7 (continued)

Literature	Methods	Remarks	Assumptions / Limitations
Hemmati-Sarapardeh et al. (2020a)	LSSVM, GEP / PSO, GA, DE, FA	Numerous ML methods were implemented to estimate the solubility of CO <sub>2</sub> in water at high pressure and temperature.	assumed for developing the models
Nait Amar and Jahanbani Ghahfarokhi (2020)	GMDH, GEP, Decision Trees, Random Forests.	Models that could forecast CO <sub>2</sub> diffusivity in brine were established with the aid of ML.	
Nait Amar et al. (2020a)	MLP, GEP, GMDH	Numerous ML methods were employed to predict the viscosity of CO <sub>2</sub> at high pressure and temperature.	
Nait Amar et al. (2020b)	MLP, RBFNN, CMIS, CMIS-GMDH	Models that could forecast CO <sub>2</sub> thermal conductivity were established with the aid of ML.	

matrix through iteration, HM with PCA-based error model, HM with PCA-based error model and noise covariance matrix, HM with PCA-based error model and considering second-order errors, and HM with PCA-based error model and update of total error covariance matrix through iteration. They deduced that the last three algorithms yielded models with high fidelity. Liu and Durlafsky (2020) also illustrated the use of optimization-based PCA (O-PCA) and CNN-based PCA as geological parametrization techniques to represent the model properties of complex reservoirs. These techniques were coupled with the MADS to do HM. Also, the proxy-based Markov Chain Monte Carlo algorithm was successfully employed with the Embedded Discrete Fracture Model (EDFM) to conduct AHM on the oil well in Vaca Muerta shale (Dachanuwattana et al., 2018). An ensemble smoother neural network (ES-NN) that comprised ensemble smoother (ES) and convolutional autoencoder (CAE) was built and used to HM the channelized reservoirs by Kim et al. (2020). They stated that the ES-NN produced better performance than the ensemble smoother-multiple data assimilation (ES-MDA).

As discussed before, metaheuristic algorithm has been efficiently proven successful as an optimization algorithm in HM. Schulze-Riegert et al. (2002) applied evolutionary algorithms to conduct HM of a sophisticated synthetic reservoir model of a North Sea reservoir. Karimi et al. (2017) used GA along with the proxy model, which was the RSM of a 3D giant reservoir model, to do HM. Kriging proxy modeling and Sobol sampling sequence were applied by Shams et al. (2019) to do AHM by implementing three metaheuristic algorithms such as Firefly Optimization (FFO), Bee Colony Optimization (BCO), and Harmony Search Optimization (HSO). Shahkarami et al. (2018) illustrated the AHM by implementing the technology of pattern recognition. They established SRM of PUNQ-S3 reservoir model by applying ten realizations and coupled the SRM with DE to perform the AHM. In addition, He et al. (2016) applied a similar methodology to develop a proxy model of SACROC unit (Scurry Area Canyon Reef Operational Committee) which was the main part of the Kelly Snyder field in the Permian Basin. They also successfully coupled the proxy with DE to do AHM. Riazi et al. (2016) demonstrated the use of LSSVM to develop a proxy model of a fractured reservoir. Thereafter, they successfully implemented PSO and ICA to do AHM. Rana et al. (2018) suggested applying Gaussian Process-based Proxy Modeling and Variogram-based Sensitivity Analysis

(GP-VARS) on the PUNQ-S3 reservoir to solve the HM problem. They mentioned that this methodology was four times computationally less demanding than using DE on the numerical simulation to do HM. The literature on History Matching is summarized in Table 8.

## 5. Pros, Cons, and Other Discussions

### 5.1. Pros

As briefly mentioned, one of the main advantages of applying ML-based approaches in the context of reservoir simulation, is the reduction of computational footprints. Even with the current improvements in computational power, numerical simulation of a very sophisticated reservoir model may take a few months in field development studies. Therefore, it is essential to find alternatives that can speed up the calculation. This is where the intelligent proxy can contribute. If an intelligent proxy model with high fidelity is successfully established, any decision problem related to reservoir management can be handled much more quickly. Thus, further inconvenience can be avoided especially when any relevant reservoir management plan needs to be updated at a high frequency.

In addition, the mechanism of the ML-based methods is very comprehensible as it generally does not involve complicated mathematical equations. Hence, when it comes to application, we believe that it will not pose any additional challenges. Albeit there are some contempents mentioning that ML-based methods are "black-box", we do not completely abide by this opinion as we think the formulations of ML-based approaches are not as opaque as claimed. Fundamentally, these methods are explainable through mathematics. For instance, the mechanism of ANN is established by treating the nodes as neurons in the human brain. Thereafter, the weights and biases which connect the nodes in different layers are continuously adjusted using any algorithm to enable the ANN to achieve learning. From this, if we can perceive how the ML-based methods work mathematically, the implementation should be convenient. Another benefit of implementing the ML-based models, particularly in the case of TDM, pertains to the exclusion of assumptions and simplifications of physics. This is different from applying the physics-based models that might still require a few assumptions to forecast the production from a reservoir which can be problematic in dealing with real field data. In other words, the complex physics of the system might not be captured well with assumptions. In this context, data acts as a guide to the solution.

Based on the previously discussed literature, the petroleum industry is gradually gaining maturity in this domain of technology. ML-based methods offer high robustness in terms of application. Robustness here indicates that these methods can generally solve any kind of engineering problem if the problem is well-formulated, and the data are properly prepared. Aside from reservoir engineering, the use of ML-based methods in drilling engineering (Barbosa et al., 2019; Mahmoud et al., 2021; Tunkiel et al., 2020), production engineering (Huang and Chen, 2021; Wei et al., 2021; Zhong et al., 2020), petrophysics (Ali et al., 2021; Blanes de Oliveira and de Carvalho Carneiro, 2021; Osarogiabon et al., 2020), etc. has been successful. Thus, they have been termed panacea for most problems. We would like to emphasize that the use of ML-based methods ought to be upheld but should not be treated as the only solution. In this case, we refer to the hand-shaking protocol proposed by Ertekin and Sun (2019).

### 5.2. Cons

There are also some limitations associated with the use of ML-based methods. One of them includes long training time caused by a large database. One needs to consider a trade-off between the size of the database and training time when he or she plans to build intelligent proxies. We reckon that creativity is required in the phase of problem formulation to avoid long training time in the later stages. The benefit of

**Table 8**  
Summary of Literature in the Domain of History Matching.

Literature	Methods	Remarks	Assumptions / Limitations
Sampaio et al. (2009)	FNN	Using FNN to perform History Matching.	Input parameters of FNN were assumed. The size of the training group was assumed. The simplicity of the case study. Models were meant for case-specific applications. A limited number of uncertain variables.
Shahkarami et al. (2014b)	FNN/ Fuzzy Logic	Implementing SRM in the workflow of AI-Assisted History Matching in a synthetic but heterogeneous reservoir model.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Masoudi et al. (2020)	Deconvolutional Neural Networks	Applying TDM to conduct History Matching in a highly sophisticated field in Malaysia.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Illarionov et al. (2020)	FNN	Doing gradient-based History Matching with the help of FNN on a field model.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Chaki et al. (2020)	Deep Neural Network / RNN	Performing History Matching on the Brugge field model.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Honorio et al. (2015)	Piecewise Reconstruction from a Dictionary (PRaD)/ pluri-PCA	Developing an assisted History Matching with PRaD and pluri-PCA based on a case study of geologically complex reservoirs.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Rammy et al. (2020)	PCA-based error model	Integrating different approaches to the PCA-based error model in the History Matching workflow on a case study.	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.
Liu and Durlöfsky (2020)	CNN-based PCA/ MADS	Proposing the use of CNN-PCA for geological parameterization	Models were meant for case-specific applications. No guideline on determining the sequence of separate TDMS. Assumption of limited prior knowledge of the geological parameters. Adaptation of the workflow concerning production rates data was not considered. Testing of the suggested methodology was required for a more complex reservoir model. Limited input parameters were considered. Assumption of independence of measurement errors.

**Table 8 (continued)**

Literature	Methods	Remarks	Assumptions / Limitations
		in the workflow of History Matching.	available in one of the case studies. Random noise was assumed to be independent. Assumption of full parallelization. Assumption of uniform distribution of uncertain parameters. Production of the well was assumed to be at a BHP of 500 psi for 8000 days. Uniform distribution of fractures. Assuming the time of measurement of observation data during History Matching. Each facies was assumed to have a constant permeability value. Multi-dimensional search space was assumed for the reservoir studied. Unavailability of information on reservoir beyond geostatistical, geological, seismic, and history data. Independence of measurement errors. Parameters were correlated in the region identified. Homogeneous porosity of 0.30.
Dachanuwattana et al. (2018)	K-NN algorithm/ Markov Chain Monte Carlo (MCMC)	Demonstrating the use of K-NN-based and MCMC-based proxies to history match a shale oil well.	The proposed methodology might not be computationally favorable with reservoirs of more than 20 wells.
Kim et al. (2020)	Ensemble Smoother-Neural Network (ES-NN)	Presenting the use of ES-NN in the workflow of History Matching.	Models were meant for case-specific applications.
Schulze-Riegert et al. (2002)	Evolutionary Algorithm in a multipurpose Environment for Parallel Optimization (MEPO)	Illustrating the application of the evolutionary algorithm in the context of MEPO in the history matching of a complex black oil model.	Reservoir properties were assumed to be measured at well locations. Models were meant for case-
Karimi et al. (2017)	Genetic Algorithm	Incorporating GA in the History Matching with the use of a proxy model.	
Shams et al. (2019)	ANN/ GA, PSO, Firefly Algorithm, Bee Colony, Harmony Search	Introducing the use of 3 nature-inspired algorithms in the History Matching along with a proxy model.	
Shahkarami et al. (2018)	FNN/ DE	Presenting the coupling of SRM and DE for History Matching in PUNQ-S3.	

(continued on next page)

Table 8 (continued)

Literature	Methods	Remarks	Assumptions / Limitations
He et al. (2016)		Presenting the coupling of SRM and DE for History Matching in the SACROC unit.	specific applications. Models were meant for case-specific applications.
Riazi et al. (2016)	LSSVM/ PSO, ICA	Establishing the LSSVM-based proxy and coupling it with the algorithms for History Matching in a fractured reservoir.	Properties of fractures were assumed to be homogeneous.
Rana et al. (2018)	Gaussian Process proxy / Variogram-based sensitivity analysis	Illustrating the efficient assisted History Matching with Gaussian Process proxy in PUNQ-S3.	Lacking validation of the proposed workflow in a more complex reservoir.

intelligent proxy modeling is better demonstrated in the case of very complicated and heterogeneous reservoir models in which the simulation time would exceed that of neural network training by certain orders of magnitude (Mohaghegh, 2017a, 2011). This implies that having an intelligent proxy for a simplistic case does not showcase its real potential. However, having an intelligent proxy to capture a sophisticated physical relationship is noteworthy. The overfitting issue is another problem that needs to be dealt with when ML-based methods are applied. If the intelligent proxy is not well-trained, the data partitioning and training will have to be repeated. Mitigating overfitting can be laborious depending upon the complexity of the database. In addition, building intelligent proxies requires a very clear objective. Thus, it is not a one-size-fits-all model. This limitation may hinder some reservoir engineers from tending to attempt intelligent modeling. In terms of modeling with real field data, only the database from a brown field is deemed reliable in developing a useful DDM. This is because the amount of data should be sufficiently big to reflect the physics of fluid flow throughout a long period of production. In this case, another limitation also arises where there might be some missing data points during the collection of real field data for establishing a DDM. Hence, as recommended by Mohaghegh (2005), a viable solution is doing statistical averaging.

### 5.3. Other Discussions

Our survey also touched upon the application of metaheuristic algorithms along with intelligent models. Several studies (Nait Amar et al., 2018b, 2018c) proposed that when the metaheuristic algorithms are followed by conventional backpropagation algorithms in neural network training, the respective ANN illustrates better predictability. In addition, for intelligent proxy modeling, implementing metaheuristic algorithms is relatively more explicit and convenient than employing the derivative-based approaches because these algorithms do not require the approximation of the gradient. Therefore, applying them can be convenient if the corresponding mechanism can be mathematized accordingly. Ezugwu et al. (2020) illustrated the benefits and drawbacks of applying 12 metaheuristic algorithms: Cuckoo Search, DE, GA, PSO, Symbiotic Organism Search, FA, ACO, Bat Algorithm, Flower Pollination Algorithm, ABC Algorithm, Bee Algorithm, and Inverse Weed Optimization. In general, most of these algorithms have a better ability to converge to the global optimum whereas some of them might have low convergence rates and yield partially optimal results. Therefore, it is recommended to understand both the advantages and disadvantages of

any chosen metaheuristic algorithm before employment. According to our studies, metaheuristic algorithms illustrate a very huge potential to be extensively applied in different domains of reservoir engineering.

To further generalize the application of ML-based methods in reservoir engineering, especially in the intelligent proxy modeling of NRS, we have summarized a few areas which might need more scrutiny. The first area is the sampling strategies. Our investigation reveals that a more efficient sampling method can be used to enable the development of more robust intelligent proxies. The efficiency of the sampling methods is defined as its ability to retrieve samples that can cover the solution space as extensively as possible. In this aspect, we opine that the coupling of two different sampling strategies, namely Latin Hypercube method and Sobol sequence, as initiated by Dige and Diwekar (2018), can be treated as an alternative to assess whether a better intelligent proxy can be developed. Exploring any better feature selection method to mitigate the curse of dimensionality is also thinkable. About this, Mohaghegh (2017a) has initiated the application of fuzzy pattern recognition in selecting more useful input variables in proxy modeling. However, we reckon that other approaches, viz. mutual information method based on Shannon entropy in information theory (Shannon, 1948; Thanh et al., 2022), can be considered to verify whether improvement can be achieved. We would like to emphasize that if an intelligent proxy is developed upon the results of a numerical model, this proxy can only act as a complement. This is because the source of data is the NRS. When the data come from real field measurements, it is however a research question to investigate whether the intelligent proxy can completely replace the NRS in solving reservoir management-related problems. To the best of our knowledge, there are not many studies that discuss the development of coupled ML-metaheuristic paradigm while there are numerous discussions regarding the separate use of ML and metaheuristic algorithms. We hope that this survey can provide insights to the research community to further explore the potential of coupled ML-metaheuristic paradigm in the context of reservoir engineering.

## 6. Summary

In this work, we have surveyed the employment of ML methods and coupled ML-metaheuristic paradigm in developing proxies of numerical simulation models where only a limited number of literature studied the latter. Nevertheless, the respective literature were included along with other articles that mainly touched upon the employment of ML in the domain of reservoir simulation to highlight the robustness of ML methods. We illustrated the general framework and several suggestions, including proper identification of the objective of proxies and data normalization, that could be implemented to successfully develop an intelligent proxy model. Albeit these recommendations seemingly appear to be trivial, it happens that they could have been overlooked in proxy modeling. In addition, we demonstrated and discussed the application of ML approaches and the hybrid approach in different domains of reservoir engineering such as well placement, monitoring production parameters, miscible gas injection, waterflooding, CCS, WAG, other EOR methods, and history matching. We also briefed on the pros and cons of using ML approaches and metaheuristic algorithms. We opined that several aspects associated with intelligent proxy modeling need to be addressed to achieve further maturity in the application of this technology. In general, we can infer that the ML methods and the coupled paradigm provide useful insights into the resolution of reservoir management issues. Furthermore, ANN is portrayed as very flexible to be implemented to build intelligent proxies. Therefore, despite not being a "one-size-fits-all" solution, these methods ought to be further explored due to their huge potential. We also conclude that the potential of coupled ML-metaheuristic paradigm can still be further investigated mostly in the context of reservoir simulation. This survey paper aims at inspiring and providing insights for other researchers and engineers concerning this.

## 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

No data was used for the research described in the article.

## Acknowledgement

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)).

## References

- Abdullah, M., Emami-Meybodi, H., Ertekin, T., 2019. Development and application of an artificial neural network tool for chemical EOR field implementations. In: Proceedings of the Society of Petroleum Engineers - SPE Europec Featured at 81st EAGE Conference and Exhibition 2019. <https://doi.org/10.2118/195492-ms>.
- Afzali, S., Rezaei, N., Zendeheboudi, S., 2018. A comprehensive review on enhanced oil recovery by water alternating gas (WAG) injection. *Fuel*. <https://doi.org/10.1016/j.fuel.2018.04.015>.
- Ahmadi, M.A., Pouladi, B., Barghi, T., 2016. Numerical modeling of CO<sub>2</sub> injection scenarios in petroleum reservoirs: application to CO<sub>2</sub> sequestration and EOR. *J. Nat. Gas Sci. Eng.* 30, 38–49. <https://doi.org/10.1016/j.jngse.2016.01.038>.
- Ahmadi, M.A., Zendeheboudi, S., James, L.A., 2018. Developing a robust proxy model of CO<sub>2</sub> injection: coupling Box–Behken design and a connectionist method. *Fuel* 215, 904–914. <https://doi.org/10.1016/j.fuel.2017.11.030>.
- Ahmed, T., 2018. Reservoir engineering handbook. Gulf professional publishing.
- Al-Alwani, M.A., Britt, L., Dunn-Norman, S., Alkinani, H.H., Al-Hameedi, A.T., Al-Attar, A., 2019. Production performance estimation from stimulation and completion parameters using machine learning approach in the Marcellus shale. In: Proceedings of the 53rd U.S. Rock Mechanics/Geomechanics Symposium.
- Alakeely, A., Horne, R.N., 2020. Simulating the behavior of reservoirs with convolutional and recurrent neural networks. *SPE Reservoir Evaluation and Engineering*. <https://doi.org/10.1016/j.petrol.2021.108602>.
- Alatrach, Y., Mata, C., Omrani, P.S., Saputelli, L., Narayanan, R., Hamdan, M., 2020. Prediction of well production event using machine learning algorithms. In: Proceedings of the Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference 2020, ADIP 2020. <https://doi.org/10.2118/202961-ms>.
- Alenezi, F., Mohaghegh, S., 2017. Developing a smart proxy for the SACROC water-flooding numerical reservoir simulation model. In: Proceedings of the SPE Western Regional Meeting Proceedings. <https://doi.org/10.2118/185691-ms>.
- Ali, M., Jiang, R., Huolin, M., Pan, H., Abbas, K., Ashraf, U., Ullah, J., 2021. Machine learning - A novel approach of well logs similarity based on synchronization measures to predict shear sonic logs. *J. Pet. Sci. Eng.* 203, 108602 <https://doi.org/10.1016/j.petrol.2021.108602>.
- Alkhalaf, A., Isichei, O., Ansari, N., Milad, R., 2019. Utilizing machine learning for a data driven approach to flow rate prediction. In: Proceedings of the Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference 2019, ADIP 2019. <https://doi.org/10.2118/197266-ms>.
- Alkinani, H.H., Al-Hameedi, A.T.T., Dunn-Norman, S., Flori, R.E., Alsaba, M.T., Amer, A.S., 2019. Applications of artificial neural networks in the petroleum industry: a review. In: Proceedings of the SPE Middle East Oil and Gas Show and Conference, MEOS, Proceedings. <https://doi.org/10.2118/195072-ms>.
- Amaechi, U.C., Ikpeka, P.M., Xianlin, M., Ugwu, J.O., 2019. Application of machine learning models in predicting initial gas production rate from tight gas reservoirs. *Rud. Geol. Naft. Zb.* <https://doi.org/10.17794/rgn.2019.3.4>.
- Ani, M., Oluyemi, G., Petrovski, A., Sina, R.G., 2016. Reservoir uncertainty analysis: the trends from probability to algorithms and machine learning. In: Proceedings of the Society of Petroleum Engineers - SPE Intelligent Energy International Conference and Exhibition. <https://doi.org/10.2118/181049-ms>.
- Artun, E., 2017. Characterizing interwell connectivity in waterflooded reservoirs using data-driven and reduced-physics models: a comparative study. *Neural Comput. Appl.* <https://doi.org/10.1007/s00521-015-2152-0>.
- Bai, T., Tahmasebi, P., 2020. Efficient and data-driven prediction of water breakthrough in subsurface systems using deep long short-term memory machine learning. *Comput. Geosci.* <https://doi.org/10.1007/s10596-020-10005-2>.
- Balaji, K., Rabiei, M., Suicmez, V., Canbaz, H., Agbarzevya, Z., Tek, S., Bulut, U., Temizel, C., 2018. Status of data-driven methods and their applications in oil and gas industry. In: Proceedings of the Society of Petroleum Engineers - SPE Europec Featured at 80th EAGE Conference and Exhibition 2018. <https://doi.org/10.2118/190812-ms>.
- Barbosa, L.F.F.M., Nascimento, A., Mathias, M.H., de Carvalho, J.A., 2019. Machine learning methods applied to drilling rate of penetration prediction and optimization - A review. *J. Pet. Sci. Eng.* 183, 106332 <https://doi.org/10.1016/j.petrol.2019.106332>.
- Barnard, E., Wessels, L.F.A., 1992. Extrapolation and interpolation in neural network classifiers. *IEEE Control Syst.* 12. <https://doi.org/10.1109/37.158898>.
- Belazreg, L., Mahmood, S.M., 2020. Water alternating gas incremental recovery factor prediction and WAG pilot lessons learned. *J. Pet. Explor. Prod. Technol.* <https://doi.org/10.1007/s13202-019-0694-x>.
- Belazreg, L., Mahmood, S.M., Aulia, A., 2020. Random forest algorithm for CO<sub>2</sub> water alternating gas incremental recovery factor prediction. *Int. J. Adv. Sci. Technol.*
- Belazreg, L., Mahmood, S.M., Aulia, A., 2019. Novel approach for predicting water alternating gas injection recovery factor. *J. Pet. Explor. Prod. Technol.* <https://doi.org/10.1007/s13202-019-0673-2>.
- Bian, X.Q., Han, B., Du, Z.M., Jaubert, J.N., Li, M.J., 2016. Integrating support vector regression with genetic algorithm for CO<sub>2</sub>-oil minimum miscibility pressure (MMP) in pure and impure CO<sub>2</sub> streams. *Fuel*. <https://doi.org/10.1016/j.fuel.2016.05.124>.
- Bikmukhametov, T., Jäschke, J., 2020. Combining machine learning and process engineering physics towards enhanced accuracy and explainability of data-driven models. *Comput. Chem. Eng.* <https://doi.org/10.1016/j.compchemeng.2020.106834>.
- Blanes de Oliveira, L.A., de Carvalho Carneiro, C., 2021. Synthetic geochemical well logs generation using ensemble machine learning techniques for the Brazilian pre-salt reservoirs. *J. Pet. Sci. Eng.* 196, 108080 <https://doi.org/10.1016/j.petrol.2020.108080>.
- British Petroleum, 2021. Statistical Review of World Energy 2021. *BP Energy Outlook 2021* 70.
- Busby, D., Pivov, F., Tadjer, A., 2017. Use of data analytics to improve well placement optimization under uncertainty. In: Proceedings of the Society of Petroleum Engineers - SPE Abu Dhabi International Petroleum Exhibition and Conference 2017. <https://doi.org/10.2118/188265-ms>.
- Cao, Q., Banerjee, R., Gupta, S., Li, J., Zhou, W., Jeyachandra, B., 2016. Data driven production forecasting using machine learning. In: Proceedings of the Society of Petroleum Engineers - SPE Argentina Exploration and Production of Unconventional Resources Symposium. <https://doi.org/10.2118/180984-ms>.
- Chaikina, I.A., Gates, I.D., 2021. A machine learning model for predicting multi-stage horizontal well production. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.108133>.
- Chaki, S., Zagayevskiy, Y., Shi, X., Wong, T., Noor, Z., 2020. Machine learning for proxy modeling of dynamic reservoir systems: deep neural network DNN and recurrent neural network RNN applications. In: Proceedings of the International Petroleum Technology Conference 2020. IPTC. <https://doi.org/10.2523/iptc-2018-ms>, 2020.
- Chen, G., Fu, K., Liang, Z., Sema, T., Li, C., Tontiwachuthikul, P., Idem, R., 2014. The genetic algorithm based back propagation neural network for MMP prediction in CO<sub>2</sub>-EOR process. *Fuel*. <https://doi.org/10.1016/j.fuel.2014.02.034>.
- Chen, G., Zhang, K., Zhang, L., Xue, X., Ji, D., Yao, C., Yao, J., Yang, Y., 2020. Global and local surrogate-model-assisted differential evolution for waterflooding production optimization. *SPE J.* <https://doi.org/10.2118/199357-PA>.
- Chen, Y., Zhu, Z., Lu, Y., Hu, C., Gao, F., Li, W., Sun, N., Feng, T., 2019. Reservoir recovery estimation using data analytics and neural network based analogue study. In: Proceedings of the Society of Petroleum Engineers - SPE/IATMI Asia Pacific Oil and Gas Conference and Exhibition 2019. <https://doi.org/10.2118/196487-ms>. APOG 2019.
- gon Chu, M., Min, B., Kwon, S., Park, G., Kim, S., Huy, N.X., 2020. Determination of an infill well placement using a data-driven multi-modal convolutional neural network. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2019.106805>.
- Crombecq, K., 2011. Surrogate Modeling of Computer Experiments With Sequential Experimental Design. *Ghent University*.
- Cross, T., Sathaye, K., Darnell, K., Niederhut, D., Crifasi, K., 2020. Predicting Water Production in the Williston Basin using a Machine Learning Model. <https://doi.org/10.15530/urtec-2020-2756>.
- da Silva, L.M., Avansi, G.D., Schiozer, D.J., 2020. Support vector regression for petroleum reservoir production forecast considering geostatistical realizations. *SPE Reserv. Eval. Eng.* <https://doi.org/10.2118/203828-PA>.
- Dachanuwattana, S., Jin, J., Zuloaga-Molero, P., Li, X., Xu, Y., Sepeshnoori, K., Yu, W., Miao, J., 2018. Application of proxy-based MCMC and EDFM to history match a Vaca Muerta shale oil well. *Fuel*. <https://doi.org/10.1016/j.fuel.2018.02.018>.
- Dai, Z., Middleton, R., Viswanathan, H., Fessenden-Rahn, J., Bauman, J., Pawar, R., Lee, S.-Y., McPherson, B., 2014. An integrated framework for optimizing CO<sub>2</sub> sequestration and enhanced oil recovery. *Environ. Sci. Technol. Lett.* 1, 49–54.
- Dargahi-Zarandi, A., Hemmati-Sarapardeh, A., Shateri, M., Menad, N.A., Ahmadi, M., 2020. Modeling minimum miscibility pressure of pure/impure CO<sub>2</sub>-crude oil systems using adaptive boosting support vector regression: application to gas injection processes. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2019.106499>.
- Deng, L., Pan, Y., 2020. Machine-learning-assisted closed-loop reservoir management using echo state network for mature fields under waterflood. *SPE Reserv. Eval. Eng.* <https://doi.org/10.2118/200862-PA>.
- Dige, N., Diwekar, U., 2018. Efficient sampling algorithm for large-scale optimization under uncertainty problems. *Comput. Chem. Eng.* 115, 431–454. <https://doi.org/10.1016/j.compchemeng.2018.05.007>.
- Ding, S., Lu, R., Xi, Y., Liu, G., Ma, J., 2020. Efficient well placement optimization coupling hybrid objective function with particle swarm optimization algorithm. *Appl. Soft Comput.* 95, 106511 <https://doi.org/10.1016/j.asoc.2020.106511>.
- Dong, P., Liao, X., Chen, Z., Chu, H., 2019. An improved method for predicting CO<sub>2</sub> minimum miscibility pressure based on artificial neural network. *Adv. Geo-Energy Res.* <https://doi.org/10.26804/ager.2019.04.02>.
- Ehsan, A., Leung, J.Y., Zanon, S.D., Dzurman, P.J., 2014. An integrated application of cluster analysis and artificial neural networks for SAGD recovery performance

- prediction in heterogeneous reservoirs. In: Proceedings of the Society of Petroleum Engineers - SPE Heavy Oil Conference Canada 2014. <https://doi.org/10.2118/170113-ms>.
- Enab, K., Ertekin, T., 2020. Screening and optimization of CO<sub>2</sub>-WAG injection and fish-bone well structures in low permeability reservoirs using artificial neural network. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.108268>.
- Enick, R.M., Olsen, D.K., Ammer, J.R., Schuller, W., others, 2012. Mobility and conformance control for CO<sub>2</sub> EOR via thickeners, foams, and gels—a literature review of 40 years of research and pilot tests. In: Proceedings of the SPE Improved Oil Recovery Symposium.
- Ersahin, A., Ertekin, T., 2020. Artificial neural network modeling of cyclic steam injection process in naturally fractured reservoirs. In: Proceedings of the SPE Reservoir Evaluation and Engineering. <https://doi.org/10.2118/195307-PA>.
- Ertekin, T., Sun, Q., 2019. Artificial intelligence applications in reservoir engineering: a status check. *Energies*. <https://doi.org/10.3390/en12152897>.
- Ezugwu, A.E., Adeleke, O.J., Akinyelu, A.A., Viriri, S., 2020. A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. *Neural Comput. Appl.* <https://doi.org/10.1007/s00521-019-04132-w>.
- Forrester, A.L.J., Sobester, A., Keane, A.J., 2008. *Engineering Design Via Surrogate Modelling: a Practical Guide*. J. Wiley.
- Ghriga, M.A., Grassi, B., Gareche, M., Khodja, M., Lebouachera, S.E.L., Andreu, N., Drouiche, N., 2019. Review of recent advances in polyethylenimine crosslinked polymer gels used for conformance control applications. *Polym. Bull.* 1–29.
- Gogna, A., Tayal, A., 2013. Metaheuristics: review and application. *J. Exp. Theor. Artif. Intell.* 25 <https://doi.org/10.1080/0952813X.2013.782347>.
- Guo, Z., Reynolds, A.C., 2018. Robust life-cycle production optimization with a support-vector-regression proxy. *SPE J.* <https://doi.org/10.2118/191378-PA>.
- Haley, P.J., Soloway, D., 2003. Extrapolation limitations of multilayer feedforward neural networks. doi:10.1109/ijcnn.1992.227294.
- Halton, J.H., 1960. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numer. Math.* 2 <https://doi.org/10.1007/BF01386213>.
- Hammerley, J.M., Handscomb, D.C., 1964. Monte Carlo Methods, Monte Carlo Methods. doi:10.1007/978-94-009-5819-7.
- Han, B., Bian, X., 2018. A hybrid PSO-SVM-based model for determination of oil recovery factor in the low-permeability reservoir. *Petroleum*. <https://doi.org/10.1016/j.petlm.2017.06.001>.
- Hanga, K.M., Kovalchuk, Y., 2019. Machine learning and multi-agent systems in oil and gas industry applications: a survey. *Comput. Sci. Rev.* <https://doi.org/10.1016/j.cosrev.2019.08.002>.
- Hassan, A., Elkhatny, S., Abdulraheem, A., 2019. Application of artificial intelligence techniques to predict the well productivity of fishbone wells. *Sustain.* <https://doi.org/10.3390/su11216083>.
- Hassani, H., Sarkheil, H., 2011. A proxy modeling approach to optimization horizontal well placement. In: Proceedings of the 45th US Rock Mech.
- Hassani, H., Sarkheil, H., Foroud, T., Karimpooli, S., 2011. A proxy modeling approach to optimization horizontal well placement. In: Proceedings of the 45th US Rock Mech. /Geomech. Symp.
- He, Q., Mohaghegh, S.D., Liu, Z., 2016. Reservoir simulation using smart proxy in SACROC unit - Case study. In: Proceedings of the SPE Eastern Regional Meeting. <https://doi.org/10.2118/184069-MS>.
- Hemmati-Sarapardeh, A., Amar, M.N., Soltanian, M.R., Dai, Z., Zhang, X., 2020a. Modeling CO<sub>2</sub> solubility in water at high pressure and temperature conditions. *Energy Fuels*. <https://doi.org/10.1021/acs.energyfuels.0c00114>.
- Hemmati-Sarapardeh, A., Larestani, A., Nait Amar, M., Hajirezaie, S., 2020b. Applications of Artificial Intelligence Techniques in the Petroleum Industry. Gulf Professional Publishing.
- Honorio, J., Chen, C., Gao, G., Du, K., Jaakkola, T., 2015. Integration of PCA with a novel machine learning method for reparameterization and assisted history matching geologically complex reservoirs. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/175038-ms>.
- Hourfar, F., Bidgoly, H.J., Moshiri, B., Salahshoor, K., Elkamel, A., 2019. A reinforcement learning approach for waterflooding optimization in petroleum reservoirs. *Eng. Appl. Artif. Intell.* <https://doi.org/10.1016/j.engappai.2018.09.019>.
- Huang, Z., Chen, Z., 2021. Comparison of different machine learning algorithms for predicting the SAGD production performance. *J. Pet. Sci. Eng.* 202, 108559 <https://doi.org/10.1016/j.petrol.2021.108559>.
- Illarionov, E., Temirchev, P., Voloskov, D., Gubanova, A., Koroteev, D., Simonov, M., Akhmetov, A., Margarit, A., 2020. 3D reservoir model history matching based on machine learning technology. In: Proceedings of the Society of Petroleum Engineers - SPE Russian Petroleum Technology Conference 2020. RPTC. <https://doi.org/10.2118/201924-ms>, 2020.
- Jaber, A.K., Al-Jawad, S.N., Alhuraishawy, A.K., 2019a. A review of proxy modeling applications in numerical reservoir simulation. *Arab. J. Geosci.* <https://doi.org/10.1007/s12517-019-4891-1>.
- Jaber, A.K., Alhuraishawy, A.K., Al-Bazzaz, W.H., 2019b. A data-driven model for rapid evaluation of miscible CO<sub>2</sub>-WAG flooding in heterogeneous clastic reservoirs. In: Proceedings of the Society of Petroleum Engineers - SPE Kuwait Oil and Gas Show and Conference 2019. KOGS. <https://doi.org/10.2118/198013-ms>, 2019.
- Jang, I., Oh, S., Kim, Y., Park, C., Kang, H., 2018. Well-placement optimisation using sequential artificial neural networks. *Energy Exploit.* <https://doi.org/10.1177/0144598717729490>.
- Jesmani, M., Jafarpour, B., Bellout, M.C., Foss, B., 2020. A reduced random sampling strategy for fast robust well placement optimization. *J. Pet. Sci. Eng.* 184, 106414.
- Jia, D., Liu, H., Zhang, J., Gong, B., Pei, X., Wang, Q., Yang, Q., 2020. Data-driven optimization for fine water injection in a mature oil field. *Pet. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(20\)60048-2](https://doi.org/10.1016/S1876-3804(20)60048-2).
- Jia, H., Deng, L., 2018. Water flooding flow area identification for oil reservoirs based on the method of streamline clustering artificial intelligence. *Pet. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(18\)30036-3](https://doi.org/10.1016/S1876-3804(18)30036-3).
- Kalam, S., Abu-Khamsin, S.A., Al-Yousef, H.Y., Gajbiye, R., 2020. A novel empirical correlation for waterflooding performance prediction in stratified reservoirs using artificial intelligence. *Neural Comput. Appl.* <https://doi.org/10.1007/s00521-020-05158-1>.
- Karacan, C.O., 2020. A fuzzy logic approach for estimating recovery factors of miscible CO<sub>2</sub>-EOR projects in the United States. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2019.106533>.
- Karimi, M., Mortazavi, A., Ahmadi, M., 2017. Applying an optimized proxy-based workflow for fast history matching. *Arab. J. Geosci.* <https://doi.org/10.1007/s12517-017-3247-y>.
- Karkevandi-Talkhooncheh, A., Rostami, A., Hemmati-Sarapardeh, A., Ahmadi, M., Husein, M.M., Dabir, B., 2018. Modeling minimum miscibility pressure during pure and impure CO<sub>2</sub> flooding using hybrid of radial basis function neural network and evolutionary techniques. *Fuel*. <https://doi.org/10.1016/j.fuel.2018.01.101>.
- Khan, M.R., Alnuaim, S., Tariq, Z., Abdulraheem, A., 2019. Machine learning application for oil rate prediction in artificial gas lift wells. In: Proceedings of the SPE Middle East Oil and Gas Show and Conference, MEOS, Proceedings. <https://doi.org/10.2118/194713-ms>.
- Khazali, N., Sharifi, M., Ahmadi, M.A., 2019. Application of fuzzy decision tree in EOR screening assessment. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2019.02.001>.
- Kim, S., Lee, K., Lim, J., Jeong, H., Min, B., 2020. Development of ensemble smoother-neural network and its application to history matching of channelized reservoirs. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107159>.
- Kristoffersen, B.S., Silva, T., Bellout, M., Berg, C.F., 2020. An automatic well planner for efficient well placement optimization under geological uncertainty. In: Proceedings of the ECMOR 2020 - 17th European Conference on the Mathematics of Oil Recovery. <https://doi.org/10.3997/2214-4669.202035211>.
- Li, H., Yu, H., Cao, N., Tian, H., Cheng, S., 2020. Applications of artificial intelligence in oil and gas development. *Arch. Comput. Methods Eng.* <https://doi.org/10.1007/s11831-020-09402-8>.
- Liu, Y., Durlfolsky, L.J., 2020. Multilevel strategies and geological parameterizations for history matching complex reservoir models. *SPE Journal*. <https://doi.org/10.2118/193895-PA>.
- Ma, H., Yu, G., She, Y., Gu, Y., 2019. Waterflooding optimization under geological uncertainties by using deep reinforcement learning algorithms. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/196190-ms>.
- Ma, Z., Leung, J.Y., 2020. Design of warm solvent injection processes for heterogeneous heavy oil reservoirs: a hybrid workflow of multi-objective optimization and proxy models. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107186>.
- Mahmoud, A.A., Elkhatny, S., Al-Abduljabbar, A., 2021. Application of machine learning models for real-time prediction of the formation lithology and tops from the drilling parameters. *J. Pet. Sci. Eng.* 203, 108574 <https://doi.org/10.1016/j.petrol.2021.108574>.
- Masini, S.R., Goswami, S., Kumar, A., Chennakrishnan, B., 2019. Decline curve analysis using artificial intelligence. In: Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference 2019. ADIP. <https://doi.org/10.2118/197932-ms>, 2019.
- Masoudi, R., Mohaghegh, S.D., Yingling, D., Ansari, A., Amat, H., Mohamad, N., Sabzabadi, A., Mandel, D., 2020. Subsurface analytics case study: reservoir simulation and modeling of highly complex offshore field in Malaysia, using artificial intelligent and machine learning. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/201693-ms>.
- McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*. <https://doi.org/10.2307/1268522>.
- Mesbah, M., Shahsavari, S., Soroush, E., Rahaei, N., Rezakazemi, M., 2018. Accurate prediction of miscibility of CO<sub>2</sub> and supercritical CO<sub>2</sub> in ionic liquids using machine learning. *J. CO<sub>2</sub> Util.* <https://doi.org/10.1016/j.jcou.2018.03.004>.
- Mohaghegh, S., 2018. Data-Driven Analytics for the Geological Storage of CO<sub>2</sub>. CRC Press, Boca Raton, Florida.
- Mohaghegh, S.D., 2022. Smart Proxy Modeling: Artificial Intelligence and Machine Learning in Numerical Simulation. CRC Press, Boca Raton. <https://doi.org/10.1201/9781003242581>.
- Mohaghegh, S.D., 2020. Subsurface analytics: contribution of artificial intelligence and machine learning to reservoir engineering, reservoir modeling, and reservoir management. *Pet. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(20\)60041-6](https://doi.org/10.1016/S1876-3804(20)60041-6).
- Mohaghegh, S.D., 2017a. Data-Driven Reservoir Modeling. Society of Petroleum Engineers.
- Mohaghegh, S.D., 2017b. Shale Analytics, Shale Analytics. doi:10.1007/978-3-319-48753-3.
- Mohaghegh, S.D., 2013. Reservoir modeling of shale formations. *J. Nat. Gas Sci. Eng.* <https://doi.org/10.1016/j.jngse.2013.01.003>.
- Mohaghegh, Shahab Dean, 2011. Reservoir simulation and modeling based on artificial intelligence and data mining (AI&DM). *J. Nat. Gas Sci. Eng.* <https://doi.org/10.1016/j.jngse.2011.08.003>.
- Mohaghegh, S.D., 2005. Recent developments in application of artificial intelligence in petroleum engineering. *JPT J. Pet. Technol.* 57. <https://doi.org/10.2118/89033-JPT>.

- Mohaghegh, S.D., Amiri, S., Gholami, V., Gaskari, R., Bromhal, G.S., 2012a. Grid-Based Surrogate Reservoir Modeling (SRM) For Fast Track Analysis of Numerical Reservoir Simulation Models At the Gridblock Level. SPE Western Regional Meeting. Society of Petroleum Engineers. <https://doi.org/10.2118/153844-MS>.
- Mohaghegh, S.D., Hafez, H., Gaskari, R., Haajizadeh, M., Kenawy, M., 2006. Uncertainty analysis of a giant oil field in the middle east using surrogate reservoir model. In: Proceedings of the 12th Abu Dhabi International Petroleum Exhibition and Conference, ADIPEC 2006: Meeting the Increasing Oil and Gas Demand Through Innovation. <https://doi.org/10.2523/101474-ms>.
- Mohaghegh, S.D., Liu, J., Gaskari, R., Maysami, M., Olukoko, O., 2012b. Application of well-based surrogate reservoir models (SRMs) to two offshore fields in Saudi Arabia, case study. In: Proceedings of the Society of Petroleum Engineers Western Regional Meeting 2012. <https://doi.org/10.2118/153845-ms>.
- Mohaghegh, S.D., Liu, J., Gaskari, R., Maysami, M., Olukoko, O.A., 2012c. Application of surrogate reservoir model (SRM) to an onshore green field in Saudi Arabia; case study. In: Proceedings of the Society of Petroleum Engineers - North Africa Technical Conference and Exhibition 2012, NATC 2012: Managing Hydrocarbon Resources in a Changing Environment. <https://doi.org/10.2118/151994-ms>.
- Mohagheghian, E., James, L.A., Haynes, R.D., 2018. Optimization of hydrocarbon water alternating gas in the Norne field: application of evolutionary algorithms. *Fuel*. <https://doi.org/10.1016/j.fuel.2018.01.138>.
- Mohammadi, K., Ameli, F., 2019. Toward mechanistic understanding of Fast SAGD process in naturally fractured heavy oil reservoirs: application of response surface methodology and genetic algorithm. *Fuel* 253, 840–856.
- Mohammadpoor, M., Torabi, F., 2020. Big Data analytics in oil and gas industry: an emerging trend. *Petroleum*. <https://doi.org/10.1016/j.petlm.2018.11.001>.
- Morales, A., Nasrabadi, H., Zhu, D., 2010. A modified genetic algorithm for horizontal well placement optimization in gas condensate reservoirs. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/135182-ms>.
- Morales, A.N., Nasrabadi, H., Zhu, D., 2011. A new modified genetic algorithm for well placement optimization under geological uncertainties. In: Proceedings of the 73rd European Association of Geoscientists and Engineers Conference and Exhibition 2011: Unconventional Resources and the Role of Technology. Incorporating SPE EUROPEC 2011. <https://doi.org/10.2118/143617-ms>.
- Mousavi, S.M., Jabbari, H., Darab, M., Nourani, M., Sadeghnejad, S., 2020. Optimal well placement using machine learning methods: multiple reservoir scenarios. In: Proceedings of the Society of Petroleum Engineers - SPE Norway Subsurface Conference 2020. <https://doi.org/10.2118/200752-ms>.
- Na-udom, A., Rungtrattanabul, J., 2015. A comparison of artificial neural network and kriging model for predicting the deterministic output response. *NU. Int. J. Sci. 10*, 1–9.
- Nait Amar, M., Ghriga, M.A., Ouaer, H., El Amine Ben Seghier, M., Pham, B.T., Andersen, P.O., 2020a. Modeling viscosity of CO<sub>2</sub> at high temperature and pressure conditions. *J. Nat. Gas Sci. Eng.* <https://doi.org/10.1016/j.jngse.2020.103271>.
- Nait Amar, M., Hemmati-Sarapardeh, A., Varamesh, A., Shamsirband, S., 2019. Predicting solubility of CO<sub>2</sub> in brine by advanced machine learning systems: application to carbon capture and sequestration. *J. CO<sub>2</sub> Util.* <https://doi.org/10.1016/j.jcou.2019.05.009>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., 2020. Prediction of CO<sub>2</sub> diffusivity in brine using white-box machine learning. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107037>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Ng, C.S.W., Zeraibi, N., 2021. Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms. *J. Pet. Sci. Eng.* 109308 <https://doi.org/10.1016/j.petrol.2021.109308>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Zeraibi, N., 2020b. Predicting thermal conductivity of carbon dioxide using group of data-driven models. *J. Taiwan Inst. Chem. Eng.* <https://doi.org/10.1016/j.jtice.2020.08.001>.
- Nait Amar, M., Zeraibi, N., 2019. An efficient methodology for multi-objective optimization of water alternating CO<sub>2</sub> EOR process. *J. Taiwan Inst. Chem. Eng.* <https://doi.org/10.1016/j.jtice.2019.03.016>.
- Nait Amar, M., Zeraibi, N., 2018. Application of hybrid support vector regression artificial bee colony for prediction of MMP in CO<sub>2</sub>-EOR process. *Petroleum*. doi:10.1016/j.petlm.2018.08.001.
- Nait Amar, M., Zeraibi, N., Jahanbani Ghahfarokhi, A., 2020c. Applying hybrid support vector regression and genetic algorithm to water alternating CO<sub>2</sub> gas EOR. *Greenh. Gases Sci. Technol.* <https://doi.org/10.1002/ghg.1982>.
- Nait Amar, M., Zeraibi, N., Redouane, K., 2018a. Optimization of WAG process using dynamic proxy, genetic algorithm and ant colony optimization. *Arab. J. Sci. Eng.* <https://doi.org/10.1007/s13369-018-3173-7>.
- Nait Amar, M., Zeraibi, N., Redouane, K., 2018b. Bottom hole pressure estimation using hybridization neural networks and grey wolves optimization. *Petroleum*. <https://doi.org/10.1016/j.petlm.2018.03.013>.
- Nait Amar, M., Zeraibi, N., Redouane, K., 2018c. Pure CO<sub>2</sub>-oil system minimum miscibility pressure prediction using optimized artificial neural network by differential evolution. *Pet. Coal* 60.
- Negash, B.M., Yaw, A.D., 2020. Artificial neural network based production forecasting for a hydrocarbon reservoir under water injection. *Pet. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(20\)60055-6](https://doi.org/10.1016/S1876-3804(20)60055-6).
- Ng, C.S.W., Ghahfarokhi, A.J., Nait Amar, M., 2022a. Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm. *Petroleum*. doi:10.1016/J.PETLM.2021.12.008.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2022b. Well production forecast in Volve field: application of rigorous machine learning techniques and metaheuristic algorithm. *J. Pet. Sci. Eng.* 208, 109468 <https://doi.org/10.1016/J.PETROL.2021.109468>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2021a. Well production forecast in volve field: application of machine learning techniques and metaheuristic algorithm. *Pet. Sci. Technol.*
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2021b. Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization. *J. Pet. Explor. Prod. Technol.* <https://doi.org/10.1007/s13202-021-01199-x>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., Torseter, O., 2021c. Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application. *Nat. Resour. Res.* 30, 2431–2462. <https://doi.org/10.1007/s11053-021-09844-2>.
- Nwachukwu, A., Jeong, H., Pycrz, M., Lake, L.W., 2018a. Fast evaluation of well placements in heterogeneous reservoir models using machine learning. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2018.01.019>.
- Nwachukwu, A., Jeong, H., Sun, A., Pycrz, M., Lake, L.W., 2018b. Machine learning-based optimization of well locations and WAG parameters under geologic uncertainty. In: Proceedings of the SPE Symposium on Improved Oil Recovery. <https://doi.org/10.2118/190239-ms>.
- Omrani, P.S., Vecchia, A.L., Dobrovolschi, I., van Baalen, T., Poort, J., Octaviano, R., Binn-Tahir, H., Muñoz, E., 2019. Deep learning and hybrid approaches applied to production forecasting. In: Proceedings of the Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference 2019. <https://doi.org/10.2118/197498-ms>.
- Onwunali, J.E., Litvak, M.L., Durlfolsky, L.J., Aziz, K., 2008. Application of statistical proxies to speed up field development optimization procedures. In: Proceedings of the Abu Dhabi International Petroleum Exhibition and Conference. Society of Petroleum Engineers. <https://doi.org/10.2118/117323-MS>.
- Osarogiabon, A.U., Olorunbi, O., Khan, F., Venkatesan, R., Butt, S., 2020. Gamma ray log generation from drilling parameters using deep learning. *J. Pet. Sci. Eng.* 195, 107906 <https://doi.org/10.1016/j.petrol.2020.107906>.
- Otchere, D.A., Arbi Ganat, T.O., Gholami, R., Ridha, S., 2021. Application of supervised machine learning paradigms in the prediction of petroleum reservoir properties: comparative analysis of ANN and SVM models. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.108182>.
- Panja, P., Velasco, R., Pathak, M., Deo, M., 2018. Application of artificial intelligence to forecast hydrocarbon production from shales. *Petroleum*. <https://doi.org/10.1016/j.petlm.2017.11.003>.
- Panjaliadeh, H., Alizadeh, A., Ghazanfari, M., Alizadeh, N., 2015. Optimization of the WAG injection process. *Pet. Sci. Technol.* 33, 294–301.
- Parada, C.H., Ertekin, T., 2012. A new screening tool for improved oil recovery methods using artificial neural networks. In: Proceedings of the Society of Petroleum Engineers Western Regional Meeting 2012. <https://doi.org/10.2118/153321-ms>.
- Plakšina, T., 2019. *Modern Data Analytics: Applied AI and Machine Learning For Oil and Gas Industry*. Library and archives of Canada.
- Pouladi, B., Keshavarz, S., Sharifi, M., Ahmadi, M.A., 2017. A robust proxy for production well placement optimization problems. *Fuel*. <https://doi.org/10.1016/j.fuel.2017.06.030>.
- Rahmanifard, H., Alimohammadi, H., Gates, I., 2020. Well Performance Prediction in Montney Formation Using Machine Learning Approaches. doi:10.15530/urtec-2020-2465.
- Rahmanifard, H., Plakšina, T., 2019. Application of artificial intelligence techniques in the petroleum industry: a review. *Artif. Intell. Rev.* <https://doi.org/10.1007/s10462-018-9612-8>.
- Rammay, M.H., Elsheikh, A.H., Chen, Y., 2020. Robust algorithms for history matching of imperfect subsurface models. *SPE J.* <https://doi.org/10.2118/193838-pa>.
- Rana, S., Ertekin, T., King, G.R., 2018. An efficient assisted history matching and uncertainty quantification workflow using Gaussian processes proxy models and variogram based sensitivity analysis: GP-VARS. *Comput. Geosci.* <https://doi.org/10.1016/j.cageo.2018.01.019>.
- Redouane, K., Zeraibi, N., Nait Amar, M., 2019. Adaptive surrogate modeling with evolutionary algorithm for well placement optimization in fractured reservoirs. *Appl. Soft Comput. J.* <https://doi.org/10.1016/j.asoc.2019.03.022>.
- Rezaian, A., Kordestany, A., Haghghat, S.M., 2010. Experimental and artificial neural network approaches to predict the effect of PVA (Poly Vinyl Acetate) on the rheological properties of water and crude oil in EOR processes. In: Proceedings of the Society of Petroleum Engineers - Nigeria Annual International Conference and Exhibition 2010. NAICE. <https://doi.org/10.2118/140680-ms>.
- Riazi, S.H., Zargar, G., Baharimoghdam, M., Moslemi, B., Sharifi Darani, E., 2016. Fractured reservoir history matching improved based on artificial intelligent. *Petroleum*. <https://doi.org/10.1016/j.petlm.2016.09.001>.
- Ross, T.J., 2010. *Fuzzy Logic with Engineering Applications: Third Edition*, Fuzzy Logic with Engineering Applications: Third Edition. doi:10.1002/9781119994374.
- Russell, S., Norvig, P., 2010. *Artificial Intelligence A Modern Approach Third Edition*, Pearson. doi:10.1017/978026208900007724.
- Sampaio, T.P., Filho, V.J.M.F., De Sa Neto, A., 2009. An application of feed forward neural network as nonlinear proxies for the use during the history matching phase. In: Proceedings of the SPE Latin American and Caribbean Petroleum Engineering Conference Proceedings. <https://doi.org/10.2118/122148-ms>.
- Sayyafzadeh, M., 2015a. A self-adaptive surrogate-assisted evolutionary algorithm for well placement optimization problems. In: Proceedings of the Society of Petroleum Engineers - SPE/IATMI Asia Pacific Oil and Gas Conference and Exhibition. APOGEE. <https://doi.org/10.2118/176468-ms>, 2015.
- Sayyafzadeh, M., 2015b. History Matching by Online Metamodeling. In: Proceedings of the SPE Reservoir Characterisation and Simulation Conference and Exhibition. Society of Petroleum Engineers. <https://doi.org/10.2118/175618-MS>.

- Schulze-Riegert, R.W., Axmann, J.K., Haase, O., Rian, D.T., You, Y.L., 2002. Evolutionary algorithms applied to history matching of complex reservoirs. *SPE Reserv. Eval. Eng.* <https://doi.org/10.2118/77301-PA>.
- Shahkarami, A., Mohaghegh, S., 2020. Applications of smart proxies for subsurface modeling. *Pet. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(20\)60057-X](https://doi.org/10.1016/S1876-3804(20)60057-X).
- Shahkarami, A., Mohaghegh, S., Gholami, V., Haghigat, A., Moreno, D., 2014a. Modeling pressure and saturation distribution in a CO<sub>2</sub> storage project using a Surrogate Reservoir Model (SRM). *Greenh. Gases Sci. Technol.* <https://doi.org/10.1002/ghg.1414>.
- Shahkarami, A., Mohaghegh, S.D., Gholami, V., Haghigat, S.A., 2014b. Artificial intelligence (AI) assisted history matching. In: Proceedings of the Society of Petroleum Engineers - SPE Western North American and Rocky Mountain Joint Meeting. <https://doi.org/10.2118/169507-ms>.
- Shahkarami, A., Mohaghegh, S.D., Hajizadeh, Y., 2018. Assisted history matching using pattern recognition technology. *Int. J. Oil, Gas Coal Technol.* <https://doi.org/10.1504/IJOGCT.2018.090966>.
- Shams, M., El-Banbi, A., Sayyuh, H., 2019. A novel assisted history matching workflow and its application in a full field reservoir simulation model. *J. Pet. Sci. Technol.* **9**, 64–87.
- Shannon, C.E., 1948. A mathematical theory of communication. *Bell Syst. Tech. J.* **27** <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>.
- Siena, M., Guadagnini, A., Della Rossa, E., Lamberti, A., Masserano, F., Rotondi, M., 2016. A novel enhanced-oil-recovery screening approach based on Bayesian clustering and principal-component analysis. In: Proceedings of the SPE Reservoir Evaluation and Engineering. <https://doi.org/10.2118/174315-PA>.
- Sinha, S., de Lima, R.P., Lin, Y., Sun, A.Y., Symon, N., Pawar, R., Guthrie, G., 2020a. Leak detection in carbon sequestration projects using machine learning methods: cranfield site, Mississippi, USA. In: In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/201552-ms>.
- Sinha, U., Dindoruk, B., Soliman, M., 2020b. Prediction of CO<sub>2</sub> minimum miscibility pressure MMP using machine learning techniques. In: Proceedings of the SPE Symposium on Improved Oil Recovery. <https://doi.org/10.2118/200326-ms>.
- Sipocz, N., Tobiesen, F.A., Assadi, M., 2011. The use of artificial neural network models for CO<sub>2</sub> capture plants. *Appl. Energy*. <https://doi.org/10.1016/j.apenergy.2011.01.013>.
- Sobol', I.M., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Comput. Math. Phys.* [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9).
- Sun, J., Ma, X., Kazi, M., 2018. Comparison of decline curve analysis DCA with recursive neural networks RNN for production forecast of multiple wells. In: Proceedings of the SPE Western Regional Meeting Proceedings. <https://doi.org/10.2118/190104-ms>.
- Sun, Q., Ertekin, T., 2020. Screening and optimization of polymer flooding projects using artificial-neural-network (ANN) based proxies. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2019.106617>.
- Tatar, A., Shokrollahi, A., Mesbah, M., Rashid, S., Arabloo, M., Bahadori, A., 2013. Implementing radial basis function networks for modeling CO<sub>2</sub>-reservoir oil minimum miscibility pressure. *J. Nat. Gas Sci. Eng.* <https://doi.org/10.1016/j.jngse.2013.09.008>.
- Temizel, C., Aktas, S., Kirmaci, H., Susuz, O., Zhu, Y., Ranjith, R., Tahir, S., 2016. Turning data into knowledge: data-driven surveillance and optimization in mature fields. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/181881-ms>.
- Thanh, H.V., Binh, D., Van, Kantoush, S.A., Nourani, V., Saber, M., Lee, K.-K., Sumi, T., 2022. Reconstructing daily discharge in a megadelta using machine learning techniques. *Water Resour. Res.* **58** <https://doi.org/10.1029/2021WR031048> e2021WR031048.
- Tian, C., Horne, R.N., 2017. Recurrent neural networks for permanent downhole gauge data analysis. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/187181-ms>.
- Tilahun, S.L., 2019. Balancing the degree of exploration and exploitation of swarm intelligence using parallel computing. *Int. J. Artif. Intell. Tools* **28**. <https://doi.org/10.1142/S0218213019500143>.
- Tillerson, R.W., 2008. In: Meeting global energy supply and demand challenges. 19th World Petroleum Congress.
- Tunkiel, A.T., Sui, D., Wiktorski, T., 2020. Data-driven sensitivity analysis of complex machine learning models: a case study of directional drilling. *J. Pet. Sci. Eng.* **195**, 107630 <https://doi.org/10.1016/j.petrol.2020.107630>.
- Urban-Rascon, E., Aguilera, R., 2020. Machine learning applied to SRV modeling, fracture characterization, well interference and production forecasting in low permeability reservoirs. In: Proceedings of the SPE Latin American and Caribbean Petroleum Engineering Conference Proceedings. <https://doi.org/10.2118/199082-ms>.
- Vahdanikia, N., Divandari, H., Hemmati-Sarapardeh, A., Nait Amar, M., Schaffie, M., Ranjbar, M., 2020. Integrating new emerging technologies for enhanced oil recovery: ultrasonic, microorganism, and emulsion. *J. Pet. Sci. Eng.* **192**, 107229 <https://doi.org/10.1016/j.petrol.2020.107229>.
- Viana, F.A.C., 2016. A Tutorial on Latin Hypercube Design of Experiments. *Qual. Reliab. Eng. Int.* **32**, 1975–1985. <https://doi.org/10.1002/qre.1924>.
- Vida, G., Shahab, M.D., Mohammad, M., 2019. Smart proxy modeling of SACROE CO<sub>2</sub>-EOR. *Fluids*. <https://doi.org/10.3390/fluids4020085>.
- Vo Thanh, H., Sugai, Y., Sasaki, K., 2020. Application of artificial neural network for predicting the performance of CO<sub>2</sub> enhanced oil recovery and storage in residual oil zones. *Sci. Rep.* <https://doi.org/10.1038/s41598-020-73931-2>.
- Wang, L., Li, Z.P., Adenutsi, C.D., Zhang, L., Lai, F.P., Wang, K.J., 2021. A novel multi-objective optimization method for well control parameters based on PSO-LSSVR proxy model and NSGA-II algorithm. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107694>.
- Wei, W., Rezaadeh, A., Wang, J., Gates, I.D., 2021. An analysis of toe-to-heel air injection for heavy oil production using machine learning. *J. Pet. Sci. Eng.* **197**, 108109 <https://doi.org/10.1016/j.petrol.2020.108109>.
- Wong, W.K., Ming, C.I., 2019. A review on metaheuristic algorithms: recent trends, benchmarking and applications. In: Proceedings of the 2019 7th International Conference on Smart Computing and Communications, ICSCC 2019. <https://doi.org/10.1109/ICSCC.2019.8843624>.
- Xiong, X., Lee, K.J., 2020. Data-driven modeling to optimize the injection well placement for waterflooding in heterogeneous reservoirs applying artificial neural networks and reducing observation cost. *Energy Explor. Exploit.* <https://doi.org/10.1177/0144598720927470>.
- Xu, K., Li, J., Zhang, M., Du, S.S., Kawarabayashi, K.I., Jegelka, S., 2020. How neural networks extrapolate: from feedforward to graph neural networks. arXiv.
- Xu, T., 1998. Coupled modeling of non-isothermal multiphase flow, solute transport and reactive chemistry in porous and fractured media: 2. Model Applications.
- Yang, L., Shami, A., 2020. On hyperparameter optimization of machine learning algorithms: theory and practice. *Neurocomputing* **415**, 295–316. <https://doi.org/10.1016/j.neucom.2020.07.061>.
- Yang, T., Arief, I.H., Niemann, M., Houbiers, M., Meisinger, K.K., Martins, A., Froelich, L., 2019. A machine learning approach to predict gas oil ratio based on advanced mud gas data. In: Proceedings of the Society of Petroleum Engineers - SPE Europe Featured at 81st EAGE Conference and Exhibition 2019. <https://doi.org/10.2118/195459-ms>.
- Yang, X.S., Chien, S.F., Ting, T.O., 2014. Computational intelligence and metaheuristic algorithms with applications. *Sci. World J.* <https://doi.org/10.1155/2014/425853>.
- Yazdanpanah, A., Hashemi, A., 2012. Production optimization using an experimental design and genetic algorithm. *J. Am. Chem. Soc.* **8**.
- Yeten, B., Castellini, A., Guyaguler, B., Chen, W.H., 2005. A comparison study on experimental design and response surface methodologies. In: Proceedings of the SPE Reservoir Simulation Symposium. Society of Petroleum Engineers. <https://doi.org/10.2118/93347-MS>.
- You, J., Ampomah, W., Kutsienyo, E.J., Sun, Q., Balch, R.S., Aggrey, W.N., Cather, M., 2019a. Assessment of enhanced oil recovery and CO<sub>2</sub> storage capacity using machine learning and optimization framework. In: Proceedings of the Society of Petroleum Engineers - SPE Europe Featured at 81st EAGE Conference and Exhibition 2019. <https://doi.org/10.2118/195490-ms>.
- You, J., Ampomah, W., Sun, Q., 2020a. Development and application of a machine learning based multi-objective optimization workflow for CO<sub>2</sub>-EOR projects. *Fuel* **264**, 116758.
- You, J., Ampomah, W., Sun, Q., 2020b. Co-optimizing water-alternating-carbon dioxide injection projects using a machine learning assisted computational framework. *Appl. Energy*. <https://doi.org/10.1016/j.apenergy.2020.115695>.
- You, J., Ampomah, W., Sun, Q., Kutsienyo, E.J., Balch, R.S., Cather, M., 2019b. Multi-objective optimization of CO<sub>2</sub> enhanced oil recovery projects using a hybrid artificial intelligence approach. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/196182-ms>.
- You, J., Ampomah, W., Sun, Q., Kutsienyo, E.J., Balch, R.S., Dai, Z., Cather, M., Zhang, X., 2020c. Machine learning based co-optimization of carbon dioxide sequestration and oil recovery in CO<sub>2</sub>-EOR project. *J. Clean. Prod.* <https://doi.org/10.1016/j.jclepro.2020.120866>.
- Yousef, A.M., Kavousi, G.P., Alnuaimi, M., Alatrach, Y., 2020. Predictive data analytics application for enhanced oil recovery in a mature field in the Middle East. *Pet. Explor. Dev.* **47** [https://doi.org/10.1016/S1876-3804\(20\)60056-8](https://doi.org/10.1016/S1876-3804(20)60056-8).
- Zarefi, F., Daliri, A., Alizadeh, N., 2008. The use of Neuro-Fuzzy proxy in well placement optimization. In: Proceedings of the Intelligent Energy Conference and Exhibition. Society of Petroleum Engineers. <https://doi.org/10.2118/112214-MS>.
- Zerfat, M.M., Ayatollahi, S., Mehranbod, N., Barzegari, D., 2011. Bayesian network analysis as a tool for efficient EOR screening. In: Proceedings of the Society of Petroleum Engineers - SPE Enhanced Oil Recovery Conference 2011. EORC. <https://doi.org/10.2118/143282-ms>, 2011.
- Zhong, Z., Sun, A.Y., Wang, Y., Ren, B., 2020. Predicting field production rates for waterflooding using a machine learning-based proxy model. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107574>.
- Zubarev, D.I., 2009. Pros and cons of applying proxy-models as a substitute for full reservoir simulations. In: Proceedings of the SPE Annual Technical Conference and Exhibition. <https://doi.org/10.2118/124815-ms>.



## **Paper 2**

### ***Smart Proxy Modeling of a Fractured Reservoir Model for Production Optimization: Implementation of Metaheuristic Algorithm and Probabilistic Application***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar, Ole Torsæter



Original Paper

# Smart Proxy Modeling of a Fractured Reservoir Model for Production Optimization: Implementation of Metaheuristic Algorithm and Probabilistic Application

Cuthbert Shang Wui Ng,<sup>1,3</sup> Ashkan Jahanbani Ghahfarokhi,<sup>1</sup> Menad Nait Amar,<sup>2</sup> and Ole Torsæter<sup>1</sup>

Received 23 August 2020; accepted 13 February 2021  
Published online: 8 March 2021

Numerical reservoir simulation has been recognized as one of the most frequently used aids in reservoir management. Despite having high calculability performance, it presents an acute shortcoming, namely the long computational time induced by the complexities of reservoir models. This situation applies aptly in the modeling of fractured reservoirs because these reservoirs are strongly heterogeneous. Therefore, the domains of artificial intelligence and machine learning (ML) were used to alleviate this computational challenge by creating a new class of reservoir modeling, namely smart proxy modeling (SPM). SPM is a ML approach that requires a spatio-temporal database extracted from the numerical simulation to be built. In this study, we demonstrate the procedures of SPM based on a synthetic fractured reservoir model, which is a representation of dual-porosity dual-permeability model. The applied ML technique for SPM is artificial neural network. We then present the application of the smart proxies in production optimization to illustrate its practicality. Apart from applying the backpropagation algorithms, we implemented particle swarm optimization (PSO), which is one of the metaheuristic algorithms, to build the SPM. We also propose an additional procedure in SPM by integrating the probabilistic application to examine the overall performance of the smart proxies. In this work, we inferred that the PSO had a higher chance to improve the reliability of smart proxies with excellent training results and predictive performance compared with the considered backpropagation approaches.

**KEY WORDS:** Reservoir simulation, Dual-porosity dual-permeability, Smart proxy modeling, Back-propagation algorithms, Particle swarm optimization.

## INTRODUCTION

Hydrocarbons are among the primary sources of energy in today's world. This is proven by a statistical review conducted by British Petroleum

(2020), which found that, in 2019, oil contributed to the largest share of the world primary energy of about 33.1%, whereas natural gas had the third largest share of 24.2%. Hence, they play a pivotal role in quenching the high demand of world energy consumption and such demand will be likely in an upward trend due to the increasing global population (Gerald et al. 2014; International Energy Agency 2018). In addition, the importance of hydrocarbons is reflected by the significant influence of their price on many other major economic do-

<sup>1</sup>Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway.

<sup>2</sup>Département Etudes Thermodynamiques, Division Laboratoires, Sonatrach, Boumerdes, Algeria.

<sup>3</sup>To whom correspondence should be addressed; e-mail: cuthbert.s.w.ng@ntnu.no

mains (Lescaroux and Mignon 2009). This is illustrated clearly by the phenomenon of how many other industries have been affected by the fluctuation of oil price (Lescaroux and Mignon 2009). Therefore, it is essential to have a sustainable hydrocarbon production not only to fulfill the demand for energy consumption, but also to maintain the global economic growth. With respect to this, carbonate reservoirs are one of the main sources of hydrocarbons. These reservoirs make up approximately 60% of the global oil reserves and about 40% of the global gas reserves (Schlumberger 2020b). Most of these reservoirs are naturally fractured, and hence, accurate modeling of fluid flow in these reservoirs is one of the most critical steps to ensure the sustainable production of hydrocarbons.

In general, modeling of fluid flow in porous media can be perceived as a numerical reservoir simulation. Reservoir simulation is one of the most frequently used tools in reservoir management, which is the application of technological, labor, and financial resources to maximize the economic performance and the hydrocarbon recovery of a reservoir (Wiggins and Startzman 1990). This is because it has been implemented extensively to help predict the performance of a reservoir as well as to provide useful information for uncertainty analysis or any optimization task that includes enhanced oil recovery, hydraulic fracturing, and so forth. However, one of the challenges of accurate modeling of fractured reservoirs stems from a lack of underlying theory or principle to describe the behavior of fluid flow in these reservoirs. To mitigate this challenge, Barenblatt (1960) established a theory pertaining to fluid flow in fractured porous media. Based on this theory, Warren and Root (1963) developed the dual-porosity method, which has been one of the most fundamental tools in simulating a fractured reservoir. However, this conventional model does not sufficiently capture the realistic behavior of fluid flow as fluid is assumed to move only through fractures, whereas the matrix blocks only supply fluid to fractures. Hence, this model was enhanced to the dual-porosity dual-permeability (DPDP) model, in which the transport of fluid between matrix blocks is considered (Uleberg and Kleppe 1996). The details regarding this model are explained further below.

Having developed the DPDP model implies that fractured reservoirs can be simulated numerically. Nonetheless, another challenge in terms of

computational effort arises as the complexity of simulated fractured reservoirs increases (including as much details as possible to “describe realistically” a reservoir). Therefore, reservoir management might not be sufficiently efficient to keep up with sustainable hydrocarbon production. Fortunately, in today’s world of digitalization, methods of artificial intelligence and machine learning (AI&ML) have come to the rescue. In this context, Ertekin and Sun (2019) provided a very comprehensive review on the implementation of AI&ML methods in the field of reservoir engineering. They also proposed the use of hand-shaking protocol that would combine the advantages of both traditional and intelligent reservoir modeling to develop more powerful computational protocols. With this, the great potential and extensive utilization of AI&ML-based methods have also been demonstrated further in many technical domains of the petroleum industry (Mohaghegh 2000a, b, c; Parada and Ertekin 2012; Nait Amar and Jahanbani Ghahfarokhi 2020; Nait Amar et al. 2020). Moreover, with the help of AI&ML, Mohaghegh (2011) has coined a new class of reservoir modeling, namely smart proxy modeling (SPM). Fundamentally, SPM is the development of an artificial neural network (ANN) that receives both input and output data from a reservoir simulation model and undergoes a training phase. After the ANN has been trained to recognize the pattern induced by the data (relationship between input and output), it can yield the estimated result that matches with that produced by the reservoir model within a few seconds or minutes. Therefore, this ANN is termed “smart proxy.” Regarding this, the word “smart” reveals the ability to learn and capture the underlying physical behavior of a simulated reservoir model through pattern recognition and the word “proxy” denotes to act on behalf of the original model (Mohaghegh 2017, 2018).

For the past decade, SPM has been considered as a technological breakthrough in the petroleum industry as it has not only reduced the reservoir simulation time significantly, but it also provided the results within an acceptable range of accuracy. The successful application of smart proxies has been demonstrated in many literatures of the oil and gas industry. Mohaghegh et al. (2006) developed surrogate reservoir model (the initial nomenclature of SPM), which was an accurate representation of a sophisticated full-field reservoir model, and used it

for uncertainty analysis. With this breakthrough, these surrogate models were implemented on different real fields in Saudi Arabia for geological uncertainty analysis (Mohaghegh et al. 2012a, c). Mohaghegh et al. (2012b, 2015) then reformulated the concept of SPM by categorizing it as grid-based and well-based. As the nomenclatures imply, grid-based SPM is done for the analysis of numerical model at grid block level, whereas well-based SPM is for the analysis at well level. Grid-based SPM has been applied in several real-life CO<sub>2</sub> sequestration projects (Mohaghegh et al. 2012b), whereas well-based SPM has been implemented for optimization of production scheduling of a real field in United Arab Emirates (Mohaghegh et al. 2015). Besides, the application of SPM was then extended gradually to other domains, such as history matching and enhanced oil recovery (EOR). He et al. (2016) coupled the use of SPM with differential evolution (DE) to perform automatic history matching. Alenezi and Mohaghegh (2016) also built a SPM that reproduced and forecasted the dynamic properties of a reservoir that has been water-flooded. Moreover, Mohaghegh (2018) discussed the utilization of SPM under the context of CO<sub>2</sub>-EOR as a storage mechanism. Furthermore, Parada and Ertekin (2012) applied SPM to establish successfully a new screening tool for four different improved oil recovery (IOR) methods, including waterflooding, miscible injection of CO<sub>2</sub> and N<sub>2</sub>, and injection of steam. Therefore, these literatures do not only show the high applicability of SPM in oil and gas industry, but they also highlight its potential for further enhancement.

Nevertheless, SPM still has few disadvantages. One of them is that a smart proxy built can only be applied to predict what the simulated reservoir might estimate only if the physics assumed in the numerical simulation is not changed. For instance, if a smart proxy is developed on a reservoir model with reservoir pressure of 4000 psia,<sup>1</sup> then it cannot be applied to perform any estimation of parameters when the reservoir pressure is not 4000 psia. To handle this problem, another smart proxy needs to be established. In addition to this, the spatio-temporal database is considered as the backbone of the SPM as it is the main component used to train the ANN model. Thus, if another smart proxy is built (as

previously mentioned), then the database needs to be prepared again. Despite having such inconvenience, the time spent on preparation of this database is still much less than the time spent by numerical simulation. Pertaining to this, the preparation of a spatio-temporal database might take about few hours (or for few minutes with the help of commercial software). However, for a sophisticated reservoir simulation model, the computation might take a few days. It is important to understand that smart proxy is another example of data-driven models as it is developed by analyzing the collected data (Alenezi and Mohaghegh 2016, 2017). Hence, careful attention is required when a spatio-temporal database is created. If incorrect data are provided to the smart proxy, it will “learn wrongly” and produce unsatisfactory results. This complies with the short phrase that goes “garbage in and garbage out.”

Although there are many literatures explaining the theoretical basis of SPM, it is still treated as “black-box” as commercial software is mostly used to build a smart proxy. Thus, in this work, one of the objectives was to provide a more vivid illustration of how SPM can be performed based on a synthetic reservoir model. Besides, we present another alternative of training algorithm apart from the backpropagation algorithm that is mostly used in SPM. More intriguingly, we include a probabilistic application to evaluate further the overall performance of the developed SPMs. We opine that this integration in SPM is insightful as it helps to better reflect the performance of the proxy models. After this introduction, we discuss briefly the mathematical concepts of the DPDP model and how ANN operates. Three different algorithms, which are two examples of backpropagation algorithms, namely stochastic gradient descent (SGD) and adaptive moment estimation (Adam) algorithms, and particle swarm optimization (PSO), were implemented as the learning algorithm to train the ANN. Hence, the fundamentals of these algorithms are discussed next. Then, we explicate the background of the reservoir model simulated based on the DPDP method and the problem setting of the production optimization case. We also explain how the respective SPM is developed upon it and used in production optimization. Then, the results and discussion will follow. Prior to proceeding to conclusions, we also provide another case study, which considers a heterogeneous fractured reservoir model, to further show the robustness of the methodology discussed in this paper.

<sup>1</sup> 1 psia = 6894.75728 Pa.

## METHODOLOGY

### Fundamentals of DPDP Model

In the conventional dual-porosity model, a grid block consists of two portions—the matrix block and the fractures. In this model, the fluid flows mainly through the fractures, whereas the matrix blocks only provide fluids to the fracture (Uleberg and Kleppe 1996). This phenomenon of fluid flow is illustrated in a two-dimensional case as in Figure 1.

