HISTORY MATCHING IN A RESERVOIR MODEL USING AN AUTOMATIC APPROACH

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 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

Reservoir simulation is a powerfuland essential numerical simulation tool extensively used by reservoir engineers as an effective way to understand reservoir petrophysical parameters and fluid characteristics of reservoir fluids. It is mostly employed to predict reservoir behavior under different circumstances thereby supporting field development decisions.

One of the main goals of the reservoir model is its ability to predict the behavior of a reservoir. Several difficulties arise during the validation of a model, since most of the oil reservoirs are inconveniently buried beneath thousands of feet of overburden. Direct observations of the reservoir are available only at well locations that are often hundreds of meters (Oliver *et. al.*, 2008). The actual reservoir in reality is a complex nonlinear system whose behavior is rarely fully known with precision (Zhang *et. al.*, 2012).The computational models depend on many parameters and features of the reservoir and the prediction’s performance of a model depends on good estimations of some physical properties, such as the permeability distribution of the reservoir (Xavier *et. al., 2013)*. A way of estimating these properties is through the history matching process.

History matching is the process of adjusting uncertain reservoir parameters until an acceptable match with the measured production data is obtained. This process consists of the inverse problem of estimating reservoir properties through matching simulated data to reservoir history, which are available in reservoirs that are operating for some time (Amorim *et. al.,* 2010). The process helps in the tuning of reservoir parameters used in simulation to match the actual data in the real reservoir.

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.,* (2016), 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 Figure1.

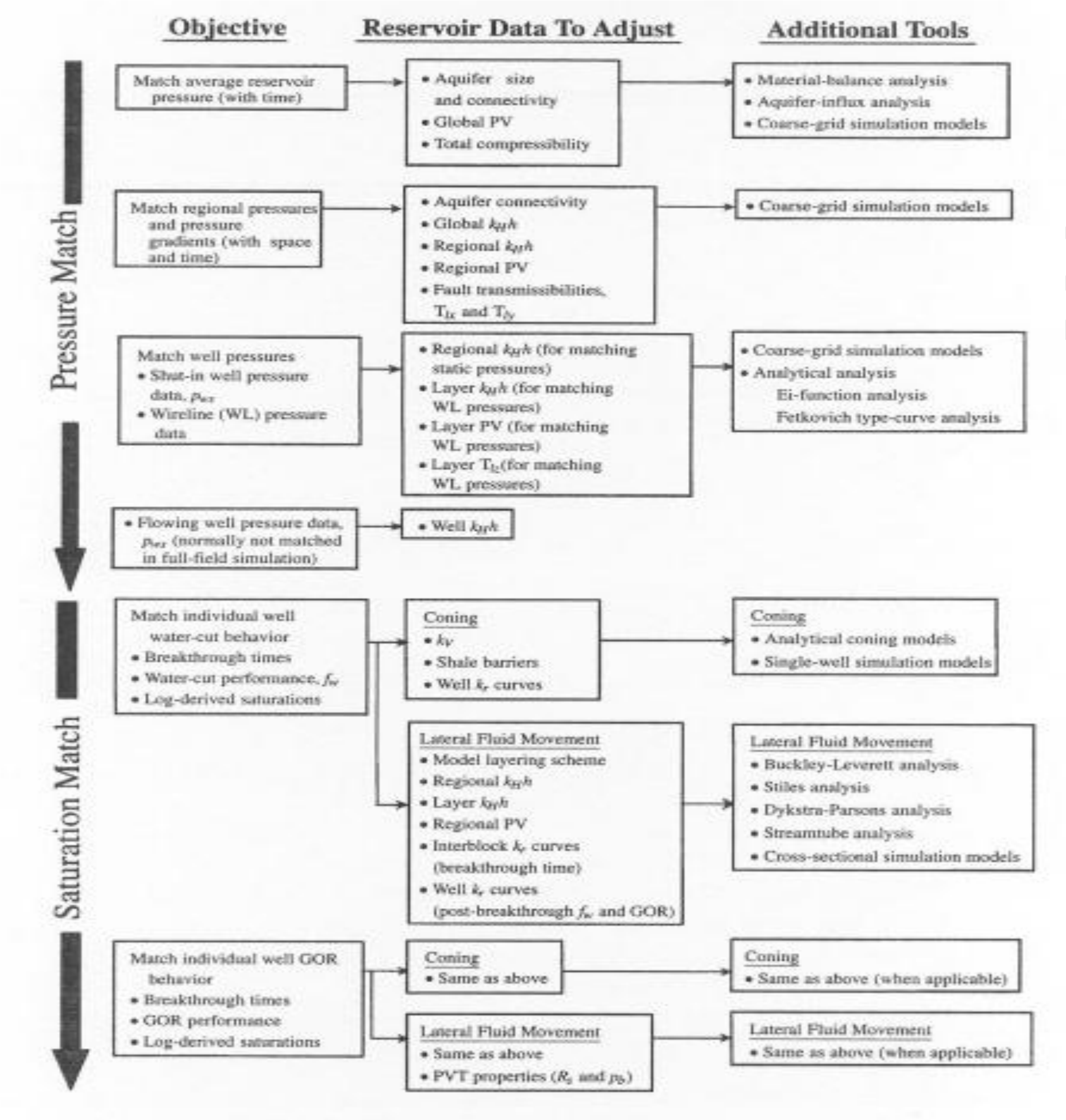


Fig.1 General strategy for history matching

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.

This method of history matching was very cumbersome, time consuming and largely ineffective due to the large amount of data typically used. Complexity and insufficient knowledge of reservoir characteristics makes this process time consuming with high computational cost. It is difficult to adjust the parameters to obtain the match due to the large number of reservoir parameters. Thus, there is necessity to use artificial intelligence (AI) which is a robust method that will perform the task automatically in a more accurate but cost-effective manner.

The methods used presently are the automatic methods which eliminates the problems posed by the manual method. It consists of using optimization algorithm(s) to minimize an objective function which measures the mismatch between the observed reservoir performance and simulation results until an acceptable difference is reached.

Automatic methods used in the Petroleum industry are summarized as follows:

- Deterministic methods; (gradient based) such as Descent method and Direct Pattern Searching are used in this category. The main challenge in these methods is the calculation of the gradients for which two distinct methods are often used namely; Finite Difference (FD) method and Adjoint-based method.
- Stochastic algorithms; including Simulated Annealing (SA), Particle Swarm Optimization (PSO), Simultaneous Perturbation Stochastic Approximation (SPSA), and some evolutionary algorithms such as; Genetic Algorithm (GA), Evolutionary Strategy (ES) and, of course, our ABC.
- A hybrid method; which is a combination of deterministic and stochastic methods, sometimes combined with helper methods and proxies (example, kriging proxy, etc.).

This study will research only on the Artificial Bee Colony (ABC) algorithm.

of automatic methods especially those backed by artificial intelligence.

The importance of history matching cannot be overemphasized. If the reservoir model used in simulation closely matches the real reservoir, then results obtained from using the model can be trusted to a high degree and field development decisions can conveniently be carried out using this result without much fear or worry. History matching on the whole improves the predictive power of simulation models by fine tuning the parameters used in the model to match that of the real-world reservoir, thus making the simulation model more effective and accurate.

Optimization algorithms help to automate the otherwise cumbersome process making it iterative, more efficient, and accurate.

2.0 LITERATURE REVIEW

History matching is no new field of study in the industry but it is a necessity in other to update the properties of the model hitherto used to closely match that of the reservoir by matching data obtained from the field to that obtained from the model. As most of the data used initially in describing the model are uncertain, results obtained therefrom cannot be applied confidently on the field. Tuning the uncertain parameters makes the model a better copy of the real field and results obtained therefrom can be trusted.

During the last two decades there has been efforts to improve automatic history matching in a way that could be applicable in the real world (Shahkarami *et. al.,* 2015). Despite all the attempts, due to increasing rate of complexity and resolution in the reservoir models, there is still hesitation about the practicality and potential of these methods to handle highly complicated real reservoir models. This makes automatic history matching still a challenging and hot research topic.

The earliest studies in the field of history matching started in the 1960's (Slater and Durrer 1971). The study mainly was based on proposing mathematical reservoir models and then calibrating these models using actual data. An important introduction in 1990s was using experimental design to develop response surfaces to replace the reservoir simulation in history matching workflow (Eide, *et. al.,* 1994).

During the years after early 90's, significant works were done to move history matching from a labor-intensive engineer-based framework to a fully or semi-fully automated approach (Palatnic, *et. al.,* 1993). Gradient optimization methods were used for history matching in late 1960's (Slater and Durrer 1971).

Gao *et. al.,* (2004) for the first time suggested the idea of combining the Simultaneous Perturbation Stochastic Approximation (SPSA) method with a simulator to perform automatic history match of multiphase flow production data.

