An Efficient Geometry-based Optimization Approach for Well Placement in Oil Fields

Zachary Hamida[1], Fouad Azizi[2], George Saad[3,*]

[1] Graduate Program in Computational Science, American University of Beirut, Beirut, Lebanon, [2]Department of Chemical and Petroleum Engineering, American University of Beirut, Beirut, Lebanon, [3]Department of Civil and Environmental Engineering, American University of Beirut, Beirut, Lebanon

Abstract

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.

Keywords: Well placement, reservoir simulation, Genetic Algorithm, Similarity Operator

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. Several reservoir engineering problems were addressed in the literature, and a big proportion was devoted for the well placement problem. Prioritizing the well placement problem is due to the high costs following decisions related to drilling and adding new wells. This problem is commonly formulated as an integer programing problem, whereby the optimization variables are the indices of the reservoir model cells.

Different optimization algorithms were suggested to solve this problem (Handels, Zandvliet, Brouwer, & Jansen, 2007; Sarma & Chen, 2008; Bittencourt & Horne, 1997; Güyagüler, Horne, Rogers, & Rosenzweig, 2002). The efficiency of these algorithms was measured by solution robustness, convergence rate and the total computational cost of the process. Handles *et al.* (Handels, Zandvliet, Brouwer, & Jansen, 2007) and Sarma *et al.* (Sarma & Chen, 2008) applied gradient-based search with variations to account for the high heterogeneity of the search space. 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 these aforementioned reasons, derivative-free algorithms present themselves as a more preferable option in solving the problem. Derivative-free search algorithms can be mainly categorized into 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) (Rios & Sahinidis, 2013). Different population-based algorithms were applied in the literature to solve the problem of well placement in oil fields (Montes, Bartolome, & Udias, 2001; Güyagüler, Horne, Rogers, & Rosenzweig, 2002; Onwunalu & Durlofsky, 2009; Afsharia, Aminshahidy, & Pishvaie, 2011). Montes *et al.* (Montes, Bartolome, & Udias, 2001) 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.* (Onwunalu & Durlofsky, 2009) applied a Particle Swarm Optimization (PSO) algorithm to search for optimal well location and type (production or injection). Afshari *et al.* (Afsharia, Aminshahidy, & Pishvaie, 2011) assessed the performance of an Improved Harmony Search (IHS) algorithm (Mahdavi *et al.* (Mahdavi, Fesanghary , & Damangir , 2007)), which has a better local search performance than the standard HS, in solving the well placement problem. Variations to these algorithms were also introduced aiming at improving the convergence rate at a minimum computational cost. Bittencourt *et al.*

(Bittencourt & Horne, 1997) used a hybrid algorithm of GA and polytope method to solve for well placement. Da cruz *et al.* (da Cruz, Horne, & Deutsch, 1999) 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.* (Güyagüler, Horne, Rogers, & Rosenzweig, 2002) 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 (Li & Jafarpour, 2012; Awotunde & Naranjo, 2014). 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 doesn't contribute in finding new good solutions.

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.

2. Genetic Algorithm

Introduced by Holland et al. (Holland, 1975), 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 1 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.

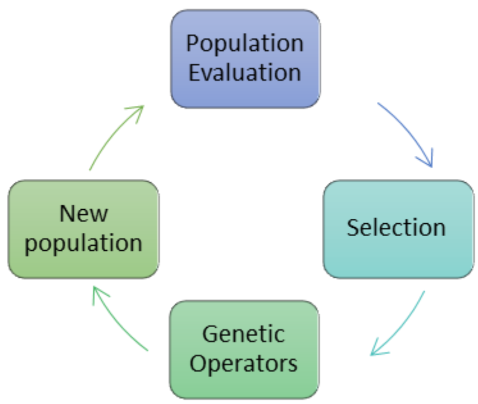


Figure 1: Genetic Algorithm Structure

2.1. 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.) (Goldberg & Deb, 2013), however, the choice of a selection technique is highly dependent on the variation in the fitness function values.

2.2. 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 moves 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.

3. Genetic Similarity Algorithm (GSA)

The proposed search algorithm presented in Figures 2a and 2b 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.