Assuming a one-dimensional and one-phase flow case, the transport of fluid through the fracture can be mathematically expressed as (Barrenblatt 1960; Warren and Root 1963):

$$\frac{\partial}{\partial x} \left( \frac{k}{\mu B} \frac{\partial P}{\partial x} \right)_{\text{fracture}} + \hat{q}_{\text{matrix\_fracture}} = \frac{\partial}{\partial t} \left( \frac{\phi}{B} \right)_{\text{fracture}} \quad (1)$$

where  $k$  is permeability,  $B$  is the formation volume factor,  $\mu$  is viscosity of fluid, and  $\phi$  is porosity. The term  $\hat{q}_{\text{matrix\_fracture}}$  indicates the supply of fluid to fractures by the matrix block, and its mathematical expression is:

$$-\hat{q}_{\text{matrix\_fracture}} = \frac{\partial}{\partial t} \left( \frac{\phi}{B} \right)_{\text{matrix}} \quad (2)$$

Because the assumption of no fluid flow between the blocks of matrix is not realistic, the dual-porosity model was extended to the DPDP model by adding a flow term in Eq. (2) (Uleberg and Kleppe

1996). Hence, the system of equations representing the DPDP model is:

$$\frac{\partial}{\partial x} \left( \frac{k}{\mu B} \frac{\partial P}{\partial x} \right)_{\text{fracture}} + \hat{q}_{\text{matrix\_fracture}} = \frac{\partial}{\partial t} \left( \frac{\phi}{B} \right)_{\text{fracture}} \quad (3)$$

$$\frac{\partial}{\partial x} \left( \frac{k}{\mu B} \frac{\partial P}{\partial x} \right)_{\text{matrix}} - \hat{q}_{\text{matrix\_fracture}} = \frac{\partial}{\partial t} \left( \frac{\phi}{B} \right)_{\text{matrix}} \quad (4)$$

Regarding the exchange term, it can be further represented as:

$$-\hat{q}_{\text{matrix\_fracture}} = \sigma \frac{k_{\text{matrix}}}{\mu} (P_{\text{matrix}} - P_{\text{fracture}}) \quad (5)$$

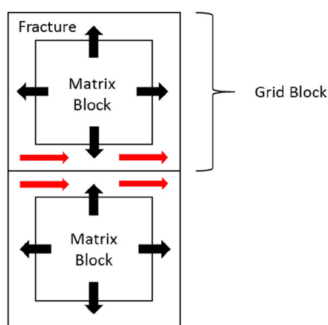
where  $P$  denotes pressure, whereas  $\sigma$  is the shape factor or the geometric factor. This shape factor represents the geometry of the matrix block, and it dictates the flow fluid between the matrix blocks and the fracture system (Kazemi et al. 1976). There are many mathematical formulations available in the literature to describe this shape factor depending upon the physical effects and mechanisms considered (Warren and Root 1963; Ahmad and Olivier 2008; Su et al. 2013). In this context, one of the most widely applied forms is the one proposed by Kazemi et al. (1976), and it was used in this study. Regarding its formulation, Kazemi et al. (1976) discussed that the shape factor can be computed in a three-dimensional case as:

$$\sigma = 4 \times \left[ \frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2} \right] \quad (6)$$

where the  $L$  term refers to the dimension of the matrix block in  $x$ -,  $y$ -, and  $z$ - directions.

### ANN

ANN is a biologically inspired mathematical model or algorithm that can predict any relevant output within an acceptable range of accuracy after learning the relationship between the inputs and outputs provided (Wilamowski and Irwin 2011; Buduma and Locasio 2017). This biological inspiration stems from the imitation of learning method used in human brains. ANN is very robust due to its high generalization ability in capturing the nonlinearity of any process investigated (Gharbi and Mansoori



**Figure 1.** Fluid flow behavior in a dual-porosity model for two-dimensional case. The red arrows indicate the flow thorough fracture network, whereas the black arrows denote the supply of fluid from matrix.

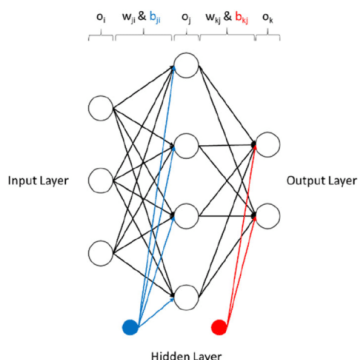


Figure 2. Structure of an ANN.

2005; Wilamowski and Irwin 2011; Nait Amar et al. 2018b). Thus, ANN is better than any traditional regression approach to solve complicated mathematical problems (Gharbi and Mansoori 2005). There are different types of ANN, such as feed-forward neural network, convolutional neural network (CNN), recurrent neural network (RNN). Multilayer perceptron (MLP), which is an example of feed-forward neural network,<sup>2</sup> was implemented here. Regarding the architecture of MLP, it is made up of three different types of layers, namely one input layer, one or more hidden layers, and one output layer (Wilamowski and Irwin 2011; Buduma and Locasio 2017). Each of these layers comprises simple calculating elements, which are known as nodes, units, or artificial neurons (Gharbi and Mansoori 2005). The output from each node in a layer is multiplied by the weights (and biases). The product enters the node in the next layer as input. These inputs are then summed and applied to activation function, also known as transfer function, to produce the output of the node. The structure or topology of an arbitrary ANN that comprises one input layer with three nodes, one hidden layer with four nodes, and one output layer with two nodes is shown in Figure 2.

<sup>2</sup> To avoid confusion, feed-forward neural network, artificial neural network, multilayer perceptron, smart proxy model, smart proxy, and proxy model technically share the same definition in this paper. However, feed-forward neural network is considered as a family of artificial neural network and it includes several types such as multilayer perceptron, radial basis function network, correlation filter neural network.

Referring to Figure 2, the mechanism of ANN can be expounded mathematically as follows. From input layer to hidden layer, the output of the node is computed as:

$$o_j = F\left(\sum_{i=1}^{N_i} w_{ji}o_i + b_{ji}\right) \quad (7)$$

Then, from hidden layer to output layer, the output of the node is calculated as:

$$o_k = F\left(\sum_{j=1}^{N_j} w_{kj}o_j + b_{kj}\right) \quad (8)$$

In Eqs. (7) and (8), the subscript  $i$  denotes the input layer, the subscript  $j$  means the hidden layer, and the subscript  $k$  indicates the output layer,  $N$  shows the number of nodes in each layer,  $o$  indicates either the output of node in the current layer or the input of node from previous layer (based upon the subscript),  $w$  is a set of weights, and  $b$  is a set of biases. Weights are considered as the fitting parameters in modeling of an ANN, whereas bias is an extra node that provides more flexibility for the ANN model to be trained. There are many forms of activation functions  $F$  that are readily used in ANN modeling. The major ones include sigmoid, rectified linear unit (ReLU), and hyperbolic tangent (Buduma and Locasio 2017). Here, the activation function used was ReLU and it is represented as:

$$F(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases} \quad (9)$$

The derivative of the ReLU function is:

$$\frac{\partial F(x)}{\partial x} = \begin{cases} 0 & \text{for } x \leq 0 \\ 1 & \text{for } x > 0 \end{cases} \quad (10)$$

Mathematically, ANN learns the relationship or recognizes the pattern between input and output data through the tuning of the sets of weights and biases between the two layers. Through a number of epochs (or iterations), these weights and biases are optimized by minimizing any predefined error function (also known as loss or cost function), such as mean squared error, average absolute percentage error. There are different examples of algorithms that can be used to optimize these weights and biases. Backpropagation algorithm has been commonly used in this context. Examples of backpropagation algorithm are gradient descent (GD), Gauss–Newton algorithm, Levenberg–Marquardt algorithm (LM), adaptive gradient algorithm (AdaGrad), root-

mean-square propagation (RMSProp), Adam, and so forth. Additionally, other metaheuristic algorithms, like PSO, DE, genetic algorithm (GA), and so forth, have also been proven useful for neural network training (Nait Amar et al. 2018a, b). As Bianchi et al. (2009) have counseled, metaheuristic algorithm is a high-level mathematical algorithm that is generally natural inspired and used to solve more sophisticated optimization problems. In this study, both backpropagation algorithm and metaheuristic approach have been employed to enable the ANN to “learn.” The selected backpropagation algorithm was GD, whereas PSO was the chosen metaheuristic training algorithm.

### Backpropagation Algorithm

For the workflow of the GD algorithm, both the inputs and outputs are fed to the ANN as the training phase starts. When the inputs enter the ANN and proceed through the layers, they are gradually processed to yield the predicted output. Thereafter, the predicted output is compared with the desired output. Errors are then propagated back through the ANN. During this backpropagation, the weights and biases are adjusted to minimize the errors. Such process is repeated iteratively until either the errors become less than the predefined tolerance or the number of iterations is reached. The GD is an algorithm that applies the first-order derivative for computation. In this context, the first-order derivative of the error function is implemented to determine the minimum in the error space. The calculation of gradient at iteration  $t$  can be expressed mathematically as:

$$g_t = \frac{\partial E(x, w_t)}{\partial w_t} = \left[ \frac{\partial E}{\partial w_{1,t}} \frac{\partial E}{\partial w_{2,t}} \frac{\partial E}{\partial w_{3,t}} \dots \frac{\partial E}{\partial w_{N,t}} \right]^T \quad (11)$$

where  $E$  indicates the error function,  $\mathbf{x}$  the input vector, and  $\mathbf{w}$  the weight (and bias) vector. Thereafter, the weights are updated by using the following equations. The same idea applies to the updating of the biases.

$$w_{t+1} = w_t + \Delta w_t \quad (12)$$

$$w_{t+1} = w_t - (\gamma \times g_t) \quad (13)$$

In Eqs. (12) and (13), the weights (and biases) at iteration  $t + 1$  are updated using the weights (and biases) at iteration  $t$ , the gradient at  $t$ , and  $\gamma$ , which is

the learning rate or step size. Therefore, the gradient is always computed at every iteration step to adjust the weights (and biases). Pertaining to the computation of gradient of error function, it is highly dependent on the forms of error function and activation function that were used. Here, the error function used was the mean squared error, whereas the activation function used was ReLU.

The mathematical formulation of the application of GD as learning algorithm is as follows. For the following derivation, the meaning of the subscripts used here is the same as explained above. The term  $t$  means the target value or the actual output,  $P$ , denotes the total number of training sets provided; thus:

$$E(x, w, b) = \frac{1}{P} \sum_{k=1}^P (t_k - o_k)^2 \quad (14)$$

Having defined the error function, the backpropagation algorithm starts by computing the weight update between the hidden and output layers. To perform this computation, the gradient of the error function with respect to the weights between the hidden and output layers is determined. Thereafter, the similar procedure is conducted to calculate the weight update between the hidden and input layer. This algorithm carries on iteratively until the value of error function (obtained by using the updated weights and biases) is less than a predefined tolerance or the initialized number of epochs is reached. For a more substantial understanding of the mathematical formulation of the backpropagation algorithm, peruse Wilamowski and Irwin (2011) and the relevant literatures. Here, the Keras module, which was developed by Chollet (2019), had been implemented with the help of the programming language Python 3.8.1 and TensorFlow 2.1.0 to use the GD algorithm to optimize the weights and biases. However, it is essential to note that in Keras module, instead of using GD algorithm, the stochastic gradient descent (SGD) algorithm is applied. The fundamentals of these two algorithms are the same. The main difference is that, in SGD, the gradient is only computed once at each iteration step (by randomly selecting a sample from the training set) and is used further (Buduma and Locasio 2017). By inducing this stochastic behavior, the computational cost is reduced drastically. Apart from SGD, Adam was another backpropagation algorithm used here; it is a more advanced and robust variant of SGD developed by Kingma and Ba (2015). Mathe-

matically, it approximates the first and second moments of gradients to adaptively calculate the learning rates for different parameters (Kingma and Ba 2015). Refer to Kingma and Ba (2015) for the details of Adam. Here, Adam was also implemented using Python 3.8.1 and TensorFlow 2.1.0.

## PSO

PSO was introduced by Kennedy et al. (1995) based upon the simulation of the social behavior of a flock of flying birds. As explained in several literatures (Kennedy et al. 1995; Shi and Eberhart 1999; Nait Amar et al. 2018a), mathematically, this algorithm operates by having a population of particles, which is also known as a swarm of particles. Each of these particles corresponds to a potential position or a solution in a search space. Then, the position of each particle is updated iteratively according to its position and velocity at previous timestep. The movements of the particles in the search space are controlled by their own best-known position (the local best position) and their best-known position in the entire swarm (the global best position). As this process occurs iteratively, the particles in the swarm will eventually converge to an optimal point, which is deemed as the best solution in the search space. The position and velocity for the  $j^{\text{th}}$  particle in a  $N$ -dimensional space at iteration  $t$  can be expressed, respectively, as:

$$x_{j,t} = \{x_{j1,t}, x_{j2,t}, x_{j3,t}, \dots, x_{jN,t}\} \quad (15)$$

$$v_{j,t} = \{v_{j1,t}, v_{j2,t}, v_{j3,t}, \dots, v_{jN,t}\} \quad (16)$$

Then, the velocity of each particle at next iteration  $t + 1$  is updated as (Shi and Eberhart 1999):

$$v_{jN,t+1} = v_{jN,t} + c_1 r_1 (pbest_{jN,t} - x_{jN,t}) + c_2 r_2 (gbest_{N,t} - x_{jN,t}) \quad (17)$$

In Eqs. (15), (16), and (17),  $v_{jN,t}$  and  $x_{jN,t}$  represent the velocity of the  $j^{\text{th}}$  particle at iteration  $t$  and its corresponding position in  $N$ -dimension quantity, respectively;  $pbest_{jN,t}$  corresponds to the  $N$ -dimension quantity of the individual  $j$  at the best position or the local best position at iteration  $t$ ;  $gbest_{N,t}$  is the  $N$ -dimension quantity of the swarm at the best position or the global best position at iteration  $t$ ;  $c_1$  denotes the cognitive learning factor (also known as cognitive weight), whereas  $c_2$  means the

social learning factor (also known as social weight);  $r_1$  and  $r_2$  are random numbers extracted between 0 and 1. Upon updating the velocity, each particle moves to a new potential solution as:

$$x_{jN,t+1} = x_{jN,t} + v_{jN,t+1} \quad (18)$$

A new parameter, inertial weight  $\omega$  introduced by Shi and Eberhart (1998), was included in Eq. (17) to improve the convergence condition. This also gradually decreases the velocity of the particles to have the swarm of particles under control (Nait Amar et al. 2018a). In other words, it plays a part in balancing the global search also known as exploration, and the local search also termed as exploitation (Shi and Eberhart 1998; Zhang et al. 2015):

$$v_{jN,t+1} = \omega v_{jN,t} + c_1 r_1 (pbest_{jN,t} - x_{jN,t}) + c_2 r_2 (gbest_{N,t} - x_{jN,t}). \quad (20)$$

In the context of the minimization problem, an objective function  $f$  to be minimized is defined. Then, to determine the local best solution at iteration  $t + 1$ , the following formula is given (Nait Amar et al. 2018a):

$$pbest_{jN,t+1} = \begin{cases} pbest_{jN,t}, & \text{iff } (pbest_{jN,t}) = f(x_{jN,t+1}) \\ x_{jN,t+1}, & \text{otherwise} \end{cases} \quad (21)$$

Then, to find the global best solution at iteration  $t + 1$ , the following mathematical formulation is presented:

$$gbest_{jN,t+1} = \min [f(pbest_{jN,t+1})] \quad (22)$$

In this study, the objective function was the error function in the ANN modeling. To apply PSO as the training algorithm of ANN, this can be simply done by treating the weights and biases as the particles in the algorithm. Hence, the total number of particles in a swarm is the total number of weights and biases. Then, the optimization can be performed using the abovementioned formulations. Here, the package of PySwarms version 1.1.0, which was built by Miranda (2019), was implemented by using the programming language Python 3.8.1 to perform the optimization. In comparison with the SGD algorithm, one of the advantages of PSO is that it is a derivative-free algorithm. This implies that it is more robust as it can be utilized to optimize a mathematical function that is not easily differentiable.



**Table 1.** Essential parameters used to develop the DPDP reservoir model

Parameters		Values		Units	
Initial reservoir pressure		$3.47 \times 10^7$		Pa	
Oil density		819.18		kg/m <sup>3</sup>	
Water density		1041.20		kg/m <sup>3</sup>	
Oil viscosity		0.0035		Pa s	
Water viscosity		0.0005		Pa s	
Initial water saturation		Matrix media		Fracture media	
Layer 1		0.1922		0.000	
Layer 2		0.1924		0.000	
Layer 3		0.1926		0.000	
Layer	Matrix block height (m)	Matrix permeability (m <sup>2</sup> )	Effective fracture permeability (m <sup>2</sup> )	Porosity	
				Matrix media	Fracture media
1	9.144	$9.869 \times 10^{-15}$	$1.480 \times 10^{-12}$	0.210	0.0015
2	6.096	$1.974 \times 10^{-14}$	$1.974 \times 10^{-12}$	0.230	0.0020
3	12.192	$1.480 \times 10^{-14}$	$2.467 \times 10^{-12}$	0.250	0.0018

## NUMERICAL SIMULATION MODEL

A three-dimensional, two-phase (black oil and water) reservoir simulation model was built to represent the “true” reservoir model. The “true” reservoir is in fact inspired by the dual-porosity model discussed in Firoozabadi and Thomas (1990), which is a two-dimensional and three-phase model (black oil, water, and gas—including free and dissolved gas). However, most of the reservoir parameters and relevant fluid properties were changed to develop the “true” model. This “true” reservoir model supplied the necessary data for the development of the respective SPM. This reservoir was a DPDP model made up of three layers with uniform thickness.<sup>3</sup> The top of this reservoir was set at the depth of 305 m. About the geometry of this model, each grid block had a length of 25 m, a width of 25 m, and a height of 15.2 m. Thus, the dimension of the reservoir model was 1525 m × 1525 m × 45.7 m, which corresponds to the number of blocks being 61 × 61 × 3. Regarding the well configuration, it was the five-spot pattern in which four injectors were, respectively, set to penetrate near the corners of this reservoir model and a producer was placed in

the center of the reservoir. The injectors (producer) would inject water to (would produce from) all the fracture layers. Besides that, the performance of each well in this model was controlled by its respective rate. The target of the field production rate was set equal to the target of the field injection rate for pressure maintenance. For instance, if the target rate of the producer was 400 m<sup>3</sup>/day, then the target rate of each of the injector was 100 m<sup>3</sup>/day (totaling up to 400 m<sup>3</sup>/day of the target of the field injection rate). The numerical simulation of this DPDP reservoir model was conducted using ECLIPSE 100 software Schlumberger (2020a). Other details of this model are summarized in Table 1.

For further clarification, as presented in Table 1, the values of matrix block heights, matrix permeability, and effective fracture permeability were initialized for *x*-, *y*-, and *z*-directions. Additionally, the relative permeability curves and the oil–water capillary pressure curves for matrix media are illustrated in Figure 3. For the two-phase flow in fracture, the linear relationship between relative permeability and saturation, which is also known as “X-curve”, is one of the most fundamental models that was determined by Romm (1966). “X-curve” has been used in several fracture-related researches in petroleum industry (Van Golf-Racht 1982; Gilman and Kazemi 1983; Firoozabadi and Thomas 1990). Besides that, regarding the oil–water capillary pressure in the fracture system, it is equal to zero as

<sup>3</sup> In the modeling of DPDP, if three layers are defined, then there will be six resultant layers in which three of them correspond to the matrix system and the remaining three layers correspond to the fracture system. These fluid flow mechanisms of these two systems are represented by extending Eqs. (3), (4), and (5) to three-dimensional and two-phase case.

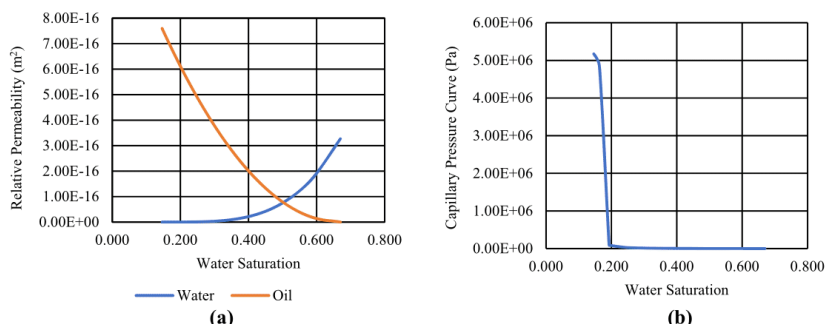


Figure 3. (a) Relative permeability curve. (b) Oil–water capillary pressure curve for the matrix media.

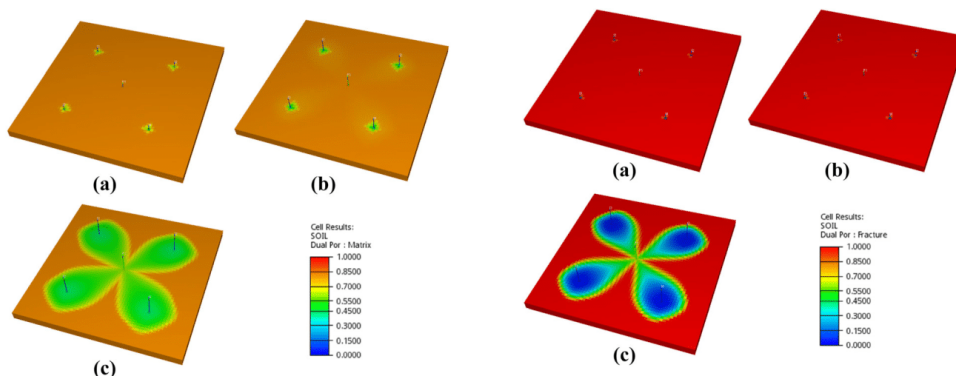


Figure 4. Overview of the matrix system of the reservoir model: (a) Layer 1; (b) Layer 2; (c) Layer 3.

Figure 5. Overview of the fracture system of the reservoir model: (a) Layer 1; (b) Layer 2; (c) Layer 3.

shown in the model discussed by Firoozabadi and Thomas (1990). In short, we selected these models of relative permeability curve and oil–water capillary pressure in both matrix and fracture systems for illustrative purpose. By using the software ResInsight developed by Ceetron Solution AS (2020), this reservoir model depicting oil saturation at the water injection rate of 636 m<sup>3</sup>/day (after the injection period of 5 years) is displayed correspondingly in Figure 4 for the matrix system and in Figure 5 for the fracture system.

Based on Figures 4 and 5, much more oil had been swept toward the producers in Layer 3 for both matrix and fracture media. Because the injectors

were (the producer was) perforated in all the fracture layers, this denoted that the injected water flowed and swept the oil in (the oil was only produced from) the fracture systems. Given the homogeneity of every layer of the model and the high effective permeabilities in *z*-direction for all the fracture layers, the cross-flow of fluids between the fracture layers was prominent to contribute to the high sweeping of oil in Layer 3 of the fracture media. This scenario also occurred to the matrix media because it needed to supply the oil to the fracture system where most of the oil has been swept and produced. In this context, we reiterate that the DPDP reservoir modeling was not the main goal of this work. In fact, we intended to design a valid DPDP model to

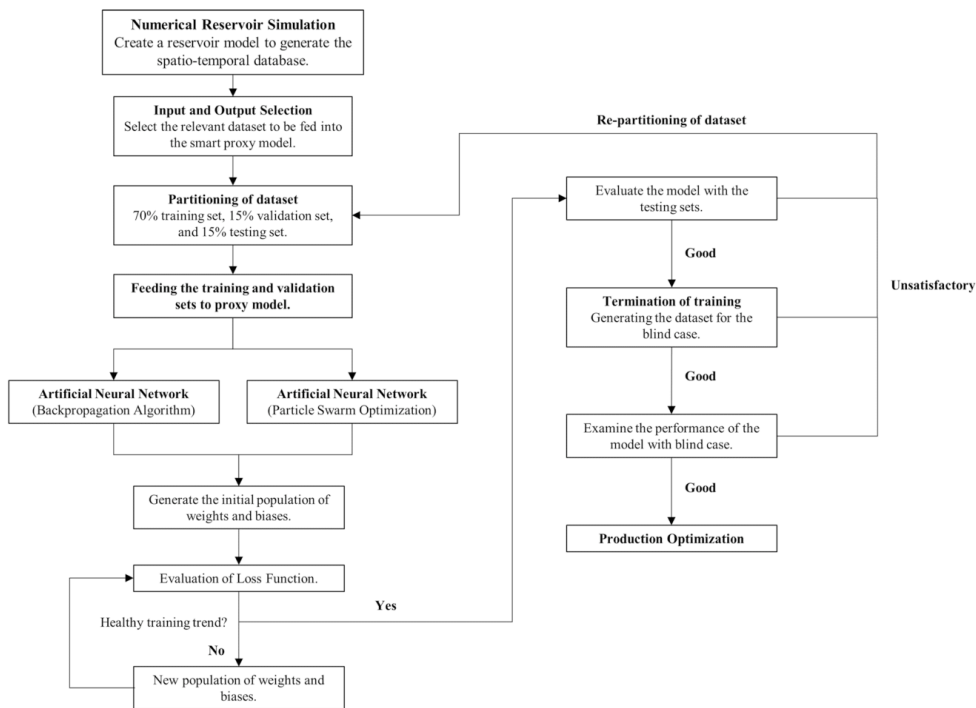


Figure 6. General workflow of SPM.

Table 2. Values of the economic parameters used in this example of production optimization

Parameters	Values	Units
Oil price, $P_o$	377.40	USD/m <sup>3</sup>
Cost of produced water, $P_w$	44.02	
Cost of injected water, $P_{inj}$	44.02	
Monthly discounted rate	0.833	%

demonstrate that our developed proxy model was functioning accurately.

**PRODUCTION OPTIMIZATION**

Smart proxy is widely developed in the petroleum industry because of its inexpensive computational effort. However, SPM is an objective-oriented task, which implies that modelers need to first know what the smart proxy is used for prior to developing

Table 3. Simulation scenarios executed for SPM

Scenario Index	Injection rate (m <sup>3</sup> /day)
1	636
2	676
3	715
4	755
5	795

it. After identifying the purposes or functions of the model, modelers would have a well-established understanding pertaining to the preparation of the spatio-temporal database (input and output data) used for neural network training. Regarding this, we used an illustrative example of production optimization as the objective of developing the smart proxy. For this illustrative example, we assumed the production lifetime of the reservoir model discussed

Table 4. Selected input and output data

Inputs		Output
Indexes	Simulation scenario	Scenarios 1, 3, and 5
Static inputs	Grid block $i$ th position	Well group (grid block $k$ th denotes the perforated $k$ th grid block)
	Grid block $j$ th position	
	Grid block $k$ th position	
	Porosity	Average values of layers with well perforation, layers of matrix media, layer of fracture media
Dynamic inputs	Permeability	
	Matrix block height	Matrix media (parameters in DPDP modeling)
	Shape factor	
	Time	Monthly basis (timestep 0 to timestep 360)
	Bottom-hole pressure	For 4 injectors and 1 producer at time $t$ at time $t$ at time $t - 1$
	Field water injection rate	
	Field oil production rate	Field oil production rate at time $t$

to be 30 years and the objective function to be the net present value (NPV). In this case, we needed to decide the target of the field injection rate that can maximize the NPV throughout the production lifetime. The NPV for this optimization example can be formulated as:

$$NPV = \sum_{k=0}^N \frac{P_o Q_{o,k} - P_w Q_{w,k} - P_{inj} Q_{inj,k}}{(1+r)^k} \quad (23)$$

where the subscripts  $o$ ,  $w$ , and  $inj$  denote oil, water (produced), and injected water, respectively;  $P$  is the price (or cost) per standard barrel (the corresponding unit is USD/m<sup>3</sup>),  $Q$  is total amount for a certain timestep (the respective unit is m<sup>3</sup>),  $r$  is the discount rate, and  $k$  is the timestep. To calculate  $Q$ , the following equation was used:

$$Q_{ic\{o,w,inj\},k} = q_{ic\{o,w,inj\},k} \times \Delta t_k \quad (24)$$

where  $q$  is the flow rate reported (either by the numerical simulation or the developed SPM) on monthly basis (the unit is m<sup>3</sup>/day) and  $\Delta t_k$  is the number of months for timestep  $k$ . Here, the smart proxy for the prediction of injection rates was not developed as the injection rates remained constant throughout the production period of the reservoir model. Hence, for practical purpose, only two SPMs were developed, which, respectively, predicted the oil production rates and the water production rates (both on monthly basis). With respect to this, it is possible to develop a SPM that predicts simultaneously two outputs, namely both oil and water rates. Nevertheless, the tuning of the weights and biases can be more challenging. Thus, for better and more fundamental demonstration of SPM, we decided not

to go with this option in this work. Upon formulating the objective function used in this example of production optimization, the setting of the economic parameters<sup>4</sup> used is presented in Table 2.

## SMART PROXY MODELING

To build a SPM, the first step is to generate the spatio-temporal database, which is used as the input and output data to train, validate, and test the model. This database is developed by retrieving the essential data from the numerical reservoir simulation. This step is very crucial because the data extracted will determine the usefulness of this proxy model. For this work, the input and output data selected from the “true” reservoir model are summarized in Table 4 (the details are explained further below). The database is considered as the backbone of SPM because it is the source of the data used to train the neural network.

### Data Preparation and Analysis

To generate data used for the neural network training, five different simulation scenarios, namely the target of the injection rates at 636 m<sup>3</sup>/day, 676 m<sup>3</sup>/day, 715 m<sup>3</sup>/day, 755 m<sup>3</sup>/day, and 795 m<sup>3</sup>/day, were run (the other parameters used in the numerical reservoir simulation were kept constant). More

<sup>4</sup> We understand that the economic parameters used here might not reflect the real-world case, but our goal here is to present the application of the smart proxy via an illustrative optimization task.

precisely, only three of them were used for the development of smart proxy, whereas the remaining two were used as the blind cases, which are discussed further below. Table 3 summarizes the five simulation scenarios, of which scenarios 1, 3, and 5 were used for SPM.

Upon running the simulations, the spatio-temporal database was readily generated. This database was developed by extracting the static and dynamic data from the numerical simulation. In this context, static data indicate that the data do not change with time (e.g., porosity, permeability), whereas dynamic data denote otherwise (e.g., instance, water injection rate, oil production rate). One of the main challenges of SPM is the humongous size of the spatio-temporal database. This occurs when the geological properties (static properties) of the simulated reservoir model are very heterogeneous (each of the grid blocks in the reservoir model has different values of porosity and permeability). The high geological heterogeneity will cause the SPM to be impractical if all these static data are used. To alleviate this problem, several literatures (Mohaghegh et al. 2012a, b, c, 2015; He et al. 2016; Alenezi and Mohaghegh 2016, 2017) recommend the application of tier system to delineate the reservoir model. In this aspect, the Voronoi graph theory was implemented to re-upscale these static properties through the lumping of the reservoir layers. By doing so, the size of the static inputs used in defining the structure of the spatio-temporal database can be decreased. However, here, despite having a total of 22,326 grid blocks in the reservoir model, it was not considered to be very complex because the porosity and permeability were homogenous per layer. Hence, the reservoir model can be simply delineated by categorizing it into the matrix media and fracture media.

After resolving the issue of reservoir complexity, the selection of input and output data needs to be considered. For a real-life reservoir model, the spatio-temporal database can still be gigantic to be entirely used as the input and output for SPM. To mitigate this challenge, the above-mentioned literatures propose to use the key performance indicator (KPI) coupled with fuzzy logic to help rank the degree of influence of different properties in the selection of input and output, and it is conducted mostly by using commercial software. In this study, for the purpose of illustration, the input and output data for SPM were determined based upon our knowledge of reservoir engineering. Thereafter, the input and output data yielded the final database

applied for training, validating, and testing the neural network as summarized in Table 4, which shows 54 static inputs and 8 dynamic inputs.

On the one hand, regarding static properties, the scenario index, which helps the neural network to identify which instance of the injection rates is used, was one of them. Besides this, the well locations make up 25 out of 54 static inputs because there were 5 wells in total and each of the locations was represented as  $i$ th,  $j$ th, and  $k$ th positions of the grid blocks (with all the fracture layers perforated). This corresponded to one group of the static inputs (Table 4). For both porosity and permeability, each of them comprised 11 static inputs, and 5 of them corresponded to the inputs of the average values of grid block where the wells were perforated and the remaining 6 corresponded to the inputs for the 3 layers in both matrix and fracture systems. Thereafter, the heights of the matrix blocks and the shape factors, respectively, contributed to 3 static inputs.

On the other hand, the bottom-hole pressures of all 5 wells contributed to 5 of the 8 dynamic inputs. Besides that, the timestep also acted as one of the dynamic inputs. The water injection rate at time  $t$  (on monthly basis) was also a dynamic input. The remaining dynamic input was the oil production rate at time  $t-1$  (on monthly basis), whereas the oil production rate at time  $t$  (on monthly basis) was used as the output data instead of being treated as input data in this neural network training. For the development of smart proxy for the prediction of water production rates, the input and output data were essentially the same. However, only the oil production rates at time  $t-1$  and  $t$  needed to be replaced with the water production rates at time  $t-1$  and  $t$ . Besides that, each of the simulation scenarios was run for 30 years. Since the oil production rates were reported on monthly basis, this corresponded to 360 months (30 years  $\times$  12 months/year). By starting from timestep = 0, there were a total of 361 timesteps for each scenario. This resulted in a total number of 68,229 records (3 scenarios  $\times$  361 timesteps/scenario  $\times$  63 records/timestep) in the database, which was to be fed into the neural network for training.

### Neural Network Training

Training the neural network is the most essential part of SPM. Prior to feeding the input and

**Table 5.** Ranges of values of training data

Parameters	Minimum value	Maximum value
Time (months)	0	360
Simulation scenario index	1	5
Well location (grid block position)	4	46
Porosity	0.0015	0.2500
Permeability (m <sup>2</sup> )	$9.869 \times 10^{-15}$	$2.467 \times 10^{-12}$
Matrix block height (m)	6.096	12.192
Shape factor (m)	0.0023	0.0091
Injector bottom-hole pressure (bar <sup>a</sup> )	334	355
Producer bottom-hole pressure (bar)	140	345
Field water injection rate (m <sup>3</sup> /day)	636	795
Field oil production rate at time $t$ and $t - 1$ (m <sup>3</sup> /day)	0	795
Field water production rate at time $t$ and $t - 1$ (m <sup>3</sup> /day)	0	619

<sup>a</sup>1 bar = 100 kPa

output data into the ANN for training, the database is normalized between 0 and 1, thus:

$$x_{\text{normalized}} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (25)$$

where  $x_{\text{normalized}}$  means the normalized value of  $x_i$ , which is the initial data, whereas  $x_{\max}$  and  $x_{\min}$ , respectively, indicate the maximum and minimum of data in a group of properties (Table 4). Pertaining to this, the ranges of the values of the training data used are shown in Table 5. By normalizing the data, the convergence condition can be further enhanced, and the ANN is more likely to “learn better” the relationship between the input and output data. Apart from this, the topology of the ANN utilized here is summarized in Table 6. The topology also included two bias nodes, which are not listed in Table 6. One of them was placed in between the input layer and the hidden layer, whereas another one was located between the hidden layer and the output layer.

In addition, the relevant parameters required to perform the backpropagation algorithms (SGD and Adam) and PSO algorithms are presented in Table 7. Regarding Adam, there are three other essential parameters, such as exponential decay rates of the estimates of the first and second moments, and constant of numerical stability. Here, the values of these three parameters were, respectively, assigned to be 0.9, 0.99, and  $10^{-7}$ . For PSO, because each of the weight (bias) is treated as one particle,

**Table 6.** Topology of the SPM

Type of layers	Number of layers	Number of nodes
Input	1	62
Hidden	1	10
Output	1	1

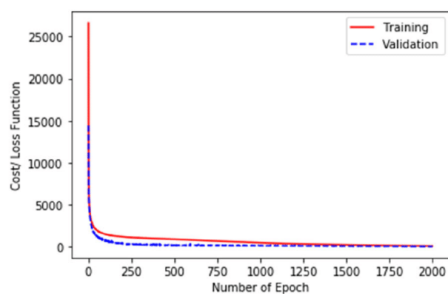
**Table 7.** Essential parameters for the SGD and PSO algorithms

SGD and Adam		PSO	
Parameters	Values	parameters	Values
Number of Epochs	2000	Number of Epochs	2000
Step size	0.01	Number of particle swarms	100
		Inertial weight	0.800
		Cognitive weight	1.005
		Social weight	1.050

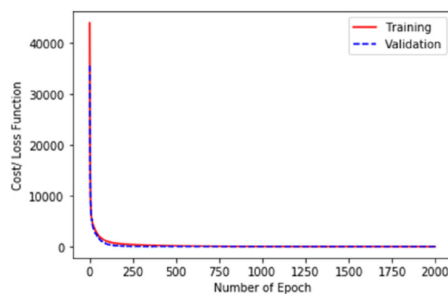
the number of particle swarms indicated the number of sets of particles used in the neural network training.

Thereafter, the normalized database was partitioned into three different sets, which are training, validation, and testing.<sup>5</sup> Here, 70% of the database (47,760 records) was used for training, 15% (10,235 records) for validation, and 15% (10,234 records) for testing. As the training set is fed into the ANN, it enables ANN to capture the underlying physical principles of the simulation by learning the relationship between input and output data. In addition, the validation set ensures that its respective error (loss) reduces, while the error produced by the training set also decreases. This downward trend reflects a healthy behavior of training process. In this study, it was essential to clarify that the validation set did not change the weights and biases (Mohaghegh 2018). It merely uses the weights and biases optimized by the training set to evaluate whether the training process is converging. In other words, the training set was employed to prevent any over-training or overfitting issue of the ANN (Mohaghegh 2018). Over-fitting occurs if the ANN memorizes the pattern of the data provided and it is unable to give a

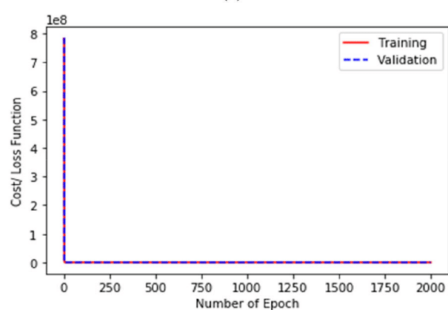
<sup>5</sup> Mohaghegh (2018) discussed that the spatio-temporal database should be divided into three different sets, namely training, calibration, and validation. In this paper, to elude confusion, the calibration set was termed as the validation set, whereas the validation set was referred to as the testing set.



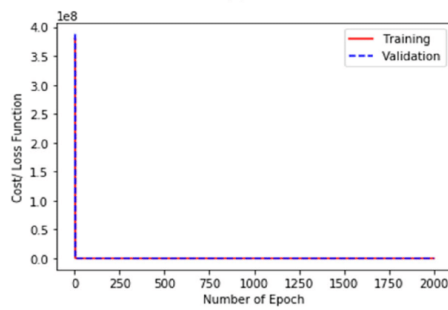
(a)



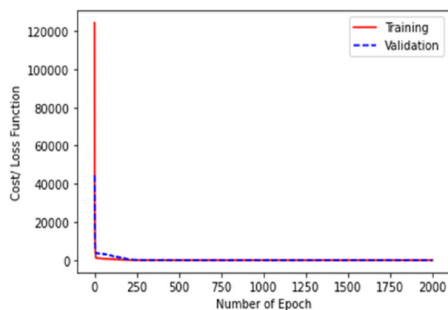
(a)



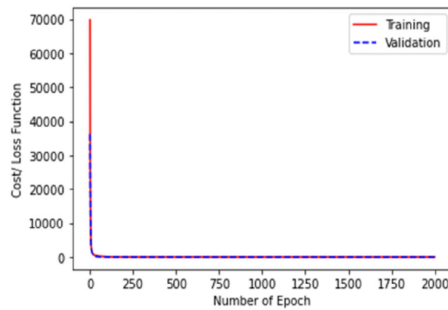
(b)



(b)



(c)



(c)

**Figure 7.** Oil production rate: plots of loss function against number of epochs for the smart proxy trained with (a) SGD, (b) PSO, and (c) Adam.

good prediction when other data are supplied. The testing set assists in checking the predictability of the trained neural network.

After the trained ANN was evaluated by the testing set, it should be provided with a new set of data (that were not from the database) to perform a blind case run. This step is crucial to further confirm the robustness of the developed SPM. Once the re-

**Figure 8.** Water production rate: plots of loss function against number of epochs for the smart proxy trained with (a) SGD, (b) PSO, and (c) Adam.

sults of the training and testing with a blind case run are within acceptable accuracy, the SPM can be employed for further analysis. The general workflow of building a SPM is summarized in Figure 6. As briefly discussed, the error function used in training the ANN was the mean squared error. However, for better evaluation of the performance of the ANN, other metrics including average absolute percentage

**Table 8.** Performance metrics of the smart proxy for oil rate prediction

		AAPE (%)	RMSE	$R^2$
Stochastic gradient descent	Training (758 data)	1.770	10.66	0.9954
	Validation (163 data)	1.567	7.512	0.9977
	Testing (162 data)	1.768	7.769	0.9971
Particle swarm optimization	Training (758 data)	0.349	2.378	0.9998
	Validation (163 data)	0.536	14.22	0.9934
	Testing (162 data)	0.352	2.408	0.9998
Adam	Training (758 data)	0.617	1.829	0.9999
	Validation (163 data)	0.649	2.036	0.9998
	Testing (162 data)	0.646	1.487	0.9999

**Table 9.** Performance metrics of the smart proxy for water rate prediction

		AAPE (%)	RMSE	$R^2$
Stochastic gradient descent	Training (758 data)	–	1.728	0.9998
	Validation (163 data)	6.461	1.685	0.9998
	Testing (162 data)	8.159	1.652	0.9999
Particle swarm optimization	Training (758 data)	6.565	0.547	0.9999
	Validation (163 data)	–	0.864	0.9999
	Testing (162 data)	7.629	0.761	0.9999
Adam	Training (758 data)	6.753	0.475	0.9999
	Validation (163 data)	4.914	0.262	0.9999
	Testing (162 data)	6.504	0.389	0.9999

error (AAPE%), root-mean-squared error (RMSE), and the correlation coefficient ( $R^2$ ) were also implemented, and their corresponding formulas are:

$$\text{AAPE}(\%) = \frac{1}{N} \sum_{i=1}^N \left| \frac{t_i - o_i}{t_i} \right| \times 100 \quad (26)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (t_i - o_i)^2} \quad (27)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (t_i - o_i)^2}{\sum_{i=1}^N (o_i - \bar{o})^2} \quad (28)$$

where  $N$  is total number of data in a set,  $t_i$  is the target or actual output value,  $o_i$  is the estimated output value, and  $\bar{o}$  is the mean of the actual output values.

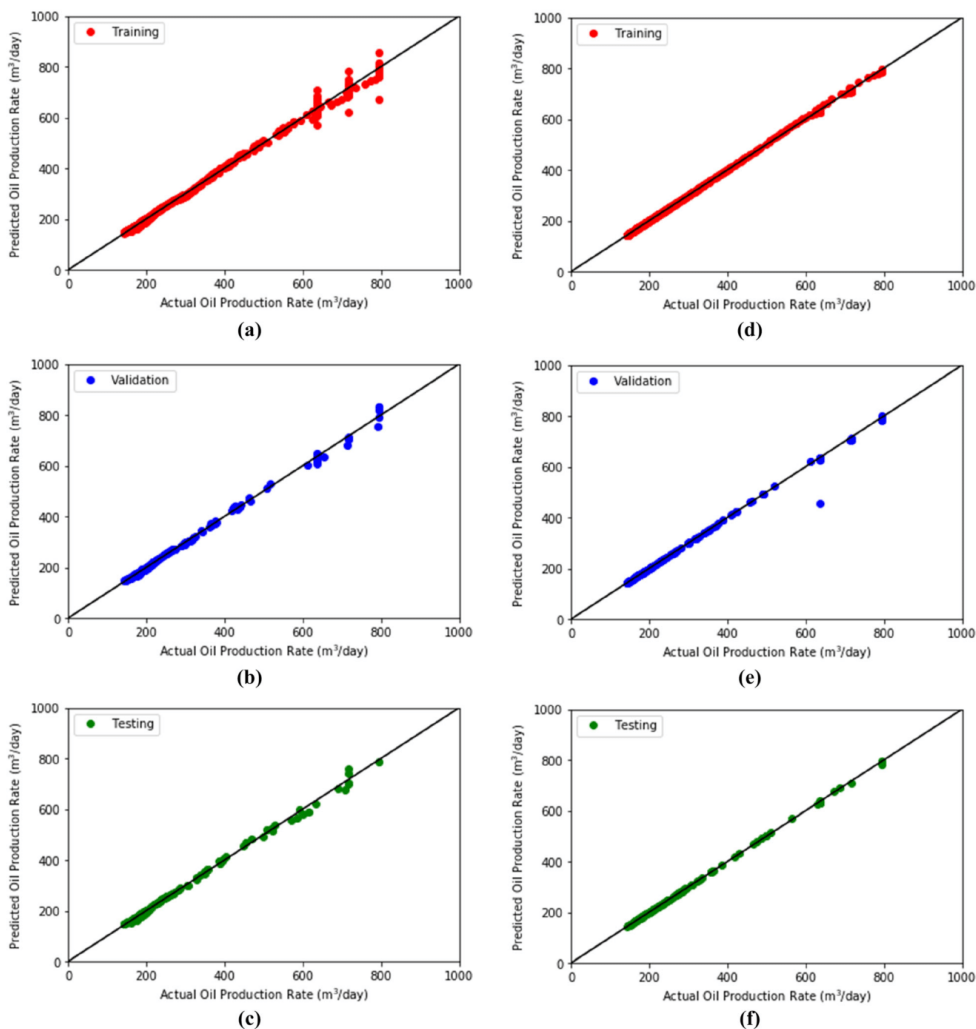
## RESULTS AND DISCUSSION

As mentioned above, we built two SPMs to correspondingly predict oil production rates and water production rates at a certain target of injection

rate. The topology presented in Table 6 was used to develop these proxy models. For each of these proxy models, the neural network training phase was performed separately by implementing the SGD, PSO, and Adam algorithms. Therefore, precisely speaking, there were 6 SPMs built here. Aside from the neural network training, the validation phase was also done simultaneously to ensure that the trained ANNs have a better generalization capability. Figures 7 and 8 show how the cost function deteriorated as the number of epochs increased in both training and validation phases when SGD, PSO, and Adam were utilized to train the ANN model. This decreasing trend gave a higher confidence that these trained ANN models had good performances in terms of prediction. This decreasing trend further confirmed that these ANNs were prevented from merely memorizing the pattern of the database provided. Thereafter, the testing phase was done to further investigate the predictive performance of the trained neural networks.

The results of the evaluation of the performance of the ANNs are presented in Table 8 for oil production rate prediction and Table 9 for the water production rate prediction. The corresponding cross-





**Figure 9.** Oil production rate: plots of correlation coefficient ( $R^2$ ): for SGD (a) training, (b) validation, (c) testing; for PSO (d) training, (e) validation, (f) testing; and for Adam (g) training, (h) validation, (i) testing.

plots between the actual output and the predicted output for the training, validation, and testing sets are illustrated in Figure 9 for oil production rate and Figure 10 for water production rate. Pertaining to the smart proxies for the prediction of oil rate, the

results shown in Table 8 indicate that Adam outperformed SGD and PSO in the training, validation, and testing phases in terms of RMSE and correlation coefficient. However, regarding AAPE, PSO had the best performance in all the three phases.

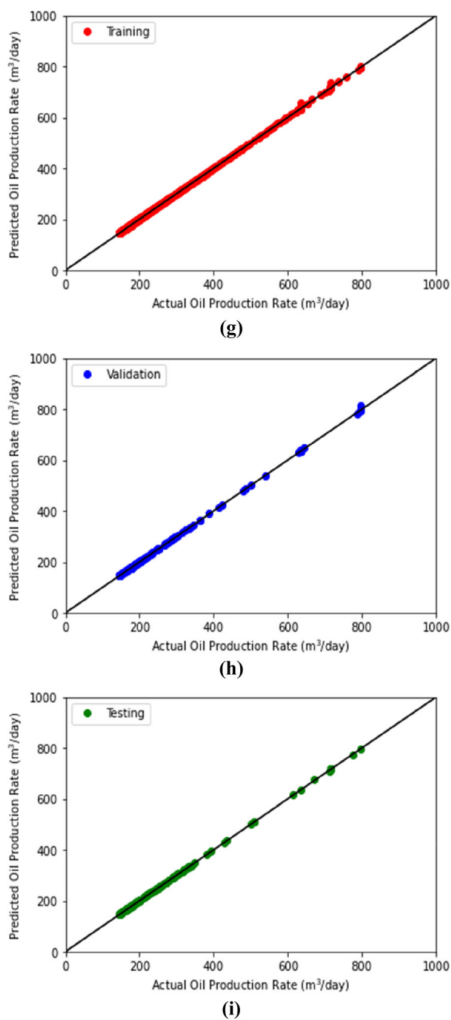


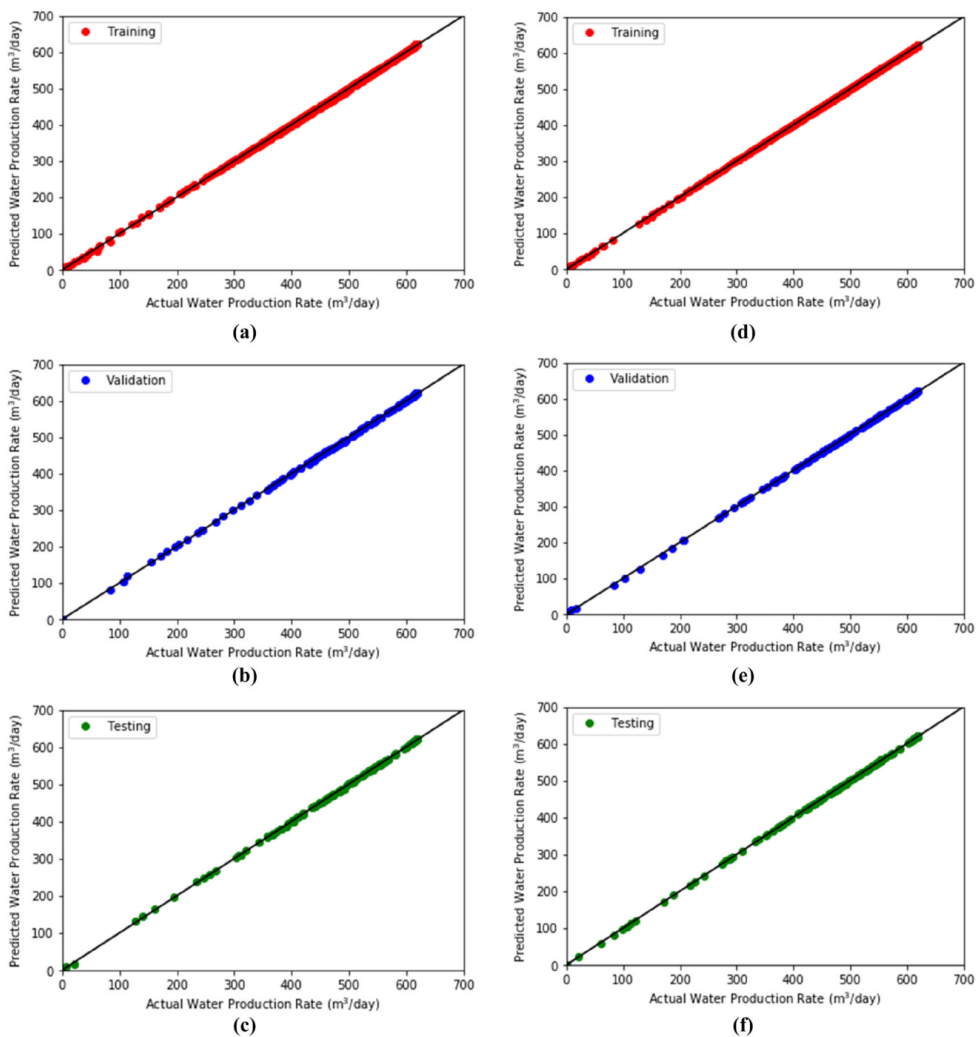
Figure 9. continued.

Additionally, better performance of Adam is also presented in Figure 9. As it can be observed, much more data samples lie on the 45-degree line as the Adam was used to develop the smart proxies compared to the cases where the SGD and PSO were utilized. Hence, Adam generally had the best performance, whereas PSO performed better than

SGD. Nonetheless, in the validation phase, SGD performed better than the PSO in terms of the minimization of RMSE and the maximization of the correlation coefficient. This can be due to the existence of an over-estimated data point (an outlier) in the validation phase of PSO (as shown in Figure 9e). Because the healthy training process is illustrated in Figure 7, it was deduced that any of these trained models was sufficiently good to be applied to predict the oil production rate. This is further justified by the results of the performance metrics in Table 8, which indicate that the correlation coefficients yielded by all the datasets exceeded 0.99 and both AAPEs and RMSEs exhibited in all the phases were considerably low.

For the prediction of water production rate (as illustrated in Figure 10), it is difficult to infer whether the backpropagation algorithm or the PSO yielded a better performance in the training, validation, and testing phases. However, according to, Adam generally had the best results as compared with SGD and PSO, whereas PSO performed better than SGD. In addition, the results of AAPE were not provided for the training phase of SGD and the validation phase of PSO because, in these phases, there were a few over-estimated data points (outliers) that caused the AAPE to be very large (more than 1000%). This is because when these data points were selected at the early stage of water breakthrough, the actual water production rate was very miniscule. Based on Eq. (26), if the numerator is in the order of magnitude of 1 or 10, then the AAPE will increase drastically. Thus, for practical reasons, the results were not shown here. Despite this, this scenario provided an insight that we needed to look at different performance metrics during SPM to determine whether the built proxy models functioned satisfactorily. Besides, these outliers did not affect the overall predictive capability of the smart proxy built here as the model was still able to capture the general data pattern during the development stage as presented in Figure 10.

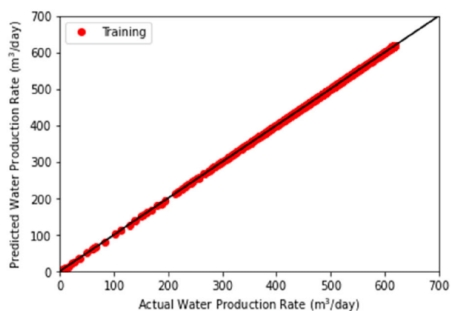
After developing the SPMs, two blind cases were run by using the target of the injection rates at 676 m<sup>3</sup>/day and 755 m<sup>3</sup>/day to provide more insightful ideas regarding the usefulness of the trained smart proxies. In other words, the spatio-temporal databases when the target of the injection rates was, respectively, at 676 m<sup>3</sup>/day and 755 m<sup>3</sup>/day created to be fed into the smart proxies to observe how well they can make predictions. It is essential to know that, in order to practically apply



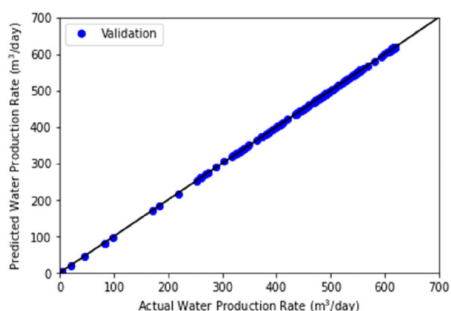
**Figure 10.** Water production rate: plots of the correlation coefficient ( $R^2$ ): for SGD (a) training, (b) validation, (c) testing; for PSO (d) training, (e) validation, (f) testing; and for Adam (g) training, (h) validation, (i) testing.

the smart proxy, the dynamic inputs should in fact be estimated by the smart proxy itself. For instance, the smart proxy in this work was developed to predict the oil production rates (also water production rates). This denotes that the oil production rate

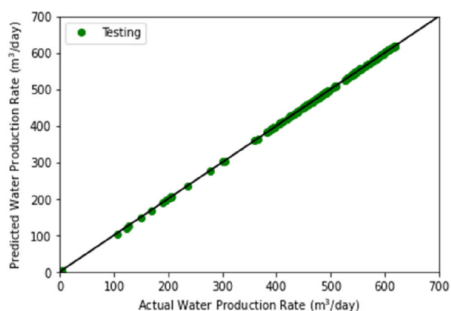
(water production rates) estimated at the timestep  $t-1$  should be used as one of the inputs to approximate the rate at the timestep  $t$ . Therefore, if there are more than one outputs to be predicted, then those estimated outputs at the current timestep



(g)



(h)

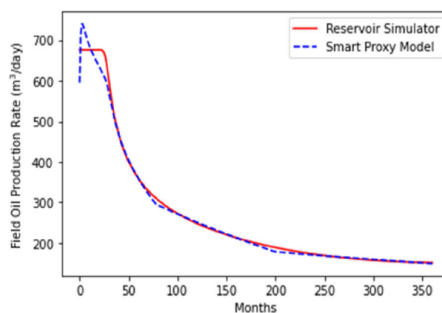


(i)

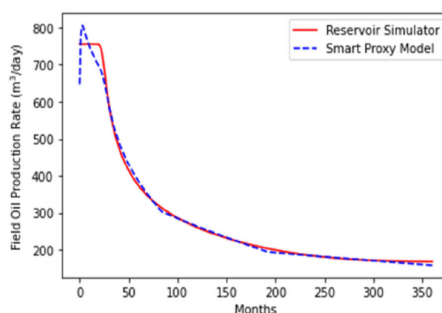
Figure 10. continued.

should be cascaded simultaneously to be the inputs at the next timestep. Alternatively, different smart proxy can be designed specifically to provide a pre-

<sup>6</sup> Building several smart proxies for estimating the dynamic inputs can reduce the convenience of SPM. So, the resolution of this issue will enable a smart proxy to be more tractable.



(a)

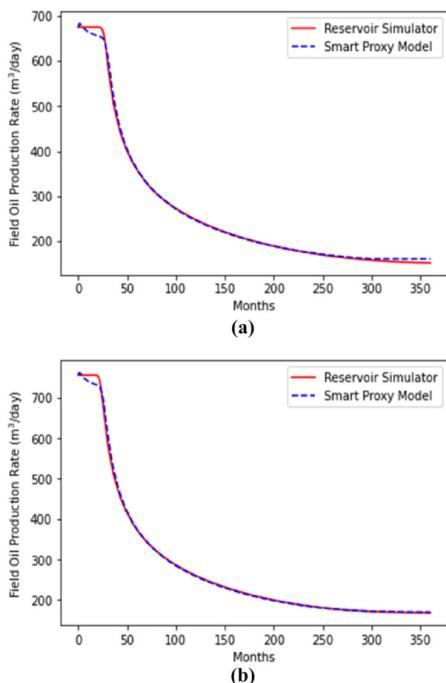


(b)

Figure 11. Oil rate prediction by SGD: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

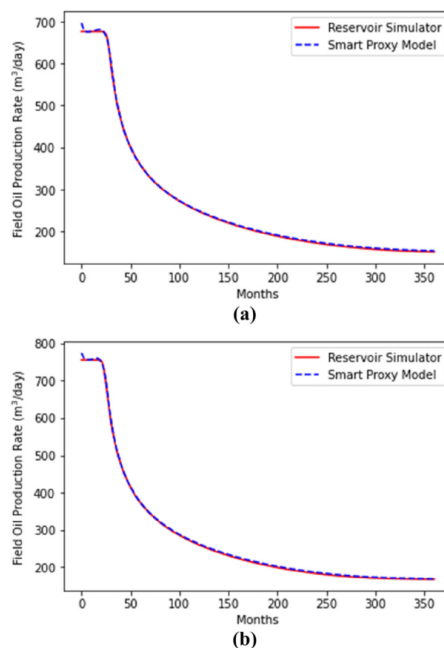
diction of any of the outputs, which is used as the input for another smart proxy. This situation reflects another disadvantage<sup>6</sup> of the application of smart proxy.

Here, only smart proxies that estimated the production rate were developed. For practical and illustrative purposes, other dynamic data, which are used as input data, were retrieved from the reservoir simulation as these data were not used directly in the optimization task discussed. Nevertheless, in this case, the oil production rate estimated by the smart proxy at the current timestep was cascaded to be the input for the approximation of the rate at the next timestep. The plots of the actual output (yielded by reservoir simulator) and the predicted output (produced by SPM) at injection rates of 676 m<sup>3</sup>/day and 755 m<sup>3</sup>/day are illustrated in Figure 11 for oil rate prediction using SGD, Figure 12 for oil rate prediction using PSO, Figure 13 for oil rate prediction



**Figure 12.** Oil rate prediction by PSO: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

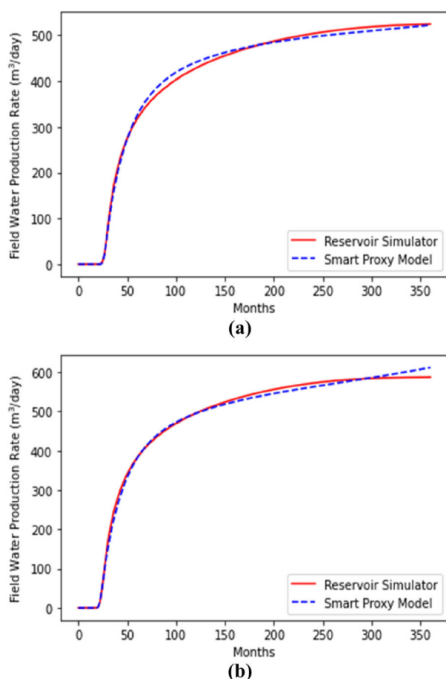
using Adam, Figure 14 for water rate prediction using SGD, Figure 15 for water rate prediction using PSO, and Figure 16 for oil rate prediction using Adam. The results of the performance analysis of the two blind cases are presented in Table 10 for oil rate prediction and in Table 11 for water rate prediction. Figures 11, 12, and 13 demonstrate that SGD results in a worse prediction at the beginning of the production (at both targets of injection rate) as compared to PSO and Adam. Despite this, the developed SPMs (trained by both algorithms) for oil rate prediction function were within an acceptable range of accuracy. This is verified by the results shown in Table 10. For water rate prediction, according to Figures 14, 15 and 16, it is explicit that the proxy trained with Adam yielded a better prediction than the models trained with SGD and PSO. However, it is challenging to determine whether PSO was better than SGD. In this case, Table 11



**Figure 13.** Oil rate prediction by Adam: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

shows that the model trained with PSO predicted better. In this case, the AAPEs resulted from the water rate prediction by using the model trained with SGD were not provided due to the same reason as discussed previously.

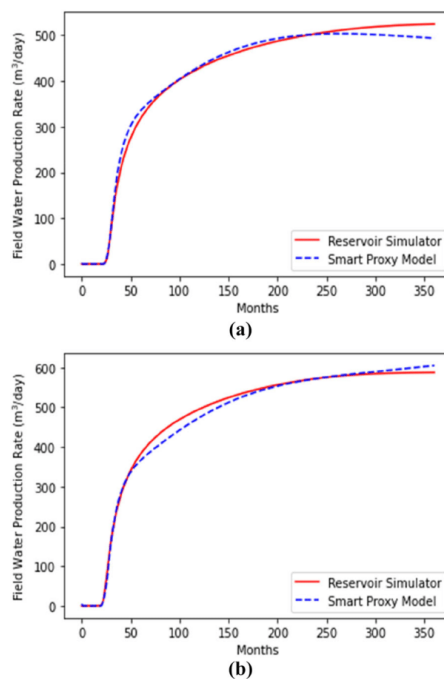
In general, when the two blind cases were employed, it was observed that the ANN models trained with any of the three algorithms for both oil and water rates prediction yielded results that are within acceptable range of accuracy. Nevertheless, the performance metrics illustrate that the SPMs built here (for prediction of both oil and water rates) trained by using Adam had a better predictive performance as compared to the models trained by SGD and PSO, whereas PSO outperformed SGD. In addition, we noticed that the SPMs (trained by using both algorithms) in this work had a better prediction of the oil production rates than the prediction of the water production rates. Hence, additional information (e.g., water breakthrough time, total production



**Figure 14.** Water rate prediction by SGD: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676  $\text{m}^3/\text{day}$ ; (b) injection rate of 755  $\text{m}^3/\text{day}$ .

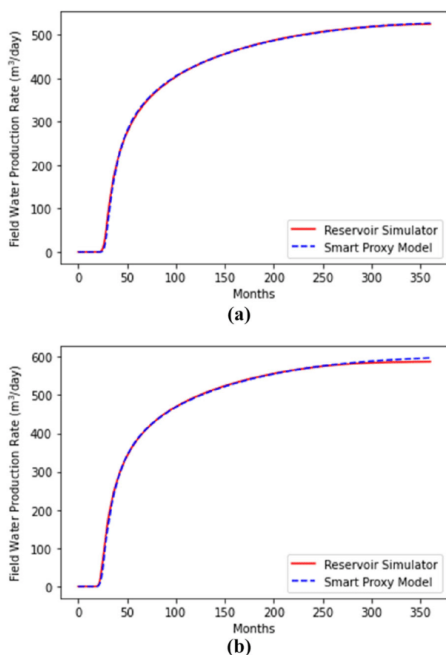
of water) can be included as input data to improve the performance of the SPM for water rate prediction.

After obtaining the flow rates predicted by the built SPMs, we proceeded to the illustrative production optimization task. As briefly discussed above, the optimization task here was to select the target of injection rate (between 676  $\text{m}^3/\text{day}$  and 755  $\text{m}^3/\text{day}$ ) that maximizes the objective function in Eq. (23). By using Eqs. (23) and (24) along with the parameters listed in Table 2, the evolution of NPV throughout the 30 years of production lifetime was determined and is presented in Figure 17. The base cases shown in Figure 17 correspond to the cases for the flow rates derived from the numerical reservoir simulation to determine the evolution of NPV. Both proxy models can reproduce the general trend of the NPV evolution that is close to the one generated by the base cases. This observation is justifiable as all



**Figure 15.** Water rate prediction by PSO: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676  $\text{m}^3/\text{day}$ ; (b) injection rate of 755  $\text{m}^3/\text{day}$ .

the proxy models yielded the general trends of both oil and water production rates as discussed earlier. Furthermore, from Table 12, all the models reached to the same decision that having the target of injection rate to be 755  $\text{m}^3/\text{day}$  for 30 years (without termination of production during the period of 30 years) will result in the maximum value of NPV. For the target rate of 676  $\text{m}^3/\text{day}$ , the percentage error of the NPV resulted from the smart proxy of SGD was about 2.67%, that of PSO was around 1.41%, and that of Adam was about 0.61%. For the target rate of 755  $\text{m}^3/\text{day}$ , the percentage errors of the NPVs resulted from both proxy models of SGD and PSO were close, namely 1.38% for SGD and 1.33% for PSO. However, for Adam, the percentage error was approximately 0.43%. In this case, the smart proxy trained by using Adam was deemed better. We understand that the economic model used here might be insufficient to reflect the real-life



**Figure 16.** Water rate prediction by Adam: plots of the comparison of rates for the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

**Table 10.** Performance metrics of the smart proxy for the two blind cases (oil rate prediction)

	Injection rate	AAPE (%)	RMSE	$R^2$
Stochastic gradient descent	676 m <sup>3</sup> /day	1.849	13.05	0.9924
	755 m <sup>3</sup> /day	1.978	13.23	0.9932
Particle swarm optimization	676 m <sup>3</sup> /day	1.391	5.701	0.9985
	755 m <sup>3</sup> /day	0.708	5.695	0.9988
Adam	676 m <sup>3</sup> /day	0.999	2.501	0.9997
	755 m <sup>3</sup> /day	1.057	2.830	0.9997

optimization case. However, we aimed to provide insights regarding the use of SPMs in production optimization on a fundamental level.

We also provide a brief discussion on the computational time of these proxy models to highlight the advantage of applying them. The computation here included all the training, validation, testing phases as well as the prediction using the two blind cases. It was done by using a PC with config-

**Table 11.** Performance metrics of the smart proxy for the two blind cases (water rate prediction)

	Injection rate	AAPE (%)	RMSE	$R^2$
Stochastic gradient descent	676 m <sup>3</sup> /day	–	13.63	0.9917
	755 m <sup>3</sup> /day	–	12.97	0.9935
Particle swarm optimization	676 m <sup>3</sup> /day	8.623	8.454	0.9968
	755 m <sup>3</sup> /day	7.266	8.975	0.9969
Adam	676 m <sup>3</sup> /day	8.049	2.790	0.9996
	755 m <sup>3</sup> /day	7.061	4.385	0.9993

urations that included Intel® Core™ i9-9900 CPU @3.10 GHz with 64.0 GB RAM. Here, the computation of one of the simulation scenarios listed in Table 3 took about 160 s to finish. When all the five simulation scenarios were run simultaneously, it spent about 290 s to be fully completed. Nevertheless, for the SPM developed here, the computation time of the proxy trained with SGD was about 110 s, that of PSO was about 50 s, and that of Adam was about 120 s.<sup>7</sup> In this aspect, the computation of the proxy trained with backpropagation algorithm was more expensive than that of PSO because PSO is a derivative-free method. In general, we saw that there was still a noticeable (even not very significant) difference in the computational time between the numerical simulation and the proxy models despite the low complexity of the reservoir model used here.

Further, we proposed and demonstrated the probabilistic application to investigate further the overall performance of the SPMs. In this case, one of the performance metrics, namely correlation coefficient  $R^2$ , was used for illustrative purpose in this part of the work. To do this probabilistic study of the built SPMs, we conducted the process of SPM iteratively for 200 times. This implies that there were 200 samples of  $R^2$  for training phase, validation phase, testing phase, and prediction for each of the two blind cases. Thereafter, the normalized cumulative frequency distribution (NCFD) for  $R^2$  that ranged between 0 and 1 was computed for the 200 samples. In this context, NCFD can be understood as the cumulative number of times for a sample to be within a range of values of  $R^2$  over 200 times. The plots of NCFD are presented in Figures 18, 19, 20, 21, and 22.

<sup>7</sup> Computational time of the proxy built for oil rate prediction was close to that of the proxy developed for water rate prediction.

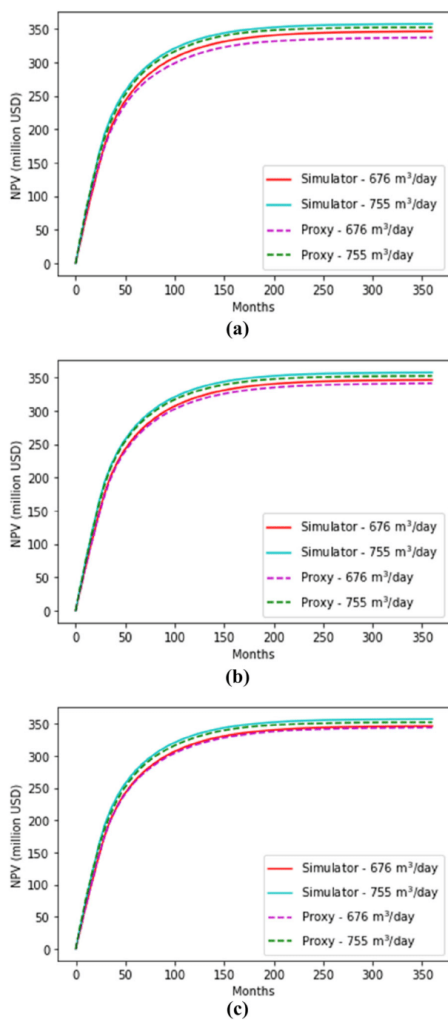


Figure 17. Evolution of NPV throughout the lifetime of production: (a) SGD; (b) PSO; (c) Adam.

Based on Figure 18, for the training phase of the SPMs, the models trained with PSO had relatively higher chance to result in a healthy training trend

than the models trained with the backpropagation algorithms. For the oil rate prediction, PSO had 0.5% chance to result in values of  $R^2$  less than 0.90, whereas SGD had 31% chance and Adam had 37.5% of chance. For the water rate prediction, PSO had about 99% chance to yield values of  $R^2$  that ranged between 0.99 and 1, whereas SGD and Adam, respectively, had only about 60% and 55% chance to achieve that. According to these results, we deduced that PSO was more likely to produce a healthy trend of training compared to SGD and Adam. This deduction is further justified by the results shown in Figure 19 for the validation phase.

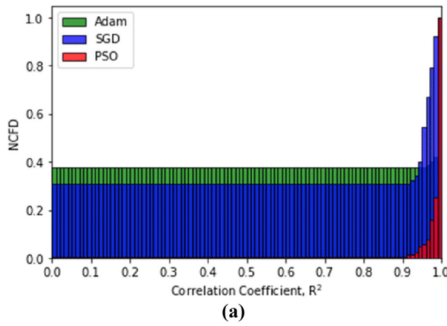
For the testing phase, it was noted that the proxy models trained by using PSO performed better than those of SGD and Adam when the models were evaluated against the testing dataset. As portrayed in Figure 20, for the case of oil rate, there was 26% chance that the model trained with PSO will produce values of  $R^2$  less than 0.99 in the testing phase, whereas there was 76% chance that the model trained with SGD will do so; for Adam, the chance was about 47%. Besides, for the case of water rate, PSO had 4% chance to have values of  $R^2$  less than 0.99, whereas SGD had 41.5% and Adam had 45.5%. This provided more confidence that PSO has a higher chance to yield a better predictive performance than SGD when the models were tested with the dataset from a blind scenario.

For the prediction of rates against the datasets from the two blind cases, it can be noticed that, in general, the proxy models by PSO more likely had a better predictive performance than those by SGD and Adam despite the fact that the former had slightly higher chance to produce  $R^2$  values that are less than 0.90 compared with that SGD had in terms of oil rate prediction for injection scenario of 676 m<sup>3</sup>/day. This is because based on the prediction of  $R^2$  that ranged between 0.99 and 1, the models by PSO were deemed more reliable than those by SGD and Adam. Besides, in terms of oil rate prediction, Adam statistically had a better chance than SGD in yielding  $R^2$  values between 0.99 and 1 for both injection scenarios. However, for water rate prediction, the chances of both algorithms were very close. We have illustrated that, here, statistically speaking, PSO had a better chance to perform better in training and building the proxy model compared to SGD and Adam. Because PSO is metaheuristics, in

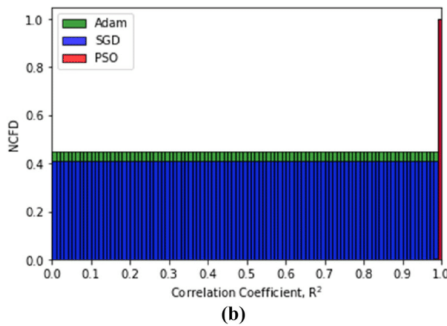


**Table 12.** Optimal NPVs generated by using all the models

Injection rates	676 m <sup>3</sup> /day				755 m <sup>3</sup> /day			
	Simulator	SGD	PSO	Adam	Simulator	SGD	PSO	Adam
NPV <sub>optimal</sub> (million USD)	346.36	337.11	341.49	344.27	357.35	352.43	352.59	355.84

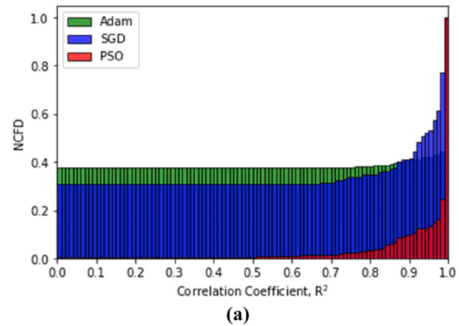


(a)

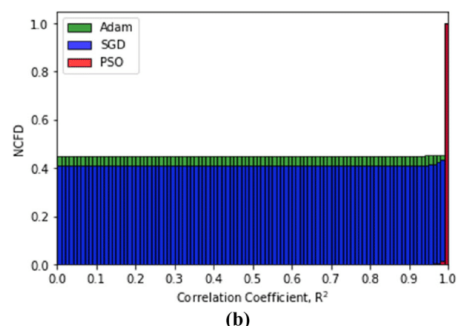


(b)

**Figure 18.** NCFD of  $R^2$  for the training phase of the SPMs: (a) oil rate; (b) water rate.



(a)



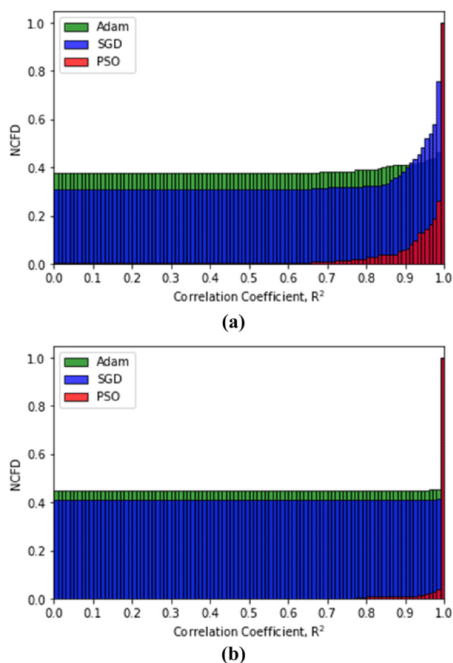
(b)

**Figure 19.** NCFD of  $R^2$  for the validation phase of the SPMs: (a) oil rate; (b) water rate.

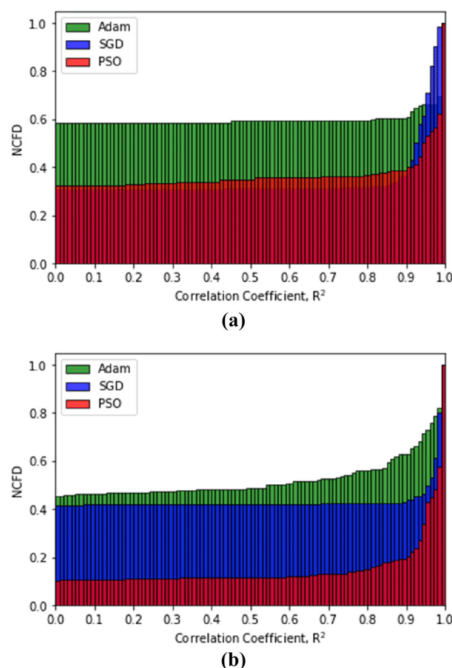
which both global search and local search are balanced, it has a higher chance to have a more exhaustive search in the solution space during the neural network training. Nevertheless, we recommend that this study is conducted using other performance metrics for a more established understanding regarding the outcomes of SPM. Integration of this statistical study in SPM can provide insights about the reliability of an algorithm in training a proxy model and the prediction accuracy of the trained models.