Hajizadeh *et. al.,* (2009) introduced a stochastic approach for automatic history matching based on a continuous Ant Colony Optimization (ACO) algorithm. Other stochastic algorithms have been examined in this area. Evolutionary algorithms have gained popularity as a standard optimization approach in history matching. These algorithms are generally inspired by the evolution theory.

There have been many examples of application of these algorithms in history matching. Some of the methods successfully applied are; the ensemble Kalman filter (Van Leeuwen, 1999), Genetic Algorithms (Erbas, *et. al.,* 2007), Particle Swarm Optimization (PSO) (Eberhart, *et. al.,* 2001), (Rwechungura, *et. al.,* 2011) Ant Colony Optimization (ACO) algorithm (Hajizadeh, *et. al.,* 2011), Markov chain Monte Carlo (Maucec, 2007), and Chaotic Optimization (Mantica, 2002).

Increased complexity and simulation time of reservoir models has created a bottleneck for history matching workflows. This is particularly true for history matching workflows that employ a form of population-based sampling algorithms.

These algorithms, depending on the number of uncertainty parameters, require hundreds to thousands simulation calls to converge to optimal regions and find history-matched solutions. This requirement has created well-known barriers for the application of stochastic population-based methods for real-life history matching and uncertainty quantification problems. At the same time, the limitation has motivated an active area of research to reduce the simulation time of reservoir models.

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.

Proxy models as an inexpensive approximation of high computational cost full field simulation models are frequently used in different areas of engineering. By increasing the time and cost required to run the reservoir simulation models, proxy models appeared in petroleum engineering. They are fast and relatively easy to develop. However, due to the practicality concerns there is a long way to completely surpass full field reservoir simulation models in reservoir management plans. Response surface models and reduced order models are the most famous types of proxy models used in petroleum engineering. Reduced order modeling aims to transfer the high dimensional models into a meaningful representation of reduced dimensionality. They have been applied in many areas including petroleum engineering. 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 approach recently going through fast development is data-driven modeling. Data-driven modeling is based on analyzing the available data about a system using machine learning methods. This approach particularly finds the connections between different components of the system without any explicit knowledge of the physical behavior of these components. Statistical methods, application of artificial neural networks, and fuzzy logic are examples of data-driven modeling approaches. A relatively new types of data-driven models utilized in reservoir modeling and simulation are Surrogate Reservoir Models (SRMs). SRMs are built based on artificial intelligence and data mining techniques and are meant to replace or complement the reservoir simulation models.

Shahkarami *et. al.,* (2015) presented a study which was aimed at examining the application of pattern recognition technologies to improve the time and efforts required for completing a successful history matching project. The pattern recognition capabilities of artificial intelligence and data mining techniques are used to develop a Surrogate Reservoir Model (SRM) and used to perform the assisted history matching process. The results of their study proved the ability of SRMs in assisting history matching process using population-based sampling algorithms and other computationally intensive operations in reservoir management workflows.

Amorim *et. al.,* (2010) presented a method for the resolution of a history matching problem that aims to estimate the permeability field of a reservoir using the pressure and the flow rate observed in the wells. This reservoir simulation was based on a two-phase incompressible flow model. The method combines the Truncated Singular Value Decomposition (TSVD) and the Gauss-Newton algorithms. According to them, the number of parameters to be estimated depended on how many grid blocks were used to discretize the reservoir. In general, this number was large and the inverse problem ill-posed. The TSVD method regularizes the problem and decreases considerably the computational effort necessary to solve it. To compute the TSVD they used the Lanczos method combined with numerical implementations of the derivative and of the adjoint formulation of the problem.

Zhang *et. al.,* (2012) used an improved GA to perform automatic history matching in a numerical reservoir model. In the study, total water cut of the blocks was used as the observed data while the parameters to be modified were the relative permeability curve, average permeability values of interlayers and permeability coefficient of variation in layers. The results of their study which showed the method to be strongly reliable and having a fast convergence speed was validated using SZ 36-1 typical reservoir of Bohai oilfield. They therefore concluded that the method is effective.

Xavier *et. al.,* (2013) 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.

Riazi *et. al.,* (2016) performed history matching on a fractured reservoir model using Least Square Support Vector Machine (LSSVM) as a proxy model with the Particle Swarm Optimization (PSO) algorithm and the Imperialist Competitive Algorithm (ICA) as the base optimization algorithms. In their approach, the proxy model is made to model the history match objective function (mismatch values) based on the history data of the field. This model is then used to minimize the objective function through PSO and ICA. The obtained results show that the procedure is effective for history matching process due to its robust dependability and fast convergence speed. 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 (2017) used a modified version of the ABC algorithm to optimize well placement. Modifications to the base ABC algorithm include; inhibition of visitation of already visited points, use of a more efficient method to handle the way out-of-boundary points were pushed back into the feasible search region. Results obtained from the study shows the modified algorithm to be better than the base 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

ABC is one of the swarm algorithms. The term swarm is used for an aggregation of animals like fishes, birds and insects such as bees, ants and termites. The individual agents of these swarm behave without supervision and each of these agents has a stochastic behavior due to her perception in the neighborhood. The intelligence of the swarm lies in the networks of interactions among these simple agents, and between agents and the environment.

The initial solutions in ABC are usually randomly generated and improved upon in the course of the optimization. Equation (3) is used to generate the initial solutions.

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

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

Improvement to solutions is done around the neighborhood of the solution to be improved.

A predetermined number of cycles called limit within which a randomly generated solution is to be improved is usually specified. Once this limit is reached, the solution is abandoned and a new solution is randomly generated to replace the old solution and the process is repeated until a stopping criteria as specified by the user is met.

ABC is relatively simple to implement as it makes use of very few parameters in performing its work. Basically, three kinds of bees are defined in the algorithm thus:

i. Employed Foragers: These bees are associated with specific food sources. Food sources are defined as potential solutions to the problem under consideration. Their responsibility is to exploit a particular food source, store information; position and nectar quality (fitness), about the food source and communicate same to other bees in the hive. Additional to this responsibility is the perturbation (generation of a trial solution around the neighborhood of the current food source) and the evaluation of the new (perturbed) food source's fitness value.

This is done using equation (4)

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

Here, $j \epsilon \ (1,2, \dots, \mathrm{D})$ and $\mathrm{k} \ \epsilon \ (1,2, \dots, \mathrm{SN})$ 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.

If the fitness of the perturbed solution is better than that from which it was formed, then the new solution replaces the old one in the memory of the employed forager else the old food source is perturbed until the specified limit is reached after which it is abandoned and a new source generated. This is a greedy-selection scheme.

i. Onlooker Bees: These bees observe the dance of the employed foragers around a food source and uses the information shared by the employed foragers to calculate the probability of choosing that particular food source and then compares it to a randomly generated number from 0 to 1. The probability values P_i for the solutions x_i are calculated by means of their fitness values using equation (5)

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

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:

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

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

A food source is chosen only when its fitness is greater than the randomly generated number, this food source is memorized and a new trial solution is generated using equation (4) by the onlooker bee around the neighborhood of this best solution. The greedy-selection scheme is also applied by the onlooker bee if the fitness of this new solution is better than that of the previous one in her memory.

ii. 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

In the process of reservoir history matching, there are so many uncertainties for reservoir parameters that it is time-consuming to modify large amounts of model parameters for a short time and easy to result in non-convergence in automatic history matching (Goda *et. al.* 2010). To improve fitting accuracy and reduce time for optimization, model parameters to be optimized in this study are the permeability vector denoted by K.

3.2 Theory

History matching is a well-known inverse problem in the oil industry (Oliver et.al., 2008). In the forward problem, the physical properties of the reservoir are known and a simulator is used to calculate the production behavior of the reservoir. In the inverse problem (history matching), the goal is to estimate plausible reservoir physical properties, given the observed production data of a real reservoir. Since physical properties of the reservoir cannot be directly measured in all extensions of the reservoir, the history matching process is used to estimate such properties. The estimated properties are used as the simulator's parameters with the goal of predicting the reservoir behavior under different production scenarios.

In this work, the inverse problem proposed aims to estimate the absolute permeability field of a reservoir by history-matching its production data, which is given by the oil rate and bottom-hole pressure measured at well locations from time to time. We denote by *K* the permeability to be determined, by *S(K)* the simulated data given the parameter *K* and by *O* the observed data. The problem consists of searching *K* that minimizes the least square formulation

$$f(K) = \|S(K) - O\|^2 \tag{5}$$

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

$$\min f(K) \tag{6}$$

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

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

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

MATLAB (an abbreviation of "MATrix LABoratory") is a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks. It allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages (Wikipedia, 2022)

MATLAB is used by millions of engineers and scientists to analyze data, develop algorithms, and create models. It combines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly. It includes the Live Editor for creating scripts that combine code, output, and formatted text in 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

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you tolearn and applyspecialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

3.4 The MATLAB System

The MATLAB system consists of five main parts:

3.4.1 The MATLAB language.

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

3.4.2 The MATLAB working environment.

This is the set of tools and facilities that you work with as the MATLAB user or programmer. It includes facilities for managing the variables in your workspace and importing and exporting data. It also includes tools for developing, managing, debugging, and profiling M-files, MATLAB's applications.

