AMERICAN UNIVERSITY OF BEIRUT

HYBRID OPTIMIZATION TECHNIQUES FOR OIL FIELD DEVELOPMENT

by
ZACHARY HAMIDA

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Science
to the Program of Computational Science
of the Faculty of Arts and Sciences
at the American University of Beirut

Beirut, Lebanon
December 2016

AMERICAN UNIVERSITY OF BEIRUT

HYBRID OPTIMIZATION TECHNIQUES FOR OIL FIELD DEVELOPMENT

by

ZACHARY HAMIDA

Approved by:

Dr. George Saad, Assistant Professor — Advisor
Department of Civil and Environmental Engineering

Dr. Fouad Azizi, Associate Professor — Co-Adviser
Department of Chemical and Petroleum Engineering

Dr. Mazen Al-Ghoul, Professor — Member of Committee
Department of Computational Science

Date of thesis defense: December 02, 2016

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: Hamida Zachary
Last First Middle

● Master's Thesis ○ Master's Project ○ Doctoral Dissertation

☑ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after :
One ---- year from the date of submission of my thesis, dissertation, or project.
Two ---- years from the date of submission of my thesis, dissertation, or project.
Three ---- years from the date of submission of my thesis, dissertation, or project.

Zachary.H 06/Feb/2017

Signature Date

ACKNOWLEDGMENTS

Regardless if this work held a highly significant scientific contribution or the least significant one; it had its greatest contribution to me as an individual. The shift in perspective and the people that I got to meet throughout my study time are unarguably priceless.

In this I want first and foremost to express my gratitude to my thesis adviser Professor George Saad. His immense knowledge, continuous support and believe in me as an individual -from day one- were one of the essential drives that got me to where I am today. I've learned from Professor George to better express myself, to live and love what I'm doing and most importantly to not let the circumstances decide the outcome.

I would like also to express my sincere thanks to my thesis co-adviser Professor Fouad Azizi for his guidance, patience and insightful comments to improve this work. Moreover, I would like to acknowledge and thank my department chairperson Professor Mazen Al-Ghoul for his tremendous support and advices which guided me throughout my study time at AUB.

Finally, I want to acknowledge and thank the people who stood by my side throughout the journey (family, friends and loved ones): Tamara Hamideh, Fatema Husini, Abdul hadi Hamida, Marie-Eve Patry, Simon Englks, Frida Romero, Jenny Kahaleh, Nabil Atallah, Karen Gebrael, Amani Maalouf, Sara Dia, Ahmad Al Ghor, Hussein Kassem, Hagop Jabotian, Lama Thebian and Sana Kalash.

AN ABSTRACT OF THE THESIS

Zachary Hamida for Master of Science in Computational Science
Major: Machine Intelligence

Title: Hybrid Optimization Techniques for Oil Field Development

This study aims at introducing a problem-specific modified Genetic Algorithm (GA) approach for optimal well placement in oil fields. The evolution method used in this algorithm includes a novel genetic operator named “Similarity Operator” alongside the standard operators (i.e. Mutation and Crossover). The role of the proposed operator is to find promising solutions that share similar features with the current elite solution in the population. For the well placement problem in oil fields, these features include the new well location with respect to pre-located wells and the porosity value at the proposed location. The presented approach highlights the importance of the interaction between the nominated location and the pre-located wells in the reservoir. In addition, it enables systematic improvements on the solution while preserving the exploration and exploitation properties of the stochastic search algorithm. The robustness of Genetic Similarity Algorithm (GSA) is assessed on both the PUNQ-S3 and the Brugge field data sets.

CONTENTS

ILLUSTRATIONS

TABLES

CHAPTER 1

INTRODUCTION

Throughout the different stages of oil field development and planning, decisions have to be made continuously to maintain the sustainability of the project's dynamic nature. These decisions main objective is to extract the hydrocarbons from the reservoir at the highest possible profit. Hydrocarbons are recovered through different stages and mechanisms known as the Reservoir Drive Mechanisms [1]. The first hydrocarbon recovery mechanism is the "Primary Recovery" whereby the natural pressure is sufficient to sweep the hydrocarbons towards production wells. This pressure results mainly from the fluids expansion within the reservoir (i.e. Solution gas drive, Gas cap drive, Water drive and Gravity drainage). Since the primary recovery stage may not last throughout the entire lifetime of the reservoir or may be insufficient in terms of production rates, "Secondary Recovery" mechanisms and techniques are employed. In the Secondary Recovery stage, a fluid is injected in the reservoir to provide an additional pressure and facilitate the oil displacement towards production wells (i.e. Waterflooding and Gasflooding). The production from the field under the Secondary Recovery stage will continue until a threshold whereby production rates are insufficient or do not meet the required target. Therefore, "Tertiary Recovery" or as commonly known "Enhanced Oil Recovery, EOR" mechanisms are employed to improve flagging production. These mechanisms are extremely expensive and mainly fall under three categories, the Thermal EOR, the Chemical EOR and the Miscible Gas Flooding. The main objective of the EOR techniques is to make it easier for the oil to be swept towards production wells by adjusting the medium properties (i.e. reducing heavy oil viscosity). Moreover, if all of the aforementioned mechanisms failed in extracting the hydrocarbons, the “Infill Recovery” is employed whereby the unproduced oil is accessed directly through additional wells.

The economic efficiency of the recovery mechanisms discussed above is mainly dependent on the production plan employed in the field. Oil field production planning ultimate goal is to find strategies associated with the highest return at the lowest possible cost. Mainly, production planning uses a production forecast in assessing the different strategies performance. This forecast is obtained through a reservoir simulator as well as a simulation model for the field. The reservoir simulation model is generally composed of a fluid model (i.e. Oil, Gas and Water), a well model (i.e. production or injection, vertical or horizontal… etc.) and a geological model representing the soil properties in the reservoir.

Since the soil properties are highly heterogeneous and field measurements are not easily obtained, the geological model suffers from uncertainties. These uncertainties are attributed to the use of interpreted data in order to fill up the missing data (i.e. Kriging) and the use of model upscaling which is a technique employed to convert the fine geological model into a coarse model in order to leverage the computational cost of the simulation. Solving for uncertainties in the geological model is an important task due to its direct impact on the accuracy of the production forecast. However, this also comes with a tradeoff of an additional computational cost [2, 3, 4]. This computational cost is determined by the computational time and capacity required to accomplish the task. Moreover, solving for oil field planning problems (i.e. finding an optimal location for additional wells) requires employing a search algorithm that may need an exhaustive number of computationally expensive simulation runs. Therefore, a significant proportion of research in oil field development and planning was devoted towards obtaining a reliable and economically efficient production plan at the lowest computational cost.

Several oil field planning problems were addressed in the literature, and a big proportion was devoted for the well placement problem [5, 6, 7, 8, 9, 10]. Prioritizing the well placement problem was due to the high costs associated with decisions related to drilling and adding new wells.

Searching the reservoir simulation model for optimal well locations is a difficult task, due to the high nonlinearity of the search space, the multiple local optimal solutions, and the need for a production forecast (simulation run) to evaluate the location efficiency. In addition, the uncertainty in the reservoir data also adds to the problem complexity as it reflects directly on the accuracy of the simulation model forecast. The aforementioned factors combined have contributed to the computational cost and complexity of the problem. To depict the basic procedures in reservoir engineering, (Figure 1) illustrates the major operations related to collecting and collaborating field data in order to build a reservoir simulation model that can be used for field planning.

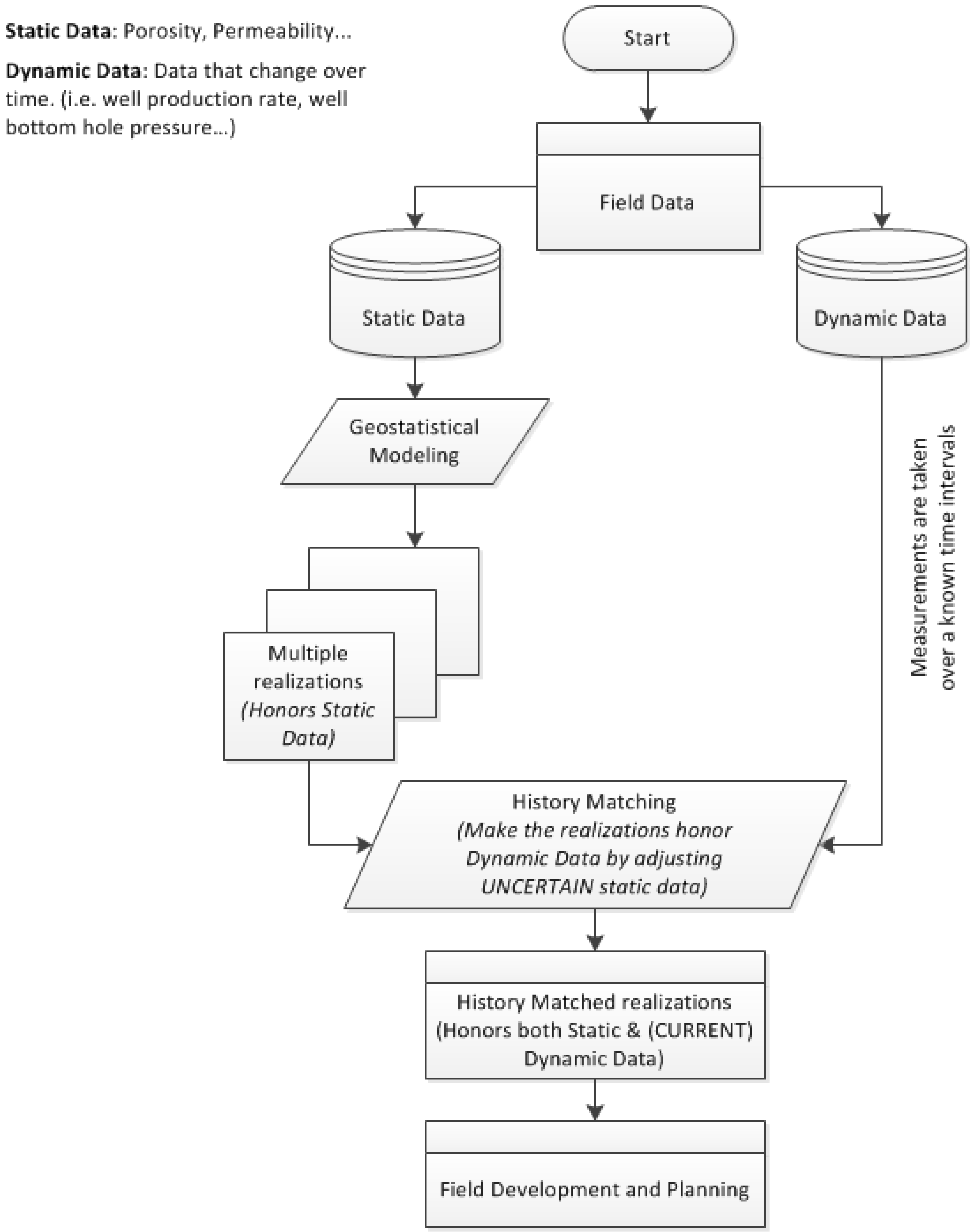


Figure 1: Reservoir Engineering Framework

Following this chapter, in Chapter 2, a comprehensive literature survey will cover the main aspects in the aforementioned reservoir engineering framework (Figure 1) as well as the scope of work for this study. Chapter 3 will provide a detailed description for the proposed optimization approach along with other different methodologies used in the literature. In Chapter 4, results and analysis of the numerical experiments are shown using two reservoir models (PUNQ-S3 and Brugge) to validate the efficiency of the proposed approach. The numerical experiments will include comparisons among the different methodologies covered in Chapter 3. Finally, the conclusions of the analysis will be presented in Chapter 5.

CHAPTER 2

LITERATURE SURVEY

The literature survey was divided into three sections; the first section will tackle uncertainty quantification and modeling techniques, the second section will highlight some of the previously applied optimization methods as well as problem formulations addressing well placement in oil fields, while the third section highlights methods concerned with reducing the computational cost associated with the geological model.

1. Literature Survey

1.1 Uncertainty Quantification

Uncertainties in the reservoir simulation model arise from the lack of an accurate representation of highly heterogeneous, yet essential reservoir properties (i.e. porosity and permeability). This underrepresentation of the properties can be improved through a technique known as History Matching. The objective of History Matching (HM) is to tune uncertain properties in the simulation model such that, the simulation results match the field observations over time. The adjustments are either done manually (Manual HM) based on experience or automatically (Automatic HM) by employing an optimization algorithms [2]. Commonly, two methodologies were assessed in the literature: the Gradual Deformation Method (GDM) and the Ensemble Kalman Filter (EnKF).

The Gradual Deformation Method (GDM) is a geostatistical parametrization technique. GDM creates realizations that evolve smoothly while preserving the global characteristics of the data. These realizations are updated through an optimization algorithm to match the production history of the field [4].

The other automatic HM approach is the Ensemble Kalman Filter (EnKF), which is a Monte Carlo based methodology for history matching and real time updates of reservoir realizations. EnKF consists of a forecast step and an assimilation step, in which the variables of the reservoir state vector are updated to honor the field measurements [3].

An extension to History Matching with EnKF, Lyons *et al.* [11] introduced the pseudo history matching using EnKF. In the pseudo history matching, field measurements (pre-located + newly added well logs) are estimated from the realization corresponding to the most likely oil recovery estimation (P50 realization). This allows integrating the probable future uncertainty within the reservoir production forecast.