### HETEROGENEOUS MODEL

To demonstrate further the robustness of the methodology, we used another fractured reservoir model as a second case study. The general architecture and fluid properties of this new model are similar to those of the previous model. However, we changed the values of some reservoir parameters, including the height of matrix block and the porosity values of both matrix and fracture media, and introduced heterogeneity to the permeability fields



**Figure 20.** NCFD of  $R^2$  for the testing phase of the SPMs: (a) oil rate; (b) water rate.



**Figure 21.** NCFD of  $R^2$  for the prediction of rate of the SPMs when target rate was 676 m<sup>3</sup>/day: (a) oil rate; (b) water rate.

**Table 13.** Modified reservoir parameters for the heterogeneous model

Layer	Matrix block height (m)	Porosity	
		Matrix media	Fracture media
1	4.572	0.150	0.0050
2	10.67	0.400	0.0020
3	7.620	0.280	0.0015

of both media. In this case, the heterogeneity only applies to permeability. The permeability values in the  $x$ -,  $y$ -, and  $z$ - directions are the same. Thus, the fractured model illustrated here is an isotropic heterogeneous model. Refer to Table 13 for the new values of the heights of matrix blocks and the porosity values. Figure 23 shows the permeability field of each layer in the unit of m<sup>2</sup>.

After building this new model by applying the same methodology, the database was extracted and used to develop the SPMs to correspondingly predict the field oil and water production rates. The injection scenarios employed in this case study were the same as in Table 3. The structure of ANN models built here also remained the same as presented in Table 6. This also applied to the use of essential parameters of the three algorithms. For practical and concise purposes, only two performance metrics, namely RMSE and  $R^2$ , were implemented to evaluate the training and predictive performance of these proxy models. Table 14 shows the results of training, validation, and testing of the SPM for oil production rate forecasting, whereas Table 15 presents those of the model for water production rate prediction. Generally, the models trained by all the three algorithms yielded excellent training results for both oil and water production rates. Based on

**Table 14.** Performance metrics of the smart proxy for oil rate prediction based on training, validation, and testing sets

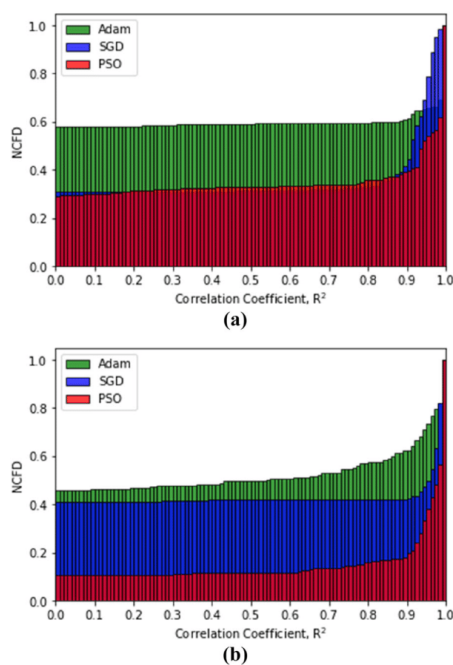
		RMSE	$R^2$
Stochastic gradient descent	Training (758 data)	7.855	0.9977
	Validation (163 data)	4.700	0.9992
	Testing (162 data)	8.202	0.9978
Particle swarm optimization	Training (758 data)	3.846	0.9995
	Validation (163 data)	3.918	0.9995
Adam	Testing (162 data)	2.739	0.9997
	Training (758 data)	3.154	0.9997
	Validation (163 data)	2.410	0.9998
	Testing (162 data)	3.391	0.9996

**Table 15.** Performance metrics of the smart proxy for water rate prediction based on training, validation, and testing sets

		RMSE	$R^2$
Stochastic gradient descent	Training (758 data)	2.401	0.9998
	Validation (163 data)	2.273	0.9998
	Testing (162 data)	2.379	0.9998
Particle swarm optimization	Training (758 data)	1.869	0.9999
	Validation (163 data)	1.961	0.9999
Adam	Testing (162 data)	1.824	0.9999
	Training (758 data)	0.540	0.9999
	Validation (163 data)	0.478	0.9999
	Testing (162 data)	0.422	0.9999

both RMSE and  $R^2$ , Adam had the best results for both oil and water production rates. Nevertheless, for the testing phase in oil rate proxy model, PSO outperformed the others. For illustrative purposes, only the production profiles estimated by the smart proxies trained by using Adam are presented; the oil profiles are shown in Figure 24, whereas the water profiles are presented in Figure 25.

Thereafter, these models also underwent the blind validation phases by using the two blind cases as explained before. Table 16 records the results of blind validation for oil rate prediction, and Table 17 shows the results for water rate forecasting. For this case study, the PSO outperformed the others when it was used to train the predictive model of oil production rate. However, for the estimation of water production rate, Adam still yielded the predictive model that produced the best results. Then, the production optimization was also done by using the

**Figure 22.** NCFD of  $R^2$  for the prediction of rate of the SPMs when target rate was 755 m<sup>3</sup>/day: (a) oil rate; (b) water rate.

same price setting as shown in Table 2 to highlight the fundamental practicality of the models developed in this case study. The optimal NPVs obtained by using each of the proxy models are tabulated in Table 18.

Based on Table 18, it was deduced that the proxy models built by using Adam produced the optimal NPV with the least percentage error under two different injection scenarios, which were 0.117% for injection rate of 676 m<sup>3</sup>/day and 0.329% for injection rate of 755 m<sup>3</sup>/day. In addition, all the proxy models reached the same option that the injection rate of 755 m<sup>3</sup>/day was economically preferable. Apart from these, for illustrative and succinct purposes, the probabilistic application was only implemented to analyze the predictive performance of each model. The results of this application are demonstrated in Figure 26 for the target rate of 676 m<sup>3</sup>/day and in Figure 27 for the target rate of 755 m<sup>3</sup>/day. In general, for this case study, it can be

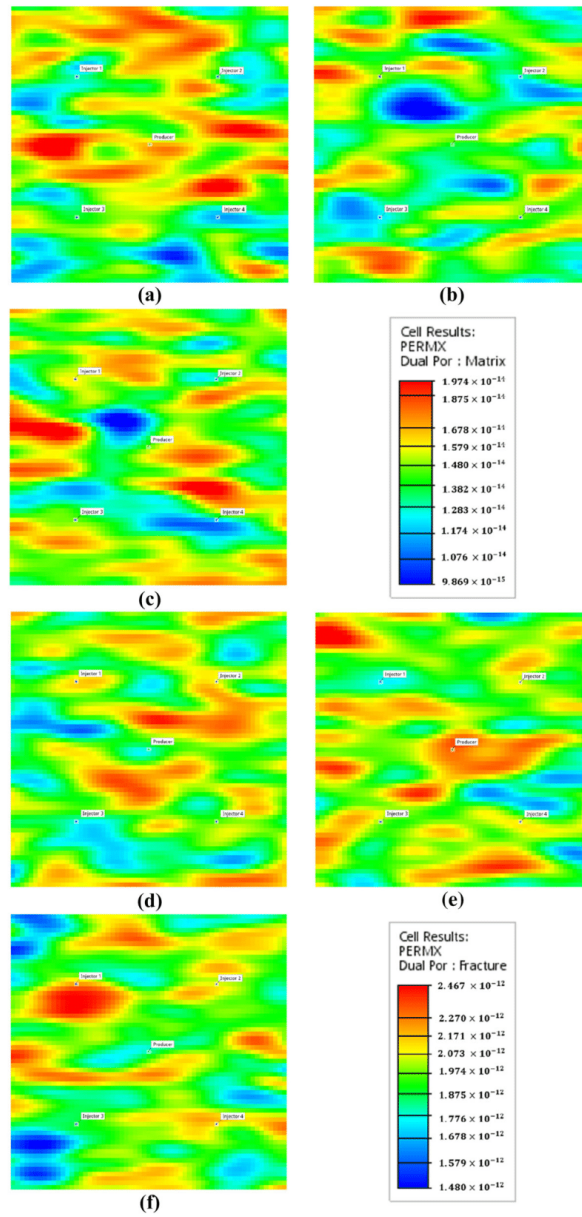
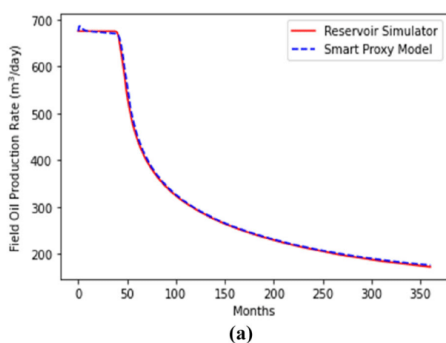


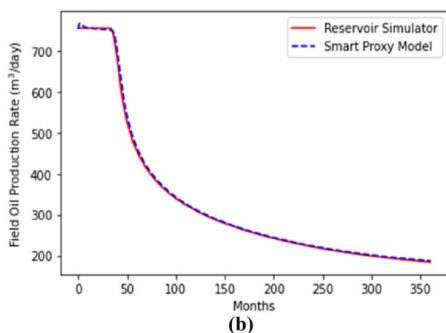
Figure 23. Overview of the isotropic heterogeneous model. The matrix system consists of (a) Layer 1, (b) Layer 2, and (c) Layer 3. The fracture system comprises (d) Layer 1, (e) Layer 2, (f) Layer 3.

**Table 16.** Oil rate prediction: performance metrics of the smart proxy for the two blind cases

	Injection rate	RMSE	$R^2$
Stochastic gradient descent	676 m <sup>3</sup> /day	12.45	0.9939
	755 m <sup>3</sup> /day	13.04	0.9944
Particle Swarm Optimization	676 m <sup>3</sup> /day	2.097	0.9998
	755 m <sup>3</sup> /day	3.827	0.9995
Adam	676 m <sup>3</sup> /day	4.489	0.9992
	755 m <sup>3</sup> /day	5.468	0.9990



(a)



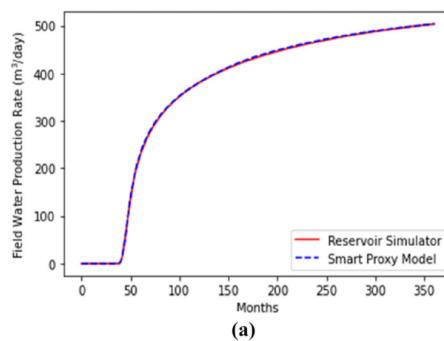
(b)

**Figure 24.** Oil rate prediction by Adam: plots of the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

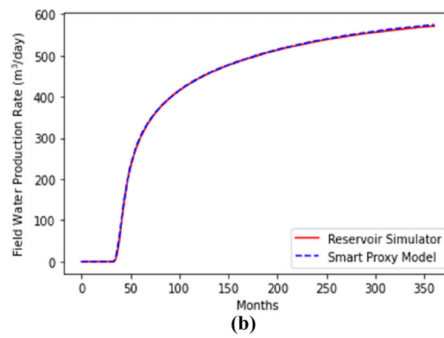
deduced that PSO had a better chance than both SGD and Adam to produce a predictive model with higher accuracy level (i.e.,  $R^2$  exceeding 0.99).

## CONCLUSIONS

Here, we have shown how SPM can be conducted by using a synthetic fractured reservoir



(a)



(b)

**Figure 25.** Water rate prediction by Adam: plots of the results predicted by the trained smart proxy for the two blind cases: (a) injection rate of 676 m<sup>3</sup>/day; (b) injection rate of 755 m<sup>3</sup>/day.

model. The purpose of this study was to provide some insights and a more concrete demonstration regarding the modeling of a smart proxy. We also briefly discussed how the spatio-temporal database can be generated, and we presented the selection of input and output data which were used in the neural network training. This procedure is of paramount importance as a good database determines the success of SPM. Apart from implementing the back-

propagation algorithms, namely SGD and Adam, to train the smart proxy, we also demonstrated how the training of a smart proxy can be coupled with PSO. Regarding this, for each training algorithm, we developed two SPMs which correspondingly predicted oil production rate and water production rate. During the development of the smart proxies, all the three algorithms showed excellent training results. However, for the proxy of water rate prediction (trained with both SGD and PSO), some of the resulting AAPEs were large due to the existence of outliers. Despite this, the proxy still showed healthy training and validation trend. In addition, both models illustrated splendid predictive performance as indicated by the results. This shows that the overall predictive performance of the smart proxies remains intact despite having outliers in the neural network training. We consider this as one of the important contributions derived from this work because most of the available literatures solely focus on the use of traditional backpropagation algorithm in SPM. Thereafter, we showed how these SPMs can be used to optimize production through an illustrative example. Besides, we used the performance metrics of correlation coefficient ( $R^2$ ) for probabilistic evaluation of the overall performance of the SPMs. We summarize our main findings and results derived from this work as follows.

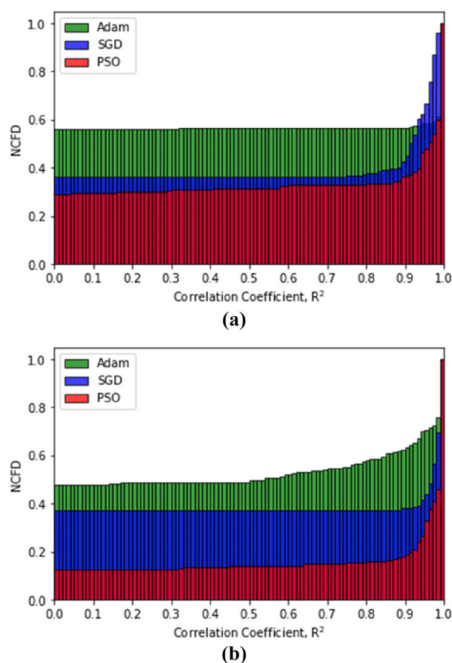
**Table 17.** Water rate prediction: performance metrics of the smart proxy for the two blind cases

	Injection rate	RMSE	$R^2$
Stochastic gradient descent	676 m <sup>3</sup> /day	5.723	0.9987
	755 m <sup>3</sup> /day	10.25	0.9966
Particle swarm optimization	676 m <sup>3</sup> /day	9.966	0.9961
	755 m <sup>3</sup> /day	7.705	0.9981
Adam	676 m <sup>3</sup> /day	1.589	0.9999
	755 m <sup>3</sup> /day	1.921	0.9999

1. Based on the deterministic analysis conducted for SPM of oil rate prediction, the performance metrics (based on training, validation, and testing) showed that Adam generally yielded lower AAPE, RMSE, and higher  $R^2$  than SGD and PSO. However, for the RMSE in the validation phase, PSO resulted in the highest value due to the existence of outliers as previously discussed. Besides, for SPM of water rate prediction, the performance metrics portrayed that Adam was also generally better than SGD and PSO.
2. For oil rate prediction of the blind cases, proxy model with Adam also had the lowest AAPE, RMSE, and the highest  $R^2$ . The same results were obtained for water rate prediction.
3. For the production optimization case, the SPMs trained with all three algorithms reached the same decision as what the base case did, which was to select the target injection rate to be 755 m<sup>3</sup>/day. However, the NPVs calculated using the data obtained from the proxy model built with Adam were much closer to those estimated by using the data from reservoir simulator.
4. According to the probabilistic analysis for prediction of oil and water rates, it is inferred that PSO has a higher chance to generate a SPM that can result in excellent training and predictive performance compared with SGD and Adam.
5. The same methodology was also applied to an isotropic heterogeneous fractured reservoir model to illustrate its robustness. For this, it was generally found out that Adam can outperform SGD and PSO in the development of the SPMs. However, for oil production rates, PSO produced a better testing result. Regarding blind validation, Adam also generally resulted in more accu-

**Table 18.** Optimal NPVs generated by using all the models

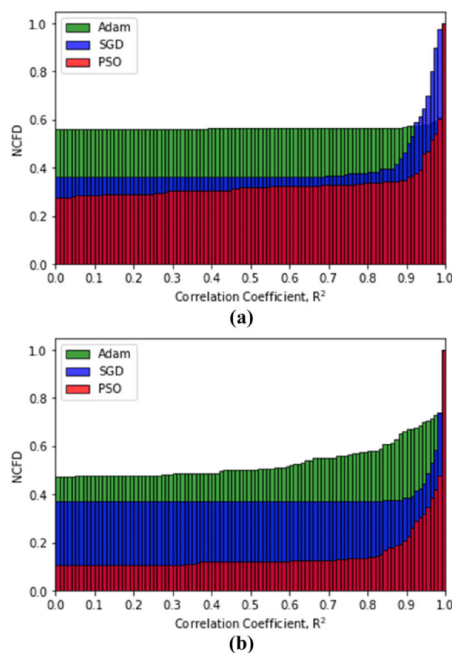
Injection rates	676 m <sup>3</sup> /day				755 m <sup>3</sup> /day			
	Simulator	SGD	PSO	Adam	Simulator	SGD	PSO	Adam
NPV <sub>optimal</sub> (million USD)	428.92	421.77	424.31	429.43	447.22	440.97	444.04	448.69



**Figure 26.** NCFD of  $R^2$  for the prediction of rate of the SPMs when target rate was 676 m<sup>3</sup>/day: (a) oil rate; (b) water rate.

rate predictive models of water production rates. Nonetheless, the predictive model of oil rates established by using PSO estimated the oil profile more accurately. Additionally, PSO showed higher chance than SGD and Adam to produce models with excellent predictive ability.

Based on the findings presented, we conclude that, in this work, a metaheuristic algorithm can be applied aptly to train and build a good smart proxy of a fractured reservoir model. Although it has been demonstrated that PSO might not deterministically outperform the considered backpropagation algorithms in smart proxy modeling, statistically it still has a better chance to yield a good performance in this case study. Nonetheless, we understand that there are still some shortcomings regarding these SPMs. We hope that these proxies can be enhanced



**Figure 27.** NCFD of  $R^2$  for the prediction of rate of the SPMs when target rate was 755 m<sup>3</sup>/day: (a) Oil rate; (b) Water rate.

to be more tractable and robust<sup>8</sup> in terms of prediction of any reservoir-related parameter. In short, we believe that we have achieved the main goals of this work, which include a vivid illustration of SPM, an integration of metaheuristic algorithm in proxy training, a presentation of practical use of the built proxies in optimization on a fundamental level, and an inclusion of a probabilistic application in evaluating a proxy model.

## ACKNOWLEDGMENTS

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)). The authors would also like

<sup>8</sup> “Robust” here implies that the smart proxy model can be used in real-life cases.

to acknowledge the support given by the Department of Geoscience and Petroleum (IGP) of NTNU to this research work.

## OPEN ACCESS

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## FUNDING

Open access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital)./FundingInformation>

## REFERENCES

- Ahmad, S. A., & Olivier, R. G. (2008). Matrix-fracture transfer function in dual-medium flow simulation: review, comparison, and validation. In *Europec/EAGE Conference and Exhibition*. Rome, Italy. <https://doi.org/https://doi.org/10.2118/113890-MS>.
- Alenezi, F., & Mohaghegh, S. D. (2016). A data-driven smart proxy model for a comprehensive reservoir simulation. In *The 4th Saudi International Conference on Information Technology (Big Data Analysis) (KACSTIT)*. Riyadh, Saudi Arabia. <https://doi.org/10.1109/KACSTIT.2016.7756063>.
- Alenezi, F., & Mohaghegh, S. D. (2017). Developing a smart proxy for the SACROC water-flooding numerical reservoir simulation model. In SPE Western Regional Meeting. Bakersfield, California, United States. <https://doi.org/https://doi.org/10.2118/185691-MS>.
- Barrenblatt, G. I., Zheltov, I. P., & Kochina, I. N. (1960). Basic concepts in the theory of homogeneous liquids in fissured rocks. *Journal of Applied Mathematics and Mechanics*, 24(5), 1286–1303.
- Bianchi, L., Dorigo, M., Gambardella, L. M., & Gutjahr, W. J. (2009). A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing*, 8, 239–287.
- British Petroleum. (2020). BP Statistical Review of World Energy 2020. <https://www.bp.com/en/global/corporate/energy-economics/statistical-review-of-world-energy.html>. Retrieved July 21, 2020.
- Buduma, N., & Locasio, N. (2017). *Fundamentals of deep learning: Designing next-generation machine intelligence algorithms*. Sebastopol, California, United States: O'Reilly.
- Ceetron Solution AS. (2020). *ResInsight*.
- Chollet, F. (2019). *Keras, version 2.3.0*.
- Ertekin, T., & Sun, Q. (2019). Artificial intelligence applications in reservoir engineering: A status check. *Energies*, 12(15), 2897.
- Firoozabadi, A., & Thomas, L. K. (1990). Sixth SPE comparative solution project: Dual-porosity simulators. *Journal of Petroleum Technology*, 42(6), 710–715.
- Gerald, P., Raftley, A. E., Ševčíková, H., Li, N., Gu, D., Spoorenberg, T., et al. (2014). World population stabilization unlikely this century. *Science*, 346(6206), 234–237.
- Gharbi, R. B. C., & Mansoori, G. A. (2005). An introduction to artificial intelligence applications in petroleum exploration and production. *Journal of Petroleum Science and Engineering*, 49(3–4), 93–96.
- Gilman, J. R., & Kazemi, H. (1983). Improvements in simulation of naturally fractured reservoirs. *Society of Petroleum Engineers Journal*, 23(4), 695–707.
- Google Brain Team. (2020). *TensorFlow, version 2.1.0*.
- He, Q., Mohaghegh, S. D., & Liu, Z. (2016). Reservoir simulation using smart proxy in SACROC unit—case study. 2016. In *SPE Eastern Regional Meeting*. Canton, Ohio, United States. <https://doi.org/https://doi.org/10.2118/184069-MS>.
- International Energy Agency. (2018). World Energy Outlook 2018. International Energy Agency. <https://www.iea.org/weo2018/>. Retrieved September 24, 2019.
- Kazemi, H., Merrill, L. S., Jr., Porterfield, K. L., & Zeman, P. R. (1976). Numerical simulation of water-oil flow in naturally fractured reservoirs. *Society of Petroleum Engineers Journal*, 16(6), 317–326.
- Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. In *IEEE International Conference on Neural Networks*. Perth, Western Australia, Australia. <https://doi.org/10.1109/ICNN.1995.488968>.
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *The 3rd International Conference for Learning Representations*. San Diego, California, United States. [arxiv.org/abs/1412.6980](https://arxiv.org/abs/1412.6980).
- Lescaroux, F., & Mignon, V. (2009). On the influence of oil prices on economic activity and other macroeconomic and financial variables. *OPEC Energy Review*, 32(4), 343–380.
- Mohaghegh, S. D. (2000a). Virtual Intelligence applications in petroleum engineering: Part 1—artificial neural networks. *Journal of Petroleum Technology*, 52(9), 64–72.
- Mohaghegh, S. D. (2000b). Virtual intelligence applications in petroleum engineering: Part 2—evolutionary computing. *Journal of Petroleum Technology*, 52(10), 40–46.
- Mohaghegh, S. D. (2000c). Virtual intelligence applications in petroleum engineering: Part 3—fuzzy logic. *Journal of Petroleum Technology*, 52(11), 82–87.
- Mohaghegh, S. D., Modavi, C. A., Hafez, H. H., Haajizadeh, M., Kenawy, M.M., & Guruswamy, S. (2006). Development of surrogate reservoir models (SRM) for fast track analysis of complex reservoirs. In *Intelligent Energy Conference and Exhibition*. Amsterdam, The Netherlands. <https://doi.org/https://doi.org/10.2118/99667-MS>.



- Mohaghegh, S. D. (2011). Reservoir simulation and modeling based on pattern recognition. In *SPE Digital Energy Conference and Exhibition*. The Woodlands, Texas, USA. <https://doi.org/https://doi.org/10.2118/143179-MS>.
- Mohaghegh, S. D., Liu, J. S., Gaskari, R., Mayasami, M., & Olukoko, O. A. (2012a). Application of surrogate reservoir models (SRM) to an onshore green field in Saudi Arabia; Case Study. In *North Africa Technical Conference and Exhibition*. Cairo, Egypt. <https://doi.org/https://doi.org/10.2118/151994-MS>.
- Mohaghegh, S. D., Amini, S., Gholami, V., Gaskari, R., & Bromhal, G. S. (2012b). Grid-based surrogate reservoir modeling (SRM) for fast track analysis of numerical reservoir simulation models at the gridblock level. In *SPE Western Regional Meeting*. Bakersfield, California, United States. <https://doi.org/https://doi.org/10.2118/153844-MS>.
- Mohaghegh, S. D., Liu, J. S., Gaskari, R., Mayasami, M., & Olukoko, O. A. (2012c). Application of surrogate reservoir models (SRM) to Two offshore fields in Saudi Arabia; Case Study. In *SPE Western Regional Meeting*. Bakersfield, California, United States. <https://doi.org/https://doi.org/10.2118/153845-MS>.
- Mohaghegh, S. D., Abdulla, F., Abdou, M., Gaskari, R., & Mayasami, M. (2015). Smart proxy: An innovative reservoir management tool; Case study of a giant mature oilfield in the UAE. In *Abu Dhabi International Petroleum Exhibition and Conference*. Abu Dhabi, UAE. <https://doi.org/https://doi.org/10.2118/177829-MS>.
- Mohaghegh, S. D. (2017). *Data-Driven Reservoir Modeling*. Richardson, Texas, United States: Society of Petroleum Engineers.
- Mohaghegh, S. D. (2018). *Data-driven analytics for the geological storage of CO<sub>2</sub>*. Boca Raton, Florida, United States: CRC Press.
- Miranda, L. J. V. (2019). *PySwarms, version 1.1.0*.
- Nait Amar, M., Zeraibi, N., & Redouane, K. (2018). Bottom hole pressure estimation using hybridization neural networks and grey wolves optimization. *Petroleum*, 4(4), 419–429.
- Nait Amar, M., Zeraibi, N., & Redouane, K. (2018). Pure CO<sub>2</sub>-oil system miscibility pressure prediction using optimized neural network by differential evolution. *Petroleum and Coal*, 60(2), 284–293.
- Nait Amar, M., & Jahanbani Ghahfarokhi, A. (2020). Prediction of CO<sub>2</sub> diffusivity in brine using white-box machine learning. *Journal of Petroleum Science and Engineering*, 190, 107037.
- Nait Amar, M., Zeraibi, N., & Jahanbani Ghahfarokhi, A. (2020). Applying hybrid support vector regression and genetic algorithm to water alternating CO<sub>2</sub> gas EOR. *Greenhouse Gases: Science and Technology*, 10(3), 613–630.
- Parada, C. H., & Ertekin, T. (2012). A New screening tool for improved oil recovery methods using artificial neural networks. In *SPE Western Regional Meeting*. Bakersfield, California, United States. <https://doi.org/https://doi.org/10.2118/153321-MS>.
- Python Software Foundation. (2020). *Python version 3.8.1*.
- Romm, E. S. (1966). *Fluid flow in fractured rocks*. Moscow, Russia: Nedra Publishing House. (in Russian).
- Schlumberger. (2020a). *ECLIPSE 100 Reservoir Engineering Software*.
- Schlumberger. (2020b). Technical Challenges - Carbonate Reservoirs; <https://www.slb.com/technical-challenges/carbonate>.
- Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. In *IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH836)*. Anchorage, Alaska, United States. <https://doi.org/10.1109/ICEC.1998.699146>.
- Shi, Y., & Eberhart, R. (1999). Empirical study of particle swarm optimization. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*. Washington DC, United States. <https://doi.org/10.1109/CEC.1999.785511>.
- Su, S., Gosselin, O., Parvizi, H., and Giddins, M. A. (2013). Dynamic matrix-fracture transfer behavior in dual-porosity models. In *EAGE Annual Conference and Exhibition incorporating SPE Europec*. London, UK. <https://doi.org/https://doi.org/10.2118/164855-MS>.
- Uleberg, K., & Kleppe, J. (1996). Dual porosity, dual permeability formulation for fractured reservoir simulation. In *RUTH Seminar*. Stavanger, Norway. <http://www.ipt.ntnu.no/~kleppe/TPG4150/fracturedpaper.pdf>.
- Van Golf-Racht, T. D. (1982). *Fundamentals of fractured reservoir engineering*. New York, United States: Elsevier Scientific.
- Warren, J. E., & Root, P. J. (1963). The behavior of naturally fractured reservoirs. *Society of Petroleum Engineers Journal*, 3(3), 245–255.
- Wiggins, M. L., & Startzman, R. A. (1990). An approach to reservoir management. In *SPE Annual Technical Conference and Exhibition 1990*. New Orleans, Louisiana, USA. <https://doi.org/https://doi.org/10.2118/20747-MS>.
- Wilamowski, B. M., & Irwin, J. D. (2011). *The industrial electronics handbook second edition: Intelligent system*. Boca Raton, Florida, United States: CRC Press.
- Zhang, Y., Wang, S., & Ji, G. (2015). A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, 2015, 1–38.

## **Paper 3**

### ***Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar



# Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization

Cuthbert Shang Wui Ng<sup>1</sup> · Ashkan Jahanbani Ghahfarokhi<sup>1</sup> · Menad Nait Amar<sup>2</sup>

Received: 20 April 2021 / Accepted: 26 May 2021 / Published online: 17 June 2021  
© The Author(s) 2021

## Abstract

With the aid of machine learning method, namely artificial neural networks, we established data-driven proxy models that could be utilized to maximize the net present value of a waterflooding process by adjusting the well control injection rates over a production period. These data-driven proxies were maneuvered on two different case studies, which included a synthetic 2D reservoir model and a 3D reservoir model (the Egg Model). Regarding the algorithms, we applied two different nature-inspired metaheuristic algorithms, i.e., particle swarm optimization and grey wolf optimization, to perform the optimization task. Pertaining to the development of the proxy models, we demonstrated that the training and blind validation results were excellent (with coefficient of determination,  $R^2$  being about 0.99). For both case studies and the optimization algorithms employed, the optimization results obtained using the proxy models were all within 5% error (satisfied level of accuracy) compared with reservoir simulator. These results confirm the usefulness of the methodology in developing the proxy models. Besides that, the computational cost of optimization was significantly reduced using the proxies. This further highlights the significant benefits of employing the proxy models for practical use despite being subject to a few constraints.

**Keywords** Waterflooding optimization · Machine learning · Artificial neural network · Data-driven proxy modeling · Nature-inspired algorithms

## Introduction

For the past decades, waterflooding or water injection has been one of the most prevalent techniques applied to increase the hydrocarbon production. Waterflooding is termed as the secondary production method that is conducted following the primary production, which is also known as natural depletion. During the phase of natural depletion, hydrocarbon fluid is recovered from the reservoirs by natural forces, such as expansion of fluid and rock, and influx of aquifer. Besides that, tertiary recovery methods, which are known as enhanced oil recovery (EOR), can be another option if secondary recovery is not effective to produce the remaining hydrocarbon. Examples of EOR methods include steam injection and polymer flooding. Pertaining to waterflooding,

due to the costs of water production and injection, it is essential for oil and gas companies to carefully plan the schemes of waterflooding to achieve higher economic returns. Such planning is understood as a part of production optimization which is a very vital aspect in reservoir management (Thakur 1996; Udy et al. 2017). Therefore, optimization of waterflooding has been one of the most widely researched topics in the field of petroleum engineering (Van Essen et al. 2009; Zhang et al. 2014; Ogbewi et al. 2018; Hong et al. 2019).

Fundamentally, waterflooding optimization involves the adjustment of some relevant variables to maximize the pre-defined objective function, like net present value (NPV), total oil production, etc., over a period. Additionally, this period can be at least in the horizon of several years or decades. Hence, it is considered as a long-term optimization problem. More intriguingly, different types of algorithms can perform this optimization as discussed in Udy et al. (2017). In general, these algorithms can be either derivative based or derivative free. Udy et al. (2017) further expounded the benefits and drawbacks of implementing derivative-based algorithm, like adjoint method and derivative-free algorithms,

✉ Cuthbert Shang Wui Ng  
cuthbert.s.w.ng@ntnu.no

<sup>1</sup> Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

<sup>2</sup> Département Etudes Thermodynamiques, Division Laboratoires, Sonatrach, Boumerdes, Algeria

including particle swarm optimization (PSO), simulated annealing (SA), and genetic algorithm (GA). Moreover, waterflooding optimization can in general be categorized into three different types, namely well control optimization (generally comprising either bottom-hole pressure (BHP) or rates optimization) (Sarma et al. 2008; Zhang et al. 2014; Lu et al. 2017), well placement optimization (Guyaguler et al. 2002; Forouzanfar and Reynolds 2013; Volkov and Bellout 2017), and combination of these methods together or with other variables, such as number of wells (Bellout et al. 2012; Forouzanfar and Reynolds 2014; Pouladi et al. 2020). In this context, the potential of waterflooding optimization for continuous improvement for more practical applications has been demonstrated.

Numerical reservoir simulation (NRS) is one of the most standardized tools utilized in the oil and gas industry to conduct the subsurface or reservoir modeling. NRS can be conveniently (and is also frequently) coupled with any mathematical algorithm to optimize waterflooding or any EOR techniques. This has also been one of the most common practices in the industry as highlighted in some literatures (Peaceman 1977; Jansen et al. 2009; Ertekin and Sun 2019; Baumann et al. 2020). Nonetheless, as perceived, NRS is developed based upon the physics to model the behavior of fluid flow in porous media. Therefore, when the system modeled becomes more complex, e.g., increased heterogeneity of the reservoir, the transport of fluid in porous media will be more difficult to be solved mathematically (Mohaghegh 2017a). This implies that the time required to complete the computation of NRS will increase drastically. Consequently, this might lead to certain level of economic loss. Fortunately, thanks to the establishment of proxy modeling, the computational challenge can be mitigated. In this aspect, the word “proxy” denotes “to act on behalf of another.” This denotes that proxy models are the replica of numerical reservoir models which can be readily employed for practical applications in the industry (Mohaghegh 2011; Ertekin and Sun 2019).

With respect to this, proxy models are alternatively known as data-driven models because their building blocks are made up of different sets of data. Hence, proxy models are believed to be able to replicate the results of NRS accurately if the data used to develop them are representative of the physics being modeled. As Mohaghegh (2017b) has counseled, there are two main classes of proxy modeling, which are reduced-order models (ROMs) and response surface models (RSMs). For ROMs, the simplification of the physics is involved and one of the most used examples of ROMs is capacitance resistance models (CRMs). CRMs were developed by Bruce (1943) and reintiated by Yousef et al. (2006) to determine inter-well connectivity. Application of CRMs in waterflooding has also been proven to be useful in some literatures (Liang et al. 2007; Sayarpour et al.

2007; Hong et al. 2017). Besides that, RSMs are considered as statistical approaches which attempt to develop a pre-defined form of mathematical function, e.g., linear, polynomial, etc., based on the data given (Mohaghegh 2017b). There are also some papers (Valladão et al. 2013; Babaei and Pan 2016) that discuss the use of RSMs in waterflooding. Despite this, Mohaghegh (2017a, b) has opined that these classes of proxy modeling involve underlying assumptions and simplifications that can impede capturing the actual physics from pattern recognition of data provided. Thus, he has coined another class of proxy modeling that is built based upon machine learning (ML) techniques and artificial intelligence (AI), which has been named as “smart proxy modeling” (SPM). The word “smart” indicates the ability of the models to learn the pattern of the data provided through the ML and AI techniques. He has also initialized the term “Petroleum Data Analytics” (PDA) that focuses on the use of data-driven analytics and big data in the upstream of petroleum industry (Mohaghegh 2017a, b) and SPM is undeniably a part of PDA. According to Mohaghegh (2017b), smart proxy models consist of an ensemble of neuro-fuzzy systems that can duplicate the results yielded by NRS and readily to be utilized for different purposes, like history matching (He et al. 2016; Shahkarami et al. 2018; Shahkarami and Mohaghegh 2020), uncertainty quantification (Mohaghegh 2006; Mohaghegh et al. 2006, 2012), utilization of CO<sub>2</sub> (Shahkarami et al. 2014; Amini and Mohaghegh 2019; Vida et al. 2019; Shahkarami and Mohaghegh 2020), waterflooding (Alenezi and Mohaghegh 2017), and analysis of shales (Kalantari-Dahaghi and Mohaghegh 2011; Mohaghegh 2013; Mohaghegh et al. 2017). Also, it is very important to understand that SPM is an objective-directed task in which the purpose of the proxies needs to be notified first prior to development. Having this understanding will help the modelers to have a better idea of what data can be useful in the development of proxy models.

There are also other interesting literatures (Nait Amar et al. 2018, 2020; Navrátil et al. 2019; Alakeely and Horne 2020; Ng et al. 2021) that discuss and present the use of ML methods in the establishment of proxies of numerical models in petroleum domain, especially for reservoir engineering. Regarding this, there is a riveting insight being provided by Nait Amar et al. (2018) about the modeling of proxies, which is the difference between static and dynamic proxy models. They discussed that in static proxies, the models were not developed as the function of time. Hence, these models were built to yield the results of a predefined variable, such as NPV and total oil production at a particular time (normally at the end of simulation). In this context, Guo and Reynolds (2018) applied support vector regression (SVR) to build a static proxy model that predicted the NPV as a function of control sets by considering different geological realizations. Then, the static proxy was maneuvered to

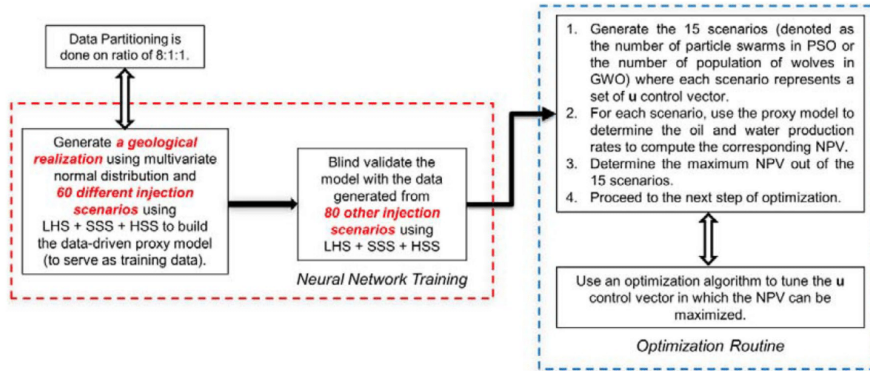


Fig. 1 General workflow of the methodology of data-driven proxy modeling and optimization

perform robust production optimization. Moreover, Wang et al. (2021) presented the application of PSO in tuning the hyperparameters of SVR that was developed as static proxies to forecast the NPV and cumulative oil production. Thereafter, this PSO–SVR model was coupled with non-dominated sorting genetic algorithm-II (NSGA-II) to conduct the Pareto optimization. Albeit the use of static proxies has been successfully shown, they articulated that the dynamic proxies (established as a function of time) offered more practical applicability and flexibility to be used, notably under time-dependent constraints (Nait Amar et al. 2018). In the context of waterflooding, Golzari et al. (2015) applied artificial neural network (ANN) modeling to build a dynamic proxy and coupled it with GA to optimize the production. They also integrated cross-validation and Jackknife Variance to evaluate the quality of the proxies and perform adaptive sampling to add new training data if necessary. Also, Teixeira and Secchi (2019) employed ANN to develop two dynamic proxies: one that could forecast the oil production rates as a function of injection rates and past oil rates and another one was to approximate the same output by having injection rates and BHP of producers as inputs.

In this paper, one of the goals is to present how dynamic proxy models can be developed based upon the data generated by the NRS models. There are two different NRS models, 2D and 3D reservoirs, being analyzed in this work. The purpose of the proxies is to be employed to carry out the well control optimization. About the proxy modeling, the ML technique that has been applied is ANN and the corresponding training algorithm is adaptive moment estimation (Adam). Furthermore, we couple these proxies with two different nature-inspired metaheuristic algorithms, namely particle swarm optimization (PSO) and grey wolf optimization (GWO) to run the respective optimization. These algorithms would also be utilized with the NRS models for comparative

analysis. After this introduction, we will explicate the formulation of the optimization problem and the methodology used to establish the proxies. In this aspect, we also provide brief discussion about ANN, PSO, and GWO. Thereafter, we explain the background of the reservoir models and illustrate the respective results of the ANN training as well as the optimization study. The discussion will then follow prior to proceeding to the conclusions.

### Methods

The entire workflow utilized to build and apply the data-driven models for the optimization of waterflood is summarized in Fig. 1. The workflow can be classified into two main parts, which include neural network training (also known as proxy modeling) and optimization routine. Prior to developing the data-driven proxies, it is essential to identify the purpose of these models as proxy modeling is an objective-directed task. In this paper, the objective is to maximize the NPV of a waterflooding project by adjusting the control of injection rate of each well periodically (every 150 days) over 3000 days. Besides that, the control of each injector is tuned within the range of 40 m<sup>3</sup>/day and 100 m<sup>3</sup>/day.

The NPV is expressed as shown in Eq. (1). Since the reservoir models presented in this paper are only oil–water systems, gas production rate is not considered in the formulation of NPV.

$$NPV(\mathbf{u}) = \sum_{j=1}^{n_{total}} \frac{(Q_o^j(\mathbf{u})P_o - Q_w^j(\mathbf{u})C_w - Q_{wi}^j(\mathbf{u})C_{wi}) \times \Delta t_j}{(1 + b)^{j/D}} \tag{1}$$

$$\mathbf{u} = [u_1, u_2, u_3, \dots, u_M]^T \tag{2}$$

where  $\mathbf{u}$  is the control vector (e.g., control rates or BHP),  $M$  is the number of control variables,  $Q_o^j$  is the field oil production rate at timestep  $j$ ,  $Q_w^j$  is the field water production rate at timestep  $j$ ,  $Q_{wi}^j$  is the field water injection rate at timestep  $j$ ,  $\Delta t_j$  is the time difference between timestep  $j$  and previous timestep,  $t_j$  is the cumulative time until timestep  $j$  that is used to discount the cashflow, and  $D$  is the reference period for discounting. In this paper,  $D$  is set to be 365 days because interest rate,  $b$  is in the unit of fraction per year and the cashflow is discounted every day.  $P_o$ ,  $C_w$ , and  $C_{wi}$  correspondingly mean oil price, cost of water production, and cost of water injection. According to Eq. (1), there are two important parameters we aim to obtain, either directly or indirectly, from the proxy models. These parameters are field oil and water production rates. Based on our analysis and investigation, we established two different proxy models, where one could predict the field liquid production rates at a specific timestep whereas the other one could estimate the field water cut at a particular timestep.

For both proxies, the input variables include the number of days at each timestep  $j$ ,  $t_j$ ; the harmonic mean of grid absolute permeability for each reservoir layer,  $\bar{k}$ ; the standard deviation of grid absolute permeability for each reservoir layer,  $k_{SD}$ ; the permeabilities of perforated grid blocks (injectors),  $k_{injector}$ ; the permeabilities of perforated grid blocks (producers),  $k_{producer}$ ; the field water injection rate (control vector); the output at the previous timestep,  $y_{j-1}$ . The mathematical formulation of the proxies<sup>1</sup> built in this paper in general can be expressed as Eq. (3). Besides that, the harmonic mean of permeability for each reservoir layer is presented as Eq. (4). Nonetheless, regarding the input variables of the permeabilities of perforated grid blocks (producers and injectors), they are case-dependent in this paper. This implies that we have applied different approaches of formulation to incorporate them as parts of the inputs relying upon the reservoir models investigated. It will be discussed in detail later. In this work, the data-driven models were represented as the ANNs. The topologies of the ANNs developed here will be divulged in the next section.

$$y_j = f\left(t_j, \bar{k}, k_{SD}, k_{injector}, k_{producer}, \mathbf{u}, y_{j-1}\right) \quad (3)$$

$$\bar{k} = \frac{\sum_{i=1}^n L_i}{\sum_{i=1}^n \frac{L_i}{k_i}} \quad (4)$$

<sup>1</sup> The permeability used here is the horizontal permeability. The data of the vertical permeability can also be included in the development of the proxies here. However, it has not been considered in this paper as the current formulation already yielded very good results.

where  $n$  is the number of grid blocks,  $L_i$  is the depth at the top of grid block  $i$ , and  $k_i$  is the grid absolute permeability.

About the development of the proxies, the first step is to generate the spatiotemporal database. To develop the database, we apply three different sampling techniques, which are Latin Hypercube sampling (LHS), Sobol Sequence sampling (SSS), and Hammersley Sequence sampling (HSS) to generate 60 sets of samples of control rates (each set consists of 20 injection rates which corresponds to one injection scenario). Respectively, peruse McKay et al. (1979), Sobol' (1967), and Hammersley and Handscomb (1964) for more information about LHS, SSS, and HSS. Each sampling method is implemented to, respectively, generate 20 sets of samples. Thereafter, each set would be fed into the reservoir simulator to yield the reservoir responses. This denotes that 60 reservoir simulations are run in total. After finishing the simulations, we extract the dynamic inputs and combined them with the static inputs to create the spatiotemporal database. It is of paramount importance to have the database normalized and arrange in a consistent format before it is supplied to the neural network for training. The fundamental ideas of the neural network training will be delineated later. Before the neural network training commences, the normalized database is partitioned into three different groups, namely training, validation, and testing, based on a ratio of 8:1:1. In this case, only the training data is used to build the data-driven models. However, after each epoch (iteration) of training, validation data would be simultaneously fed into the neural network to elude the issue of overfitting (Mohaghegh 2017a; Shahkarami and Mohaghegh 2020). A healthy training can be ensured by having the simultaneous decreasing trends of the training and validation errors as shown in Fig. 2. After the training is completed, the testing data would be used to evaluate the predictability of the models.

Upon the completion of these three stages, the data-driven proxies ought to undergo the blind validation before being practically employed. The data used in blind validation should not be part of the above-mentioned spatiotemporal database. Therefore, to conduct the blind validation, we utilize LHS, SSS, and HSS to generate other 80 sets of samples of control rates. Then, 80 reservoir simulations are run to produce the outputs of field liquid production rates and field water cut. These outputs are compared with the predicted outputs yielded by the proxy models. Only when the comparative study shows excellent results, we can safely infer that the proxy models can practically be employed. By having successfully established these two proxies, the field oil and water production rates required for the optimization purpose can be obtained. In this paper, we implemented PSO and GWO to optimize the well control. The information about the algorithms and the parameters used to carry out the optimization will be presented later. We did not only

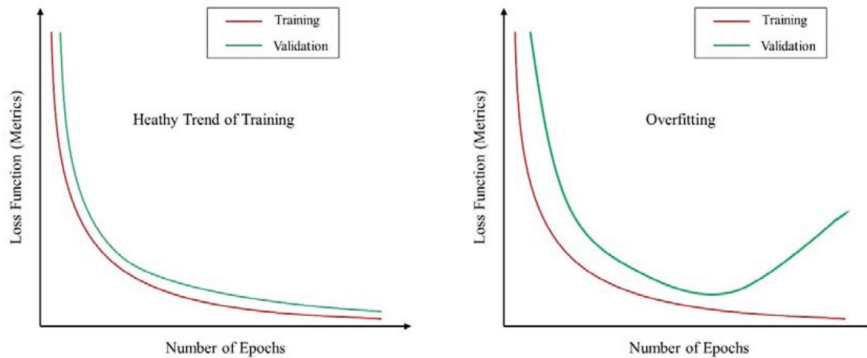


Fig. 2 Comparison between healthy trend of training and overfitting issue. Adapted from Shahkarami and Mohaghegh (2020)

couple these algorithms with the proxies developed here, but also applied them along with the numerical reservoir simulation. The optimal well controls resulted from the two approaches were then compared. Additionally, the proxy-optimized well controls were fed into the reservoir simulator to yield the results that could be used to further illustrate the robustness of the proxy models.

### Artificial neural network

ANN is a famous ML method that is established based on the inspiration from the working process of the biological neural networks in human brains. ANN consists of a lot of computing elements which are termed as nodes or artificial neurons. It has been proven to be useful and effective in capturing and learning the sophisticated relationship between input and output data derived from any physical process as in traditional regression approaches. Examples of ANNs include feedforward neural network (FNN), convolutional neural network, recurrent neural network, radial basis function networks, and adaptive neuro-fuzzy inference system. Different types of activation functions can also be employed to develop an ANN and the common ones are the sigmoid function, the hyperbolic tangent, and the rectified linear unit (ReLU) function (Buduma and Locascio 2017).

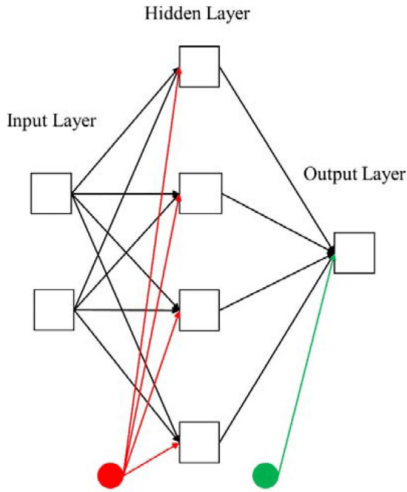
In this paper, FNN with the ReLU function as its activation function was utilized. In general, FNN, also called multilayer perceptron (MLP), has three layers, e.g., the input layer, the hidden layer, and the output layer. To guarantee that the MLP can study the relationship between input and output data provided, it must undergo the training stage. During the training stage, the learning ability of the MLP is achieved by adjusting the sets of weights and biases to reduce the predefined loss function, including mean squared error (MSE) and mean absolute percentage error. In this

paper, MSE was chosen as the loss function. Such optimization is generally conducted through the backpropagation (BP) approaches. These methods involve the application of different derivative-based algorithms, for instance, steepest descent gradient, the Levenberg–Marquardt algorithm, the Powell–Beale conjugate gradient, and Adam. In this work, Adam was applied as the training algorithm. For the details of Adam, refer to this literature (Kingma and Ba 2015). The relevant parameters used for the training of all the neural network proxies in this paper are tabulated in Table 1.

Prior to entering the MLP, the data have to be normalized to improve the training performance of the MLP as recommended in Hemmati-Sarapardeh et al. (2020). In this paper, we used Eq. (5) to normalize the data between 0 and 1. After normalization of data, the forward propagation of the input data will happen to compute the outputs. The resulting output data will thereafter be compared with the actual output data to determine the errors. After this, the errors are propagated back through the MLP to iteratively tune the weights and biases to reach the optimal point. The architecture of an arbitrary FNN is demonstrated in Fig. 3 in which the red node acts as the bias node between the input and hidden layers whereas the node between the hidden and output layers is shown in green.

**Table 1** Parameters used to conduct neural network training using Adam

Adam parameters	Values
Number of iterations (epochs)	2000
Learning rate	0.001
Exponential decay rates for the 1st moment estimates, $\beta_1$	0.9
Exponential decay rates for the 2nd moment estimates, $\beta_2$	0.999
Numerical stability constant, $\epsilon$	$10^{-7}$



**Fig. 3** The structure of a simple FNN model

$$x_{norm} = \frac{x_j - x_{min}}{x_{max} - x_{min}} \tag{5}$$

where  $x_{norm}$  represents the normalized data point,  $x_j$  refers to a data point,  $x_{min}$  corresponds to the data point with the lowest value, and  $x_{max}$  is the data point with the highest value. To assess the quality of the prediction done by the proxies, we used coefficient of determination,  $R^2$  as the performance metrics. The respective formula is shown in Eq. (6).

$$R^2 = 1 - \frac{\sum_{j=1}^N (y_j^{real} - y_j^{pred})^2}{\sum_{j=1}^N (y_j^{pred} - \bar{y}^{real})^2} \tag{6}$$

where  $y_j^{real}$  means the actual data point,  $y_j^{pred}$  denotes the predicted data point,  $\bar{y}^{real}$  refers to the mean of all the actual data points, and  $N$  is the total number of data points. As explained, we have built four neural network proxies in this paper, two for each of the reservoir models studied. The topologies of the neural proxies are presented in Table 2 for 2D reservoir model and Table 3 for 3D reservoir model. These architectures were determined via the trial and error approach.

**Particle Swarm Optimization**

The PSO algorithm is one of the most popular swarm-based metaheuristic algorithms that has been initiated by Kennedy and Eberhart (1995) through simulating the social habit of

**Table 2** The architecture of ANN for 2D Reservoir Model

Layers	Field liquid production rate		Field water cut	
	Number of layers	Number of nodes	Number of layers	Number of nodes
Input	1	23	1	23
Hidden	1	100	2	50
Output	1	1	1	1

**Table 3** The architecture of ANN for 3D Reservoir Model

Layers	Field liquid production rate		Field water cut	
	Number of layers	Number of nodes	Number of layers	Number of nodes
Input	1	29	1	29
Hidden	1	100	2	50
Output	1	1	1	1

flying birds. Mathematically, this flock of birds is represented as a population of particles known as a swarm of particles. Each particle indicates a potential position (solution) in a search space and it is updated iteratively according to its position and velocity at previous iteration step. The motions of the particles are regulated by their own most optimal position (the local best position) and their most optimal position in the entire swarm (the global best position). After some iterations, the convergence of the particles in the swarm to an optimal point (the best solution) will occur. The position and velocity of the  $j^{th}$  particle in a  $k$  dimensional space at step  $t$  are formulated as follows:

$$x_{j,t} = \{x_{j1,t}, x_{j2,t}, x_{j3,t}, \dots, x_{jk,t}\} \tag{7}$$

$$v_{j,t} = \{v_{j1,t}, v_{j2,t}, v_{j3,t}, \dots, v_{jk,t}\}. \tag{8}$$

Thereafter, the velocity of each particle at next step  $t + 1$  is updated based on Eq. (9) and the position of a particle at the next iteration  $t + 1$  is updated by using Eq. (10).

$$v_{jk,t+1} = \omega v_{jk,t} + c_1 r_1 (pbest_{jk,t} - x_{jk,t}) + c_2 r_2 (gbest_{k,t} - x_{jk,t}) \tag{9}$$

$$x_{jk,t+1} = x_{jk,t} + v_{jk,t+1} \tag{10}$$

where  $v_{jk,t}$  and  $x_{jk,t}$  indicate the velocity of the  $j$ th particle at step  $t$  and its corresponding position in the  $k$ th dimension quantity, respectively. Apart from this,  $pbest_{jk,t}$  refers to the  $k$ th dimension quantity of the individual  $j$  at the local best position at iteration  $t$ .  $gbest_{k,t}$  is the  $k$ th dimension quantity of the swarm at the global best position at iteration  $t$ .  $c_1$  and  $c_2$ , respectively, denote the cognitive and social learning factors.  $\omega$  is known as the inertial weight which was introduced



by Shi and Eberhart (1998) to enhance the convergence condition.  $r_1$  and  $r_2$  are randomly retrieved between 0 and 1. In terms of the minimization problem, a cost function  $f$  (to be minimized) is defined. Then, to find out the local best solution at  $t + 1$ , Eq. (11) is used. To determine the global optimal solution at  $t + 1$ , Eq. (12) is applied.

$$pbest_{jk,t+1} = \begin{cases} pbest_{jk,t}, & \text{if } f(pbest_{jk,t}) \leq f(x_{jk,t+1}) \\ x_{jk,t+1}, & \text{otherwise} \end{cases} \quad (11)$$

$$gbest_{k,t+1} = \min[f(pbest_{jk,t+1})]. \quad (12)$$

The procedure described above is repeated until the stopping condition is satisfied. During the optimization process, 15 particle swarms were initially generated, 100 iterations were run, and the values of  $\omega$ ,  $c_1$ , and  $c_2$  were, respectively, set to be 0.8, 1.1, and 1.1.

### Grey wolf optimization

The GWO is another well-known metaheuristic algorithm that was established by Mirjalili et al. (2014). This algorithm was developed in accordance with the natural inspiration derived from the social hierarchy of leadership and hunting style of grey wolves (Mirjalili et al. 2014). Pertaining to the paradigm of this algorithm, it is essential to recognize that the population of grey wolves is divided into four different classes, such as alpha ( $\alpha$ ), beta ( $\beta$ ), delta ( $\delta$ ), and omega ( $\omega$ ). Based upon the social hierarchy,  $\omega$  wolves are the lowest among others and they are preceded by  $\delta$ ,  $\beta$ , and  $\alpha$ . To mathematize the mechanism of GWO, a population of wolves is expressed as a set of random solutions. The fitness value of this set of solutions is then calculated and assessed by applying a predefined objective function (Xu et al. 2020). After that, the wolves' populations are divided into the four previously stated classes based on the computed fitness value. When the optimization takes place, the three most optimal wolves:  $\alpha$ ,  $\beta$ , and  $\delta$ , would eventually guide the other  $\omega$  wolves toward the prey that acts as the global solution in the search space. This procedure is carried out via the iterative update of the positions of the wolves as shown below:

$$\bar{D} = \left| \bar{C} \cdot \bar{X}_p(t) - \bar{X}(t) \right| \quad (13)$$

$$\bar{X}(t+1) = \left| \bar{X}_p(t) - \bar{A} \cdot \bar{D} \right| \quad (14)$$

$$\bar{A} = 2\bar{a} \cdot \bar{r}_1 - \bar{a} \quad (15)$$

$$\bar{C} = 2\bar{r}_2 \quad (16)$$

where  $t$  means the current iteration step,  $\bar{X}$  implies the position of a grey wolf,  $\bar{X}_p$  is the position of the prey,  $\bar{a}$  is normally lowered from 2 to 0. Also,  $\bar{r}_1$  and  $\bar{r}_2$  are the random vectors between 0 and 1. In GWO, the position of the prey (the global optimal solution) is not exactly known. Hence, it is assumed that the positions of  $\alpha$ ,  $\beta$ , and  $\delta$  are considered as the optima. Then, the other  $\omega$  wolves re-calibrate their positions with respect to those of  $\alpha$ ,  $\beta$ , and  $\delta$  as follows:

$$\bar{D}_\alpha = \left| \bar{C}_1 \cdot \bar{X}_\alpha(t) - \bar{X}(t) \right| \quad (17)$$

$$\bar{D}_\beta = \left| \bar{C}_2 \cdot \bar{X}_\beta(t) - \bar{X}(t) \right| \quad (18)$$

$$\bar{D}_\delta = \left| \bar{C}_3 \cdot \bar{X}_\delta(t) - \bar{X}(t) \right| \quad (19)$$

where  $\bar{X}_\alpha(t)$  corresponds to the position of  $\alpha$  wolves at step  $t$ ,  $\bar{X}_\beta(t)$  is the position of  $\beta$  wolves at step  $t$ , and  $\bar{X}_\delta(t)$  represents the position vector of  $\delta$  wolves at iteration  $t$ .  $\alpha$ ,  $\beta$ , and  $\delta$  wolves will then update their positions at iteration  $t + 1$  based on Eqs. (20), (21), and (22). The position of the solution at step  $t + 1$  is thereafter determined based upon Eq. (23).

$$\bar{X}_1 = \left| \bar{X}_\alpha(t) - \bar{A}_1 \cdot \bar{D}_\alpha \right| \quad (20)$$

$$\bar{X}_2 = \left| \bar{X}_\beta(t) - \bar{A}_2 \cdot \bar{D}_\beta \right| \quad (21)$$

$$\bar{X}_3 = \left| \bar{X}_\delta(t) - \bar{A}_3 \cdot \bar{D}_\delta \right| \quad (22)$$

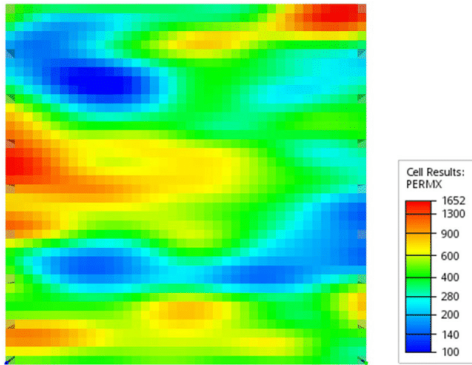
$$\bar{X}(t+1) = \frac{\bar{X}_1 + \bar{X}_2 + \bar{X}_3}{3}. \quad (23)$$

These steps are repeated until the stopping condition is met. During the optimization process, 15 populations of grey wolves were initially generated, and 100 iterations were run.

## Results

### Case study 1: 2D reservoir model

We first illustrate the development of a data-driven proxy model of a 2D heterogeneous and 2-phase (water and black oil) reservoir model. The heterogeneity only applies to the permeability in this case study. Besides that, the horizontal permeabilities in both  $x$  and  $y$  directions are assumed to be the same whereas the vertical permeability is set to be 10 times smaller. Also, homogeneity applies to porosity and it is assumed to be 0.4. Regarding the size of the grid blocks,



**Fig. 4** The overview of the 2D reservoir model. The color bar indicates the values of horizontal permeability in x-direction in the units of millidarcy (mD)

it is  $10 \text{ m} \times 10 \text{ m} \times 10 \text{ m}$  and the total number of grid blocks is  $40 \times 40 \times 1$ . Therefore, the dimension of the whole model is  $400 \text{ m} \times 400 \text{ m} \times 10 \text{ m}$ . Its top is at the depth of 1500 m. Pertaining to the configuration of well, there are only two wells being drilled, namely one horizontal injector and one horizontal producer. The injector is drilled at the left edge whereas the producer is placed at the right edge. The 2D reservoir model is illustrated in Fig. 4. As briefly discussed, the performance of the injector is controlled by the rate within the range of  $40 \text{ m}^3/\text{day}$  and  $100 \text{ m}^3/\text{day}$  whereas the producer is controlled by the BHP with the lower limit of 180 bar. Regarding the perforation in the x-direction, the injector is perforated at the 1<sup>st</sup> grid block whereas the producer is perforated at the 40th grid block. However, for y-direction, both wells are completed at 1st, 5th, ..., 35th, 40th grid blocks. Permeabilities of these grid blocks (constituting 18 variables in total) are directly retrieved and used as the input parameter for neural network training. The numerical simulation is performed using ECLIPSE 100 software Schlumberger.

After running the required numerical reservoir simulations and extracting the input and output data, the neural network training was correspondingly performed on the data-driven proxies of field liquid rate and field water cut by using the specification listed in Table 2. Based on Eq. (3) and Table 2, there are 23 input parameters applied to train the proxies. The performances of training, validation, and testing of both proxies are evaluated by using the coefficient of determination,  $R^2$ , and shown in Table 4. Besides that, for the blind validation phase, the proximity of the actual and targeted outputs is assessed by applying 80 injection schedules. Thereafter, the mean of the respective coefficient of determination is calculated for each proxy and tabulated in Table 5. For illustration purpose, only the result for a

**Table 4**  $R^2$  of training, validation, and testing results of the data-driven proxies

Dataset	Field liquid production rate	Field water cut
Training	0.9999	0.9999
Validation	0.9999	0.9999
Testing	0.9999	0.9999

**Table 5** Mean  $R^2$  of blind validation of proxies based on different sampling techniques

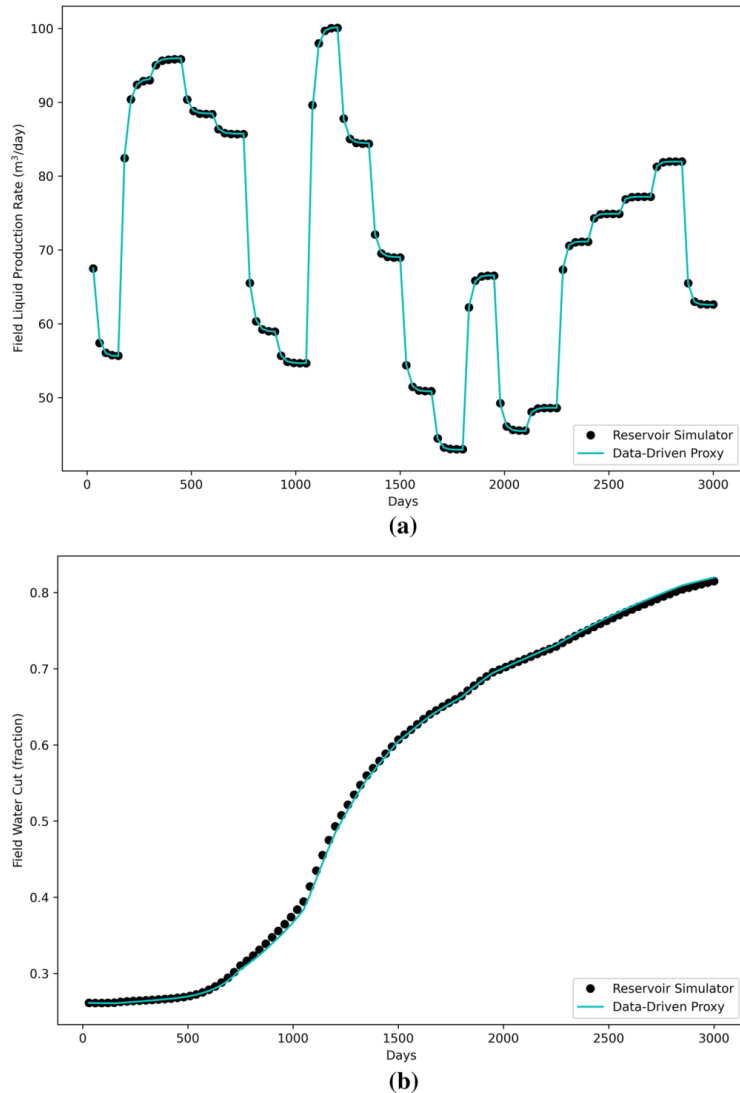
Sampling methods	Field liquid production rate	Field water cut
LHS	0.9999	0.9995
SSS	0.9999	0.9995
HSS	0.9999	0.9995

randomly selected injection schedule of blind validation (out of 80) is demonstrated for each sampling method. In this case, the comparison between the actual and the predicted field liquid production rates (also field water cuts) is, respectively, plotted as shown in Fig. 5 for LHS, in Fig. 6 for SSS, and in Fig. 7 for HSS. According to these results, it is inferred that these proxy models are ready for practical application.

In this aspect, we defined the economic parameters as depicted in Table 6 to be used in the optimization process. As mentioned earlier, PSO and GWO would be employed to conduct the optimization of NPV. The optimized controls of field water injection rates are, respectively, illustrated in Fig. 8 for PSO and in Fig. 9 for GWO. Pertaining to this, the resulted optimal NPV of three different scenarios is demonstrated in Table 7 in which Scenario 1 represents the optimization by only using the reservoir simulator ( $\text{NPV}_{\text{sim}}$ ), Scenario 2 denotes the optimal NPV obtained by feeding the proxy-optimized control into the simulator ( $\text{NPV}_{\text{sim-proxy}}$ ), and Scenario 3 means the optimization by only using the proxies ( $\text{NPV}_{\text{proxy}}$ ). Pertaining to the optimal NPVs yielded from three different scenarios, it can be noted that the data-driven proxies have in general overestimated the optimal NPV for both algorithms. However, the absolute percentage error between  $\text{NPV}_{\text{proxy}}$  and either  $\text{NPV}_{\text{sim}}$  or  $\text{NPV}_{\text{sim-proxy}}$  is miniscule.

For PSO, the absolute percentage error between  $\text{NPV}_{\text{sim}}$  and  $\text{NPV}_{\text{proxy}}$  is around 0.14%. This shows that when the data-driven proxies are coupled with PSO, they can yield reasonable results to approximate the NPV calculated with the results from simulator. Furthermore, to understand whether the proxies produce accurate results for the calculation of NPV, the absolute percentage error between  $\text{NPV}_{\text{sim}}$  and  $\text{NPV}_{\text{sim-proxy}}$  is found out to be about 0.13%. Additionally, the absolute percentage error between  $\text{NPV}_{\text{sim-proxy}}$  and

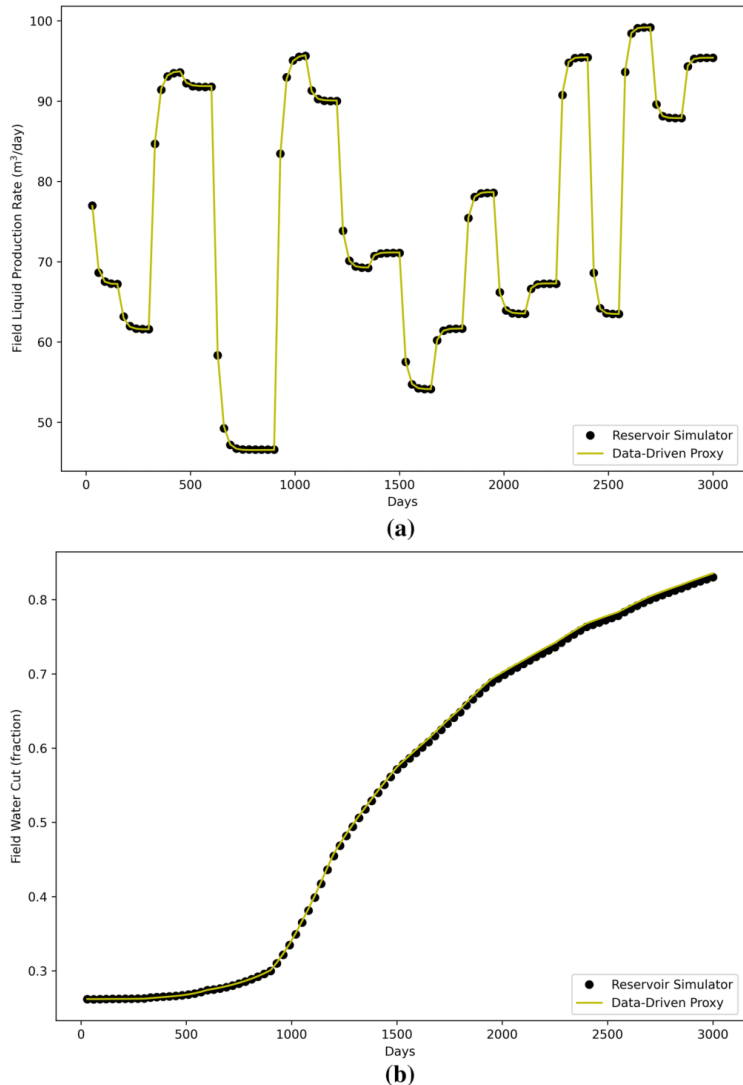
**Fig. 5** Results of blind validation of data-driven proxies of LHS sample set 11 (out of 80). **a** Field liquid production rate. **b** Field water cut



$NPV_{proxy}$  is 0.27%. This proves the reliability of the developed proxies. Thereafter, for GWO, the absolute percentage error between  $NPV_{sim}$  and  $NPV_{proxy}$  is 0.34% while it is 0.40% between  $NPV_{sim}$  and  $NPV_{sim-proxy}$ . Also, the absolute percentage error between  $NPV_{sim-proxy}$  and  $NPV_{proxy}$  is 0.74%. Despite the higher accuracy of optimization results portrayed by PSO, it can be noted that GWO generally performs better than PSO in the context of optimization in this

case study. To further demonstrate the high proximity of the data-driven models, the plots of PSO-optimized and GWO-optimized field water production rates of Scenario 2 against Scenario 3 are correspondingly illustrated in Figs. 10 and 11.  $R^2$  obtained for Fig. 10 is 0.9998 whereas that of Fig. 11 is 0.9989. The similar plots for field oil production rates are presented for PSO in Fig. 12 and GWO in Fig. 13. Then, the values of  $R^2$  calculated for Figs. 12 and 13 are 0.9999.

**Fig. 6** Results of blind validation of data-driven proxies of SSS sample set 32 (out of 80). **a** Field liquid production rate. **b** Field water cut

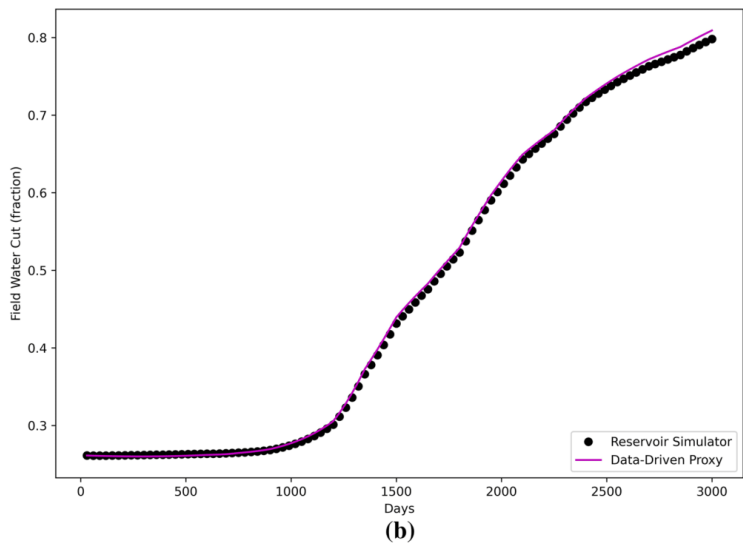
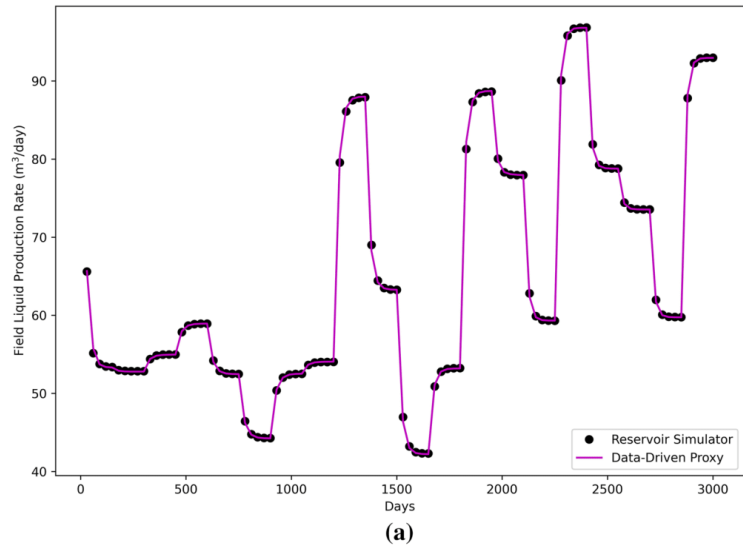


### Case study 2: 3D reservoir model (Egg Model)

To further demonstrate the methodology of proxy modeling proposed in this paper, we present the use of a more sophisticated reservoir model as another case study. The model was initiated by Jansen et al. (2014) and termed as “Egg Model.” It was employed as case study in several papers

(Van Essen et al. 2009; Hong et al. 2017). In general, it is considered as a channelized depositional model where the heterogeneity only pertains to permeability. However, porosity is homogeneous and set to be 0.2. Besides that, the initial water saturation is 0.1 and it applies to all grid blocks. The size of the grid blocks is 8 m × 8 m × 4 m and the total number of grid blocks is 60 × 60 × 7. However, the

**Fig. 7** Results of blind validation of data-driven proxies of HSS sample set 69 (out of 80). **a** Field liquid production rate. **b** Field water cut

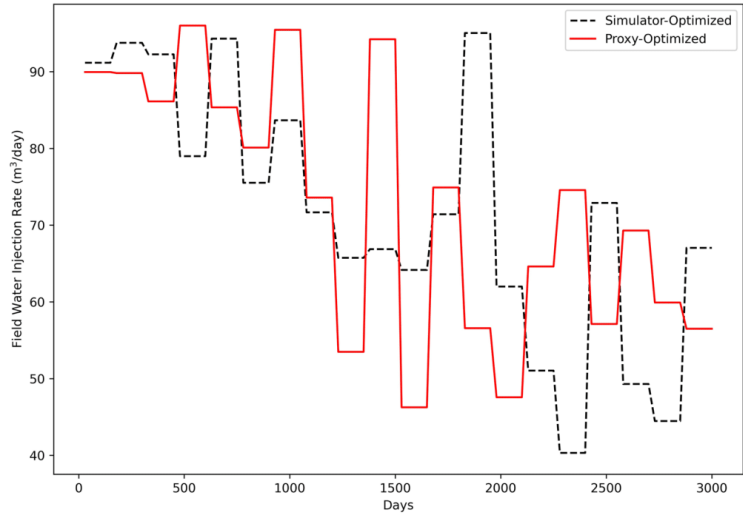


total number of active grid blocks is 18533. The horizontal permeability map of Egg Model is illustrated in Fig. 14. The details of the geological properties of this model can be found in Jansen et al. (2014). Besides that, the model comprises eight vertical injectors and four vertical producers. The only modification done on the Egg Model to fulfill the need of the analysis here is changing the control of injectors. Since there are eight injectors and each of them is controlled

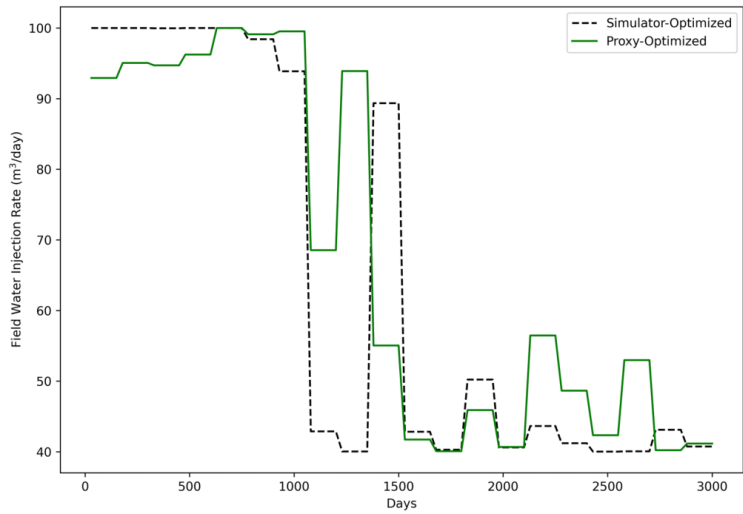
by the rate within the range of  $40 \text{ m}^3/\text{day}$  and  $100 \text{ m}^3/\text{day}$ , the field injection rates are altered between  $320 \text{ m}^3/\text{day}$  and  $800 \text{ m}^3/\text{day}$ . Regarding the four producers, each of them is controlled by the BHP with the lower limit of 395 bar.

