Journal of Engineering Research and Reports

Volume 26, Issue 6, Page 371-387, 2024; Article no.JERR.116622
ISSN: 2582-2926

History Matching in a Reservoir Model Using an Automatic Approach

Chinedu Ikenna Ndubuka [a*] and Okon, Edet Ita [b]

[a] *Department of Petroleum Engineering, University of Uyo, Nigeria.*
[b] *Department of Petroleum & Gas Engineering, Topfaith University, Mkpatak, Akwa Ibom State, Nigeria.*

Authors' contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: https://doi.org/10.9734/jerr/2024/v26i61187

Open Peer Review History:
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/116622

Original Research Article

Received: 01/04/2024
Accepted: 04/06/2024
Published: 06/06/2024

ABSTRACT

History matching may be seen as an optimization problem based on minimizing an objective function that measures the mismatch between reservoir history and simulated data. Manual methods of history matching are largely cumbersome and grossly ineffective especially when the optimization parameters are large. Recourse to the manual method leads to development of models which cannot accurately predict the reservoir behaviour and thus are not suitable for predicting future behaviour of the reservoir. The only way out is the use of automatic methods especially those backed by artificial intelligence. The study aims at applying an automatic method to perform history matching in a reservoir model. The objectives will be to; Perform automatic history matching using ABC, match permeability distribution in the reservoir using oil production and bottom hole flowing pressure data and compare the effectiveness and convergence speed of the algorithm. History matching aims at fine-tuning the parameters used in building a reservoir model to closely match that of the real field. In this study, a very promising novel optimization algorithm has been employed to history a well-known reservoir model namely the PUNQ-S3 model. The model used for this study

**Corresponding author: Email: eddymyprince@gmail.com;*

Cite as: *Ndubuka, Chinedu Ikenna, and Okon, Edet Ita. 2024. "History Matching in a Reservoir Model Using an Automatic Approach". Journal of Engineering Research and Reports 26 (6):371-87. https://doi.org/10.9734/jerr/2024/v26i61187.*

is the popular PUNQ-S3 reservoir model. PUNQ (Production forecasting with Uncertainty Quantification) is a joint industrial-academic project with the aim of developing efficient history matching and uncertainty quantification methods. Results obtained proves the algorithm used to be a very efficient optimization tool as the data used as the history of the study is nearly equaled by the optimization tool. We therefore conclude that the ABC algorithm be employed in performing tasks that demand high degree of accuracy.

Keywords: Matlab; eclipse; history matching; reservoir simulation.

1. INTRODUCTION

In order to better understand reservoir petrophysical parameters and fluid characteristics, reservoir engineers use reservoir simulation, a potent and crucial numerical modeling technique. It is primarily used to forecast reservoir behavior in various scenarios, which helps with field development decisions. The reservoir model's capacity to forecast reservoir behavior is one of its primary objectives. Validating a model presents a number of challenges because the majority of oil reservoirs are awkwardly buried behind thousands of feet of overburden. Only at well locations, which are frequently hundreds of meters away, are direct observations of the reservoir possible [1]. As a complicated nonlinear system, the reservoir itself is rarely fully understood with precision [2]. The performance of a model's prediction depends on accurate assessments of certain physical properties, such the reservoir's permeability distribution [3]. Computational models rely on a variety of reservoir metrics and features. The history matching procedure is one method of approximating these features.

The process of modifying erratic reservoir parameters until a satisfactory match with the measured production data is achieved is known as history matching. When reservoirs have been in operation for a while, the inverse problem of estimating reservoir attributes can be solved by comparing simulated data to reservoir history [4]. The procedure aids in fine-tuning the simulation's reservoir parameters to match the real reservoir's data. The main objectives of history matching are:

- Improve and validate the reservoir simulation model
- Better understanding of reservoir processes
- Improve the reservoir description and data acquisition program
- Identify unusual operating conditions

According to Riazi et. al., [5], the main stages of the history matching process involves:

- selecting parameters,
- defining the mathematical model,
- defining the objective function,
- sensitivity analysis and stop conditions

The general strategy for history matching according to Ertekin et. al*.,* 2002 is shown in Fig. 1.

Historically, history matching was carried out manually by:

1) Running simulation for historical period.
2) Comparing results to actual field data
3) Adjust simulation input to improve match
4) Selecting input data based on knowledge and experience.

Because so much data was usually involved, this history matching process was exceedingly laborious, time-consuming, and generally useless. This method is time-consuming and computationally expensive because to its complexity and lack of knowledge about the features of the reservoir. Because there are so many reservoir parameters, it is challenging to modify the parameters in order to get the match. Artificial intelligence (AI), a reliable technique that will carry out the task automatically in a more accurate but economical manner, is therefore required. The automatic methods that are currently in use solve the issues that the manual method left behind. The process involves utilizing optimization algorithms to reduce an objective function that gauges the discrepancy between the simulation findings and the observed reservoir performance, up until a point at which the difference is deemed acceptable. Automatic methods used in the Petroleum industry are summarized as follows:

- This category uses deterministic (gradient-based) techniques like Direct Pattern Searching and the Descent Method. The primary difficulty with these approaches is calculating the gradients, which is often accomplished using one of two different approaches: the adjoint-based method or the finite difference (FD) method.
- Some evolutionary algorithms, like Genetic Algorithm (GA), Evolutionary Strategy (ES), and, of course, our ABC; these include evolutionary algorithms such as Simulated Annealing (SA), PSO, and Simultaneous Perturbation Stochastic Approximation (SPSA).
- A hybrid approach, which combines helper techniques and proxies (such as kriging proxies) with stochastic and deterministic methods.