1.2 Oil Field Development and Planning

Different optimization algorithms were suggested to solve oil field planning problems (i.e. Allocating additional wells, optimizing flowrates through existing and additional wells) [12, 13, 14, 9]. The efficiency of these algorithms was measured by solution robustness, convergence rate and the total computational cost of the optimization process. Handles *et al.* [12] and Sarma *et al.* [13] applied gradient-based optimization with variations in an attempt to reduce the prospect of converging to sub-optimal solution. The gradient-based search algorithms have a systematic convergence due to having a search direction. However, these algorithms may suffer from limitations and drawbacks that weaken their reliability; namely, difficult implementation, high computational cost (i.e. calculating search direction), inability to explore the search space efficiently, and a tendency to converge to the first sub-optimal solution. For the aforementioned reasons, derivative-free algorithms present themselves as a more reliable option in solving the problem. Derivative-free search algorithms can be mainly categorized in two groups: local search methods which apply local adjustments on the solution candidates (i.e. simplex method) and global search methods (i.e. population-based algorithms) [15]. Different population-based algorithms were applied in the literature to solve the problem of well placement in oil fields [16, 9, 17, 6]. Montes *et al*. [16] applied Genetic Algorithm (GA) search to solve for well placement in oil fields and assessed the impact of different parameters on the algorithm performance (i.e. mutation to cross over ratio, starting point…etc.). Onwunalu *et al.* [17] applied a Particle Swarm Optimization (PSO) algorithm to search for optimal well location and type (production or injection). Afshari *et al.* [6] assessed the performance of an Improved Harmony Search (IHS) algorithm (Mahdavi *et al.* [18]), which has a better local search performance than the standard HS, in solving the well placement problem. As the aforementioned algorithms have a general context that doesn't account for computationally expensive objective function, variations to these algorithms were introduced aiming at improving the convergence rate at a minimum computational cost. Bittencourt *et al.* [14] used a hybrid algorithm of GA and polytope method to solve for optimal well placement. Da cruz *et al.* [10] introduced the Quality Map approach to present a different way of evaluating well locations while limiting the use of the reservoir simulator. Güyagüler *et al.* [9] used Hybrid Genetic Algorithm (HGA) (Genetic Algorithm + simplex method + surrogate model) to search for optimal location and flow rates for the added wells.

Although population-based algorithms have had a superior performance in terms of usability and convergence rate, the search-space of the well allocation problem still imposes difficulties that may hinder the efficiency of these algorithms. For example, population-based algorithms might evaluate and propose locations of a low quality for wells, such as locations adjacent to pre-located wells or locations not within the active cells of the reservoir model. This is due to the stochasticity of the operators used in nominating locations for wells. Also, early convergence or premature convergence of a population may contribute to increasing the number of ineffective simulation runs. These factors combined consume a significant portion of the total computational cost required to find an optimal well location.

Customization techniques for these algorithms were applied to make them adapt to the search-space of the problem [19, 7]. This was mainly achieved through the objective function formulation or applying constraints on the search space.

One of the commonly used approaches in formulating the objective function is the penalty and reward approach. This approach suggests adding a penalty parameter to the objective function to account for the problem non-practical solutions. Although this type of formulation can aid the search algorithm in identifying the less plausible solutions, it does not contribute in finding new good solutions.

1.3 Uncertainty Assessment

Throughout the optimization process, comparing the inputs efficiency is based on their corresponding outputs. Since the reservoir model has multiple geological realizations, each realization may have a unique production curve over time. Thus, for each single input (i.e. well formation), multiple outputs are obtained (Figure 2).

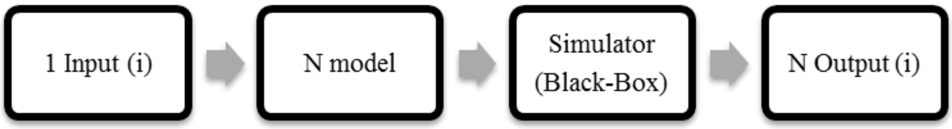


Figure 2: Input evaluation under uncertainty

The common approach in dealing with this problem is to consider the average of the outputs as a correspondent to the evaluated input scenario. However, some studies have had extended investigations with other factors such as the standard deviation of the output as well as the number of geological realizations used.

Chang *et al.* [20] applied Quality Map assisted with NSGA-II (modified GA) search algorithm to solve for different oil field development problems. The NSGA-II is a multi-objective non-dominate sorting genetic algorithm. In their study, Chang accounted for model uncertainty by maximizing the mean while minimizing the standard deviation of the model response distribution.

Rashid *et al.* [21] improved previous work based on Pareto Optimal concept to assess the risk of selecting and comparing different output distributions. The Pareto optimal concept is used to assess the tradeoff between maximizing the mean and minimizing the standard deviation in contrast to the risk averse and the risk greedy decision makers.

Mishra *et al.* [22] suggested reducing the number of realizations through generating representative realizations. Mishra used the statistical moment equations to find weights (w) for a set of realizations such that the statistical properties of all realizations are preserved.

Shirangi *et al.* [23] used Optimization with Sample Validation (OSV) method to reduce the number of realizations by selecting a representative sample of realizations from the available realizations of the model.

Wang *et al.* [8] introduced the Retrospective Optimization (RO) framework to be used in oil field development and planning. The RO framework solves a sequence of sample-path optimization problems. The number of realizations (sample size) is increased from sub-problem to sub-problem, and the initial solution for the current sub-problem is simply the returned solution from the previous sub-problem.

2. Study Objective

In this study, a new genetic operator named "Similarity Operator" is proposed to efficiently solve the well placement problem in oil fields. The operator will function alongside the standard genetic algorithm (GA) operators (i.e. Crossover and Mutation) and aims at searching for solutions that share similar features with the current elite solution in the population. This new framework will be referred to as Genetic Similarity Algorithm (GSA). The addition of this new operator will provide potentially good solutions while preserving the exploration and exploitations properties of the standard operators.

The use of Genetic Algorithm was mainly intended to demonstrate the significant performance-improvement that can be obtained in contrast with a standard implementation of the algorithm. Since population-based algorithms have a general context and their performance is highly dependent on the parameter settings as well as the problem description, every search algorithm may have an edge in solving for a certain problem [24, 25]. Therefore, the customizations introduced in the GSA framework accounts for the broad generality of the previously applied approaches by incorporating information about the search-space of the problem when searching for new solutions.

CHAPTER 3

METHODOLOGY

1. Genetic Algorithm

Introduced by Holland et al. [26], Genetic Algorithm (GA) is a stochastic search algorithm motivated by the principle of evolution. GA has an efficient performance in problems with high number of input variables as well as high number of local optima. The algorithm explores the search space through a population (generation) of solutions (individuals), and these solutions evolve based on a fitness value obtained from the objective function. The fittest individuals within a generation will undergo genetic operators (i.e. mutation and crossover) to generate a new generation replacing the previous one. Figure 3 illustrates the different stages in GA search for solutions. Since this study is suggesting a change in the GA framework, it is convenient to tackle the role of each stage and operator within GA. The following is a brief description for the GA main stages.

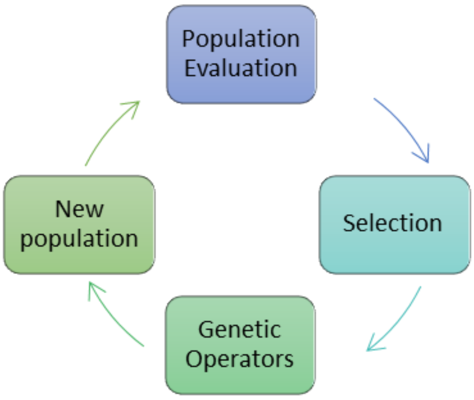


Figure 3: Genetic Algorithm Structure

1.1 Initial Population

The initial population of individuals can be generated manually using a set of selected individuals or randomly from within the search space of the problem.

1.2 Selection

After evaluating all the individuals in a generation, the algorithm will rank these individuals based on their fitness value. The ranking determines the individual probability of survival in the selection process. Different selection techniques were developed in the literature (i.e. roulette wheel, tournament, uniform …etc.) [27], however, the choice of a selection technique is highly dependent on the variation in the fitness function values.

1.3 Genetic Operators

Genetic operators perform operations over individuals that survive the selection stage. Each genetic operator contributes to the next generation with a predefined proportion of individuals. The following are some of the commonly used genetic operators:

- Elitism: The Elite operator role is to move the best individuals in the population to the next generation without changes. This operator helps preserving good solutions in the population; however, it may also contribute to the occurrence of early convergence in the population due to replicating the same individual(s) multiple times in the next generations.
- Mutation: The goal of mutation is to reassure the diversity in the population. The operator alters values within a single individual at different locations in the encoded string. Different instances of the mutation operators were developed (i.e. Gaussian, uniform and bit flip), accounting for different types of problems. Mainly the choice of the mutation operator is dependent on the search space properties (i.e. integer or continuous).
- Crossover: The crossover combines and merges the selected individuals to generate new individuals. Similar to the Mutation operator, many instances of the Crossover operator (i.e. single point, two point, arithmetic …etc.) were developed and used depending on the problem being solved.

2. Hybrid Optimization Framework

The hybrid optimization framework is commonly used in solving optimization problems associated with a computationally heavy objective function [9]. The framework's main components are a light computational cost surrogate model (i.e. ANN) and a search algorithm (i.e. GA). The role of the surrogate model is to nominate potentially good solutions to be evaluated and validated by the simulator. The surrogate model is constructed and trained simultaneously based on data generated from the search scheme. This will allow the surrogate model to become a better replica for the heavy simulator as the search algorithm explores new locations in the search space. A general structure of a Hybrid Genetic Algorithm (HGA) framework is shown in (Figure 4).

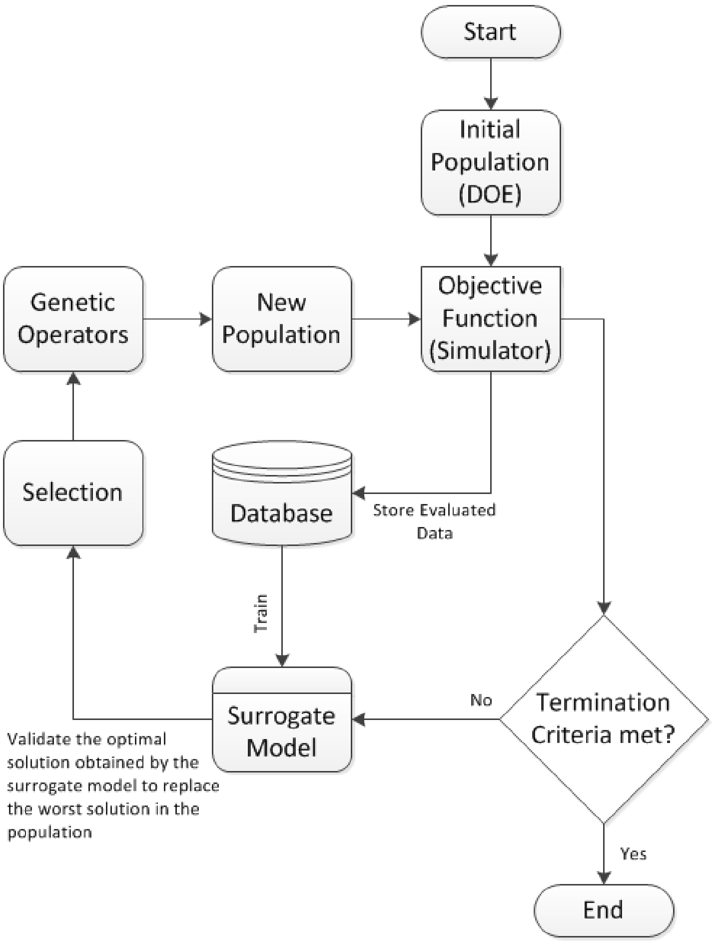


Figure 4: Hybrid Genetic Algorithm (HGA) framework

As shown in Figure 4, the initial population is generated using one of the Design of Experiment (DOE) approaches (i.e. Latin Hypercube sampling) and thereafter it's evaluated using the reservoir simulator. After each simulation run, the fitness function associated to the input is computed. The input-output data are stored sequentially in a database at the end of each generation, so it can be used later to train the surrogate model. Following the training step, the surrogate model can provide estimates for the objective function value at unobserved input data. Since these estimates have a relatively low computational cost, an extensive search for an optimal estimate can be conducted. Thereafter, the input corresponding to the optimal estimate will be validated using the reservoir simulator. If the validated input data had a better fitness function value than the worst individual in the genetic population, it will replace it. Following this step, selection and genetic operators can operate on the final (updated) population and scores to generate a new population.

2.1 Surrogate Models

Surrogate models are used to emulate the response of computationally expensive functions. These models are built using input-output data; therefore, the accuracy of a surrogate model prediction is highly dependent on the quantity and quality of the data used to construct it. Different types of surrogate models were used in the literature [28, 29, 30, 31]. The following are some of the commonly used models for oil field development problems,

2.1.1 Artificial Neural Network (ANN)

Artificial Neural Networks (ANNs) were influenced from the biological nervous systems of the human's brain [32]. ANN models are commonly represented graphically with a sequence of nodes (inputs, hidden neurons and outputs) connected through links (weights). The inputs and outputs are taken from the training data (observations). The weights in ANN are tuned such that, for a given input in the training set, the predicted output using ANN comes close to match the output provided in the training set. Moreover, the number of hidden neurons reflects on the level of complexity of the ANN model, as an increase of hidden neurons will result in an increase of weights to be learned and subsequently more computation time will be needed at the training stage. (Figure 5) illustrates the mathematical structure of ANN.