3.4.3 Handle Graphics.

This is the MATLAB graphics system. It includes high-level commands for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands that allow you to fully customize the appearance of graphics as well as to build complete Graphical User Interfaces on your MATLAB applications.

3.4.4 The MATLAB mathematical function library.

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

3.4.5 The MATLAB Application Program Interface (API).

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It include facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

3.4.6 Eclipse

The ECLIPSE industry-reference simulator offers the industry’s most complete and robust set of numerical solutions for fast and accurate prediction of dynamic behavior for all types of reservoirs and development schemes. The ECLIPSE simulator has been the benchmark for commercial reservoir simulation for more than 25 years thanks to its extensive capabilities, robustness, speed, parallel scalability, and unmatched platform coverage.

With over 30 years of continuous development and innovation, the ECLIPSE simulator is the most feature-rich and comprehensive reservoir simulator on the market—covering the entire spectrum of reservoir models, including black oil, compositional, thermal finite-volume, and streamline simulation. By choosing from a wide range of add-on options—such as local grid refinements, coalbed methane, gas field operations, advanced wells, reservoir coupling, and surface networks—simulator capabilities can be tailored to meet your needs, enhancing your reservoir modeling capabilities.

Used at over 800 sites in 70 countries, the ECLIPSE simulator draws upon an unrivalled level of reservoir engineering expertise across the entire industry. As well as being the de facto simulation standard in the petroleum industry, the ECLIPSE simulator is used extensively by academia, regulatory authorities, and petroleum financial planners. Evidenced by citations in over 1,500 SPE technical papers, the ECLIPSE simulator is widely acknowledged as the industry’s leading reservoir simulator.

3.5 Case Study

The model used for this study 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.

PUNQ-S3 reservoir simulation model, as described by Floris [Floris et al. 2001], is a 5-layer model. The top depth of PUNQS3 reservoir is 2430 m. It has a dip angle of about 1.5 degree and is bounded by a fault to the east and south and a relatively strong aquifer on the north and west provides a pressure support. Because of this pressure support, no injection wells have been drilled in this reservoir. There is also a small gas cap in the PUNQ-S3 reservoir model in layer 1. There is no well completed in this layer because of effect of free gas production on recovery of the reservoir. Six production wells are marked with black dots as it can be seen in Figure 1. These wells are located near the initial gas-oil contact. Producers 1 (PRO-1), 4 (PRO-4) and 12 (PRO-12) are perforated in layers 4 and 5. Producer 5 (PRO-5) and 11 (PRO-11) have been completed in layers 3 and 4 and producer 15 (PRO-15) has been perforated only in layer 4. PRO-4 has been completed near aquifer and water breakthrough has been observed in 7th year. Free gas production starts in 4th and 5th year in PRO-1 and PRO-4. PUNQ-S3 model contains 19×28×5 grid blocks, which about two third of the grid blocks (1761) are active. The grid blocks have equal 180 meter sides in x and y directions. The reservoir simulation case has been modeled with corner point geometry with Carter-Tracy aquifer type. Complete data set for this reservoir is available online [PUNQ 2010].

The PUNQ-S3 reservoir model is a popular benchmark model to test and compare the novel methods developed for history matching and uncertainty quantification. Many anthers have published the results of their research for PUNQ-S3 reservoir model. Soleng used a steady state genetic algorithm for history matching of this model [Soleng, 1999]. Manceau presented an integrated method for history matching and uncertainty analysis based on Fast Fourier Transform-Moving Average (FFTMA) and gradual deformation techniques and applied it to PUNQ-S3 case [Manceau et al. 2001]. Mantica coupled chaotic optimization with gradual deformation and applied it to the same reservoir model [Mantica et al. 2002]. Demyanov applied Neighbourhood Algorithm (NA) coupled with a geostatistical framework for history matching of PUNQ-S3 model [Demyanov 2004]. Gao applied two types of Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm to the problem of reservoir history matching and tested this on PUNQ-S3 reservoir [Gao et al. 2007].

3.6 Method

1. The permeability distribution of the field is the parameter to be matched.
2. To perform history matching using an optimization algorithm, an objective function must be defined. The objective function measures the mismatch (difference) between the simulated data generated by ECLIPSE 100 after each simulation and the reservoir history.
3. This function is used by the algorithm to determine the goodness of the match. The objective function in this study is as defined in Equation 9.

4. A new permeability is generated by the ABC algorithm based on the value of the objective function.
5. The generated permeability is used to run a simulation in ECLIPSE 100.
6. For each simulation run, data are generated. The data so generated are read by Matlab and used in calculating the objective function.
7. The calculated objective function is sent to the ABC algorithm which generates a new permeability value based on the value of the objective function and the cycle continues.
8. Optimization is complete when the objective function attains a predefined minimum value or a maximum number of optimization cycle as defined by the user is reached. This is the stopping condition.

4. RESULTS AND DISCUSSION

Petro-physical properties of the PUNQ-S3 model used is as shown in Figures2 –5.Values of the permeability of the reservoir model at the well locations are the variables adjusted in this study while the oil production rate and the well bottom-hole pressures are taken as the history of the model. These were accordingly matched using the ABC algorithm and results of the matching presented in figure 2 below.