About the completion, all the wells are perforated in seven layers. If we apply the formulation presented in the case study of 2D reservoir model to include the grid block permeability as the input variables, then this will result in 84

**Fig. 8** Optimized control rates by using simulator and data-driven proxies with the implementation of PSO



**Fig. 9** Optimized control rates by using simulator and data-driven proxies with the implementation of GWO



**Table 6** Economics parameters used for NPV calculation

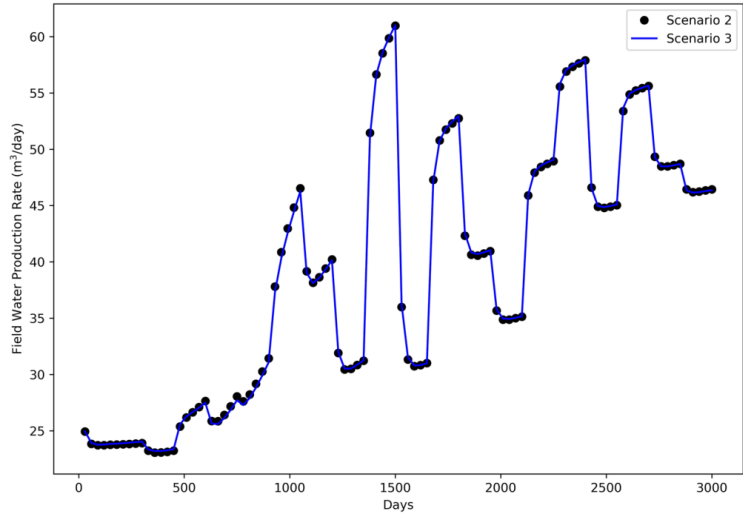
Variables	Values	Units
Oil price, $P_o$	314.50	USD/m <sup>3</sup>
Cost of produced water, $C_w$	37.50	USD/m <sup>3</sup>
Cost of injected water, $C_{wi}$	37.50	USD/m <sup>3</sup>
Discount rate	0.10	per year

variables of  $k_{injector}$  and  $k_{producer}$ . Thus, for practical purpose of eluding the curse of dimensionality, we determined the arithmetic mean of the permeability of the perforated grid blocks for each well. This could reduce the number of permeability variables from 84 to 12. Given there are 7 layers in this Egg Model, there will be a total of 14 variables of  $\bar{k}$  and  $k_{SD}$ . According to Eq. (3), there are 29 input variables to be included to train the neural network. By employing the same

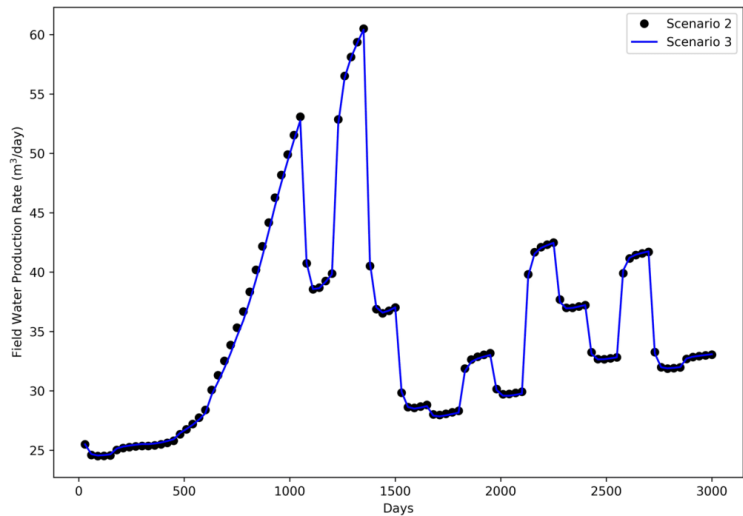
**Table 7** Optimization results of three scenarios for PSO and GWO

Optimization Algorithm	Scenario 1		Scenario 2		Scenario 3	
	GWO	PSO	GWO	PSO	GWO	PSO
NPV <sub>optimal</sub> (million USD)	16.52	16.26	16.46	16.24	16.58	16.29

**Fig. 10** Plot of PSO-optimized field water production rates, comparison between Scenarios 2 and 3



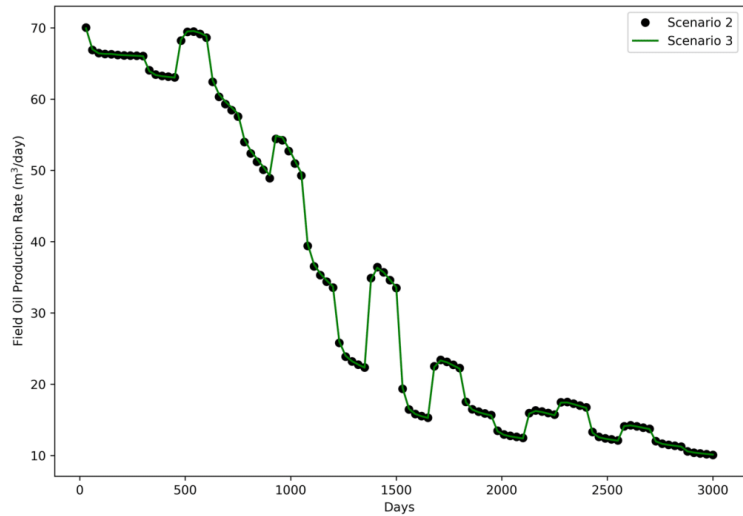
**Fig. 11** Plot of GWO-optimized field water production rates, comparison between Scenarios 2 and 3



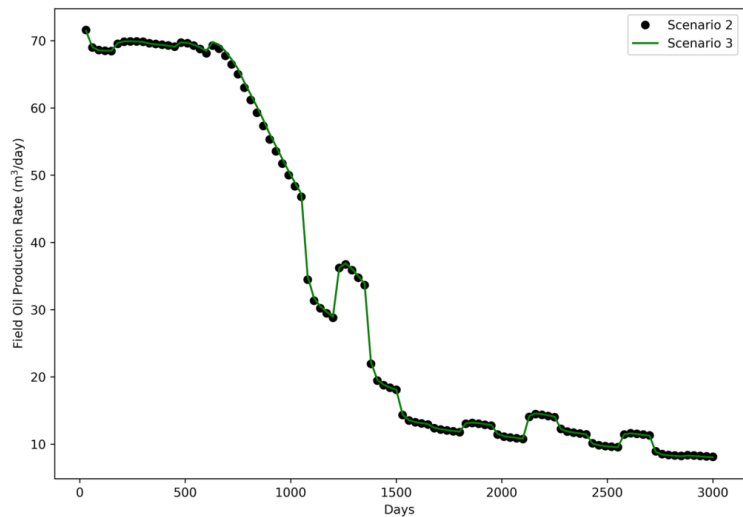
methodology as explained earlier and the specifications presented in Table 3, the neural network training is conducted.

Performance metrics of training, validation, and testing of the proxies of the Egg Model are presented in Table 8. Also,

**Fig. 12** Plot of PSO-optimized field oil production rates, comparison between Scenarios 2 and 3



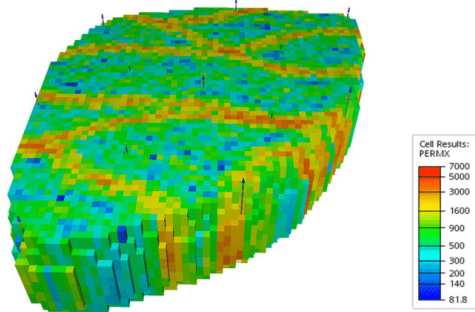
**Fig. 13** Plot of GWO-optimized field oil production rates, comparison between Scenarios 2 and 3



the mean of the corresponding coefficient of determination is computed for each proxy and shown in Table 9. For illustration purpose, like Figs. 5, 6, and 7, the graphs of the comparison between the actual and the predicted field liquid production rates (also field water cut) are shown in Fig. 15 for LHS, in Fig. 16 for SSS, and in Fig. 17 for HSS. To perform the optimization of NPV, different economic parameters, as shown in Table 10, are used because using the parameters in

Table 6 will result in a mathematically trivial solution in this case study. Table 11 illustrates the results of optimization of the three scenarios. The optimized controls of field water injection rates are shown in Fig. 18 for PSO and in Fig. 19 for GWO. In this case study, the overestimation of NPV by the data-driven proxies for both algorithms is also noticed. Nevertheless, this overestimation is practically infinitesimal in which the application of these proxies is still feasible.





**Fig. 14** The overview of the 3D Egg Model. The color bar indicates the values of horizontal permeability in x-direction in the units of millidarcy (mD)

**Table 8**  $R^2$  of training, validation, and testing results of the data-driven proxies

Dataset	Field liquid production rate	Field water cut
Training	0.9999	0.9999
Validation	0.9999	0.9999
Testing	0.9999	0.9999

**Table 9** Mean  $R^2$  of blind validation of proxies based on different sampling techniques

Sampling methods	Field liquid production rate	Field water cut
LHS	0.9999	0.9992
SSS	0.9999	0.9990
HSS	0.9999	0.9990

During the optimization using PSO, the absolute percentage error between  $NPV_{sim}$  and  $NPV_{proxy}$  is approximately 2.43%. Given the higher complexity of the Egg Model, the results produced are within satisfactory level of accuracy to approximate the  $NPV_{sim}$ . Also, the absolute percentage error between  $NPV_{sim}$  and  $NPV_{sim-proxy}$ , which is determined to be about 0.68%, portrays a higher confidence in the usefulness of the results obtained by the proxies. Besides that, the absolute percentage error between  $NPV_{sim-proxy}$  and  $NPV_{proxy}$  is 3.13%. It can be inferred that the proxies of Egg Model are deemed reliable as well. For the case of GWO, the absolute percentage error between  $NPV_{sim}$  and  $NPV_{proxy}$  is 2.72% while it is 1.81% between  $NPV_{sim}$  and  $NPV_{sim-proxy}$ . Also, the absolute percentage error between  $NPV_{sim-proxy}$  and  $NPV_{proxy}$  is 4.62%. In this case study, GWO can achieve a higher accuracy of optimization than PSO. In addition, GWO generally outperforms PSO in terms of optimization,

except for Scenario 2. The high proximity of these data-driven models is also captured through the demonstration of the plots of PSO-optimized field water production rates of Scenario 2 against Scenario 3 in Fig. 20 and those for GWO in Fig. 21.  $R^2$  computed for Fig. 20 is 0.9986 whereas that of Fig. 21 is 0.9978. The similar plots for field oil production rates are also shown for PSO in Fig. 22 and GWO in Fig. 23. Then,  $R^2$  determined for Fig. 22 is 0.9991 whereas that of Fig. 23 is 0.9981.

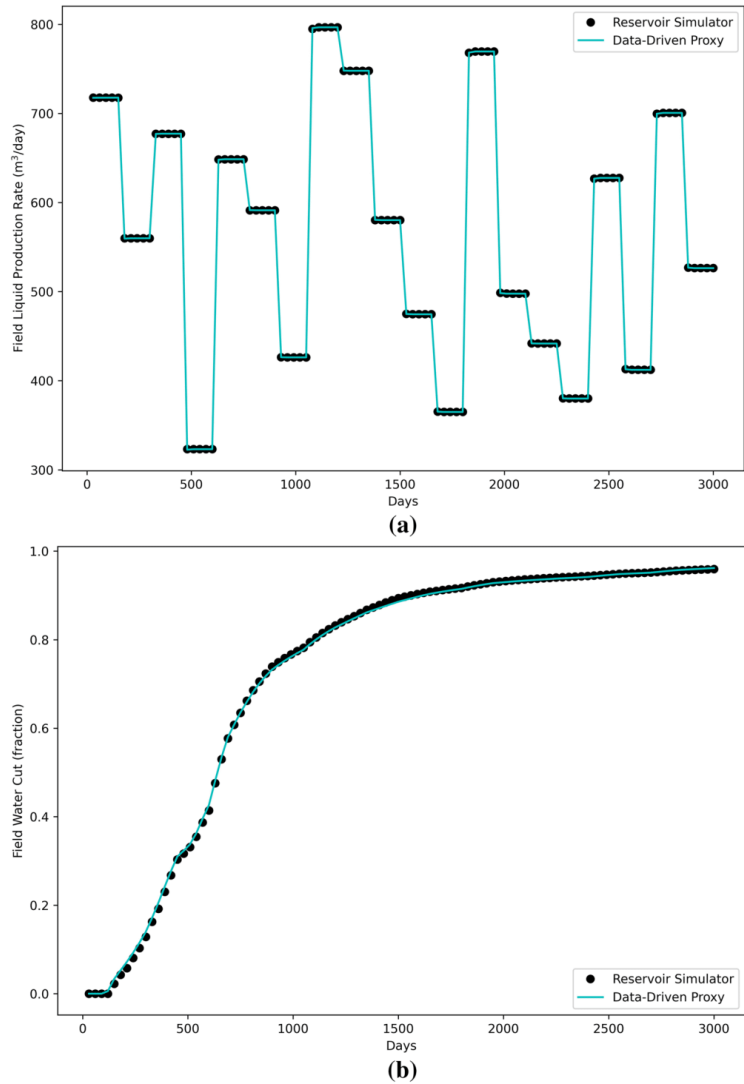
To demonstrate the accuracy and robustness of the approaches proposed in this study, the plots of field water (and oil) production rates under Scenario 2 against those under an unoptimized scheme are provided. To elude any confusion, the optimized rates used to produce these plots are derived from simulator. The unoptimized scheme, which is also known as “base case,” comprises a constant field injection rate of 560 m<sup>3</sup>/day over the whole production period. The corresponding NPV of base case is determined to be 152.57 million USD. Refer to Table 11 for the NPVs of the optimized cases (Scenario 2). Figure 24 illustrates the plot of PSO-optimized field water production rates (Scenario 2) against that of base case whereas Fig. 25 portrays the similar plot for field oil production rates. For GWO method, the plots for field water and oil production rates are, respectively, shown in Figs. 26 and 27. According to these four figures, we can fairly deduce that the optimization schemes have been performed practically well in the case study of Egg Model.

## Discussion

About the results of NPV optimization in Tables 7 and 11, it is observed that in all three scenarios for both case studies, GWO reached a higher optimal NPV than PSO, except for Scenario 2 in Egg Model. This shows that GWO generally outperforms PSO to yield better optimization results based upon the analysis conducted in this paper. Despite this, the underperformance of GWO in Scenario 2 of Egg Model can be due to the lack of efficiency in the sampling of data for the neural network training. This means that the data sampled might not be efficiently extensive to cover the solution space of optimization induced by GWO. Nonetheless, based on the results presented, the data-driven proxies are still able to practically serve their objective when coupled with GWO. The sampling strategy used in this paper is deemed straightforward and still has room for improvement. This domain is not emphasized much as it is not the focus of our study here. In this aspect, the efficient sampling algorithm initiated by Dige and Diwekar (2018) can be taken into account as future work to enhance the sampling strategy in this paper.

There are a few limitations about the data-driven proxies developed in this paper. One of them pertains to the

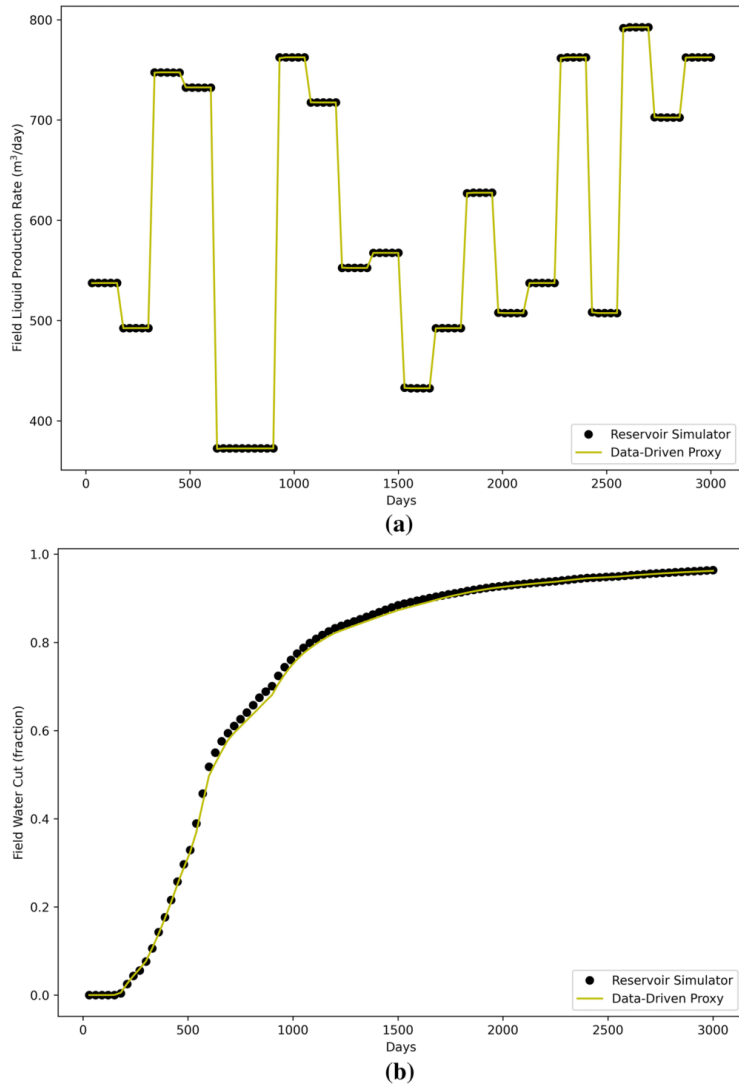
**Fig. 15** Results of blind validation of data-driven proxies of LHS sample set 11 (out of 80). **a** Field liquid production rate. **b** Field water cut



applicability of the models. As discussed in several literatures (Mohaghegh 2011, 2017a, 2017b; Ng et al. 2021), the data-driven model is only relevant to the reservoir model being studied. This denotes that it cannot be implemented as the substitute for another reservoir. In addition, the developed proxy models are only able to capture the physics of the reservoir system that is represented by the spatiotemporal

database. For instance, if the proxy is established for a reservoir model that is waterflooded, then it cannot be employed for the analysis of other enhanced oil recovery (EOR) methods, such as CO<sub>2</sub> injection and water-alternating-gas (WAG). The elimination of the control switch problem is considered as another limitation. This means that the reservoir simulation system is designed in a way that for the injectors, the

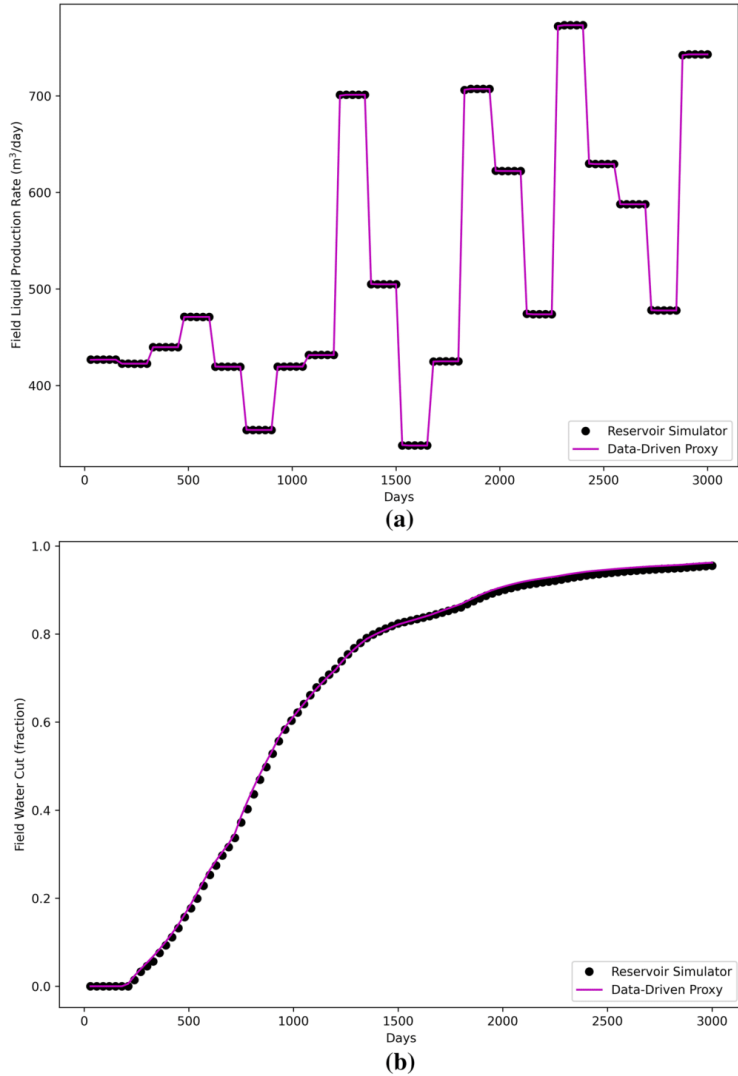
**Fig. 16** Results of blind validation of data-driven proxies of SSS sample set 32 (out of 80). **a** Field liquid production rate. **b** Field water cut



control will not switch from injection rates to BHP during the optimization process. The similar condition also applies to the producers (but BHP is the control for producers). Moreover, for the case study of Egg Model, the well rate is determined by equally dividing the field rate by the number of wells. This implies that the optimization problem presented here is slightly simplified for illustration purpose.

The aspect of computational cost is the catalyst for the rapid development of the proxy models. As discussed earlier, NRS can induce high computational footprints especially when the reservoir model is geologically very sophisticated. Therefore, applying proxy models for further analysis is undeniably time saving. To further demonstrate this advantage, we compare the computation time required by

**Fig. 17** Results of blind validation of data-driven proxies of HSS sample set 69 (out of 80). **a** Field liquid production rate. **b** Field water cut



**Table 10** Economics parameters used for NPV calculation

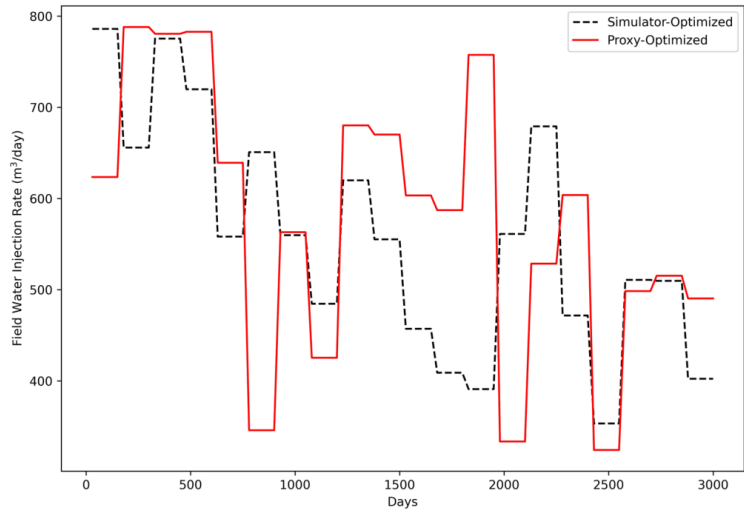
Parameters	Values	Units
Oil price, $P_o$	440.30	USD/m <sup>3</sup>
Cost of produced water, $C_w$	12.58	USD/m <sup>3</sup>
Cost of injected water, $C_{wi}$	12.58	USD/m <sup>3</sup>
Discount rate	0.10	Per year

performing the optimization on both reservoir and proxy models. It was conducted on a PC which has the specification of Intel® Core™ i9-9900 CPU @3.10 GHz with 64.0 GB RAM. In this context, the time used by both PSO and GWO are very close. For the 2D reservoir model, the optimization took about 3 h whereas its respective proxies utilized about 1 h and 40 min to finish the optimization. Therefore, it is seen that the proxy models were able to save

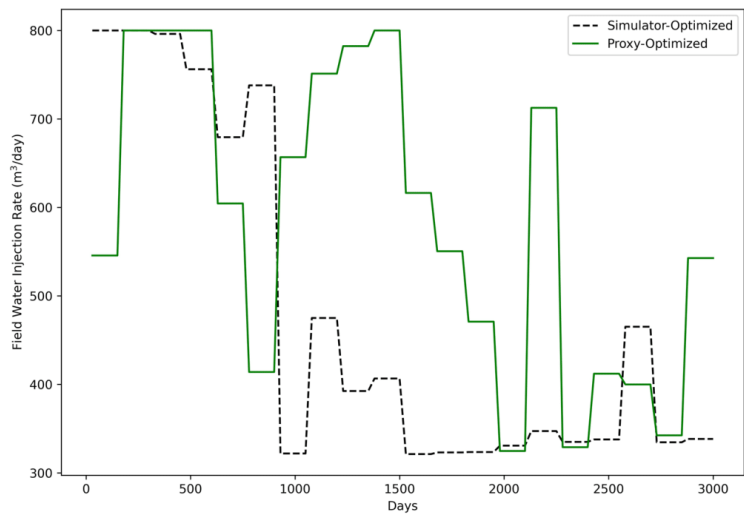
**Table 11** Optimization results of three scenarios for PSO and GWO

Optimization algorithm	Scenario 1		Scenario 2		Scenario 3	
	GWO	PSO	GWO	PSO	GWO	PSO
NPV <sub>optimal</sub> (million USD)	157.14	155.78	154.29	154.73	161.42	159.57

**Fig. 18** Optimized control rates by using simulator and data-driven proxies with the implementation of PSO



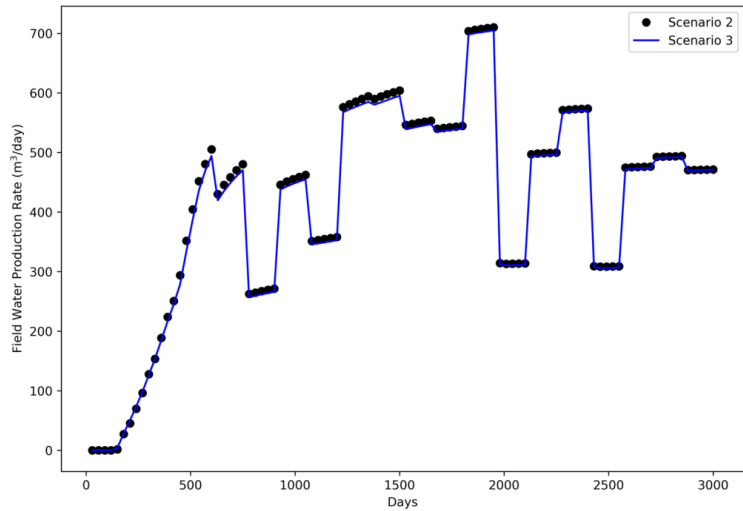
**Fig. 19** Optimized control rates by using simulator and data-driven proxies with the implementation of GWO



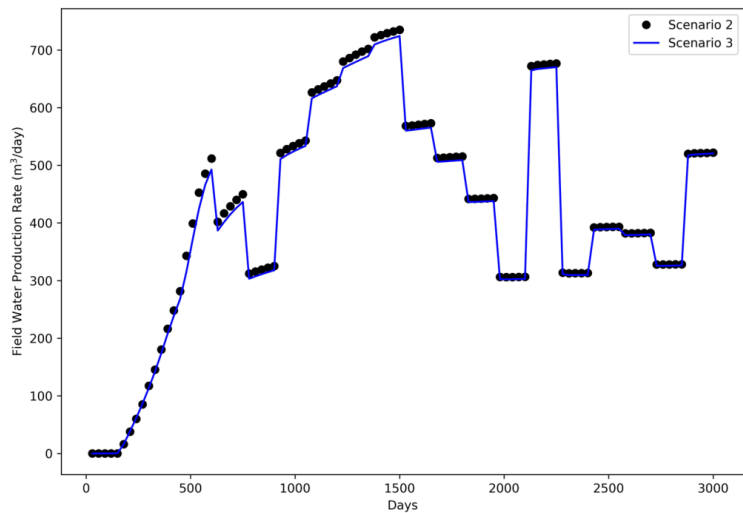
about 50% of the whole computational time. More intriguingly, this advantage is more obvious for the case study of Egg Model. When applying the optimization algorithm on

the Egg Model, it took about 13 h to run the optimization. However, the corresponding proxies only needed 2 h to do so. This illustrates that the data-driven proxies were 6 times

**Fig. 20** Plot of PSO-optimized field water production rates, comparison between Scenarios 2 and 3



**Fig. 21** Plot of GWO-optimized field water production rates, comparison between Scenarios 2 and 3

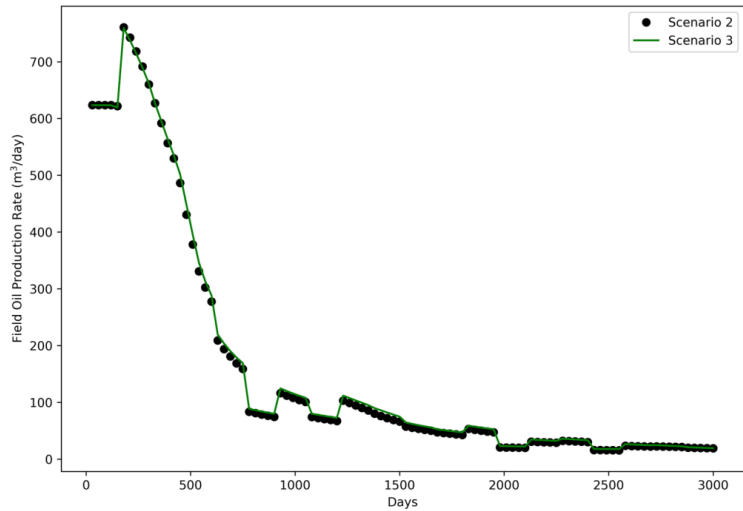


faster than the initial Egg Model in terms of optimization time.

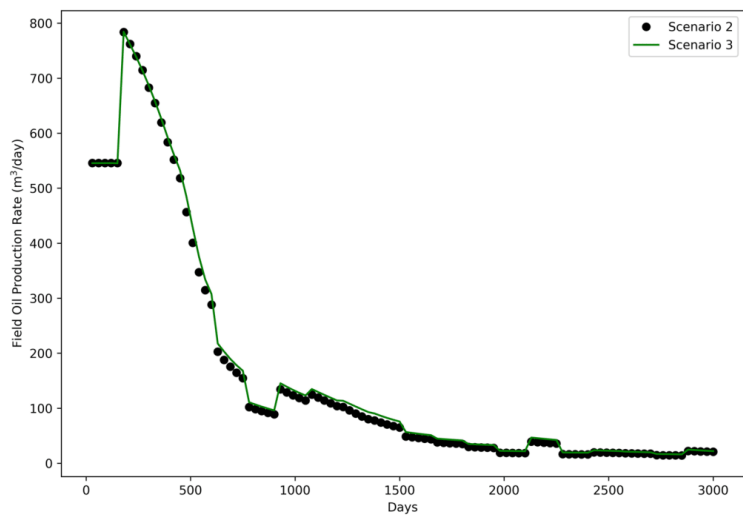
There is also an important concern about the number of reservoir simulations required for building the proxy models. Some literatures (Mohaghegh 2011; He et al. 2016; Vida et al. 2019; Shahkarami and Mohaghegh 2020) suggested a rule of thumb that 10 to 15 simulations could be sufficient for the development of robust proxy models. Nonetheless, Nait Amar et al. (2018) had run 75

simulations to generate the necessary database to develop the proxy models. Moreover, Golzari et al. (2015) even performed 200 simulations to build the data-driven models. Therefore, there is no strict rule of how many simulations are exactly needed to establish the spatiotemporal database. It is widely dependent upon the purpose of application of the data-driven models. Also, the bigger the database, the more accurate the data-driven model can be. Despite this, we need to understand that there is always a trade-off

**Fig. 22** Plot of PSO-optimized field oil production rates, comparison between Scenarios 2 and 3



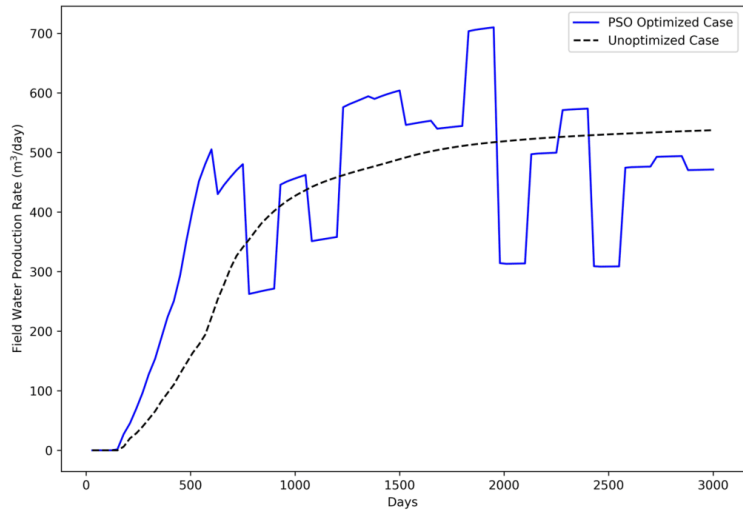
**Fig. 23** Plot of GWO-optimized field oil production rates, comparison between Scenarios 2 and 3



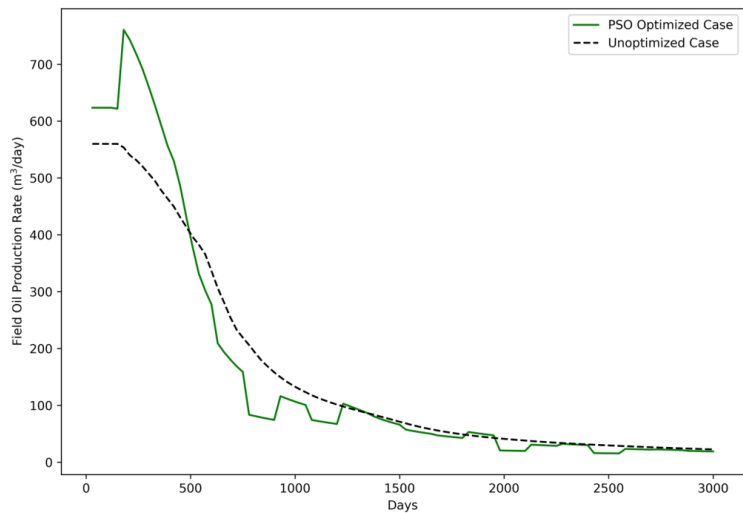
between the size of the database and the computational time. When the database is humongous, it means that the neural network training might take longer time to complete. This challenge is termed as the curse of dimensionality. In this paper, we empirically selected to run 60 simulations as explained to establish our spatiotemporal database. Upon building the proxies, we performed blind validation with 80 new data samples as discussed. Based on the results

shown, it can be deduced that this database was deemed to be practically sufficient to yield useful proxies. As presented in other literatures (Nait Amar et al. 2018; Amini and Mohaghegh 2019; Shahkarami and Mohaghegh 2020), the number of blind validation cases usually is about 10 or even less. In our work, we presented 80 blind validation cases to further demonstrate the higher feasibility of practical application of our data-driven proxies.

**Fig. 24** Plot of PSO-optimized field water production rates (Scenario 2) and those of unoptimized case



**Fig. 25** Plot of PSO-optimized field oil production rates (Scenario 2) and those of unoptimized case



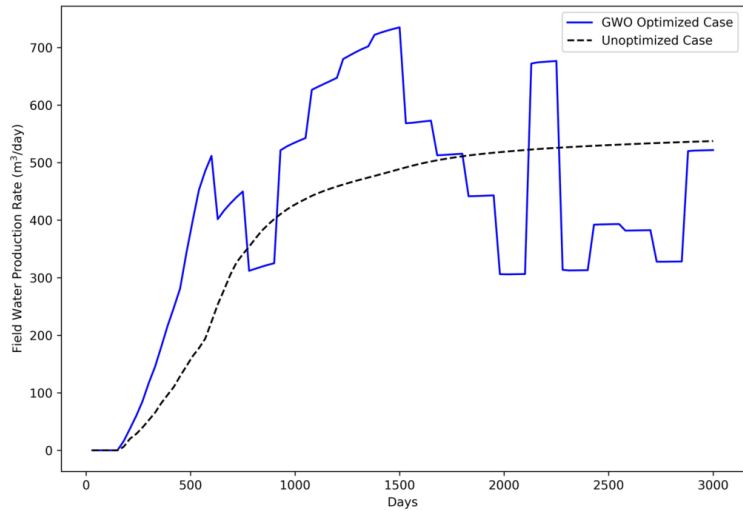
## Conclusions

In this work, we implemented ML technique to build dynamic proxy models and conduct the optimization of well control rates on two waterflooding case studies, i.e., a 2D synthetic reservoir model and the 3D Egg Model. The main objective was to achieve the maximization of NPV by determining the optimal control rate with the help of two

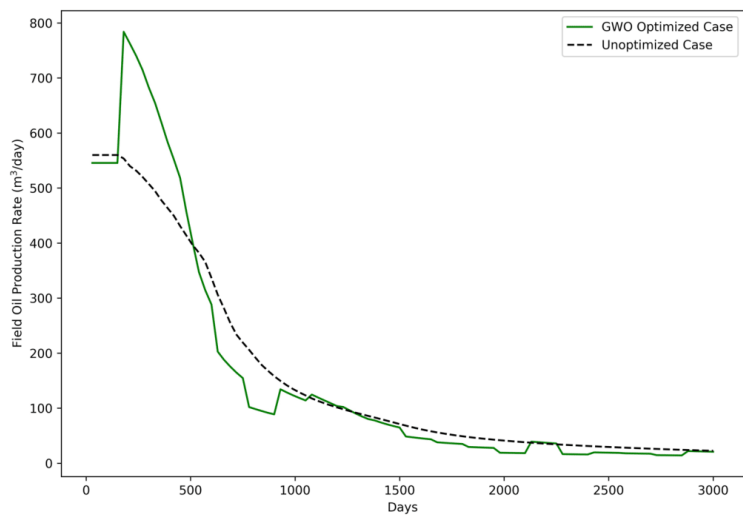
metaheuristic algorithms, which include PSO and GWO. In order to do that, for each case study, we maneuvered the modeling of ANN to build two proxy models in which one could predict the field liquid production rates at a certain time, and another could forecast the field water cut. Thereafter, we successfully coupled these models with the optimization algorithms to perform the waterflooding optimization. Based upon our investigation, GWO generally outperformed



**Fig. 26** Plot of GWO-optimized field water production rates (Scenario 2) and those of unoptimized case



**Fig. 27** Plot of GWO-optimized field oil production rates (Scenario 2) and those of unoptimized case



PSO in the context of optimization. However, the accuracy of results (prediction of optimized field liquid production rates and field water cut) was slightly higher when the proxies were coupled with PSO. This could be due to the sampling strategy applied in this study. Nonetheless, we conclude that the data-driven proxies have successfully served their purpose of application. Also, the results derived from this study verify the validity of the methodology presented in data-driven proxy modeling.

**Acknowledgements** This research is a part of BRU21—NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)).

**Funding** This study is supported by Norwegian University of Science and Technology (NTNU).

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing,

adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Alakeely A, Horne RN (2020) Simulating the behavior of reservoirs with convolutional and recurrent neural networks. *SPE Reserv Eval Eng*
- Alenezi F, Mohaghegh S (2017) Developing a smart proxy for the SACROC water-flooding numerical reservoir simulation model. In: *SPE Western Regional Meeting Proceedings*
- Amini S, Mohaghegh S (2019) Application of machine learning and artificial intelligence in proxy modeling for fluid flow in porous media. *Fluids*. <https://doi.org/10.3390/fluids4030126>
- Babaei M, Pan I (2016) Performance comparison of several response surface surrogate models and ensemble methods for water injection optimization under uncertainty. *Comput Geosci*. <https://doi.org/10.1016/j.cageo.2016.02.022>
- Baumann EJM, Dale SI, Bellout MC (2020) FieldOpt: a powerful and effective programming framework tailored for field development optimization. *Comput Geosci*. <https://doi.org/10.1016/j.cageo.2019.104379>
- Bellout MC, Echeverría Ciaurri D, Durlafsky LJ et al (2012) Joint optimization of oil well placement and controls. *Comput Geosci*. <https://doi.org/10.1007/s10596-012-9303-5>
- Bruce WA (1943) An electrical device for analyzing oil-reservoir behavior. *Trans AIME*. <https://doi.org/10.2118/943112-g>
- Buduma N, Locascio N (2017) Fundamentals of deep learning : Designing Next-Generation Machine Intelligence Algorithms
- Dige N, Diwekar U (2018) Efficient sampling algorithm for large-scale optimization under uncertainty problems. *Comput Chem Eng* 115:431–454. <https://doi.org/10.1016/j.compchemeng.2018.05.007>
- Ertekin T, Sun Q (2019) Artificial intelligence applications in reservoir engineering: a status check. *Energies*
- Forouzanfar F, Reynolds AC (2013) Well-placement optimization using a derivative-free method. *J Pet Sci Eng*. <https://doi.org/10.1016/j.petrol.2013.07.009>
- Forouzanfar F, Reynolds AC (2014) Joint optimization of number of wells, well locations and controls using a gradient-based algorithm. *Chem Eng Res Des*. <https://doi.org/10.1016/j.cherd.2013.11.006>
- Golzari A, Haghghat Sefat M, Jamshidi S (2015) Development of an adaptive surrogate model for production optimization. *J Pet Sci Eng*. <https://doi.org/10.1016/j.petrol.2015.07.012>
- Guo Z, Reynolds AC (2018) Robust life-cycle production optimization with a support-vector-regression proxy. *SPE J*. <https://doi.org/10.2118/191378-PA>
- Guyaguler B, Horne RN, Rogers L, Rosenzweig JJ (2002) Optimization of well placement in a gulf of Mexico waterflooding project. *SPE Reserv Eval Eng*. <https://doi.org/10.2118/78266-PA>
- Hammersley JM, Handscomb DC (1964) Monte Carlo methods
- He Q, Mohaghegh SD, Liu Z (2016) Reservoir simulation using smart proxy in SACROC unit - Case study. In: *SPE Eastern Regional Meeting*
- Hemmati-Sarapardeh A, Larestani A, Nait Amar M, Hajirezaie S (2020) Introduction. In: *Applications of artificial intelligence techniques in the petroleum industry*
- Hong AJ, Bratvold RB, Nævdal G (2017) Robust production optimization with capacitance-resistance model as precursor. *Comput Geosci*. <https://doi.org/10.1007/s10596-017-9666-8>
- Hong A, Bratvold RB, Lake LW (2019) Fast analysis of optimal improved-oil-recovery switch time using a two-factor production model and least-squares Monte Carlo algorithm. *SPE Reserv Eval Eng*. <https://doi.org/10.2118/191327-PA>
- Jansen JD, Douma SD, Brouwer DR, et al (2009) Closed-loop reservoir management. In: *SPE Reservoir Simulation Symposium Proceedings*
- Jansen JD, Fonseca RM, Kahrobaei S et al (2014) The egg model - a geological ensemble for reservoir simulation. *Geosci Data J*. <https://doi.org/10.1002/gdj3.21>
- Kalantari-Dahaghi A, Mohaghegh SD (2011) A new practical approach in modelling and simulation of shale gas reservoirs: application to New Albany Shale. *Int J Oil, Gas Coal Technol*. <https://doi.org/10.1504/IJOGCT.2011.038925>
- Kennedy J, Eberhart R (1995) Particle swarm optimization. In: *IEEE International Conference on Neural Networks - Conference Proceedings*
- Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*
- Liang X, Weber DB, Edgar TF, et al (2007) Optimization of oil production based on a capacitance model of production and injection rates. In: *SPE Hydrocarbon Economics and Evaluation Symposium*
- Lu R, Forouzanfar F, Reynolds AC (2017) An efficient adaptive algorithm for robust control optimization using StoSAG. *J Pet Sci Eng*. <https://doi.org/10.1016/j.petrol.2017.09.002>
- McKay MD, Beckman RJ, Conover WJ (1979) A Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*. <https://doi.org/10.2307/1268522>
- Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. *Adv Eng Softw*. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- Mohaghegh SD (2006) Quantifying uncertainties associated with reservoir simulation studies using surrogate reservoir models. In: *Proceedings - SPE Annual Technical Conference and Exhibition*
- Mohaghegh SD (2011) Reservoir simulation and modeling based on artificial intelligence and data mining (AI&DM). *J Nat Gas Sci Eng*. <https://doi.org/10.1016/j.jngse.2011.08.003>
- Mohaghegh SD (2013) Reservoir modeling of shale formations. *J. Nat. Gas Sci. Eng.*
- Mohaghegh SD (2017a) Data-driven reservoir modeling
- Mohaghegh SD (2017b) Shale analytics
- Mohaghegh SD, Hafez H, Gaskari R, et al (2006) Uncertainty analysis of a giant oil field in the middle east using surrogate reservoir model. In: *12th Abu Dhabi international petroleum exhibition and conference, ADIPEC 2006: meeting the increasing oil and gas demand through innovation*
- Mohaghegh SD, Liu J, Gaskari R, et al (2012) Application of surrogate reservoir model (SRM) to an onshore green field in Saudi Arabia; case study. In: *Society of Petroleum Engineers - North Africa Technical Conference and Exhibition 2012, NATC 2012: Managing Hydrocarbon Resources in a Changing Environment*
- Mohaghegh SD, Gaskari R, Maysami M (2017) Shale analytics: Making production and operational decisions based on facts: A case study in marcellus shale. In: *Society of Petroleum Engineers - SPE Hydraulic Fracturing Technology Conference and Exhibition 2017*
- Nait Amar M, Zeraibi N, Redouane K (2018) Optimization of WAG process using dynamic proxy, genetic algorithm and ant

- colony optimization. Arab J Sci Eng. <https://doi.org/10.1007/s13369-018-3173-7>
- Nait Amar M, Zeraibi N, Jahanbani Ghahfarokhi A (2020) Applying hybrid support vector regression and genetic algorithm to water alternating CO<sub>2</sub> gas EOR. Greenh Gases Sci Technol. <https://doi.org/10.1002/ghg.1982>
- Navrátil J, Kollias G, King AJ, et al (2019) Accelerating physics-based simulations using neural network proxies: an application in oil reservoir modeling. arXiv
- Ng CSW, Jahanbani Ghahfarokhi A, Nait Amar M, Torsæter O (2021) Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application. Nat Res Resour. <https://doi.org/10.1007/s11053-021-09844-2>
- Ogbeiwí P, Aladeitan Y, Udebhulu D (2018) An approach to waterflood optimization: case study of the reservoir X. J Pet Explor Prod Technol. <https://doi.org/10.1007/s13202-017-0368-5>
- Peaceman DW (1977) Fundamentals of numerical reservoir simulation. Elsevier, Amsterdam
- Pouladi B, Karkevandi-Talkhooncheh A, Sharifi M et al (2020) Enhancement of SPSA algorithm performance using reservoir quality maps: application to coupled well placement and control optimization problems. J Pet Sci Eng. <https://doi.org/10.1016/j.petrol.2020.106984>
- Sarma P, Chen WH, Durlfolsky LJ, Aziz K (2008) Production optimization with adjoint models under nonlinear control-state path inequality constraints. SPE Reserv Eval Eng. <https://doi.org/10.2118/99959-pa>
- Sayarpour M, Zuluaga E, Kabir CS, Lake LW (2007) The use of capacitance-resistive models for rapid estimation of waterflood performance and optimization. In: Proceedings - SPE Annual Technical Conference and Exhibition
- Shahkarami A, Mohaghegh S (2020) Applications of smart proxies for subsurface modeling. Pet Explor Dev. [https://doi.org/10.1016/S1876-3804\(20\)60057-X](https://doi.org/10.1016/S1876-3804(20)60057-X)
- Shahkarami A, Mohaghegh S, Gholami V et al (2014) Modeling pressure and saturation distribution in a CO<sub>2</sub> storage project using a surrogate reservoir model (SRM). Greenh Gases Sci Technol. <https://doi.org/10.1002/ghg.1414>
- Shahkarami A, Mohaghegh SD, Hajizadeh Y (2018) Assisted history matching using pattern recognition technology. Int J Oil Gas Coal Technol. <https://doi.org/10.1504/IJOGCT.2018.090966>
- Shi Y, Eberhart R (1998) Modified particle swarm optimizer. In: Proceedings of the IEEE Conference on Evolutionary Computation, ICEC
- Sobol IM (1967) On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput Math Math Phys. [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9)
- Teixeira AF, Secchi AR (2019) Machine learning models to support reservoir production optimization. In: IFAC-PapersOnLine
- Thakur GC (1996) What is reservoir management? JPT J Pet Technol. <https://doi.org/10.2118/26289-JPT>
- Udy J, Hansen B, Maddux S, et al (2017) Review of field development optimization of waterflooding, EOR, and well placement focusing on history matching and optimization algorithms. Processes
- Valladão DM, Torrado RR, Flach B, Embid S (2013) On the stochastic response surface methodology for the determination of the development plan of an oil & gas field. In: Society of Petroleum Engineers - SPE Intelligent Energy International 2013: Realising the Full Asset Value
- Van Essen GM, Zandvliet MJ, Van Den Hof PMJ et al (2009) Robust waterflooding optimization of multiple geological scenarios. SPE J. <https://doi.org/10.2118/102913-PA>
- Vida G, Shahab MD, Mohammad M (2019) Smart proxy modeling of SACROC CO<sub>2</sub>-EOR. Fluids. <https://doi.org/10.3390/fluids4020085>
- Volkov O, Bellout MC (2017) Gradient-based production optimization with simulation-based economic constraints. Comput Geosci. <https://doi.org/10.1007/s10596-017-9634-3>
- Wang L, Li ZP, Adenutsi CD et al (2021) A novel multi-objective optimization method for well control parameters based on PSO-LSSVR proxy model and NSGA-II algorithm. J Pet Sci Eng. <https://doi.org/10.1016/j.petrol.2020.107694>
- Xu C, Nait Amar M, Ghriga MA et al (2020) Evolving support vector regression using Grey Wolf optimization; forecasting the geomechanical properties of rock. Eng Comput. <https://doi.org/10.1007/s00366-020-01131-7>
- Yousef AA, Gentil P, Jensen JL, Lake LW (2006) A capacitance model to infer interwell connectivity from production- and injection-rate fluctuations. SPE Reserv Eval Eng. <https://doi.org/10.2118/95322-pa>
- Zhang K, Zhang LM, Yao J et al (2014) Water flooding optimization with adjoint model under control constraints. J Hydrodyn. [https://doi.org/10.1016/S1001-6058\(14\)60009-3](https://doi.org/10.1016/S1001-6058(14)60009-3)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## **Paper 4**

### ***Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar

# Production optimization under waterflooding with long short-term memory and metaheuristic algorithm



Cuthbert Shang Wui Ng <sup>a,\*</sup>, Ashkan Jahanbani Ghahfarokhi <sup>a</sup>, Menad Nait Amar <sup>b</sup>

<sup>a</sup> Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

<sup>b</sup> Département Etudes Thermodynamiques, Division Laboratoires, Sonatrach, Boumerdes, Algeria

## ARTICLE INFO

### Article history:

Received 26 August 2021

Received in revised form

26 November 2021

Accepted 31 December 2021

### Keywords:

Production optimization

Numerical reservoir simulation

Machine learning

Long short-term memory (LSTM)

Dynamic proxies

Particle swarm optimization (PSO)

## ABSTRACT

In petroleum domain, optimizing hydrocarbon production is essential because it does not only ensure the economic prospects of the petroleum companies, but also fulfills the increasing global demand of energy. However, applying numerical reservoir simulation (NRS) to optimize production can induce high computational footprint. Proxy models are suggested to alleviate this challenge because they are computationally less demanding and able to yield reasonably accurate results. In this paper, we demonstrated how a machine learning technique, namely long short-term memory (LSTM), was applied to develop proxies of a 3D reservoir model. Sampling techniques were employed to create numerous simulation cases which served as the training database to establish the proxies. Upon blind validating the trained proxies, we coupled these proxies with particle swarm optimization to conduct production optimization. Both training and blind validation results illustrated that the proxies had been excellently developed with coefficient of determination,  $R^2$  of 0.99. We also compared the optimization results produced by NRS and the proxies. The comparison recorded a good level of accuracy that was within 3% error. The proxies were also computationally 3 times faster than NRS. Hence, the proxies have served their practical purposes in this study.

© 2022 Southwest Petroleum University. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In petroleum industry, reservoir management (RM) is one of the domains that has been emphasized by many oil and gas companies. According to Wiggins and Startzman [1], RM is termed as the employment of available technology, financial and labor resources to optimize the economic performance and recovery of a reservoir. They [1] further expounded that RM could be fathomed as a sequence of operations from its initial discovery of a reservoir to its final abandonment. In this case, production optimization is one of the pivotal parts in RM. Oil and gas companies attempt to optimize hydrocarbon production not only to fulfill the increasing demand

for energy, but also to ensure their higher economic returns. One of the approaches of increased production is to perform waterflooding or water injection. Waterflooding is generally implemented to produce additional volume of hydrocarbon after primary recovery which relies upon natural mechanisms such as gas cap drive and gravitational drainage [2]. Additionally, careful planning and implementation of waterflooding are important to avoid any unnecessary expenditure during the implementation phase. Hence, waterflooding optimization has been emphasized in the research field [3–7] for years to help the oil and gas companies to improve their application of this technique.

To be more precise, waterflooding optimization is considered as one of the engineering problems that requires some mathematical algorithms to come up with some design parameters, which either maximize or minimize any predefined objective function [2,8]. Regarding this, these design parameters include well production rates, well injection rates, bottomhole pressure of well, initiation time of waterflooding, and so forth. More intriguingly, waterflooding problem can also be formulated into a multi-objective problem in which more than one objective

\* Corresponding author.

E-mail address: [cuthbert.s.w.ng@ntnu.no](mailto:cuthbert.s.w.ng@ntnu.no) (C.S.W. Ng).

Peer review under responsibility of Southwest Petroleum University.



Production and Hosting by Elsevier on behalf of KeAi

function is optimized [9–11]. This formulation provides more useful insights to the chemical or petroleum engineers as it has closer proximity to the real-life problem. Additionally, numerical reservoir simulation (NRS) is one of the most widely applied tools of reservoir modeling during the field development stage. NRS can be conveniently employed along with other algorithms to solve any problem related to production optimization. However, one of its drawbacks is that more computational effort is required if it is used to model a geologically sophisticated reservoir [12,13]. This is because NRS uses mathematical equations and physics-based approach to model the flow of fluid in the subsurface. Thus, the computational time of the fluid flow modeling undeniably increases as the complexity of the reservoir modeled increases. Mitigating this computational challenge has been one of the most prevalent research topics.

Thanks to data-driven technology, the computational challenge can be alleviated. Data-driven technology is a framework that applies any input and output data provided to establish a relationship among them [13]. A model that is yielded from this technology is known as “data-driven model”. In this aspect, the main building block of this technology is data. More importantly, machine learning (ML) is one of the techniques used for data-driven modeling. Examples of ML generally include artificial neural network, support vector machine, random forest, extreme gradient boosting, and so on. In addition, data-driven model has displayed its ability to be used as a proxy or surrogate model of NRS. Regarding this, a proxy or surrogate model in general acts as a substitute of NRS and is computationally faster and able to replicate the results of NRS within satisfied level of accuracy. In this context, Dr. Shahab Mohaghegh is one of the pioneers in the petroleum industry to have coined the term of smart proxy model (SPM). SPM is a proxy model that comprises an ensemble of numerous inter-linked neuro-fuzzy systems, which are trained to understand the fluid flow behaviors from NRS [13,14]. SPM has been demonstrated to be successful in different fields of application, including uncertainty analysis [15,16], CO<sub>2</sub> sequestration and utilization [17,18], history matching [19,20], waterflooding [21], and unconventional resources [22,23]. Apart from these, there are other captivating literatures [24–33] discussing the use of ML-based models in the petroleum domains. These literatures in general also elaborated on the high applicability of ML techniques to be employed as a substitute of NRS. Nevertheless, one of the limitations of ML-based proxy modeling is the sufficiency of data. This is because the established ML-based model might not be able to “learn properly” without being supplied with sufficient data. However, when it is provided with too much data, this might undermine the significance of proxy modeling as a lot of simulation runs have to be performed.

Other than being used as proxy models, ML techniques have portrayed their value in the development of predictive models. In this case, Talebkeikhah et al. [34] successfully implemented seven ML methods, based on 1000 experimental points from some Iranian crude samples, to develop the predictive models of viscosity at reservoir conditions. These methods include radial basis function neural network, multilayer perceptron, support vector regression, adaptive neuro-fuzzy inference system, decision trees, and random forest. Besides that, Nait Amar et al. [35] illustrated how the best two out of various developed ML-based models were chosen and combined under the paradigm of committee machine intelligent system (CMIS) to develop a model that could forecast thermal conductivity of carbon dioxide. They further showed the use of weight average approach and group method of data handling (GDMH) to establish the CMIS models. A similar approach was employed and discussed by Mehrjoo et al. [36] to create a predictive model of interfacial tension of methane-brine

systems at high pressure and salinity conditions. Also, based on 1985 experimental points, Nait Amar et al. [37] successfully applied gene expression programming to perform the modeling of density of binary and tertiary mixtures of ionic liquids and molecular solvents.

In this work, we have used an advanced ML technique that is long short-term memory (LSTM) to build two proxy models, which are correspondingly applied to predict field liquid production rate (FLPR) and field water cut (FWCT). It is essential to point out that the proxy models built here are considered as “dynamic proxies”, which are time-dependent. As Nait Amar et al. [24] stated, time-dependent proxies offer higher flexibility in terms of application under time-dependent constraints. As the two abovementioned dynamic proxies were developed, they were coupled with particle swarm optimization (PSO) to conduct the waterflooding optimization. The details would be presented in the next few sections. After this introduction, the paper is followed by the theoretical framework that generally briefs the techniques involved and the general methodology used in this work. Thereafter, results and discussion about the main findings of this work are presented. The paper then ends with some conclusive remarks derived from this work.

## 2. Theoretical framework

### 2.1. Long short-term memory (LSTM)

LSTM is a more advanced version of recurrent neural network (RNN) that is developed to process sequential data, such as texts, sentences, and so on [38]. A simple RNN is generally designed to preserve and deliver information from the current step to the next one [38]. However, a simple RNN suffers the problem of vanishing gradient in which a long-term information cannot be fully utilized [39]. Thus, large amount of previous information is unable to be stored to perform forecast within higher level of accuracy. To elude the problem of vanishing gradient, Hochreiter and Schmidhuber [39] built the LSTM in 1997. The fundamental topology of the LSTM used in this study is demonstrated in Fig. 1. The mathematical formulation of LSTM is shown below:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (1)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$\tilde{c}_t = \gamma(W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t \quad (4)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (5)$$

$$h_t = o_t \times \gamma(c_t) \quad (6)$$

The mechanism of LSTM revolves around a cell state  $c_t$ . Around the cell state, information is either added or removed via three gates, for instance forget gate  $f_t$ , input gate  $i_t$ , and output gate  $o_t$ . These gates evaluate if the sequential input data should be retained to save pertinent information to the latter stages. Thereafter, according to Equation (1), the forget gate decides on the addition or omission of information. Regarding this, the information in terms of input and hidden state will be saved (removed) if  $f_t$  is close to one (zero). Besides that, the input gate is calculated to update the cell state. Via this update, the evaluation of the importance of the input delivered to the next cell is done. Furthermore, the output gate

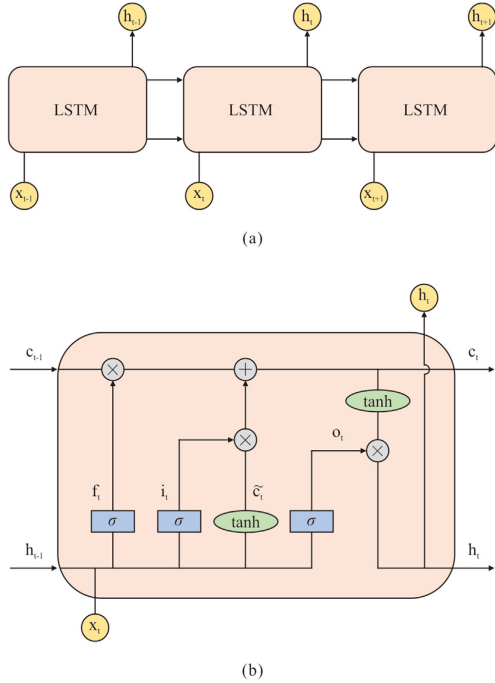


Fig. 1. Architecture of Long short-term memory (LSTM): (a) general topology of LSTM. (b) detailed structure of LSTM.

computes the output for the hidden states based on equation (6). It can be noticed that the activation function and the recurrent activation function used in LSTM are respectively hyperbolic tangent function (indicated as tanh) and sigmoid function (denoted as  $\sigma$ ).

### 2.2. Particle swarm optimization (PSO)

In 1995, Kennedy and Eberhart [40] established an optimization algorithm which was known as PSO. In this case, PSO is considered as an example of nature-inspired algorithms because it is formulated by simulating the behavior of flying stock of birds. Mathematically speaking, a swarm of particles indicates several possible solutions to an optimization problem. The status of each particle is computed according to its position and velocity. In this context, the dimension of both position and velocity is the same as the number of optimization parameters. In general, the algorithm commences through the random initialization of the position and velocity of each particle. A cost function, like mean squared error (MSE), is then employed to determine the fitness of each particle. After that, pbest and gbest are computed and saved to update the velocity at current iteration based on equation (7). In this context, pbest and gbest are found out for every iteration. pbest is the best position of a particle in the dimensional space and gbest is the overall best position of a particle hitherto in the whole swarm. Upon determining the velocity at next iteration, the position of a particle for the next iteration is updated as captured by equation (8). After a predefined number of iterations, each particle updates its position by minimizing the fitness value until the convergence of the optimal position occurs.

$$v_{jk, t+1} = \omega v_{jk, t} + c_1 r_1 (pbest_{jk, t} - x_{jk, t}) + c_2 r_2 (gbest_{k, t} - x_{jk, t}) \tag{7}$$

$$x_{jk, t+1} = x_{jk, t} + v_{jk, t+1} \tag{8}$$

In equation (7),  $v_{jk, t}$  corresponds to the velocity of the  $j$ th particle at step  $t$  in  $k$ th dimension.  $x_{jk, t}$  is its respective position.  $c_1$  and  $c_2$  correspondingly represent the cognitive and social learning factors that regulate the local and global search of the optimal solution. These parameters are selected by trial-and-error approach.  $r_1$  and  $r_2$  are random numbers extracted from uniform distribution of  $(0, 1)$ .  $\omega$  is inertial weight that was suggested by Shi and Eberhart [41] to better handle the convergence issue.

Apart from PSO, we would like to reiterate that there are several other metaheuristic algorithms that can be employed to perform modeling and optimization tasks. Examples of these algorithms [42] include, but are not limited to, genetic algorithm, differential evolution, simulated annealing, and ant colony optimization. In this aspect, PSO has been selected due to its computational efficiency and perceivable concept as being briefed in the literature [43]. Also, it has exhibited good results in some of our previous works [29,31,44].

### 2.3. Formulation of optimization problem and dynamic proxy

One of the most important perceptions about developing a proxy model is that it is an objective-oriented task. This implies that the background of the optimization problem must be clearly understood to provide better insights of proxy modeling. By perceiving the optimization problem, the modelers would know what variables or design parameters should be involved in creating the relevant proxies. Hence, formulation of optimization problem is indeed necessary in the development of proxies. In this work, the selected objective function is net present value (NPV), and it is mathematically shown in equation (9).

$$NPV(\mathbf{u}) = \sum_{i=1}^{n_{total}} \frac{(Q_o^i(\mathbf{u})P_o - Q_w^i(\mathbf{u})P_w - Q_{wi}^i(\mathbf{u})P_{wi}) \times \Delta t_i}{(1 + interest\ rate)^{t_i/D}} \tag{9}$$

where  $\mathbf{u}$  is the vector of optimization parameters,  $Q^i$  is the field production (injection) rate at timestep  $i$  and  $P$  represents price or cost. The subscripts of  $o, w,$  and  $wi$  respectively indicate oil, water, and water injected. In this work,  $P_o$  is 70 USD/bbl whereas both  $P_w$  and  $P_{wi}$  are 2 USD/bbl. Also, the optimization parameter used here is the field injection rate. Therefore, the optimization problem pertains to the adjustment of field water injection rate per 150 days for the period of 3000 days. Moreover,  $\Delta t_i$  is the difference of time between current and previous timestep. Besides that,  $t_i$  is the elapsed time from beginning until step  $i$  and  $D$  is the reference time for discounting.  $D$  is 365 days as interest rate has a unit of fraction per year and discounting of cash flow is done daily. The interest rate used here is 0.1 per year.

It is noticeable that the dynamic proxies developed here need to yield two parameters, which are field oil and water production rates (FOPR and FWPR). Therefore, by implementing LSTM method, we built two different dynamic proxy models, which respectively predict FLPR and FWCT at a specific timestep. Moreover, the input parameters are the number of days at every timestep  $i, t_i$ ; the harmonic mean of grid absolute permeability for every layer of formation,  $k_{harmonic}$ ; the standard deviation of grid absolute permeability for each formation layer,  $k_{Std\ Dev}$ ; the permeabilities of completed grid blocks (injectors and producers),  $k_{inj,prod}$ ; the field

water injection rate,  $\mathbf{u}$ ; the output value at previous timestep,  $y_{i-1}$ . The mathematical formulation of the proxies<sup>1</sup> is illustrated in equation (10). The harmonic mean of permeability for every formation layer is given by equation (11).

$$y_i = f \left( t_i, k_{\text{harmonic}}, k_{\text{Std Dev}}, k_{\text{(inj, prod)}}, \mathbf{u}, y_{i-1} \right) \quad (10)$$

$$k_{\text{harmonic}} = \frac{\sum_{j=1}^m L_j}{\sum_{j=1}^m \frac{L_j}{k_j}} \quad (11)$$

where  $L_j$  represents the depth at the top of grid block  $j$ ,  $k_j$  refers to the grid absolute permeability, and  $m$  denotes the number of grid blocks. Regarding the inputs of the permeabilities of completed grid blocks (injectors and producers), the reservoir model studied here is the “egg model” that was developed by Jansen et al. [45]. There are 7 layers in the reservoir model with 8 injectors and 4 producers. To avoid the curse of dimensionality, the arithmetic mean of the permeability of the completed grid blocks for every well is calculated and this will yield 12 permeability variables. There are also 14 variables of  $k_{\text{harmonic}}$  and  $k_{\text{Std Dev}}$  given egg model has 7 layers. In total, there are 29 input variables used to train the dynamic proxies.

About the geological properties of egg model, its permeability is heterogeneous whereas its porosity is homogeneous with a value of 0.2. The initial water saturation for each grid block is 0.1. The dimension of each block is  $8 \text{ m} \times 8 \text{ m} \times 4 \text{ m}$  with a total number of  $60 \times 60 \times 7$  (only 18533 grid blocks are active). The horizontal permeability distribution of egg model is illustrated in Fig. 2. Refer to Jansen et al. [45] for the remaining details of the geological properties of this model. To be able to conduct the studies here, the control of both injectors and producers has been altered. In this aspect, the eight injectors are identical, and the rate is within the range of  $40 \text{ m}^3/\text{day}$  and  $100 \text{ m}^3/\text{day}$ . Hence, the optimization problem considering the constraint is summarized as shown below:

$$\text{subject to } \begin{cases} \max \text{NPV (FWIR)} \\ 320 \text{ Sm}^3/\text{day} \leq \text{Field Water Injection Rate} \leq 800 \text{ Sm}^3/\text{day} \\ \text{Bottomhole Pressure of Each Producer} \geq 395 \text{ bar} \end{cases} \quad (12)$$

#### 2.4. Data preparation, neural network training, and blind validation procedure

After completing the formulation of optimization problem and dynamic proxy modeling, we have a clearer idea of input and output variable types. Thereafter, we employ the methodology discussed and used in Ref. [31] to conduct the proxy modeling. With respect to this, a database needs to be generated and formatted that can be used to train the dynamic proxies. To create this database, we generate 60 different injection schedules by employing three sampling techniques, such as Latin Hypercube Sampling [46], Hammersley Sequence Sampling [47], and Sobol Sequence Sampling [48]. Each technique constitutes 20 schedules. Thereafter, each of the schedules is fed into the reservoir simulator to provide the necessary information to build the database.

<sup>1</sup> The permeability refers only to the horizontal permeability, here. Also, the permeability in both x- and y-directions are the same.

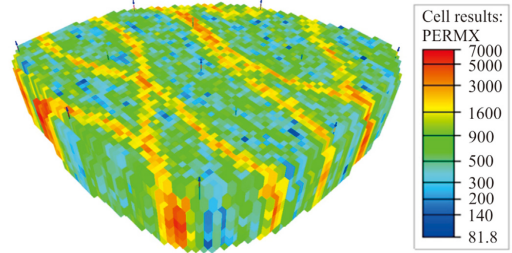


Fig. 2. Permeability distribution of egg model.

For illustrative purposes, the summary of the database is presented in Table 1. It is essential to highlight that the statistical parameters provided in Table 1 are determined “categorically”. For instance, for the variable of  $k_{\text{harmonic}}$ , the maximum and minimum values are determined by finding the highest and lowest values of all the 7 variables of  $k_{\text{harmonic}}$  (knowing that there are 7 layers). By following this logic, the pertinent mean and standard deviation are computed.

Then, when the database is ready, it is normalized between 0 and 1 “categorically” using the following formula:

$$X_{\text{normalized}} = \frac{X_n - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \quad (13)$$

where  $X_{\text{normalized}}$  implies the normalized value of  $X_n$  whereas  $X_{\text{max}}$  and  $X_{\text{min}}$  correspondingly represent the maximum and minimum values of  $X$ . Then, the database was divided into training set (80% of the points), validation (10% of the data), and testing sets (the remaining 10%). Validation set is employed to prevent any overfitting issue during training whereas testing set is used to evaluate

the predictability of the model prior to proceeding to blind validation phase. If excellent performance is illustrated during training, validation, and testing stages, then we would proceed to generate the database of blind validation. In this case, we reapply each of the three abovementioned sampling methods to respectively create additional 80 injection scenarios. Thereafter, we evaluate if the prediction performance of the dynamic proxies is within satisfied level of accuracy. Upon finishing the blind validation phase, the proxies are prepared for application. In this paper, we have utilized two statistical metrics to evaluate the training and prediction performance of the models, namely coefficient of determination and root mean squared error. The formula of each metrics is correspondingly displayed as Equations (14) and (15).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i^{\text{proxy}} - y_i^{\text{sim}})^2}{\sum_{i=1}^n (y_i^{\text{proxy}} - \bar{y})^2} \quad (14)$$



**Table 1**  
Summary of database.

Types of data	Number of data points	Maximum value	Minimum value	Mean value	Standard deviation
<b>Static data</b>					
$t_j$	1 × 6000	3000	30	1515	865.98
$k_{\text{harmonic}}$	7 × 6000	632.21	593.84	616.18	15.29
$k_{\text{Std Dev}}$	7 × 6000	1458.26	660.57	1010.98	262.06
$k_{\text{inj}}$	8 × 6000	1890.14	333.03	783.62	471.59
$k_{\text{prod}}$	4 × 6000	3759.54	361.41	1332.09	1404.51
<b>Dynamic data</b>					
$\mathbf{u}$	1 × 6000	800	320	559.76	138.51
$y_{i-1}$ and $y_i$ (FLPR)	2 × 6000	800.04	0	556.82	143.66
$y_{i-1}$ and $y_i$ (FWCT)	2 × 6000	1	0	0.710	0.319

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (Y_i^{\text{proxy}} - Y_i^{\text{sim}})^2}{n}} \quad (15)$$

where  $Y_i$  indicates the output value, the superscripts proxy and sim represent the proxy model and reservoir simulator model, respectively,  $\bar{Y}$  is the mean value of the output, and  $n$  is the number of data points.

### 3. Results and discussion

Before proceeding to the results of our dynamic proxy models, it is essential to briefly explain that the trial-and-error approach has been implemented to determine the topology of our proxies. In this case, the dynamic proxy of FLPR has been built with one input layer, one hidden layer, and one output layer. There are 50 nodes used in the hidden layer. Besides that, the dynamic proxy of FWCT has the similar architecture as that of FLPR but with an additional hidden layer. Both hidden layers consist of 50 nodes. Besides that, one of the backpropagation algorithms, namely adaptive moment estimation (Adam), has been applied to train both proxies. Peruse King and Ba [49] for details. Pertaining to the parameters considered for Adam, the number of training iterations is 2000, the learning rate is 0.001, exponential decay rate for the 1st moment estimates is 0.9, that for the 2nd moment estimates is 0.999, and numerical stability is  $10^{-7}$ .

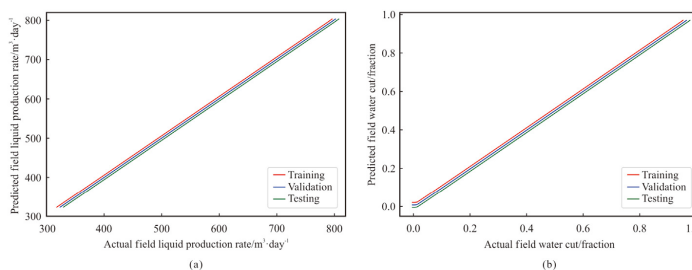
Fig. 3 illustrates the cross plot between the actual values and the predicted values for both proxies of FLPR and FWCT. Based on this plot, it is deducible that albeit the proxy of FLPR slightly outperforms that of FWCT, both proxies have undergone an excellent training phase. This is further supported by the results of training, validation, and testing performance displayed in Table 2. With respect to this, it can be confirmed that the overfitting issue has been prevented as the validation performances of both proxies are

**Table 2**  
Training, validation, and testing performances of the dynamic proxies.

		LSTM-FLPR	LSTM-FWCT
Training	$R^2$	0.9999	0.9999
	RMSE	0.2447	0.0021
Validation	$R^2$	0.9999	0.9999
	RMSE	0.2565	0.0020
Testing	$R^2$	0.9999	0.9999
	RMSE	0.2361	0.0016

as good as those of training. This also proves that both proxies have gone through a healthy trend of training. It is often important to ensure that the proxies have been trained “healthily”. Otherwise, the developed proxies will have a very weak predictability by only “memorizing” and being able to predict the data from the training set within satisfied level of accuracy. In addition, it is demonstrated that both proxies have good prediction ability as they have shown splendid testing results. Nevertheless, both proxies still must proceed to blind validation stage to further evaluate their predictability before being practically applied to perform optimization in this work.