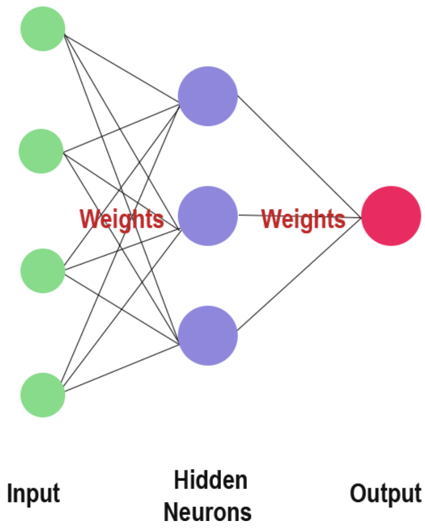


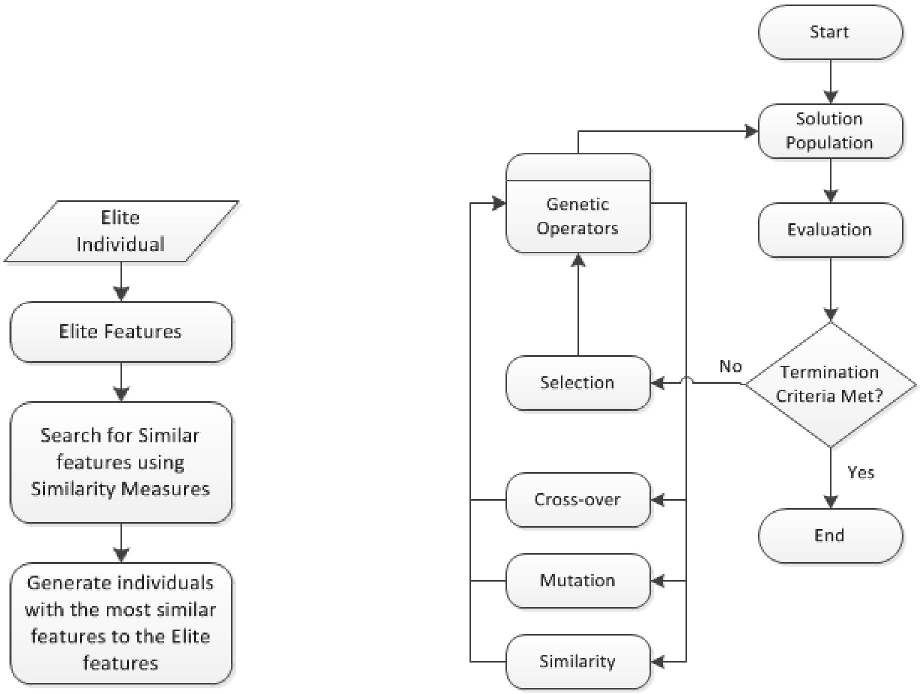
Figure 5: Artificial Neural Network (ANN) with one hidden layer

2.1.2 Support Vector Machine Regression (SVR)

Support vector machine are class of algorithms with applications to classification and regression problems [33]. The initial concept behind Support Vector Machine algorithm is to generate a hyperplane separating two classes of data points with the largest possible margin. The points' tangents to the hyperplane sides are the support vector that decides the hyperplane boundaries. Support Vector algorithms are characterized by the presence of a deterministic unique solution. This solution represents the unique linear maximum margin separating the different classes in classification problems or the unique linear minimum margin for regression problems. In cases where the data can't be separated by a linear margin, Kernels are employed such that the data are projected to a higher dimension where it can be separated linearly.

3. Genetic Similarity Algorithm (GSA)

The proposed search algorithm presented in Figures 6a and 6b is based on the aforementioned GA with an additional operator named "Similarity Operator" to help explore the search space more efficiently. The Similarity Operator aims at finding promising solutions by exploring the search space in a systematic manner. The solutions proposed by the operator share certain search-space features with the current elite solution in the population. The techniques used in building the operator will be described in details in the following sections.



(a) (b)

Figure 6: (a) Similarity Operator Flow chart; (b) Genetic Similarity Algorithm

3.1 Similarity Measures

Similarity measures are distance-based measures which reveal how quantitatively different two data objects are from each other. The type of similarity measure used depends on the type of the data being measured (i.e. categorical, binary, continuous…etc.). One of the commonly used similarity measures is the Minkowski distance which can be defined as follows:

$$d(i,j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \cdots + |x_{in} - x_{jn}|^q} \qquad (1)$$

Whereby i and j stand for the first and the second n-dimensional objects respectively, q is a positive integer and $d(i,j)$ is the measure of difference between the two objects. For $q = 1$, $d(i,j)$ becomes the Manhattan distance and for $q = 2$, $d(i,j)$ becomes the Euclidean distance. The value of $(d(i,j) = 0)$ indicates that the two objects are exactly the same, whereby for any value greater than zero, the two objects have differences in their respective features.

Similarity measures are considered the core routine for some major data mining techniques such as clustering (i.e. k-means clustering) [34] and classification (i.e. kNN) [35]. In the proposed approach, we used similarity measures to identify locations in the reservoir model that share similar features with the location corresponding to the best objective function value.

3.2 Similarity Operator

The proposed operator shown in Figure 6a aims at finding individuals with features or properties quantitatively similar to the elite individual in the current generation. These features are selected based on their impact on the problem solution.

Injection and/or production wells formation in the reservoir have a major impact on the search process for a new well location. As this formation is fixed throughout the search, it can be used as a guide for the search algorithm. To illustrate the interaction between an added well and pre-located wells formation, a spatial point-distance approach is used. In this approach, the Euclidian distance for each cell to the nearest wellbore is calculated and stored in a new raster. This raster can reveal the cells that share similar distances from a nearest pre-located wellbore. Figure 7 illustrates the distance to nearest-wellbore concept for a given well formation. Green cells represent locations that are in the proximity of existing wells, and pink cells represent further away locations.

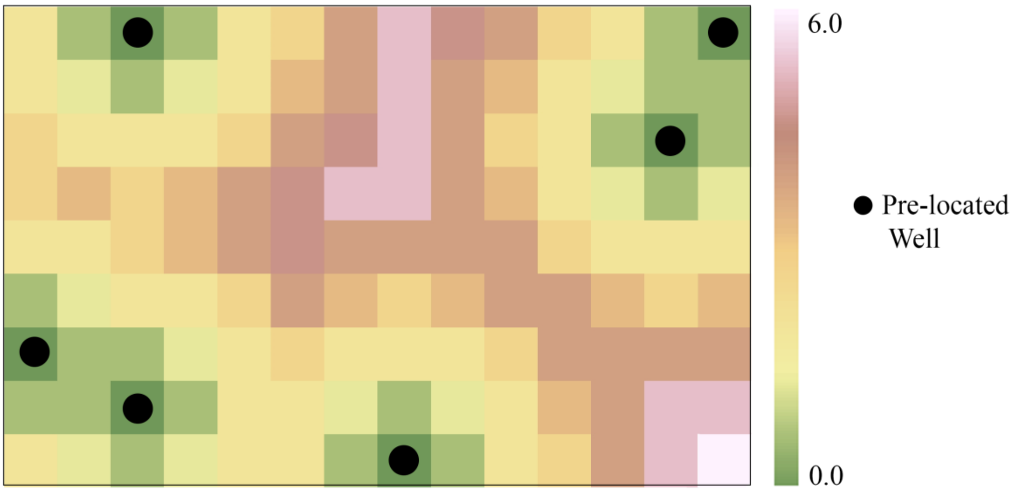


Figure 7: Distance for each cell to nearest pre-located well (example data)

Another significant feature (property) that can be used by the operator is the porosity value at the cell. The porosity value is dimensionless and can provide information about the flow within the model grid through its correlation with the permeability. Figure 8 shows a porosity data set for the same well formation example shown above.

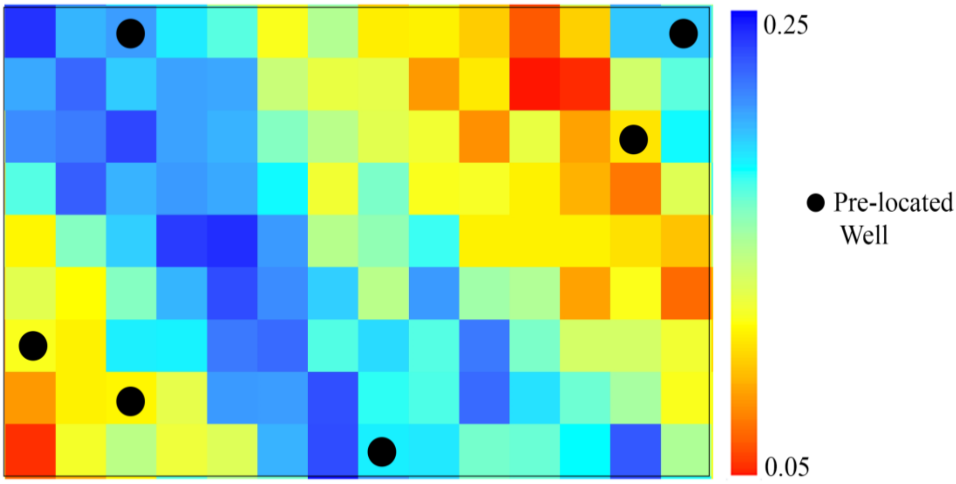


Figure 8: Porosity raster for a reservoir (example data)

Combining and normalizing the two aforementioned features in a single scatter plot (Figure 9) can provide a non-biased visual representation for the selected features (properties) in the reservoir model.

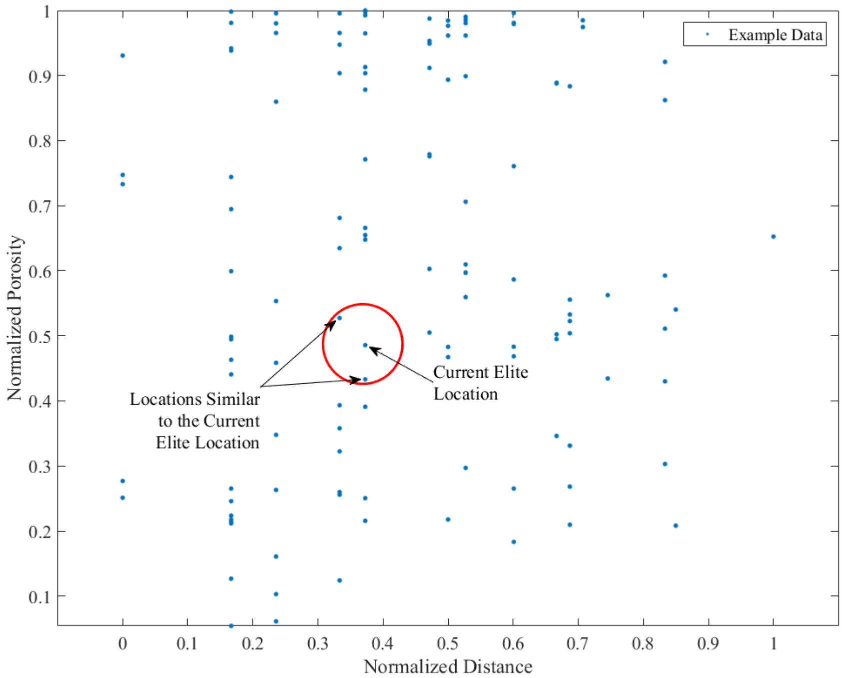


Figure 9: Scatter Plot for Normalized Distance to Nearest Well vs. Normalized Porosity (example data)

Figure 9 allows deciding graphically how similar two (or more) locations in the model are to each other with respect to the selected features. The Similarity Operator will act in a similar manner when deciding the similarity level between two locations. The operator will take as an input the current elite individual in the GA generation (highlighted in the plot), identify its features, and then look up individuals with the most similar features based on the following distance measure:

$$SO_i = \sqrt{(X_i - X_E)^2 + (Y_i - Y_E)^2} \quad ; \quad i = 1,2,3 \dots n \qquad (2)$$

Whereby n is the number of active cells in the model, SO_i is the resulting vector of similarity values, X_i and Y_i are the vectors representing the normalized distance and the normalized porosity respectively for the n cells in the model, X_E and Y_E are the normalized distance and the normalized porosity respectively from the elite individual only. The cell corresponding to the minimum value in the SO_i vector is identified as the most similar to the elite individual. Thus, it will be proposed by the operator in the next generation. This approach will enable systematic improvements on the solution throughout the generations along with preserving the population diversity. Figure 10 presents a pseudo code for the sequence of operations within the similarity operator.

Similarity Operator

Var. 1: N = number of grid-blocks in the search space.
Var. 2: n = number of individuals proposed by the operator.
Input: Elite Location (x, y)
Output: new individuals
Extract Elite location features: e (distance to pre-located well, Porosity)
for i=1: N **do**
    Procedure S(i) = similarity(e, grid-block features (i))
end for
Procedure Sort (S, Ascending)
for j = 1: step = n: N/n **do**
    if S(j : n) ∉ Output
        Output ⟵ S(j : n)
        Procedure re-map features to location indices S(j : n) ⟶ L(1 : n)
        new individuals = L(1 : n)
        BREAK
    end if
end for
END

Figure 10: Pseudo code for the similarity operator

CHAPTER 4

RESULTS AND ANALYSIS

The efficiency of the presented methodology along with other algorithms will be tested on two different models, the PUNQ-S3 oil field model and the Brugge oil field model. Each algorithm performance is tested and compared under MATLAB platform using 3 machines with 12 cores (24 hyper-threading) each at frequency of 2.0 GHz and 32 GB of RAM.

1. Reservoir Optimizer Software

The software used in this study entitled “Reservoir Optimizer” was developed using MATLAB along with fully functional GUI to process the different computer experiments. This software facilitates the exchange of input/output data with the reservoir simulator Schlumberger ECLIPSE (Figure 11) along with enabling the use of different optimization algorithms to solve oil fields planning problems. The reservoir simulator was employed in batch mode and the results were reported through an ASCII formatted log file organized by the Reservoir Optimizer software. Figure 12 illustrates the architect of the Reservoir Optimizer whereby each level in the flowchart represents the different options available at the current window.