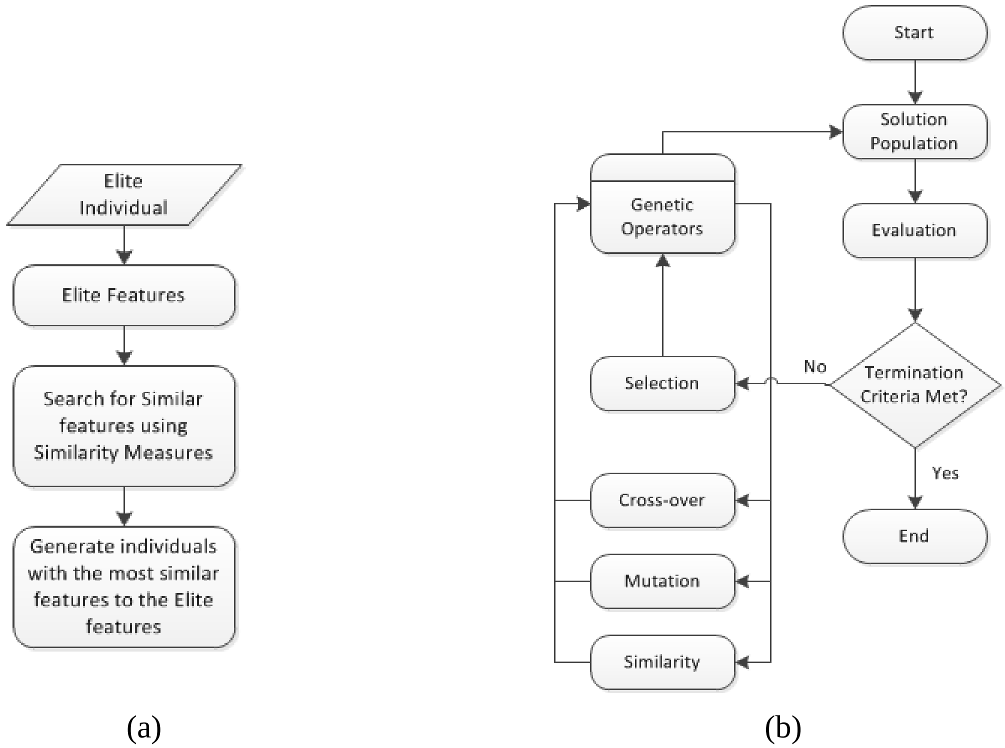


Figure 2: (a) Similarity Operator Flow chart; (b) Genetic Similarity Algorithm

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:

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

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) (Forgy, 1965) and classification (i.e. kNN) (Cover & Hart, 1967). In the proposed approach, we used similarity measures to identify locations in the reservoir model sharing similar features with a given location.

3.2. Similarity Operator

The proposed operator shown in Figure 2a 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 3 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.

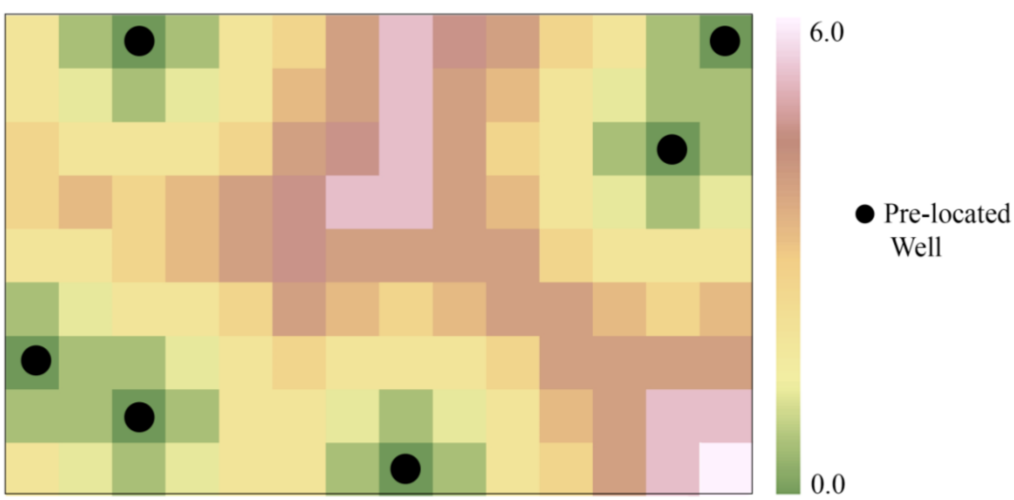


Figure 3: Distance for each cell to nearest pre-located well (example data)

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 4 shows a porosity data set for the same well formation example shown above.