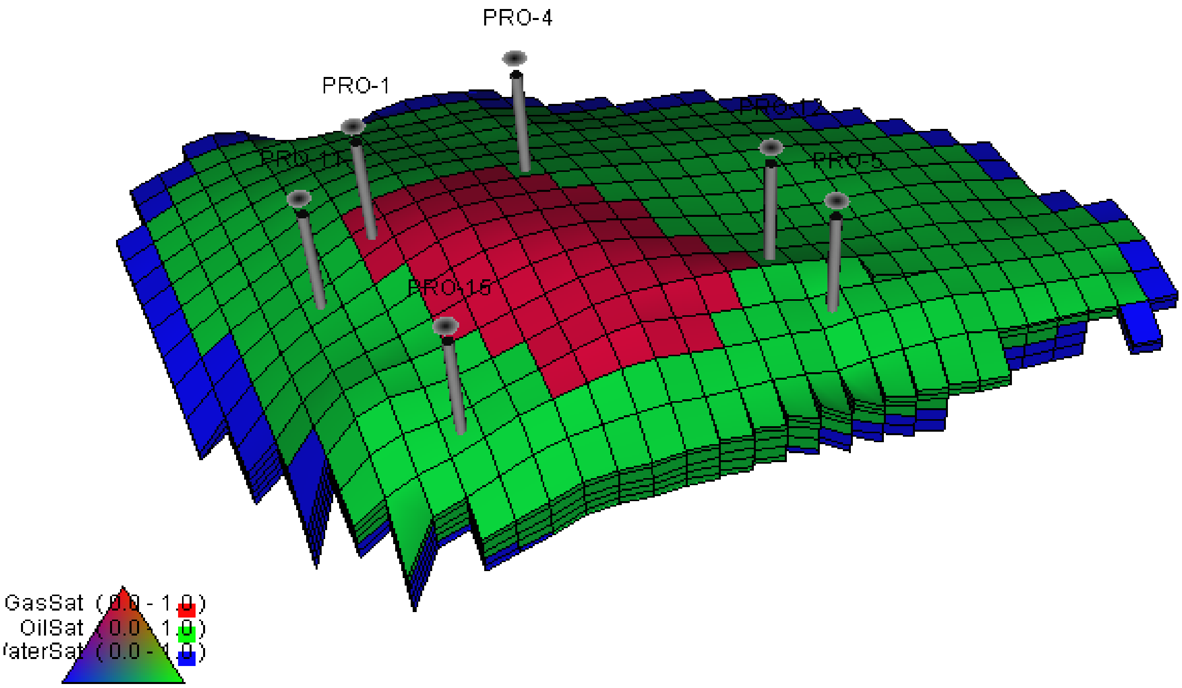


Fig. 2Fluid saturation distribution in the model

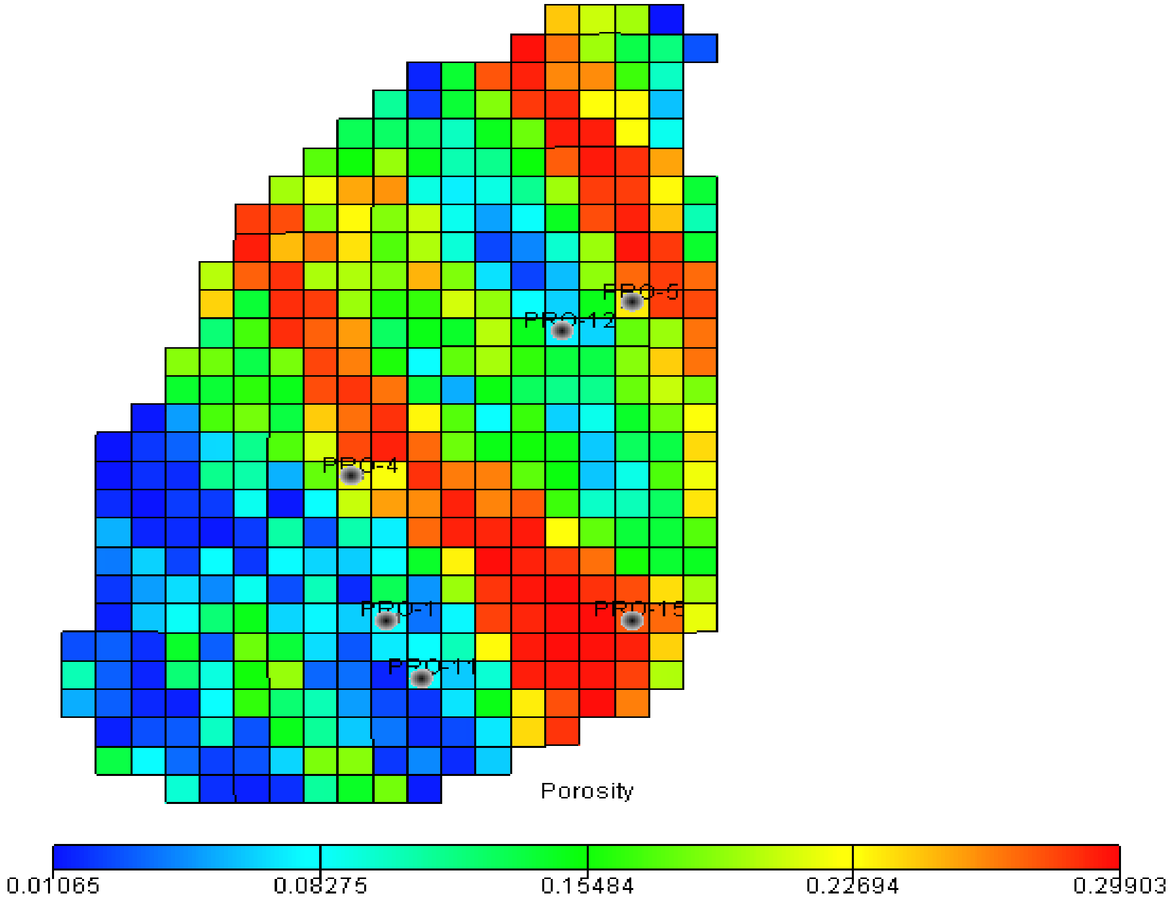


Fig. 3Porosity distribution in the model

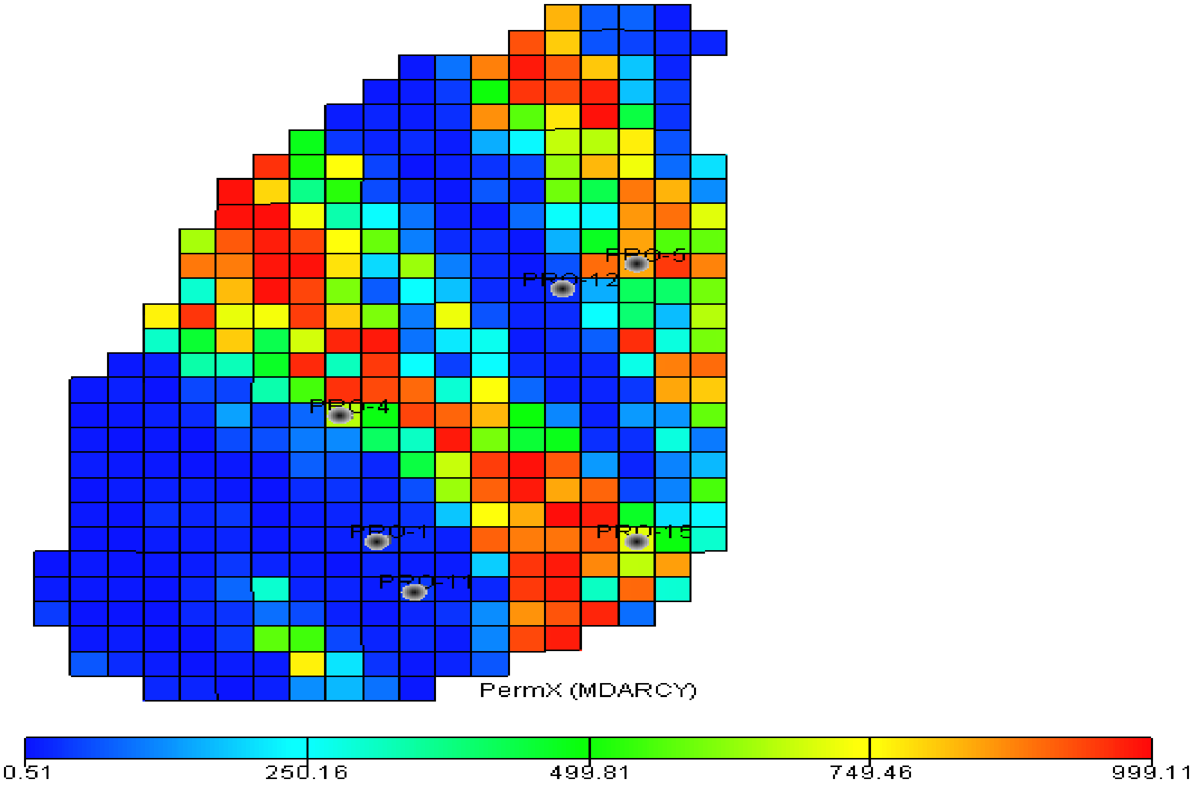


Fig. 4 Permeability distribution in the model