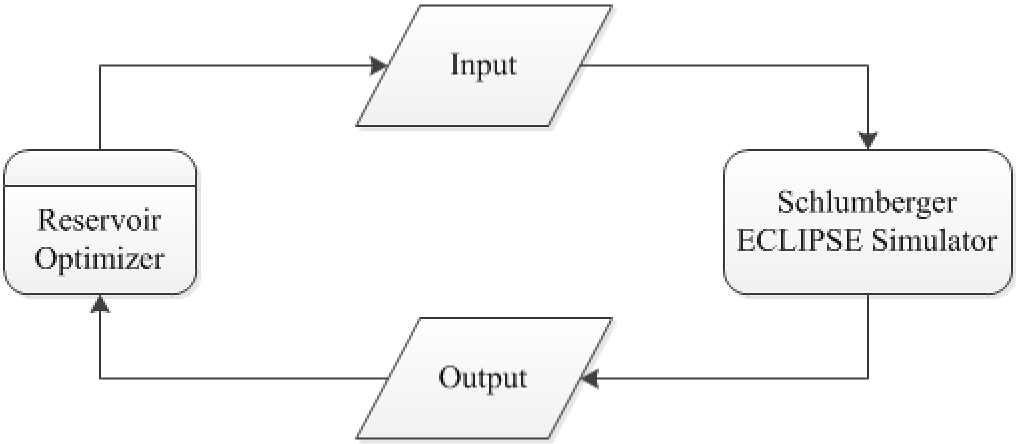


Figure 11 Communications flowchart between Reservoir Optimizer and ECLIPSE Simulator

The ECLIPSE simulator suite consists of two separate simulators: ECLIPSE 100 (used in this study) specializing in black oil modeling, and ECLIPSE 300 specializing in compositional modeling. ECLIPSE 100 is a fully-implicit, three phase, three dimensional, general purpose black oil simulator with gas condensate options. ECLIPSE 300 is a compositional simulator with cubic equation of state, pressure dependent K-value and black oil fluid treatments. Both programs are written in FORTRAN and operate on any computer with an ANSI-standard FORTRAN90 compiler and with sufficient memory [36].

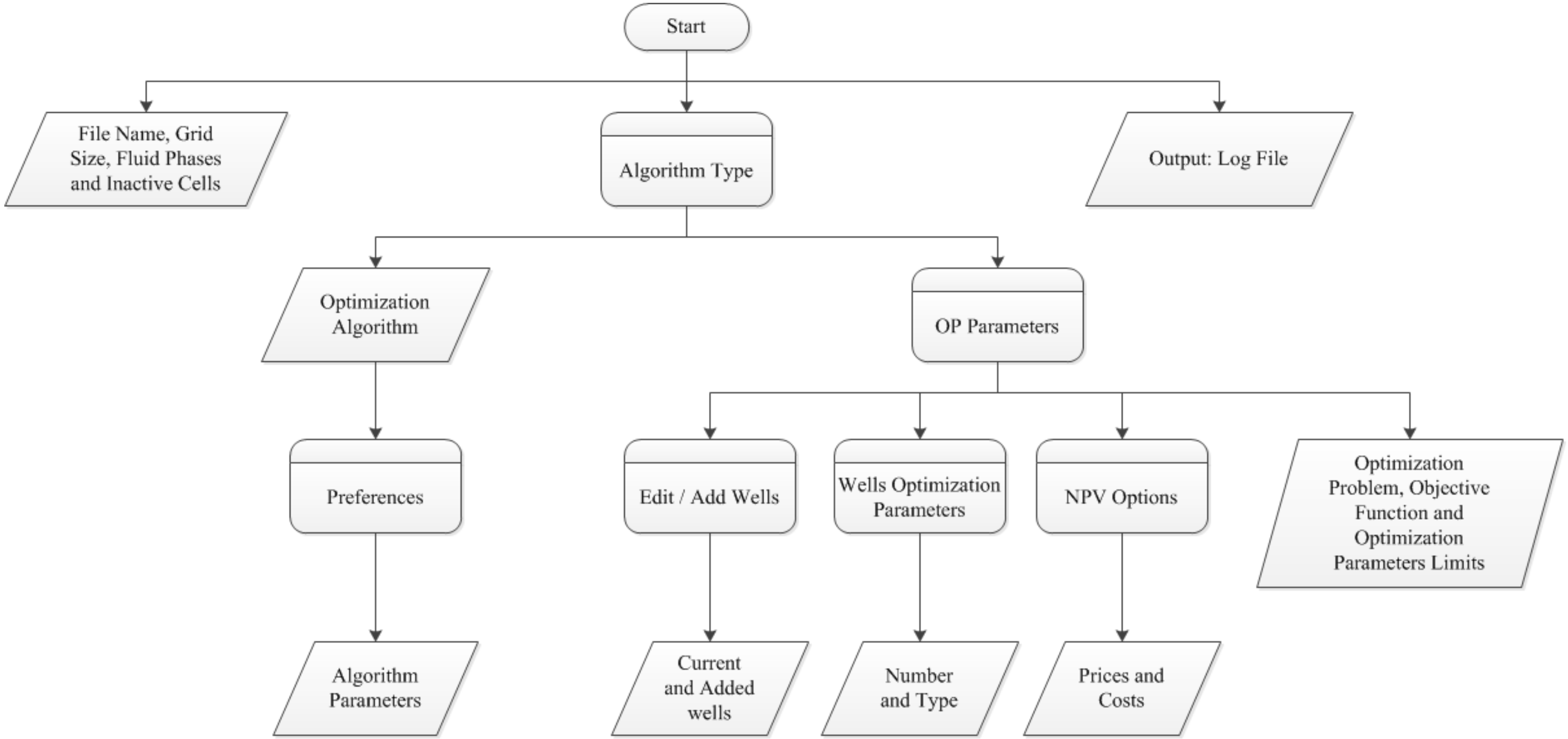


Figure 12 Reservoir Optimizer Software Flowchart

2. Numerical Experiments

2.1 PUNQ-S3 Model

PUNQ-S3 is a small-size reservoir that was taken from a study on a real field as part of PUNQ project [2]. The model contains 19×28×5 grid-blocks, of which 1761 are active. The field is bounded to the east and south by a fault and by a strong aquifer to the north and west. A small gas cap is located in the center of the dome shaped structure. The field initially has six production wells located around the gas oil contact. Due to the strong aquifer, no injection wells are operating as the aquifer pressure will provide enough pressure for production at the current stage. The geometry of the field has been modeled using corner-point geometry. Figures 13 and 14 show the field geometry and porosity distribution for the PUNQ-S3 model, respectively.

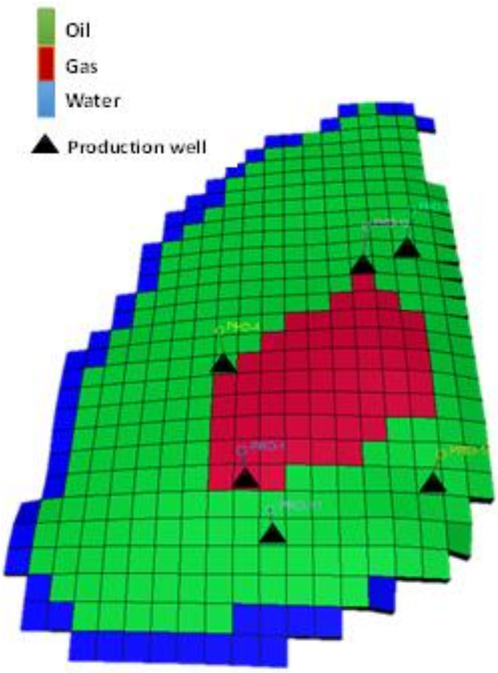


Figure 13: PUNQ-S3 Oil Field Model

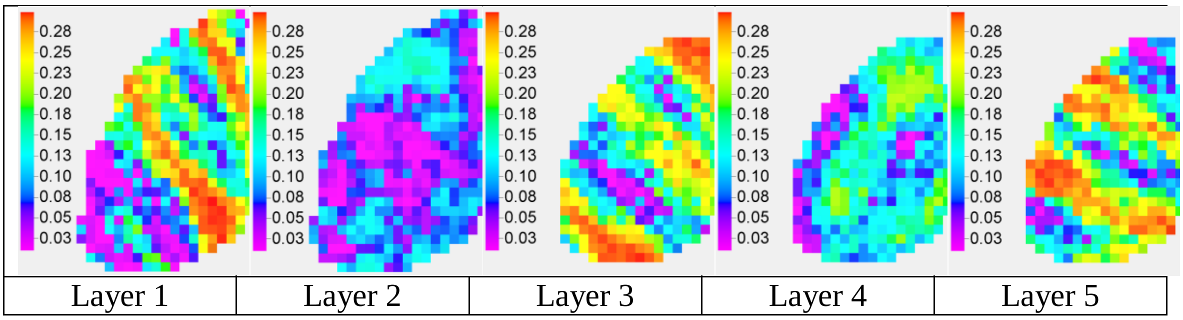


Figure 14: Porosity Distribution in PUNQ-S3 model

2.1.1 Optimal Allocation for a Single Injection Well:

In order to assess the convergence rate of the proposed approach, the simple problem of optimally allocating a single injection well in PUNQ-S3 model is considered. The objective of the problem is to maximize the total cumulative oil produced from the field over a specific period of time. As the total number of possible solutions for this problem is relatively small, thus brute force solving can be used to identify the optimal location. Brute force attempts to try all the possible solutions for a given problem. In the case of optimal well placement, the method will consider evaluating all the possible locations in the model. In PUNQ-S3 model, the number of possible locations in any layer is 532 (active + inactive) cells. The Brute force approach is guaranteed to find the global optimal solution; however, it requires an extensively large number of simulation calls. The total field production time simulated in the experiment is 28.5 years, with 22.5 years of initial production depending only on the aquifer pressure, followed by 6 years of water injection by the new injection well. The optimal location obtained under the experiment conditions is (I=19, J=17, K=5) corresponding to cumulative oil produced COP = 6.357 x 10^6 m^3. Figure 15 shows a response surface for the COP resulting from placing the injection well at each cell in the model.

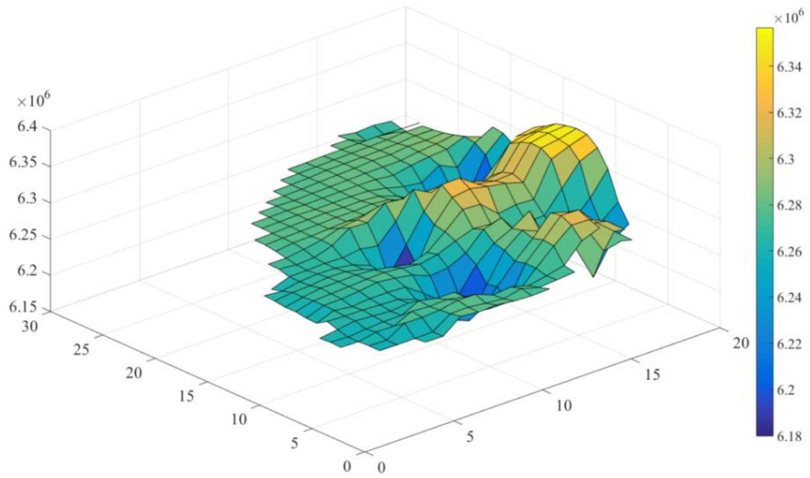


Figure 15: PUNQ-S3 Single Injection Well Allocation Response Surface

After identifying the optimal location, the convergence rate of the proposed approach compared to other search frameworks used in the literature [16, 5, 37, 38], can be assessed. These algorithms are: standard Genetic Algorithm, Hybrid Genetic Algorithm I (GA + Artificial Neural Networks) and Hybrid Genetic Algorithm II (GA + Support Vector Machine Regression). The setup of each search algorithm is given in tables 1, 2 and 3.

Table 1 Genetic Algorithm Configurations

	Genetic Algorithm							
	GA-A1	GA-A2	GA-A3	GA-A4	GA-B1	GA-B2	GA-B3	GA-B4
Pop. Size	10	10	10	10	20	20	20	20
Elite	10%	10%	20%	20%	5%	5%	10%	10%
Crossover	70%	50%	60%	50%	75%	55%	70%	55%
Mutation	20%	40%	20%	30%	20%	40%	20%	35%

Table 2 Surrogate Models Setup

Surrogate Model	
ANN	
# Layers	2
# Hidden Neurons	20,20
SVM Regression	
Kernel Type	RBF

Table 3 Genetic Similarity Algorithm Configurations

Genetic Similarity Algorithm												
Pop. Size	10	10	10	10	10	10	20	20	20	20	20	20
Elite	10%	10%	10%	10%	10%	10%	5%	5%	5%	5%	5%	5%
Crossover	60%	50%	40%	30%	20%	10%	65%	55%	45%	35%	25%	15%
Mutation	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
Similarity	10%	20%	30%	40%	50%	60%	10%	20%	30%	40%	50%	60%
Case Label	GSA-A1	GSA-A2	GSA-A3	GSA-A4	GSA-A5	GSA-A6	GSA-B1	GSA-B2	GSA-B3	GSA-B4	GSA-B5	GSA-B6

Since the reservoir simulator consumes the largest proportion of the computation time in any assessment, the number of simulation calls is considered as the comparison metric in the experiments.

Each algorithm was seeded with the same initial population. The optimization runs were repeated 100 times and the number of simulation calls until reaching the optimal solution was monitored for each run. Comparison of results for different cases is shown in tables 4 and 5; whereby μ is the average number of simulation calls until reaching the optimal solution, σ is the standard deviation of the number of simulation calls when the optimal solution is reached and % is the percentage of the number of times the algorithm converged to an optimal solution under the given stopping criteria. It is worth noting that for the cases where premature convergence was encountered, the number of simulation calls until reaching the optimal solution will be unknown, thus, it was omitted from the statistical analysis.