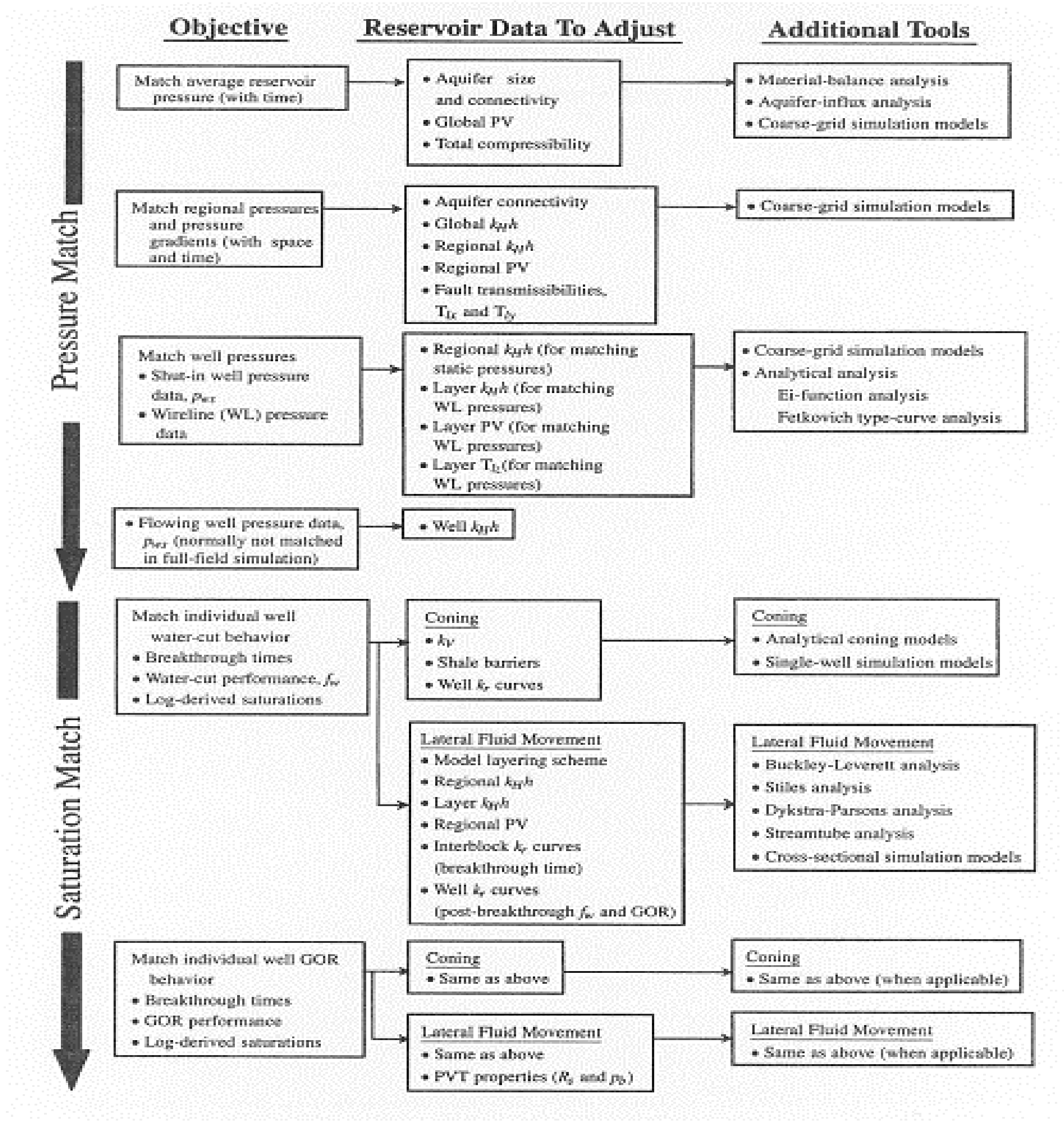

Fig. 1. General strategy for history matching

Only the Artificial Bee Colony (ABC) algorithm will be studied in this paper. Artificial intelligence is a powerful tool for automating processes. It is impossible to overstate how important historical matching is. data from utilizing the model can be highly trusted if the simulation's reservoir model closely resembles the real reservoir. This allows field development decisions to be made using the data with less anxiety or dread. All things considered, history matching increases the forecasting ability of simulation models by precisely adjusting the model's parameters to match those of the real-world reservoir, improving the model's effectiveness and accuracy.Optimization algorithms help to automate the otherwise cumbersome process making it iterative, more efficient, and accurate.

2. LITERATURE REVIEW

Although history matching is not a brand-new area of research in the field, it is essential to updating the model's attributes, which have previously been used to closely match the reservoir's features by comparing data collected in the field with data received from the model. The conclusions derived from the model cannot be implemented with confidence in the field because the majority of the data used to describe it at first are questionable. A better model of the real field can be achieved by fine-tuning the unknown parameters, and the resulting results can be relied upon. There have been attempts over the past 20 years to enhance automatic history matching in a way that may find practical use [6].

There is still doubt regarding the viability and potential of these approaches to handle extremely complex real reservoir models, despite all of the attempts, because of the pace of complexity and resolution in the reservoir models. Because of this, automatic history matching is still a difficult and popular study area. The 1960s saw the beginning of the first historical matching research [7]. The primary methodology of the study involved developing mathematical reservoir models and calibrating them with real data. Using experimental design to create reaction surfaces in place of reservoir simulation in the history matching workflow was a significant innovation in the 1990s [8].

Substantial efforts were made in the years after the early 1990s to transition history matching from an engineer-based framework that required a lot of labor to a completely or partially automated method [9]. In the late 1960s, history matching was done using gradient optimization techniques [7]. The notion of integrating a simulator and the Simultaneous Perturbation Stochastic Approximation (SPSA) approach for automatic history matching of multiphase flow production data was first proposed by Gao et al. (2004). A continuous Ant Colony Optimization (ACO) method is the foundation of the stochastic approach for automatic history matching presented by Hajizadeh et al. [10]. In this context, other stochastic algorithms have been studied. In history matching, evolutionary algorithms are becoming more and more used as a conventional optimization technique. These algorithms are generally inspired by the evolution theory.

Numerous instances exist where these methods have been utilized for historical matching. The ensemble Kalman filter [11], genetic algorithms [12], particle swarm optimization (PSO) [13], Ant Colony Optimization (ACO) algorithm (Hajizadeh, et al., 2011), Markov chain Monte Carlo [14], and chaotic optimization [15] are a few of the techniques that have been successfully applied. A barrier for history matching procedures has been caused by reservoir models' increased complexity and simulation time. This is especially true for workflows using history matching and population-based sampling methods. Hundreds to thousands of simulation calls are needed for these algorithms to converge to optimal regions and identify history-matched solutions, depending on the number of uncertainty parameters. Due to this criterion, applications of stochastic population-based techniques for uncertainty quantification and real-world history matching have encountered well-known obstacles. Simultaneously, the limitation has spurred research efforts to shorten reservoir model simulation times.

2.1 Two Distinct Areas form the Current Focus of Research Activities

1) mathematical models to improve the physics-based simulation
2) reduced order/data-driven approaches as a proxy to full field simulation.

In many engineering domains, proxy models are widely employed as a low-cost approximation to complete field simulation models, which have substantial computational costs. Proxy models entered the field of petroleum engineering as a result of the reservoir simulation models' increased cost and processing time. They develop quickly and with considerable ease. Nonetheless, there is still a long way to go before comprehensive field reservoir simulation models in reservoir management plans are fully achieved due to practical considerations. In petroleum engineering, the two most well-known types of proxy models are response surface models and reduced order models. Transforming the high dimensional models into a meaningful representation of lower dimensionality is the goal of reduced order modeling. Petroleum engineering is one of the numerous fields in which they have been used. In recent years, there have been some attempts in using reduced order models for history matching, uncertainty quantification, and optimization Gildin, et. al., (2014).

Another strategy that has been rapidly developing recently is data-driven modeling. The foundation of data-driven modeling is machine learning analysis of the available data about a system. In particular, this method establishes links between various system components without requiring explicit knowledge of their physical characteristics. Data-driven modeling techniques include the use of fuzzy logic, artificial neural networks, and statistical techniques. Surrogate Reservoir Models (SRMs) are a relatively new class of data-driven models used in reservoir modeling and simulation. SRMs are designed to either supplement or replace reservoir simulation models. They are based on artificial intelligence and data mining techniques.

A study by Shahkarami et al. [6] sought to determine how pattern recognition technologies might be applied to shorten the time and effort needed to finish a successful history matching project. The assisted history matching process is carried out by using data mining techniques and artificial intelligence's pattern recognition capabilities to create a Surrogate Reservoir Model (SRM). Their study's findings demonstrated how SRMs can support the history matching process in reservoir management workflows by aiding in the use of population-based sampling techniques and other computationally demanding processes.

A history matching problem aiming at estimating a reservoir's permeability field from the pressure and flow rate recorded in the wells was introduced by Amorim et al. [4]. The two-phase incompressible flow model served as the foundation for this reservoir simulation. The Gauss-Newton and Truncated Singular Value Decomposition (TSVD) techniques are combined in this method. They claimed that the number of grid blocks utilized to discretize the reservoir determined the number of parameters that needed to be approximated. This value was generally high, and the inverse problem was poorly formulated. The TSVD approach regularizes the problem and significantly reduces the amount of computing power required to solve it. They combined the Lanczos approach with numerical implementations of the adjoint formulation of the problem and the derivative to compute the TSVD. Zhang et al. [2] carried out automatic history matching in a numerical reservoir model by utilizing an enhanced GA. The study's total water cut of the blocks served as the observed data, while the relative permeability curve, average interlayer permeability values, and permeability coefficient of variation among layers served as the study's modifiable parameters. Their study's findings, which demonstrated the method's strong reliability and quick convergence speed, were verified using the SZ 36-1 typical reservoir of the Bohai oilfield. They came to the conclusion that the approach works as a result. Xavier et. al., [3] presented a study of GA for the history matching problem of reservoir 2D flow simulation model. In their work, they studied the effect of parameter adjustment to the algorithm performance.

Using Least Square Support Vector Machine (LSSVM) as a proxy model and the PSO and Imperialist Competitive Algorithm (ICA) as base optimization algorithms, Riazi et al. [5] performed history matching on a fractured reservoir model. Their method involves building a proxy model based on the field's historical data to represent the history match objective function (mismatch values). Next, using PSO and ICA, this model is applied to minimize the objective function. The approach is effective for the history matching process, as demonstrated by the obtained results, because of its strong reliability and quick convergence. They concluded that due to high speed and need for small data sets, LSSVM is the best tool to build a proxy mode. Also, the comparison of PSO and ICA shows that PSO is less time-consuming and more effective.