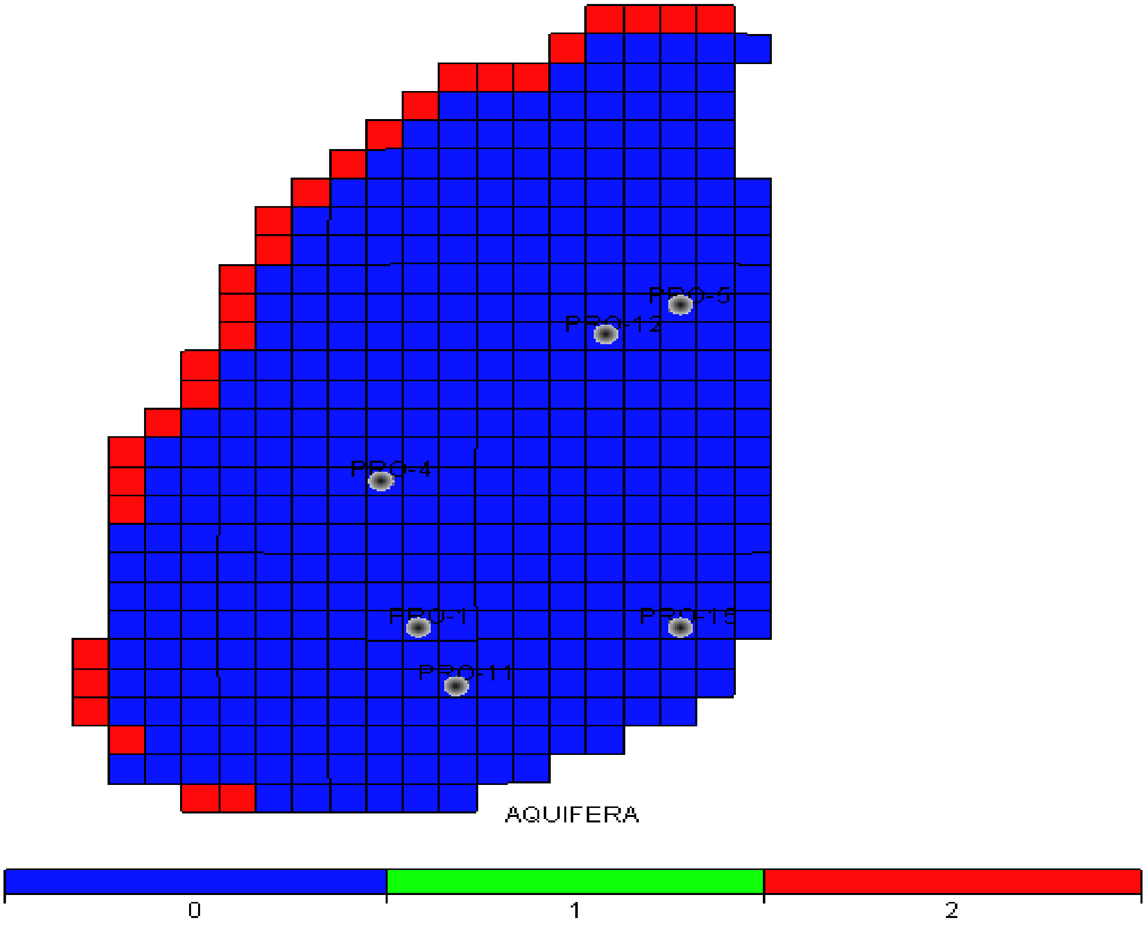


Fig. 5 Aquifer bound around the model

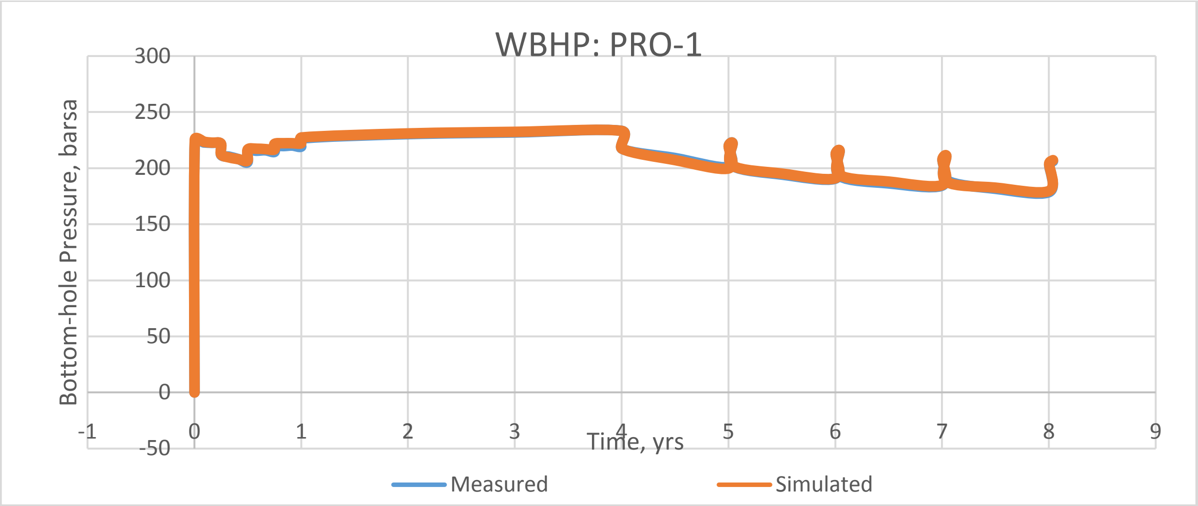


Fig. 6 Well bottom hole flowing pressure for Well 1

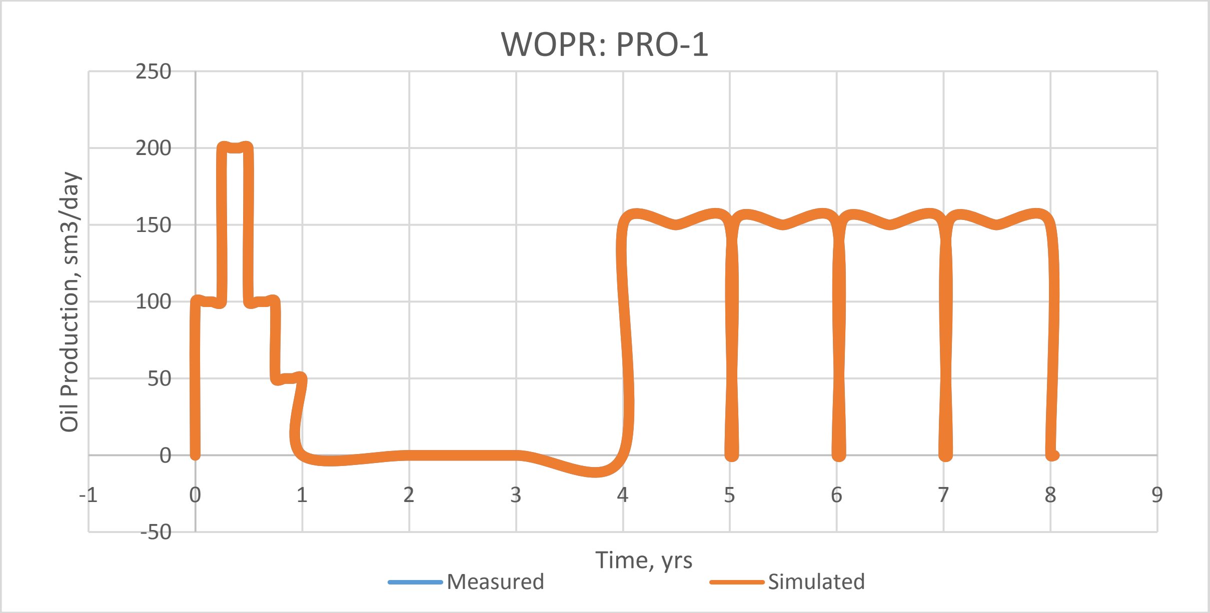
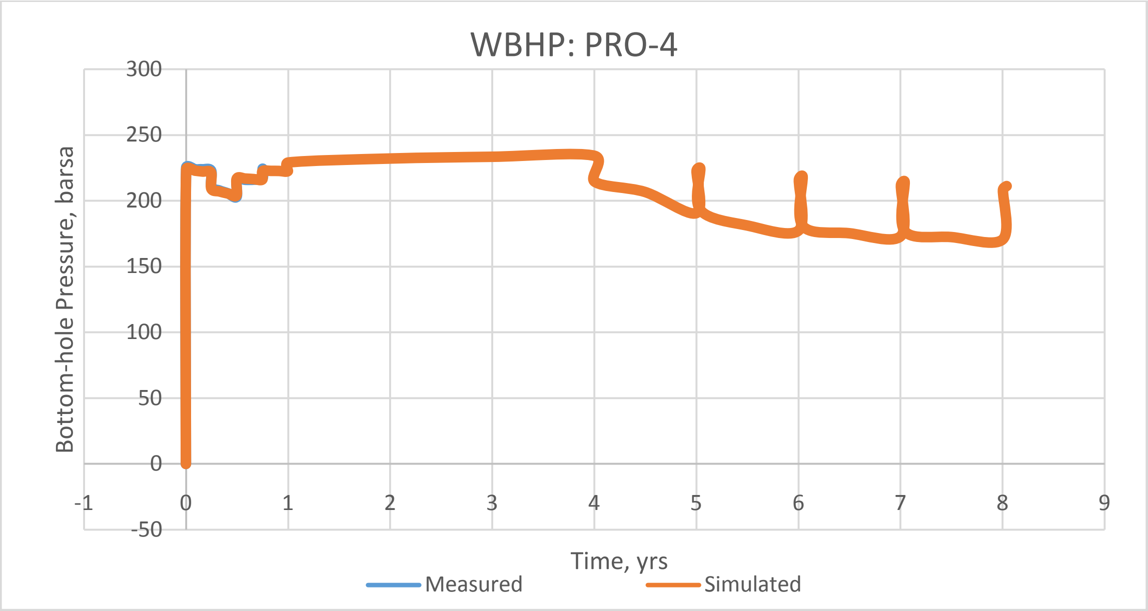