Table 4 Comparing Results for Population Size = 10

	Population Size = 10					
	GA-A1	**GA-A2**	**GA-A3**	**GA-A4**	**HGA I**	**HGA II**
μ	167.7	175.2	166	173.5	188.9	178.2
σ	192.6	166.2	202.13	204	169.1	146.2
%	88	96	82	90	99	100
	GSA-A1	**GSA-A2**	**GSA-A3**	**GSA-A4**	**GSA-A5**	**GSA-A6**
μ	130.8	123.5	127.4	132.5	112.8	107.8
σ	83.4	68.5	71	72.6	54.2	49.8
%	100	100	100	100	100	100

Table 5 Comparing Results for Population Size = 20

	Population Size = 20					
	GA-B1	**GA-B2**	**GA-B3**	**GA-B4**	**HGA I**	**HGA II**
μ	182.7	192.6	210.6	179.2	219.7	186.3
σ	135	87	220.97	89.62	144.8	111
%	96	100	93	100	100	100
	GSA-B1	**GSA-B2**	**GSA-B3**	**GSA-B4**	**GSA-B5**	**GSA-B6**
μ	177.3	162.5	145.5	144.4	127	128.7
σ	86.1	85	69.6	66.3	56.7	51.3
%	100	100	100	100	100	100

Figures 16 and 17 illustrate the improvement achieved by using the proposed Genetic Similarity Algorithm compared to the standard Genetic Algorithm in different configurations. This comparison was chosen because GA similarly to GSA has no dependency on a surrogate model.

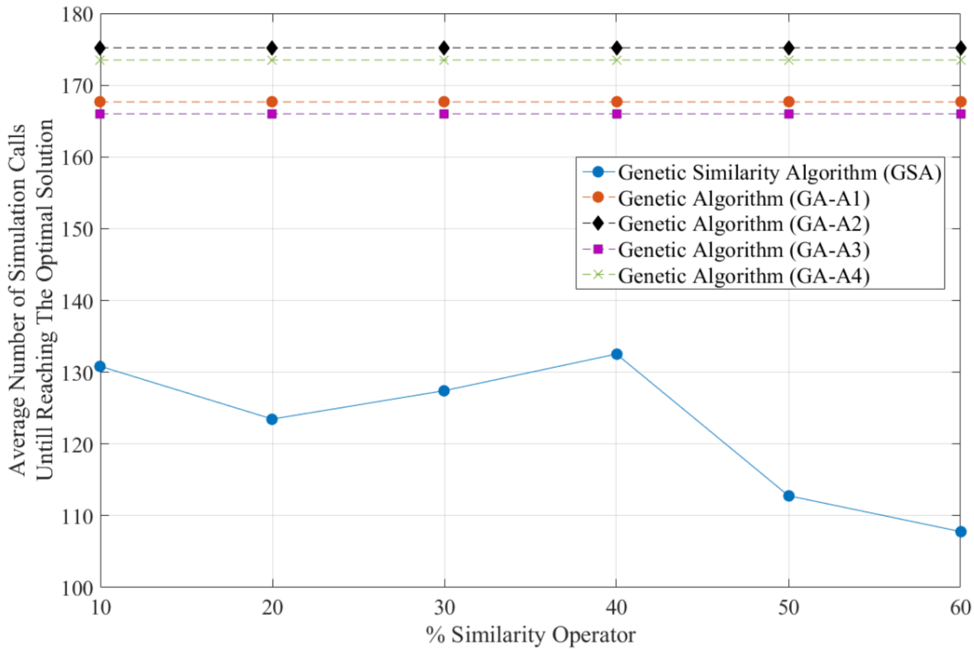


Figure 16: Genetic Algorithm vs. Genetic Similarity Algorithm with different fractions of Similarity Operator for Pop. Size = 10

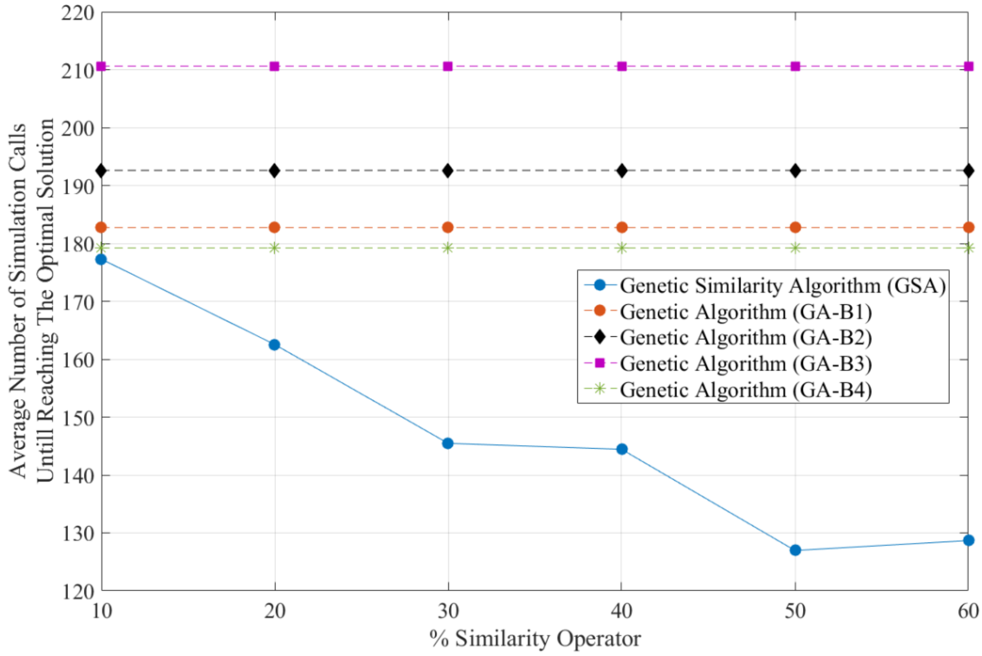
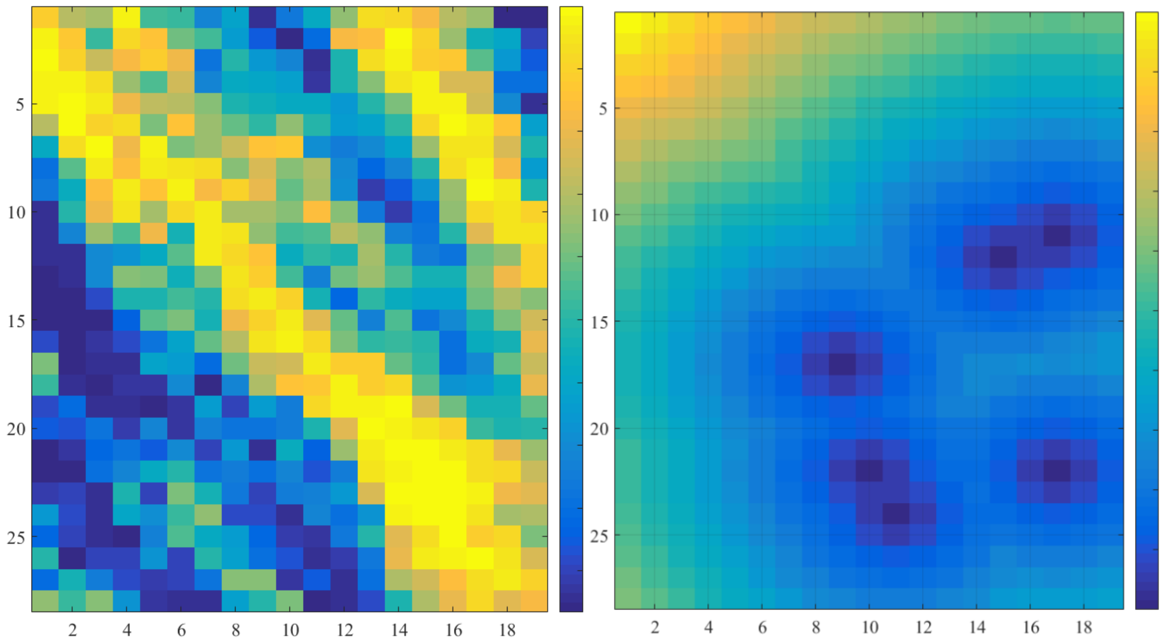


Figure 17: Genetic Algorithm vs. Genetic Similarity Algorithm with different fractions of Similarity Operator for Pop. Size = 20

It's evident that GSA has outperformed other approaches in terms of convergence rate and solution robustness. Also, it can be noted that increasing the contribution of Similarity operator in a population is likely to yield an overall improvement in the algorithm performance.

To breakdown the operator workflow in the previous example, Figures 18a and 18b show the two selected features representing each cell in PUNQ-S3 model.



(a) (b)

Figure 18: (a) (NP) Normalized porosity raster of Layer 1 (PUNQ-S3); (b) (ND) Normalized Distance from each cell to nearest pre-located well (PUNQ-S3)

Combining the porosity raster (Figure 18a) with the distance raster (Figure 18b) through a weighted sum will result into a new raster shown in Figure 19. This raster can visually assist in identifying cells in the model with similar features (porosity and distance). The operator will identify the current elite individual value (i.e. location E shown in Figure 19) in the raster and thereafter will search and find locations having similar values (i.e. locations S1 and S2) in the raster.

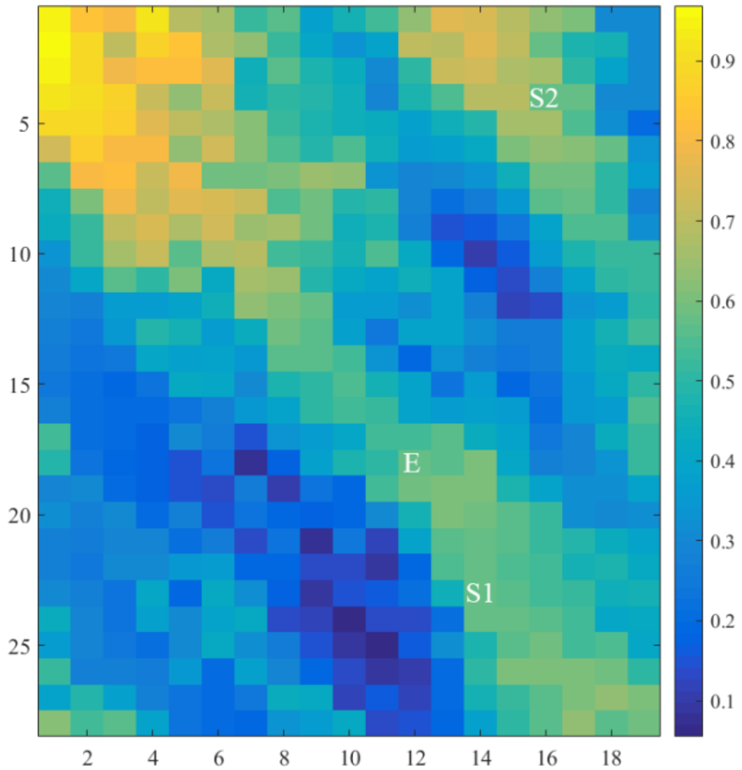


Figure 19: Similarity raster approximated from combining (0.5 NP + 0.5 ND)

It can be noted from Figure 19 that cells with extreme values, such as values near 0, are less likely to be proposed by the operator, as they represent cells adjacent to pre-located wells and/or cells with a very low permeability medium.

2.1.2 Optimal Allocation for Multi Production Wells:

For this example, allocating three additional production wells along with the existing wells in PUNQ-S3 model is considered. The wells operate at the same production rate of 150 m^3/day with a BHP constraint of 120 bar. The production time was taken over a period of 16.5 years. A single realization for the field was considered in this assessment whereby the porosity and permeability were assumed to be the true state of the reservoir. Two optimization variables for each well, {x, y}, were defined, which lead to a total of 6 optimization variables for the three wells. The size of the initial population is 10 individuals and the maximum number of generations is 200. A set of 10 complete optimization runs were performed, whereby each time the algorithms were seeded with the same initial population. The objective function in this assessment was to maximize the cumulative oil produced (COP) throughout the imposed production plan. Table 6 shows the setup for each algorithm used in this assessment. Under the aforementioned conditions, the GSA performance was compared against GA performance based on the number of simulations required to reach an optimal solution.

Table 6 GA and GSA setup for solving multiple production wells placement problem

	Genetic Algorithm	**Genetic Similarity Algorithm**
Pop. Size	10	10
Elite	10%	10%
Crossover	70%	40%
Mutation	20%	20%
Similarity	-	30%

The average performance of 10 GA optimization runs is shown in Figure 20. The algorithm has reached an optimal solution (on average) after 1760 fitness function evaluations (simulation calls) corresponding to COP = 6.016×10^6 m^3.

On the other hand, considering the GSA average performance shown in Figure 21, it is evident that GSA outperformed GA in convergence rate and solution robustness. GSA has reached (on average) a solution greater than GA optimal solution after 400 fitness function evaluations. Furthermore, the best optimal solution under the experiment conditions, was obtained by GSA algorithm and corresponds roughly to COP = 6.074 x10^6 m^3. Table 7 shows the statistical details of the comparison.