Afiakinye [16] optimized well location using a modified version of the ABC algorithm. The basic ABC algorithm has been modified to prevent visits to sites that have previously been visited and to handle the process of pushing out-of-boundary points back into the feasible search zone more effectively. The study's findings demonstrate that the updated algorithm outperforms the original algorithm.

2.2 The optimization tool to be used in this study is the Artificial Bee Colony and is discussed as follows

2.2.1 Artificial bee colony algorithm

Among the swarm algorithms is ABC. An assembly of fish, birds, and insects like termites, ants, and bees is referred to as a swarm. Due to their observation of their surroundings, the individual agents within the swarm act in an unsupervised manner and exhibit stochastic behavior. The networks of interactions between these simple agents and between agents and their surroundings are what give the swarm its intelligence. Typically, the first ABC solutions are produced at random and refined during the optimization process. The first answers are derived from equation (3).

$$x_j(i) = LB_i + (UB_i - LB_i) \times r \tag{1}$$

Where *r* ~ (0, 1) is a random number between 0 and 1.

The neighborhood of the solution that needs to be enhanced is where improvements are made to solutions. Usually, a fixed number of cycles, known as the limit, are given to enhance a randomly produced solution. When this limit is reached, the solution is dropped, and a randomly generated new solution takes its place. The procedure is then repeated until the user-specified stopping condition is satisfied. Because ABC uses so few parameters to operate, it is comparatively easy to build. Basically, three kinds of bees are defined in the algorithm thus:

- Worker Foragers: These bees are linked to certain food supplies. Food sources are described as possible fixes for the issue at hand. Their duties include gathering and storing data regarding a specific food source, as well as informing other bees in the hive about its position and nectar quality (fitness). This duty also includes the perturbation (creation of a trial solution in the immediate vicinity of the existing food source) and the assessment of the fitness value of the new (perturbed) food source. This is done using equation (4)

$$v(i,j) = x(i,j) + rand[-1,1]\big(x(i,j) - x(k,j)\big) \tag{2}$$

Here,
$j \epsilon\ (1,2,\ldots,\mathrm{D})$ and $\mathrm{k}\ \epsilon\ (1,2,\ldots,\mathrm{SN})$ where $(\mathrm{k} \neq \mathrm{i})$ are randomly generated indices and rand [-1, 1] is a random number in the range [-1, 1], which works as a scaling factor. SN is the population or colony size and D is the dimension of the of the optimization problem. The old food source is perturbed until the designated limit is reached, at which point it is abandoned and a new source is generated. If the fitness of the perturbed solution is greater than that from which it was formed, then the new solution replaces the old one in the employed forager's memory. This is a greedy-selection scheme.

- Onlooker Bees: These bees watch as the employed foragers circle a food source. Using the information they exchange, they determine the likelihood that they will select that specific food source and compare it to a number that is created at random between 0 and 1. Equation (5) is used to determine the probability values Pi for the solutions xi based on their fitness values.

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i} \tag{3}$$

Where *fit*$_i$ is the fitness value of solution x_i. P_i values were normalized into [0, 1].

In order to calculate the fitness values of solutions, equation (6) was employed:

$$fit_i = \begin{Bmatrix} \dfrac{1}{1+f_i} & if\ f_i \geq 0 \\ 1 + abs(f_i) & if\ f_i < 0 \end{Bmatrix} \tag{4}$$

Here, f_i is the value of the objective function for solution x_i.

Only when a food source's fitness exceeds the number that is randomly generated is it selected; this food source is then committed to memory, and the observing bee uses equation (4) to create a new trial solution in the vicinity of the best solution. If this new solution proves to be more fit than the prior one in her memory, the onlooker bee will also employ the greedy-selection technique.

- Scout bees: These bees are recruited by the onlooker bees to randomly choose new food sources around the entire search space to replace food sources that could not be improved within the set limit. They are different in their operation to the other previously described bees in the sense that they are not bounded to generating food sources around old sources so as to improve them but can generate food around the entire search space without exiting it. In the procedures above, it is presumed that an onlooker bee whose food source has sufficiently depleted, or cannot be improved after reaching the specified limit becomes a scout bee.

3. MATERIALS AND METHODS

3.1 Simplification of Reservoir Model

According to Goda et al. (2010), there are so many unknowns surrounding reservoir parameters during the reservoir history matching process that it takes a long time to change a lot of model parameters for a brief period of time and can easily lead to non-convergence in automatic history matching. The permeability vector, designated by K, is the model parameter that needs to be optimized in this study in order to increase fitting accuracy and decrease optimization time.

3.2 Theory

In the oil sector, history matching is a well-known inverse problem [1]. In the forward problem, the reservoir's physical characteristics are known, and its production behavior is computed via a simulator. Given the observed production data of an actual reservoir, the objective of the inverse problem (history matching) is to predict probable physical attributes of the reservoir. The history matching procedure is used to estimate the physical parameters of the reservoir because they cannot be physically monitored in all of its extensions. The purpose of using the predicted attributes as simulator parameters is to forecast reservoir behavior under various production scenarios. The objective of this work's inverse problem is to estimate a reservoir's absolute permeability field by history-matching its production data, which is provided by the oil rate and bottom-hole pressure periodically observed at well locations. We indicate the to-be-determined permeability by K, the simulated data given the parameter K by S(K), and the observed data by O. Finding K that minimizes the least square formulation is the problem.

$$f(K) = \|S(K) - O\|^2 \qquad (5)$$

The problem to be solved is then formulated as a minimization problem thus,

$$\min f(K) \qquad (6)$$

In the context of Evolutionary Algorithms, *f* (*K*) is called *fitness function* on dealing with Evolutionary optimization algorithms. Its importance will be elucidated in the following sections. In this work we transform the fitness function in a relative error measurement, as follows

$$f(K) = \frac{\|S_o(K) - O_o\|^2}{\|O_o\|^2} + \frac{\|S_p(K) - O_p\|^2}{\|O_p\|^2} \qquad (7)$$

Where subscript *o* and *p* denote oil rate and bottom-hole pressure observations, respectively.

3.3 Implementation Details and Computer Platform

The algorithms will be implemented in Matlab® 2015 run on an HP Intel Pentium processor of 4.0GB RAM, 500GB Hard disk and 2.2GHz processing speed. The modelling of the ereservoir will be done with ECLIPSE 100® run on the same system.

3.3.1 Matlab

The proprietary multi-paradigm programming language and numerical computing environment known as MATLAB (an acronym for "MATrix LABoratory") was created by MathWorks. It enables the design of user interfaces, the execution of algorithms, the graphing of functions and data, matrix manipulations, and interfaces with programs written in other languages (Wikipedia, 2022). Millions of scientists and engineers use MATLAB to build models, design algorithms, and analyze data. It blends a programming language that represents matrix and array mathematics directly with a desktop environment tailored for iterative analysis and design processes. It comes with the Live Editor, which is used to write scripts that integrate formatted text, code, and output into an executable notebook (Mathworks, 2022). MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

Typical uses include:

- Math and computation

- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including Graphical User Interface building

An array is the fundamental data element in MATLAB, an interactive system that doesn't need to be dimensioned. This makes it possible to complete a lot of technical computing tasks more faster than you could if you were writing a program in a scalar, noninteractive language like C or Fortran. This is especially true for problems involving matrix and vector formulations. With the help of numerous users throughout the years, MATLAB has changed over time. It is the typical teaching aid in university settings for beginning and advanced science, engineering, and math classes. MATLAB is the preferred tool in industry for highly productive research, development, and analysis. Toolboxes are a kind of application-specific solutions available in MATLAB. Toolboxes are essential for the majority of MATLAB users since they let you learn and use specialist technologies. A toolbox is an extensive set of MATLAB functions (M-files) that enhances the MATLAB environment to address specific issue classes. Toolboxes are available for many different domains, such as simulation, wavelets, fuzzy logic, neural networks, control systems, and signal processing.