Fig. 7 Oil production rate for Well 1



\

Fig. 8 Well bottom hole flowing pressure for Well 4

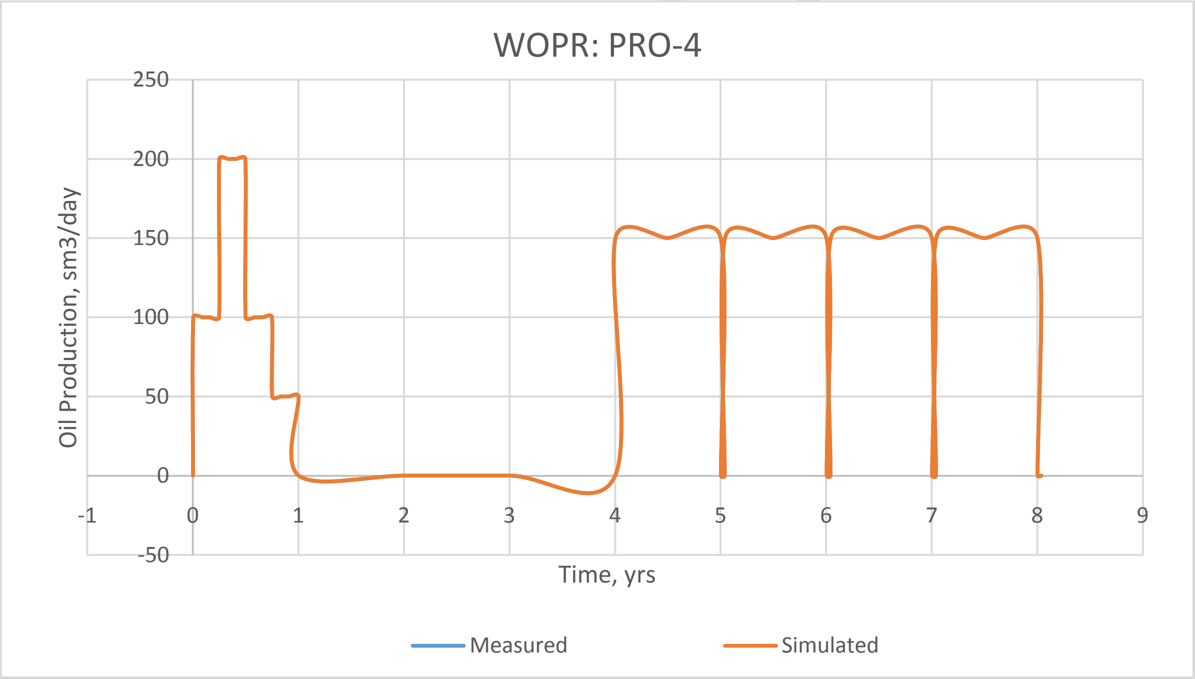


Fig. 9 Oil production rate for Well 4

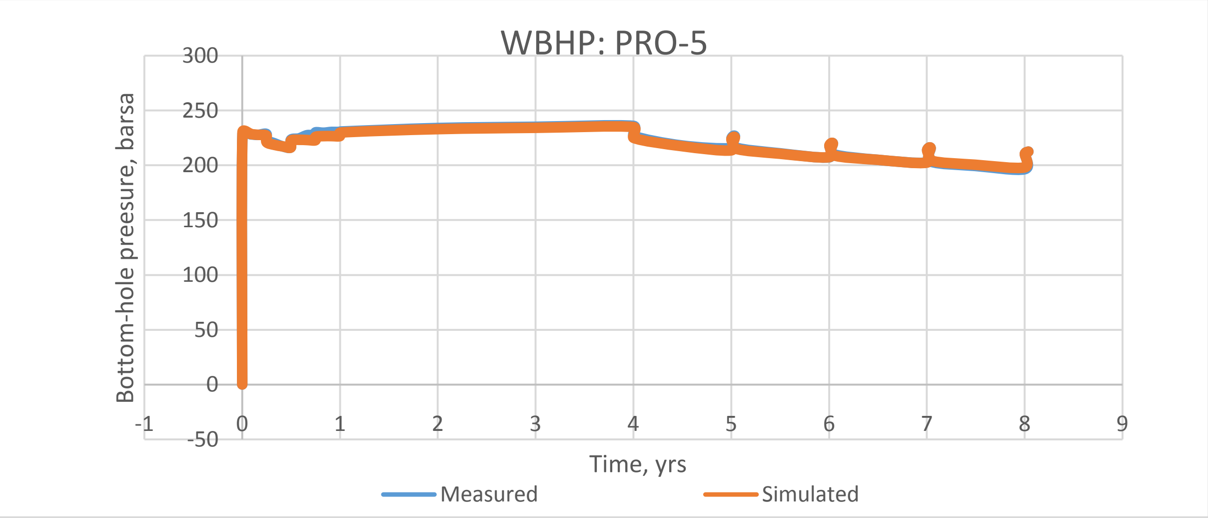


Fig. 10 Well bottom hole flowing pressure for Well 5

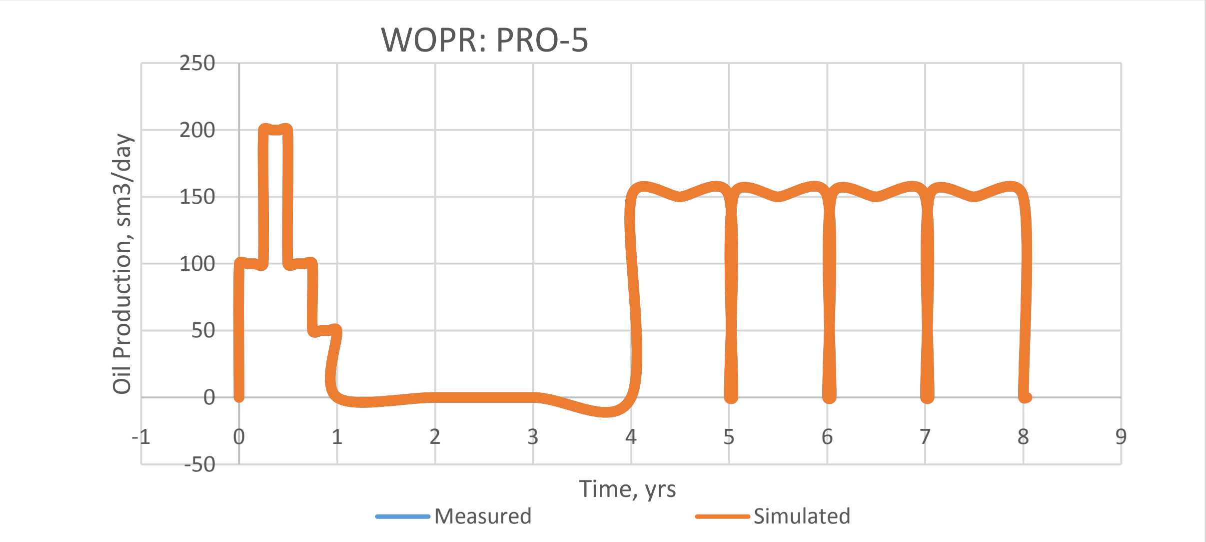


Fig. 11 Oil production rate for Well 5

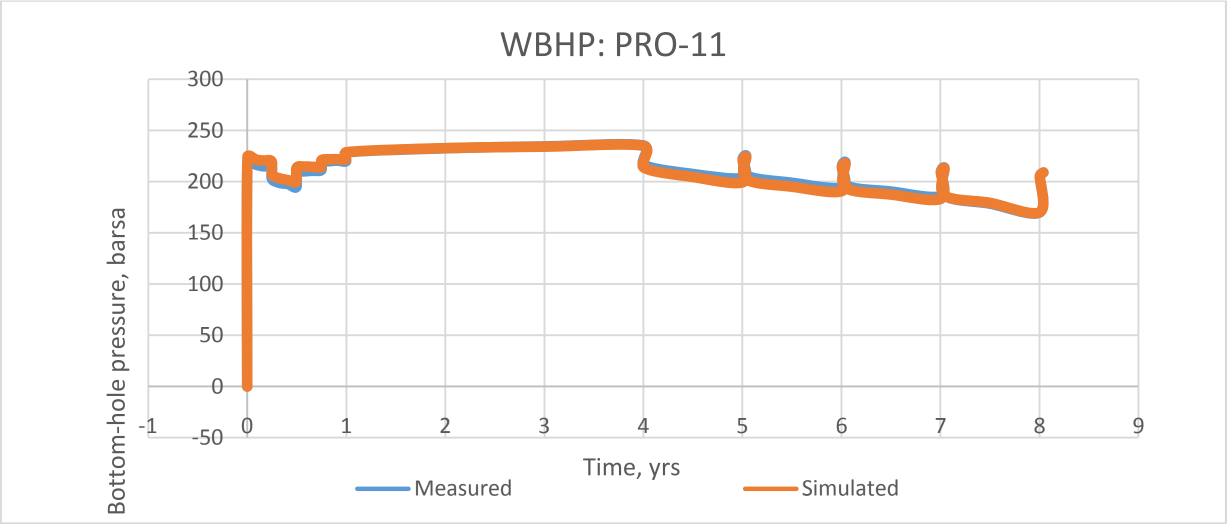


Fig. 12 Well bottom hole flowing pressure for Well 11

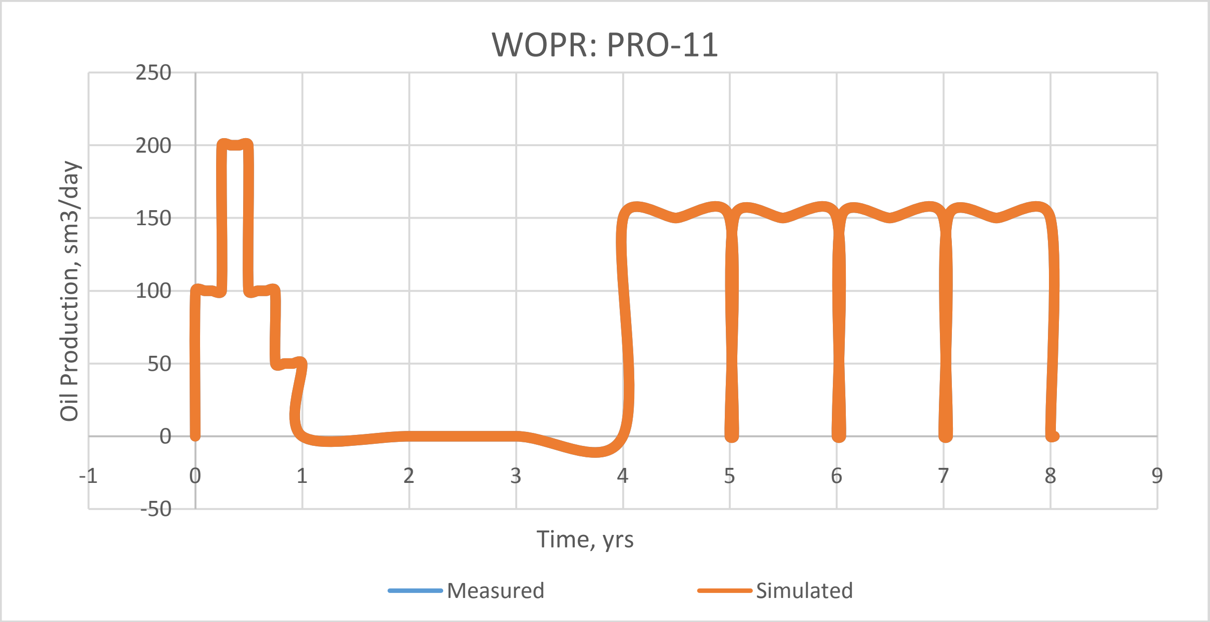


Fig. 13 Oil production rate for Well 11

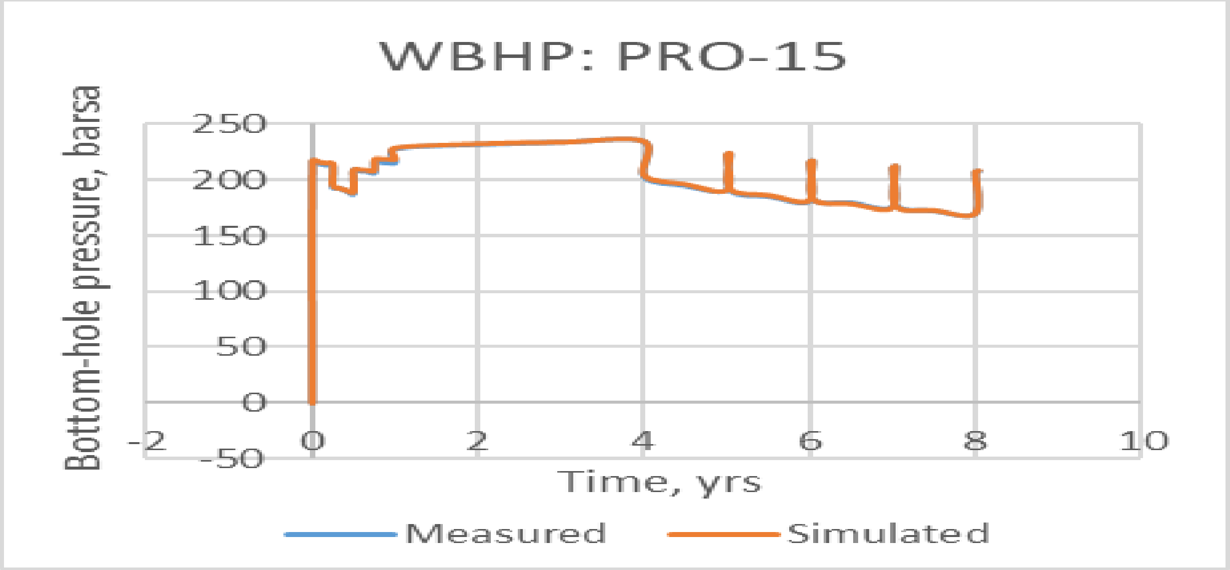


Fig. 14 Well bottom hole flowing pressure for Well 15

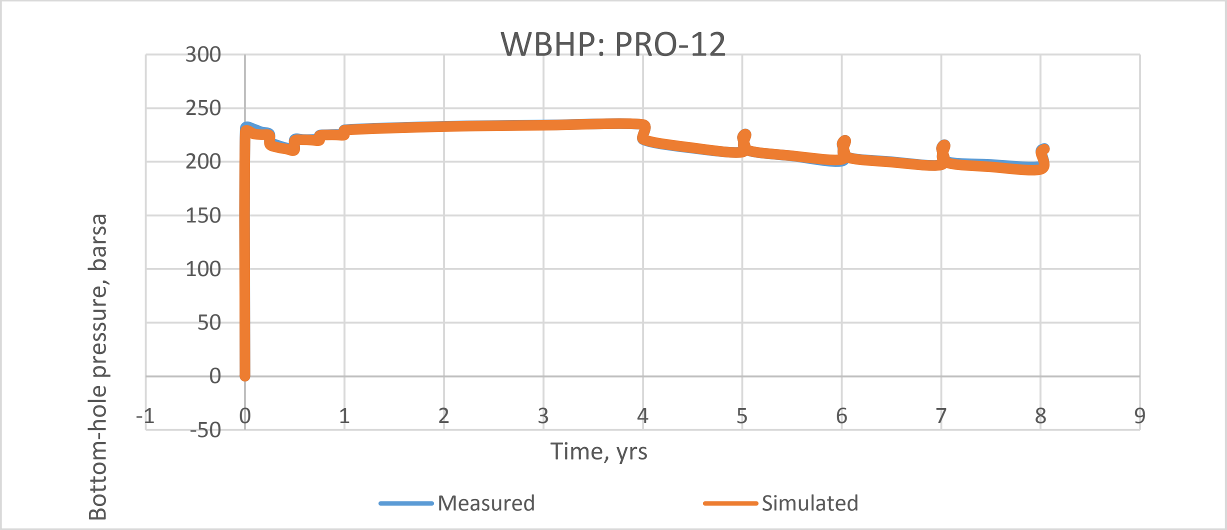


Fig. 15 Well bottom hole flowing pressure for Well 12

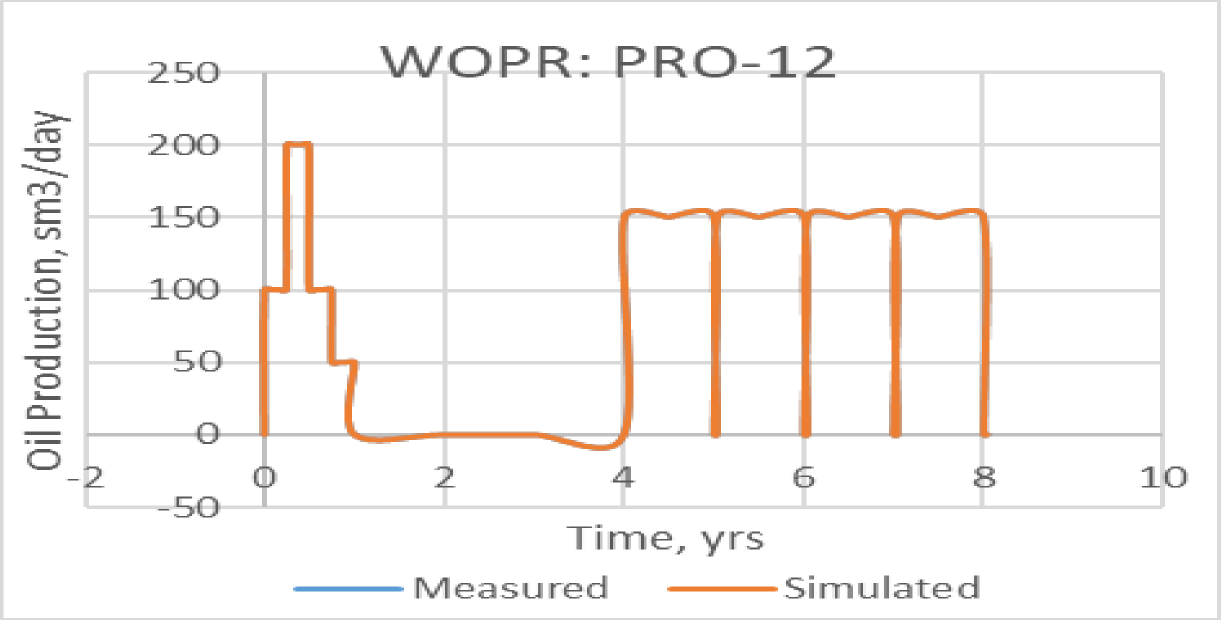


Fig. 16 Oil production rate for Well 12

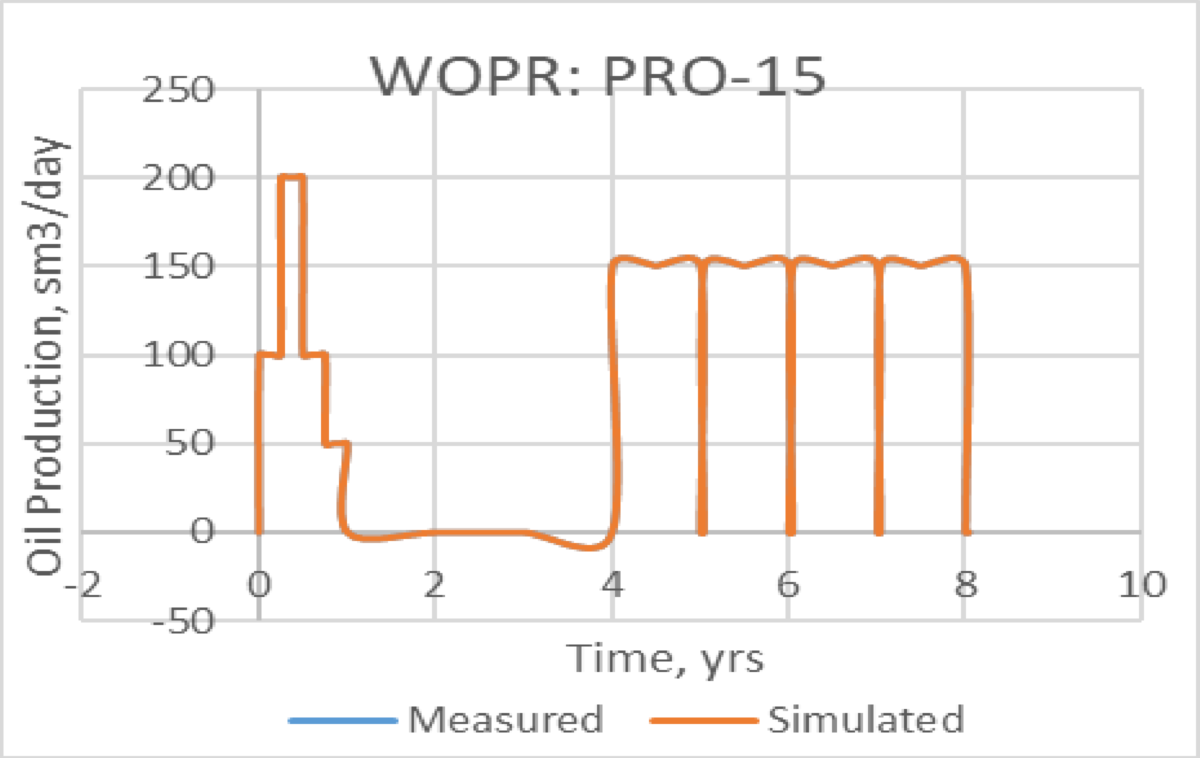


Fig. 17 Oil production rate for Well 15

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 figures 6to17 above 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.

Table 1 ABC algorithm parameters

Stopping criteria	100 cycles
Objective function	3.2369

5.0 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.

6.0 REFERENCES

Afiakinye, M. O. (2017) Well Placement Optimization Using Modified Artificial Bee Colony Agorithm. University of Uyo, Uyo.

Amorim, E. P., Goldfield, P., Dickstein, F., Santos, R. W., Xavier, C. R. (2010) Automatic History Matching in Petroleum Reservoir Using the TSVD Method, Springer pp. 475-487

Artificial Intelligence: https://www.britannica.com/technology/artificial-intelligence [retrieved 24 March, 2018]

Eberhart, R., and Yuhui Shi. (2001). Particle swarm optimization: developments, applications and resources. Proceedings of the 2001 Congress on Evolutionary Computation, **1**, pp. 81–86. Seoul, South Korea.

Eide, A., Holden, L., Reiso, E., and Aanonsen, S. (1994). Automatic History Matching by Use of Response Surfaces and Experimental Design. 4th European Conference on the Mathematics of Oil Recovery. Roros, Norway.

Erbas, D., and Christie, M. (2007). Effect of Sampling Strategies on Prediction Uncertainty Estimation. SPE Reservoir Simulation Symposium. Houston, Texas.