Table 7 Optimal solution analysis based on 10 optimization runs

	Average (m^3)	**Std.**	**Max. (m^3)**	**Min. (m^3)**
GSA	6.059 x 10^6	15,640	6.074 x 10^6	6.032 x 10^6
GA	6.016 x 10^6	34,231	6.069 x 10^6	5.959 x 10^6

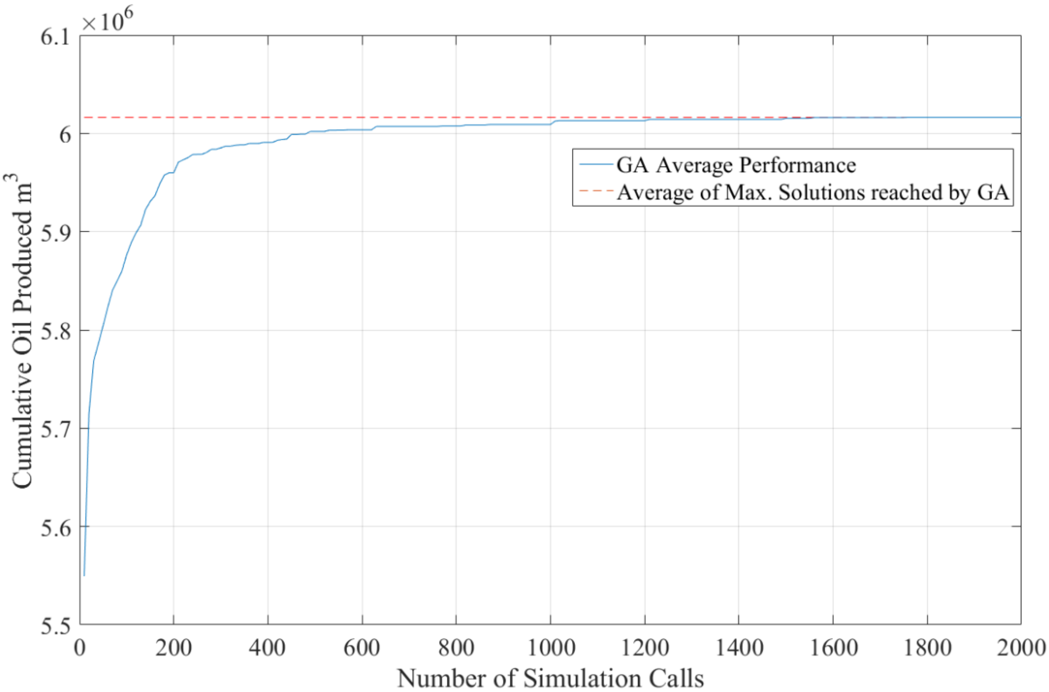


Figure 20: GA average performance over 10 optimization runs for optimal well placement of 3 production wells in PUNQ-S3 model

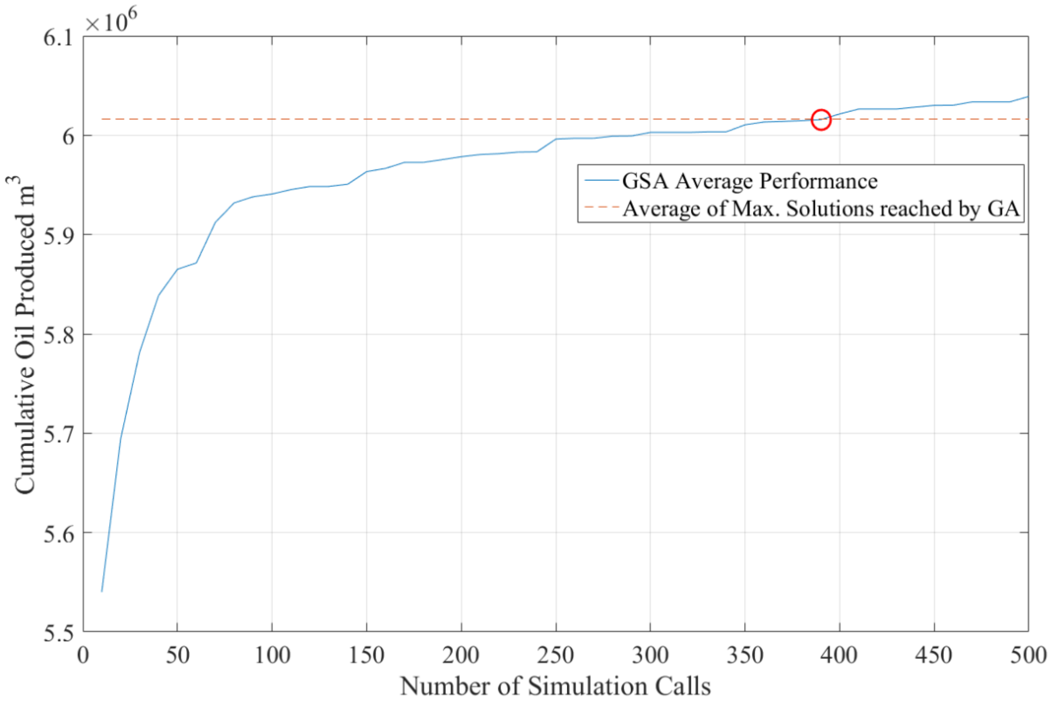


Figure 21: GSA algorithm average performance over 10 optimization runs for optimal well placement of 3 production wells in PUNQ-S3 model compared with the average of max solutions reached by GA

Figure 22 shows the optimal solution obtained by GSA for 3 added production wells under the experiment conditions.

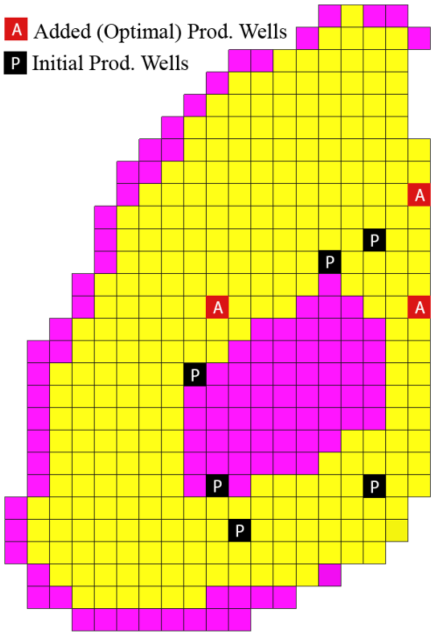


Figure 22: Optimal solution obtained by GSA for 3 added production wells under the experiment conditions

2.2 Brugge Model

The Brugge field [39] is a (139x48x9) grid-blocks synthetic oil field, surrounded by an inactive aquifer. The field initially has 30 wells (20 production and 10 injection wells). The structure of the field consists of an E-W elongated half-dome with a large boundary fault at its northern edge (NBF), and one internal fault with a modest throw at an angle of some 20 degrees to the NBF. The dimensions of the field are roughly 10x3 km. Figure 23 shows the 3D model of Brugge field, and Figure 24 shows the porosity distribution in the field.

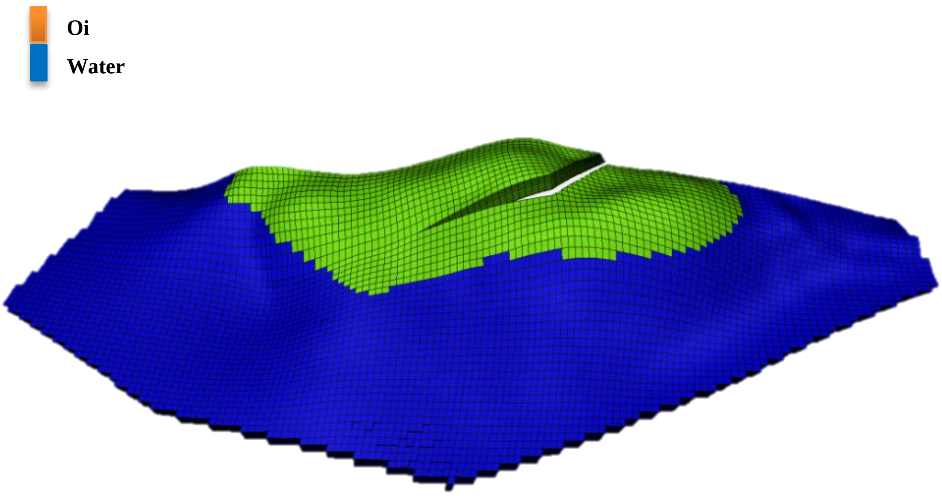


Figure 23: Brugge Oil Field Model

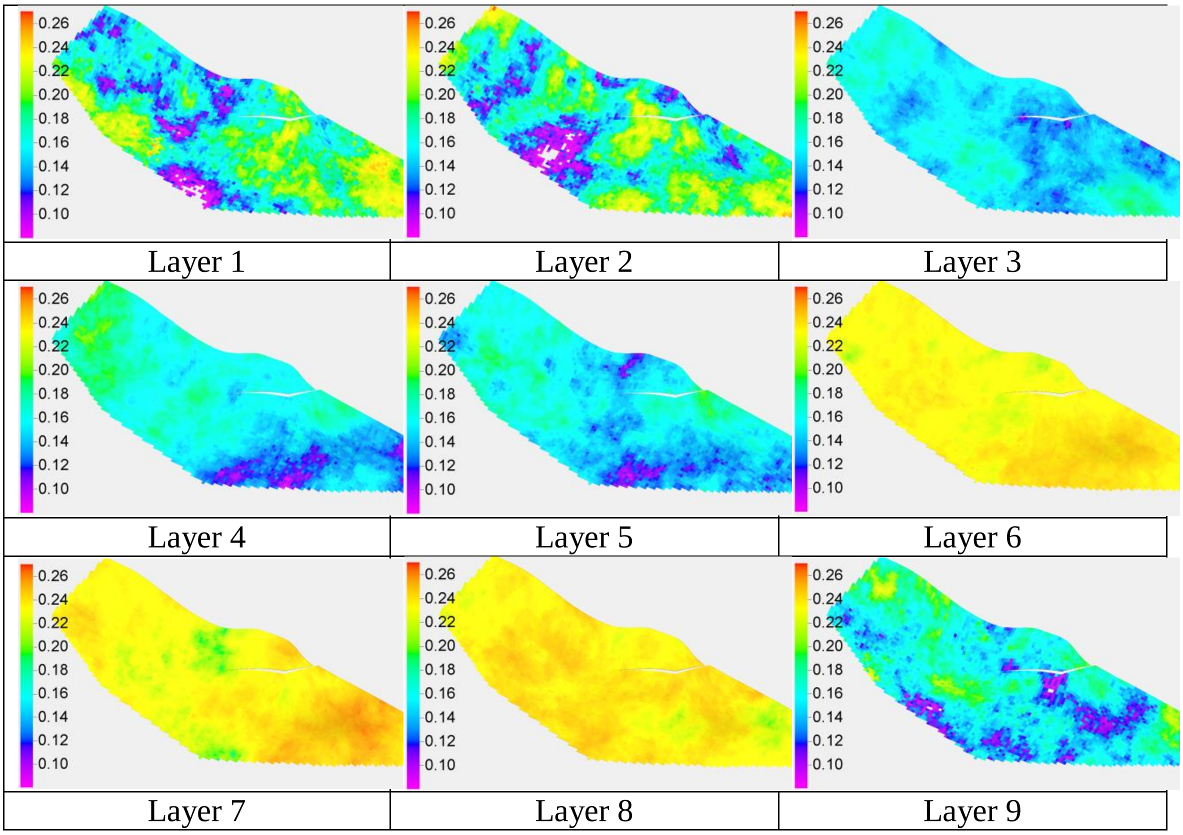


Figure 24: Porosity Distribution in Brugge model

2.2.1 Optimal Allocation for Injection Wells:

In this example, slight modifications to the Brugge field well formation are introduced. The modified Brugge model in this study will have initially 2 injection wells only and 20 production wells.

For this model, the objective is optimizing the allocation of 5 new injection wells over a period of 20 years. The production wells will operate at a fixed flow rate up to $320\ m^3/day$ and BHP pressure equivalent to $50\ bar$. On the other hand, the initial and optimized injection wells operate at $650\ m^3/day$ and $500\ m^3/day$ respectively with a BHP pressure set up to $180\ bar$.

The objective function in this assessment is the total Net Present Value (NPV) over the aforementioned period of production. As a single realization of porosity and permeability was considered, the NPV can be defined as follows:

$$NPV = \sum_{i=1}^{20} \frac{p_{oil}\, q_{oil}^{i} - [\, p_{wI}\, q_{wI}^{i} + p_{wP}\, q_{wP}^{i}]}{(1+\gamma)^{i}} \tag{3}$$

Whereby p_{oil} and q_{oil}^{i} are oil price (80 USD/bbl) and total oil produced (bbl/year) per year i; p_{wI} and p_{wP} are the prices for the water injected and water produced respectively ($p_{wI} = p_{wP}$ = 5 USD/bbl); q_{wI}^{i} and q_{wP}^{i} are the total water injected and produced (bbl/year). The yearly discount rate was taken as (γ = 10%) [39].

A set of 10 complete optimization runs (total of 20,000 simulation runs) were performed using each algorithm (GA and GSA). The population size in the assessment was set to 20 individuals and the maximum number of generations (termination criteria) is 100. The search space shown in Figure 23 was constrained to cover the area saturated with oil as well as a small portion of the aquifer. Both algorithms had the same initial population seeded in each of the 10 optimization runs. The setup for both algorithms is shown in Table 8. The average performance curve of both algorithms at each generation is reported Figure 25, and the end results statistical details are reported in Table 9. It's evident that GSA outperformed GA with a two times faster convergence rate and a more robust solution. Figure 26 graphically presents the optimal solution obtained by GSA for 5 added injection wells under the experiment conditions

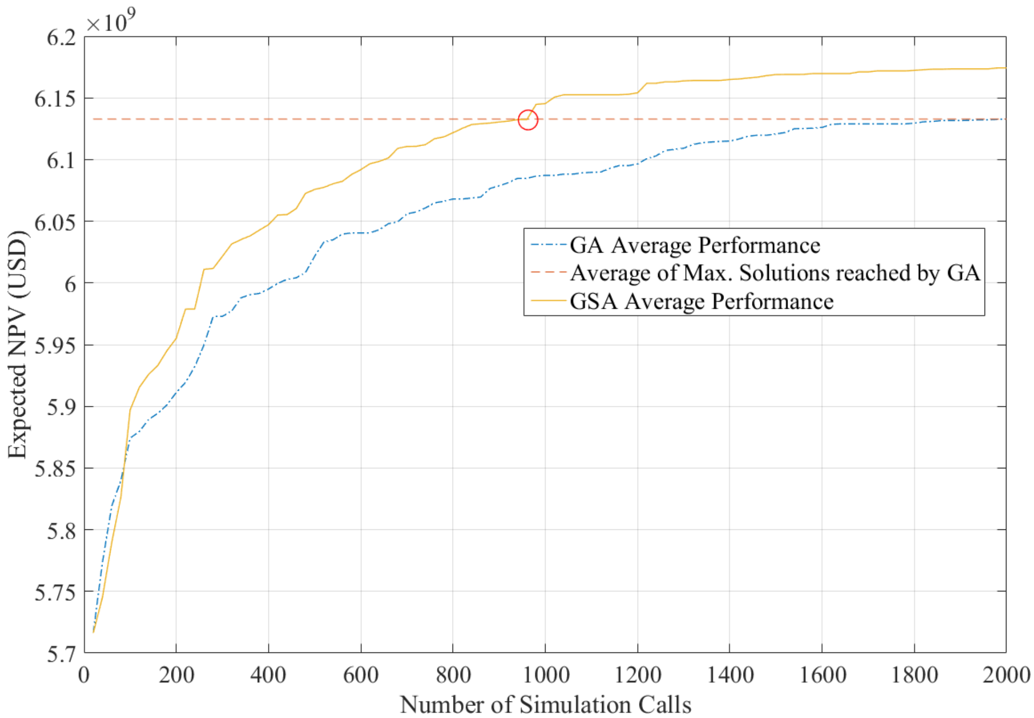


Figure 25: GSA and GA algorithms average performance over 10 optimization runs for optimal well placement of 5 injection wells in Brugge model