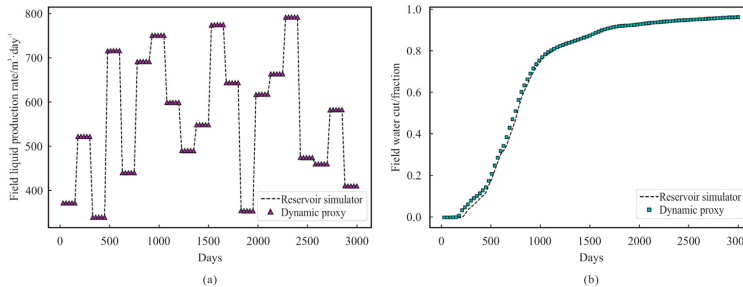
To conduct the blind validation, three different sampling methods have been used to correspondingly create 80 additional injection schedules as mentioned earlier. Hence, each of these schedules will yield a set of performance metrics for each proxy. To provide a better evaluation of blind validation performance, the mean of the metrics for each sampling technique is shown instead in Table 3. Based upon the results, it can be inferred that both proxies have been successfully blind validated and are prepared to be used for optimization. However, for illustrative purpose, the blind validation results of one of the samples retrieved by using Latin Hypercube method are displayed in Fig. 4. Although the blind validation dataset has not been used to develop the models, the models can still predict the outputs reasonably well. This further



**Fig. 3.** Cross plot between actual and predicted values considering training, validation, and testing sets: (a) FLPR. (b) FWCT.

**Table 3**  
Blind validation performances of the dynamic proxies considering three sampling techniques.

		LSTM-FLPR	LSTM-FWCT
Latin hypercube	Mean R <sup>2</sup>	0.9999	0.9992
	Mean RMSE	0.2513	0.0078
Sobol sequence	Mean R <sup>2</sup>	0.9999	0.9989
	Mean RMSE	0.2109	0.0093
Hammersley sequence	Mean R <sup>2</sup>	0.9999	0.9989
	Mean RMSE	0.2040	0.0092



**Fig. 4.** Blind validation of Latin Hypercube sample set 32 (out of 80): (a) FLPR. (b) FWCT.

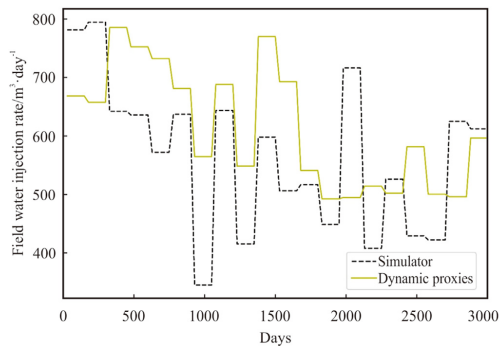
provides higher confidence regarding the integrity of the proxies built in this paper.

As it has been explained, both proxies of FLPR and FWCT have been coupled with PSO to conduct the waterflooding optimization. In this aspect, the FWIR would be periodically tuned to maximize the NPV for a certain period of production. Regarding the parameters of PSO, the inertial weight is 0.8 whereas both the social and cognitive learning factors are 1.05. Also, the number of iterations is initialized to be 100 in tandem with 15 particle swarms per iteration. The case in which the optimization is done by applying both proxies, is termed as “dynamic proxies”. Thereafter, to assess the proximity of results of optimization, the optimized FWIRs resulted from the case of “dynamic proxies” are fed into the simulator to compute its respective NPV. Such case of optimization is known as “simulator–dynamic proxies” in this paper. To have a more comprehensive comparison, the reservoir simulator has also been coupled with PSO to conduct the optimization. This case is labeled as “simulator”.

Upon completing these three cases, the optimal NPV obtained from each case is recorded in Table 4. In general, it is noticeable that the proxies have illustrated practically accurate results. When comparing the NPVs of “simulator–dynamic proxies” and “dynamic proxies”, the error is calculated to be 2.6%. Furthermore, the error between “simulator” and “dynamic proxies” is determined to be 1.6%. For illustrative purpose, the optimized FWIRs derived from “simulator” and “dynamic proxies” are plotted in Fig. 5. More interestingly, regarding the strength of the models, the computational time for “dynamic proxies” is about 4 h whereas that of “simulator” is about 12 h. Hence, the dynamic proxies are 3 times faster than the simulator for optimization in this study. This highlights the significance of the application of dynamic proxies. To

**Table 4**  
Optimal NPV considering three cases.

Models	Simulator	Simulator–dynamic proxies	Dynamic proxies
NPV <sub>optimal</sub> (million USD)	155.89	154.39	158.34



**Fig. 5.** Optimized FWIRs derived from Simulator and Dynamic Proxies.

further check the integrity of these proxies, the plot of optimized field water (oil) production rates between “simulator–dynamic proxies” and “dynamic proxies” is illustrated in Figs. 6–7. The respective statistical evaluation is also tabulated in Table 5. Based on these results, both proxies have practically served their purposes of application by reaching satisfied level of accuracy with less demanding computational effort.

Nonetheless, there are a few limitations about the models developed in this work. As mentioned earlier, one of the limitations includes the application of the models. In this aspect, proxy modeling is an objective-driven task. Therefore, the established models can only be aptly employed to solve the optimization problem outlined. Besides that, there is a concern about the behavior of the training database as noise, which is an important issue to flow rate signal, is not considered in the data used. Hence, the models might not demonstrate high applicability when noisy data is introduced for optimization purpose. This is indeed part of the future works that is worth investigating.

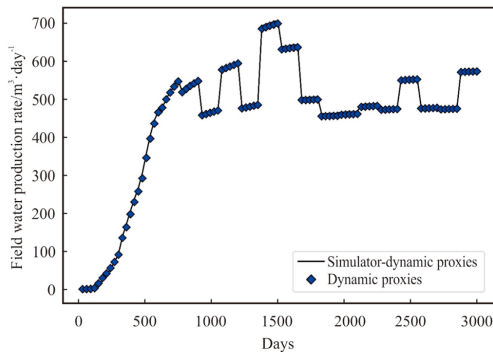


Fig. 6. Optimized FWPR derived from Simulator-Dynamic Proxies and Dynamic Proxies.

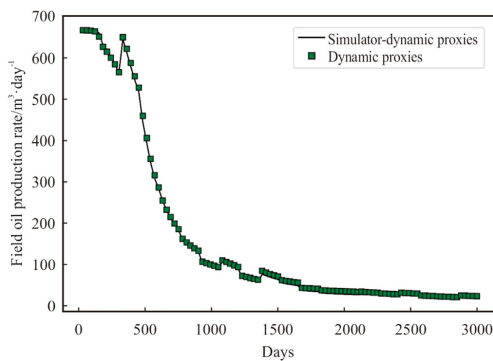


Fig. 7. Optimized FWOR derived from Simulator-Dynamic Proxies and Dynamic Proxies.

**Table 5**  
Statistical evaluation of optimized FWPR and FOPR.

		Optimized FWPR	Optimized FOPR
Optimization	R <sup>2</sup>	0.9990	0.9993
	RMSE	5.531	5.604

#### 4. Conclusions

In this study, we applied the LSTM approach to develop two dynamic proxies, which correspondingly could predict FLPR and FWCT based upon a 3D reservoir model known as the “Egg Model”. One of the main objectives of this investigation was to study the applicability of LSTM to be employed as proxy models for production optimization. According to the training and blind validation results, it could be deduced that these two proxies could accurately emulate the outputs yielded by the reservoir simulator. Moreover, we coupled these dynamic proxies with PSO to conduct the optimization. From the results of optimization and comparative analysis, the dynamic proxies were able to yield optimal results close to simulator only within 3% error, but 3 times faster. This finding further highlights the significance of dynamic proxies in terms of

application. Although these proxies are case-dependent, they have excellently served their purpose of use in this study. Besides that, these summarized findings also confirm the cogency of the methodology used to establish these dynamic proxies. Finally, we also believe that there is still room for improvement of the methodology discussed in this paper. One of them includes the consideration of noise-handling ability as highlighted earlier. Besides that, the introduction of decision variables with higher dimensionality and the application of multi-objective optimization are parts of possible future studies. As the methodology achieves a satisfactory level of maturity, its potential use can later be extended to optimization of CO<sub>2</sub> storage and/or EOR.

#### Declaration of interest statement

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.

#### Acknowledgement

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)).

#### References

- [1] M.L. Wiggins, R.A. Startzman, An approach to reservoir management, SPE Repr. Ser. (1998), <https://doi.org/10.2118/20747-ms>.
- [2] L.W. Lake, R. Johns, B. Rossen, G.A. Pope, Fundamentals of Enhanced Oil Recovery, others, 2014.
- [3] B. Guyaguler, R.N. Horne, L. Rogers, J.J. Rosenzweig, Optimization of well placement in a gulf of Mexico waterflood project, SPE Reservoir Eval. Eng. (2002), <https://doi.org/10.2118/78266-PA>.
- [4] A. Mamghaderi, A. Bastami, P. Pourafshary, Optimization of waterflooding performance in a layered reservoir using a combination of capacitance-resistive model and genetic algorithm method, J. Energy Resour. Technol. (2013), <https://doi.org/10.1115/1.4007767>.
- [5] J.L. Mogollón, T.M. Lokhandwala, E. Tilleró, New trends in waterflooding project optimization, SPE Lat. Am. Caribb. Pet. Eng. Conf. Proc. (2017), <https://doi.org/10.2118/185472-ms>.
- [6] A.J. Hong, R.B. Bratvold, G. Nævdal, Robust production optimization with capacitance-resistance model as precursor, Comput. Geosci. (2017), <https://doi.org/10.1007/s10596-017-9666-8>.
- [7] P. Ogbewi, Y. Aladeitan, D. Udehbulu, An approach to waterflood optimization: case study of the reservoir X, J. Pet. Explor. Prod. Technol. (2018), <https://doi.org/10.1007/s13202-017-0368-5>.
- [8] S.S. Rao, Engineering optimization: theory and practice. <https://doi.org/10.1002/9781119454816>, 2019.
- [9] M.C. Bellout, D. Echeverría Ciaurri, L.J. Durlofsky, B. Foss, J. Kleppe, Joint optimization of oil well placement and controls, Comput. Geosci. (2012), <https://doi.org/10.1007/s10596-012-9303-5>.
- [10] X. Liu, A.C. Reynolds, Gradient-based multi-objective optimization with applications to waterflooding optimization, Comput. Geosci. 20 (2016), <https://doi.org/10.1007/s10596-015-9523-6>.
- [11] M. Al-Aghbari, M. Al-Wadhahi, A.M. Gujarathi, Multi-objective optimization of Brugge field for short-term and long-term waterflood management, Arabian J. Sci. Eng. (2021), <https://doi.org/10.1007/s13369-021-05614-7>.
- [12] S.D. Mohaghegh, Reservoir simulation and modeling based on artificial intelligence and data mining (AI&DM), J. Nat. Gas Sci. Eng. (2011), <https://doi.org/10.1016/j.jngse.2011.08.003>.
- [13] S.D. Mohaghegh, Data-Driven Reservoir Modeling, 2017.
- [14] S.D. Mohaghegh, S. Amini, V. Gholami, R. Gaskari, G. Bromhal, Grid-Based Surrogate Reservoir Modeling (SRM) for fast track analysis of numerical reservoir simulation models at the grid block level, Soc. Pet. Eng. West. Reg. Meet. (2012), <https://doi.org/10.2118/153844-ms>, 2012.
- [15] S.D. Mohaghegh, Quantifying uncertainties associated with reservoir simulation studies using surrogate reservoir models, Proc. SPE Annu. Tech. Conf. Exhib. (2006), <https://doi.org/10.2523/102492-ms>.
- [16] S.D. Mohaghegh, H. Hafez, R. Gaskari, M. Haajizadeh, M. Kenawy, Uncertainty analysis of a giant oil field in the middle east using surrogate reservoir model, in: 12th Abu Dhabi Int. Pet. Exhib. Conf. ADIPEC 2006 Meet. Increasing Oil Gas Demand through Innov. 2006, <https://doi.org/10.2523/101474-ms>.
- [17] G. Vida, M.D. Shahab, M. Mohammad, Smart proxy modeling of SACROC CO<sub>2</sub>-EOR, Fluids (2019), <https://doi.org/10.3390/fluids4020085>.
- [18] A. Shahkarami, S. Mohaghegh, Applications of smart proxies for subsurface

- modeling. *Petrol. Explor. Dev.* (2020), [https://doi.org/10.1016/S1876-3804\(20\)60057-X](https://doi.org/10.1016/S1876-3804(20)60057-X).
- [19] A. Shahkarami, S.D. Mohaghegh, V. Gholami, S.A. Haghghat, Artificial intelligence (AI) assisted history matching, in: *Soc. Pet. Eng. SPE West. North Am. Rocky Mt. Jt. Meet.* 2014, <https://doi.org/10.2118/169507-ms>.
- [20] Q. He, S.D. Mohaghegh, Z. Liu, Reservoir simulation using smart proxy in SACROC unit - case study, in: *SPE East. Reg. Meet.* 2016, <https://doi.org/10.2118/184069-MS>.
- [21] F. Alenezi, S. Mohaghegh, Developing a smart proxy for the SACROC waterflooding numerical reservoir simulation model, in: *SPE West. Reg. Meet. Proc.*, 2017, <https://doi.org/10.2118/185691-ms>.
- [22] J. Jalali, S.D. Mohaghegh, Reservoir simulation and uncertainty analysis of enhanced CBM production using artificial neural networks, in: *SPE East. Reg. Meet.* 2009, <https://doi.org/10.2118/125959-ms>.
- [23] A. Kalantari-Dahaghi, S.D. Mohaghegh, A new practical approach in modelling and simulation of shale gas reservoirs: application to New Albany Shale, *Int. J. Oil Gas Coal Technol.* (2011), <https://doi.org/10.1504/IJOGCT.2011.038925>.
- [24] M. Nait Amar, N. Zeraibi, K. Redouane, Optimization of WAG process using dynamic proxy, genetic algorithm and ant colony optimization, *Arabian J. Sci. Eng.* (2018), <https://doi.org/10.1007/s13369-018-3173-7>.
- [25] M.A. Menad, Z. Nouredine, An efficient methodology for multi-objective optimization of water alternating CO2 EOR process, *J. Taiwan Inst. Chem. Eng.* 99 (2019) 154–165, <https://doi.org/10.1016/j.jtice.2019.03.016>.
- [26] J. Kim, H. Yang, J. Choe, Robust optimization of the locations and types of multiple wells using CNN based proxy models, *J. Petrol. Sci. Eng.* 193 (2020), 107424, <https://doi.org/10.1016/j.petrol.2020.107424>.
- [27] J. Kim, K. Lee, J. Choe, Efficient and robust optimization for well patterns using a PSO algorithm with a CNN-based proxy model, *J. Petrol. Sci. Eng.* 207 (2021), 109088, <https://doi.org/10.1016/j.petrol.2021.109088>.
- [28] L. Deng, Y. Pan, Data-driven proxy model for waterflood performance prediction and optimization using Echo State Network with Teacher Forcing in mature fields, *J. Petrol. Sci. Eng.* 197 (2021), 107981, <https://doi.org/10.1016/j.petrol.2020.107981>.
- [29] C.S.W. Ng, A. Jahanbani Ghahfarokhi, M. Nait Amar, O. Torsæter, Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application, *Nat. Resour. Res.* 30 (2021) 2431–2462, <https://doi.org/10.1007/s11053-021-09844-2>.
- [30] M. Nait Amar, A. Jahanbani Ghahfarokhi, C.S.W. Ng, N. Zeraibi, Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms, *J. Petrol. Sci. Eng.* (2021), 109038, <https://doi.org/10.1016/j.petrol.2021.109038>.
- [31] C.S.W. Ng, A. Jahanbani Ghahfarokhi, M. Nait Amar, Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization, *J. Pet. Explor. Prod. Technol.* (2021), <https://doi.org/10.1007/s13202-021-01199-x>.
- [32] S.H. Yousefi, F. Rashidi, M. Sharifi, M. Soroush, A.J. Ghahfarokhi, Interwell connectivity identification in immiscible gas-oil systems using statistical method and modified capacitance-resistance model: a comparative study, *J. Petrol. Sci. Eng.* 198 (2021), 108175, <https://doi.org/10.1016/j.petrol.2020.108175>.
- [33] M. Nait Amar, N. Zeraibi, A. Jahanbani Ghahfarokhi, Applying hybrid support vector regression and genetic algorithm to water alternating CO2 gas EOR, *Greenh. Gases Sci. Technol.* (2020), <https://doi.org/10.1002/ghg.1982>.
- [34] M. Talebkeikah, M. Nait Amar, A. Naseri, M. Humand, A. Hemmati-Sarapardeh, B. Dabir, M.E.A. Ben Seghier, Experimental measurement and compositional modeling of crude oil viscosity at reservoir conditions, *J. Taiwan Inst. Chem. Eng.* 109 (2020) 35–50, <https://doi.org/10.1016/j.jtice.2020.03.001>.
- [35] M. Nait Amar, A. Jahanbani Ghahfarokhi, N. Zeraibi, Predicting thermal conductivity of carbon dioxide using group of data-driven models, *J. Taiwan Inst. Chem. Eng.* 113 (2020) 165–177, <https://doi.org/10.1016/j.jtice.2020.08.001>.
- [36] H. Mehrjoo, M. Riazi, M. Nait Amar, A. Hemmati-Sarapardeh, Modeling interfacial tension of methane-brine systems at high pressure and high salinity conditions, *J. Taiwan Inst. Chem. Eng.* 114 (2020) 125–141, <https://doi.org/10.1016/j.jtice.2020.09.014>.
- [37] M. Nait Amar, M.A. Ghriga, A. Hemmati-Sarapardeh, Application of gene expression programming for predicting density of binary and ternary mixtures of ionic liquids and molecular solvents, *J. Taiwan Inst. Chem. Eng.* 117 (2020) 63–74, <https://doi.org/10.1016/j.jtice.2020.11.029>.
- [38] M.Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, M. Hasan, B.C. Van Essen, A.A.S. Awwal, V.K. Asari, A state-of-the-art survey on deep learning theory and architectures, *Electron* 8 (2019), <https://doi.org/10.3390/electronics8030292>.
- [39] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.* 9 (1997) 1735–1780, <http://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf>.
- [40] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *IEEE Int. Conf. Neural Networks Conf. Proc.* 1995, <https://doi.org/10.4018/ijmfp.2015010104>.
- [41] Y. Shi, R. Eberhart, Modified particle swarm optimizer, in: *Proc. IEEE Conf. Evol. Comput. ICEC*, 1998, <https://doi.org/10.1109/icec.1998.699146>.
- [42] K.L. Du, M.N.S. Swamy, Search and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature, Springer International Publishing, 2016, <https://doi.org/10.1007/978-3-319-41192-7>.
- [43] A.E. Ezugwu, O.J. Adeleke, A.A. Akinyelu, S. Viriri, A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems, *Neural Comput. Appl.* 32 (2020), <https://doi.org/10.1007/s00521-019-04132-w>.
- [44] C.S.W. Ng, A. Jahanbani Ghahfarokhi, M. Nait Amar, Well production forecast in Volve field: application of rigorous machine learning techniques and metaheuristic algorithm, *J. Petrol. Sci. Eng.* 208 (2022), 109468, <https://doi.org/10.1016/j.petrol.2021.109468>.
- [45] J.D. Jansen, R.M. Fonseca, S. Kahrobai, M.M. Siraj, G.M. Van Essen, P.M.J. Van den Hof, The egg model - a geological ensemble for reservoir simulation, *Geosci. Data J.* (2014), <https://doi.org/10.1002/gdj3.21>.
- [46] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* (1979), <https://doi.org/10.2307/1268522>.
- [47] J.M. Hammersley, D.C. Handscomb, Monte Carlo methods, <https://doi.org/10.1007/978-94-009-5819-7>, 1964.
- [48] I.M. Sobol, On the distribution of points in a cube and the approximate evaluation of integrals, *USSR Comput. Math. Math. Phys.* (1967), [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9).
- [49] D.P. Kingma, J.L. Ba, Adam: a method for stochastic optimization, in: *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, 2015.

## **Paper 5**

### ***Adaptive Proxy-based Robust Production Optimization with Multilayer Perceptron***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi



# Adaptive Proxy-based Robust Production Optimization with Multilayer Perceptron

Cuthbert Shang Wui Ng<sup>\*</sup>, Ashkan Jahanbani Ghahfarokhi

Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

## ARTICLE INFO

### Keywords:

Machine learning  
Data-driven modeling  
Multilayer perceptron  
Nature-inspired algorithms  
Adaptive training  
Robust production optimization

## ABSTRACT

Machine learning (ML) has been a technique employed to build data-driven models that can map the relationship between the input and output data provided. ML-based data-driven models offer an alternative path to solving optimization problems, which are conventionally resolved by applying simulation models. Higher computational cost is induced if the simulation model is computationally intensive. Such a situation aptly applies to petroleum engineering, especially when different geological realizations of numerical reservoir simulation (NRS) models are considered for production optimization. Therefore, data-driven models are suggested as a substitute for NRS. In this work, we demonstrated how multilayer perceptron could be implemented to build data-driven models based on 10 realizations of the Egg Model. These models were then coupled with two nature-inspired algorithms, viz. particle swarm optimization and grey wolf optimizer to solve waterflooding optimization. These data-driven models were adaptively re-trained by applying a training database that was updated via the addition of extra samples retrieved from optimization with the proxy models. The details of the methodology will be divulged in the paper. According to the results obtained, we could deduce that the methodology generated reliable data-driven models to solve the optimization problem, as justified by the excellent performance of the ML-based proxy model (with a coefficient of determination,  $R^2$  exceeding 0.98 in training, testing, and blind validation) and accurate optimization result (less than 1% error between the Expected Net Present Values optimized using NRS and proxy models). This study aids in an enhanced understanding of implementing adaptive training in tandem with optimization in ML-based proxy modeling.

## 1. Introduction

At the dawn of 21st century, energy has become an essential part of daily life due to its significant contribution and utilization in different sectors of human activities. The importance of energy had been further illustrated when the global energy demand in 2021 generally was expected to increase by 4.6%, which would exceed that of the pre-COVID-19 level, as reported by [International Energy Agency \(2021\)](https://www.iea.org/reports/global-energy-outlook-2021). Hence, meticulous planning of energy extraction and usage is required to ensure that the increasing global population can be commensurate with the availability of energy. In this aspect, petroleum is considered one of the primary sources of energy. Different technological methods, viz. enhanced oil recovery (EOR), artificial lift, hydraulic fracturing, etc., have been developed and employed to guarantee a sufficient supply of energy. Nevertheless, to produce petroleum sustainably and economically, oil and gas companies often incorporate a thorough blueprint of field development (FD) and reservoir management (RM). This is where

engineering optimization plays a pivotal role.

In the domains of FD and RM, engineering optimization of decision variables has been ubiquitous and user-friendly because of the rapid development of today's technology. In petroleum engineering, these decision variables include, but are not limited to, EOR initiation time, the number of wells, well control, well placement, well trajectory, etc. In tandem with the growth of computing power, the transport of fluid flow in porous media can be modeled with ease by using numerical reservoir simulation (NRS). Thereafter, petroleum engineers can utilize NRS to perform optimization more conveniently. Moreover, the results yielded by running different cases on NRS provide additional insight for the engineers to formulate their plans for FD and RM. Despite this, NRS encounters computational issues when the reservoir modeled is geologically sophisticated. This implies that running one scenario of NRS is computationally expensive and this might cause inconvenience to obtain a fast solution for RM when plans are updated at a high frequency. Moreover, this computational challenge will be further

<sup>\*</sup> Corresponding author.

E-mail address: [cuthbert.s.w.ng@ntnu.no](mailto:cuthbert.s.w.ng@ntnu.no) (C.S.W. Ng).

<https://doi.org/10.1016/j.acags.2022.100103>

Received 3 February 2022; Received in revised form 27 September 2022; Accepted 8 October 2022

Available online 10 October 2022

2590-1974/© 2022 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

exacerbated if several geological realizations are needed for robust production optimization, which means production optimization under geological uncertainty as discussed in Hong et al. (2017). Therefore, to reduce the computational cost, proxy modeling is suggested as one of the alternative solutions.

Proxy modeling, which is also surrogate modeling or meta-modeling (Zubarev, 2009), is the development of one or more models that can be applied as a substitute for a base model (NRS). Also, proxy modeling is mostly data-driven, and its building block mainly stems from data. Therefore, data must be acquired before proceeding to the establishment of proxy models. This explains why proxy modeling can alternatively be termed data-driven modeling. Besides that, there are generally two classes implemented to establish data-driven models, namely statistics-based and machine learning-based (ML-based) methods. The use of the statistical method in proxy modeling has been extensively discussed in different petroleum-related pieces of literature and the relevant examples comprise response surface methodology (also known as polynomial regression) (Babaei and Pan, 2016; Olabode et al., 2018) and kriging (Fursov et al., 2020; Hamdi et al., 2021). Apart from data-driven methods, the reduced physics approach is another option for proxy modeling. Regarding the reduced physics approach, the capacitance resistance model, proposed by Bruce (1943), is one of the epitomes. It has been extensively investigated and employed in petroleum engineering as discussed in several works of literature (Hong et al., 2017; Yousefi et al., 2021). Albeit these approaches have demonstrated fruitful results, some literature (Mohaghegh, 2017; Zubarev, 2009) also briefed their limitations in proxy modeling. Mohaghegh (2017) expounded that the reduced physics method requires simplification of the physics and assumptions in terms of modeling an actual system. Zubarev (2009) investigated the performance of 4 different proxy modeling techniques, such as response surface method, thin-plate splines, kriging, and artificial neural network. He deduced that in terms of proxy modeling, kriging would require higher computational effort whereas the response surface method would decrease the precision of prediction.

This paper mainly sheds light on the application of ML-based methods. ML is defined as a computer algorithm that can enhance the performance of a model through experience, reflected by data (Mitchell, 1997). Examples of ML are, but are not circumscribed to, artificial neural network, gradient boosting machine, support vector machine, k-nearest neighbor, and random forest. ML has been evidenced to be useful in different domains of knowledge, including speech recognition (Nassif et al., 2019; Seehapoch and Wongthanavasu, 2013) and image analysis (Komura and Ishikawa, 2018; Poostchi et al., 2018). Furthermore, the implementation of ML has been widely generalized in different aspects of petroleum engineering, specifically reservoir and production engineering. In this context, ML has displayed successful applications in numerous pertinent areas, such as the design of well trajectory (Kristoffersen et al., 2021, 2022), CO<sub>2</sub> sequestration (Nait Amar et al., 2020a; Nait Amar and Jahanbani Ghahfarokhi, 2020; Vo Thanh et al., 2022), history matching (He et al., 2016; Jo et al., 2022), and flow assurance issue (Benamara et al., 2019; Nait Amar et al., 2021a). ML-based proxy models have also been efficiently coupled with mathematical optimization algorithms in performing production optimization. In this aspect, some articles (Guo and Reynolds, 2018; Sen et al., 2021) have illustrated the application of ML techniques in robust production optimization. Besides that, the employment of derivative-free mathematical algorithms, which are generally nature-inspired, has been studied in some works (Nait Amar et al., 2020b, 2021b; Ng et al., 2021a). These nature-inspired algorithms have broadly been used due to their ability to converge to the global optimum in solution space (Ezugwu et al., 2020; Yang, 2014).

For further details, developing or training the ML-based proxy models is considered "learning". Precisely speaking, these models are attempting to learn by discovering the pattern of the data supplied. If the database provided is not updated throughout the process of

development, such training is generally termed "offline learning". Proxy models constructed from "offline learning" can occasionally yield a less optimal solution to an optimization problem due to lower prediction accuracy. Such an issue has been highlighted by Salehian et al. (2022) in which proxy models built from "online learning" are recommended as a possible improvement. According to Geng and Smith-Miles (2015), online learning shares the same definition as adaptive learning or incremental learning. This terminology expounds that this method involves a continuous update of the database. The fundamental idea lies in selecting the "useful" candidate to be added to the database used to train the data-driven model. Generally, generating this candidate involves sampling the data that fulfills predefined inflill criteria (Forrester et al., 2008; Liu et al., 2012, 2018; Xu et al., 2012). The metrics of these criteria include, but are not limited to, Expected Improvement, Lower Confidence Bound, and Probability of Improvement. Adaptive proxy modeling has been a common practice exercised during the implementation of a statistical-based approach as briefed and demonstrated in some published works (Forrester et al., 2008; Li et al., 2015; Liu et al., 2018; Redouane et al., 2019). Nonetheless, as Golzari et al. (2015) pointed out, for managing higher dimensional problems, ML-based methods generally illustrate higher aptitude in handling non-linearity in terms of time-series prediction. Therefore, in this work, we choose to utilize ML-based proxy models.

The workflow presented in this paper can be considered as a variant of surrogate-based global optimization (SBGO), perceived as the simultaneous application of adaptive sampling and optimization with the aid of a global optimizer (Ye and Pan, 2019). In simpler terms, it performs as a hybridization of training and optimization. As outlined in Ye and Pan (2019), SBGO revolves around the employment of statistical approaches to develop the proxies and derivative-free algorithms as optimizers. However, we discuss and illustrate the use of ML-based proxy models here instead. Concerning this, the ML technique demonstrated in this work consists of multilayer perceptrons (MLP). Moreover, the developed proxy models aim to conduct robust production optimization under waterflooding. Therefore, these models are coupled with nature-inspired algorithms to conduct the optimization. Two examples of nature-inspired algorithms were selected, viz. particle swarm optimization (PSO) and grey wolf optimizer (GWO). As discussed in this paper (Yang, 2014), nature-inspired algorithms generally achieve a balance between exploration and exploitation over the search space. Exploration means the diversification of solutions in the search space whereas exploitation refers to a more focused search on a local region. A good combination of both, which is generally achieved by nature-inspired algorithms (considering the algorithms are optimally tuned), usually avoids the convergence to local optima. Slightly different from the general practice in SBGO, the candidate (adaptively chosen to be added to the database) is retrieved from the results of the iterative optimization with the proxy model. Concerning this, based on our studies done in this work, using these optimal results, which are obtained from the proxy model with the help of nature-inspired algorithms, to re-train the proxy model has the potential to increase its fidelity. The pertinent details will be revealed under the section of Results and Discussion.

After this introduction, the paper is formulated as shown: Section 2 briefs the basic theoretical concepts of MLP, PSO, and GWO regarding some of our previous works (Ng et al., 2021a, 2021b, 2022a, 2022b). Thereafter, section 3 provides a comprehensive explanation of the methodology applied to develop the proxy models in this work. Section 4 expounds on the results yielded and the relevant discussion. Then, the main findings are conclusively summarized.

## 2. Previous related works

The methodology presented in this work was established based on the insights gained from our previous works (Ng et al., 2021a, 2021b, 2022a, 2022b). Since this paper is considered an extension of these

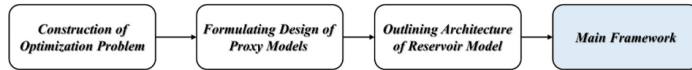


Fig. 1. General workflow implemented in this work.

previous works, the 3D Egg Reservoir Model that has been used in Ng et al. (2021a) was selected as the reservoir model for proxy modeling here. However, the proxy models built here consider 10 different geological realizations. With geological uncertainty (only about permeability), well control optimization is conducted under water-flooding. The methodology was developed using Python (Van Rossun and Drake, 2009). For the modeling of MLPs, they were developed with the help of the Scikit-learn package (Pedregosa et al., 2011). PSO was formulated by applying the toolkit built by James V. Miranda (2018) whereas GWO was constructed with the toolkit by Lickevic and Bartoshevic (2021).

2.1. Multilayer perceptron (MLP)

It is unassailable that artificial neural network (ANN) is one of the most prominent ML techniques used in a wide variety of domains (Lopez-Garcia et al., 2020; Runge and Zmeureanu, 2019). The method has demonstrated its excellent performance in learning how the input data is related to output data for any physically sophisticated process. Biological neural networks in the brain are mainly the inspiration for its formulation (Rosenblatt, 1958). MLP is one of the most widely employed variants of ANN in building data-driven models (Buduma and Locascio, 2017). In essence, MLP consists of many artificial neurons or calculating nodes. MLP also comprises three types of layers, viz. the input layer, the hidden layer, and the output layer. Each layer has its neurons in which these neurons are interconnected with the use of weights and biases. For more information about the mathematical implementation of MLP, refer to our previous works (Ng et al., 2021a, 2021b, 2022b). The training process for MLP typically involves finding the optimal values of weight and bias sets to minimize the predefined loss function, such as mean squared error (MSE) and mean absolute percentage error. MSE was selected as the loss function whereas Adam (Adaptive Moment Estimation) was applied for training. For the details of Adam, peruse the literature (Kingma and Ba, 2015).

2.2. Nature-Inspired Algorithms

Kennedy and Eberhart (1995) proposed PSO that attempts to simulate the behavior of flying stock of birds. A swarm of particles mathematically represents some possible solutions to an optimization problem. The status of each particle is calculated by using its position and velocity. About the mechanism of PSO, random initialization of the position and velocity of each particle is first done. Thereafter, to calculate the fitness of every particle, a cost function is required. Upon computing the fitness, the local and global best positions of a particle are determined to update the velocity at the current step. After assessing the velocity at the next iteration, the position of a particle for the next iteration is updated. As several iterations complete, each particle updates its position by minimizing the fitness value until the convergence of the optimal position occurs.

Mirjalili et al. (2014) developed GWO based on the inspiration of the leadership hierarchy and hunting behavior of grey wolves. Fundamentally, the population of grey wolves is divided into four different groups, e.g., alpha ( $\alpha$ ), beta ( $\beta$ ), delta ( $\delta$ ), and omega ( $\omega$ ). Among all,  $\omega$  wolves are the most inferior and preceded by  $\delta$ ,  $\beta$ , and  $\alpha$ . Mathematically, a wolf population represents a set of random solutions. Thereafter, the fitness value of each solution set is evaluated by using a predefined objective function (Xu et al., 2020). According to the fitness value, the population of wolves is divided into the four previously mentioned groups. As optimization commences, the three best wolves:  $\alpha$ ,  $\beta$ , and  $\delta$ , would gradually lead the other  $\omega$  wolves towards the prey, which is treated as the global solution in the search space. This is done by iteratively updating the positions of the wolves. These algorithms are preferred in this work due to their good performance in our previous studies (Ng et al., 2021a, 2022a), where they demonstrated improved optimization results compared to the base case when they were coupled with the proxy models. For more details about the algorithms of both PSO and GWO, please peruse these articles (Kennedy and Eberhart, 1995; Mirjalili et al., 2014; Ng et al., 2021a; Xu et al., 2020).

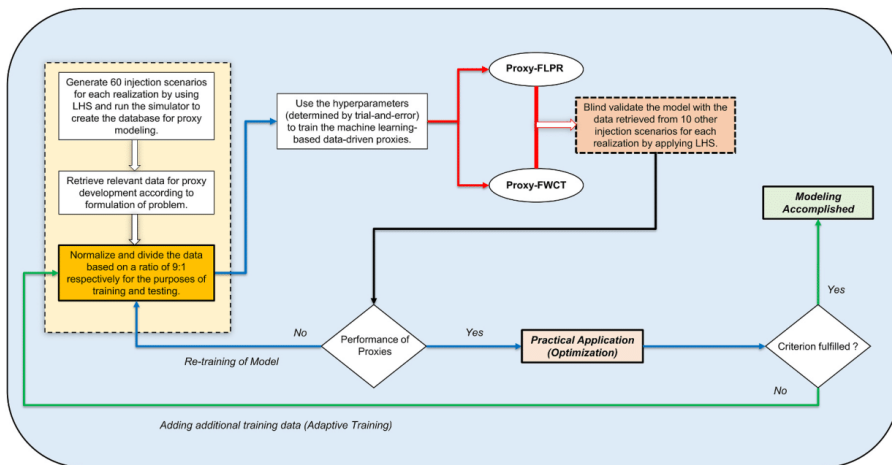


Fig. 2. Details of the main framework (Backbone of AP-ROpt).



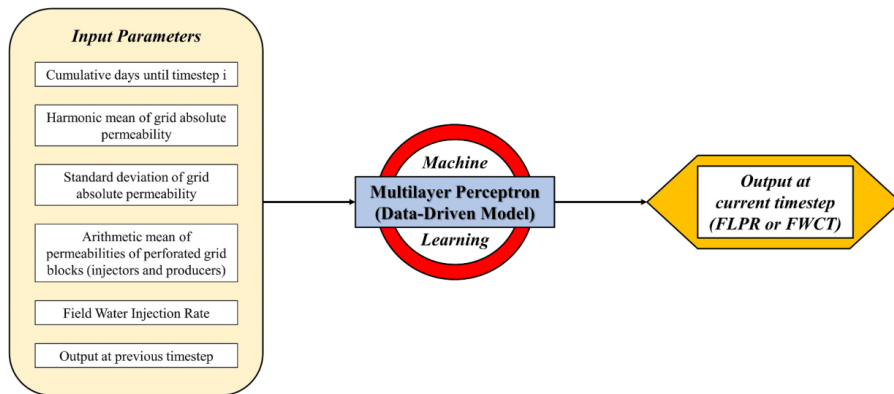


Fig. 3. Diagram of input and output parameters.

3. Methods and materials

For convenient articulation, the methodology proposed here is termed “Adaptive Proxy-based Robust Production Optimization” (AP-ROpt). The general workflow of implementing the AP-ROpt is illustrated in Fig. 1, consisting of 4 steps. The Main Framework can be further categorized into 2 parts, viz. Establishment of Proxy Models and Optimization. The details of the Main Framework are illustrated in Fig. 2.

3.1. Formulating the optimization problem

A database is one of the most essential elements in data-driven modeling. Before acquiring the database, it is of great importance for modelers to clarify and define the functionality of the data-driven models since data-driven proxy modeling is objective-oriented. Therefore, modelers should perceive what engineering problem is to be solved via the use of proxy models. In this paper, the engineering problem defined is the optimization of well control under waterflooding which is similar to the one discussed in Ng et al. (2021a). Thus, only the field water injection rate is considered as the control parameter. The objective function used in the optimization is the expected net present value (ENPV) as shown:

$$ENPV(\mathbf{u}) = \frac{\sum_{t=1}^{n_r} \left( \sum_{i=1}^{n_{total}} \frac{\Delta t_i \times (Q_{i,oil}(\mathbf{u})P_{oil} - Q_{i,water\ prod}(\mathbf{u})P_{water\ prod} - Q_{i,water\ inj}(\mathbf{u})P_{water\ inj})}{(1 + interest\ rate)^{t/365}} \right)}{n_r} \quad (1)$$

Based on the objective function,  $n_r$  is the total number of realizations which is 10 here,  $\mathbf{u}$  represents the control vector,  $Q_i$  indicates the field rates of produced oil, produced and injected water at timestep  $i$ ,  $P$  means

the respective price. In addition,  $\Delta t_i$  (unit in days) is the time difference between timestep  $i$  and previous timestep,  $t_i$  (unit in days) is the cumulative time until timestep  $i$ , and the reference period for discounting cash flow is 365 days. The oil price is 440.3 USD/m<sup>3</sup>, the cost of handling water produced, and water injection is 12.58 USD/m<sup>3</sup>, and the interest rate is 0.10 per year.

3.2. Design of proxy models

Having explicitly defined the optimization problem, modelers would have better insights into what parameters can be yielded by the proxy models, directly or indirectly. More importantly, the decision variables (optimization parameters) are treated as one of the inputs of the proxy models. According to Eq. (1), the parameters required from the proxies are field oil and water production rates whereas field water injection rates act as decision variables. To attain this goal, we followed the ideas based on our previous studies and investigation in which two different proxy models were built. One of them can forecast the field liquid production rates (FLPR) at a certain timestep given a timeframe whereas another one has the same functionality in terms of field water cut prediction (FWCT). For both proxies, the input parameters comprise the cumulative days until timestep  $i$ , self-defined geological parameters, field water injection rate (decision variables), and the output at the previous timestep,  $y_{i-1}$ . About the self-defined geological parameters, they comprise the harmonic mean (and standard deviation) of grid absolute permeability for every reservoir layer as well as the arithmetic mean of permeabilities of perforated grid blocks (injectors and producers). This corresponds to 29 input parameters and 1 output parameter. The input parameters were selected based on our knowledge of

Table 1 Summary of the initial database for the development of proxy models.

Types of Data	Number of Data Points	Maximum Value	Minimum Value	Mean Value	Standard Deviation
Static Data					
Cumulative days until timestep $i$	1 × 60,000	3000	30	1515	865.98
Harmonic mean of grid absolute permeability	7 × 60,000	749.41	577.57	641.71	37.88
Standard deviation of grid absolute permeability	7 × 60,000	1701.24	654.44	1149.07	252.72
Arithmetic mean of permeabilities of perforated grid blocks (injectors)	8 × 60,000	3994.57	132.99	1109.78	963.85
Arithmetic mean of permeabilities of perforated grid blocks (producers)	4 × 60,000	5000	200	1581.12	1372.47
Dynamic Data					
Field Water Injection Rate	1 × 60,000	800	320	559.96	138.34
Previous Output and Current Output (FLPR)	2 × 60,000	798.67	0	557.24	143.43
Previous Output and Current Output (FWCT)	2 × 60,000	1	0	0.7067	0.3401

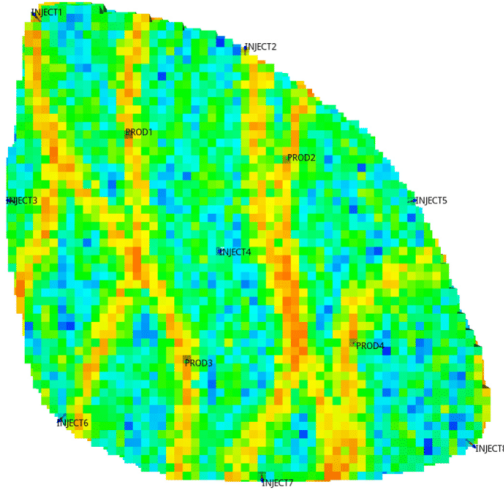


Fig. 4. Horizontal permeability distribution of Realization 1 of 3D Egg Model with labeled well locations. The same locations apply to all realizations. The warm color indicates higher horizontal permeability whereas the cold color implies the otherwise.

reservoir engineering and insights gained from previous studies. Refer to Fig. 3 for the diagram of input and output parameters and Table 1 for the initial database used for training. The number 60000 in Table 1 was determined by having 60 injection scenarios  $\times$  100 timesteps  $\times$  10 realizations. According to the insights gained from our previous works (Ng et al., 2021a, 2022a), 60 scenarios have been illustrated to be adequate to produce the proxy models with a good degree of accuracy. Therefore, the same number of scenarios is applied in this study. Readers are also referred to Ng et al. (2021a) for more comprehensive information about the formulation of the proxy models.

### 3.3. Outlining architecture of reservoir model

As mentioned, the reservoir model implemented in this paper is the Egg Model and the simulation was performed using the Eclipse 100 software (Schlumberger, 2019). This model is a benchmark case, developed by Jansen et al. (2014) for research purposes. The Egg Model has 7 layers, and it is built as a channelized depositional system. It also has eight injectors and four producers in which the trajectory of each well is vertical. The well configuration is shown in Fig. 4. Peruse Jansen et al. (2014) and Ng et al. (2021a) for the details of the topology of the reservoir model. Regarding the details of optimization, it involves adjustment of field water injection rates within 320 and 400 Sm<sup>3</sup>/day (each injector has an equal allocation of the total rate) by having the maximum bottomhole pressure of each producer set at 395 bars. This adjustment is done every 150 days over 3000 days of the production period. This results in 20 control variables. However, the proxy models have been designed to consider a timestep of 30 days and every control variable remains the same for 5 timesteps (150 days). Therefore, during optimization, 100 variables are involved. We have considered 10 realizations in this work and the corresponding reservoir architecture of each realization is presented in Fig. 5.

### 3.4. Main Framework

#### 3.4.1. Establishment of proxy models

In the establishment of proxy models, the generation of a training

database often comes first. Here, we implemented Latin Hypercube sampling (LHS) to create 60 sample sets of control rates in which one set represents one injection scenario. In this case, these 60 scenarios are the same for every realization. Peruse McKay et al. (1979) for the details of LHS. Each scenario was then sent to the reservoir simulator to produce the simulation outputs. Upon the completion of 600 simulations, the dynamic inputs were retrieved and merged with the static inputs to develop the database. Normalization of the database is a highly recommended practice before being fed into the training phase of ML models. The database was normalized between 0 and 1 according to the formula below:

$$X_{\text{normalized}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (2)$$

where  $X_{\text{normalized}}$  is the normalized value of  $X$ .  $X_{\max}$  and  $X_{\min}$  correspond to the maximum and minimum values of  $X$ , respectively. It is important to note that during the adaptive training, an additional sample has been included in the training database. Therefore, the values of  $X_{\max}$  and  $X_{\min}$  also need to be updated (considering all input and output parameters) and normalization is repeated. Thereafter, the normalized database was partitioned into training and testing with a ratio of 9:1. Regarding this partition of data, only the training data is employed to establish the models. Since the package of scikit-learn was selected, within the training data, a portion of it would be extracted to conduct the validation phase. Concerning this, MLP would undergo a validation phase in which 1/9 of the 90% training data was treated as the validation set. Nevertheless, evaluation of the developed models was performed meticulously to ensure that the overfitting issues had been eliminated.

Regarding the topology of proxy models and hyperparameters used in the training, the values were slightly different for both FLPR and FWCT. For FLPR, the learning rate was 0.001, the number of hidden layers was 4 (each layer had 50 hidden nodes), and tolerance was  $10^{-6}$ . For FWCT, the learning rate was 0.005, the number of hidden layers was 4 (each layer had 15 hidden nodes), and tolerance was  $10^{-6}$ . Rectified Linear Unit (ReLU) was implemented as an activation function for all layers. Considering an arbitrary function of  $f(x)$ , ReLU is mathematically expressed as  $f(x) = \max(x, 0)$ . The maximum number of iterations for both models was defined as 1000 in which the early stopping mechanism was activated. These setting parameters were decided via a trial-and-error approach. After the training and testing phases, the data-driven proxies must be blind-validated before being practically employed. In this aspect, data of blind validation should be independent of the above-mentioned database. Hence, we implemented LHS to generate 10 other injection scenarios for each realization (a total of 100 blind validation cases) to be fed into the reservoir simulator to yield the relevant outputs, which would then be compared with the outputs predicted by the proxy models. The comparative result is a deciding factor to evaluate if the proxies should either undergo re-training or proceed to optimization. If the performance of proxies is not up to certain quality, then re-training will be done. In this paper, two statistical metrics were chosen to assess the performance of proxies, viz. coefficient of determination ( $R^2$ ) and root mean squared error (RMSE). The formula of metrics is as follows in Eqs. (3) and (4).

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i^{\text{pred}} - Y_i^{\text{real}})^2}{\sum_{i=1}^n (Y_i^{\text{pred}} - \bar{Y})^2} \quad (3)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (Y_i^{\text{pred}} - Y_i^{\text{real}})^2}{n}} \quad (4)$$

where  $n$  represents the total number of data points,  $i$  denotes the index of data points,  $Y_i$  is the corresponding output, the superscripts pred and real represent the proxy model and reservoir simulator model, respec-

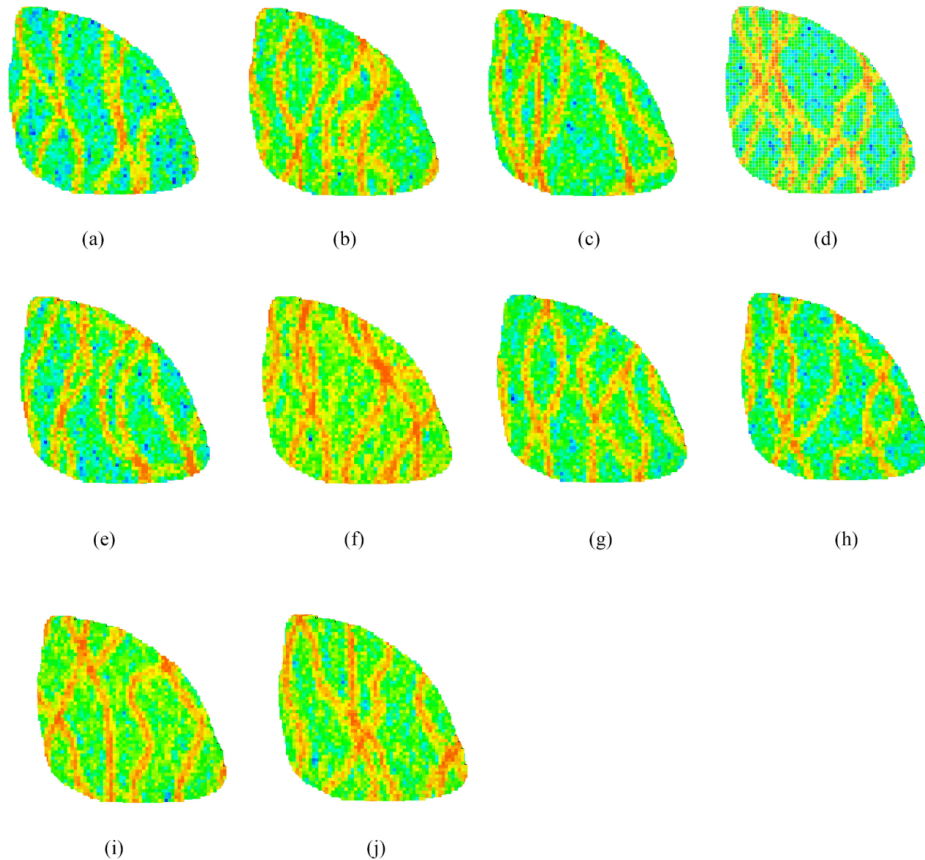


Fig. 5. Horizontal permeability distribution of 10 different geological realizations of the 3D Egg Model. The warm color indicates higher horizontal permeability whereas the cold color implies the otherwise. (a) to (j) respectively refer to Realization 1 to 10.

tively. Also,  $\bar{Y}$  indicates the mean value of output. Re-training is only needed if the mean  $R^2$  values of blind validation<sup>1</sup> for FLPR and FWCT are less than predefined values. In this case, the predefined values for FLPR and FWCT were decided to be 0.998 and 0.970, respectively via trial-and-error.

#### 3.4.2. Optimization with proxy models and reservoir simulation

In the phase of optimization, PSO and GWO were applied to determine the optimal well control. In the case of proxy models, as the optimization iterations were completed, the proxy-optimized control would be obtained and treated as a new injection scenario to be fed into the reservoir simulator. The response of the simulator was again compared with that of the proxy. If the criterion check was satisfied, then the whole workflow was considered complete. Otherwise, the optimal control would be treated as a new dynamic input to be added to the training database. The loop of workflow would then start again. It

would only cease if the criterion were fulfilled, or the number of additional simulations exceeded a predefined value. In this study, the average between the mean<sup>2</sup>  $R^2$  of FWPR and FOPR was used as the criterion check. The predefined threshold was arbitrarily set as 0.994. About the parameters used in PSO, the inertia weight was 0.80 whereas the cognitive and social learning factors were 1.05.  $r_1$  and  $r_2$  were sampled from a uniform distribution of (0, 1). For GWO, the default parameters set by [Lickevic and Bartoshevich \(2021\)](#) were applied. For PSO (GWO), 100 iterations and 20 swarm particles (100 iterations and 20 populations) were employed. These optimization algorithms were not only implemented in this workflow for proxy models but also coupled with the reservoir simulator. The details of the results would be presented and discussed in the following section.

## 4. Results and Discussion

Before outlining a holistic discussion about the findings of this work,

<sup>1</sup> Mean  $R^2$  of blind validation refers to the arithmetic average of 100  $R^2$  values (each calculated over 100 timesteps) as 10 blind validation scenarios are considered for each of the 10 realizations.

<sup>2</sup> The term "mean  $R^2$ " refers to the arithmetic mean of 10 values of  $R^2$  as 10 realizations were used to develop the proxy models.

**Table 2**  
Results of training, testing, and blind validation of the developed proxy models.

		ROpt- MLP-FLPR	ROpt-MLP-FWCT
Training	R <sup>2</sup>	0.9999	0.9995
	RMSE	0.9288	0.0073
Testing	R <sup>2</sup>	0.9999	0.9995
	RMSE	0.9485	0.0074
Blind Validation	R <sup>2</sup>	0.9999	0.9872
	RMSE	0.9459	0.0328

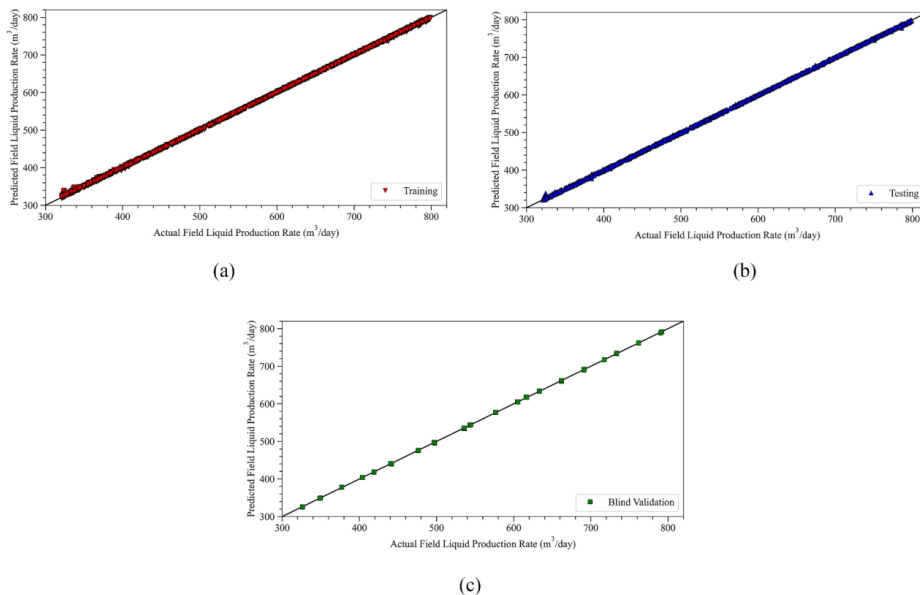
we illustrate the results of the training, testing, and blind validation phases as shown in Table 2 to provide a better insight into the performance of the developed proxy models. Table 2 consists of two statistical metrics, namely R<sup>2</sup> and RMSE, that have been useful to reflect the accuracy of the proxy models built in this work. For better illustrative purposes, the cross plots between the actual and predicted data in each phase of proxy modeling are demonstrated in Fig. 6 for FLPR and in Fig. 7 for FWCT. In terms of training, testing, and blind validation, MLP-FLPR generally displays better performance than MLP-FWCT. Despite this, the results obtained by MLP-FWCT have sufficiently confirmed its reliability for further employment.

Upon completing the modeling part, these models are readily employed for adaptive learning and optimization. In this aspect, the models would be correspondingly coupled with PSO and GWO to determine the optimal field injection rates within the range as previously explained. For benchmarking, we also coupled these two algorithms with E100 software to perform the optimization with NRS models. The optimal control determined using the simulator and proxy models are correspondingly shown in Figs. 8 and 9. Though there may be low proximity between the optimal control yielded by these two approaches, we would like to emphasize that the main objective here is to create substitute models that can achieve an optimized objective function close to the “ground truth” (generated by the NRS) at much less computational cost.

Thereafter, the proxy-optimized control rates were fed back into the reservoir simulator to yield the necessary parameters for the calculation of ENPV. By acquiring the results, the ENPVs for three cases of reservoir simulator, simulator-proxies (referring to the results in which the optimal control derived from only using the proxy models, is fed back to the simulator), and proxies are computed and recorded in Table 3. Under an assumption of a base case with a maximum constant field injection rate, the ENPV of the base case is 155.76 million USD. During the optimization with the reservoir simulator, GWO resulted in a better improvement on ENPV with 3.75% as PSO only enhanced the ENPV by 2.76%. A similar outcome is also illustrated in the case of simulator-proxies. In terms of optimization, this generally shows that GWO slightly outperforms PSO in this study. For better purposes of illustration and comparison, the optimized NPVs of each realization for the cases of simulator and simulator-proxies (considering cases that involve the use of simulator) are correspondingly illustrated in Figs. 10 and 11.

Regarding the accuracy of results, it can be noted that GWO records a lower percentage error between the two ENPVs produced by simulator-proxies and dynamic proxies, which is about 0.20% whereas that of MLP-PSO is 0.90%. For both algorithms, the differences between the ENPVs of simulator and simulator-proxies are practically small. Nevertheless, GWO records ENPV of simulator-proxies that is closer to the “ground truth” (ENPV of reservoir simulator). Thus, proxy models coupled with GWO yielded slightly more accurate results than those of PSO in this work. Despite this, PSO still portrayed promising applicability due to its practically good accuracy of the result attained. This further enlightens us that the proxy models built here have sufficient capability to provide solutions for this optimization problem. For illustrative purposes, the plots of the optimized field water and oil rates for each optimization case considering 10 realizations are presented in Figs. 12 and 13, respectively.

Table 4 is displayed for closer scrutiny in both Figs. 8 and 9. These metrics are calculated by correspondingly comparing FWPR and FOPR generated by simulator-proxies and proxy models. According to Table 4,



**Fig. 6.** Cross plot between the actual FLPR and FLPR predicted by the proxy model. a) Training, b) Testing, and c) Blind Validation (only illustrating 1 blind validation scenario in Realization 2).

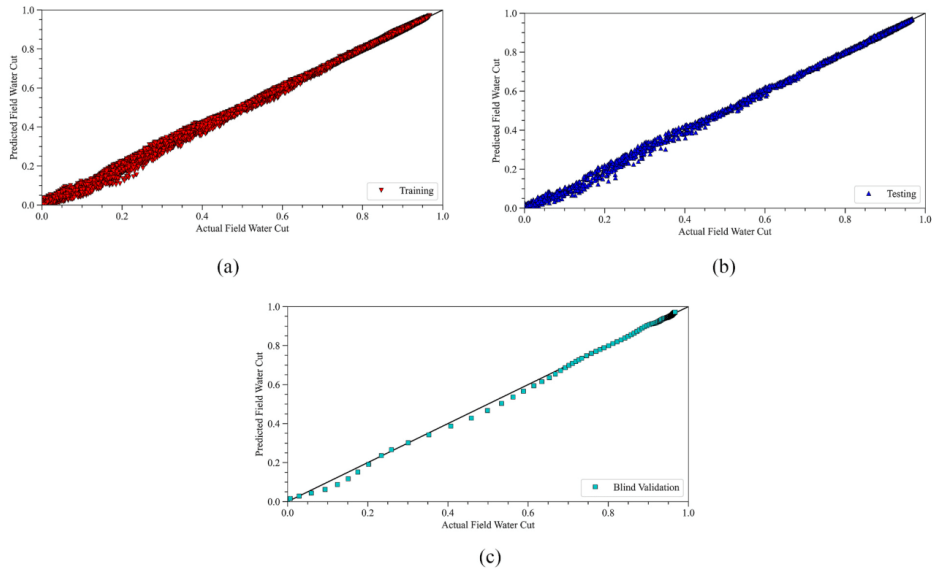


Fig. 7. Cross plot between the actual FWCT and FWCT predicted by the proxy model a) Training, b) Testing, and c) Blind Validation (only illustrating 1 blind validation scenario in Realization 2).

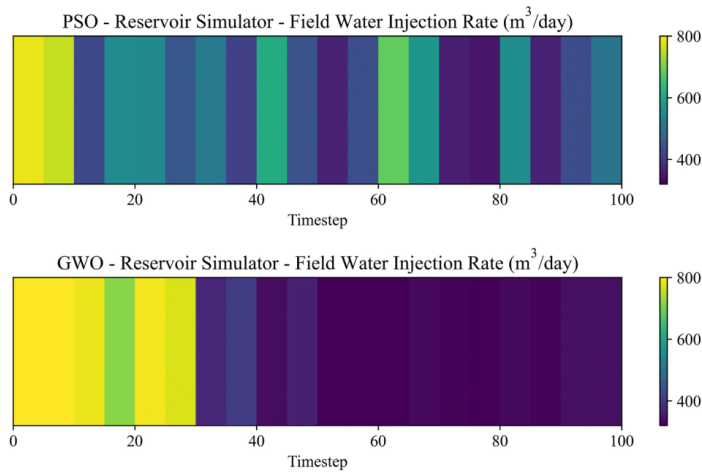


Fig. 8. Optimized control of Field Water Injection Rate (FWIR) resulted by coupling reservoir simulator with nature-inspired algorithms.

It can be opined that the worst performing realizations in the cases of FWPR and FOPR still produced results within a satisfactory level of accuracy. This confirmed the good applicability of the workflow proposed here. Considering all 10 realizations, Table 5 presents the mean  $R^2$  and RMSE (considering 10 different geological realizations) between the optimized FWPR (and FOPR) generated by simulator-proxies and proxy models. Based on these results, MLP-GWO showed a closer approximation of the results. Also, these results proved that the developed proxy models successfully served their purpose of application.

The proxy modeling and optimization were done by using a PC with Intel® Core™ i9-9900 CPU @3.10 GHz (64.0 GB RAM) (Ng et al., 2021a). Regarding computational time, both MLP-GWO and MLP-PSO have exhibited excellent computational efficiency. In this case, MLP-GWO spent about 13 h performing adaptive training and optimization whereas MLP-PSO used about 16 h. In addition, about the number of additional simulations induced, MLP-PSO has adaptively employed 66 additional simulations for the extension of the training database. For MLP-GWO, it adaptively created 54 other simulations. For the

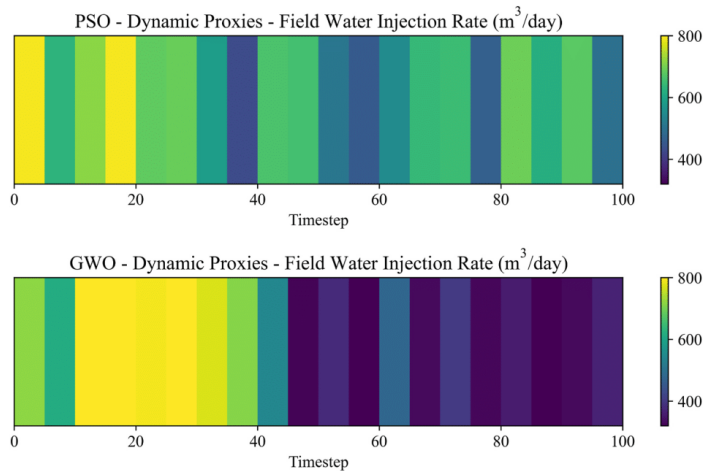


Fig. 9. Optimized control of Field Water Injection Rate (FWIR) resulted by coupling proxy models with nature-inspired algorithms.

**Table 3**  
Optimized ENPV of three cases considering PSO and GWO (in the unit of million USD).

Optimization Algorithm	Reservoir Simulator	Simulator-Proxies	Dynamic Proxies
PSO	160.06	158.93	160.37
GWO	161.60	159.80	159.48

optimization with the simulator E100, PSO required 159 h and GWO needed 238 h. Based upon this, GWO generally reflects a more significant added value of the application of proxy models in this work.

We would like to reiterate that the primary aim of the established proxy models is to locate the optimal solution to the waterflooding optimization problem. In this case, the optimal solution provided by these data-driven models results in an objective function that is close to

the one obtained by applying only the reservoir simulator. We also fathom that there are a few limitations regarding the workflow proposed here. Hyperparameter (topology of MLP) optimization is one of them. During adaptive training, when additional data is retrieved from additional simulation and added to the training database for proxy modeling, there is a possibility that the predefined hyperparameters are less reliable in achieving more accurate training results. However, integrating hyperparameter optimization can certainly induce higher computational effort. Despite having excellent results in this work, achieving a good trade-off between accuracy and computational time (considering hyperparameter optimization) certainly needs to be researched to increase the applicability of this methodology. Besides that, another shortcoming concerns the tuning parameters of the algorithms. These parameters were decided via a trial-and-error approach which could be subject to a degree of limited sensitivity. There is also another discussion about the impact of random number generators on the whole

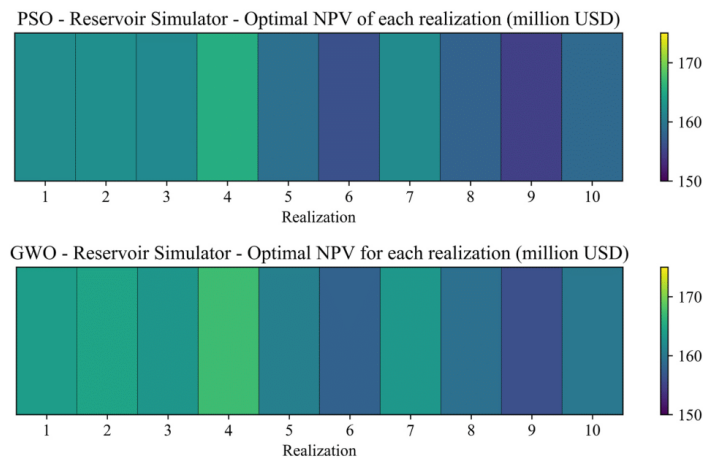


Fig. 10. Optimized NPV of each realization (reservoir simulator).

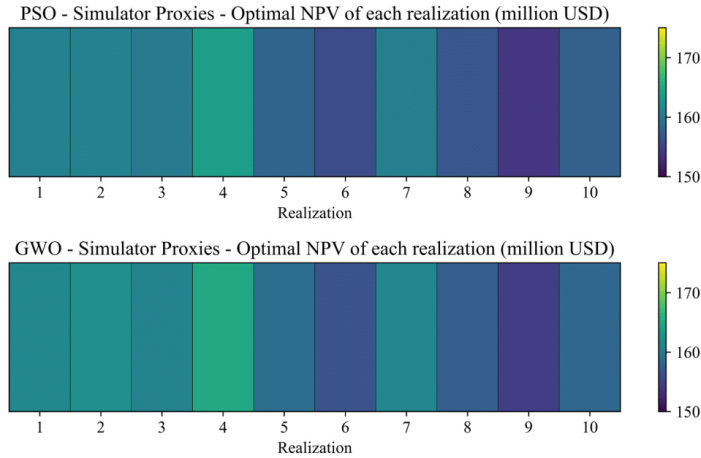


Fig. 11. Optimized NPV of each realization (simulator-proxies).

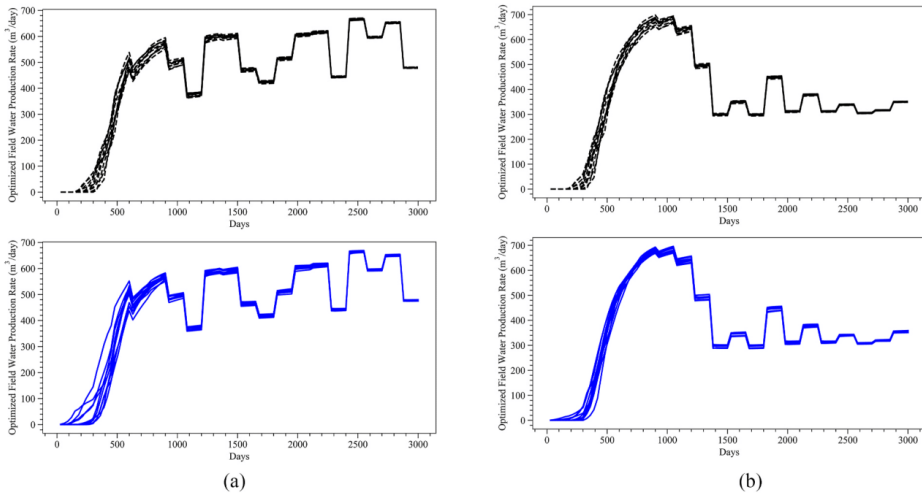


Fig. 12. Plots of the optimized FWPR considering 10 realizations. The black dashed lines indicate the case of simulator-proxies whereas the blue lines imply the cases of proxy models. a) PSO and b) GWO.

framework. Therefore, an in-depth study on tuning parameters and random number generators is needed to further reinforce the maturity of the workflow discussed here.

Apart from these, we only considered 10 realizations in this work since there is an apparent computational challenge arisen when more realizations are included in the methodology of the workflow. Hence, integrating the dimensionality reduction technique (for instance, as proposed in this paper (Salehian et al., 2021) through the selection of representative realization via clustering method) into the workflow proposed here is another domain that can be pondered upon in the future. Also, the additional training data is generated “online” via optimization with proxy models. Albeit this additional data is gotten through nature-inspired algorithms, the accuracy of proxy models might cause premature convergence to local optima. The accuracy of proxy

models is influenced by the complexity of the optimization problem being solved. Thus, this subject is upon consideration for further research when it comes to more sophisticated real-life applications. Furthermore, proxy models are often case-dependent and hence, the models built here can only be implemented to solve the optimization problem discussed here. Hence, modifications of the methodology are likely inevitable and require further investigation to instill higher confidence in application in future studies. In short, through this study, we aimed at developing a methodology that serves as a foundation for further enhancement in the future.

5. Summary and Conclusions

In this paper, we implemented the AP-ROpt that adaptively retrieved

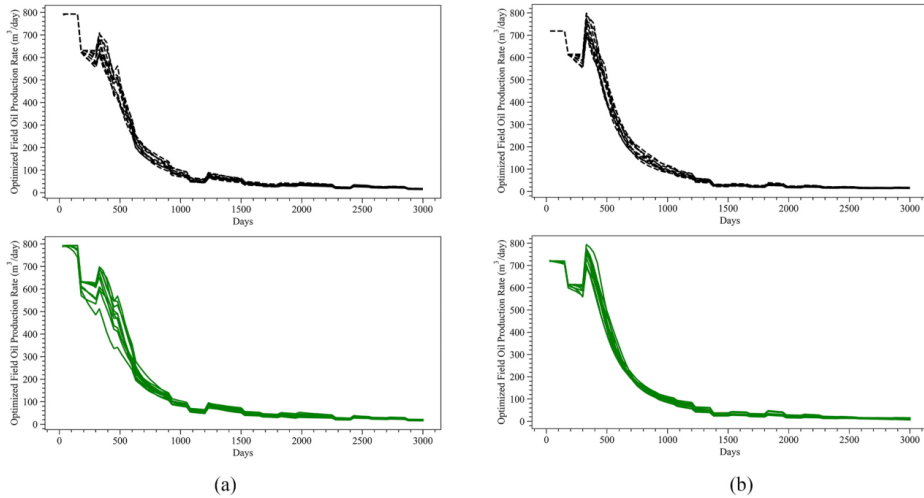


Fig. 13. Plots of the optimized FOPR considering 10 realizations. The black dashed lines indicate the cases of simulator-proxies whereas the green lines imply the cases of proxy models. a) PSO and b) GWO.

**Table 4**  
Performance metrics of the best and worst performing realizations of FWPR and FOPR under the case of proxy models along with PSO and GWO.

		R <sup>2</sup>	RMSE
MLP-PSO (FWPR)	Best Realization: 6	0.9983	7.44
	Worst Realization: 5	0.9651	34.63
MLP-PSO (FOPR)	Best Realization: 7	0.9995	5.42
	Worst Realization: 5	0.9756	34.64
MLP-GWO (FWPR)	Best Realization: 8	0.9989	6.30
	Worst Realization: 9	0.9898	18.67
MLP-GWO (FOPR)	Best Realization: 8	0.9992	6.12
	Worst Realization: 9	0.9930	18.66

**Table 5**  
Mean R<sup>2</sup> and RMSE of optimized FWPR (and FOPR) generated by comparing the results of simulator-proxies and proxy models.

		ROpt-FWPR	ROpt-FOPR
MLP-PSO	Mean R <sup>2</sup>	0.9932	0.9954
	Mean RMSE	13.59	13.11
MLP-GWO	Mean R <sup>2</sup>	0.9956	0.9972
	Mean RMSE	11.21	11.31

the optimal control (resulted from optimization with the established proxy models) and added it to the training database to further enhance the performance of the proxy models. This methodology is inspired by some of our previous works. The whole workflow of the methodology was performed in a closed-loop manner. Regarding this, by using 10 different realizations of the 3D Egg Model as the reservoir model, we employed MLP, an ML technique, to build two different proxy models which respectively forecast FLPR and FWCT. Then, they were coupled with PSO and GWO to optimize ENPV through the adjustment of FWIR.

We first implemented a trial-and-error approach to determine the optimal topology of these proxy models. Based on the training, testing, and blind validation results, the performance of these models was validated to be apt for further application. After the execution of the methodology, the results confirmed that a near-optimal solution (as compared with the solution from optimization with only reservoir

simulation) could be achieved with much less computational demand. For PSO, the computation was improved by nearly 10 times whereas for GWO, it has become about 18 times faster. High reduction in computational efforts is the main advantage attained in this work. Nevertheless, we are still cognizant of the limitations of this methodology, including consideration of only geological uncertainty, integration of hyperparameter optimization, and limited applicability to other optimization problems, viz. CO<sub>2</sub> sequestration and history matching.

With this, we would like to summarize that a fundamental methodology has been built upon which further improvement can be maneuvered, and this highlights the benefit garnered from this work. Also, the proxy models established here have sufficiently achieved their goal of the application. About this, integrating adaptive training with optimization, which yields an excellent result of proxy modeling under geological uncertainty, is considered the key finding here. We hereby opine that this workflow can be practically useful to improve any developed data-driven model that yields optimization results with a low satisfying level of accuracy. Nonetheless, refinements can still be done when dealing with more real-life applications.

**Authors' contributions**

Cuthbert Shang Wui Ng: Conceptualization, Methodology, Modeling, Programming, Coding, Data Preparation and Analysis, Investigation, Writing, Editing. Ashkan Jahanbani Ghahfarokhi: Supervising, Methodology, Writing, Reviewing and Editing.

**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**

The authors are unable or have chosen not to specify which data has been used.



## Acknowledgment

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)).