Gao, G., Zafari, M., and Reynolds, C. (2005). Quantifying Uncertainty for the PUNQ-S3 Problem in a Bayesian Setting with RML and EnKF. Reservoir Simulation Symposium. Houston, Texas: Society of Petroleum Engineers.

Goda HM, Behrenbruch P, Maier HR. (2007) Alternative modeling approaches for the estimation of irreducible water saturation: australian hydrocarbon basins. J. Pet. Sci. Eng., 57: 60-69.

Hajizadeh, Y. (2010). Ants Can Do History Matching. SPE Annual Technical Conference and Exhibition. Florence, Italy: Society of Petroleum Engineers.

Hajizadeh, Y., Christie, M., and Demyanov, V. (2009). Ant Colony Optimization for History Matching SPE EUROPEC/EAGE Annual Conference and Exhibition. Amsterdam, The Netherlands: Society of Petroleum Engineers

Hassan, R., Cohanim, B., Weck, O., Venter, G., (2005): A Copmarison of Particle Swarm Optimization and the Genetic Algorithm, in: Proceedings of the 1st AIAA Multidisciplinary Design Optimization Specialist Conference, 18e21 April, Austin, Texas, pp. 1e13.

Holland, J. Adaptation in natural and artificial systems (1992): An introductory analysis with applications to biology, control, and artificial intelligence. First edition: 1975 The University of Michigan.

Jahangiri, H. R. (2007). Production Optimization using Smart Well Technology with Differential Evolution Algorithm. Graduate Student Symposium, University of Southern California.

Kennedy, J., Eberhart, R., (1995) Particle swarm optimization, in: Proceedings of the IEEE International Conference on Neural Networks, Piscataway, NJ, USA, vol. 4, pp. 1942e1948.

Mantica, S. C. (2002). Combining Global and Local Optimization Techniques for Automatic History Matching Production and Seismic Data. *SPE Journal*, 123–130.

Martinez-Soto, R., Rodriguez, A., Castillo, O., Aguilar, T., (2012): Gain optimization for inertia wheel pendulum stabilization using particle swarm optimization and genetic algorithms, in: International Journal of Innovative Computing, Information and Control vol. 8, pp. 4421e4430

Maucec, M. D. (2007). 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

Mohamed, L., Christie, M., Demyanov, V., (2010):Reservoir Model History Matching with Particle Swarms, Institute of Petroleum Engineering, HerioteWatt University, Edinburgh, UK, SPE 129152.