3.4 The MATLAB System

The MATLAB system consists of five main parts:

3.4.1 The MATLAB language

Control flow statements, functions, data structures, input/output, and object-oriented programming capabilities are all included in this high-level matrix/array language. It enables "programming in the large" to develop comprehensive, large-scale, sophisticated application programs as well as "programming in the small" to quickly create short, dirty, throw-away programs.

3.4.2 The MATLAB working environment

This is the collection of resources and tools available to you as a programmer or user of MATLAB. It has tools for managing the variables in your workspace as well as data input and export capabilities. It also comes with tools for creating, organizing, troubleshooting, and profiling M-files, which are programs used with MATLAB.

3.4.3 Handle graphics

The MATLAB graphics system is this. High-level commands for image processing, animation, presentation graphics, and two- and three-dimensional data visualization are included. It also comes with low-level instructions that let you create entire Graphical User Interfaces for your MATLAB applications and completely change the look of graphics.

3.4.4 The MATLAB mathematical function library

This is a huge collection of computing algorithms that includes more difficult functions like matrix inverse, matrix eigenvalues, Bessel functions, and rapid Fourier transformations, as well as more basic functions like sum, sine, cosine, and complex arithmetic.

3.4.5 The MATLAB application program interface (API)

Using this library, you can create C and Fortran programs that communicate with MATLAB. It has tools for reading and writing MAT files, calling MATLAB as a computational engine, and calling MATLAB routines (dynamic linking).

3.4.6 Eclipse

The most comprehensive and reliable collection of numerical solutions available to the industry for quick and precise dynamic behavior prediction for all kinds of reservoirs and development plans is provided by the ECLIPSE industry-reference simulator. For over 25 years, the ECLIPSE simulator has set the standard for commercial reservoir simulation because of its wide range of features, stability, speed, parallel scalability, and unparalleled platform support.

The ECLIPSE simulator is the most feature-rich and complete reservoir simulator available, including all reservoir models, including black oil, compositional, thermal finite-volume, and streamline modeling. It has been developed and innovated over 30 years. Your reservoir modeling skills can be improved by customizing the simulator's capabilities with a variety of add-on options, including advanced wells, reservoir coupling, surface networks, coalbed methane, local grid refinements, and gas field operations.

The ECLIPSE simulator, which is in use at more than 800 locations across 70 countries, benefits from the highest caliber of reservoir engineering knowledge in the sector. In addition to being widely utilized by academic institutions, regulatory bodies, and petroleum financial planners, the ECLIPSE simulator is the industry standard for modeling in the petroleum sector. The ECLIPSE simulator is widely regarded as the best reservoir simulator in the business, as shown by its citations in more than 1,500 SPE technical publications.

3.5 Case Study

The well-known PUNQ-S3 reservoir model served as the model for this investigation. The goal of the collaborative industry-academia project PUNQ (Production forecasting with UNcertainty Quantification) is to provide effective techniques for uncertainty quantification and history matching. The PUNQ-S3 reservoir simulation model is a five-layer model, according to Floris [Floris et al. 2001]. The PUNQS3 reservoir's maximum depth is 2430 meters. It is surrounded by a fault to the east and south, with a moderately strong aquifer to the north and west providing pressure support. It has a dip angle of around 1.5 degrees. In this reservoir, no injection wells have been drilled due to the pressure support. In layer 1 of the PUNQ-S3 reservoir model, there is also a tiny gas cap.

This layer has no completed wells due to the impact of free gas production on reservoir recovery. Fig. 1 shows that six producing wells are designated with black dots. The first gas-oil contact is close to where these wells are situated. Perforations are found in layers 4 and 5 for Producers 1 (PRO-1), 4 (PRO-4) and 12 (PRO-12). Producers 5 (PRO-5) and 11 (PRO-11) have finished layer 3 and layer 4, whereas producer 15 (PRO-15) has just layer 4 perforation. Near an aquifer, PRO-4 has been finished, and in the seventh year, water breakthrough has been seen. In PRO-1 and PRO-4, free gas production begins in the fourth and fifth year.

About two thirds of the 19 by 28 by 5 grid blocks (1761) in the PUNQ-S3 model are active. The 180-meter sides of the grid blocks are equal in both the x and y axes. The Carter-Tracy aquifer type's corner point geometry was used to model the reservoir simulation example. The whole reservoir data set is accessible online [PUNQ 2010]. A common benchmark model for evaluating and contrasting the innovative techniques created for uncertainty quantification and history matching is the PUNQ-S3 reservoir model. The findings of numerous others' studies on the PUNQ-S3 reservoir model have been published. For the history matching of this model, Soleng employed a steady state genetic algorithm [Soleng, 1999].

Using the PUNQ-S3 scenario as an example, Manceau [Manceau et al. 2001] introduced an integrated approach for history matching and uncertainty analysis based on gradual deformation techniques and Fast Fourier Transform-Moving Average (FFTMA) methods. Using the same reservoir model, Mantica combined progressive deformation with chaotic optimization [15]. Demyanov used a geostatistical framework in conjunction with the Neighbourhood Algorithm (NA) to match the PUNQ-S3 model's history [Demyanov 2004]. Gao evaluated two different versions of the Simultaneous Perturbation Stochastic Approximation (SPSA) method on the PUNQ-S3 reservoir in order to solve the reservoir history matching problem [Gao et al. 2007].

3.6 Methods

1. The parameter that needs to be matched is the field's permeability distribution.

2. An objective function has to be constructed in order to use an optimization technique for history matching. The objective function calculates the discrepancy, or mismatch, between the reservoir history and the simulated data produced by ECLIPSE 100 following each simulation.

3. The algorithm uses this function to assess how good of a match is. Equation 9 defines the objective function used in this investigation.

4. The ABC algorithm creates a new permeability based on the objective function's value.

5. A simulation in ECLIPSE 100 is performed using the generated permeability. For each

simulation run, data are generated. The data so generated are read by Matlab and used in calculating the objective function.

6. The objective function is computed and then submitted to the ABC algorithm, which iteratively generates a new permeability value based on the objective function's value.

7. When the objective function reaches a user-defined maximum number of optimization cycles or a predetermined minimum value, optimization is said to have finished.

4. RESULTS AND DISCUSSION

The PUNQ-S3 model's petro-physical characteristics are displayed in Figs. 2–5. The study's variables are the well sites' permeability values from the reservoir model, whilst the model's history is derived from the well bottom-hole pressures and oil production rate. As a result, these were matched using the ABC method, with the matching results shown in Fig. 2.