## References

- Babaei, M., Pan, I., 2016. Performance comparison of several response surface surrogate models and ensemble methods for water injection optimization under uncertainty. *Comput. Geosci.* <https://doi.org/10.1016/j.cageo.2016.02.022>.
- Benamara, C., Nait Amar, M., Gharbi, K., Hamada, B., 2019. Modeling Wax Disappearance Temperature Using Advanced Intelligent Frameworks. *Energy and Fuels.* <https://doi.org/10.1021/acs.energyfuels.9b03296>.
- Bruce, W.A., 1943. An electrical device for analyzing oil-reservoir behavior. *Trans. AIME.* <https://doi.org/10.2118/943112-g>.
- Buduma, N., Locascio, N., 2017. *Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms, Designing Next-Generation Machine Intelligence Algorithms.*
- Ezugwu, A.E., Adeleke, O.J., Akinyelu, A.A., Viriri, S., 2020. A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. *Neural Comput. Appl.* <https://doi.org/10.1007/s00521-019-04132-w>.
- Forrester, A.L.J., Sobester, A., Keane, A.J., 2008. *Engineering Design via Surrogate Modelling : a Practical Guide.* J. Wiley.
- Fursow, I., Christie, M., Lord, G., 2020. Applying kriging proxies for Markov chain Monte Carlo in reservoir simulation. *Comput. Geosci.* <https://doi.org/10.1007/s10596-020-09968-z>.
- Geng, X., Smith-Miles, K., 2015. Incremental learning. In: Li, S.Z., Jain, A.K. (Eds.), *Encyclopedia of Biometrics.* Springer US, Boston, MA, pp. 912–917. [https://doi.org/10.1007/978-1-4899-7488-4\\_304](https://doi.org/10.1007/978-1-4899-7488-4_304).
- Golzari, A., Haghghat Sefat, M., Jamshidi, S., 2015. Development of an adaptive surrogate model for production optimization. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2015.07.012>.
- Guo, Z., Reynolds, A.C., 2018. Robust life-cycle production optimization with a support-vector-regression proxy. *SPE J.* <https://doi.org/10.2118/191378-PA>.
- Hamdi, H., Clarkson, C.R., Esmail, A., Sousa, M.C., 2021. Optimizing the Huff “n” Puff gas injection performance in shale reservoirs considering the uncertainty: a duvernay shale example. In: *SPE Reservoir Evaluation and Engineering.* <https://doi.org/10.2118/195438-PA>.
- He, Q., Mohaghegh, S.D., Liu, Z., 2016. Reservoir simulation using smart proxy in SACROC unit - case study. In: *SPE Eastern Regional Meeting.* <https://doi.org/10.2118/184069-MS>.
- Hong, A.J., Bratvold, R.B., Nævdal, G., 2017. Robust production optimization with capacitance-resistance model as precursor. *Comput. Geosci.* <https://doi.org/10.1007/s10596-017-9666-8>.
- International Energy Agency, 2021. *Global Energy Review 2021, Global Energy Review 2020.*
- James, V., Miranda, L., 2018. PySwarms: a research toolkit for particle swarm optimization in Python. *J. Open Source Softw.* <https://doi.org/10.21105/joss.00433>.
- Jansen, J.D., Fonseca, R.M., Kahrobaei, S., Siraj, M.M., Van Essen, G.M., Van den Hof, P. M.J., 2014. The egg model - a geological ensemble for reservoir simulation. *Geosci. Data J.* <https://doi.org/10.1002/gdj3.21>.
- Jo, S., Jeong, H., Min, B., Park, C., Kim, Y., Kwon, S., Sun, A., 2022. Efficient deep-learning-based history matching for fluvial channel reservoirs. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.109247>.
- Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: *IEEE International Conference on Neural Networks - Conference Proceedings.* <https://doi.org/10.4018/jijnfmp.2015010104>.
- Kingma, D.P., Ba, J.L., 2015. Adam: a method for stochastic optimization. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.*
- Komura, D., Ishikawa, S., 2018. Machine learning methods for histopathological image analysis. *Comput. Struct. Biotechnol. J.* <https://doi.org/10.1016/j.csbj.2018.01.001>.
- Kristoffersen, B.S., Silva, T.L., Bellout, M.C., Berg, C.F., 2021. Efficient well placement optimization under uncertainty using a virtual drilling procedure. *Comput. Geosci.* <https://doi.org/10.1007/s10596-021-10097-4>.
- Kristoffersen, B.S., Bellout, M.C., Silva, T.L., Berg, C.F., 2022. Reduced well path parameterization for optimization problems through machine learning. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.109523>.
- Li, W., Zhang, D., Lin, G., 2015. A surrogate-based adaptive sampling approach for history matching and uncertainty quantification. In: *Society of Petroleum Engineers - SPE Reservoir Simulation Symposium.* <https://doi.org/10.2118/173298-ms>, 2015.
- Lickevic, V., Bartoshevic, P., 2021. *SwarmPackagePy.*
- Liu, J., Han, Z., Song, W., 2012. Comparison of infill sampling criteria in kriging-based aerodynamic optimization. In: *28th Congress of the International Council of the Aeronautical Sciences 2012.* ICAS 2012.
- Liu, H., Ong, Y.S., Cai, J., 2018. A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design. *Struct. Multidiscip. Optim.* <https://doi.org/10.1007/s00158-017-1739-8>.
- Lopez-Garcia, T.B., Coronado-Mendoza, A., Domínguez-Navarro, J.A., 2020. Artificial neural networks in microgrids: a review. *Eng. Appl. Artif. Intell.* <https://doi.org/10.1016/j.engappai.2020.103894>.
- McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics.* <https://doi.org/10.2307/1268522>.
- Mirjalili, S., Mirjalili, S.M., Lewis, A., 2014. Grey wolf optimizer. *Adv. Eng. Software.* <https://doi.org/10.1016/j.advengsoft.2013.12.007>.
- Mitchell, Tom, 1997. *Machine Learning Textbook.* McGraw Hill.
- Mohaghegh, S.D., 2017. *Data-Driven Reservoir Modeling.* Society of Petroleum Engineers.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., 2020. Prediction of CO2 diffusivity in brine using white-box machine learning. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107037>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Zeraibi, N., 2020a. Predicting thermal conductivity of carbon dioxide using group of data-driven models. *J. Taiwan Inst. Chem. Eng.* <https://doi.org/10.1016/j.jtice.2020.08.001>.
- Nait Amar, M., Zeraibi, N., Jahanbani Ghahfarokhi, A., 2020b. Applying hybrid support vector regression and genetic algorithm to water alternating CO2 gas EOR. *Greenh. Gases Sci. Technol.* <https://doi.org/10.1002/ghg.1982>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Ng, C.S.W., 2021a. Predicting wax deposition using robust machine learning techniques. *Petroleum.* <https://doi.org/10.1016/j.petlm.2021.07.005>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Ng, C.S.W., Zeraibi, N., 2021b. Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.109038>.
- Nassif, A.B., Shahin, I., Attili, I., Azzeh, M., Shaalan, K., 2019. Speech recognition using deep neural networks: a systematic review. *IEEE Access* <https://doi.org/10.1109/ACCESS.2019.2896880>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2021a. Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization. *J. Pet. Explor. Prod. Technol.* <https://doi.org/10.1007/s13202-021-01199-x>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., Torstater, O., 2021b. Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application. *Nat. Resour. Res.* <https://doi.org/10.1007/s11053-021-09844-2>.
- Ng, C.S.W., Ghahfarokhi, A.J., Nait Amar, M., 2022a. Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm. *Petroleum.* <https://doi.org/10.1016/j.petlm.2021.12.008>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2022b. Well production forecast in Volve field: application of rigorous machine learning techniques and metaheuristic algorithm. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.109468>.
- Olabode, O.A., Orodu, O.D., Isehunwa, S.O., Mamudu, A., Rotimi, O.J., 2018. Effect of foam and WAG (water alternating gas) injection on performance of thin oil rim reservoirs. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2018.07.043>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* <https://doi.org/10.1016/j.jmlr.2012.06.004>.
- Poostchi, M., Silamut, K., Maude, R.J., Jaeger, S., Thoma, G., 2018. Image analysis and machine learning for detecting malaria. *Transl. Res.* <https://doi.org/10.1016/j.trsl.2017.12.004>.
- Redouane, K., Zeraibi, N., Nait Amar, M., 2019. Adaptive surrogate modeling with evolutionary algorithm for well placement optimization in fractured reservoirs. *Appl. Soft Comput. J.* <https://doi.org/10.1016/j.asoc.2019.03.022>.
- Rosenblatt, F., 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* <https://doi.org/10.1037/h0042519>.
- Runge, J., Zmeureanu, R., 2019. Forecasting energy use in buildings using artificial neural networks: a review. *Energy.* <https://doi.org/10.3390/en12173254>.
- Salehian, M., Sefat, M.H., Muradov, K., 2021. A robust, multi-solution framework for well placement and control optimization. *Comput. Geosci.* <https://doi.org/10.1007/s10596-021-10099-2>.
- Salehian, M., Haghghat Sefat, M., Muradov, K., 2022. Multi-solution well placement optimization using ensemble learning of surrogate models. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.110076>.
- Schlumberger, 2019. *Eclipse Reservoir Simulation Software Reference Manual.* schlumberger.
- Seehapoch, T., Wongthanavasu, S., 2013. Speech emotion recognition using support vector machines. In: *Proceedings of the 2013 5th International Conference on Knowledge and Smart Technology. KST.* <https://doi.org/10.1109/KST.2013.6512793>, 2013.
- Sen, D., Chen, H., Datta-Gupta, A., Kwon, J., Mishra, S., 2021. Machine learning based rate optimization under geologic uncertainty. *J. Pet. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2021.109116>.
- Van Rossum, G., Drake, F.L., 2009. *Python 3 Reference Manual.* CreateSpace, Scotts Valley, CA.
- Vo Thanh, H., Yasin, Q., Al-Mudhafar, W.J., Lee, K.-K., 2022. Knowledge-based machine learning techniques for accurate prediction of CO2 storage performance in underground saline aquifers. *Appl. Energy* <https://doi.org/10.1016/j.apenergy.2022.118985>.
- Xu, Q., Wehrle, E., Baier, H., 2012. Adaptive surrogate-based design optimization with expected improvement used as infill criterion. *Optimization* <https://doi.org/10.1080/02331934.2011.644286>.
- Xu, C., Nait Amar, M., Ghrga, M.A., Ouair, H., Zhang, X., Hasanipanah, M., 2020. Evolving support vector regression using grey wolf optimization; forecasting the

- geomechanical properties of rock. Eng. Comput. <https://doi.org/10.1007/s00366-020-01131-7>.
- Yang, X.-S., 2014. Chapter 1 - introduction to algorithms. In: Yang, X.-S. (Ed.), *Nature-Inspired Optimization Algorithms*. Elsevier, Oxford, pp. 1–21. <https://doi.org/10.1016/B978-0-12-416743-8.00001-4>.
- Ye, P., Pan, G., 2019. Surrogate-based global optimization methods for expensive black-box problems: recent advances and future challenges. In: *Proceedings - 2019 2nd International Conference of Intelligent Robotic and Control Engineering*. IRCE. <https://doi.org/10.1109/IRCE.2019.00026>, 2019.
- Yousefi, S.H., Rashidi, F., Sharifi, M., Soroush, M., Ghahfarokhi, A.J., 2021. Interwell connectivity identification in immiscible gas-oil systems using statistical method and modified capacitance-resistance model: a comparative study. *J. Pet. Sci. Eng.* 198, 108175 <https://doi.org/10.1016/J.PETROL.2020.108175>.
- Zubarev, D.I., 2009. Pros and cons of applying proxy-models as a substitute for full reservoir simulations. In: *Proceedings - SPE Annual Technical Conference and Exhibition*. <https://doi.org/10.2118/124815-ms>.


## **Paper 6**

### ***Fast Well Control Optimization with Two-Stage Proxy Modeling***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Wilson Wiranda

Article

# Fast Well Control Optimization with Two-Stage Proxy Modeling

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi \*  and Wilson Wiranda

Department of Geoscience and Petroleum, Norwegian University of Science and Technology, 7031 Trondheim, Norway

\* Correspondence: ashkan.jahanbani@ntnu.no

**Abstract:** Waterflooding is one of the methods used for increased hydrocarbon production. Waterflooding optimization can be computationally prohibitive if the reservoir model or the optimization problem is complex. Hence, proxy modeling can yield a faster solution than numerical reservoir simulation. This fast solution provides insights to better formulate field development plans. Due to technological advancements, machine learning increasingly contributes to the designing and building of proxy models. Thus, in this work, we have proposed the application of the two-stage proxy modeling, namely global and local components, to generate useful insights. We have established global proxy models and coupled them with optimization algorithms to produce a new database. In this paper, the machine learning technique used is a multilayer perceptron. The optimization algorithms comprise the Genetic Algorithm and the Particle Swarm Optimization. We then implemented the newly generated database to build local proxy models to yield solutions that are close to the “ground truth”. The results obtained demonstrate that conducting global and local proxy modeling can produce results with acceptable accuracy. For the optimized rate profiles, the  $R^2$  metric overall exceeds 0.96. The range of Absolute Percentage Error of the local proxy models generally reduces to 0–3% as compared to the global proxy models which has a 0–5% error range. We achieved a reduction in computational time by six times as compared with optimization by only using a numerical reservoir simulator.

**Keywords:** global and local proxy modeling; machine learning; derivative-free optimization; reservoir simulation



check for updates

Citation: Ng, C.S.W.; Jahanbani Ghahfarokhi, A.; Wiranda, W. Fast Well Control Optimization with Two-Stage Proxy Modeling. *Energies* **2023**, *16*, 3269. <https://doi.org/10.3390/en16073269>

Academic Editor: Dameng Liu

Received: 23 February 2023

Revised: 29 March 2023

Accepted: 3 April 2023

Published: 6 April 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Numerical reservoir simulation (NRS) is one of the most essential aspects of reservoir engineering. NRS is highly relied upon for the modeling of fluid flow in porous media. This implies that a reservoir is better when sufficient data are acquired to develop a reservoir model through NRS. Using NRS, fluids can be more efficiently extracted from the underground to meet the global energy demand. However, NRS suffers from computational issues, despite today’s advanced computing power. This limitation is still not entirely addressed, especially when many details are included in building the NRS model. Concerning this, numerous measures are proposed, including proxy modeling.

Proxy modeling pertains to the modeling of a substitute for a base paradigm, namely NRS. Such an approach can provide a fast solution when the decision-making is urgent. There are different examples of proxy modeling available for employment. In this case, the machine learning (ML) technique is one of them. In general, ML can be perceived as a computer algorithm that is built to deduce a pattern or relationship between the input variables and the output provided [1]. Some prevalent examples of ML include artificial neural networks, support vector machines, and gradient boosting machines. These methods have been demonstrated to be successful in establishing proxy models. Regarding this, some literature presented the use of an ensemble of neuro-fuzzy networks as ML-based

proxy models in several aspects of reservoir engineering, including carbon, capture, and storage [2] and shale analytics [3].

Apart from these, a variant of the gradient boosting machine, e.g., extreme gradient boosting machine (XGBoost), was implemented for fast analysis of well placements in a heterogeneous reservoir [4]. The articles [5,6] also discussed the use of some more advanced ML methods in simulating the behavior of reservoirs and production trends, which is an important criterion to be manifested by a proxy model. The potential implementation of ML methods in proxy modeling was also further highlighted in the domain of secondary recovery. Waterflooding is one of the most prevalent secondary recovery techniques. Aside from its economical employment [7], it has been well-received in the oil and gas industry due to its ability to maintain the reservoir pressure, prevent subsidence, and simultaneously increase the oil recovery from oil fields. Regarding the technicality of waterflooding, “voidage replacement” has been a common parameter to guide water injection, where the total volume of production is equal to the total volume of injection. The challenge of using a voidage replacement ratio (ratio of the injected to the produced fluid volumes) with a fixed injector location is the allocation of the water injection for each well.

Changing the injection operations can optimize the waterflooding performance. These operations include the well control adjustment in which the net present value (NPV) is set to be the objective function. Conventionally, NRS is used to obtain the result for each water injection scenario. For a full-field scenario, using NRS will be time-consuming to maximize the objective function, especially if the geology of the reservoir is sophisticated or the dimension of optimization variables is high. Therefore, ML-based proxy models are suggested to mitigate the computational challenges. Several previous works [8,9] have established a methodology in this context. Nonetheless, the efficiency of the methodology in resolving the optimization problem with higher dimensionality still requires improvement. One of the potential solutions lies in the establishment of two different classes of proxy models, namely global and local proxy models, as discussed in [10,11]. Fundamentally, local proxy models aim at refining the quality of proxy models in which solutions closer to the “true” optimal can be determined.

Furthermore, to conduct a successful waterflooding optimization, an optimization algorithm is another essential tool. There are two main types of algorithms, e.g., gradient-based and gradient-free. In recent studies of optimization algorithms, gradient-free algorithms have gained increasing attention due to their ability to converge to the global optimal [12]. The nature-inspired algorithm is the epitome of gradient-free algorithms. Its successful integration with the ML-based proxy models has been displayed in several pieces of literature in reservoir and production engineering [13–15]. In this study, two optimization algorithms are used: the Genetic Algorithm (GA) and the Particle Swarm Optimization (PSO). These algorithms are only applied to determine the optimal sets of well control under waterflooding. These algorithms also illustrated good potential to be used as training algorithms in data-driven modeling [16,17].

In this paper, we aim to illustrate how ML and nature-inspired algorithms can be coupled with the two-stage proxy modeling to optimize waterflooding. A benchmark model (UNISIM-I-D) was used to demonstrate that global and local proxy modeling could be used to replicate the behavior of a real reservoir. The UNISIM-I-D model was created based on Namorado Field, located in Campos Basin in Brazil. The proxy models are developed using the multi-layer perceptron (MLP). These proxy models were initiated to replicate the NRS and coupled with the above-mentioned algorithms for well control optimization. The proxy models were built using selected geological properties, time, and output from the NRS. Using the Latin Hypercube Sampling (LHS) method, which was proposed by McKay et al. [18], multiple injection scenarios were created and divided into the training set and the blind validation set. NRS was performed on the injection scenarios to obtain the simulation results. After a successful training and the validation test of the proxy models, the simulation results could be generated without using NRS. Using the results from the global proxy model, the local proxy model was trained based on the retrieved

samples of optimization results. With this method, the optimization result was obtained by using the local proxy model without the requirement to run the repetitive process of optimization. Using the global and local proxy models, the optimized water injection control for the UNISIM-I-D model was determined with higher computational efficiency.

Following this introduction, Section 2 of this paper discusses the details of the UNISIM-I-D model. Sections 3 and 4 respectively explain the algorithms and the ML method applied. Thereafter, Section 5 expounds the integration of the concepts presented to scaffold the establishment of the methodology presented. Section 6 comprises a discussion on the results obtained from this work. The concluding remarks can be found in Section 7.

## 2. Reservoir Description

The UNISIM-I-D model was created on the Namorado Field, located in the Campos Basin in Brazil with known properties [19]. With the benchmark model, it is possible to ensure the applicability of developed reservoir management methodologies to real reservoirs. In this study, we used the upscaled model to decrease the computational effort for multiple scenarios. The grid cell resolution of the upscaled model is  $100 \times 100 \times 8$  m, discretized into a corner point grid  $81 \times 58 \times 20$  cells, with a total of 36,739 active total cells.

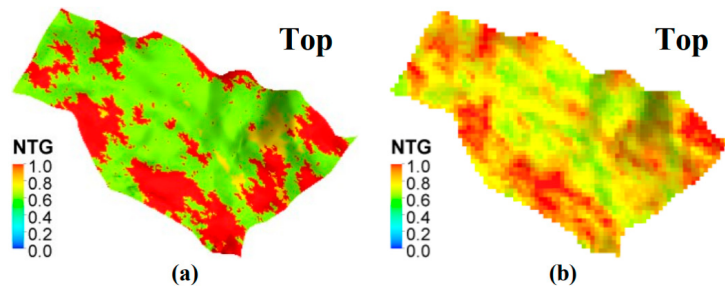
### 2.1. Static Properties Description

The UNISIM-I-D model facies distribution is reflected based on the net-to-gross distribution. The original fine model has the following rules to set the net-to-gross (NTG) based on the facies shown in Table 1. The facies modeling is defined using the Sequential Indicator Simulation with a vertical trend [20].

**Table 1.** Facies and Net to Gross rules.

Facies	Net to Gross
0	1.0
1	0.8
2	0.6
3	0.0

Class 0 is defined as reservoir facies with good properties whereas classes 1 and 2 are the medium reservoir properties. Class 3 is defined as non-reservoir. The reservoir active grid is upscaled and results in a continuous distribution of the NTG (Figure 1).



**Figure 1.** UNISIM-I-D NTG distribution: (a) Fine grid and (b) Upscaled model [19].

Figure 1 shows that after upscaling, the NTG became continuous due to the nature of the arithmetic volume-weighted method. The method is used to maintain the hydrocarbon volume constant during flow simulation.

The effective porosity model is derived from the density log and shaliness of the properties. After the effective porosity is modeled from log data, it is distributed to the whole model using the Sequential Gaussian Simulation (SGS) [21]. After the porosity is modeled on the fine grid, it is upscaled using the same method as NTG upscaling. The results of upscaled porosity are shown in Figure 2.

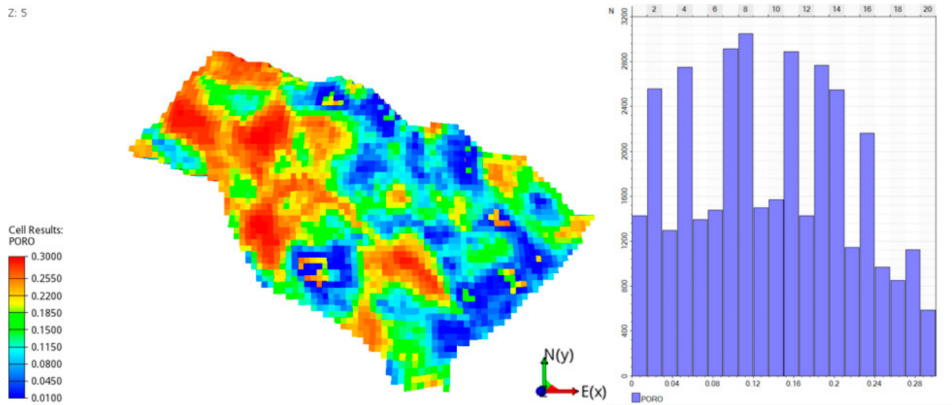


Figure 2. UNISIM-I-D upscaled porosity results.

The permeability model was initially derived from the core analysis data, and a relationship between porosity and permeability was established (Figure 3). This horizontal permeability is distributed to the model using the correlation, while the vertical permeability is defined by using a multiplier (which ranges from 0 to 1.5) times the horizontal permeability. The permeability was upscaled by using the flow-based upscaling technique, FLOWSIM [22]. The results of the porosity and the horizontal permeability relationship of the upscaled case is depicted in Figure 3b.

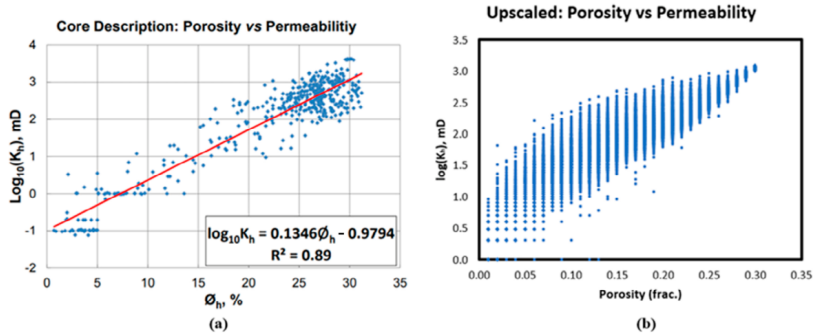


Figure 3. Porosity versus permeability: (a) Core analysis data [19] in which the blue dot refers to the core sample whereas the red line refers to the equation and (b) Upscaled model.

Due to the upscaling method, the horizontal permeability has slightly different values in I and J directions. Meanwhile, the relationship between the vertical and the horizontal permeability is scattered due to the different grid resolution in vertical direction. Figure 4 shows the relationship between the horizontal and the vertical permeabilities.

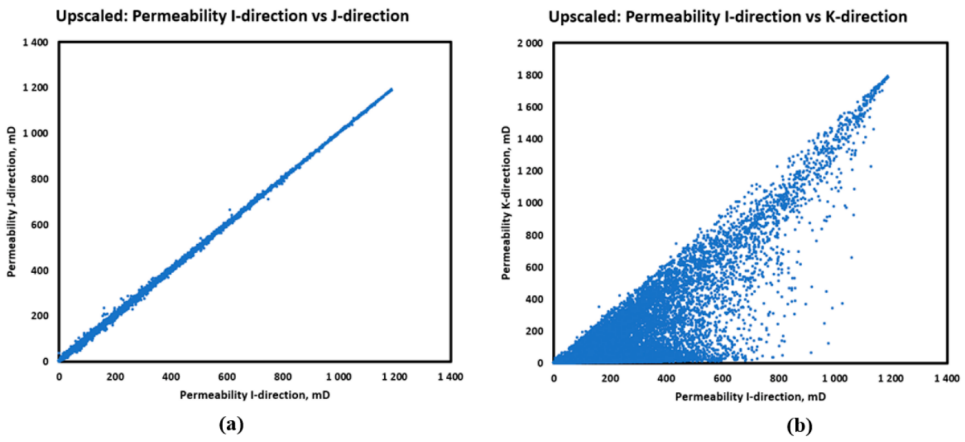


Figure 4. Upscaled permeability relationship in: (a) I-J direction; (b) I-K direction.

The final static properties used in this study are shown in Figure 5.

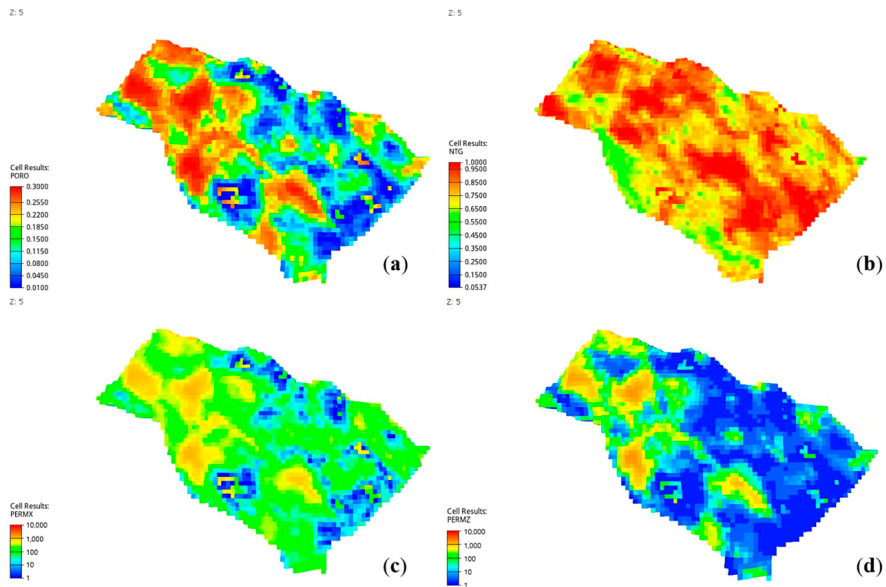


Figure 5. Static properties used for simulation (a) Porosity, (b) Net-to-Gross, (c) Permeability I-direction, and (d) Permeability K-direction.

2.2. Dynamic Properties Description

In this section, the fluid properties and fluid-rock interaction properties used in the simulation are defined. The fluid model used in the simulation is the Black Oil model with



the initialization of the oil phase, the dissolved gas and the water phase. Figure 6 shows the oil properties and Figure 7 demonstrates the gas properties used in the model.

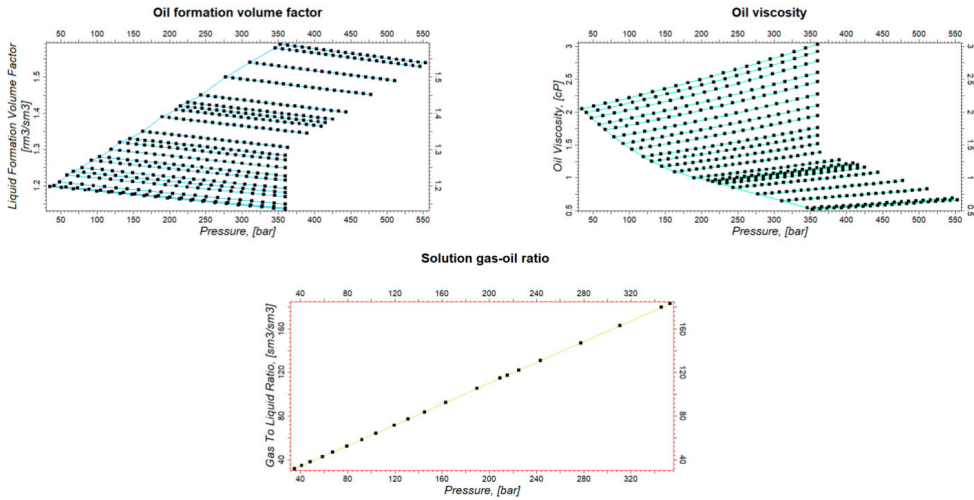


Figure 6. Oil properties used for simulation.

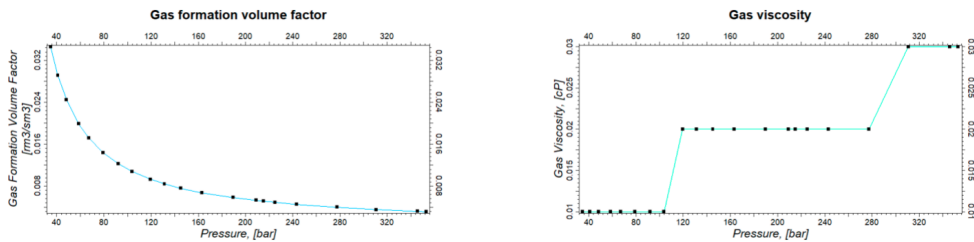


Figure 7. Gas properties used for simulation.

The water properties are defined in Table 2.

Table 2. Water phase properties.

Properties	Value
Reference pressure	0.98067 bara
Water formation volume factor at reference pressure	1.021 $\text{rm}^3/\text{sm}^3$
Water compressibility	$4.8579 \times 10^{-5} \text{ bar}^{-1}$
Water viscosity	0.3 cP
Water viscosibility	0 $\text{bar}^{-1}$

The fluid-rock interaction is defined by the relative permeability and the capillary pressure curves in the simulation. The relative permeabilities used in the simulation are presented in Figure 8 and capillary pressure curves are illustrated in Figure 9.

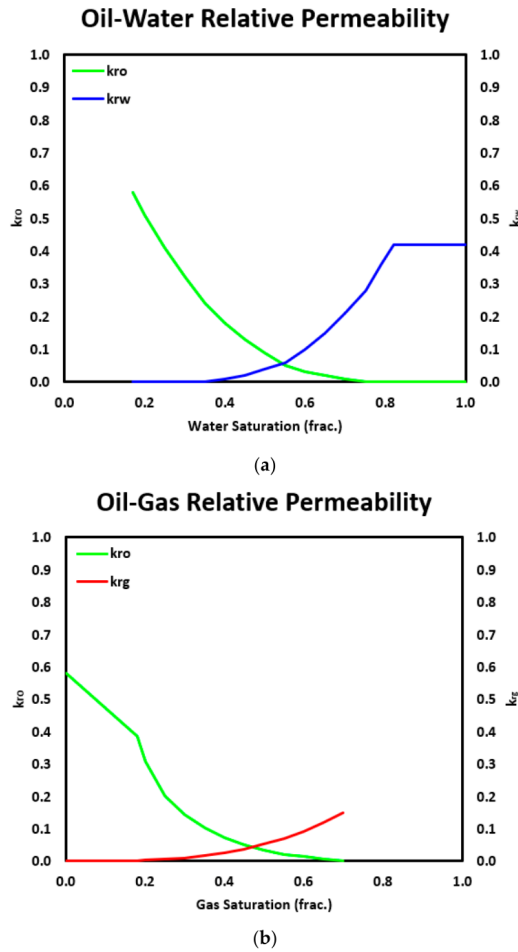


Figure 8. Relative permeability (a) Oil-Water and (b) Oil-Gas.

Another dynamic rock property is rock compaction. The rock compaction used is the standard model, based on the equations of Newman 1973 [23], Hall [24], and Van Der Knaap [25], to generate rock compaction tables based on the known rock compressibility at a reference pressure, as shown in Table 3.

Table 3. Rock compressibility at reference pressure.

Properties	Value
Rock compressibility	$5.4 \times 10^{-5} \text{ bar}^{-1}$
Reference pressure	315.77 bara

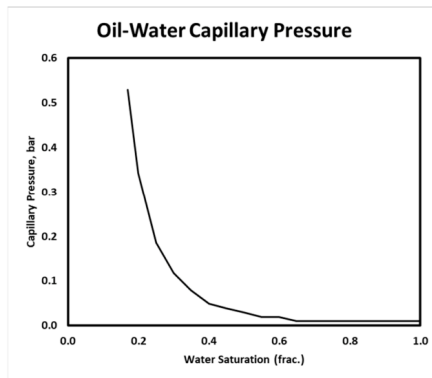


Figure 9. Oil-water capillary pressure.

2.3. Initialization

The initialization of the model is conducted by defining two regions with different water-oil contacts. The region boundary is defined by the normal fault shown in Figure 10a. The horst (blue area) has a higher water-oil contact at the depth  $-3100$  m than the graben (magenta area) with water-oil contact at the depth  $-3174$  m, as shown in Figure 10. The initial pressure is defined based on the reference point at depth of  $3000$  m where the pressure is  $320.68$  bara. Figure 10b shows the distribution of the initial fluid saturation with the different water-oil contacts for both regions.

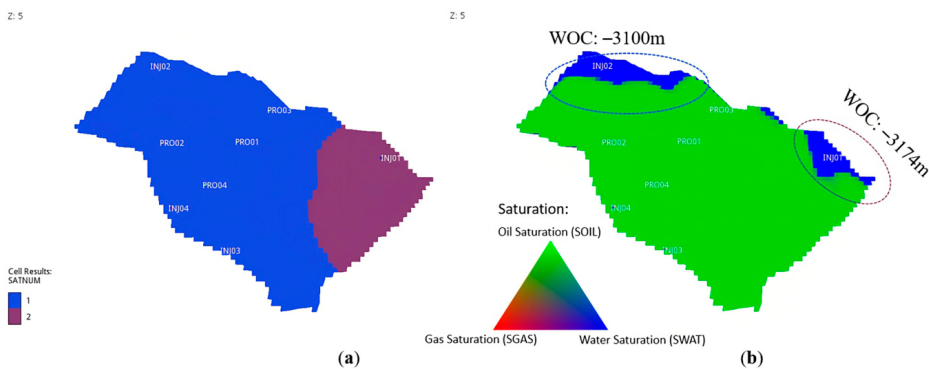


Figure 10. Water-oil contact (WOC) definition: (a) Region definition (b) Distribution of initial fluid saturation.

With all the parameters from the static, dynamic, and initialization of the reservoir model, the initial in-place volume is presented in Table 4. It is confirmed, with the in-place volume mentioned in the original UNISIM-I-D benchmark model [19], that the model used in this study is unmodified.

**Table 4.** Initial in-place volumes.

Properties	Value
Initial Oil In Place	130 MM Sm <sup>3</sup>
Initial Dissolved Gas In Place	14.7 B Sm <sup>3</sup>

### 3. Algorithms

Regarding the selection of mathematical algorithms, nature-inspired algorithms, specifically the Genetic Algorithm (GA) and the Particle Swarm Optimization (PSO), were preferred due to their structural simplicity and successful implementation in several articles [8,9,14]. These algorithms are derivative-free, implying that computation or approximation of gradient is unnecessary. They also present a good capability of eluding premature convergence. This is because they accomplish a good balance between exploration and exploitation in optimization. Exploration aims at diversifying the solution over the search space. Exploitation targets to leverage the search for solution over the local region (a more refined search space).

GA, proposed by Holland [26], is one of the population-based metaheuristic algorithms. Its formulation relies heavily upon the Darwinism Theory of Evolution. GA, in general, implements different types of genetic operators when it comes to the exploration and exploitation of the solution (search) space. Fundamentally, an individual solution is encoded as a string, that is known as a chromosome. Therefore, an initial population of chromosomes will be generated as potential solutions. The quality of each chromosome is evaluated by employing an objective function (also known as fitness). These chromosomes will undergo the genetic operators, for instance, selection, crossover, elitism, and mutation over several iterations. During the selection process, several chromosomes are chosen as parents to yield new offspring. Then, elitism ensures the survival of the best chromosome (highest fitness) which can be inherited in the next generation. Crossover involves the exchange of certain parts (also termed “genes”) of chromosomes to produce new ones. Mutation modifies certain genes of chromosomes to elude convergence to the local optima [27]. Mathematically, the chromosome population will be subject to these genetic operators for some iterations until the stopping criterion is met. The final chromosome with the highest fitness is treated as the final solution.

PSO is another example of the population-based algorithms that was implemented in this work. PSO was formulated by Kennedy and Eberhart [28], according to the simulation of a moving stock of birds or a school of fish. In this aspect, an individual solution is perceived as the particle, in which the initial population of particles (a swarm of particles) is randomly generated as potential solutions. The quality of each particle is assessed using an objective function. As PSO commences, the position and velocity of each particle are randomly initialized. Throughout the iterations, a particle recognizes the previous optimal value of the objective function. The respective position vector is the local best position (pbest). The global best position (gbest) is the best position of particles achieved hitherto in the swarm. At every iteration, the motion of particles is dictated by three parameters, namely cognitive factors, social factors, and inertia weight. Generally, the cognitive factor enables the attraction of particles towards the pbest. The social factor aids in attraction towards the gbest. Inertia weight could be initialized to improve convergence. The pbest and gbest are determined iteratively to update the velocity at the current step. As the velocity at the next iteration is evaluated, the update on the position of a particle at the next iteration is performed. Over some iterations, each particle updates its position via the minimization of the objective function until the stopping criterion is reached.

### 4. Machine Learning

Machine learning (ML) is defined as a computer algorithm that can derive inferences in the pattern of data provided. There are numerous examples of ML techniques, including support vector machines, random forests, and artificial neural networks (ANN). ANN is

one of the most popular methods of ML that has been applied extensively. Its mechanism primarily resembles the neural system in human brains. Mathematically, it comprises different fundamental components, including layers, activation functions, and nodes. The layers are input, hidden, and output. Each layer consists of several nodes that are represented by weights and biases. Starting from the input layer, weights and biases are consecutively interconnected layer to layer. Thereafter, the respective product will be fed into a preselected activation function to yield a new value that will propagate to the next layer. This process of propagation continues until it reaches the output layer. For the relevant details, refer to the literature [29]. Application of ANN generally gravitates to the development of data-driven models which are used for prediction and/or optimization. There are also different variants of ANN, such as multilayer perceptron (MLP), recurrent neural network (RNN), and convolutional neural network (CNN). In this work, only MLP is considered due to its successful use in resolving engineering problems.

### 5. Proxy Modeling and Optimization Problem

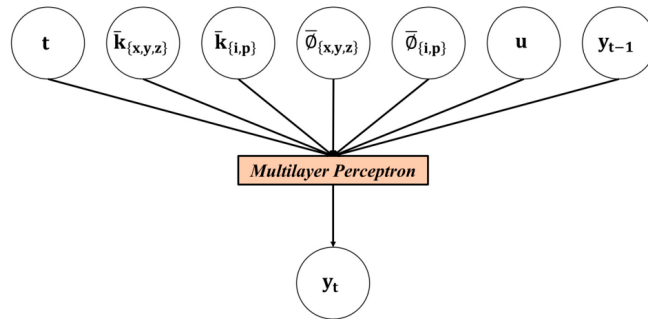
To establish proxy models, we need to be cognizant of the functions of the proxy models before proceeding into the development phase. In our study, we formulate a waterflooding optimization problem, in which the pertinent objective function is set to be the net present value (NPV). This NPV function is mathematically expressed in Equation (1). The control vector is represented by  $\mathbf{u}$  and the field rates are indicated by  $Q$ , in which the subscripts refer to the types of fluids.  $P$  refers to the price or cost of fluid produced/injected.  $n_{total}$  is the total number of timesteps whereas  $t_i$  refers to the cumulative time until timestep  $i$ .  $\Delta t_i$  refers to the timestep difference between the time  $i$  and the previous timestep. Such an optimization problem resonates with some of our previous works [8,9]. However, one of the distinctive differences pertains to the number of optimization variables (decision variables) included. In the case of this optimization, NPV is maximized every 365 days by optimally adjusting each injection rate (within the range of 0 Sm<sup>3</sup>/day and 2500 Sm<sup>3</sup>/day) and bottomhole pressure (BHP) of each producer (within 175 bar and 200 bar). The total production period lasts for 9125 days.

$$NPV(\mathbf{u}) = \sum_{i=1}^{n_{total}} \frac{\Delta t_i \times (Q_{i,oil}(\mathbf{u})P_{oil} - Q_{i,wat\ prod}(\mathbf{u})P_{wat\ prod} - Q_{i,wat\ inj}(\mathbf{u})P_{wat\ inj} + Q_{i,gas}(\mathbf{u})P_{gas})}{(1 + \text{interest rate})^{t_i/365}} \tag{1}$$

Since the UNISIM-I-D reservoir model comprises four injectors and four producers, this results in 200 variables (8 variables/timestep × 25 timesteps) to be optimized to achieve a higher NPV. Based on the NPV function, we assume that the produced gas will be sold. Regarding the economic parameters, the oil price is 503.2 USD/m<sup>3</sup>, the cost of handling produced water and injecting water are 62.9 USD/m<sup>3</sup> and 50.32 USD/m<sup>3</sup>, respectively, and the gas price is 0.265 USD/m<sup>3</sup>. The interest rate is 0.10 per year. From the NPV function, we need to develop models that can predict the values of the Field Oil Production Rate (FOPR), Field Water Production Rate (FWPR), Field Water Injection Rate (FWIR), and Field Gas Production Rate (FGPR) at each timestep. Keeping in mind our investigation and previous studies [8,17,30], we decided to build three different proxy models, which can forecast Field Liquid Production Rate (FLPR), Field Water Cut (FWCT), and FWIR. FLPR and FWIR are in the units of Sm<sup>3</sup>/day whereas FWCT is expressed in a fraction. These proxy models provide the necessary values to compute the NPV. It is essential to know that FGPR (Sm<sup>3</sup>/day) can be obtained by multiplying FOPR by the constant gas-oil-ratio  $R_s$ , which is 113.45 Sm<sup>3</sup>/Sm<sup>3</sup>.

Proxy modeling can be perceived as establishing a relationship between the input and the output variables. Our previous studies and some literature suggest that integrating static and dynamic properties can increase the reliability of the proxy models. Therefore, we have formulated the mathematical function of the proxy models, as shown in Figure 11. In Figure 11,  $\bar{k}_{\{x,y,z\}}$  represents the arithmetic mean of grid block permeability for each layer in  $x$ -,  $y$ -, and  $z$ -directions.  $\bar{\phi}_{\{x,y,z\}}$  refers to the arithmetic mean of grid block porosity for every

layer.  $\bar{k}_{\{i,p\}}$  and  $\bar{\phi}_{\{i,p\}}$  respectively correspond to the arithmetic mean of permeability and porosity of the perforated grid blocks for each injector and producer. Parameters  $u$  and  $\Delta t$  respectively refer to control variables and cumulative time (days) until the current timestep.  $y_{t-1}$  and  $y_{t-1}$  correspond to output at previous and current timestep. As discussed, there are 20 layers and 8 wells in the UNISIM-I-D model, and this yields 112 static inputs. Considering the dynamic inputs, such as the number of days, 8 control variables, and output at the previous timestep, there are 122 input variables.



**Figure 11.** Relationship between input parameters and output for the proxy models.

Understanding the objective of the optimization problem and the formulation of proxy models provides a clear direction to proceed into the workflow, as shown in Figure 12. This workflow involves the design of two types of proxy models which we correspondingly term as the Global Proxy Models and the Local Proxy Models. As displayed in the workflow, Latin Hypercube Sampling (LHS) is initiated to create 310 control scenarios. These 300 scenarios are fed into NRS to generate a training database for Global Proxy Modeling. The other 10 scenarios are applied to create the database for blind validation. The maximum NPV resulting from these scenarios is 5456.70 million USD. Before proceeding to the training process, the database is normalized to be between 0 and 1 based on the maximum and minimum data, as discussed in [9]. After developing the global proxy models, they are coupled with GA or PSO to generate the database for local proxy modeling. The topologies of global and local proxy models are decided via a trial-and-error approach, which is portrayed in Table 5. The terms “Local Proxy-GA” and “Local Proxy-PSO” in the table imply that the local proxy models are built from the database generated using the global proxy models coupled with GA and PSO for optimization, respectively. In Table 5, the number of hidden nodes applies to each hidden layer. Moreover, the activation function in the output layer for each proxy model is linear. The training uses the algorithm Adam, also known as Adaptive Moment Estimation [31], iterations of 2000, a learning rate of 0.001, and a tolerance of  $10^{-6}$ . The early stopping feature is activated. The validation fraction is set to 1/9. The remaining parameters are the default values, as suggested in Scikit-Learn [32].

The inertia weight is 0.80 whereas the cognitive and social learning factors are both parameterized as 1.05.  $r_1$  and  $r_2$  are sampled from a uniform distribution between 0 and 1. For the GA, the crossover probability is 0.8, the mutation probability is 0.8, the elite ratio is 1/30, the parents’ portion is 0.6, and the type of crossover is two-point. The abovementioned parameters for both GA and PSO were initialized via a trial-and-error approach. For both algorithms, the number of optimization iteration is 200 and the population size is 30.

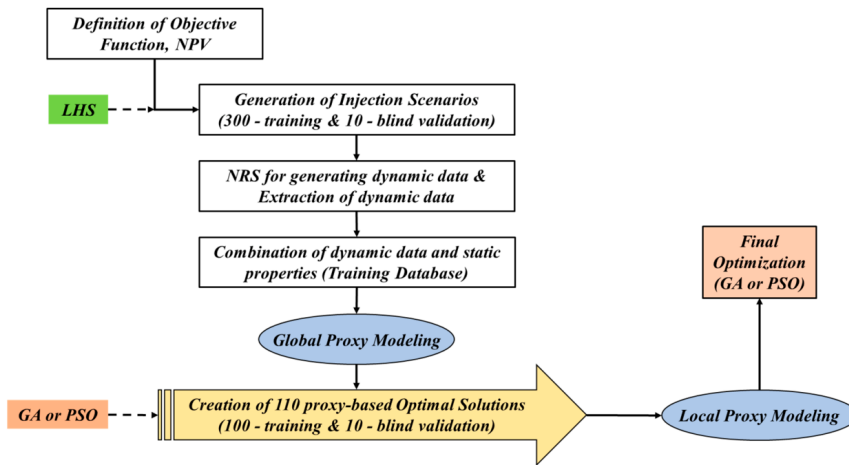


Figure 12. Workflow of the proposed methodology.

Table 5. Topology of the MLP.

Type of Proxy Models	Number of Hidden Layers	Number of Hidden Nodes	Activation Functions (Hidden Layers)
FLPR			
Global Proxy Model	3	250	ReLU
Local Proxy-GA	3	250	ReLU
Local Proxy-PSO	3	200	ReLU
Type of Proxy Models	Number of Hidden Layers	Number of Hidden Nodes	Activation Functions (Hidden Layers)
FWCT			
Global Proxy Model	3	150	ReLU
Local Proxy-GA	3	150	ReLU
Local Proxy-PSO	3	150	ReLU
Type of Proxy Models	Number of Hidden Layers	Number of Hidden Nodes	Activation Functions (Hidden Layers)
FWIR			
Global Proxy Model	3	200	ReLU
Local Proxy-GA	3	200	ReLU
Local Proxy-PSO	3	200	ReLU

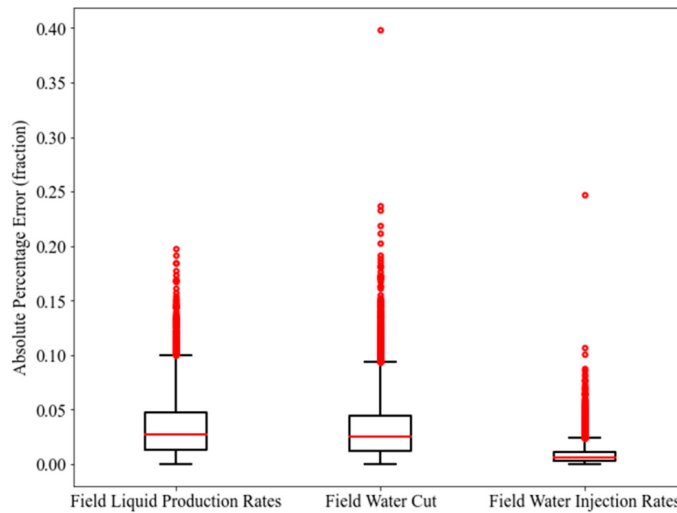
As the training and blind validation results of global proxy models illustrate good results, these models are coupled with metaheuristic algorithms to conduct the waterflooding optimization. The optimization is run 110 times (indicating 110 optimal scenarios in which 100 scenarios are for training and the other 10 are for blind validation) and the resulting optimal solutions (control variables) are sent back to the simulator to create a training database for local proxy modeling. For this, the calculated NPV is ensured to exceed the abovementioned maximum NPV. When the local proxy models illustrate good results of training and blind validation, these models are implemented for the final optimization. The final optimization is performed 200 times for further analysis. The relevant findings are summarized and discussed in the following section.

### 6. Results and Discussion

The MLP was chosen as the ML technique to develop the proxy models in this work. The proxy modeling was performed using the Scikit-Learn with the aid of Python programming language [33]. As explained in the workflow, there are two stages of proxy modeling. To assess the reliability of these proxy models, we implemented three statistical metrics, namely Coefficient of Determination ( $R^2$ ), Root Mean Squared Error (RMSE), and Average Absolute Percentage Error (AAPE). Different examples of statistical metrics in tandem with their formulations can be referred to in [34]. The training and testing results of the first stage of proxy modeling (global proxy modeling) are presented in Table 6. In addition, the boxplots of the Absolute Percentage Error (APE) for the training and testing data points are demonstrated in Figures 13 and 14.

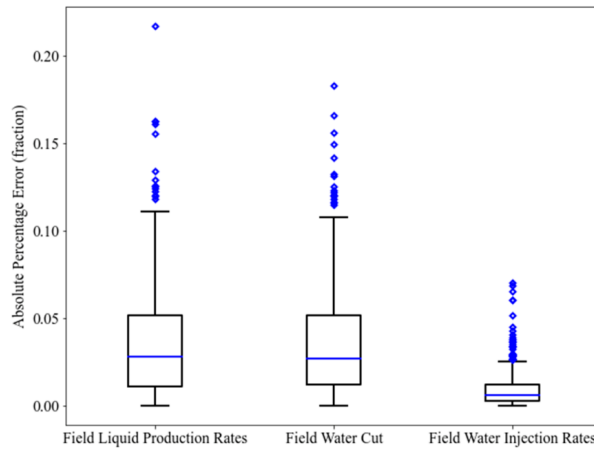
**Table 6.** The training and testing results of global proxy modeling.

Models (Training)	$R^2$	RMSE	AAPE
FLPR	0.9510	150.06	3.357
FWCT	0.9933	0.0074	3.196
FWIR	0.9982	54.93	0.842
Models (Training)	$R^2$	RMSE	AAPE
FLPR	0.9516	153.01	3.440
FWCT	0.9920	0.0081	3.435
FWIR	0.9980	59.37	0.874



**Figure 13.** Boxplot of the Absolute Percentage Error of the training data points (global proxy modeling).





**Figure 14.** Boxplot of the Absolute Percentage Error of the testing data points (global proxy modeling).

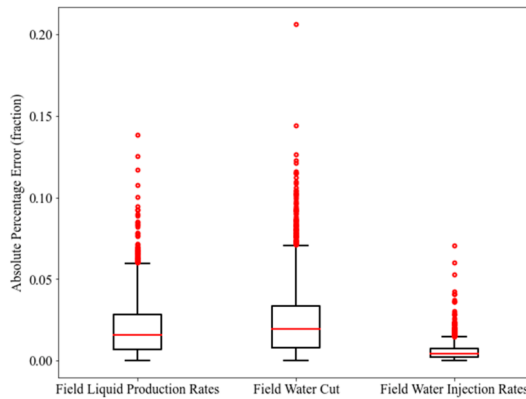
From the boxplots, it can be seen that MLP-FWIR displays the smallest range of APE as compared to MLP-FLPR and MLP-FWCT, in terms of training and testing. Furthermore, the statistics on  $R^2$  and AAPE provided in Table 6 also confirm the better performance of MLP-FWIR for training and testing. This better performance does not undermine the predictability of MLP-FLPR and MLP-FWCT. Numerous outliers are noticed in the boxplots for all the three models. Hence, the predictability of these models needs to be further justified by applying blind validation cases. To conduct this justification, ten blind validation cases were generated, as explained in Figure 12. The performance metrics of the proxy models for these blind validation cases are displayed in Table 7. The results consist of the mean of all the ten blind validation cases. It is observed that MLP-FWIR still outperforms the other two models. In MLP-FLPR, the mean  $R^2$ , the mean RMSE, and the mean AAPE might be less satisfactory. From Tables 6 and 7, it is worth noting that MLP-FLPR generally illustrates relatively poor performance. This could be due to the complexity of the reservoir model used. This implies that the database provided might not adequately reflect the physics of the reservoir. In MLP-FWCT too, a similar issue can be observed in terms of the AAPE. Despite this fact, these models are still considered practical to generate insightful optimal solutions for local proxy modeling.

**Table 7.** The blind validation results of global proxy modeling.

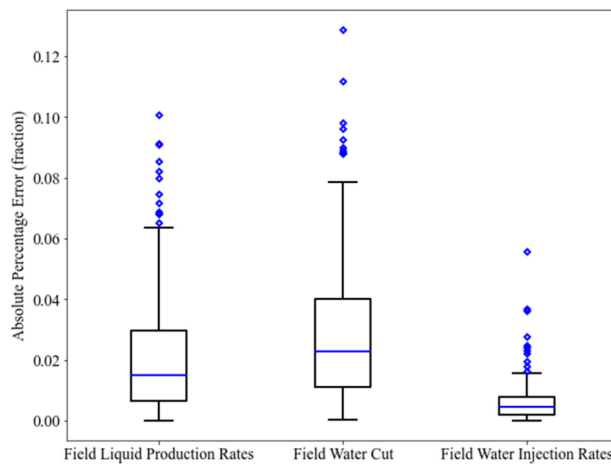
Models (Blind Validation)	Mean $R^2$	Mean RMSE	Mean AAPE
FLPR	0.9267	183.18	4.274
FWCT	0.9892	0.0092	4.075
FWIR	0.9974	64.03	1.169

Upon completion of the first stage of proxy modeling, the proxy models are readily employed for optimization with the GA and the PSO. However, optimization at this phase aims at creating a “useful” database for the training of local proxy models. This database consists of the data that have a closer proximity to the “true” optimal solution. When the new “training” database is ready, it can be applied to establish the local proxy models. In this case, two different algorithms result in two different databases. It is anticipated that the performance metrics of the local proxy models demonstrated more improvement

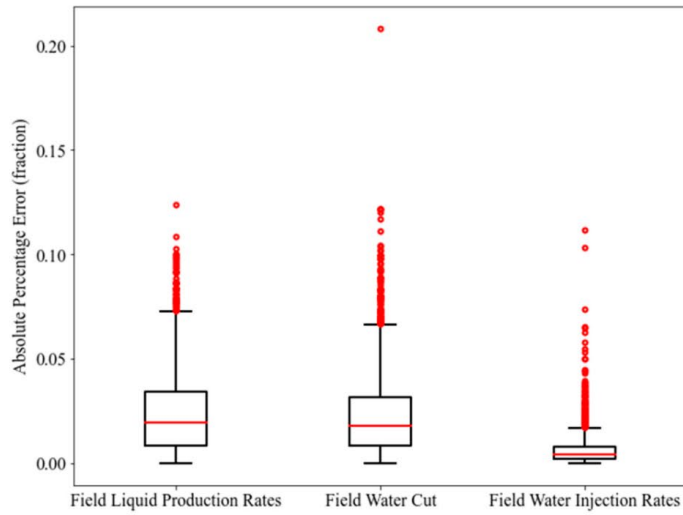
as compared with the global proxy models. For illustrative purposes, the corresponding boxplots of the APE in the training and testing phases are portrayed in Figures 15 and 16 for GA as well as in Figures 17 and 18 for PSO, respectively. For a more comprehensive evaluation, the training and testing results of the second stage of proxy modeling (local proxy modeling) are demonstrated in Table 8 for GA and Table 9 for PSO. The statistics in Tables 8 and 9, highlight an improvement in terms of  $R^2$ , RMSE, and AAPE as compared with the results from Table 6. This fulfills the goal of conducting the second stage of proxy modeling. In terms of blind validation, ten additional cases were created. The statistics displayed in Tables 10 and 11 for blind validation, also show a good level of enhancement in the mean  $R^2$ , the mean RMSE, and the mean AAPE as compared with those shown in Table 7.



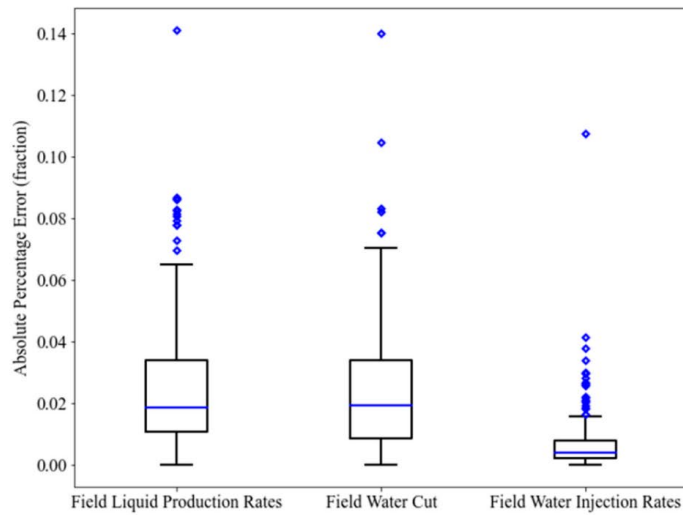
**Figure 15.** Boxplot of the Absolute Percentage Error of the training data points (local proxy modeling-GA).



**Figure 16.** Boxplot of the Absolute Percentage Error of the testing data points (local proxy modeling-GA).



**Figure 17.** Boxplot of the Absolute Percentage Error of the training data points (local proxy modeling-PSO).



**Figure 18.** Boxplot of the Absolute Percentage Error of the testing data points (local proxy modeling-PSO).

**Table 8.** The training and testing results of local proxy modeling (GA).

Models (Training)	R <sup>2</sup>	RMSE	AAPE
FLPR	0.9660	108.69	1.959
FWCT	0.9961	0.0086	2.403
FWIR	0.9975	43.08	0.534
Models (Testing)	R <sup>2</sup>	RMSE	AAPE
FLPR	0.9659	119.79	2.123
FWCT	0.9956	0.0094	2.774
FWIR	0.9978	46.51	0.588

**Table 9.** The training and testing results of local proxy modeling (PSO).

Models (Training)	R <sup>2</sup>	RMSE	AAPE
FLPR	0.9632	124.66	2.383
FWCT	0.9962	0.0076	2.276
FWIR	0.9974	52.09	0.620
Models (Testing)	R <sup>2</sup>	RMSE	AAPE
FLPR	0.9630	128.53	2.442
FWCT	0.9953	0.0086	2.396
FWIR	0.9962	66.73	0.666

**Table 10.** The blind validation results of local proxy modeling (PSO).

Models (Blind Validation)	Mean R <sup>2</sup>	Mean RMSE	Mean AAPE
FLPR	0.9578	118.80	2.262
FWCT	0.9935	0.0105	3.037
FWIR	0.9975	42.46	0.581

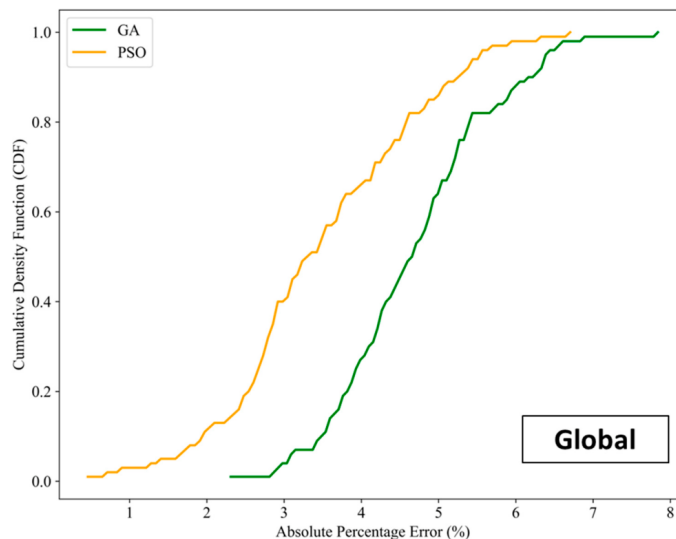
**Table 11.** The blind validation results of local proxy modeling (GA).

Models (Blind Validation)	Mean R <sup>2</sup>	Mean RMSE	Mean AAPE
FLPR	0.9418	152.38	3.012
FWCT	0.9905	0.0112	3.155
FWIR	0.9971	51.76	0.681

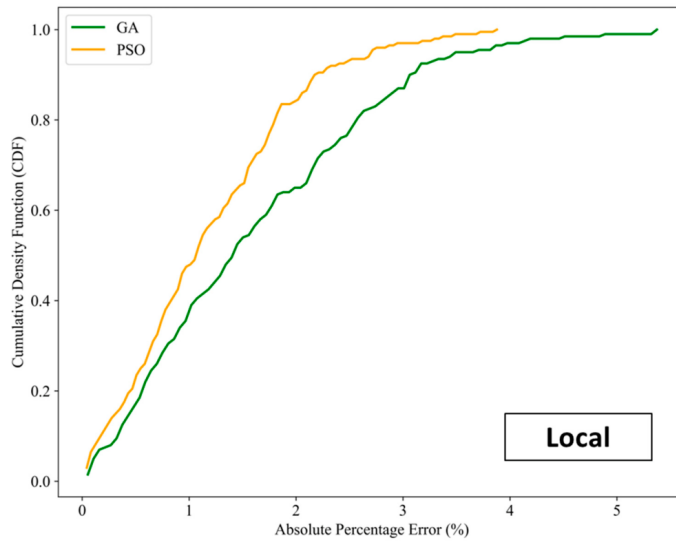
One of the main goals of this work, which was achieving significant computational efficiency in tandem with good accuracy of results, was attained. For both the GA and the PSO algorithms, the framework (considering global and local proxy modeling as well as optimization) took about two days to complete. However, when the optimization was conducted with the reservoir simulator, both algorithms required about twelve days to finish. This demonstrates that the proposed framework can reduce the computational time by six times. It is essential to note that the framework runs the optimization 100 times in the case of global proxy modeling and 200 times for local proxy modeling. Nevertheless, the optimization with the reservoir simulator was only performed once. For this, the optimized NPVs obtained using the simulator coupled with GA and PSO are 6054.61 million USD and 5832.55 million USD, respectively.

To further highlight the improvement of accuracy attained by conducting the two-stage proxy modeling, the cumulative density frequency (CDF) of absolute percentage error between the actual NPV and the NPV predicted by both global and local proxy models are plotted in Figures 19 and 20, respectively. Due to the expensive computational demand of the reservoir simulator for the optimization task as explained above, the actual NPVs are calculated by feeding the optimized control variables obtained using the corresponding proxy model into the reservoir simulator. As the CDF plots display, the range of the APE yielded by local proxy models reduces as compared with that of global proxy models. Most of the resulting samples lie within the APE range of 0%–3% for both types of local proxy models. This verifies that local proxy modeling permits higher accuracy of optimal results. Additionally, proxy models coupled with PSO exhibit a higher chance of achieving results within a more desired level of accuracy (compared with GA). In terms of NPV calculation, the GA produces bigger values than the PSO. This is confirmed by the CDF plots of the NPVs in Figure 21, which show the actual NPVs obtained from the local proxy models.

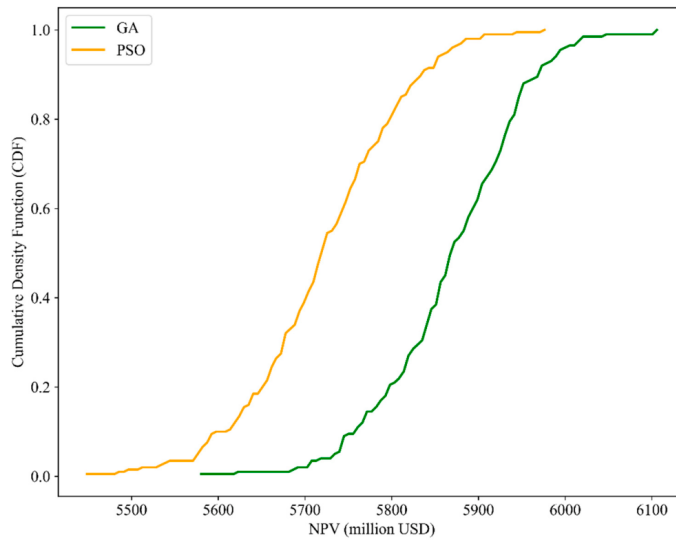
The details highlighted in Figure 21 were obtained when the optimization was run 200 times. For each optimization run, there are 200 iterations. Thereafter, as explained previously, for each run, the resultant optimal control variables are fed into the reservoir simulator. This denotes that there will be 200 optimal NPV samples. With this, the highest NPV achieved (out of the 200 optimal solutions) is 6105.79 million USD for the GA. Using the respective control only in tandem with proxy models, the resulting NPV is 6131.79 million USD. In the case of the PSO, by feeding the 200 proxy-optimized solutions into the simulator, the highest NPV obtained is 5976.20 million USD. The computed NPV, by only employing proxy models, is 5854.37 million USD. The aforementioned scenario with the highest NPV of 5456.70 USD million was assumed to be the base case. By considering the NPVs obtained using the proxy models, it can be noticed that the GA resulted in an improvement of 12.4% (over the base case) whereas the PSO enhanced it by 7.29%. This shows that the optimality of the solution can be refined through the framework presented. Nonetheless, more studies need to be conducted to comprehensively discern if conducting further local proxy modeling enables a closer approximation to the “ground truth”.



**Figure 19.** Cumulative Density Frequency plot of absolute percentage error between actual NPV and predicted NPV (global proxy modeling).



**Figure 20.** Cumulative Density Frequency plot of absolute percentage error between actual NPV and predicted NPV (local proxy modeling).



**Figure 21.** Cumulative Density Frequency plot of NPVs.

Plots of GA-optimized FOPR, FWPR, FWIR, and FGPR are shown in Figure 22. The corresponding metrics are tabulated in Table 12. PSO-optimized rates are shown in Figure 23 and the respective metrics are tabulated in Table 13. Based on these tables, it can be concluded that the values of RMSE and AAPE in general correspond less satisfactorily to the

values of  $R^2$ . This is reflected by the error estimation shown by several data points in both Figures 22 and 23. Despite this fact, the proxy models still successfully capture the production profiles and serve their practical purposes. For illustration, only the cases with the highest NPV are shown in the plots. To avoid confusion, the term “simulator-proxies” refers to the results obtained from the reservoir simulator by implementing the optimal control produced by local proxy models. Based on these plots, the predictability of the local proxy models is further validated. The FOPR, FWPR, FWIR, and FGPR profiles obtained by the local proxy models generally match well with the profiles of simulator-proxies.

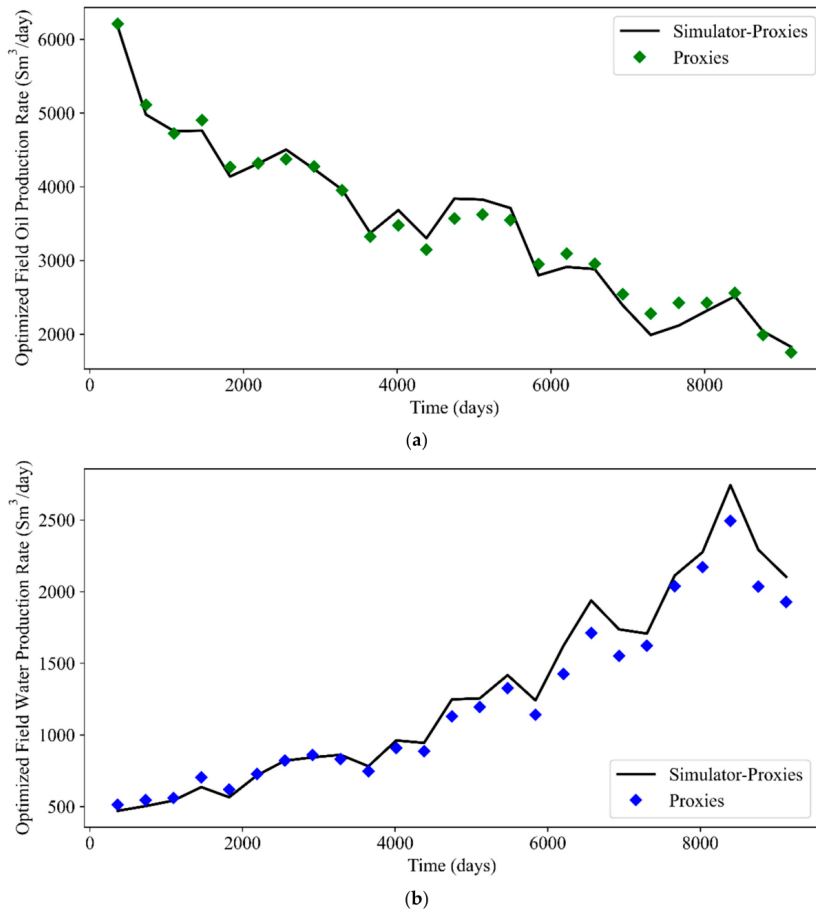


Figure 22. Cont.

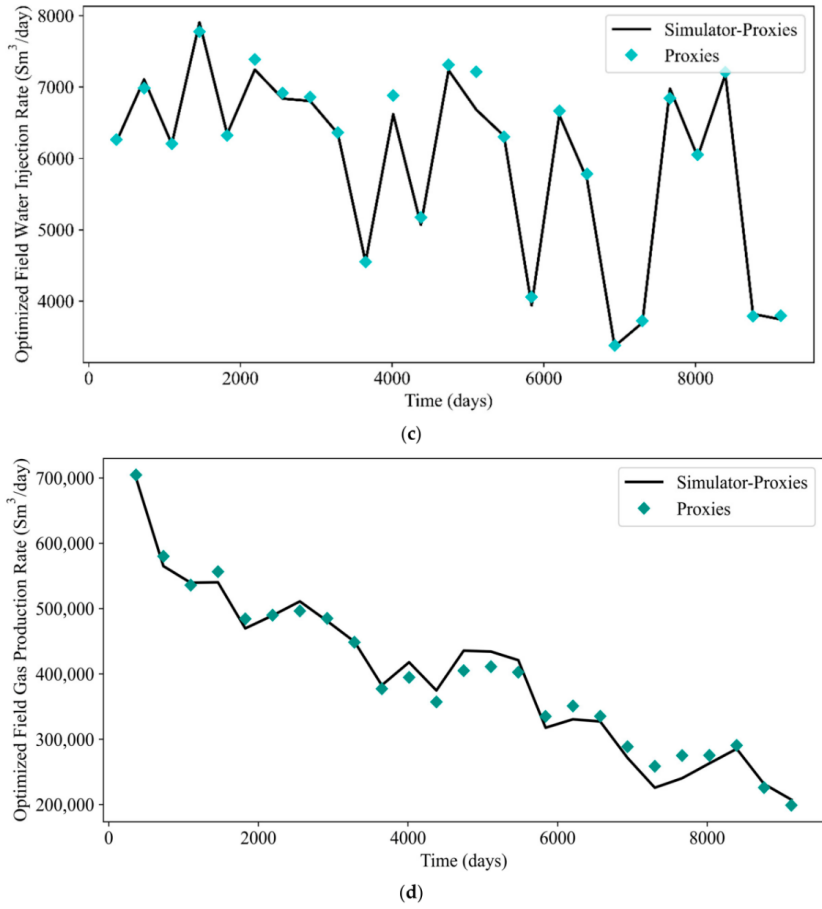
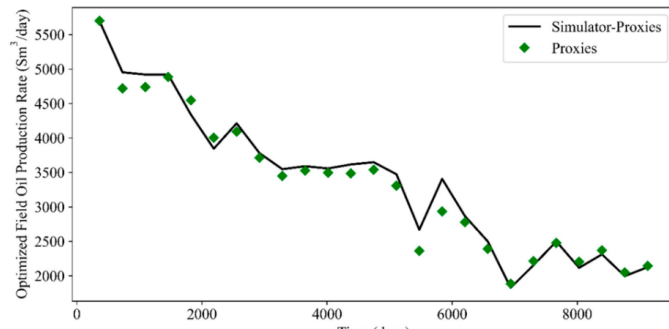


Figure 22. Plots of GA-optimized rates: (a) FOPR, (b) FWPR, (c) FWIR, and (d) FGPR.

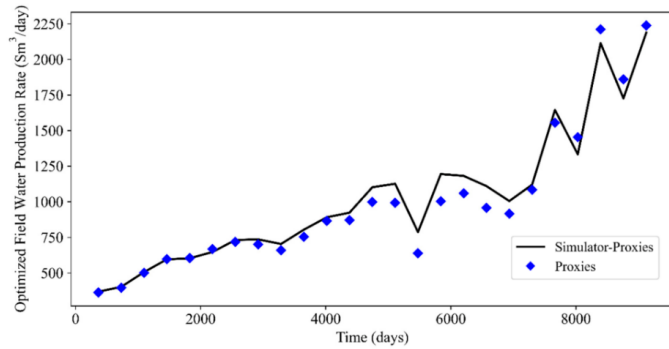
Table 12. Performance metrics of GA-optimized rates.

Optimized Rate	R <sup>2</sup>	RMSE	AAPE
FOPR	0.9808	150.89	0.042
FWPR	0.9656	120.59	6.746
FWIR	0.9888	138.05	1.398
FGPR	0.9808	17,118.74	4.207

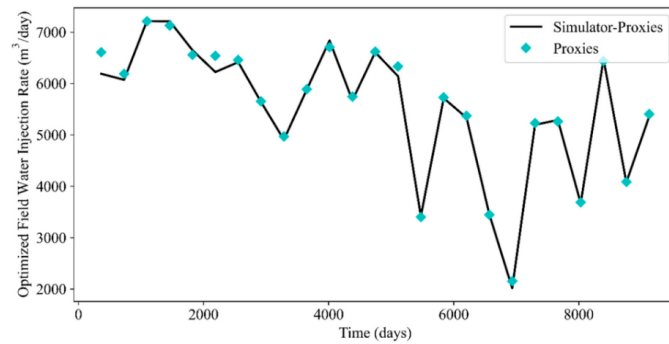




(a)



(b)



(c)

Figure 23. Cont.

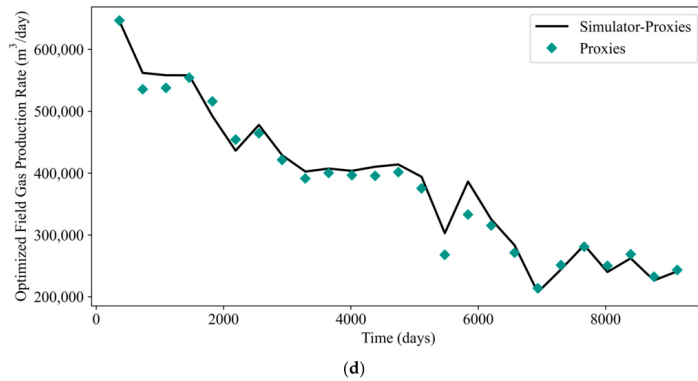


Figure 23. Plots of PSO-optimized rates: (a) FOPR, (b) FWPR, (c) FWIR, and (d) FGPR.

Table 13. Performance metrics of PSO-optimized rates.

Optimized Rate	R <sup>2</sup>	RMSE	AAPE
FOPR	0.9777	155.81	3.608
FWPR	0.9648	88.79	6.278
FWIR	0.9902	125.97	1.472
FGPR	0.9777	17,677.16	3.608

In general, the proposed framework has showcased good practical applications, considering the trade-off between accuracy and computational efficiency. Nonetheless, it is still subject to several limitations that are worth investigating further. The models developed from this framework are not “one-size-fits-all”. They are case-specific to serve the objective of the optimization problem under study. Furthermore, the proposed framework is yet to be verified in different optimization problems, such as well placement and choke optimization. This framework is limited to a geological realization and its maturity still needs to be justified considering geological uncertainty. Moreover, the proposed framework displays a good path to solving an optimization problem with 200 decision variables (a problem with a considerably high dimension). However, in terms of handling problems with even higher dimensionality, as reflected by most real-life applications, it is evident that several approaches can be integrated into this framework to reduce the pertinent dimension to increase its practicality. To the best of our knowledge, conducting production optimization with an efficiently reduced dimension of optimization variables, is still subject to extensive research. Regarding real-life applications, the proposed framework can also be extended to the paradigm of Top-Down Modeling [35] that only considers real field data to build the models.

Integrating another step of parameter optimization regarding both the structure of MLP and the variables of the nature-inspired algorithms will certainly be insightful. Attempting other advanced ML techniques, including Tree-based Pipeline Optimization Tool [36], can be researched to integrate the use of automated hyperparameter optimization in its workflow. In terms of solving a more sophisticated optimization problem, e.g., multi-objective optimization, the integration of NSGA-II (Non-dominated Sorting Genetic Algorithm II), suggested by Deb et al. [37], into the proposed framework can be considered. Some detailed studies are thus needed to achieve such enhancement by honoring the balance between computational speed and the accuracy of results predicted by the proxy models. Additionally, a combination of nature-inspired algorithms and derivative-based

algorithms can also be studied and possibly used instead of only applying the nature-inspired algorithms. This has a good potential to improve the exploitation component of optimization as the exploration is taken care of by nature-inspired algorithms [38].

## 7. Conclusions

In this work, we have presented a framework of methodology that couples proxy models with derivative-free algorithms to conduct waterflooding optimization. The approach of proxy modeling has been modified by introducing two different stages, namely global and local proxy modeling. Global proxy models were developed using a database that was generated by employing the sampling technique and reservoir simulation. Upon developing the global proxy models, an optimization algorithm was employed with these models to create a new database. This new database was then applied to develop more refined proxy models (the local proxy models). We have selected MLP as the ML method to develop the proxy models. For each stage of proxy modeling, we built three models to predict the output of FLPR, FWCT, and FWIR at every timestep. These output values were then utilized to compute the NPV for optimization purposes. The optimization was performed using GA and PSO. It is important to note that FGPR is also involved in the computation of NPV. However, for the optimization problem, the profile of FGPR is similar to that of FOPR since the solution gas oil ratio,  $R_s$ , remains constant for the whole production period.