Table 8 GA and GSA setup for solving multi injection wells placement problem

	Genetic Algorithm	Genetic Similarity Algorithm
Pop. Size	20	20
Elite	5%	5%
Crossover	75%	45%
Mutation	20%	20%
Similarity	-	30%

Table 9 Optimal solution analysis based on 10 optimization runs

	Average (USD)	Std.	Max. (USD)	Min. (USD)
GSA	6.174×10^{9}	1.467×10^{7}	6.197×10^{9}	6.151×10^{9}
GA	6.133×10^{9}	2.928×10^{7}	6.188×10^{9}	6.094×10^{9}

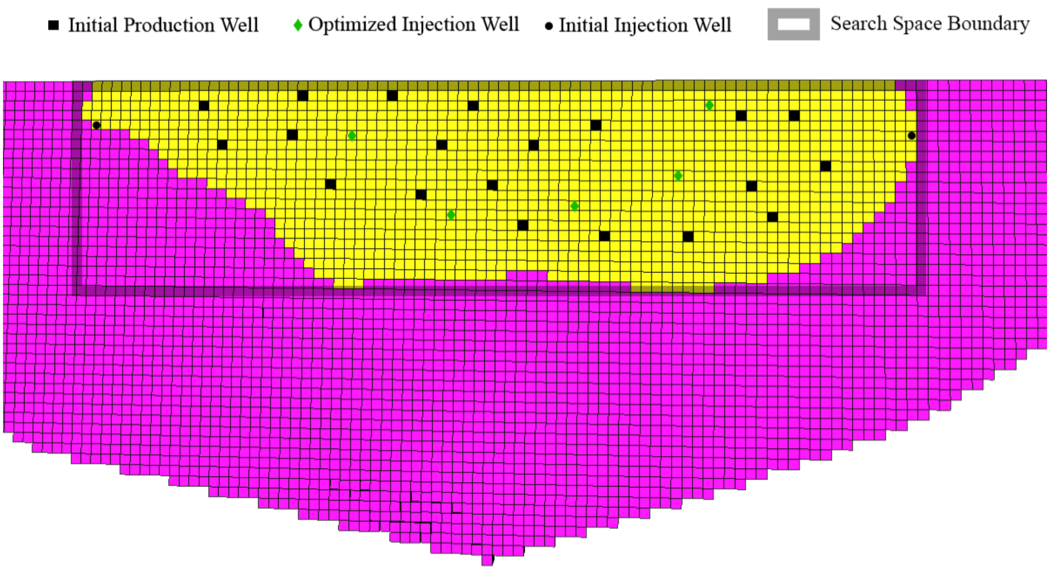


Figure 26: Optimal solution obtained by GSA for 5 added injection wells under the experiment conditions

CHAPTER 5

CONCLUSIONS

In this study, a novel genetic operator within GA is introduced to design a problem-specific search algorithm named Genetic Similarity Algorithm (GSA). The performance of GSA is compared against standard GA in solving the well placement problem in oil fields. The comparison metrics were the convergence rate as well as solution robustness. A variety of example problems were investigated, involving placement of injection and production wells in two reservoir models. A first example tackled the simple case of optimal allocation of a single well, and showed the better convergence rate of GSA compared to other approaches. Subsequently, more difficult problems were addressed, whereby two cases for multi well placement were considered in the two different reservoir models. Not only, GSA has outperformed GA's performance in convergence rate and solution robustness in all studied cases, it also proved to be less prone to straying (premature convergence).

Although GSA proved efficient for optimal well placement in the investigated examples, it can be inferred from the structure of the presented algorithm that increasing the problem dimensionality (i.e. the number of wells to be optimized) may limit the search efficiency achieved by the Similarity Operator. This is attributed to the increase in the number of solutions with high similarity to the elite solution. In such a case, if the elite solution remained the same for a high number of generations, the solutions generated by the Similarity Operator will diverge from the elite solution at a slower rate. Consequently, the exploration efficiency of the operator will decrease. Nevertheless, a slight improvement in the performance may still hold over the standard approaches as the operator will still contribute with diverse and potentially good solutions to the next generations.

The results presented in this thesis demonstrate that the addition of the proposed Similarity Operator can significantly improve convergence rate as well as solution robustness at an insignificant computational cost, which is only required initially to compute the similarity between the features representing the various cells.

Bibliography

[1] P. Glover, "Formation Evaluation MSc Course Notes," in *Chapter 3: Reservoir Drives*, University of Leeds.

[2] F. Floris, M. Bush, M. Cuypers, F. Roggero and A. R. Syversveen, "Methods for quantifying the uncertainty of production forecasts: a comparative study," *Petroleum Geoscience*, vol. 7, no. S, pp. S87-S96, 2001.

[3] Y. Gu and D. S. Oliver, "History Matching of the PUNQ-S3 Reservoir Model Using the Ensemble Kalman Filter," *SPE Journal*, vol. 10, no. 2, pp. 217 - 224, 2005.

[4] F. Roggero and L. Y. Hu, "Gradual Deformation of Continuous Geostatistical Models for History Matching," in *SPE Annual Technical Conference and Exhibition*, New Orleans, Louisiana, 1998.

[5] S. I. Aanonsen, A. L. Eide, L. Holden and J. O. Aasen, "Optimizing Reservoir Performance Under Uncertainty with Application to Well Location," in *SPE Annual Technical Conference and Exhibition*, Dallas, Texas, 1995.

[6] S. Afsharia, B. Aminshahidy and M. R. Pishvaie, "Application of an improved harmony search algorithm in well placement optimization using streamline simulation," *Journal of Petroleum Science and Engineering*, vol. 78, no. 3-4, pp. 664-678, 2011.

[7] A. A. Awotunde and C. Naranjo, "Well Placement Optimization Constrained to

Minimum Well Spacing," in *SPE Latin America and Caribbean Petroleum Engineering Conference*, Maracaibo, Venezuela, 2014.

[8] H. Wang, D. Echeverría-Ciaurri, L. Durlofsky and A. Cominelli, "Optimal Well Placement Under Uncertainty Using a Retrospective Optimization Framework," *SPE Journal*, vol. 17, no. 01, pp. 112 - 121, 2012.

[9] B. Güyagüler, R. N. Horne, L. Rogers and J. J. Rosenzweig, "Optimization of Well Placement in a Gulf of Mexico Waterflooding Project," *SPE Reservoir Evaluation & Engineering*, vol. 5, no. 3, pp. 229 - 236, 2002.

[10] P. S. da Cruz, R. N. Horne and C. V. Deutsch, "The Quality Map: A Tool for Reservoir Uncertainty Quantification and Decision Making," in *SPE Annual Technical Conference and Exhibition*, Houston, Texas, 1999.

[11] J. Lyons and H. Nasrabadi, "Well placement optimization under time-dependent uncertainty using an ensemble Kalman filter and a genetic algorithm," *Journal of Petroleum Science and Engineering*, vol. 109, p. 70–79, 2013.

[12] M. Handels, M. Zandvliet, R. Brouwer and J. D. Jansen, "Adjoint-Based Well-Placement Optimization Under Production Constraints," in *SPE Reservoir Simulation Symposium*, Houston, 2007.

[13] P. Sarma and W. H. Chen, "Efficient Well Placement Optimization with Gradient-based Algorithms and Adjoint Models," in *Intelligent Energy Conference and Exhibition*, Amsterdam, 2008.

[14] A. C. Bittencourt and R. N. Horne, "Reservoir Development and Design

Optimization," in *SPE Annual Technical Conference and Exhibition*, San Antonio, 1997.

[15] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: a review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, no. 3, p. 1247–1293, 2013.

[16] G. Montes, P. Bartolome and A. L. Udias, "The Use of Genetic Algorithms in Well Placement Optimization," in *SPE Latin American and Caribbean Petroleum Engineering Conference*, Buenos Aires, 2001.

[17] J. E. Onwunalu and L. J. Durlofsky, "Application of a particle swarm optimization algorithm for determining optimum well location and type," *Computational Geosciences,* vol. 14, no. 1, pp. 183-198, 2009.

[18] M. Mahdavi, M. Fesanghary and E. Damangir , "An improved harmony search algorithm for solving optimization problems," *Applied Mathematics and Computation*, vol. 188, no. 2, p. 1567–1579, 2007.

[19] L. Li and B. Jafarpour, "A variable-control well placement optimization for improved reservoir development," *Computational Geosciences,* vol. 16, no. 4, p. 871–889, 2012.

[20] Y. Chang, Z. Bouzarkouna and D. Devegowda, "Multi-objective optimization for rapid and robust optimal oilfield development under geological uncertainty," *Computational Geosciences,* vol. 19, no. 4, pp. 933-950, 2015.

[21] K. Rashid, W. J. Bailey, B. Couet and D. Wilkinson, "An Efficient Procedure for

Expensive Reservoir-Simulation Optimization Under Uncertainty," *SPE Economics & Management*, vol. 5, no. 4, pp. 21 - 33, 2013.

[22] S. Mishra, M. K. Choudhary and A. Datta-Gupta, "A Novel Approach for Reservoir Forecasting Under Uncertainty," *SPE Reservoir Evaluation & Engineering*, vol. 5, no. 1, pp. 42-48, 2002.

[23] M. G. Shirangi and L. J. Durlofsky, "Closed-Loop Field Development Under Uncertainty by Use of Optimization With Sample Validation," *SPE Journal*, vol. 20, no. 5, pp. 908 - 922, 2015.

[24] K. Deb and N. Padhye, "Enhancing performance of particle swarm optimization through an algorithmic link with genetic algorithms," *Computational Optimization and Applications*, vol. 57, no. 3, pp. 761-794, 2014.

[25] N. Padhye, P. Bhardawaj and K. Deb, "Improving differential evolution through a unified approach," *Journal of Global Optimization*, vol. 54, no. 5, pp. 771-799, 2013.

[26] J. H. Holland, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, University of Michigan Press, 1975.

[27] D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," in *Foundations of genetic algorithms*, 2013, pp. 69-93.

[28] R. G. Regis, "Evolutionary Programming for High-Dimensional Constrained Expensive Black-Box Optimization Using Radial Basis Functions," *IEEE*

Transactions on Evolutionary Computation, vol. 18, no. 3, 2014.

[29] K. Rashida, S. Ambanib and E. Cetinkayac, "An Adaptive Multiquadric Radial Basis Function Method For Expensive Black-Box Mixed-Integer Nonlinear Constrained Optimization," *Engineering Optimization,* vol. 45, no. 2, pp. 185-206, 2013.

[30] H. -M. Gutmann, "A Radial Basis Function Method for Global Optimization," *Journal of Global Optimization,* vol. 19, pp. 201-227, 2001.

[31] V. M. Johnson and L. L. Rogers, "Applying soft computing methods to improve the computational tractability of the subsurface simulation-optimization problem," *Journal of Petroleum Science and Engineering,* vol. 29, pp. 153-175, 2001.

[32] S. Russell and P. Norvig, Artificial intelligence: a modern approach, Prentice Hall, 1995.

[33] V. Vapnik, The Nature of Statistical Learning Theory, New York: Springer, 1995.

[34] E. W. Forgy, "Cluster analysis of multivariate data: efficiency versus interpretability of classifications," *Biometrics,* no. 21, p. 768–769, 1965.

[35] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory,* vol. 13, no. 1, pp. 21 - 27, 1967.

[36] Schlumberger, "ECLIPSE* reservoir simulation software," Schlumberger, 2011.

[37] Y. Pan and R. N. Horne, "Improved Methods for Multivariate Optimization of Field Development Scheduling and Well Placement Design," in *SPE Annual Technical Conference and Exhibition*, New Orleans, Louisiana, 1998.

[38] A. Centilmen, T. Ertekin and A. S. Grader, "Applications of Neural Networks in Multiwell Field Development," in *SPE Annual Technical Conference and Exhibition*, Houston, Texas, 1999.

[39] L. Peters, R. Arts, G. Brouwer, C. Geel , S. Cullick, R. J. Lorentzen, Y. Chen, N. Dunlop, F. C. Vossepoel, R. Xu, P. Sarma, A. H. Alhuthali and A. Reynolds, "Results of the Brugge benchmark study for flooding optimization and history matching," *Society of Petroleum Engineers,* vol. 13, no. 3, pp. 391 - 405, 2010.