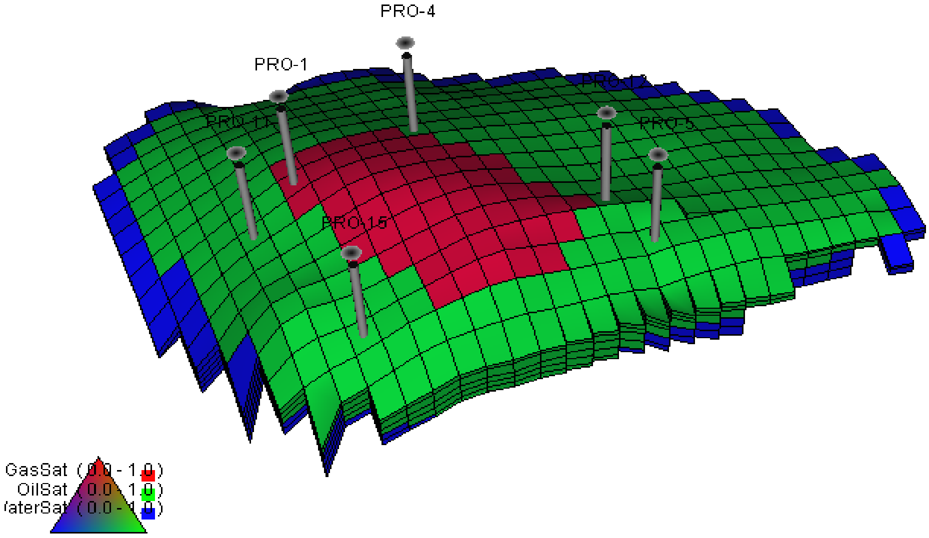

Fig. 2. Fluid saturation distribution in the model

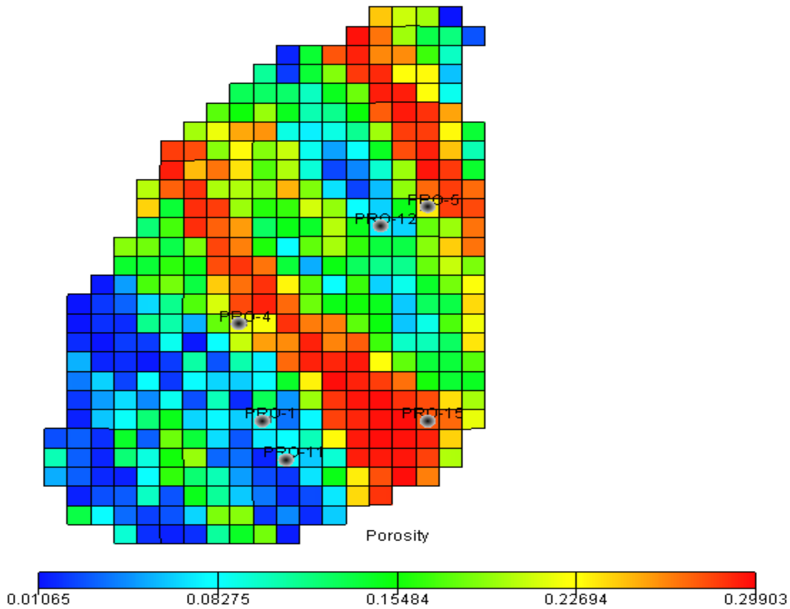

Fig. 3. Porosity distribution in the model

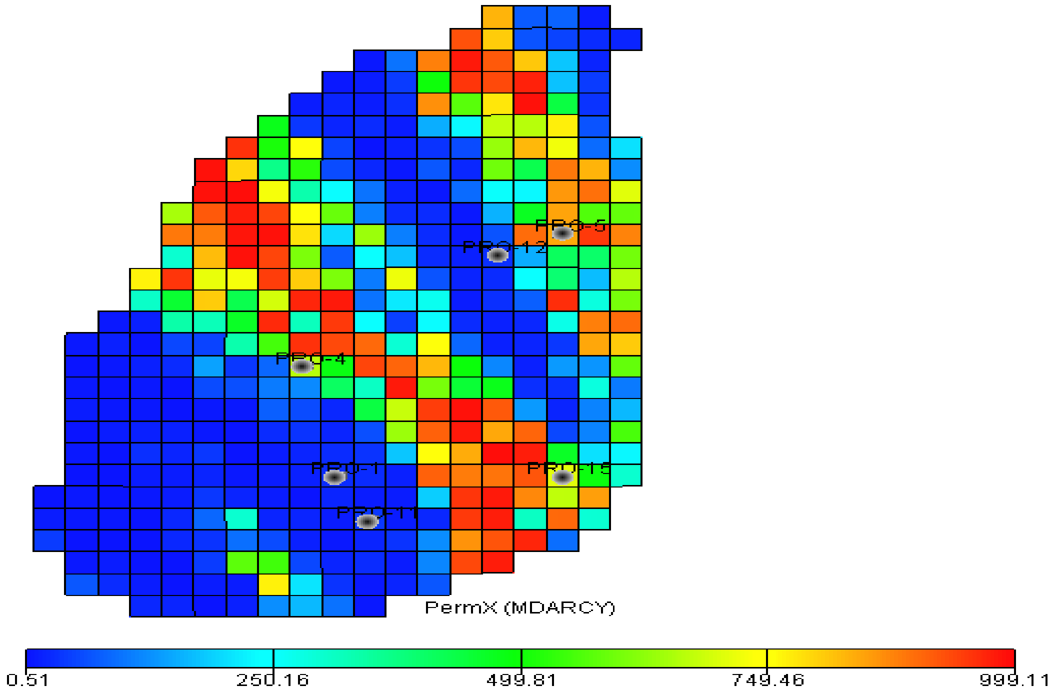

Fig. 4. Permeability distribution in the model

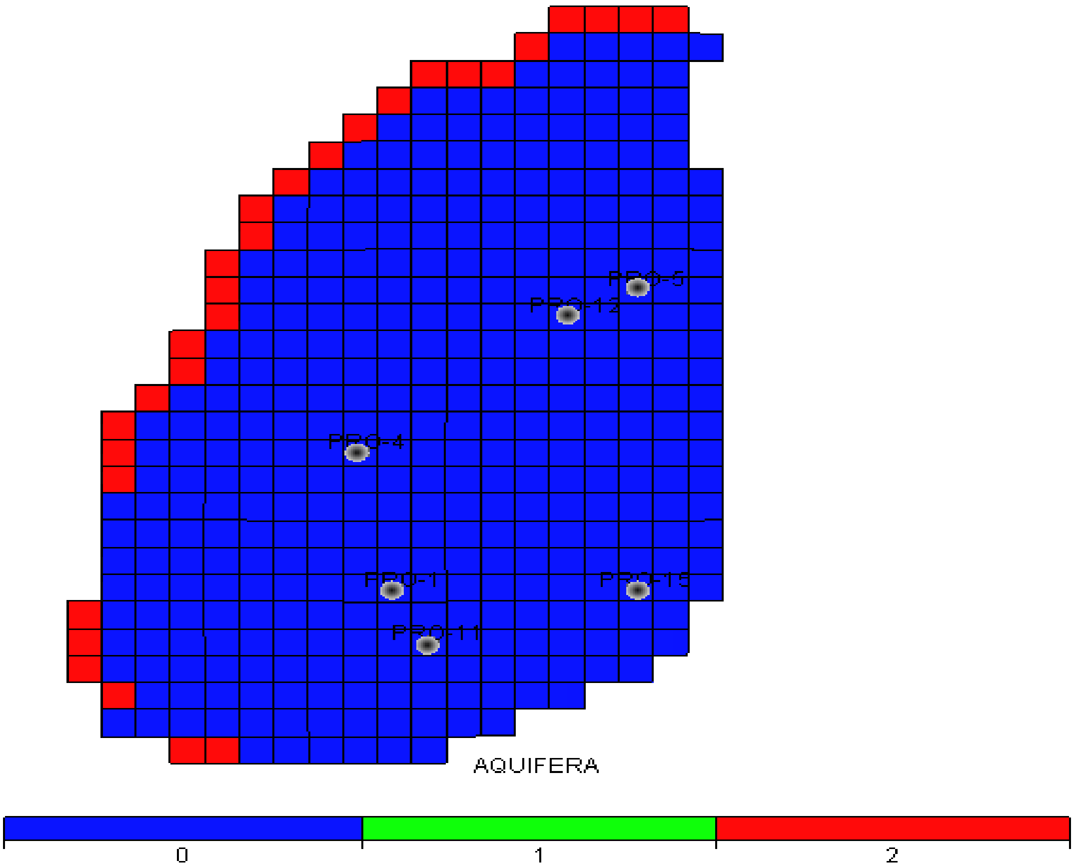

Fig. 5. Aquifer bound around the model

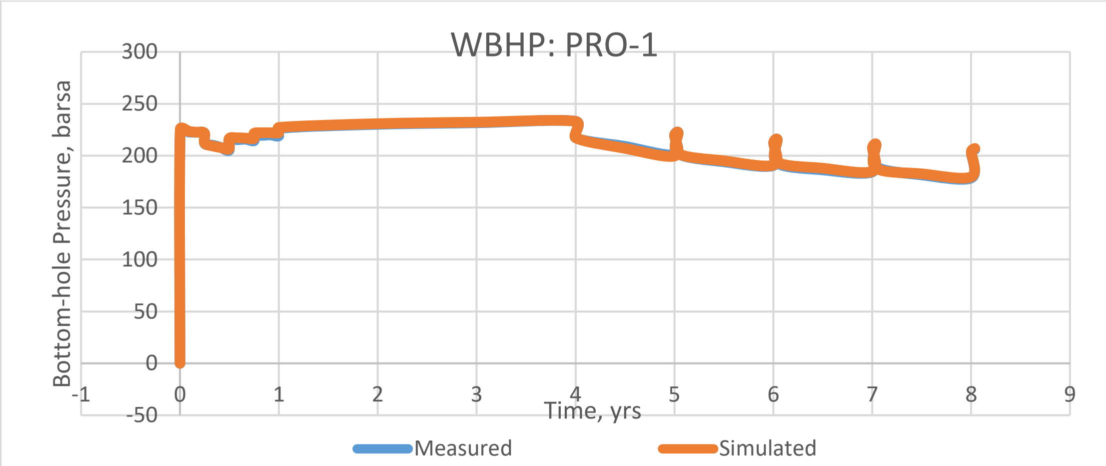

Fig. 6. Well bottom hole flowing pressure for Well 1

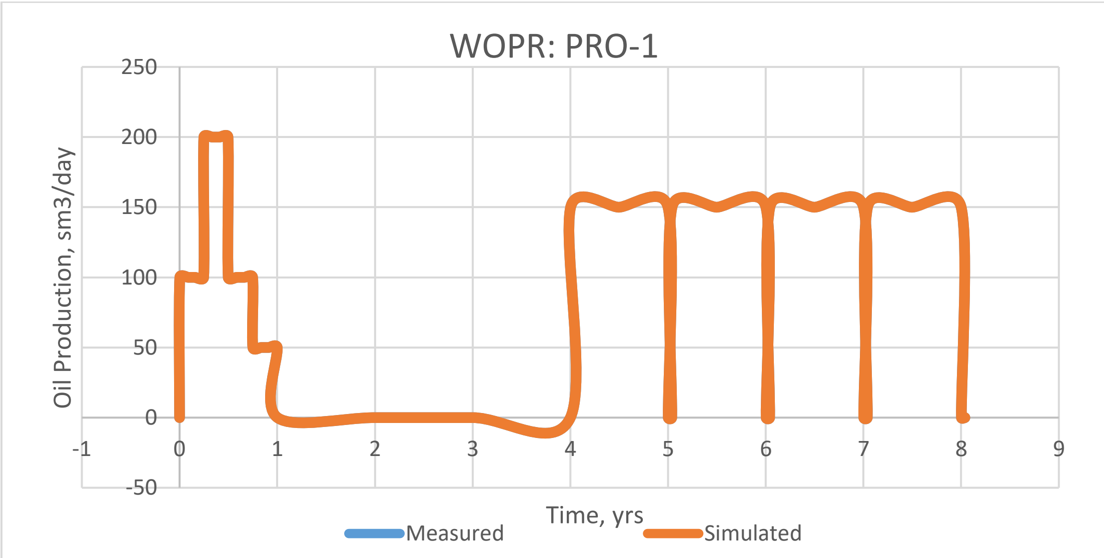

Fig. 7. Oil production rate for Well 1

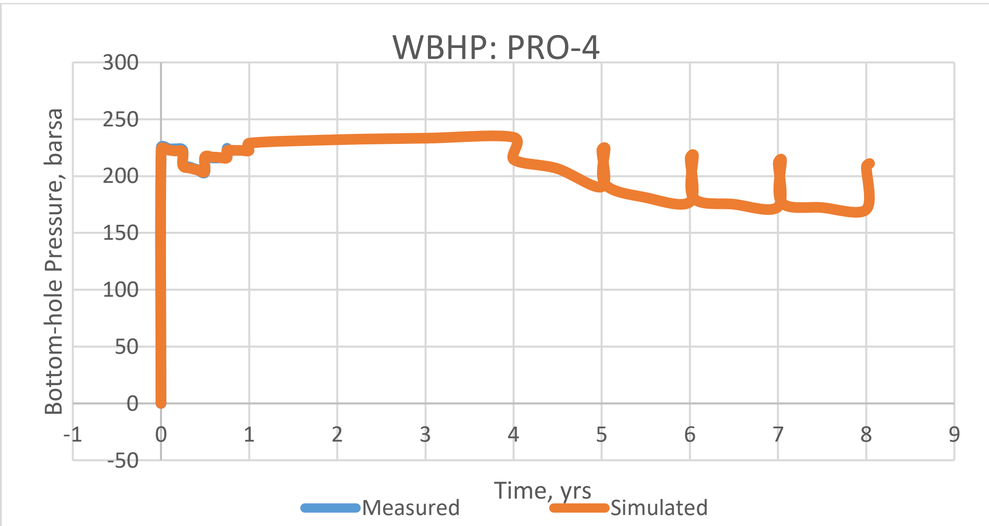

Fig. 8. Well bottom hole flowing pressure for Well 4

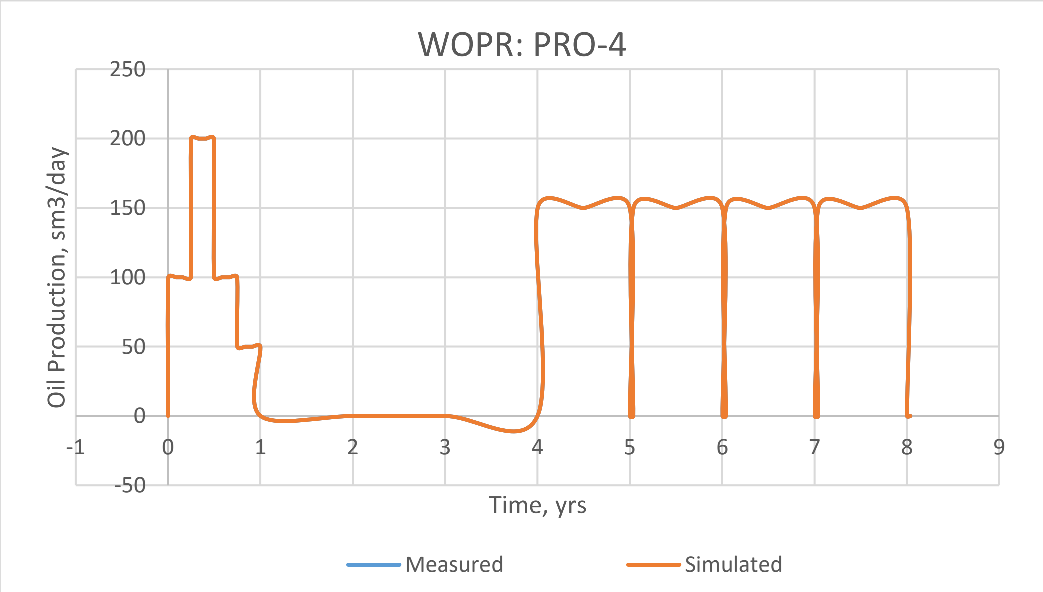

Fig. 9. Oil production rate for Well 4

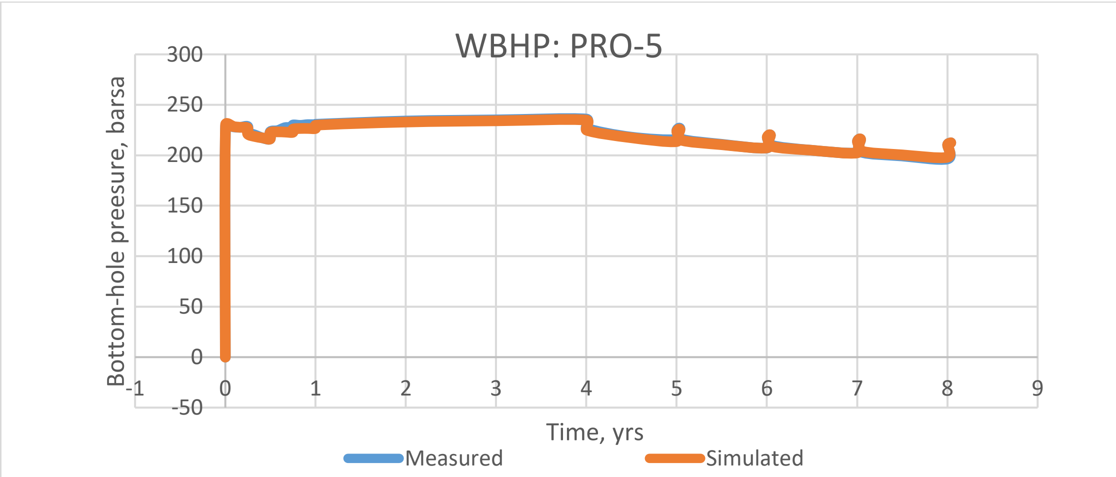

Fig. 10. Well bottom hole flowing pressure for Well 5

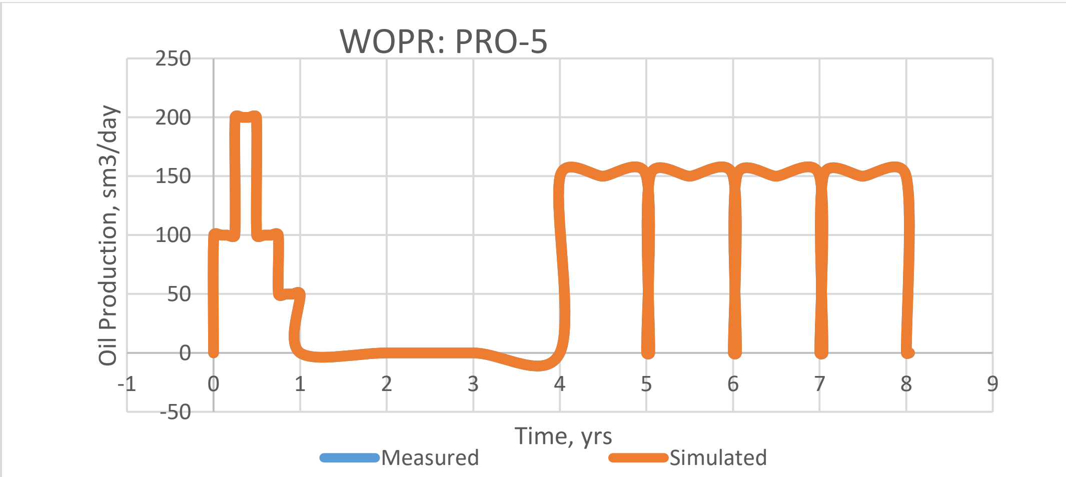

Fig. 11. Oil production rate for Well 5

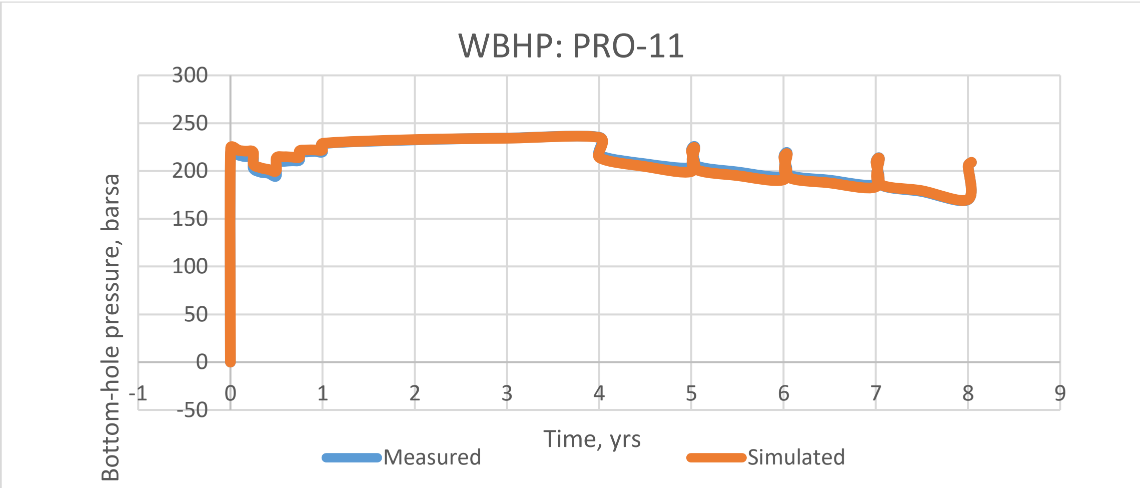

Fig. 12. Well bottom hole flowing pressure for Well 11

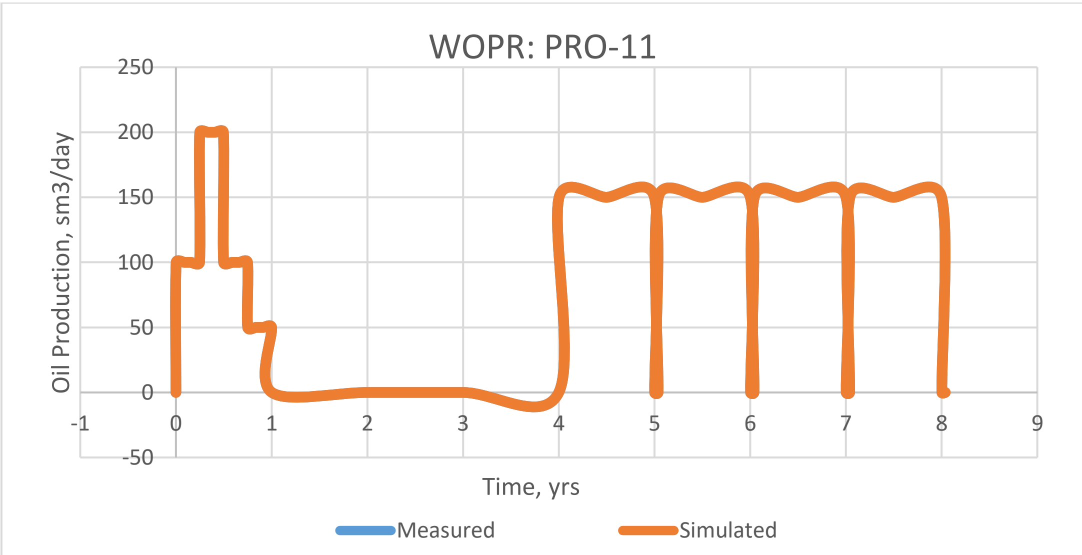

Fig. 13. Oil production rate for Well 11

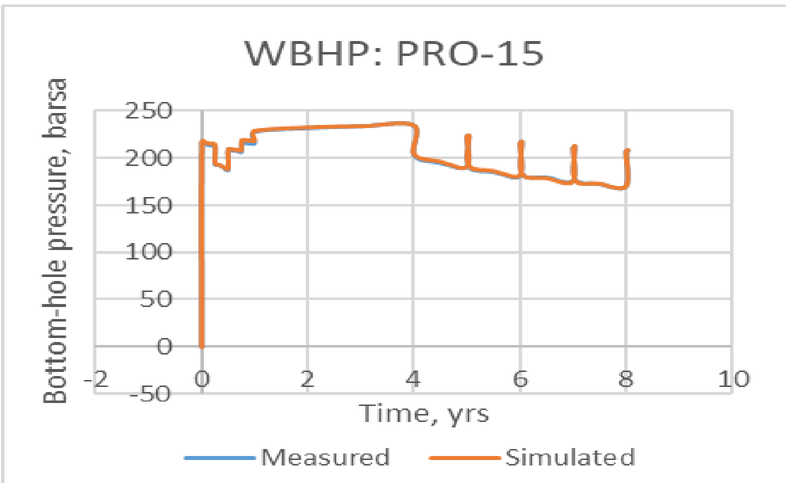

Fig. 14. Well bottom hole flowing pressure for Well 15

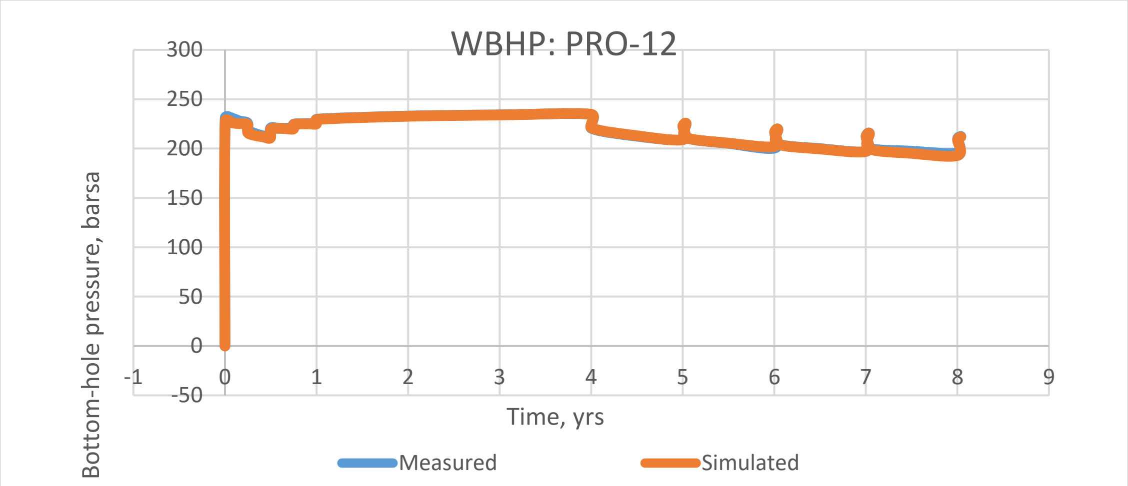

Fig. 15. Well bottom hole flowing pressure for Well 12

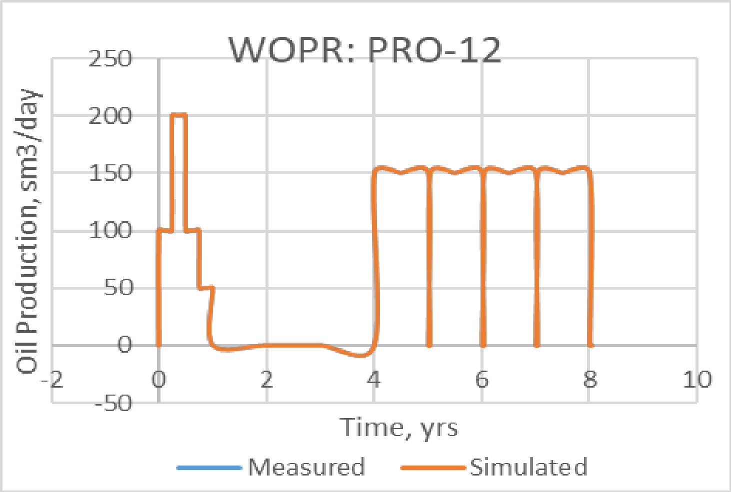

Fig. 16. Oil production rate for Well 12

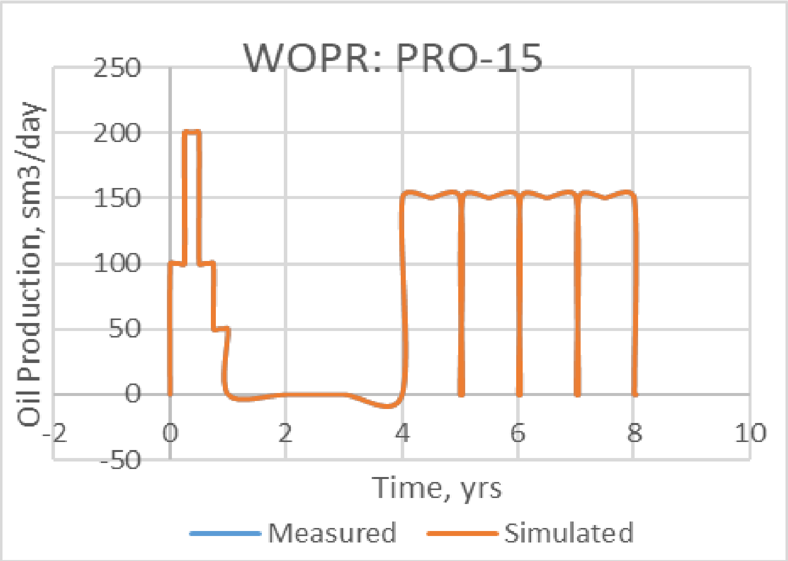

Fig. 17. Oil production rate for Well 15