The results obtained suggest that the two-stage proxy modeling can improve optimal solution. Such improvement is noticeable in terms of training, testing, blind validation, and optimization. Additionally, the computational efficiency of this framework is higher than solely relying on the reservoir simulator for optimization. The accuracy of results is not sacrificed upon attaining such a higher computational efficiency. This signifies the benefit of this framework for practical purposes. The primary objective of the proposed framework has been accomplished, although there are several limitations associated with it, such as lack of generalization and consideration of geological uncertainty. Nonetheless, a rudimentary framework has been successfully developed here, and further improvements should be considered for more real-life and robust applications. Detailed studies, including identifying the impact of each step of the framework (such as training of models and optimization), are recommended to strive for higher maturity of its employment.

**Author Contributions:** C.S.W.N.: Conceptualization, Methodology, Software, Formal Analysis, Data Curation, Investigation, Writing—original draft, Writing—reviewing and editing, Visualization. A.J.G.: Methodology, Writing—reviewing and editing, Supervision. W.W.: Software, Data Curation, Writing—original draft, Writing—reviewing and editing, Visualization. All authors have read and agreed to the published version of the manuscript.

**Funding:** The APC was funded by Norwegian University of Science and Technology.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This data can be found here: <https://www.unisim.cepetro.unicamp.br/benchmarks/br/unisim-i/unisim-i-d>.

**Acknowledgments:** This research is a part of BRU21—NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry (<http://www.ntnu.edu/bru21> (accessed on 14 February 2023)).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Russell, S.; Norvig, P. *Artificial Intelligence A Modern Approach*, 3rd ed.; Pearson: Hoboken, New Jersey, USA, 2010.
2. Mohaghegh, S. *Data-Driven Analytics for the Geological Storage of CO<sub>2</sub>*; CRC Press: Boca Raton, FL, USA, 2018.
3. Mohaghegh, S.D. *Shale Analytics*; Springer: Berlin/Heidelberg, Germany, 2017.
4. Nwachukwu, A.; Jeong, H.; Pyrcz, M.; Lake, L.W. Fast Evaluation of Well Placements in Heterogeneous Reservoir Models Using Machine Learning. *J. Pet. Sci. Eng.* **2018**, *163*, 463–475. [[CrossRef](#)]
5. Alakeely, A.; Horne, R.N. Simulating the Behavior of Reservoirs with Convolutional and Recurrent Neural Networks. *SPE Reserv. Eval. Eng.* **2020**, *23*, 992–1005. [[CrossRef](#)]

6. Alakeely, A.; Horne, R. Simulating Oil and Water Production in Reservoirs with Generative Deep Learning. *SPE Reserv. Eval. Eng.* **2022**, *25*, 751–773. [[CrossRef](#)]
7. Brundred, L.L.; Brudred, L.L., Jr. Economics of Water Flooding. *J. Pet. Technol.* **1955**, *7*, 12–17. [[CrossRef](#)]
8. Ng, C.S.W.; Jahanbani Ghahfarokhi, A.; Nait Amar, M. Application of Nature-Inspired Algorithms and Artificial Neural Network in Waterflooding Well Control Optimization. *J. Pet. Explor. Prod. Technol.* **2021**, *11*, 3103–3127. [[CrossRef](#)]
9. Ng, C.S.W.; Ghahfarokhi, A.J.; Nait Amar, M. Production Optimization under Waterflooding with Long Short-Term Memory and Metaheuristic Algorithm. *Petroleum* **2022**, *9*, 53–60. [[CrossRef](#)]
10. Chen, G.; Zhang, K.; Zhang, L.; Xue, X.; Ji, D.; Yao, C.; Yao, J.; Yang, Y. Global and Local Surrogate-Model-Assisted Differential Evolution for Waterflooding Production Optimization. *SPE J.* **2020**, *25*, 105–118. [[CrossRef](#)]
11. Chen, G.; Zhang, K.; Xue, X.; Zhang, L.; Yao, C.; Wang, J.; Yao, J. A Radial Basis Function Surrogate Model Assisted Evolutionary Algorithm for High-Dimensional Expensive Optimization Problems. *Appl. Soft Comput.* **2022**, *116*, 108353. [[CrossRef](#)]
12. Yang, X.-S. Chapter 1—Introduction to Algorithms. In *Nature-Inspired Optimization Algorithms*; Yang, X.-S., Ed.; Elsevier: Oxford, UK, 2014; pp. 1–21. ISBN 978-0-12-416743-8.
13. Nait Amar, M.; Jahanbani Ghahfarokhi, A.; Ng, C.S.W.; Zeraibi, N. Optimization of WAG in Real Geological Field Using Rigorous Soft Computing Techniques and Nature-Inspired Algorithms. *J. Pet. Sci. Eng.* **2021**, *206*, 109038. [[CrossRef](#)]
14. Nait Amar, M.; Zeraibi, N.; Jahanbani Ghahfarokhi, A. Applying Hybrid Support Vector Regression and Genetic Algorithm to Water Alternating CO<sub>2</sub> Gas EOR. *Greenh. Gases Sci. Technol.* **2020**, *10*, 613–630. [[CrossRef](#)]
15. Ng, C.S.W.; Nait Amar, M.; Jahanbani Ghahfarokhi, A.; Insland, L.S. A Survey on the Application of Machine Learning and Metaheuristic Algorithms for Intelligent Proxy Modeling in Reservoir Simulation. *Comput. Chem. Eng.* **2023**, *170*, 108107. [[CrossRef](#)]
16. Nait Amar, M.; Zeraibi, N.; Redouane, K. Bottom Hole Pressure Estimation Using Hybridization Neural Networks and Grey Wolves Optimization. *Petroleum* **2018**, *4*, 419–429. [[CrossRef](#)]
17. Ng, C.S.W.; Jahanbani Ghahfarokhi, A.; Nait Amar, M. Well Production Forecast in Volve Field: Application of Rigorous Machine Learning Techniques and Metaheuristic Algorithm. *J. Pet. Sci. Eng.* **2022**, *208*, 109468. [[CrossRef](#)]
18. McKay, M.D.; Beckman, R.J.; Conover, W.J. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics* **1979**, *42*, 55–61. [[CrossRef](#)]
19. Avansi, G.D.; Schiozer, D.J. UNISIM-I: Synthetic Model for Reservoir Development and Management Applications. *Int. J. Model. Simul. Pet. Ind.* **2015**, *9*, 21–30.
20. Ravenne, C.; Galli, A.; Doligez, B.; Beucher, H.; Eschard, R. Quantification of Facies Relationships via Proportion Curves. In *Geostatistics Rio 2000, Proceedings of the Geostatistics Sessions of the 31st International Geological Congress, Rio de Janeiro, Brazil, 6–17 August 2000*; Springer: Dordrecht, The Netherlands, 2002.
21. Gaspar, A.T.; Santos, A.; Maschio, C.; Avansi, G.; Filho, J.H.; Schiozer, D. *Study Case for Reservoir Exploitation Strategy Selection Based on UNISIM-I Field*; UNICAMP Universidade Estadual de Campinas: Campinas, Brazil, 2015.
22. Deutsch, C. Calculating Effective Absolute Permeability in Sandstone/Shale Sequences. *SPE Form. Eval.* **1989**, *4*, 343–348. [[CrossRef](#)]
23. Newman, G.H. Pore-volume compressibility of consolidated, friable, and unconsolidated reservoir rocks under hydrostatic loading. *J. Pet. Technol.* **1973**, *25*, 129–134. [[CrossRef](#)]
24. Hall, H.N. Compressibility of Reservoir Rocks. *J. Pet. Technol.* **1953**, *5*, 17–19. [[CrossRef](#)]
25. van der Knaap, W. Nonlinear Behavior of Elastic Porous Media. *Trans. AIME* **1959**, *216*, 179–187. [[CrossRef](#)]
26. Holland, J.H. Genetic Algorithms. *Sci. Am.* **1992**, *267*, 66–73. [[CrossRef](#)]
27. Lynch, M. Evolution of the Mutation Rate. *Trends Genet.* **2010**, *26*, 345–352. [[CrossRef](#)] [[PubMed](#)]
28. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In *Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995*.
29. Buduma, N.; Locascio, N. *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*; O'Reilly Media: Sebastopol, CA, USA, 2017; ISBN 9781491925614.
30. Ng, C.S.W.; Jahanbani Ghahfarokhi, A. Adaptive Proxy-Based Robust Production Optimization with Multilayer Perceptron. *Appl. Comput. Geosci.* **2022**, *16*, 100103. [[CrossRef](#)]
31. Kingma, D.P.; Ba, J.L. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015*.
32. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
33. Van Rossum, G.; Drake, F.L. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, USA, 2009.
34. Hemmati-Sarapardeh, A.; Larestani, A.; Nait Amar, M.; Hajrezaie, S. *Applications of Artificial Intelligence Techniques in the Petroleum Industry*; Gulf Professional Publishing: Houston, TX, USA, 2020.
35. Mohaghegh, S.D. *Data-Driven Reservoir Modeling*; Society of Petroleum Engineers: Richardson, TX, USA, 2017; ISBN 9788578110796.
36. Olson, R.S.; Moore, J.H. TPOT: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning BT—Automated Machine Learning: Methods, Systems, Challenges. In *Workshop on Automatic Machine Learning*; Hutter, F., Kotthoff, L., Vanschoren, J., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 151–160. ISBN 978-3-030-05318-5.

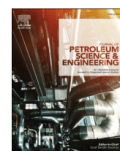
37. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
38. Alimo, S.R.; Beyhaghi, P.; Bewley, T.R. Optimization Combining Derivative-Free Global Exploration with Derivative-Based Local Refinement. In Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, Australia, 12–15 December 2017; pp. 2531–2538.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

## **Paper 7**

### ***Well production forecast in Volve field: Application of rigorous machine learning techniques and metaheuristic algorithm***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi, Menad Nait Amar



# Well production forecast in Volve field: Application of rigorous machine learning techniques and metaheuristic algorithm

Cuthbert Shang Wui Ng<sup>a,\*</sup>, Ashkan Jahanbani Ghahfarokhi<sup>a</sup>, Menad Nait Amar<sup>b</sup>

<sup>a</sup> Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

<sup>b</sup> Département Etudes Thermodynamiques, Division Laboratoires, Sonatrach, Boumerdes, Algeria

## ARTICLE INFO

### Keywords:

Production prediction  
Data-driven techniques  
Machine learning  
Support vector regression  
Neural networks  
Particle swarm optimization

## ABSTRACT

Developing a model that can accurately predict the hydrocarbon production by only employing the conventional mathematical approaches can be very challenging. This is because these methods require some underlying assumptions or simplifications, which might cause the respective model to be unable to capture the actual physical behavior of fluid flow in the subsurface. However, data-driven methods have provided a solution to this challenge. With the aid of machine learning (ML) techniques, data-driven models can be established to help forecasting the hydrocarbon production within acceptable range of accuracy. In this paper, different ML techniques have been implemented to build the models that predict the oil production of a well in Volve field. These techniques comprise support vector regression (SVR), feedforward neural network (FNN), and recurrent neural network (RNN). Particle swarm optimization (PSO) has also been integrated in training the SVR and FNN. These developed models can practically estimate the oil production of a well in Volve field as a function of time and other parameters: on stream hours, average downhole pressure, average downhole temperature, average choke size percentage, average wellhead pressure, average wellhead temperature, daily gas production, and daily water production. All these models illustrate splendid training, validation, and testing results with correlation coefficients,  $R^2$  being greater than 0.98. Moreover, these models show good predictive performance with  $R^2$  exceeding 0.94. Comparative analysis is also done to evaluate the predictability of these models.

## 1. Introduction

Accurate prediction of hydrocarbon production is necessary to ensure that the petroleum engineers have useful information to perform economic evaluation and optimization routines. Nonetheless, achieving high accuracy in production prediction is very challenging due to the sophistication of the subsurface conditions. Furthermore, the non-linearity between hydrocarbon production and any relevant petrophysical parameter often adds complexity to the modeling of production forecasting. Despite having successfully modeled the relationship between hydrocarbon production and any of these petrophysical parameters, lack of these data in real life raises additional difficulty (Ma and Liu, 2018). Therefore, developing a reliable predictive model of hydrocarbon production based upon available data has been one of the research interests in petroleum domain for few decades. This is because with such models, petroleum engineers will have a more profound understanding of the reservoir performance to solve any reservoir management-related issue.

One of the classical approaches in forecasting the hydrocarbon production is the decline curve analysis (DCA). This method was first developed by Arps (1945) and its application has been extended in the oil and gas industry (Fanchi et al., 2013; Hong et al., 2019; Jochen and Spivey, 1996). Due to its simple implementation, it is widely used as only historical production data is required. However, this illustrates that decline curve model is not robust as other important data, such as bottomhole pressure, wellhead pressure, choke size, etc. that affect the production are not considered. Being empirical in nature, it is also insufficient to fully reflect the physics of the fluid flow in subsurface and might either underestimate or overestimate the production estimate (Mohaghegh, 2017, 2020). Apart from DCA, numerical reservoir simulation (NRS) is another alternative applied to forecast the hydrocarbon production. Nonetheless, the predictive performance of the NRS is highly dependent on how the history matching (HM), which is a laborious task, is done (Liu et al., 2019). Additionally, NRS requires different data, including geological data, fluid properties, location of wells, etc. As new data is available in real time, the simulation model needs to be

\* Corresponding author.

E-mail address: [cuthbert.s.w.ng@ntnu.no](mailto:cuthbert.s.w.ng@ntnu.no) (C.S.W. Ng).

<https://doi.org/10.1016/j.petrol.2021.109468>

Received 9 July 2021; Received in revised form 26 August 2021; Accepted 3 September 2021

Available online 11 September 2021

0920-4105/© 2021 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

updated via HM to have a higher accuracy in production forecasting. Thus, the shortcomings of these methods are evident.

With the advancement of computing technology and data analytics, data-driven modeling has become another solution to hydrocarbon production forecasting. This method is not only simple to be implemented but can also capture the complex relationship between input and output of datasets provided. Data or measurement from real field is a representation of the “physics” that deciphers the “actual system” in the reservoir (Mohaghegh, 2017, 2020). Therefore, underlying assumption is not needed to simplify the physics in building a data-driven model that forecasts the production. In this context, the data-driven models learn the relationship between hydrocarbon production and other data obtained from real field through machine learning (ML) techniques: artificial neural network (ANN), support vector regression (SVR), etc. In recent years, the coupling of these ML methods with data analytics has achieved a great milestone in different domains of reservoir engineering, such as prediction of bottomhole pressure (Nait Amar et al., 2018; Nait Amar and Zeraibi, 2020), prediction of essential parameters needed in CO<sub>2</sub>-EOR (Nait Amar et al., 2020a; Nait Amar and Jahanbani Ghahfarokhi, 2020; Nait Amar and Zeraibi, 2018), optimization in water alternating CO<sub>2</sub>-EOR (Nait Amar et al., 2020b; Nait Amar and Zeraibi, 2019), waterflooding optimization (Ng et al., 2021a, 2021b), and forecast of hydrocarbon production (Aydin, 2015; Cao et al., 2016; Elmabrouk et al., 2014; Frausto-Solís et al., 2015; Zanjani et al., 2020).

Apart from these, coupling the application of metaheuristic algorithms with the ML techniques in data-driven modeling is another intriguing research domain. Metaheuristic algorithms are generally nature-inspired and derivative-free. Hence, their implementation is not only considered to be simplistic, but also powerful in terms of convergence to the global optimum (Ezugwu et al., 2020). Their employment in data-driven modeling has exhibited positive results as discussed by several literatures (Akande et al., 2017; Han and Bian, 2018; Nait Amar et al., 2018; Nait Amar and Zeraibi, 2020; Panja et al., 2018). On the other hand, a more advanced ANN technique: RNN, which Li et al. (2019) termed as deep learning, could also efficiently simulate the reservoir behaviors. Alakeely and Horne (2020) successfully implemented these deep learning methods to perform the estimation of bottomhole pressure. Moreover, Calvette et al. (2020) illustrated that RNN could be implemented to approximate the smart well production based upon a synthetic case study. The robustness of RNN was further demonstrated when it could also be coupled with ensemble Kalman filter (EnKF) to predict production of a waterflooded synthetic model (Bao et al., 2020). Besides, several literatures (Lee et al., 2019; Zhan et al., 2020) also highlighted the usefulness of RNN in forecasting the production from unconventional reservoirs. Thus, the use of ML in reservoir engineering shows a great potential.

Besides reservoir engineering, there are some contemporary works done on the employment of ML in the domains of production and drilling engineering. About production engineering, Mamudu et al. (2020) illustrated a dynamic risk analysis of petroleum production by developing ANN based on different geological realizations to help predicting the production. Bayesian network was also built to evaluate the risk of production. Moreover, Kondori et al. (2021) successfully established the connectionist models to evaluate the recovery performance of low water salinity injection. The connectionist models were developed with least squares support vector machine coupled with simulated annealing algorithm and adaptive network-based fuzzy inference system. Syed et al. (2020) also discussed how ML methods could be applied to optimize and conduct preventive maintenance on the artificial lift system. There are also other insightful literatures (Crnogorac et al., 2020; Khamis et al., 2020; Lin et al., 2020; Zhong et al., 2020) touching upon the implementation of ML in the production domain. For drilling engineering, Adedigba et al. (2018) conducted a risk assessment of offshore drilling operations with the help of data-driven model that is the Bayesian Tree Augmented Naïve Bayes algorithm. Fundamentally, this model could forecast the probability of kick that was updated in real time and utilized

to model the time dependent blowout risk. Additionally, Ozbayoglu et al. (2021) demonstrated the development of ANN by using the experimental data gathered and employed this ANN to optimize flow rate and speed of pipe rotation under effective cutting transport. Furthermore, there are other interesting contemporary literatures (Alali et al., 2021; Barbosa et al., 2019; Gan et al., 2020; Muojeke et al., 2020; Olukoga and Feng, 2021) about the application of ML in the drilling aspect.

This paper aims at applying different ML methods to develop data-driven models for the forecast of hydrocarbon production. Regarding the dataset, it is from a real-life well in Volve field (one of the latest databases released by Equinor (2020) to the public for research purposes) used to build the models. The details regarding the data will follow later. A portion of the data from the well is employed to develop the models whereas the remaining part of the data is used as the blind case to further verify the predictive performance of the models. About the ML methods, we first consider applying SVR and FNN. Also, we have employed particle swarm optimization (PSO) in the training of FNN and SVR models. Since hydrocarbon production is an example of time series data, RNN approach is also considered as it has been proven useful to forecast time series data (Alom et al., 2019; Connor et al., 1994; Zhang and Xiao, 2000). In terms of RNN modeling in this paper, three different types of RNNs: the simple RNN, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU), are developed. In total, seven data-driven models, which comprise FNN with backpropagation algorithm (FNN-BP), FNN trained with PSO (FNN-PSO), SVR tuned with trial-and-error approach (SVR-TE), hybrid model of SVR and PSO (SVR-PSO), simple RNN, LSTM, and GRU, have been established for comparative analysis on their respective predictive capabilities.

The paper is followed by some brief explanations regarding the theory of SVR, FNN, PSO, RNN, LSTM, and GRU. The next section discusses the methodology involved and explains how the available data is pre-processed and utilized in developing these models. The procedures in the development of the models are also expounded. The results and discussion will then follow prior to proceeding to conclusions that summarize the main findings of this work.

## 2. Theory

### 2.1. Support vector regression (SVR)

SVR is a subset of support vector machine that is an advanced supervised machine learning method that uses data for regression analysis, which was proposed by Vapnik (1995). It develops a function that can estimate the relationship between the desired outputs  $y = \{y_1, y_2, \dots, y_k\}$  defined on  $\mathbb{R}$ , and inputs  $x = \{x_1, x_2, \dots, x_k\}$  in which  $x_j \in \mathbb{R}$  and  $k$  is the number of data points. The function can be formulated as shown below:

$$f(x) = w \cdot \Psi(x) + b \quad (1)$$

$\Psi(x)$  refers to the function that maps the input space vector  $x$  into a high dimensional feature space to enable the initial non-linear problem to be expressed and conveniently solved as a linear regression function.  $w$  denotes the weight vector whereas  $b$  is the bias term. To determine  $w$  and  $b$ , the minimization of the following regularized risk function should be done as recommended by Vapnik (1995):

$$E(C) = \frac{C}{k} \sum_{j=1}^k L(f(x_j) - y_j) + \frac{1}{2} \|w\|^2 \quad (2)$$

In equation (2), the first term indicates the empirical error, and the second term means the degree of flatness of the function. Pertaining to this, the constant  $C$  acts as the penalty parameter that governs the trade-off between the complexity of the model and the empirical error. To solve for the empirical error, Vapnik (1995) suggested to use  $\epsilon$ -insensitive loss function which is represented below:



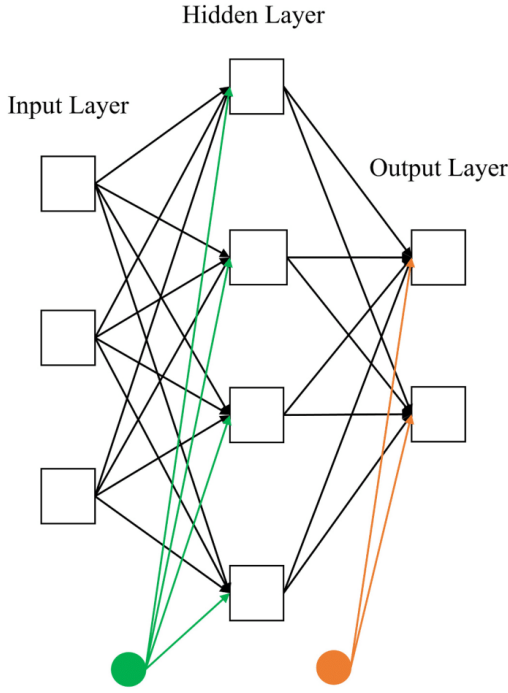


Fig. 1. The structure of an FNN model.

$$L(f(x) - y) = \begin{cases} 0, & \text{if } |f(x) - y| \leq \epsilon \\ |f(x) - y| - \epsilon, & \text{otherwise} \end{cases} \quad (3)$$

$\epsilon$  is the error tolerance. Thereafter, the parameters can be optimized in the following equation through the formulation of the constrained optimization problem (Forrester et al., 2008):

$$\begin{aligned} \min C & \sum_{j=1}^k (\xi_j^- + \xi_j^+) + \frac{1}{2} \|w\|^2 \\ \text{subject to} & = \begin{cases} y_j - (w \cdot \Psi(x) + b) \leq \epsilon + \xi_j^+ \\ (w \cdot \Psi(x) + b) - y_j \leq \epsilon + \xi_j^- \\ \xi_j^-, \xi_j^+ \geq 0, j = 1, 2, \dots, k \end{cases} \end{aligned} \quad (4)$$

$\xi_j^-$  and  $\xi_j^+$  are non-negative slack variables. To solve this constrained optimization problem, the optimization function can be transformed into dual space by using Lagrange multipliers (Shawe-Taylor and Cristianini, 2004). The obtained solution is shown below:

$$f(x) = \sum_{j=1}^k (\alpha_j - \alpha_j^*) K(x_j, x_m) + b \quad (5)$$

In equation (5),  $\alpha_j$  and  $\alpha_j^*$  are Lagrange multipliers which must fulfill the constraints of  $0 \leq \alpha_j$  and  $0 \leq \alpha_j^* \leq C$  whereas the term  $K(x_j, x_m)$  denotes the kernel function. In the literature (Forrester et al., 2008), there are different kernel functions available, but the commonly used ones include, but not limited to, radial basis function (RBF), polynomial function, and Gaussian function as illustrated in several literatures

(Chiroma et al., 2014; Kavzoglu and Colkesen, 2009; Qu and Zhang, 2016). In this paper, RBF is used as the kernel function and defined as shown below:

$$K(x_j, x_m) = \exp(-\gamma \|x_j - x_m\|) \quad (6)$$

where  $\gamma$  is the kernel parameter. The performance and accuracy of SVR is heavily influenced by the combination of  $\gamma$ ,  $C$ , and  $\epsilon$ . Therefore, implementing metaheuristic algorithms to optimize these parameters can be done to achieve an ideal performance of SVR. In addition, this can also overcome any inconvenience due to the use of traditional trial and error approach in tuning the parameters.

## 2.2. Feedforward neural network (FNN)

FNN is a ML algorithm that is formulated based on the functionalities of the biological neural networks. FNN comprises many calculating units which are known as artificial neurons or nodes. It has been demonstrated to be more successful in approximating the complex non-linear relationships between input and output vectors of a database than the conventional regression methods (Gharbi and Mansoori, 2005). There are different types of activation function used in FNN modeling, but the classical ones are the sigmoid function, the hyperbolic tangent, and the rectified linear unit (ReLU) function (Buduma and Locascio, 2017). In this paper, FNN, which is one of the most widely used ANNs as demonstrated in some literatures (Amini and Mohaghegh, 2019; Mohaghegh, 2011; Senthilkumar, 2010), is the chosen network with ReLU function as its activation function. It is also known as multilayer perceptron (MLP) and is made up of three layers, namely the input layer, the hidden layer, and the output layer. The topology of an arbitrary FNN is shown in Fig. 1. The green node is the bias node between the input and hidden layers whereas the orange node is the bias node between the hidden and output layers.

To ensure that the MLP learns the relationship between the input and output vectors of the database supplied, the MLP needs to undergo the training phase. Fundamentally, this training phase aims at optimizing the sets of weights and biases which minimize the pre-defined cost function, such as mean squared error (MSE). One of the classical methods of training is the backpropagation (BP) approach and it involves use of different algorithms, like steepest descent gradient, the Levenberg-Marquardt algorithm, the Powell-Beale conjugate gradient, Adam, and so on. In principle, after the forward propagation of the MLP, the resulting outputs will be compared with the targeted outputs. Errors are propagated back through the MLP in which the weights and biases are iteratively tuned and updated to achieve the optimum level. Apart from the conventional backpropagation algorithm, the metaheuristic algorithms can also be implemented to train the MLP. Therefore, in this paper, both backpropagation and metaheuristics algorithms are used to do the neural network training. Adam is the chosen backpropagation algorithm (Kingma and Ba, 2015) whereas Particle Swarm Optimization (PSO) is the metaheuristic algorithm used.

## 2.3. Particle swarm optimization (PSO)

PSO is an example of the metaheuristic population-based optimization algorithms that was proposed by Kennedy and Eberhart (1995) according to the social behavior of flying birds. The fundamental idea regarding the mechanism of PSO is that each particle corresponds to a potential solution to an optimization problem. The status of the particle is determined based upon its position and velocity in a dimensional space that is equal to the number of unknown parameters being optimized. Thereafter, the fitness value of the particle is computed by using a cost function such as MSE. Through several iterations, each particle updates its position until it converges to the optimum position through the minimization of the fitness value. In this context, pbest and gbest are determined at every iteration step. pbest refers to the local best position

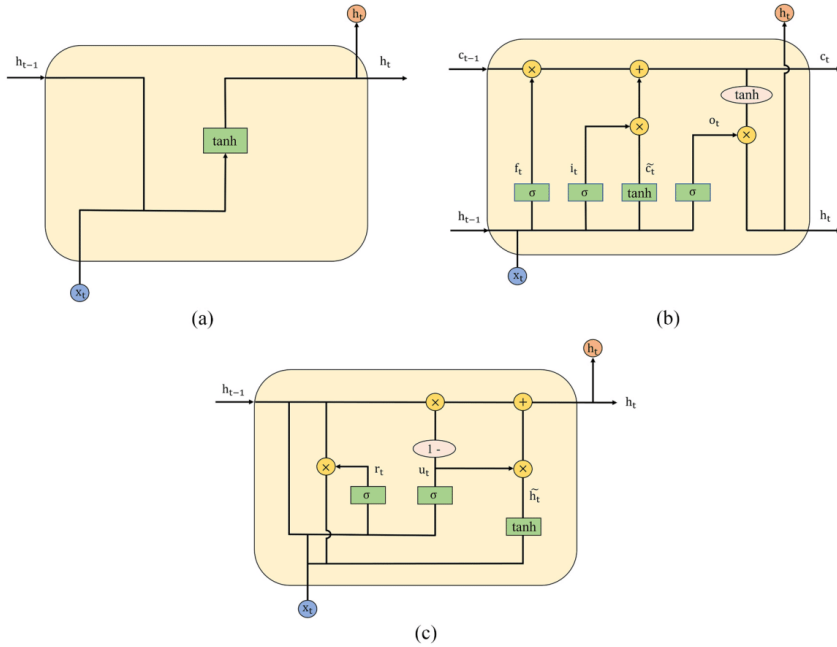


Fig. 2. Illustration of three types of RNN used in this study (a) simple RNN (b) LSTM (c) GRU.

Table 1

Data provided for each well in the Volve field.

Abbreviation from Database	Description
DATEPRD	Date of Record
ON_STREAM_HRS	On stream hours
AVG_DOWNHOLE_PRESSURE	Average Downhole Pressure
AVG_DOWNHOLE_TEMPERATURE	Average Downhole Temperature
AVG_DP_TUBING	Average Differential Pressure of Tubing
AVG_ANNULUS_PRESS	Average Annular Pressure
AVG_CHOKE_SIZE_P	Average Choke Size Percentage
AVG_WHP_P	Average Wellhead Pressure
AVG_WHT_P	Average Wellhead Temperature
BORE_OIL_VOL	Oil Volume from Well
BORE_WAT_VOL	Water Volume from Well
BORE_GAS_VOL	Gas Volume from Well
BORE_WI_VOL	Water Volume Injected
FLOW_KIND	Type of Flow (production or injection)
WELL_TYPE	Type of Well (oil production or water injection)

or the best position of a particle in the dimensional space (the lowest fitness value in this case) whereas gbest indicates the global best position or the overall best position of a particle hitherto in the entire population. The algorithm starts by randomly initializing the position and velocity of each particle. Thereafter, the respective fitness of each particle is computed in which pbest and gbest are determined and recorded. The velocity at current iteration step is then updated based on equation (7). The position of a particle for the next iteration step is updated based on equation (8). In the subsequent steps, positions and velocities of particles are updated iteratively by the pbest and gbest.

$$v_{jk,t+1} = \omega v_{jk,t} + c_1 r_1 (pbest_{jk,t} - x_{jk,t}) + c_2 r_2 (gbest_{k,t} - x_{jk,t}) \quad (7)$$

$$x_{jk,t+1} = x_{jk,t} + v_{jk,t+1} \quad (8)$$

Table 2

Selected input and output data for data-driven modeling.

Parameters	
Input Data	Units
Time	Days
On stream hours	hours
Average Downhole Pressure	bar
Average Downhole Temperature	°C (degree Celsius)
Average Choke Size Percentage	%
Average Wellhead Pressure	bar
Average Wellhead Temperature	°C (degree Celsius)
Gas Volume from Well	m <sup>3</sup> (daily)
Water Volume from Well	
Output Data	Units
Oil Volume from Well	m <sup>3</sup> (daily)

Table 3

Mean and standard deviation of input and output parameters of the production case considering all the data points.

Baseline Information		
Input and Output	Mean	Standard Deviation
Time	547	315.67
On stream hours	23.02	3.89
Average Downhole Pressure	261.01	15.54
Average Downhole Temperature	99.38	5.14
Average Choke Size Percentage	90.44	21.88
Average Wellhead Pressure	30.73	4.21
Average Wellhead Temperature	86.25	8.47
Gas Volume from Well	49,263.63	30,342.37
Water Volume from Well	3171.60	674.34
Oil Volume from Well	326.88	204.97

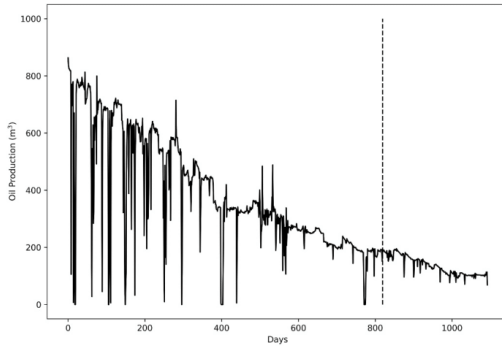


Fig. 3. Oil production of the well NO159-F-14H.

Table 4  
Parameters used in neural network training for both Adam and PSO.

Adam Parameters	Values
Number of iterations	2000
Learning rate	0.01
Exponential decay rates for the 1st moment estimates, $\beta_1$	0.9
Exponential decay rates for the 2nd moment estimates, $\beta_2$	0.999
Numerical stability constant, $\epsilon$	$10^{-7}$
PSO Parameters	Values
Number of iterations	2000
Number of particle swarms	100
Inertial Weight, $\omega$	0.8
Cognitive Learning Factor, $c_1$	1.05
Social Learning Factor, $c_2$	1.05

Table 5  
Optimized hyperparameters in SVR modeling.

Models	$\gamma$	C	$\epsilon$	$\kappa_1$	$\kappa_2$	$\kappa_3$
SVR-TE	0.5000	89.00	0.001000	-	-	-
SVR-PSO	0.4028	89.27	0.001802	0.4072	0.0171	0.5757

Table 6  
Performance metrics of the results estimated using the training, validation, and testing sets.

Datasets	Models	R <sup>2</sup>	RMSE
Training	SVR-TE	0.9951	13.88
	SVR-PSO	0.9944	14.68
	FNN-BP	0.9948	14.00
	FNN-PSO	0.9945	14.92
	Simple RNN	0.9945	14.46
	LSTM	0.9962	12.03
Validation	GRU	0.9962	12.17
	SVR-TE	0.9880	21.37
	SVR-PSO	0.9889	20.79
	FNN-BP	0.9911	19.13
	FNN-PSO	0.9923	15.75
	Simple RNN	0.9921	18.27
Testing	LSTM	0.9910	19.51
	GRU	0.9940	15.75
	SVR-TE	0.9764	30.83
	SVR-PSO	0.9936	16.61
	FNN-BP	0.9936	16.44
	FNN-PSO	0.9898	19.91
Testing	Simple RNN	0.9941	15.37
	LSTM	0.9922	17.64
	GRU	0.9915	18.24

In equation (7),  $v_{j,k,t}$  refers to the velocity of the  $j^{\text{th}}$  particle at iteration  $t$  in  $k^{\text{th}}$  dimension whereas  $x_{j,k,t}$  represents its corresponding position.  $c_1$  and  $c_2$  respectively refer to the cognitive and social learning factors which govern the local and global search of the best position. They are determined by trial-and-error approach.  $r_1$  and  $r_2$  are random numbers retrieved from uniform (0, 1).  $\omega$  is inertial weight that was recommended by Shi and Eberhart (1998) to enhance the convergence performance.

2.4. Recurrent neural network (RNN)

RNN is a subset of ANN, which is established to handle the input data that has sequential characteristics (Alakeely and Horne, 2020; Alom et al., 2019). Examples of these sequential inputs include sets of words or sentences, document texts, stock price, etc. Fundamentally, RNN can preserve any previous information to the current task and such ability widens its application in different aspects, including speech recognition (Amberkar et al., 2018; Graves et al., 2013) and language processing (Guan et al., 2019; Sutskever et al., 2014). The fundamental mechanism of a basic RNN is that information can be preserved and sent from the current to the successive step (Alom et al., 2019) as illustrated by its architecture as shown in Fig. 2a. Apart from this simple RNN, there are also other representations of RNN, such as Hopfield network, Echo state, Bi-directional, LSTM, GRU, and so forth. In this paper, we applied three examples of RNNs, including the simple RNN, LSTM, and GRU, to perform the well production forecast. The details regarding LSTM and GRU will be expounded later. The simple RNN used in this study consists of one hidden layer and one output layer and the respective mathematical formulation is presented below:

$$h_t = \gamma(W_h x_t + U_h h_{t-1} + b_h) \tag{9}$$

$$y_t = \gamma(W_y h_t + b_y) \tag{10}$$

where  $h_t$  is known as the vector of hidden-state or hidden layer. It is computed as shown in equation (9) by summing up three terms and placing the summation into the activation function that is represented as  $\gamma$ . In this work, the activation function used is the hyperbolic tangent. Also,  $y_t$  is the output vector that is determined by adding two terms into the activation function as shown in equation (10). For the other terms,  $x_t$  is the input vectors,  $W$  and  $U$  represent the weights, and  $b$  is the bias term. It is important to know that the subscripts  $t$  and  $t-1$  correspondingly refer to the current and previous timesteps. The subscript  $h$  indicates the properties of the hidden layer whereas the subscript  $y$  represents those of the output layer. The use of these notations also applies to the mathematical formulations of LSTM and GRU in the following sections. For LSTM, the subscripts  $f$ ,  $i$ ,  $c$ , and  $o$  correspondingly denote the relevant properties of forget gate, input gate, cell state and output gate. For GRU, the subscripts  $u$  and  $r$  respectively mean the properties of update gate and reset gate. The pertinent details will follow later.

2.5. Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU)

Albeit the simple RNN can be practically robust, it still has a limitation, namely having the problem of vanishing gradient (Alom et al., 2019; Hochreiter and Schmidhuber, 1997; Li et al., 2019). This limitation circumvents the simple RNN from exploiting the long-term information (Alom et al., 2019; Hochreiter and Schmidhuber, 1997; Li et al., 2019). This implies that it is unable to store large amount of information from previous iterations for a more accurate prediction of the outputs. Therefore, more complicated versions of RNN, which are LSTM and GRU, have been utilized. LSTM was first developed by Hochreiter and Schmidhuber (1997) to ensure the long-term dependencies on the previous information. The architecture of the LSTM employed in this study is portrayed in Fig. 2b. The respective formulas are expressed below:

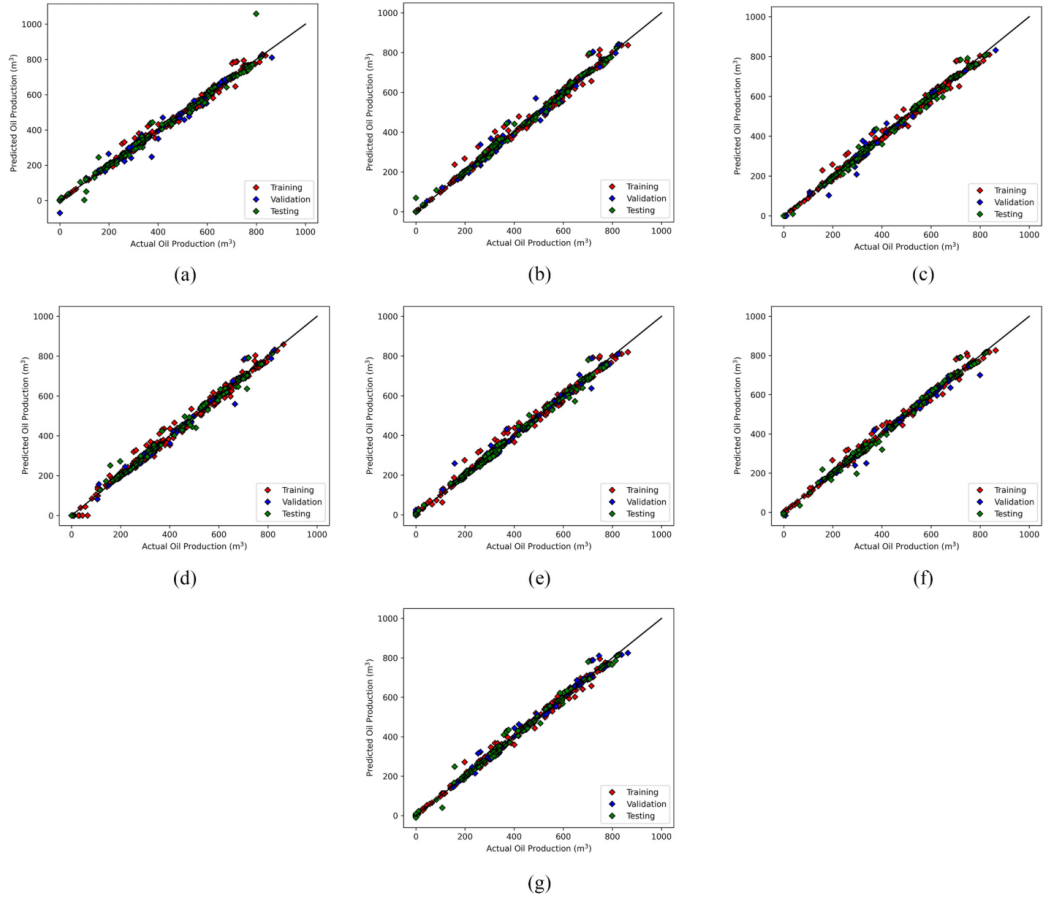


Fig. 4. Cross plot of the actual and predicted oil production (a) SVR-TE (b) SVR-PSO (c) FNN-BP (d) FNN-PSO (e) simple RNN (f) LSTM (g) GRU.

**Table 7**  
Performance metrics of the results estimated by using the blind case.

Datasets	Models	R <sup>2</sup>	RMSE
Blind Validation	SVR-TE	0.9476	7.34
	SVR-PSO	0.9644	6.04
	FNN-BP	0.9538	6.89
	FNN-PSO	0.9574	6.61
	Simple RNN	0.9665	5.87
	LSTM	0.9712	5.45
	GRU	0.9700	5.56

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{11}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{12}$$

$$\tilde{c}_t = \gamma(W_c x_t + U_c h_{t-1} + b_c) \tag{13}$$

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t \tag{14}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{15}$$

$$h_t = o_t \times \gamma(c_t) \tag{16}$$

The fundamental idea of LSTM revolves around a cell state  $c_t$  (shown as the horizontal top line in Fig. 2b) in which the addition or removal of any information is conducted through three gates, namely forget gate  $f_t$ , input gate  $i_t$ , and output gate  $o_t$  (Alom et al., 2019). These gates make assessments as if the sequential input data is valuable or not to be kept (Alom et al., 2019; Li et al., 2019). By doing so, relevant information can be preserved to the downstream. First, the forget gate plays a pivotal role to decide if information should be kept or omitted based upon equation (11). In this aspect, the information in the form of input and hidden state will be discarded (retained) if  $f_t$  approximates zero (one) (Li et al., 2019). Pertaining to the input gate, it is computed to update the cell state and through this update, the importance of the input being sent to the next cell is assessed. Moreover, about the output gate, it determines the output for the hidden states as shown in equation (16). It can be noticed that the recurrent activation function used in LSTM is a sigmoid function that is denoted as  $\sigma$ .

GRU is another development of RNN, which was initiated by Cho et al. (2014), that is employed in this paper. As compared to LSTM, GRU

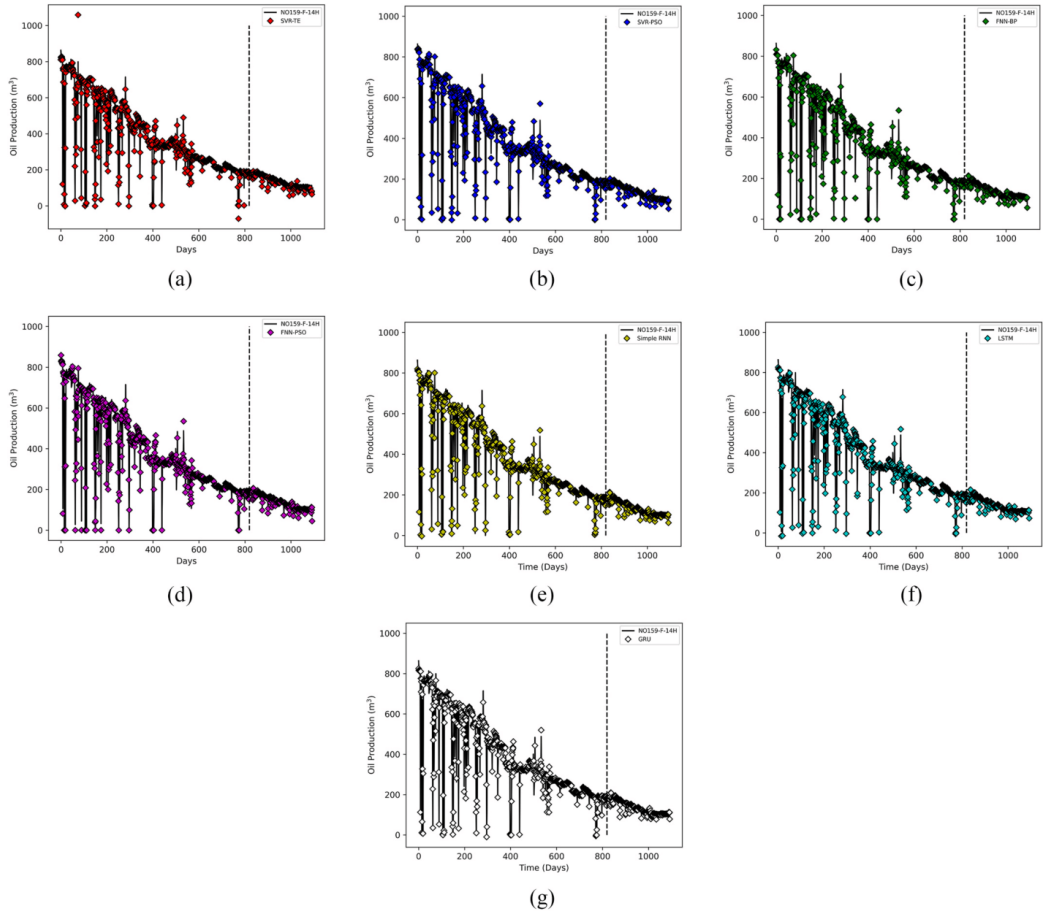


Fig. 5. Oil production profile (a) SVR-TE (b) SVR-PSO (c) FNN-BP (d) FNN-PSO (e) simple RNN (f) LSTM (g) GRU.

**Table 8**  
Performance metrics of all seven models considering all data points.

Datasets	Models	R <sup>2</sup>	RMSE
All	SVR-TE	0.9935	16.52
	SVR-PSO	0.9952	14.21
	FNN-BP	0.9956	13.65
	FNN-PSO	0.9952	14.15
	Simple RNN	0.9957	13.51
	LSTM	0.9961	12.69
	GRU	0.9964	12.28

only consists of two gates, which are the reset gate  $r_t$  and the update gate  $u_t$ . The function of the reset gate is to evaluate as if new information should be passed, which is like those of forget and input gates (Li et al., 2019). Thereafter, the reset gate decides on how extensively the previous information should be forgotten. According to the formulas of GRU shown below, it can be inferred that its simpler framework enables it to be more computationally favorable as compared to LSTM (Alom et al., 2019).

$$u_t = \sigma(W_u x_t + U_u h_{t-1} + b_u) \tag{17}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{18}$$

$$\tilde{h}_t = \gamma(W_h x_t + U_h [r_t \times h_{t-1}] + b_h) \tag{19}$$

$$h_t = (1 - u_t) \times h_{t-1} + u_t \times \tilde{h}_t \tag{20}$$

### 3. Methodology

Having a good model that helps predicting hydrocarbon production is crucial in reservoir management. As mentioned previously, we have developed seven models in this work: FNN-BP, FNN-PSO, SVR-TE, SVR-PSO, simple RNN, LSTM, and GRU. To build these data-driven models, we need to first know the source of data because it is the main building blocks of these models. The details regarding the data will follow.

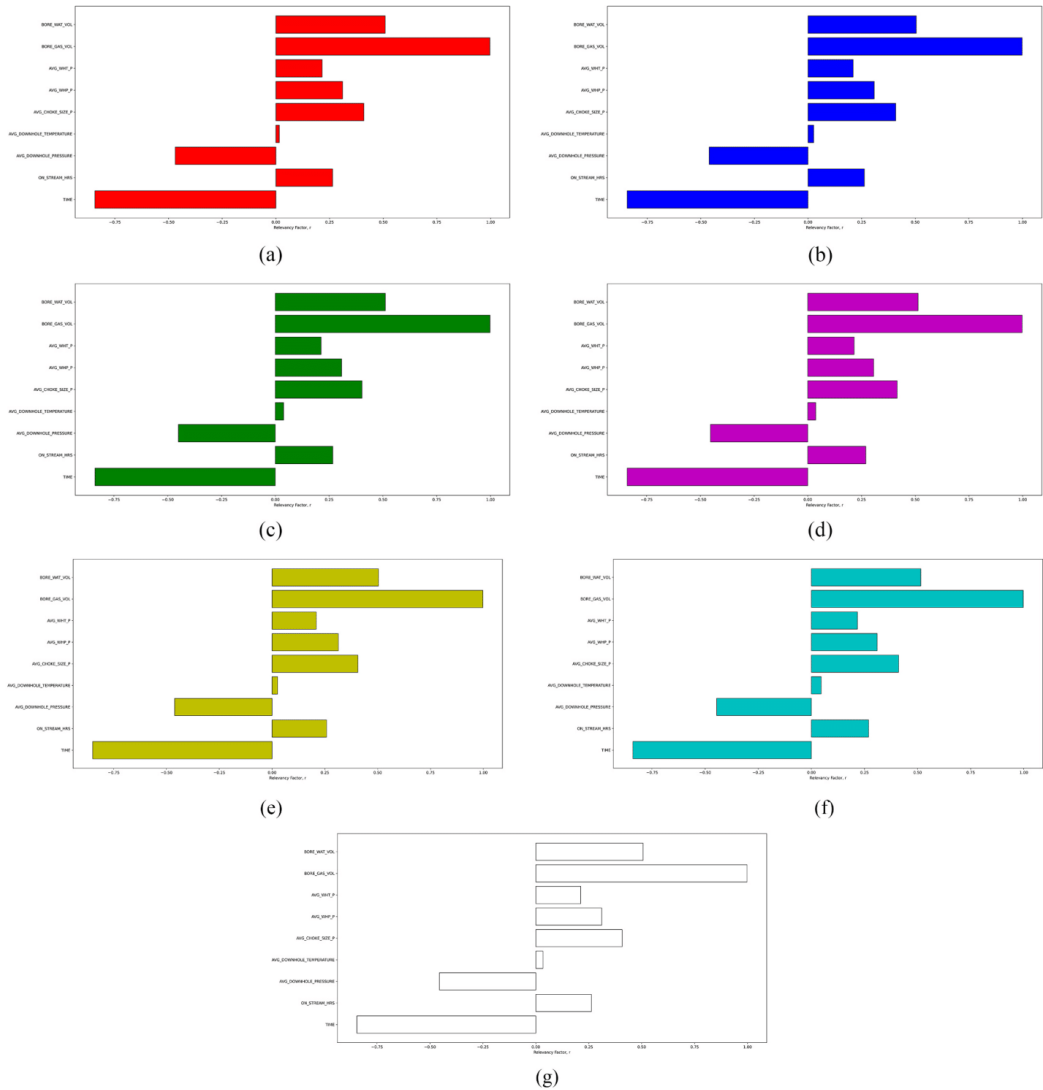


Fig. 6. Relative effect of each input parameter on the output predicted by each model considering all data points (a) SVR-TE (b) SVR-PSO (c) FNN-BP (d) FNN-PSO (e) simple RNN (f) LSTM (g) GRU.

3.1. Field data

In this work, the data from Volve field (Equinor, 2018) on the Norwegian Continental Shelf was utilized. According to the field development plan report retrieved from Equinor (2020), Volve field is a 2 km by 3 km oil-bearing reservoir and is located at a depth between 2750 m and 3210 m below sea level. It comprises sandstone and has average properties with permeability of about 1000 mD (from well testing), porosity of 0.21, and net-to-gross ratio of 0.93. The water saturation of oil-bearing zone is on average 0.2. At the depth of 3060 m, the reservoir

pressure and temperature are 340 bar and 110 °C, respectively. Pertaining to the characteristics of crude oil from Volve field, according to ExxonMobil (2018), the API gravity is 29.1°, the specific gravity is 0.881, and the viscosity at 20 °C is 22.5 cSt. For more details, kindly peruse the crude oil assay released by ExxonMobil (2018).

Equinor (2018) has released this database to public in May 2018 for the purpose of research and development. In this aspect, there are different types of data in the database, including seismic data, well log data, reservoir simulation model, etc. However, only the real-field production data is used in this study. Regarding the production data,

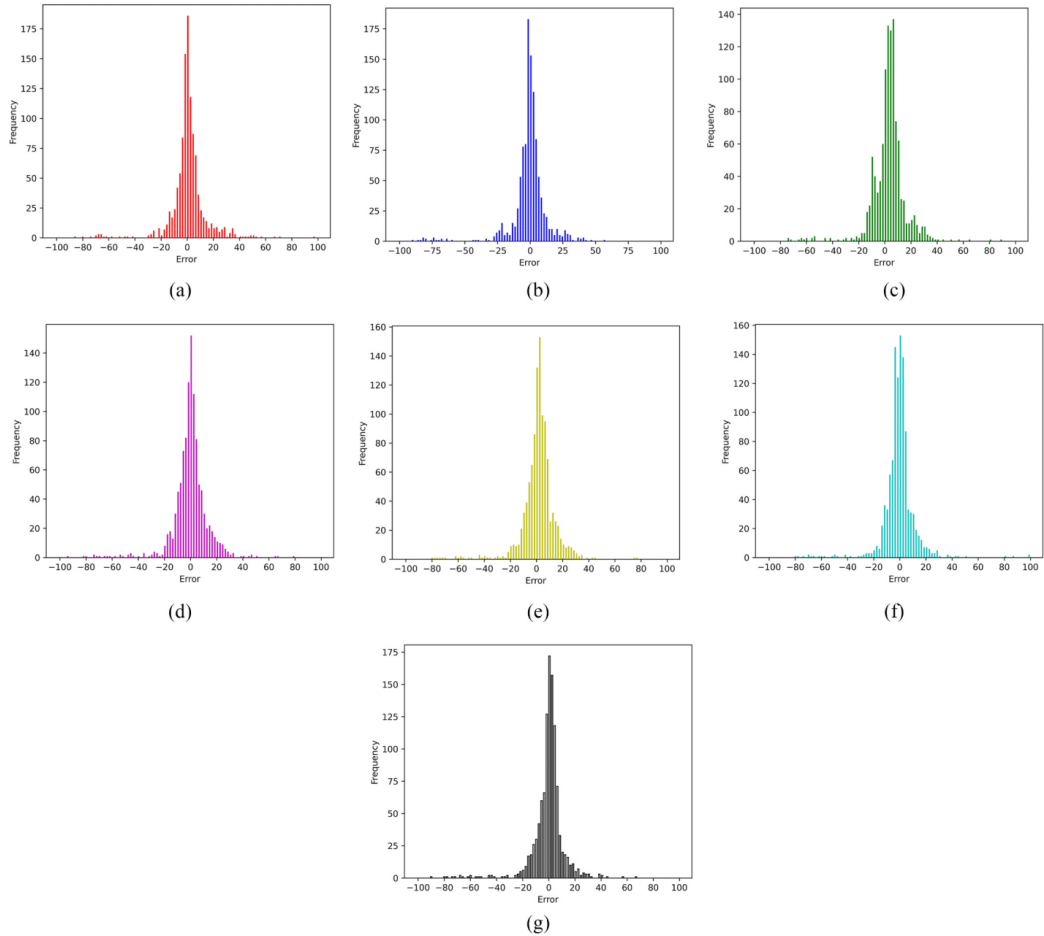


Fig. 7. Distribution of errors (a) SVR-TE (b) SVR-PSO (c) FNN-BP (d) FNN-PSO (e) simple RNN (f) LSTM (g) GRU.

it consists of the data of 7 wells, namely NO15/9-F-1 C, NO15/9-F-11H, NO15/9-F-12H, NO159-F-14H, NO15/9-F-15D, NO15/9-F-4AH, and NO15/9-F-5AH. Each well consists of the data as shown in Table 1:

The production data was recorded daily. For illustrative purpose, only the well NO159-F-14H is used in this study. For this well, the production period lasts from February 2008 to September 2016. However, for practical purpose, only the data between July 2013 and July 2016, which lasts for 1093 days, is used. In addition to this, not all the data provided will be used and the selected data used for data-driven modeling is presented in Table 2. The selection of input and output data was done based upon knowledge of reservoir and production engineering, but it can be conveniently done by using feature selection method (Zanjani et al., 2020). To further facilitate the readers' understanding of the production scenario, the mean and standard deviation of each parameter are determined and presented in Table 3. In addition, the oil production profile of the well NO159-F-14H between July 2013 and July 2016 is plotted in Fig. 3. The dashed vertical line in Fig. 3 will be explained later.

### 3.2. Model development

The data needs to be pre-processed before it is used to build the models. As explained earlier, there are 10 types of data being utilized and each type contributes to 1093 data points. Hence, this sums up to 10,930 data points. Each data point is then normalized as follows:

$$x_{i,\text{normalized}} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (21)$$

In equation (21),  $x_{i,\text{normalized}}$  is the normalized value of  $x_i$  that is any data point out of the 1093 data points under each type of data as shown in Table 2.  $x_{\max}$  and  $x_{\min}$  denote the maximum and minimum values under each data type in Table 2. Thereafter, the normalized data points are divided into two different sets, namely the modeling set and the prediction set, based on a ratio of 7.5:2.5. This implies that the first 8190 data points out of 10,930 data points will be employed to develop the data-driven models whereas the remaining 2740 data points are used as the blind case to evaluate the predictive performance of the models. It is essential to divulge that the division for modeling and prediction sets is

done arbitrarily for practical purposes. It relies upon the consideration of the modeler about the size of the dataset preserved for prediction. For a more vivid illustration, the modeling set corresponds to the data points on the left of the dashed vertical line in Fig. 3 whereas the prediction set corresponds to the right of the line. Besides, 70% of the data points from the modeling set is used as the training set and the remaining 30% is equally divided into the validation and testing sets. In this context, only the training set is utilized to develop and train the models. The validation set is employed to prevent the overfitting of the models whereas the testing set ensures that the models have a good predictive performance prior to being verified by the data from the blind case (Mohaghegh, 2017). The performance of the models is determined by using two different metrics, which are the correlation coefficient  $R^2$  and the root mean squared error (RMSE). The formulas of the performance metrics are presented as follows:

$$R^2 = 1 - \frac{\sum_{j=1}^N (q_j^{\text{exp}} - q_j^{\text{cal}})^2}{\sum_{j=1}^N (q_j^{\text{cal}} - \bar{q})^2} \quad (22)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{j=1}^N (q_j^{\text{exp}} - q_j^{\text{cal}})^2} \quad (23)$$

where  $N$  means the total number of data,  $q_j^{\text{exp}}$  is the actual oil production at timestep  $j$ ,  $q_j^{\text{cal}}$  is the oil production estimated by the models at timestep  $j$ , and  $\bar{q}$  is the mean actual oil production. For the development of FNN-BP, FNN-PSO, and the three RNNs, the data from the training set is fed into the neural network to enable the network to learn the relationship between input and output data. Pertaining to this, the pre-defined cost function implemented in the neural network training is the MSE. Therefore, during the training phase, the weights and biases will be iteratively adjusted as explained to minimize the cost function.

Pertaining to the specifics of the data-driven models, the architectures of both FNNs are the same, which include one input layer with 9 nodes, one hidden layer with 30 nodes, and one output layer with only one node. For the three RNNs, each of them also comprises only one hidden layer and one output layer. Besides that, each of the three RNN representations also has 30 hidden nodes and 1 output node. The number of hidden nodes and layers for both FNNs and RNNs is determined by using the trial-and-error approach. The relevant parameters used to conduct this neural network training phase are presented in Table 4. From Table 4, it is better to reiterate that Adam has only been implemented to train all the RNNs and FNN-BP. For FNN-PSO, since each of the weights (biases) is represented as one particle, the number of particle swarms is the number of sets of particles employed in the training phase.

Regarding the development of SVR and SVR-PSO, it is important to achieve the optimum values of the hyperparameters  $\gamma$ ,  $C$ , and  $\epsilon$  to develop models with good performance. For SVR-PSO, the hyperparameters are tuned such that the objective function will be minimized. The objective function consists of the corresponding MSE of the training, validation, and testing sets, and it is expressed as shown in equation (24). For SVR-PSO, there are three additional parameters to be adjusted, namely the weighting factors  $\kappa_1$ ,  $\kappa_2$ , and  $\kappa_3$  for each MSE. To conduct the tuning with PSO, 200 iterations and 20 particle swarms are used. Furthermore, the inertial weight used here is 0.40 while both learning factors are 1.05. Refer to Table 5 for the values of these optimized hyperparameters.

$$\text{MSE}_{\text{SVR-PSO}} = \kappa_1 \times \text{MSE}_{\text{Training}} + \kappa_2 \times \text{MSE}_{\text{Validation}} + \kappa_3 \times \text{MSE}_{\text{Testing}} \quad (24)$$

To generate the initial population of the swarm particles, we used the distribution of uniform (0.01, 1.5) for  $\gamma$ , uniform (12, 90) for  $C$ , uniform (0.0001, 0.1) for  $\epsilon$ , uniform (0, 0.5) for  $\kappa_1$  and uniform (0, 0.5) for  $\kappa_2$ . Without determining the initial  $\kappa_3$ , we optimized it by subtracting the sum of optimized  $\kappa_1$  and  $\kappa_2$  at each iteration from 1.

## 4. Results and discussion

We have established seven data-driven models to predict the daily oil production of a real-field well. To determine if these models will exhibit excellent predictive performance, their corresponding training performance need to be evaluated first. Pertaining to this, the models with excellent training results will generally be able to produce predictions within a good level of accuracy. In this work, the training performance of each of the seven models is presented in Table 6. In addition to the performance metrics computed using the training data, those calculated using the validation and testing data are also shown. During the development stage, if the models demonstrate good performance with the validation data, it implies that the overfitting issue may be eluded. Thereafter, the predictability of the models can be evaluated using the testing data. It is important to understand that only the training data is employed to build the models. The other data is utilized to provide useful insights regarding the training process.

From Table 6, it is inferred all the seven models demonstrate excellent results of training, validation, and testing with  $R^2$  exceeding 0.99 and RMSE being at most 30.83. To be more precise, LSTM has the best training performance in terms of  $R^2$  and RMSE compared to the other models. However, when the models are fed with the validation data, GRU exhibits the best results. In addition to this, as the models are verified with the testing data, simple RNN performs the best. Therefore, RNN-based models generally illustrate better outcomes than both SVR-based and FNN-based models in terms of training, validation, and testing. Despite these better results exhibited by these RNN-based models, the performances of SVR-based and FNN-based models are deemed to be practically excellent. Nevertheless, the performance metrics shows that all models have undergone an excellent development phase. We need to be cognizant that having satisfactory modeling outcome does not necessarily imply that the models can directly be used. They still must be evaluated by the data from the blind case to further verify their robustness.

The cross-plots of the actual and the predicted oil production are presented for SVR-TE in Fig. 4a, SVR-PSO in Fig. 4b, FNN-BP in Fig. 4c, FNN-PSO in Fig. 4d, simple RNN in Fig. 4e, LSTM in Fig. 4f, and GRU in Fig. 4g. In general, most of the data points lie on the 45° line which indicates high accuracy. Nevertheless, Fig. 4a exhibits that there is an outlier of the validation data being less than zero and another outlier of the testing data being highly overestimated. This implies that the overall training performance can still be improved albeit the performance metrics suggest otherwise. Moreover, Fig. 4d shows that there are some outliers from the training data that are underestimated by FNN-PSO. These outliers do not greatly affect the overall training performance of the model but contribute to the relatively less satisfying training performance compared with FNN-BP. For the RNN-based models, these plots generally add more confidence that the overall training performance of each of the three models is practically excellent. Additionally, there is no obvious outlier being detected in the plots, which are produced by using these models.

After the modeling phase is completed, we need to provide data from the blind case to justify if the models are ready to be employed. As explained, the data from the blind case is retrieved from the data points of the remaining 274 days. When these data are supplied into the built models, their performance metrics are calculated and recorded in Table 7. For a more vivid illustration, all the data points (1093 data points of oil production) are plotted alongside the prediction yielded by all the seven models in Fig. 5. For clarification, the statistics provided in Table 7 only consider the data points on the right side of the vertical dashed line in the figures. Based on Table 7, it can be observed that the use of PSO improves the predictive performance of the models in this work. For SVR, using PSO to tune the hyperparameters improves the  $R^2$  by 1.77% and the RMSE by 17.7%. Therefore, using a metaheuristic algorithm to tune the hyperparameters does not only reduce the computational effort, but also helps to attain a higher accuracy of



prediction. For FNN modeling, when PSO is utilized to conduct the training, the  $R^2$  and RMSE are respectively enhanced by 0.38% and 4.06%. Albeit the improvement is not significant, it provides useful insight that the application of metaheuristic algorithm is viable in modeling FNN and can have a good predictive performance.

Moreover, it is deduced that LSTM has the best performance with  $R^2$  being greater than 0.97 and RMSE being about 5.4. However, it is also important to observe that in this study, the performance of LSTM is slightly better than those of GRU and simple RNN. With respect to simple RNN, LSTM correspondingly improves  $R^2$  and RMSE by 0.49% and 7.2% whereas the enhancements induced by GRU are respectively 0.36% and 5.3%. In other words, the improvement of prediction accuracy is not very significant by applying more complicated representation of RNN. Therefore, from Fig. 5, the robustness of ML techniques in capturing the fluctuating trend of the data is clearly portrayed. In this context, the conventional DCA approach is only able to perform the “curve fitting” and reflect the general declining trend of the data. In addition to this, for the purpose of more comprehensive comparison, the performance metrics considering all the 1093 data points are calculated and tabulated in Table 8 for each model. As the result shows, GRU outperforms the other models. In general, all the models can capture the overall trend of the data points. Nonetheless, for SVR-TE, it can be noted that there are both overestimation and underestimation of values in two of the data points. This corresponds to the outliers mentioned earlier. Despite this, SVR-TE still performs reasonably well in estimating the output of the data from the blind case.

Furthermore, the relevancy factor ( $r$ ) has been implemented to evaluate the relative importance of these input variables on the predicted output by the models. In this case, higher absolute value of  $r$  indicates more significant relative effect on the output (Chen et al., 2014; Nait Amar, 2020; Nait Amar et al., 2021). The relevancy factor can be mathematically expressed as follows:

$$r(k_k, q) = \frac{\sum_{j=1}^N (k_{kj} - \bar{k}_j)(q_j - \bar{q})}{\sqrt{\sum_{j=1}^N (k_{kj} - \bar{k}_j)^2 \sum_{j=1}^N (q_j - \bar{q})^2}} \quad (25)$$

In equation (25), the data point index (or timestep in this case) is indicated as  $j$ ,  $k_k$  denotes the  $k$ th input parameter, and  $\bar{k}_j$  means the respective average value. Besides that,  $q$  and  $\bar{q}$  correspondingly represent the predicted output value and its average. The relevancy factor of each input parameter is depicted in Fig. 6. As shown, gas volume from well (or gas production) has the most influential impact on the output, which is oil volume from well (oil production). Distribution of the errors corresponding to the predictions (of all data points) performed by all the seven models are also demonstrated as histogram in Fig. 7. It can be observed that all seven models display a normal distribution that has a center being close to errors with zero values. Such distribution provides extra confidence to the integrity and robustness of the models developed in this paper.

## 5. Conclusions

In this work, SVR-TE, SVR-PSO, FNN-BP, FNN-PSO, simple RNN, LSTM, and GRU models have been developed to predict the oil production of a well in Volve field. These models have been trained, validated, and tested to ensure that they have learnt the relationship between input and output models before being blind validated.

Generally, RNN-based models outperformed the SVR-based and FNN-based models in terms of training and prediction. To be more specific, LSTM outperformed the other six models in the case of training. Besides that, GRU performed the best in the validation phase whereas simple RNN yielded the best outcome in the testing phase. However, the training performance and predictability of SVR-based and FNN-based models are still practically excellent. Apart from these, we can infer

that PSO contributes to the enhancement of SVR modeling in terms of training, but not in the case of FNN modeling due to the existence of several outliers. Nevertheless, we illustrated that the application of PSO in data-driven modeling could induce improvements although such improvements might not be significant for FNN modeling. Additionally, during the prediction phase, LSTM produced the most accurate results. Also, when considering all the data points, the performance metrics computed by using the results estimated by GRU were the best. Finally, the resemblance of the error distribution produced by each predictive model to a normal distribution with center close to zero further displayed the reliability of the models built in this work.

## Authors' contributions

Cuthbert Shang Wui Ng: Methodology, Data Curation, Analysis and Investigation, Modeling, Software, Writing, Editing, Revising. Ashkan Jahanbani Ghahfarokhi: Supervision, Methodology, Writing, Reviewing, Revising, and Editing. Menad Nait Amar: Methodology, Writing, Reviewing, Editing and Revising.

## 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.

## Acknowledgement

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)).

## References

- Adedigba, S.A., Oloruntobi, O., Khan, F., Butt, S., 2018. Data-driven dynamic risk analysis of offshore drilling operations. *J. Petrol. Sci. Eng.* 165 <https://doi.org/10.1016/j.petrol.2018.02.049>.
- Akande, K.O., Owolabi, T.O., Olatunji, S.O., AbdulRaheem, A.A., 2017. A hybrid particle swarm optimization and support vector regression model for modelling permeability prediction of hydrocarbon reservoir. *J. Petrol. Sci. Eng.* 150 <https://doi.org/10.1016/j.petrol.2016.11.033>.
- Alakeely, A., Horne, R.N., 2020. Simulating the behavior of reservoirs with convolutional and recurrent neural networks. In: *SPE Reservoir Evaluation and Engineering*. <https://doi.org/10.2118/201193-PA>.
- Alali, A.M., Abughaban, M.F., Aman, B.M., Ravela, S., 2021. Hybrid data driven drilling and rate of penetration optimization. *J. Petrol. Sci. Eng.* 200 <https://doi.org/10.1016/j.petrol.2020.108075>.
- Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Hasan, M., Van Essen, B.C., Awwal, A.A.S., Asari, V.K., 2019. A state-of-the-art survey on deep learning theory and architectures. *Electron.* <https://doi.org/10.3390/electronics8030292>.
- Amberkar, A., Awasarmol, P., Deshmukh, G., Dave, P., 2018. Speech Recognition using Recurrent Neural Networks. In: *Proceedings of the 2018 International Conference on Current Trends towards Converging Technologies*. <https://doi.org/10.1109/ICCTCT.2018.8551185>. ICCTCT 2018.
- Amini, S., Mohaghegh, S., 2019. Application of machine learning and artificial intelligence in proxy modeling for fluid flow in porous media. *Fluid.* <https://doi.org/10.3390/fluids4030126>.
- Arps, J.J., 1945. Analysis of decline curves. *Trans. AIME* 160. <https://doi.org/10.2118/945228-g>.
- Aydin, G., 2015. Regression models for forecasting global oil production. *Petrol. Sci. Technol.* 33 <https://doi.org/10.1080/10916466.2015.1101474>.
- Bao, A., Gildin, E., Huang, J., Coutinho, E.J.R., 2020. Data-driven end-to-end production prediction of oil reservoirs by EnKF-enhanced recurrent neural networks. In: *SPE Latin American and Caribbean Petroleum Engineering Conference Proceedings*. <https://doi.org/10.2118/199005-ms>.
- Barbosa, L.F.F.M., Nascimento, A., Mathias, M.H., de Carvalho, J.A., 2019. Machine learning methods applied to drilling rate of penetration prediction and optimization - a review. *J. Petrol. Sci. Eng.* 183, 106332. <https://doi.org/10.1016/j.petrol.2019.106332>.
- Buduma, N., Locascio, N., 2017. *Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms, Designing Next-Generation Machine Intelligence Algorithms*.
- Calvette, T., Gurwicz, A., Abreu, A.C., Pacheco, M.A.C., 2020. Forecasting smart well production via deep learning and data driven optimization. In: *Offshore Technology Conference Brasil 2019*. <https://doi.org/10.4043/29861-ms>. OTCB 2019.