APPENDIX

A. ECLIPSE Input File Sections

Section	Description	Type
RUNSPEC	Title, problem dimensions, switches, phases present, components etc.	*Required*
GRID	Specification of geometry of computational grid (location of grid block corners), and of rock properties (porosity, absolute permeability, etc.) in each grid block.	*Required*
EDIT	Modifications to calculated pore volumes, grid block center depths and transmissibilities.	*Optional*
PROPS	Tables of properties of reservoir rock and fluids as functions of fluid pressures, saturations and compositions (density, viscosity, relative permeability, capillary pressure, etc.). Contains the equation of state description in compositional runs.	*Required*
REGIONS	Splits computational grid into regions for calculation of; • PVT properties (Fluid densities and viscosities) • Saturation properties (Relative permeabilities and capillary pressures) • Initial conditions (Equilibrium pressures and saturations) • Fluids in place (Fluid in place and inter-region flows) • EoS regions (For compositional runs) • If this section is omitted, all grid blocks are put in region 1	*Optional*
SOLUTION	Specification of initial conditions in reservoir - may be: • Calculated using specified fluid contact depths to give potential equilibrium • Read from a restart file set up by an earlier run • Specified by the user for every grid block (Not recommended for general use)	*Required*
SUMMARY	Specification of data to be written to the Summary file after each time step. Necessary if certain types of graphical output (for example water-cut as a function of time) are to be generated after the run has finished. If this section is omitted no Summary files are created.	*Optional*
SCHEDULE	Specifies the operations to be simulated (production and injection controls and constraints) and the times at which output reports are required. Vertical flow performance curves and simulator tuning parameters may also be specified in the SCHEDULE section.	*Required*

B. User Interface Design

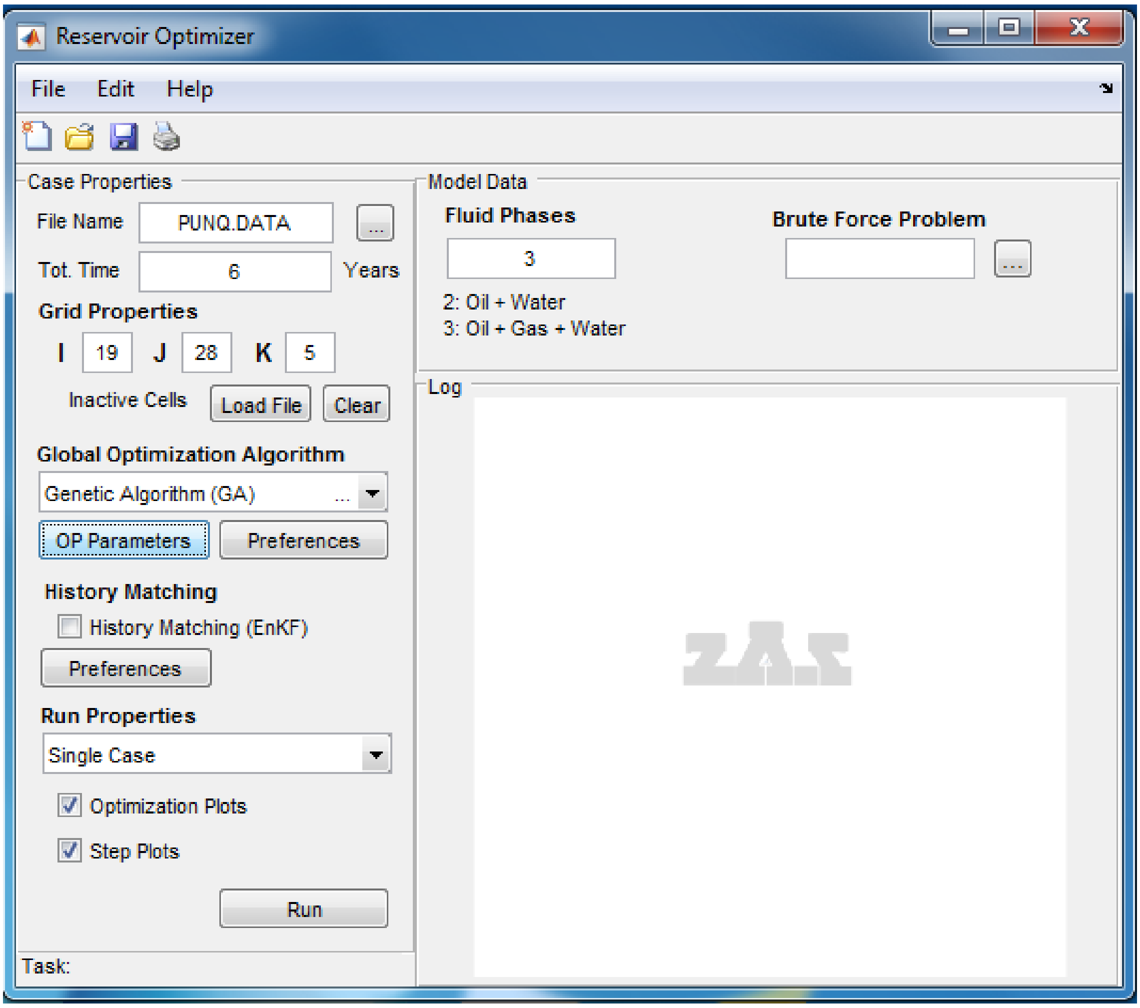


Figure 27 Main Interface for Reservoir Optimizer Software

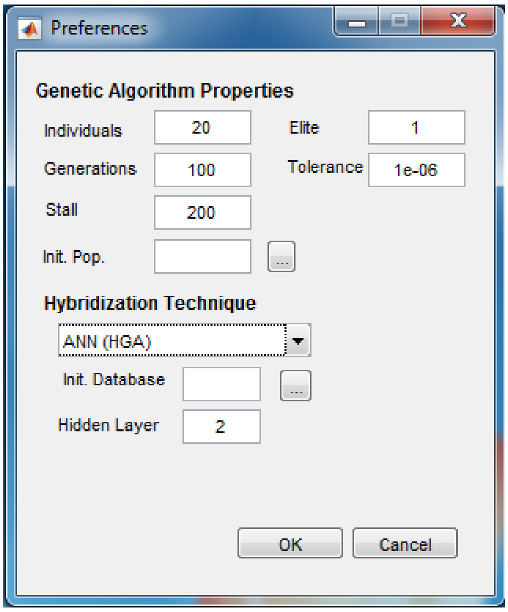


Figure 28 Search Algorithm Preferences

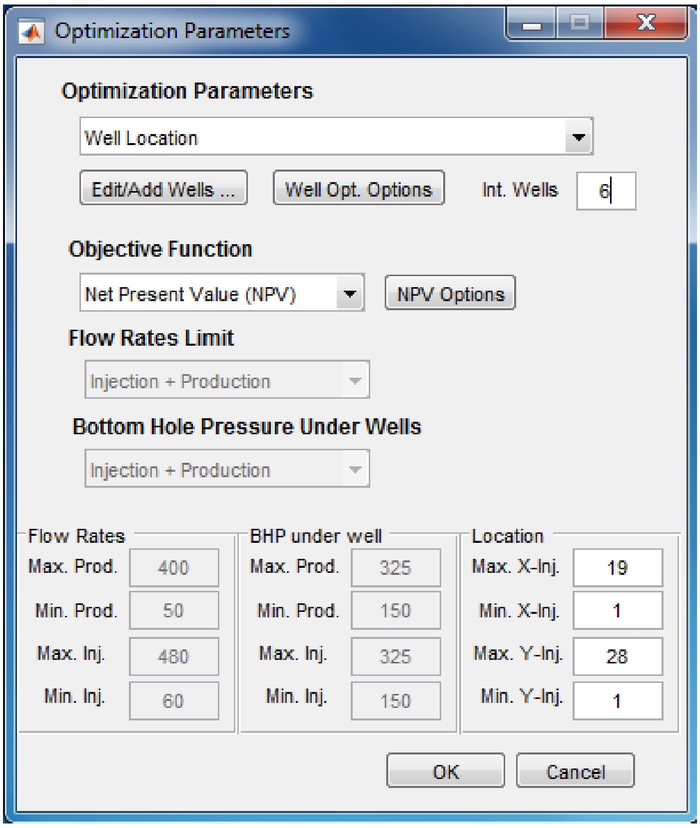


Figure 29 Specifying Optimization Parameters and Constraints

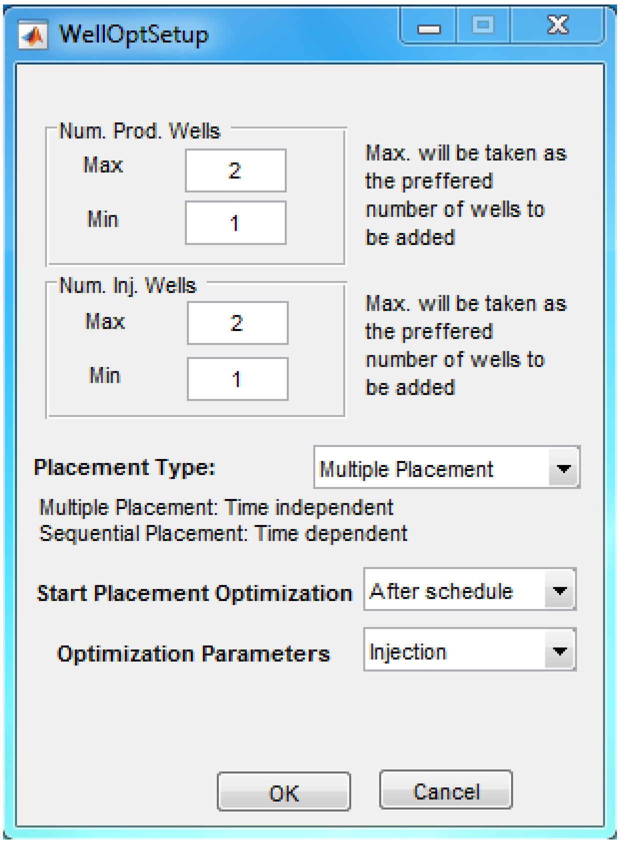


Figure 30 Optimization Parameters and Constraints for the Well Placement Problem

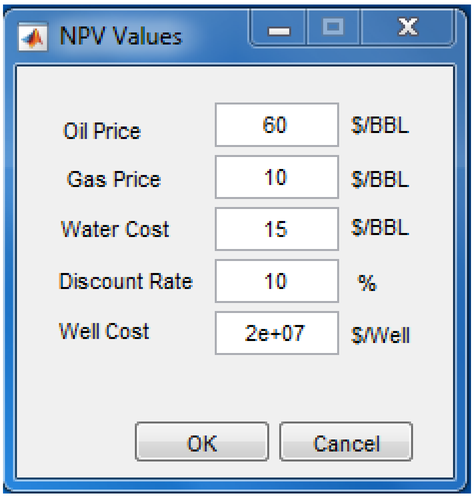


Figure 31 Net Present Value (NPV) Parameters

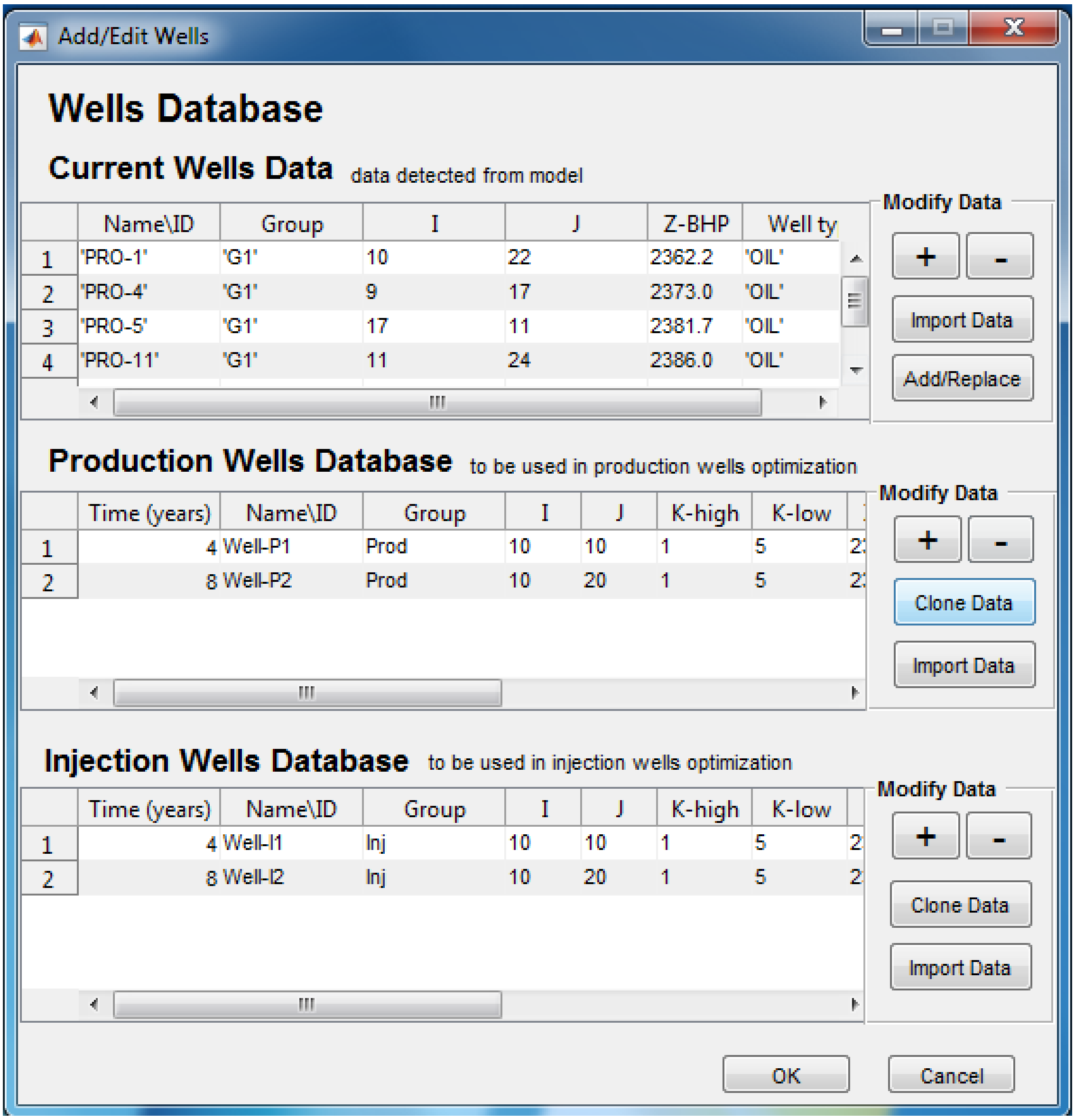


Figure 32 Current Wells and the Initial Setup for Added Wells