Table 1. ABC algorithm parameters

Stopping criteria	100 cycles
Objective function	3.2369

Optimization in this procedure involves the minimization of the defined objective function. The objective function essentially measures the difference between the generated simulated data at a given permeability and the data in the reservoir history. For each permeability distribution generated by the algorithm, simulation is run and the objective function calculated. This continues until a predefined stopping condition is reached.

The result gotten is summarized in Table 1.

It can be seen from Figs. 6 to 17 that the history of the reservoir is very closely matched in this study. Very slight insignificant variations are seen for the bottom hole flowing pressures of all the wells excepts well 4.

This shows the method used to be very accurate.

5. CONCLUSION

History matching aims at fine-tuning the parameters used in building a reservoir model to closely match that of the real field. In this study, a very promising novel optimization algorithm has been employed to history a well-known reservoir model namely the PUNQ-S3 model.

Results obtained proves the algorithm used to be a very efficient optimization tool as the data used as the history of the study is nearly equaled by the optimization tool.

We therefore conclude that the ABC algorithm be employed in performing tasks that demand high degree of accuracy.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Oliver DS, Albert CR, Liu N. Inverse theory for Petroleum Reservoir Characterization and History Matching. Cambridge University Press, Cambridge; 2008.
2. Zhang X, Hou J, Wang D, Mu T, Wu J, Lu X. An automatic history matching method of reservoir numerical simulation based on improved genetic algorithm Elsevier. 2012; 29:3924-39.
3. Xavier CR, Elisa Portes dos Santos, Vincius da Fonseca Vieira, Rodrigo Weber dos Santos. Genetic algorithm for the history matching problem Elsevier. 2013; 18:946-955.
4. Amorim EP, Goldfield P, Dickstein F, Santos RW, Xavier CR. Automatic history matching in petroleum reservoir using the TSVD method. Springer. 2010;475-487.
5. Riazi SH, Zargar G, Baharimoghadam M, Moslemi B, Oliver Darani ES. Fractured reservoir history matching improved based on artificial intelligent KeAi; 2016.
6. Shahkarami A, Mohaghegh SD, Gholami V, Haghighat SA. Artificial Intelligence (AI) Assisted History Matching. SPE Western North American and Rocky Mountain Joint Meeting. Denver, Colorado: Society of Petroleum Engineers; 2015.
7. Slater G, Durrer E. Adjustment of reservoir simulation models to match field performance. Society of Petroleum Engineers Journal. 1971;295–305.
8. Eide A, Holden L, Reiso E, Aanonsen S. Automatic history matching by use of response surfaces and experimental design. 4th European Conference on the Mathematics of Oil Recovery. Roros, Norway; 1994.