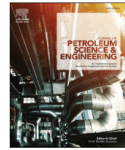
- Cao, Q., Banerjee, R., Gupta, S., Li, J., Zhou, W., Jeyachandra, B., 2016. Data driven production forecasting using machine learning. In: Society of Petroleum Engineers - SPE Argentina Exploration and Production of Unconventional Resources Symposium. <https://doi.org/10.2118/180984-ms>.
- Chen, G., Fu, K., Liang, Z., Sema, T., Li, C., Tontiwachwuthikul, P., Idem, R., 2014. The genetic algorithm based back propagation neural network for MMP prediction in CO<sub>2</sub>-EOR process. *Fuel*. <https://doi.org/10.1016/j.fuel.2014.02.034>.
- Chiroma, H., Abdulkareem, S., Abubakar, A.I., Herawan, T., 2014. Kernel functions for the support vector machine: comparing performances on crude oil price data. In: *Advances in Intelligent Systems and Computing*. [https://doi.org/10.1007/978-3-319-07692-8\\_26](https://doi.org/10.1007/978-3-319-07692-8_26).
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing*. Proceedings of the Conference. <https://doi.org/10.3115/v1/d14-1179>.
- Connor, J.T., Martin, R.D., Atlas, L.E., 1994. Recurrent neural networks and robust time series prediction. *IEEE Trans. Neural Netw.* 5 <https://doi.org/10.1109/72.279188>.
- Crnogorac, M., Tanasijević, M., Danilović, D., Maričić, V.K., Leković, B., 2020. Selection of artificial lift methods: a brief review and new model based on fuzzy logic. *Energies* 13. <https://doi.org/10.3390/en13071758>.
- Elmabrouk, S., Shirif, E., Mayorga, R., 2014. Artificial neural network modeling for the prediction of oil production. *Petrol. Sci. Technol.* 32 <https://doi.org/10.1080/10914666.2011.605093>.
- Equinor, 2020. *Volvo Data Village*. Azure Storage Explorer.
- Equinor, 2018. *Disclosing all Volvo data* [WWW document], 6.28.21. <https://www.equinor.com/en/news/14jun2018-disclosing-volve-data.html>.
- ExxonMobil, 2018. *Crude Oil Assay Volvo*.
- Ezugwu, A.E., Adeleke, O.J., Akinyelu, A.A., Viriri, S., 2020. A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. *Neural Comput. Appl.* 32 <https://doi.org/10.1007/s00521-019-04132-w>.
- Fanchi, J.R., Cooksey, M.J., Lehman, K.M., Smith, A., Fanchi, A.C., Fanchi, C.J., 2013. Probabilistic decline curve analysis of barnett, fayetteville, haynesville, and woodford gas shales. *J. Petrol. Sci. Eng.* 109 <https://doi.org/10.1016/j.petrol.2013.08.002>.
- Forrester, A.L.J., Sobester, A., Keane, A.J., 2008. *Engineering Design via Surrogate Modelling: a Practical Guide*. J. Wiley.
- Frausto-Solis, J., Chi-Chim, M., Sheremetov, L., 2015. Forecasting oil production time series with a population-based simulated annealing method. *Arabian J. Sci. Eng.* 40 <https://doi.org/10.1007/s13369-015-1587-z>.
- Gan, C., Cao, W.H., Liu, K.Z., Wu, M., Wang, F.W., Zhang, S.B., 2020. A new hybrid bat algorithm and its application to the ROP optimization in drilling processes. *IEEE Trans. Ind. Informatics* 16. <https://doi.org/10.1109/TII.2019.2943165>.
- Gharbi, R.B.C., Mansoori, G.A., 2005. An introduction to artificial intelligence applications in petroleum exploration and production. *J. Petrol. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2005.09.001>.
- Graves, A., Mohamed, A.R., Hinton, G., 2013. Speech recognition with deep recurrent neural networks. In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. <https://doi.org/10.1109/ICASSP.2013.6638947>.
- Guan, M., Cho, S., Petro, R., Zhang, W., Pasche, B., Topaloglu, U., 2019. Natural language processing and recurrent network models for identifying genomic mutation-associated cancer treatment change from patient progress notes. *JAMIA Open* 2. <https://doi.org/10.1093/jamiaopen/ooy061>.
- Han, B., Bian, X., 2018. A hybrid PSO-SVM-based model for determination of oil recovery factor in the low-permeability reservoir. *Petroleum*. <https://doi.org/10.1016/j.petm.2017.06.001>.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural Comput.* 9, 1735–1780.
- Hong, A., Bratvold, R.B., Lake, L.W., Ruiz Maraggi, L.M., 2019. Integrating model uncertainty in probabilistic decline-curve analysis for unconventional-oil-production forecasting. In: *SPE Reservoir Evaluation and Engineering*. <https://doi.org/10.2118/194503-PA>.
- Jochen, V.A., Spivey, J.P., 1996. Probabilistic reserves estimation using decline curve analysis with the bootstrap method. In: *Society of Petroleum Engineers - SPE Annual Technical Conference and Exhibition*. <https://doi.org/10.2523/36633-ms>.
- Kavzoglu, T., Colkesen, I., 2009. A kernel functions analysis for support vector machines for land cover classification. *Int. J. Appl. Earth Obs. Geoinf.* 11, 352–359. <https://doi.org/10.1016/j.jag.2009.06.002>.
- Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: *IEEE International Conference on Neural Networks - Conference Proceedings*. <https://doi.org/10.4018/jjmfmp.2015010104>.
- Khamis, M., Elhaj, M., Abdurrahman, A., 2020. Optimization of choke size for two-phase flow using artificial intelligence. *J. Pet. Explor. Prod. Technol.* 10 <https://doi.org/10.1007/s13202-019-0734-6>.
- Kingma, D.P., Ba, J.L., 2015. Adam: a method for stochastic optimization. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.
- Kondori, J., Miah, M.I., Zendeheboudi, S., Khan, F., Heagle, D., 2021. Hybrid connectionist models to assess recovery performance of low salinity water injection. *J. Petrol. Sci. Eng.* 197 <https://doi.org/10.1016/j.petrol.2020.107833>.
- Lee, K., Lim, J., Yoon, D., Jung, H., 2019. Prediction of shale-gas production at duvernay formation using deep-learning algorithm. *SPE J.* 24 <https://doi.org/10.2118/195698-PA>.
- Li, Y., Sun, R., Horne, R., 2019. Deep learning for well data history analysis. In: *Proceedings - SPE Annual Technical Conference and Exhibition*. <https://doi.org/10.2118/196011-ms>.
- Lin, Z., Liu, X., Lao, L., Liu, H., 2020. Prediction of two-phase flow patterns in upward inclined pipes via deep learning. *Energy* 210, 118541. <https://doi.org/10.1016/j.energy.2020.118541>.
- Liu, W., Liu, W.D., Gu, J., 2019. Petroleum production forecasting based on machine learning. In: *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/3373419.3373421>.
- Ma, X., Liu, Z., 2018. Predicting the oil production using the novel multivariate nonlinear model based on Arps decline model and kernel method. *Neural Comput. Appl.* 29 <https://doi.org/10.1007/s00521-016-2721-x>.
- Mamudu, A., Khan, F., Zendeheboudi, S., Adedigba, S., 2020. Dynamic risk assessment of reservoir production using data-driven probabilistic approach. *J. Petrol. Sci. Eng.* 184 <https://doi.org/10.1016/j.petrol.2019.106486>.
- Mohaghegh, S.D., 2020. Subsurface analytics: contribution of artificial intelligence and machine learning to reservoir engineering, reservoir modeling, and reservoir management. *Petrol. Explor. Dev.* [https://doi.org/10.1016/S1876-3804\(20\)60041-6](https://doi.org/10.1016/S1876-3804(20)60041-6).
- Mohaghegh, S.D., 2017. *Data-Driven Reservoir Modeling*. Society of Petroleum Engineers.
- Mohaghegh, S.D., 2011. Reservoir simulation and modeling based on artificial intelligence and data mining (AI&DM). *J. Nat. Gas Sci. Eng.* <https://doi.org/10.1016/j.jngse.2011.08.003>.
- Muojeke, S., Venkatesan, R., Khan, F., 2020. Supervised data-driven approach to early kick detection during drilling operation. *J. Petrol. Sci. Eng.* 192 <https://doi.org/10.1016/j.petrol.2020.107324>.
- Nait Amar, M., 2020. Modeling solubility of sulfur in pure hydrogen sulfide and sour gas mixtures using rigorous machine learning methods. *Int. J. Hydrogen Energy* 45, 33274–33287. <https://doi.org/10.1016/j.ijhydene.2020.09.145>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., 2020. Prediction of CO<sub>2</sub> diffusivity in brine using white-box machine learning. *J. Petrol. Sci. Eng.* <https://doi.org/10.1016/j.petrol.2020.107037>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Ng, C.S.W., Zeraibi, N., 2021. Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms. *J. Petrol. Sci. Eng.* 109038. <https://doi.org/10.1016/j.petrol.2021.109038>.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Zeraibi, N., 2020a. Predicting thermal conductivity of carbon dioxide using group of data-driven models. *J. Taiwan Inst. Chem. Eng.* <https://doi.org/10.1016/j.jtice.2020.08.001>.
- Nait Amar, M., Zeraibi, N., 2020. A combined support vector regression with firefly algorithm for prediction of bottom hole pressure. *SN Appl. Sci.* 2 <https://doi.org/10.1007/s42452-019-1835-z>.
- Nait Amar, M., Zeraibi, N., 2019. An efficient methodology for multi-objective optimization of water alternating CO<sub>2</sub> EOR process. *J. Taiwan Inst. Chem. Eng.* <https://doi.org/10.1016/j.jtice.2019.03.016>.
- Nait Amar, M., Zeraibi, N., 2018. Application of hybrid support vector regression artificial bee colony for prediction of MMP in CO<sub>2</sub>-EOR process. *Petroleum*. <https://doi.org/10.1016/j.petm.2018.08.001>.
- Nait Amar, M., Zeraibi, N., Jahanbani Ghahfarokhi, A., 2020b. Applying hybrid support vector regression and genetic algorithm to water alternating CO<sub>2</sub> gas EOR. *Greenh. Gases Sci. Technol.* <https://doi.org/10.1002/ghg.1982>.
- Nait Amar, M., Zeraibi, N., Redouane, K., 2018. Bottom hole pressure estimation using hybridization neural networks and grey wolves optimization. *Petroleum*. <https://doi.org/10.1016/j.petm.2018.03.013>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2021a. Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization. *J. Pet. Explor. Prod. Technol.* <https://doi.org/10.1007/s13202-021-01199-x>.
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., Torsater, O., 2021b. Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application. *Nat. Resour. Res.* 30, 2431–2462. <https://doi.org/10.1007/s11053-021-09844-2>.
- Olukoga, T.A., Feng, Y., 2021. Practical machine-learning applications in well-drilling operations. *SPE Drill. Complet.* <https://doi.org/10.2118/205480-pa>.
- Ozbayoglu, E., Ozbayoglu, M., Ozdilli, B.G., Erge, O., 2021. Optimization of flow rate and pipe rotation speed considering effective cuttings transport using data-driven models. *Energies* 14. <https://doi.org/10.3390/en14051484>.
- Panja, P., Velasco, R., Pathak, M., Deo, M., 2018. Application of artificial intelligence to forecast hydrocarbon production from shales. *Petroleum*. <https://doi.org/10.1016/j.petm.2017.11.003>.
- Qu, H., Zhang, Y., 2016. A new kernel of support vector regression for forecasting high-frequency stock returns, 2016 *Math. Probl. Eng.*. <https://doi.org/10.1155/2016/4907654>.
- Senthilkumar, M., 2010. Use of artificial neural networks (ANNs) in colour measurement. In: *Colour Measurement: Principles, Advances and Industrial Applications*. <https://doi.org/10.1533/9780857909195.1.125>.
- Shawe-Taylor, J., Cristianini, N., 2004. *Kernel Methods for Pattern Analysis*. Kernel Methods for Pattern Analysis. <https://doi.org/10.1017/cbo9780511809682>.
- Shi, Y., Eberhart, R., 1998. Modified particle swarm optimizer. In: *Proceedings of the IEEE Conference on Evolutionary Computation*. *ICEC*. <https://doi.org/10.1109/icec.1998.699146>.
- Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: *Advances in Neural Information Processing Systems*.

- Syed, F.I., Alshamsi, M., Dahaghi, A.K., Neghabhan, S., 2020. Artificial lift system optimization using machine learning applications. *Petroleum*. <https://doi.org/10.1016/j.petlm.2020.08.003>.
- Vapnik, V.N., 1995. The Nature of Statistical Learning Theory, the Nature of Statistical Learning Theory. <https://doi.org/10.1007/978-1-4757-2440-0>.
- Zanjani, M.S., Salam, M.A., Kandara, O., 2020. Data-driven hydrocarbon production forecasting using machine learning techniques. *Int. J. Comput. Sci. Inf. Secur.* 18.
- Zhan, C., Sankaran, S., LeMoine, V., Graybill, J., Mey, D.O.S., 2020. Application of machine learning for production forecasting for unconventional resources. In: SPE/AAPG/SEG Unconventional Resources Technology Conference 2020. <https://doi.org/10.15530/urtec-2019-47>. URTEC 2020.
- Zhang, J.S., Xiao, X.C., 2000. Predicting chaotic time series using recurrent neural network. *Chin. Phys. Lett.* 17 <https://doi.org/10.1088/0256-307X/17/2/004>.
- Zhong, Z., Sun, A.Y., Wang, Y., Ren, B., 2020. Predicting field production rates for waterflooding using a machine learning-based proxy model. *J. Petrol. Sci. Eng.* 194, 107574. <https://doi.org/10.1016/j.petrol.2020.107574>.

## **Paper 8**

### ***Optimizing initiation time of waterflooding under geological uncertainties with Value of Information: Application of simulation-regression approach***

Cuthbert Shang Wui Ng, Ashkan Jahanbani Ghahfarokhi



# Optimizing initiation time of waterflooding under geological uncertainties with Value of Information: Application of simulation-regression approach

Cuthbert Shang Wui Ng<sup>\*</sup>, Ashkan Jahanbani Ghahfarokhi

Department of Geoscience and Petroleum, Norwegian University of Science and Technology, Trondheim, Norway

## ARTICLE INFO

### Keywords:

Decision analysis  
Waterflooding  
Value of information  
Simulation-regression approach  
Machine learning  
Optimization under uncertainty

## ABSTRACT

Reservoir Management (RM) is an example of sequential decision problems in the oil and gas industry. Therefore, implementing Decision Analysis (DA) tool to systematically resolve such problems has been a common practice. The value of Information (VOI) framework acts as one of these tools that helps reservoir engineers to manage RM problems. Regarding this, the Least-Squares Monte Carlo (LSM) algorithm, which is one of the simulation-regression approaches, has been employed to estimate VOI for a better quality of decision-making (DM). Integration of the LSM algorithm in RM is coined as "Sequential Reservoir Decision-Making" (SRDM). This approximate method is essential to resolve a sequential decision problem with high dimensionality caused by many possible outcomes of uncertainties. This challenge is generally known as the "curse of dimensionality". In this work, a modified LSM algorithm has been applied under the SRDM paradigm to optimize the waterflooding initiation time considering geological uncertainties. The modification considers the effects of information acquired previously and at the current decision time before a decision is made. The reservoir model used in this work is the OLYMPUS benchmark model. Apart from utilizing Linear Regression (LR) in the LSM algorithm, the use of two machine learning (ML) techniques, viz. Gaussian Process Regression (GPR) and Support Vector Regression (SVR), have been illustrated to estimate the VOI. Based on the results, LR, GPR, and SVR correspondingly estimate the VOI as 11.52 million USD, 11.17 million USD, and 12.46 million USD. This means that SVR displays an improvement of 8.18% compared to the VOI assessed by LR. This shows its good applicability in VOI estimation and it can be concluded that integrating ML techniques into the SRDM paradigm demonstrates high potential for RM applications.

## 1. Introduction

Decision Analysis (DA) is one of the knowledge domains that has been ubiquitous in different aspects of engineering studies. According to Howard (1980), DA can be understood as a systematic methodology that transforms an opaque (hard to understand) decision problem into a transparent (easy to perceive) one via a series of transparent steps. Concerning this, Value of Information (VOI) is one of the most prevalent decision-making (DM) tools. VOI is the approximation of additional value induced when information is brought to a decision problem (Howard, 1966). Despite having such a lucid definition of DA, many engineers are still subject to misconception. They tend to include as many details as possible when they are developing their DM tool, including VOI. This might not be a good practice because only important or pertinent factors should be considered in DM models.

Furthermore, it is enlightening for engineers to realize that the VOI

technique is formulated to evaluate if the improvement in DM by acquiring the information is worth the cost required to gain it. In another word, the VOI analysis is an a priori analysis that quantitatively assesses the benefits of obtaining additional information before the data is gathered and a decision is made (Hong et al., 2018). As Bratvold and Begg (2010) have counseled, for an information-gathering activity to be worthwhile, its VOI should exceed the cost of the activity itself. Also, it must have the ability to change the decision maker's beliefs about uncertainty and the decisions made otherwise. Hence, engineers ought to be cognizant that VOI does not in fact "reduce uncertainty", but it facilitates the adjustment of the decisions concerning underlying uncertainty. Thus, VOI is often coupled with uncertainty and DM, in which information cannot be valued without a specific decision context (Bratvold et al., 2009; Hong et al., 2018).

The use of the VOI methodology has been growing in the oil and gas industry, especially in the aspects of reservoir management (RM), for the

<sup>\*</sup> Corresponding author.

E-mail address: [cuthbert.s.w.ng@ntnu.no](mailto:cuthbert.s.w.ng@ntnu.no) (C.S.W. Ng).

<https://doi.org/10.1016/j.petrol.2022.111166>

Received 21 April 2022; Received in revised form 7 October 2022; Accepted 24 October 2022

Available online 28 October 2022

0920-4105/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

past decade. RM refers to the employment of available technology, labor resources, and financial assets to maximize economic returns through hydrocarbon production from a reservoir (Satter et al., 1998; Wiggins and Startzman, 1998). RM generally entails a series of operations and decisions, stemming from the initial phase of field discovery to the final phase of field abandonment. Furthermore, most of the DM problems are considered sequential and involve a lot of uncertainties. This implies that information is continuously acquired to enhance the quality of DM. Therefore, the VOI framework aptly applies in the resolution of such sequential DM problems. Nevertheless, the real-world challenge in this context is to determine the analytical solution of VOI.

One of the methods to approximately compute the VOI is by applying a decision tree (DT), which is a part of dynamic programming. DT is efficient for visualization and communication of the frame of a sequential decision. Fundamentally, a sequential decision problem can be represented as a DT and solved by rolling back the DT itself. For a more comprehensive implementation of DT, refer to these books (Bratvold and Begg, 2010; Howard and Abbas, 2016). Unfortunately, the DT method will encounter the “curse of dimensionality” if it is used to solve a more sophisticated decision problem (Powell, 2011). In this aspect, three main sources of the “curse of dimensionality” comprise the number of possible outcomes (or uncertainties), the number of decision points (time where a decision needs to be made), and the number of alternatives at each decision point (Powell, 2011). Least-Squares Monte Carlo (LSM) algorithm that was developed by Longstaff and Schwartz (2001) can replace DT in resolving a more complex problem, but it is only efficient to handle sequential decision problems with many uncertain quantities and limited number of alternatives. The increase in the number of alternatives and decision points causes an exponential increase in computational effort and thus, the “curse of dimensionality” arises.

LSM is placed under the umbrella of the simulation-regression approach in terms of the determination of VOI. As the name of the approach implies, it can be perceived that there are two main frameworks, namely simulation and regression analysis. Monte Carlo simulation (MCS) is one of the standard practices to capture the effect of uncertainties on the production profile. In reservoir engineering, uncertainties generally pertain to the geological properties of a reservoir. Hence, numerical reservoir simulation (NRS) can leverage MCS to perform forward modeling to generate production data under uncertainties for any RM decision problem. Thereafter, regression analysis is conducted in the form of backward calculation to estimate the VOI. The details of this analysis will follow later. Linear Regression (LR) has been used to perform the regression analysis. However, as the research domain has been developing, modifications or improvements to the LSM algorithm<sup>1</sup> have been done to resolve different sequential decision problems. More detailed descriptions will follow.

The application of VOI analysis in the oil and gas industry has been overviewed by Bratvold et al. (2009). In addition, several articles have illustrated the implementation of the simulation-regression approach under the VOI paradigm. Willigers and Bratvold (2009) performed the valuation of real options in an oil and gas project through the implementation of LSM. Stemming from this work, LSM was further employed for the valuation of swing contracts in the field of natural gas and electricity (Willigers et al., 2011). Alkhatib et al. (2013) discussed the use of LSM to yield an optimal policy of surfactant flooding in both homogeneous and heterogeneous reservoirs considering geological uncertainty. Hong et al. (2019) further extended the use of LSM by coupling this algorithm with a proxy model, known as the Two-Factor Production Model that was comprehensively discussed (Parra Sanchez,

2010) to evaluate the optimal switch time of waterflooding. The LSM algorithm was modified to integrate the dependency of both currently and previously measured data. Therefore, the algorithm was named modified LSM. Based upon the formulation of modified LSM, Tadjer et al. (2021a) evaluated the VOI under polymer flooding. Besides LR, they successfully utilize machine learning (ML) techniques (nonlinear regression), including neural network regressor and Tree-based Pipeline Optimization Tool (TPOT), which was proposed by Olson et al. (2016), as an alternative. Dutta et al. (2019a) also displayed how Principal Component Regression and Partial Least-Squares Regression could be implemented as a nonlinear regression approach to assess VOI for sequential spatial data collection in subsurface energy application. There are also other papers (Dutta et al., 2019b; Eidsvik et al., 2017) expounding on the application of simulation-regression approaches for the estimation of VOI. Furthermore, these nonlinear simulation-regression approaches have been illustrated in other interesting tasks with the emphasized application in Carbon, Capture, and Storage (CCS). Tadjer et al. (2021b) implemented TPOT as the regression technique to determine the VOI of performing carbon storage in Utsira formation. Also, Anyosa et al. (2021) applied some ML-based regression methods, including k-Nearest Neighbors, Random Forest, and Convolution Neural Networks, to do VOI analysis to evaluate the value of seismic monitoring of CO<sub>2</sub> storage at Smeaheia site.

The work that is conducted here is inspired by a previous work (Ng, 2019). In this paper, the modified LSM algorithm is implemented to determine the optimal initiation time of waterflooding in the OLYMPUS reservoir model under geological uncertainties. This initiation is decided based on the acquisition of information from both oil and water production data. Moreover, 50 different geological realizations have been employed to capture the uncertainties in the DM process. The pertinent details will follow in later sections. Apart from the conventional LR approach for regression analysis, other nonlinear regression techniques are also utilized. Examples of the nonlinear techniques (alternatively termed ML-based methods) chosen in this work consist of Gaussian Process Regression and Support Vector Regression. The corresponding computed VOI and the decisions to be made for each geological realization by incorporating different regression methods are then analyzed and compared for further discussion.

After this introduction, the paper is structured by having the following sections. Section 2 provides the theoretical framework of VOI in which the mathematical implementation of VOI estimation is presented. The background of the decision problem and the details of the OLYMPUS reservoir model as well as the economic model employed are thereafter briefed under Section 3. Then, Section 4 discusses the mechanism of the modified LSM along with its integration into “Sequential Reservoir Decision-Making” (SRDM). Section 5 explains the other nonlinear regression ML-based methods used in this work. Thereafter, Section 6 highlights the results and relevant discussions. Some concluding remarks are summarized in Section 7.

## 2. Value of Information (VOI)

In any information acquisition activity, VOI relies upon two important uncertainties, namely distinction of interest and observable distinction (Bratvold et al., 2009). The distinction of interest is not observable and aimed to be learned. Therefore, any information obtained in the form of any test result is considered as the observable distinction that helps the decision makers to perceive better the distinction of interest. In the context of RM, specifically production optimization, the production data gained until time  $t$  (when the decision is to be made) is treated as an observable distinction. It is computationally challenging to analytically represent the distribution of observable distinction due to its high dimension. Therefore, the use of

<sup>1</sup> LSM is precisely a combined application of MCS and LR. The use of sampling techniques other than MCS for the generation of different realizations of simulation and other data-driven methods as substitutes for LR is better termed as simulation-regression method.

Monte Carlo sampling plays a role to remediate this issue. Based on the assumption of risk neutrality<sup>2</sup>, VOI can be mathematically represented as follows:

$$\gamma = \left[ \begin{array}{c} \text{Value of Information} \\ \text{Expected Value with} \\ \text{Information} \end{array} \right] - \left[ \begin{array}{c} \text{Expected Value without} \\ \text{Information} \end{array} \right] \quad (1)$$

The estimated  $\gamma$  under a decision problem can be negative. Negative  $\gamma$  denotes that it is not economically feasible to acquire information. Hence, the lower limit of VOI is always treated as zero. Besides that, for Expected Value without Information (EVWOI), the corresponding decision without information (DWOI) is the alternative that optimizes the EV over all the realizations. For Expected Value with Information (EVWI), the respective optimal decision is Decision with Information (DWI). The mathematical formulations of EVWOI and EVWI are respectively displayed as:

$$\text{EVWOI} = \max_{\mathbf{a} \in \mathbf{A}} \left[ \int \mu(\mathbf{x}, \mathbf{a}) p(\mathbf{x}) d\mathbf{x} \right] \approx \max_{\mathbf{a} \in \mathbf{A}} \left( \frac{1}{N_r} \sum_{r=1}^{N_r} \mu(\mathbf{x}^r, \mathbf{a}) \right) \quad (2)$$

where  $\mathbf{a}_{\text{DWOI}}^{\text{optimal}} = \arg \max_{\mathbf{a} \in \mathbf{A}} \left( \frac{1}{N_r} \sum_{r=1}^{N_r} \mu(\mathbf{x}^r, \mathbf{a}) \right)$

$$\text{EVWI} = \int \max_{\mathbf{a} \in \mathbf{A}} [E(\mu(\mathbf{x}, \mathbf{a}) | \mathbf{y})] p(\mathbf{y}) d\mathbf{y} \approx \frac{1}{N_r} \sum_{r=1}^{N_r} \max_{\mathbf{a} \in \mathbf{A}} [E[(\mu(\mathbf{x}, \mathbf{a}) | \mathbf{y}^r)]] \quad (3)$$

where  $\mathbf{a}_{\text{DWI}}^{\text{optimal}} = \arg \frac{1}{N_r} \sum_{r=1}^{N_r} \max_{\mathbf{a} \in \mathbf{A}} [E[(\mu(\mathbf{x}, \mathbf{a}) | \mathbf{y}^r)]]$

According to the formulations above,  $p(\mathbf{x})$  is a prior probability distribution of distinction of interest that is represented as an ensemble of  $\mathbf{x} = \{x^1, x^2, \dots, x^{N_x}\}$ .  $\mathbf{a}$  is used to denote the available alternatives, which are from a set of possible alternatives,  $\mathbf{A}$ . Furthermore,  $\mu(\mathbf{x}^r, \mathbf{a})$  is the function that yields the prospect values corresponding to a specific realization and selected alternatives.  $\mathbf{y}$  is a collection of observable data, in which  $\mathbf{y} = \{y^1, y^2, \dots, y^{N_y}\}$  and  $p(\mathbf{y})$  is the marginal probability distribution. For each realization of  $\mathbf{x}^r$ , forward modeling can be done to determine  $\mathbf{y}^r$ .

VOI can also be understood as VOII (Value of Imperfect Information) because it is very challenging to acquire perfect information regarding a DM context in real life. Information is perfect if it is always true. Equation (3) portrays the estimation of EVWII. Thus, in RM, perfect information is the information that reveals the true properties of a reservoir and the impacts of the recovery mechanism. Besides that, the value of perfect information (VOPI), which is the difference between Expected Value with Perfect Information (EVWPI) and EVWOI, acts as the upper limit of VOI. In this context, the decision with perfect information (DWPI) corresponds to an alternative that optimizes the relevant objective function for each realization based upon prior distributions. Finding the average of such values of the objective function over all the realizations yields EVWPI as follows:

$$\text{EVWPI} = \int \max_{\mathbf{a} \in \mathbf{A}} [\mu(\mathbf{x}, \mathbf{a})] p(\mathbf{x}) d\mathbf{x} \approx \frac{1}{N_r} \sum_{r=1}^{N_r} \max_{\mathbf{a} \in \mathbf{A}} [\mu(\mathbf{x}^r, \mathbf{a})] \quad (4)$$

where  $\mathbf{a}_{\text{DWPI}}^{\text{optimal}} = \arg \frac{1}{N_r} \sum_{r=1}^{N_r} \max_{\mathbf{a} \in \mathbf{A}} [\mu(\mathbf{x}^r, \mathbf{a})]$

<sup>2</sup> Risk neutrality is a risk attitude apart from risk-averse and risk-seeking. Please refer to this literature (Hillson and Murray-Webster, 2017) for more explanation of risk attitudes. In a simpler term, a risk-neutral decision maker applies the Expected Value (EV) in the process of DM. This implies that the decision maker will have the same preference over two alternatives with the same EV.

### 3. Background of the decision problem and models

#### 3.1. Problem setting

The decision problem discussed here is a part of RM and similar problems have been briefed in several pieces of literature (Hong et al., 2019; Ng, 2019; Tadjer et al., 2021a). Fundamentally, this decision problem involves the optimization of the initiation time of waterflooding in a 3D reservoir model (the benchmark model OLYMPUS). In the framework of this sequential decision problem, the production period of the OLYMPUS model is assumed to be 10 years. Thereafter, each year, a decision is needed if it is better to switch from primary recovery to waterflooding (in other term, to start waterflooding) or continue only with primary recovery. The termination time of production (under both primary recovery and waterflooding) is then optimized too. Concerning these, the initiation of waterflooding and termination of production can only occur once.

#### 3.2. Reservoir model

The reservoir model implemented in this paper is the OLYMPUS model and simulation is performed by using the Eclipse 100 software (Schlumberger, 2019). This benchmark case, a synthetic field model developed by Fonseca et al. (2020), mainly consists of an oil-water system and has an approximate dimension of  $9 \times 3$  km. The geological properties of this model have a typical resemblance to those of the North Sea field with Brent-type oil. The model has a thickness of 50 m with two different zones separated by an impermeable shale layer. In addition, the model is made up of 341,728 grid blocks in which the average dimension of each block is  $50 \times 50 \times 3$  m. However, the total number of active grid blocks are 192,750.

Moreover, to resolve the sequential decision problem as explained earlier, an ensemble of 50 realizations is used to capture the effect of uncertainty in this context. The uncertain variables consist of facies, porosity, permeability, net-to-gross ratio, initial water saturation, and transmissibility across the faults. For further details of the geological and petrophysical aspects of OLYMPUS, please peruse Fonseca et al. (2020). About the well configuration in this model, there are 7 injectors and 11 producers. Each of the injectors is controlled by keeping the maximum well rate of 2000  $\text{sm}^3/\text{day}$  with bottomhole pressure target of 250 bars. Besides that, each of the producers is controlled by having the maximum bottomhole pressure at 150 bars. With these sets of control, the initiation time of all the injectors is optimized by applying the VOI analysis. The architecture (permeability in x-direction, PERMX in the unit of mD) of one of the realizations of the OLYMPUS model used here is presented in Fig. 1.

#### 3.3. Economic model

The economic model used in this work is represented by net present value (NPV), which is illustrated as follows:

$$\text{NPV} = \sum_{i=0}^{N_t} \frac{\Delta t_i (P_o q_o^i - P_w q_w^i - P_{wi} q_{wi}^i) - \text{CAPEX}_i}{(1 + \text{interest rate})^i} \quad (5)$$

Based on the NPV equation above, P indicates the price in which the subscripts o, w, and wi respectively mean oil, water produced, and water injected.  $q^i$  indicates the production (or injection) rates at timestep i.  $\Delta t_i$  is the difference between timesteps i and i-1. The timestep is on yearly basis. CAPEX denotes capital expenditure. In addition, the values of economic variables applied in this paper are tabulated in Table 1. Based on Table 1, it can be noted that in the case of waterflooding, there are three types of CAPEX, such as capital expenditure for having only primary recovery, additional capital expenditure for starting waterflooding after primary recovery, and capital expenditure for starting waterflooding without having primary recovery.

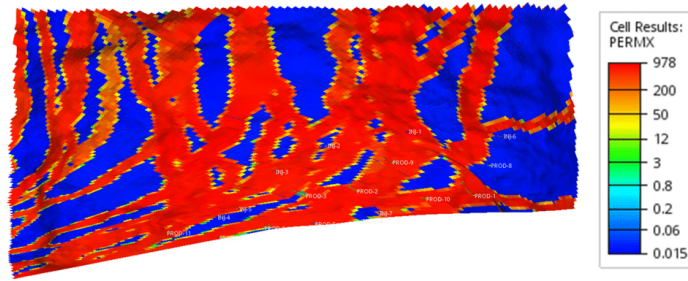


Fig. 1. Top view of Realization 5 of OLYMPUS.

Table 1

Values of economic variables used in this work.

Variables	Values	Units
Oil Price	408.85	USD/m <sup>3</sup>
Water Production Price	50.32	
Water Injection Price	50.32	
CAPEX (Primary Recovery)	40	million USD
Additional CAPEX	30	
CAPEX (Waterflooding)	85	
Interest Rate	6%	per year

#### 4. Simulation-regression paradigm

In general, EVWII under a specific decision context can be estimated by using the simulation-regression approach. This approach leverages MCS and regression analysis to determine the conditional EV upon given data. By implementing MCS, it is possible to alleviate the curse of dimensionality induced by the number of uncertain outcomes. Upon the completion of MCS, the backward induction is conducted with the aid of regression analysis to approximate the conditional EV of each alternative. The following are the details of the mechanisms corresponding to MCS and regression analysis (backward induction):

Monte Carlo simulation (MCS):

- 1) MCS is implemented to generate several realizations with different state variables (for instance, permeability, porosity, net-to-gross ratio, and so on) that can be understood as  $x^f$ .
- 2) Forward modeling of these realizations is done to create data (corresponding to oil/water production rates as well as water injection rates) to which noise will be added by using the statistics of the measurement errors. The noise is modeled by using zero mean and standard deviation of 0.15 here.
- 3) NPV of each decision alternative  $\mu(x^f, a)$  is computed and EVWOI is thereafter determined by using equation (2).

Regression Analysis:

- 1) Beginning from the last decision point in time, NPVs are regressed on the generated data  $y^f$  (considering only  $q_o$  and  $q_w$ ) to determine the expected NPV (ENPV) of decision alternative  $a$  being conditional on the data,  $y^f$ . This procedure corresponds to the calculation of the term  $E[(\mu(x, a)|y^f)]$  as demonstrated on the right-hand side of equation (3).
- 2) The step above is repeated for every decision alternative.
- 3) The best decision alternative,  $a_{\text{optimal}}^{\text{optimal}}$  is made by selecting the decision alternative that yields the maximum conditional ENPV given the known data for every realization, and EVWII is computed based on equation (3). VOI is then calculated by using equation (1).

#### 4.1. Sequential Reservoir Decision-Making (SRDM)

Regarding the details of regression techniques used, if the technique employed is the least-squares method (or LR), then it is termed the LSM algorithm as mentioned before. In this aspect, Hong et al. (2019) have discussed the application of LSM in the resolution of the initiation time of Improved Oil Recovery as an epitome of SRDM. On closer scrutiny, as LSM is employed for SRDM, the termination time given a specific initiation time of waterflooding must be first determined. Due to its nature of backward induction, the algorithm commences in Year 10.

In (the beginning of) Year 10, the optimal termination time is found by assuming that waterflooding has been started at this time. Therefore, there are only two available options (or decision alternatives), which are “terminate in Year 10” and “continue with waterflooding in Year 10”. Therefore, the NPVs corresponding to these two options are regressed on the production data ranging from Year 1 to Year 9 given that waterflooding has started in Year 10. Thereafter, between these two options, that of higher estimated NPV is the optimal option in this case for each realization. Averaging the NPVs of these options over all realizations results in the ENPV of waterflooding initiation in (the beginning of) Year 10.

When the time rolls back to (the beginning of) Year 9, given waterflooding started the same year, there are three available options, namely “terminate in Year 9”, “continue with waterflooding in Year 9 but terminate in Year 10”, and “continue with waterflooding in Year 9”. The last option corresponds to those determined in the previous step. Hence, the NPVs of the last two options are first regressed on the production data from Year 1 to Year 9 (these two options are regressed first due to the availability of data from Year 1 to Year 9). Based upon these estimated NPVs, the two options are compared and the respective optimal NPV is recorded for each realization. Then, the chosen option for every realization is compared with the option of “terminate in Year 9” through another regression analysis using the production data ranging from Year 1 to Year 8. This whole step will determine the ENPV of waterflooding initiation in (the beginning of) Year 9. The same logic is conducted every previous year. This procedure assists us to select the optimal stopping time for each year by assuming that waterflooding is initiated in that particular year.

Upon completing this procedure, the optimal option of waterflooding initiation considering termination has been determined. Then, these options are compared with the option of “continuing only with primary recovery”. In this case, in (the beginning of) Year 10, the NPVs of “initiating waterflooding in Year 10 with its respective optimal termination time” and “continuing only with primary recovery” are regressed on the production data from Year 1 to Year 9. The higher approximated NPV is then used to select the optimal option for each realization.

Then, in (the beginning of) Year 9, the NPVs of these optimal options in Year 10 and the option of “continuing only with primary recovery” in Year 9 are again regressed on the production data from Year 1 to Year 8.



The same workflow is implemented until the time becomes Year 1. Other techniques can also be used in this context. Two other approaches, namely Gaussian Process Regression (GPR) and Support Vector Regression (SVR) are chosen in addition to Linear Regression. GPR and SVR will be briefed in the following section.

## 5. Machine learning techniques

### 5.1. Gaussian Process Regression (GPR)

GPR is a non-parametric ML approach that can be employed to perform data-driven modeling based on the Bayesian principle and Gaussian process. In a more technical sense, the Gaussian process (GP) can be perceived as a collection of random variables which possesses a multivariate joint distribution. In GPR, there is a function that can yield the output at certain inputs, in which the Gaussian noise with the normal distribution is included. GP acts as a distribution over functions and is defined by a mean function and covariance function. The covariance function (also known as kernel function) captures the dependence between different values of the function at their respective inputs. In this work, the employed kernel function is a squared exponential function. By having defined the mean and covariance functions, GP can be employed to retrieve a priori function values and posterior function values that are conditioned on the observed variables.

When it comes to the prediction of function values at new inputs, the joint distribution of the observed values and function values at these new points can be developed. Thereafter, GPR can be used to derive the “updated” posterior distribution by conditioning on these observed values. By doing so, the respective mean function can be determined by the posterior distribution and is treated as the prediction of regression. So, it is important to understand that GPR does not result in a deterministic model that best fits the data provided. However, it yields the predicted output by embracing the probability. For more comprehensive details of GPR, please counsel the following literature (Liu et al., 2020; Rasmussen and Williams, 2018). The modeling of GPR in this work is performed with the aid of Statistics and Machine Learning Toolbox in MATLAB R2021b (MathWorks, 2022). The hyperparameters are set at default values apart from the initial value for the noise standard deviation of GP which is set at 4.

### 5.2. Support Vector Regression (SVR)

SVR is another popular example of supervised learning techniques that is applied to approximate the relationship between inputs and the respective outputs with the weight vector and the bias term as the parameters. In general, SVR involves the mapping of the input space vector into feature space with higher dimensionality. This is to transform the initial non-linear problem into a more conveniently solvable linear regression function. Then, the regularized risk function can be minimized to estimate the weight vector and the bias term. To achieve this, the constrained optimization problem is established by introducing the non-negative slack variables (Forrester et al., 2008). This optimization function can be transformed into dual space by using Lagrange multipliers for the resolution of the constrained optimization problem (Shawe-Taylor and Cristianini, 2004). In this paper, the Gaussian function is used as the kernel function. Regarding the development of the SVR model, it is done by applying Statistics and Machine Learning Toolbox in MATLAB R2021b (MathWorks, 2022). The default hyperparameters are used, but standardization of data is implemented.

## 6. Results and discussion

Under the problem setting discussed, the DWOI consists of 2 years of primary recovery and then 8 years of waterflooding. This yields a field production of 10 years. DWOI corresponds to the alternative with the highest ENPV over all realizations. The respective EVWPI is 1479.06

million USD. This denotes that without acquiring any production data, the net profit considering all the realizations is 1479.06 million USD if there are 2 years of primary recovery followed by 8 years of water injection. Furthermore, DWOI corresponds to the alternative with the highest NPV for each realization. Averaging these NPVs results in EVWPI and it is calculated to be 1625.54 million USD. Then, VOPI is 146.47 million USD. This implies that if the cost of the information-gathering activity exceeds 146.47 million USD, this activity needs to be abandoned.

Also, the normalized frequency distributions (NFD) and the normalized cumulative frequency distributions (NCFD) of DWOI for the lifetime of primary recovery, those of secondary recovery (waterflooding), and a total lifetime of production are illustrated correspondingly in Fig. 2. Based on Fig. 2a, the NFD displays that about 30% of 50 geological realizations result in 1 and 2 years of primary recovery, which sums up to 60% of total realizations. On scrutiny, the NCFD portrays that 88% of the realizations recommend the lifetime of primary recovery to be equal to or less than 2 years. Thus, a considerably short lifetime of primary recovery is essential to achieve EVWPI. Additionally, about 36% of realizations yield 8 years of waterflooding as shown by the NFD plot in Fig. 2b. In this context, 70% of all the realizations propose water injection for at most 8 years. Around 56% of realizations proceed with a total of 10 years of production as demonstrated in Fig. 2c. In the case of NCFD, 44% of the realizations propose having a total lifetime of at most 9 years. It means that more than 50% of the realizations suggest 10 years of total lifetime.

Regarding the DWII of waterflooding for each realization, it has been previously expounded that 3 different techniques are employed to perform the backward induction (regression analysis) in the modified LSM algorithm to provide an SRDM solution. Out of these 3 techniques, GPR and SVR are generally considered ML-based. The regression analysis in the modified LSM algorithm can be treated as an example of a training process for ML techniques. Therefore, to elude the issue of overfitting during the training process, 5-fold cross-validation is used during regression analysis. Fig. 3 portrays the plots of observed NPV against approximated NPV for each alternative during regression analysis at each decision point in time with LR, GPR, and SVR.

As illustrated in Fig. 3, the Pearson Correlation Coefficients,  $\rho$  respectively obtained for LR, GPR, and SVR are 0.9634, 0.9772, and 0.9296. To further assess the quality of proximity, coefficient of determination,  $R^2$  for LR, GPR, and SVR are correspondingly computed to be 0.9281, 0.9549, and 0.8642. According to these results, it is noticeable that GPR has outperformed both LR and SVR in terms of NPV approximation. On closer scrutiny, LR in the modified LSM algorithm yields an EVWPI of 1490.58 million USD. GPR and SVR resulted in the corresponding EVWPIs of 1490.23 million USD and 1491.52 million USD. Moreover, VOIs of LR, GPR, and SVR are respectively 11.52 million USD, 11.17 million USD, and 12.46 million USD. Despite having the highest  $R^2$ , GPR results in the lowest VOI in this case. This shows that the higher accuracy of the approximated NPV (vs. observed NPV) for each alternative is unable to avoid the suboptimality of alternatives at certain paths (or realizations). Besides that, as compared with the case of LR, SVR enhances the VOI estimation by 8.18%. This reflects a good potential for VOI enhancement by implementing a nonlinear method under the framework of LSM.

Besides that, it can be deduced from these results that EVWPIs are higher than EVWOIs and this implies that it is worthwhile to include the effect of future information in decision-making. In this case, applying LR in the modified LSM algorithm would enhance the ENPV by 0.78%. An increase by 0.76% and 0.84% is also attained through the implementation of GPR and SVR, respectively. This practically illustrates the benefit of acquiring additional data in lieu of abiding by the initial plan as suggested by DWOI. Moreover, the NFD and NCFD of DWII for the lifetime of primary recovery, those of secondary recovery (waterflooding), and the total lifetime of production are illustrated correspondingly in Fig. 4 for LR, Fig. 5 for GPR, and Fig. 6 for SVR. Based on

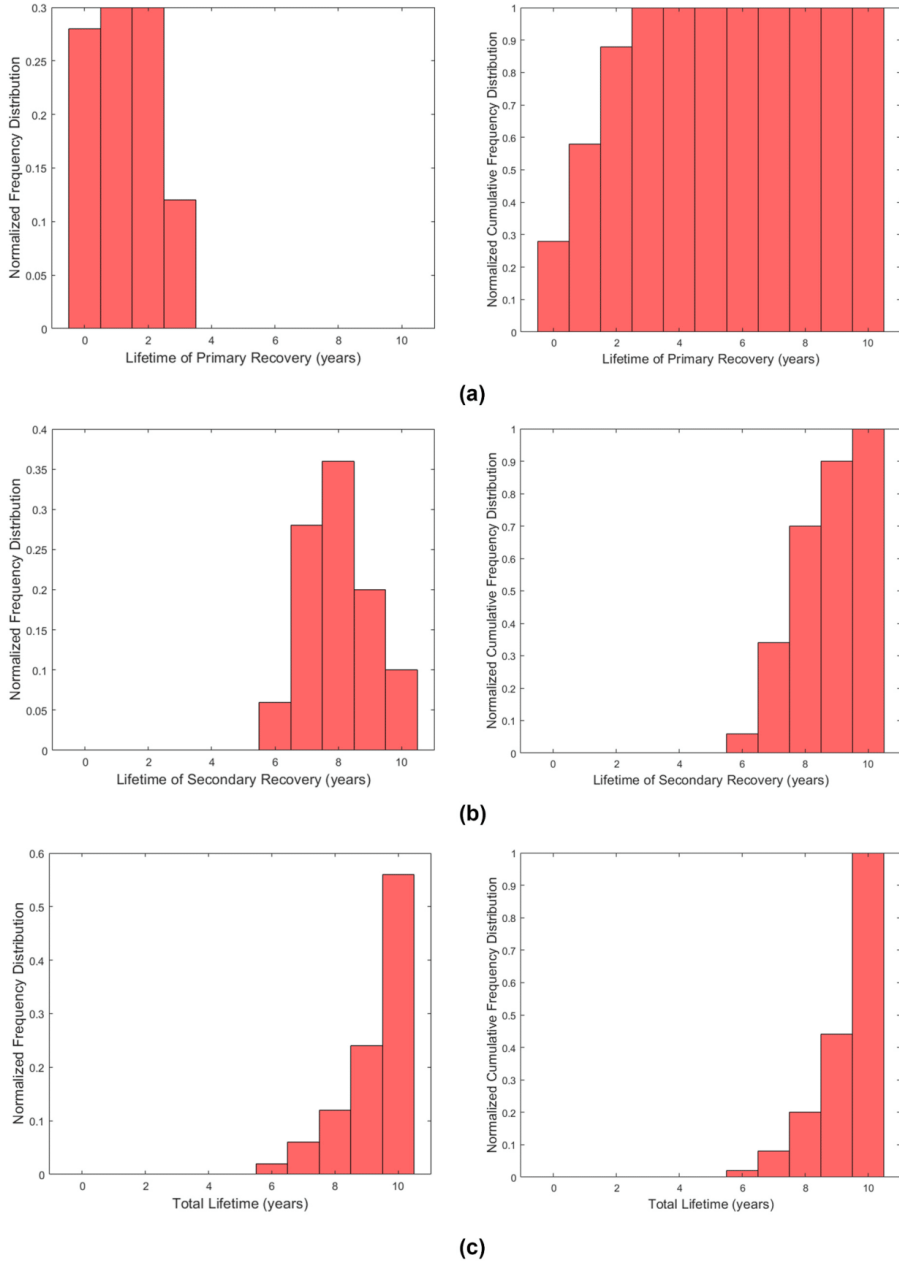


Fig. 2. Distribution of DWPI for: (a) Lifetime of Primary Recovery. (b) Lifetime of Secondary Recovery (Waterflooding). (c) Total Lifetime.

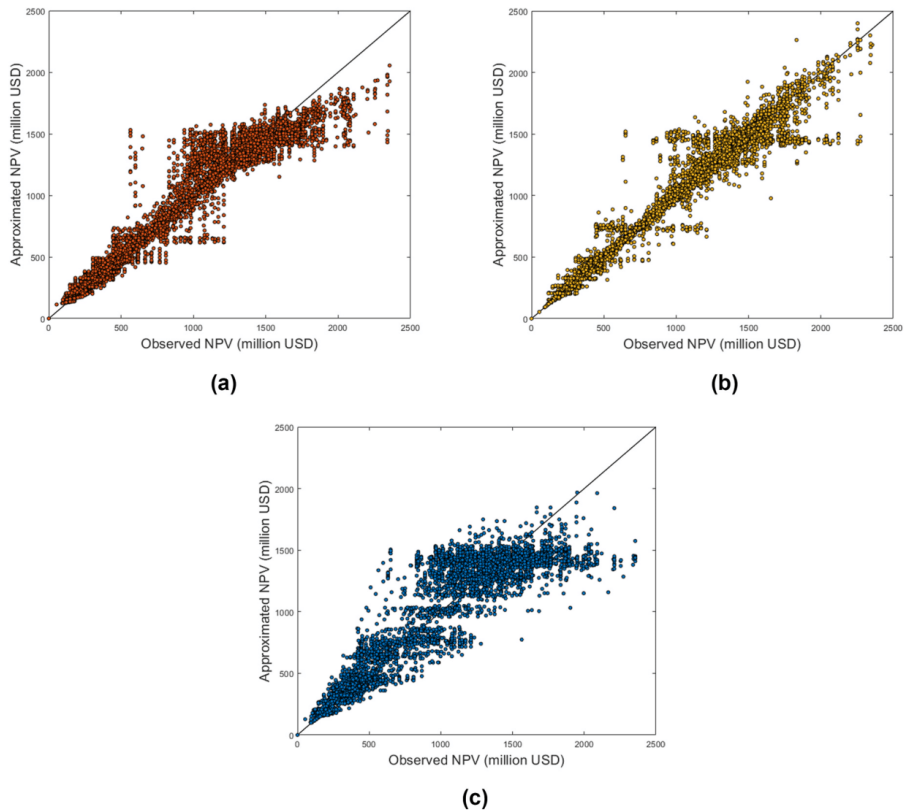


Fig. 3. Plots of observed NPV against approximated NPV for each alternative with (a) LR, (b) GPR, and (c) SVR.

the NCFD plots, it can be observed that only 20% of the realizations result in at most 1 year of primary recovery in the case of LR. Thus, there is 80% chance that 2 years of primary recovery produces the optimal results. For the lifetime of secondary recovery, 90% of all the realizations propose a water injection duration of at most 8 years. This results in 66% chance of 10 years of total lifetime.

Besides that, in the case of GPR, the NCFD plot illustrates that 36% of the realizations recommend the primary recovery of at most 1 year. In addition, there is 84% chance that the water injection should take place for at most 8 years. Regarding the total lifetime, GPR results in a total period of 10 years with 58% chance. When it comes to the NCFD plot of SVR, 98% of the realizations result in primary recovery for at most 2 years whereas 94% of those suggest a waterflooding of at most 8 years. Thereby, 62% of all the realizations result in 10 years of total lifetime. In general, these three techniques (LR, GPR, and SVR) would mostly result in the optimal decision of 2 years of primary recovery and 8 years of waterflooding that contribute to a total of 10 years of production. As it has been explained, DWPIs as suggested in Fig. 2 signify the most optimal decisions. So, if the distribution of DWII is closer to that of DWPI, there is a better chance for the respective EVWII to be higher. Nonetheless, the distributions of DWII for the Lifetime of Primary Recovery estimated by using LR, GPR, and SVR are considerably different from that of DWPI. This also explains the obvious difference between each of the EWII and EWPI.

Fig. 7 compares the cumulative distribution function (CDF) of the

NPVs corresponding to DWOI, DWII (considering all 3 techniques), and DWPI. According to Fig. 7, the more rightward the CDF curve is, the higher the ENPV is. The CDF of  $NPV_{DWOI}$  and the three CDFs of  $NPV_{DWII}$  are close to each other. This proximity resonates with the slight improvement (less than 1%) in the EVWII for the determination of VOI, as discussed earlier. This can be due to the suboptimality of alternatives made for some realizations as the ML-based regressions used are approximate methods.

Fig. 8 (Fig. 9) shows the plot of the mean oil (water) production rate corresponding to DWOI and DWII of 3 different techniques. In the case of DWOI, the mean oil production rate starts increasing after Year 2 because waterflooding is initiated at that time. This is also reflected by the increase in the mean water production rate after Year 2 as shown in Fig. 9. For DWIIs of the three techniques, the initiation time of waterflooding is generally different for different realizations based on the acquisition of information under the framework of SRDM. In this aspect, the issue of suboptimality, as discussed earlier, would occur, and affect the trends of the plots. A tremendous increase after Year 2 is observed. This can be explained by referring to Figs. 4–6, from which more than 50% of the NFD (optimal decision policy) correspond to the lifetime of primary recovery for 2 years.

Despite being limited by the curse of dimensionality due to the increase in the number of alternatives, this work successfully displays the integration of NRS into the framework of modified LSM for optimization of the initiation time of waterflooding under uncertainties. Different

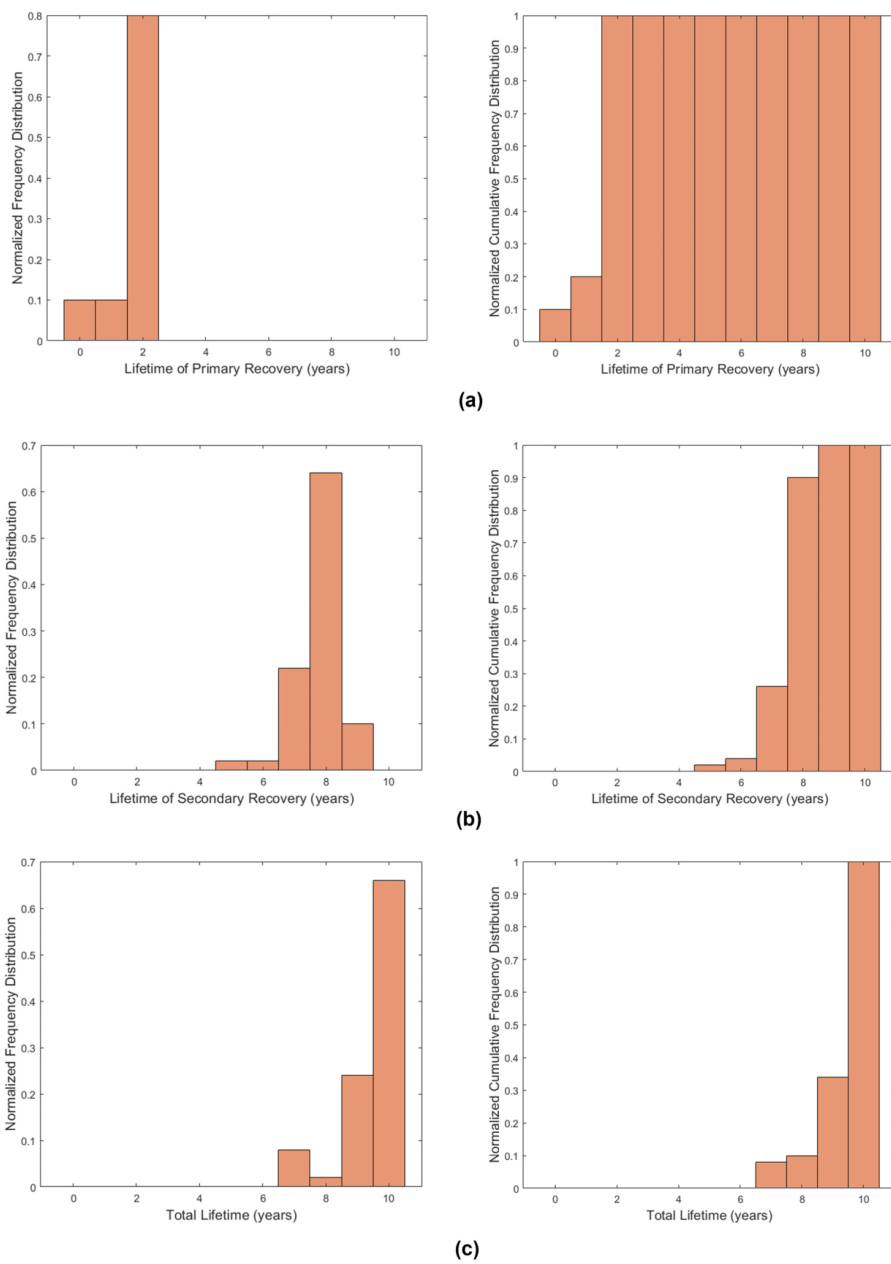


Fig. 4. Distribution of DWII for lifetime of primary recovery, lifetime of secondary recovery (waterflooding), and total lifetime in LR.

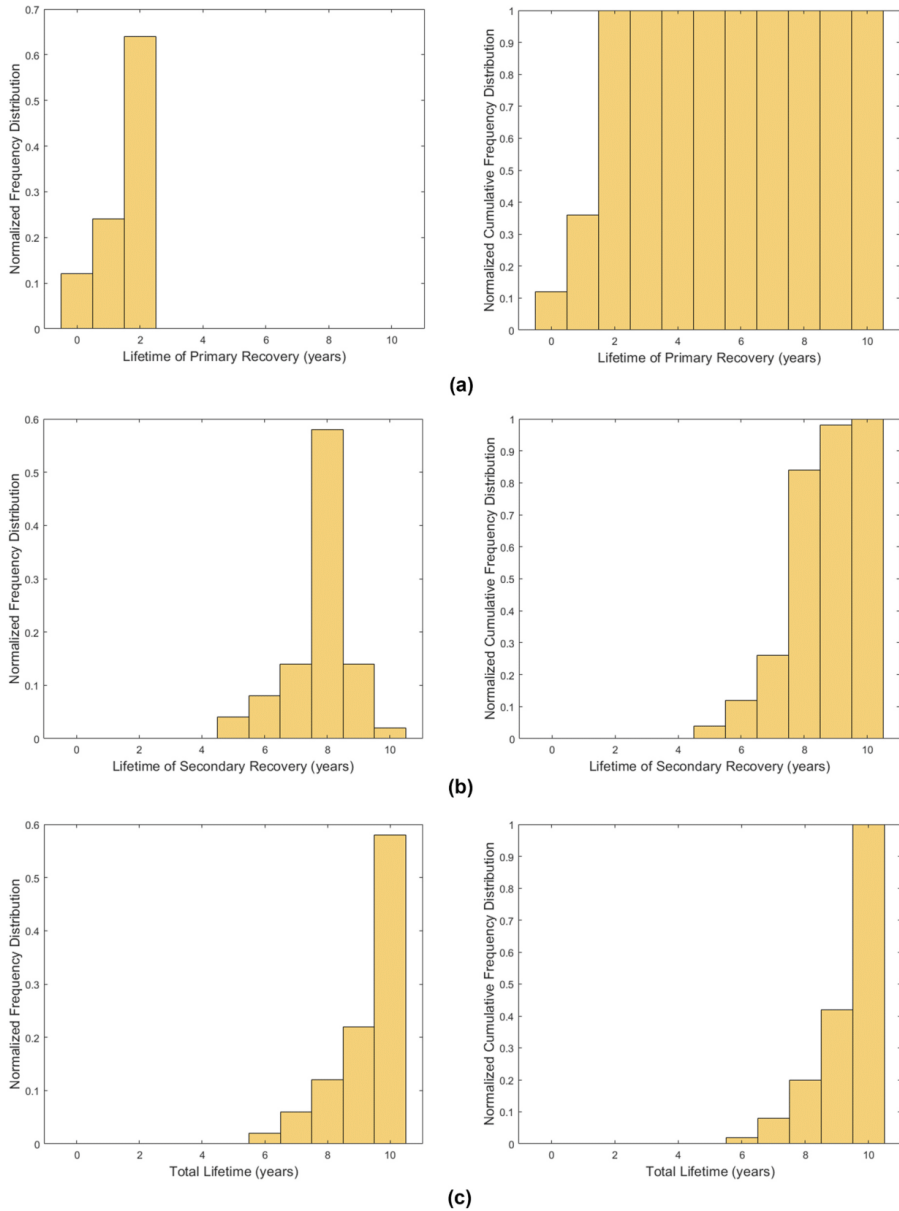


Fig. 5. Distribution of DWII for lifetime of primary recovery, lifetime of secondary recovery (waterflooding), and total lifetime in GPR.

geological uncertainties are considered as previously mentioned to increase the verisimilitude of the case study. Including other types of uncertainty, such as economic uncertainty, is another viable part of future works that can further reinforce the practicality of this methodology. In that case, prices or costs can be modeled by employing a stochastic approach, viz. Two-Factor Price Model (Jafarizadeh and

Bratvold, 2013). Nevertheless, for practical purposes, decision makers need to honor the trade-off between uncertainties and the availability of resources (e.g., financial or labor). This is to ensure that limited resources will not be exhausted to include as many uncertainties as possible.

Some possible extensive applications can be considered in the future

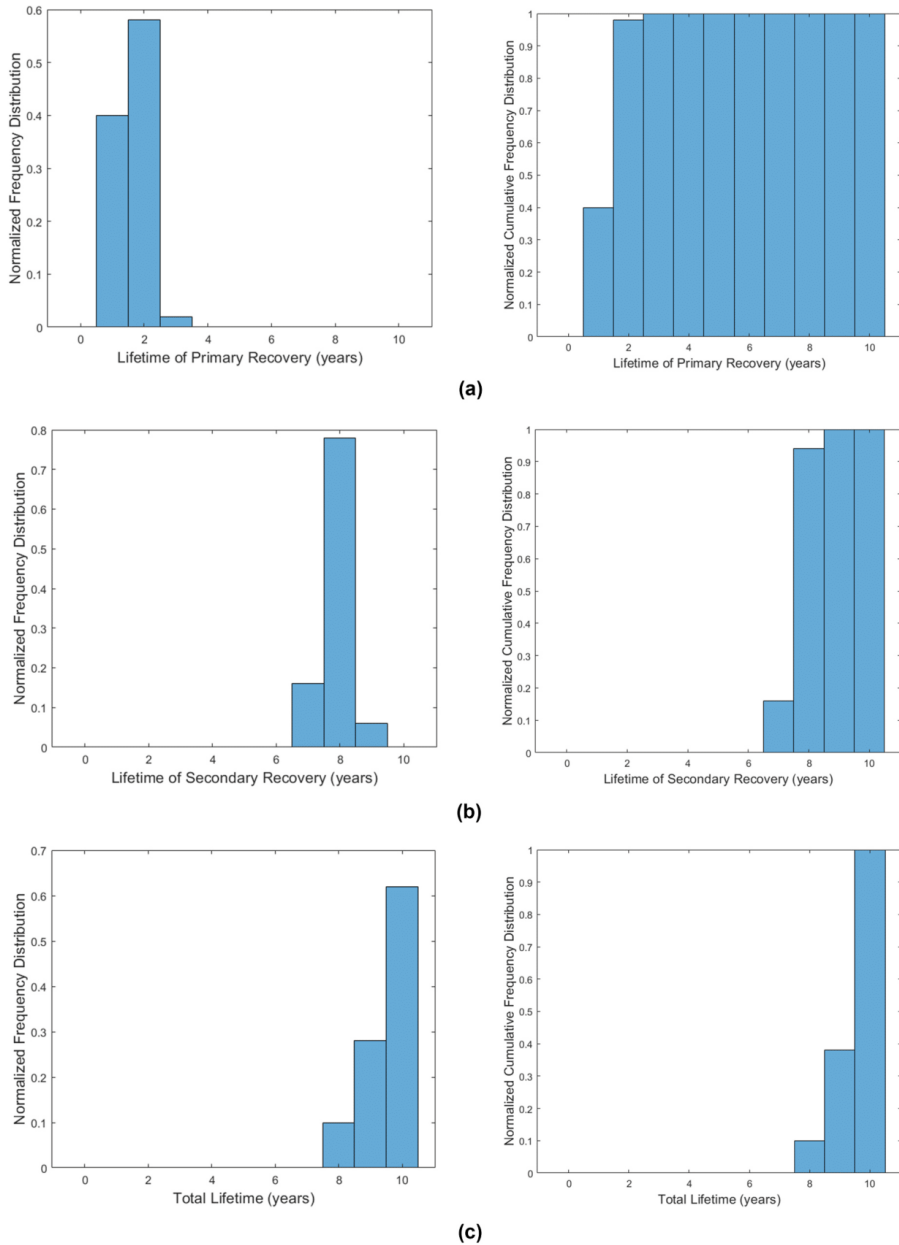


Fig. 6. Distribution of DWII for lifetime of primary recovery, lifetime of secondary recovery (waterflooding), and total lifetime in SVR.

for the work presented here. One of them includes the application of smart proxy models (Mohaghegh, 2022), as substitutes for NRS, under the framework of the modified LSM algorithm. In general, NRS is considered one of the prevalent tools when it comes to RM issues on a

field scale. Nonetheless, using NRS, a geologically complex reservoir model is likely to be computationally prohibitive to be coupled with the modified LSM algorithm. Concerning this, the application of smart proxy models in tandem with the algorithm can be a good recommendation to

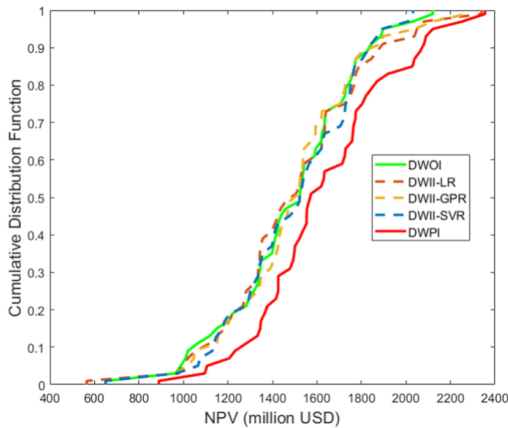


Fig. 7. CDF of NPVs corresponding to DWOI, DWII (considering all techniques), and DWPI.

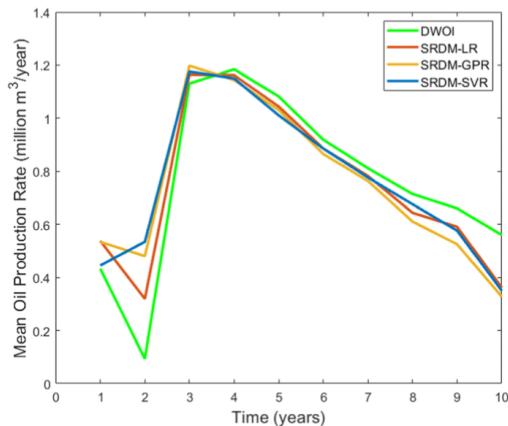


Fig. 8. Plot of mean oil production rate corresponding to DWOI and DWII of all three techniques.

be considered in the future. This demonstrates not only the versatility of the algorithm but also reduces the computational efforts induced by NRS under the paradigm of the SRDM. In this context, the good computational reduction capability of the smart proxy models has been demonstrated in several pieces of literature (Nait Amar et al., 2020, 2021; Ng et al., 2021a, 2021b, 2022).

This work mainly sheds light on the use of supervised learning in the context of the SRDM framework. The robustness of ML can be further highlighted if the use of a more advanced technique, namely reinforcement learning (RL), is embedded in this framework. RL (van Otterlo and Wiering, 2012), generally expounds on the interaction between an intelligent model (an agent) and an environment (a problem setting) to take actions based on the reward. In other words, RL can be perceived as a DM tool. It is thereby worth investigating how RL can be combined with the LSM algorithm for wider applications to improve DM in the aspects of RM. Additionally, the SRDM approach discussed here can also be extended to other domains of reservoir and production engineering,

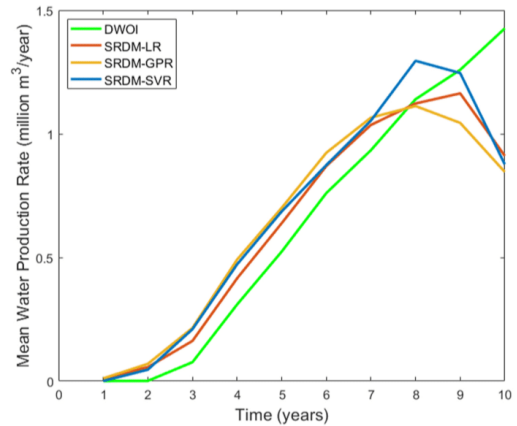


Fig. 9. Plot of mean water production rate corresponding to DWOI and DWII of all three techniques.

such as water-alternating-gas injection, well-placement optimization, and inflow control valve optimization. These optimization problems, to certain extents, are perceived as an example of switching problems, which have been proven to be efficiently resolved by employing the modified LSM algorithm.

## 7. Summary and conclusions

In this work, it has been illustrated and expounded on how the modified LSM algorithm, an epitome of the simulation-regression approach, can be implemented as an SRDM approach to resolve a sequential decision problem in reservoir engineering. Concerning this, optimization of the initiation time of waterflooding in the OLYMPUS reservoir model under geological uncertainties is chosen as the pertinent sequential decision problem. Being different from the initial LSM algorithm proposed by Longstaff and Schwartz (2001), this modified variant integrates the dependency on previously and currently acquired data. In this aspect, the effect of information is shown to be integrated into the context of DM. To enlighten the readers, the mathematical formulations to compute VOI have been concisely explained. There is also a discussion and illustration of how the modified LSM algorithm (as a variant of the simulation-regression approach) can play a part in determining VOI, which is one of the most prevalent DM tools. Besides that, it has been discussed how this algorithm can be implemented as an SRDM approach to resolve the issue of waterflooding initiation time. LR has been the conventional technique of the LSM algorithms. Apart from LR, two other ML-based techniques, viz. GPR and SVR are employed to conduct the regression analysis. Based on our investigation, the DWOI is 2 years of primary recovery followed by 8 years of waterflooding, and the resulting EVVOI is 1479.06 million USD. With the aid of the SRDM approach, the VOIs that are correspondingly estimated by using LR, GPR, and SVR are 11.52 million USD, 11.17 million USD, and 12.46 million USD. Thereafter, the EVWVIs which are estimated by LR, GPR, and SVR, correspond to 1490.58 million USD, 1490.23 million USD, and 1491.52 million USD, respectively. Thus, SVR improves ENPV by the highest percentage, which is 0.84% despite displaying the lowest accuracy during regression analysis. VOI that is approximated by GPR (with the highest accuracy of regression analysis) shows a slightly inferior result, that is improvement of the ENPV by 0.76%. This can be generally explained by the sub-optimality of decisions due to approximation error. Nevertheless, SVR illustrates an improvement of the estimated VOI.

Albeit it is demonstrated that employing non-linear regression ML

based techniques does not guarantee an improvement of VOI in this work (as compared with the VOI approximated by using LR), it provides an insightful demonstration regarding the application of these ML techniques in the context of VOI determination. Also, applying this SRDM approach to the OLYMPUS model can serve as a step closer to the resolution of real-world sequential decision problems. Despite the positive results garnered from this study, several limitations, especially on computational cost due to the forward modeling of a more sophisticated reservoir and the higher number of alternatives, are to be addressed to further improve this methodology. Uncertainty modeling of prices and integration with RL are also considered to improve the robustness of this methodology in the future.

#### Credit author statement

Cuthbert Shang Wui Ng: Data curation, Formal analysis, Methodology, Investigation and Modeling, Coding and Programming, Writing, Editing. Ashkan Jahanbani Ghahfarokhi: Supervision, Writing, Reviewing and Editing.

#### 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

The authors are unable or have chosen not to specify which data has been used.

#### Acknowledgment

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital Automation Solutions for the Oil and Gas Industry ([www.ntnu.edu/bru21](http://www.ntnu.edu/bru21)). The authors acknowledge the suggestions, ideas, and insights provided by Dr. Reidar Brumer Bratvold and Dr. Aojie Hong in the early phase of the formulation of this work. The authors also thank Mr. Wilson Wiranda from the Department of Geoscience and Petroleum, NTNU, for the preparation of the OLYMPUS data.

#### References

- Alkhatib, A., Babaei, M., King, P.R., 2013. Decision making under uncertainty: applying the least-squares Monte Carlo method in surfactant-flooding implementation. *SPE J. Anyosa, S., Bunting, S., Eidsvik, J., Romdhane, A., Bergmo, P., 2021. Assessing the value of seismic monitoring of CO2 storage using simulations and statistical analysis. Int. J. Greenh. Gas Control* 105.
- Bratvold, R.B., Begg, S.H., 2010. *Making Good Decisions*, first. ed. Society of Petroleum Engineers, Richardson, Texas.
- Bratvold, R.B., Bickel, J.E., Lohne, H.P., 2009. Value of information in the oil and gas industry: past, present and future. *SPE Reservoir Eval. Eng.* 12.
- Dutta, G., Mukerji, T., Eidsvik, J., 2019a. Value of information analysis for subsurface energy resources applications. *Appl. Energy* 252.
- Dutta, G., Mukerji, T., Eidsvik, J., 2019b. Value of information of time-lapse seismic data by simulation-regression: comparison with double-loop Monte Carlo. *Comput. Geosci.* 23.
- Eidsvik, J., Dutta, G., Mukerji, T., Bhattacharjya, D., 2017. Simulation-regression approximations for value of information analysis of geophysical data. *Math. Geosci.* 49.

- Fonseca, R.M., Rossa, E. Della, Emerick, A.A., Hanea, R.G., Jansen, J.D., 2020. Introduction to the special issue: overview of OLYMPUS optimization benchmark challenge. *Comput. Geosci.*
- Forrester, A.I.J., Sobester, A., Keane, A.J., 2008. *Engineering Design via Surrogate Modelling: a Practical Guide*. J. Wiley.
- Hillson, D., Murray-Webster, R., 2017. *Understanding and managing risk attitude. In: Second Edition, Understanding and Managing Risk Attitude*, second ed.
- Hong, A., Bratvold, R.B., Lake, L.W., 2019. Fast analysis of optimal improved-oil-recovery switch time using a two-factor production model and least-squares Monte Carlo algorithm. *SPE Reservoir Eval. Eng.*
- Hong, A.J., Bratvold, R.B., Thomas, P., Hanea, R.G., 2018. Value-of-information for model parameter updating through history matching. *J. Pet. Sci. Eng.* 165.
- Howard, R.A., 1980. An assessment of decision analysis. *Oper. Res.* 28, 4–27.
- Howard, R.A., 1966. *Information value theory. IEEE Trans. Syst. Sci. Cybern.* 2.
- Howard, R.A., Abbas, A.E., 2016. *Foundations of Decision Analysis*. Pearson Education Limited, Harlow, England.
- Jafarizadeh, B., Bratvold, R.B., 2013. Sell spot or sell forward? Analysis of oil-trading decisions with the two-factor price model and simulation. *SPE Econ. Manag.*
- Liu, H., Ong, Y.S., Shen, X., Cai, J., 2020. When Gaussian process meets big data: a review of scalable GPs. *IEEE Transact. Neural Networks Learn. Syst.* 31.
- Longstaff, F.A., Schwartz, E.S., 2001. Valuing American options by simulation: a simple least-squares approach. *Rev. Financ. Stud.* 14.
- MathWorks, T., 2022. *MATLAB (R2021b)*. MathWorks Inc.
- Mohaghegh, S.D., 2022. *Smart Proxy Modeling: Artificial Intelligence and Machine Learning in Numerical Simulation*. CRC Press, Boca Raton.
- Nait Amar, M., Jahanbani Ghahfarokhi, A., Ng, C.S.W., Zeraibi, N., 2021. Optimization of WAG in real geological field using rigorous soft computing techniques and nature-inspired algorithms. *J. Pet. Sci. Eng.* 109038.
- Nait Amar, M., Zeraibi, N., Jahanbani Ghahfarokhi, A., 2020. Applying hybrid support vector regression and genetic algorithm to water alternating CO2 gas EOR. *Greenh. Gases Sci. Technol.*
- Ng, C.S.W., 2019. Using the Least-Squares Monte Carlo Algorithm to Optimize IOR Initiation Time. *Norwegian University of Science and Technology, Trondheim.*
- Ng, C.S.W., Ghahfarokhi, A.J., Nait Amar, M., 2022. Production Optimization under Waterflooding with Long Short-Term Memory and Metaheuristic Algorithm. *Petroleum.*
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2021a. Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization. *J. Pet. Explor. Prod. Technol.*
- Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., Torsæter, O., 2021b. Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application. *Nat. Resour. Res.* 30, 2431–2462.
- Olson, R.S., Bartley, N., Urbanowicz, R.J., Moore, J.H., 2016. Evaluation of a tree-based pipeline optimization tool for automating data science. In: *GECCO 2016 - Proceedings of the 2016 Genetic and Evolutionary Computation Conference.*
- Parra Sanchez, C., 2010. *A Life Cycle Optimization Approach to Hydrocarbon Recovery*. The University of Texas at Austin, Austin.
- Powell, W.B., 2011. *Approximate dynamic programming: solving the curses of dimensionality. In: Second Edition, Approximate Dynamic Programming: Solving the Curses of Dimensionality*, second ed.
- Rasmussen, C.E., Williams, C.K.I., 2018. *Gaussian Processes for Machine Learning, Gaussian Processes for Machine Learning.*
- Satter, A., Varnon, J.E., Hoang, M.T., 1998. Integrated reservoir management. *SPE Repr. Ser.*
- Schlumberger, 2019. *Eclipse Reservoir Simulation Software Reference Manual*. schlumberger.
- Shawe-Taylor, J., Cristianini, N., 2004. *Kernel Methods for Pattern Analysis, Kernel Methods for Pattern Analysis.*
- Tadjar, A., Bratvold, R.B., Hong, A., Hanea, R., 2021a. Application of machine learning to assess the value of information in polymer flooding. *Pet. Res.* 6.
- Tadjar, A., Hong, A., Bratvold, R.B., 2021b. A sequential decision and data analytics framework for maximizing value and reliability of CO2 storage monitoring. *J. Nat. Gas Sci. Eng.* 96, 104298.
- van Otterlo, M., Wiering, M., 2012. In: Wiering, M., van Otterlo, M. (Eds.), *Reinforcement Learning and Markov Decision Processes BT - Reinforcement Learning: State-Of-The-Art*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 3–42.
- Wiggins, M.L., Startzman, R.A., 1998. An approach to reservoir management. *SPE Repr. Ser.*
- Willigers, B.J.A., Begg, S.H., Bratvold, R., 2011. Valuation of swing contracts by least-squares Monte Carlo simulation. In: *SPE Economics and Management.*
- Willigers, B.J.A., Bratvold, R.B., 2009. Valuing oil and gas options by least-squares Monte Carlo simulation. In: *SPE Projects, Facilities and Construction.*



## Correction to Typing Error (Errata Sheet)

Notes: A few typing errors are notified in the equations in some of the papers. The corrected equations were actually implemented in the calculation and analysis of these papers. Therefore, the results of the studies are not affected. These corrected equations are presented below. The authors have contacted the respective journal to issue the corrigenda.

Paper 2: Smart Proxy Modeling of a Fractured Reservoir Model for Production Optimization: Implementation of Metaheuristic Algorithm and Probabilistic Application.

Equation (22)

$$gbest_{N,t+1} = \min[f(pbest_{jN,t+1})]$$

Equation (28)

$$R^2 = 1 - \frac{\sum_{i=1}^N (t_i - o_i)^2}{\sum_{i=1}^N (t_i - \bar{t})^2}$$

Paper 3: Application of nature-inspired algorithms and artificial neural network in waterflooding well control optimization.

Equation (6)

$$R^2 = 1 - \frac{\sum_{j=1}^N (y_j^{\text{real}} - y_j^{\text{pred}})^2}{\sum_{j=1}^N (y_j^{\text{real}} - \overline{y^{\text{real}}})^2}$$

Paper 4: Production optimization under waterflooding with Long Short-Term Memory and metaheuristic algorithm.

Equation (14)

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i^{\text{sim}} - Y_i^{\text{proxy}})^2}{\sum_{i=1}^n (Y_i^{\text{sim}} - \bar{Y})^2}$$

Paper 5: Adaptive Proxy-based Robust Production Optimization with Multilayer Perceptron.

Equation (3)

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i^{\text{real}} - Y_i^{\text{pred}})^2}{\sum_{i=1}^n (Y_i^{\text{real}} - \bar{Y})^2}$$

Paper 7: Well production forecast in Volve field: Application of rigorous machine learning techniques and metaheuristic algorithm.

Equation (22)

$$R^2 = 1 - \frac{\sum_{j=1}^N (q_j^{\text{exp}} - q_j^{\text{cal}})^2}{\sum_{j=1}^N (q_j^{\text{exp}} - \bar{q})^2}$$

Equation (25)

$$r(I_k, q) = \frac{\sum_{j=1}^N (I_{kj} - \bar{I}_k)(q_j - \bar{q})}{\sqrt{\sum_{j=1}^N (I_{kj} - \bar{I}_k)^2 \sum_{j=1}^N (q_j - \bar{q})^2}}$$

**“Alles hat ein Ende, nur die Wurst hat zwei.”**

(Everything has an end, only the sausage has two.)