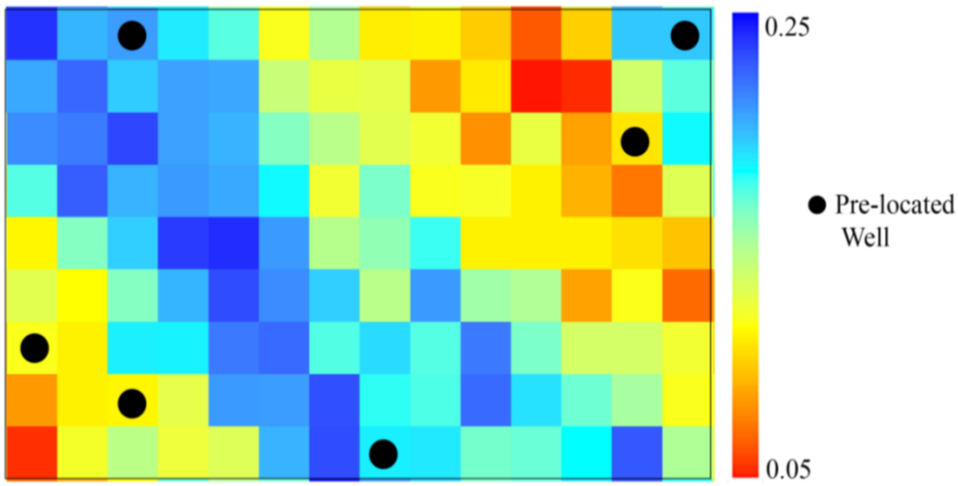


Figure 4: Porosity raster for a reservoir (example data)

Combining and normalizing the two aforementioned features in a single scatter plot (Figure 5) can provide a non-biased visual representation for the selected features (properties) in the reservoir model.

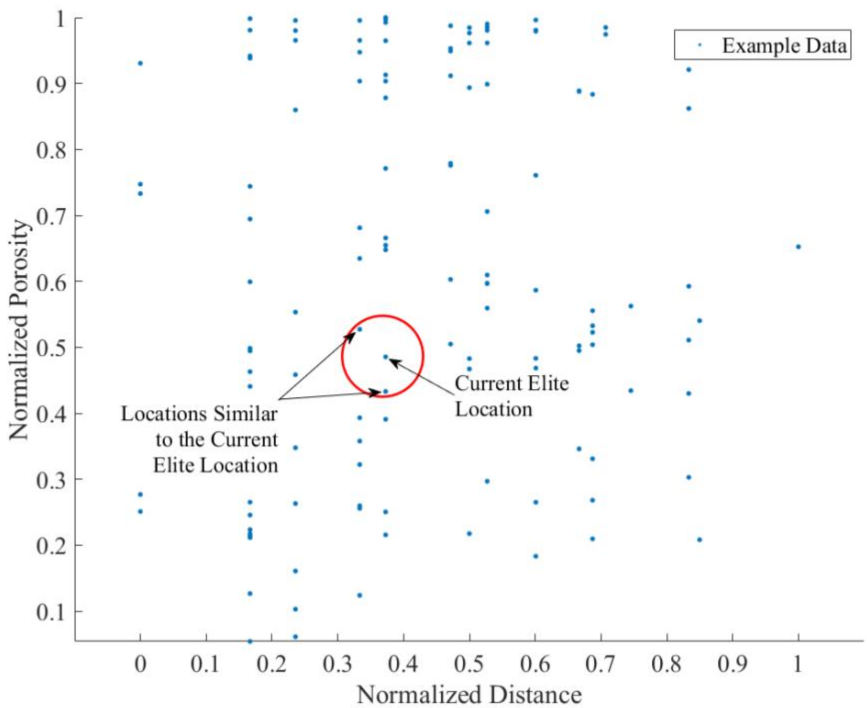


Figure 5: Scatter Plot for Normalized Distance to Nearest Well vs. Normalized Porosity (example data)

Figure 5 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 shown in Figure 2a 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:

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

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 6 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 6: Pseudo code for the similarity operator

4. Numerical Examples

The efficiency of the presented methodology will be tested on two different models, the PUNQ-S3 oil field model and the Brugge oil field model. For each of the two cases, the optimization framework is fully developed under the MATLAB environment and the reservoir simulator used in this study is ECLIPSE by Schlumberger (Schlumberger, 2011).

4.1. 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 (Floris, Bush, Cuypers, Roggero, & Syversveen, 2001). 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 7 and 8 show the field geometry and porosity distribution for the PUNQ-S3 model, respectively.