Oliver, D.S., Albert, C.R., Liu, N. (2008): Inverse theory for Petroleum Reservoir Characterization and History Matching. Cambridge University Press, Cambridge

Palatnic, B., Zakirov, L., Haugen, S., and van Roosmalen, J. (1993). New Approaches to Multiple History Matching. Seventh European Symposium on Improved Oil Recovery. Moscow, Russia.

Riazi, S. H., Zargar, G., Baharimoghadam, M., Moslemi, B., Darani, E. S., (2016): Fractured reservoir history matching improved based on artificial intelligent KeAi

Riazi, S.H., Heydari, H., Ahmadpour, E., Gholami, A., Parvizi, S., (2014): Development of novel correlation for prediction of hydrate formation temperature based on intelligent optimization algorithms, J. Nat. Gas Sci. Eng. 18 377e384.

Rwechungura, R. W., Dadashpour, M., and Kleppe, J. (2011). Application of Particle Swarm Optimization for Parameter Estimation Integrating Production and Time Lapse Seismic Data. Offshore Europe. Aberdeen, UK: Society of Petroleum Engineers.

Shahkarami, A., Mohaghegh, S. D., Gholami, V., and Haghighat, S. A. (2015). Artificial Intelligence (AI) Assisted History Matching. SPE Western North American and Rocky Mountain Joint Meeting. Denver, Colorado: Society of Petroleum Engineers.

Slater, G., and Durrer, E. (1971). Adjustment of Reservoir Simulation Models to Match Field Performance. *Society of Petroleum Engineers Journal,* 295–305. Sousa, S. (2007). Scatter search metaheuristic applied to the history matching problem. SPE Annual Technical Conference and Exhibition. Anaheim, California, U.S.A.

Van Leeuwen, P. J. (1999). Comment on “Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, 127(6), 1374–1377.

Wu, X.-h., Bi, L., Yang, Y., and Wang, X. (2013). Multiphase Upscaling Using Approximation Techniques. SPE Reservoir Simulation Symposium. The Woodlands, Texas, USA: Society of Petroleum Engineers. 22 SPE-173405-MS

Xavier, C. R., Elisa Portes dos Santos, Vincius da Fonseca Vieira, Rodrigo Weber dos Santos (2013): Genetic Algorithm for the History Matching Problem Elsevier 18: 946-955.

Zhang, X., Hou, J., Wang, D., Mu, T., Wu, J., Lu, X., (2012): An Automatic History Matching Method of Reservoir Numerical Simulation Based on Improved Genetic Algorithm Elsevier 29:3924-39.