9. Palatnic B, Zakirov L, Haugen S, van Roosmalen J. New approaches to multiple history matching. Seventh European Symposium on Improved Oil Recovery. Moscow, Russia; 1993.
10. Hajizadeh Y, Christie M, Demyanov V. Ant colony optimization for history matching SPE EUROPEC/EAGE annual conference and exhibition. Amsterdam, The Netherlands: Society of Petroleum Engineers; 2009.
11. Van Leeuwen PJ. Comment on Data assimilation using an ensemble Kalman filter technique. Monthly Weather Review. 1999;127(6):1374–1377.
12. Erbas D, Christie M. Effect of sampling strategies on prediction uncertainty estimation. SPE Reservoir Simulation Symposium. Houston, Texas; 2007.

13. Eberhart R, Yuhui Shi. Particle swarm optimization: Developments, applications and resources. Proceedings of the 2001 Congress on Evolutionary Computation. Seoul, South Korea. 2001; 1:81–86.
14. Maucec MD. Streamline-based history matching and uncertainty, Markov-chain Monte Carlo study of an offshore oil field. SPE Annual Technical Conference and Exhibition. Anaheim, California. 20 SPE-173405-MS; 2007.
15. Mantica SC. Combining global and local optimization techniques for automatic history matching production and seismic Data. SPE Journal. 2002;123–130.
16. Afiakinye MO. Well Placement Optimization Using Modified Artificial Bee Colony Agorithm. University of Uyo, Uyo; 2017.

Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle5.com/review-history/116622