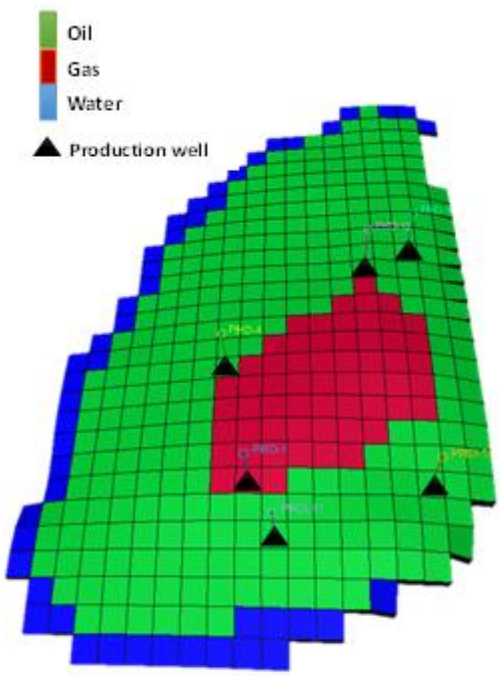


Figure 7: PUNQ-S3 Oil Field Model

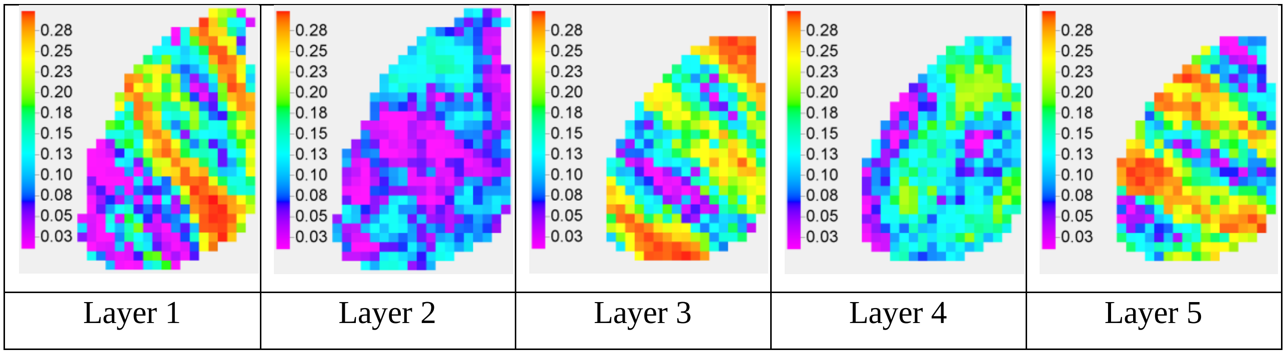


Figure 8: Porosity Distribution in PUNQ-S3 model

4.1.1. Optimal Allocation for a Single Injection Well: 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) corresponding to cumulative oil produced COP = 6.357 ξ 10^6 m^3. Figure 9 shows a response surface for the COP resulting from placing the injection well at each cell in the model.

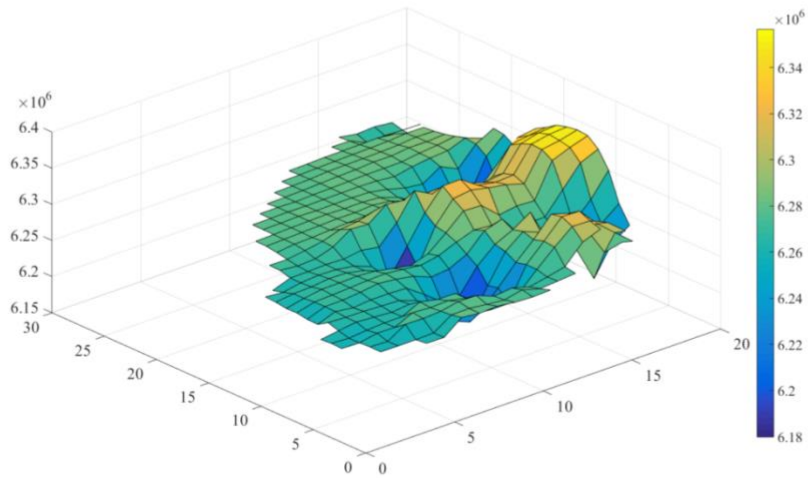


Figure 9: PUNQ-S3 Single Injection Well Allocation Response Surface

After identifying the optimal location, the convergence rate of the proposed approach compared to other search frameworks used in the literature (Montes, Bartolome, & Udias, 2001; Aanonsen, Eide, Holden, & Aasen, 1995; Pan & Horne, 1998; Centilmen, Ertekin, & Grader, 1999), 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 Setup

Genetic Algorithm		
Pop. Size	10	20
Elite	10%	5%
Crossover	70%	75%
Mutation	20%	20%

Table 2 Surrogate Models Setup

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

Table 3 Genetic Similarity Algorithm different setup cases

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 reported for each run. Comparison results for the different cases are 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 and % is the percentage of times the algorithm converged to an optimal solution under the given stopping criteria.

Table 4 Comparing Results for Population Size = 10

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

Table 5 Comparing Results for Population Size = 20

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

Figures 10 and 11 illustrate the improvement achieved by using the proposed Genetic Similarity Algorithm compared to the standard Genetic Algorithm. This comparison was chosen because GA similarly to GSA has no dependency on a surrogate model.

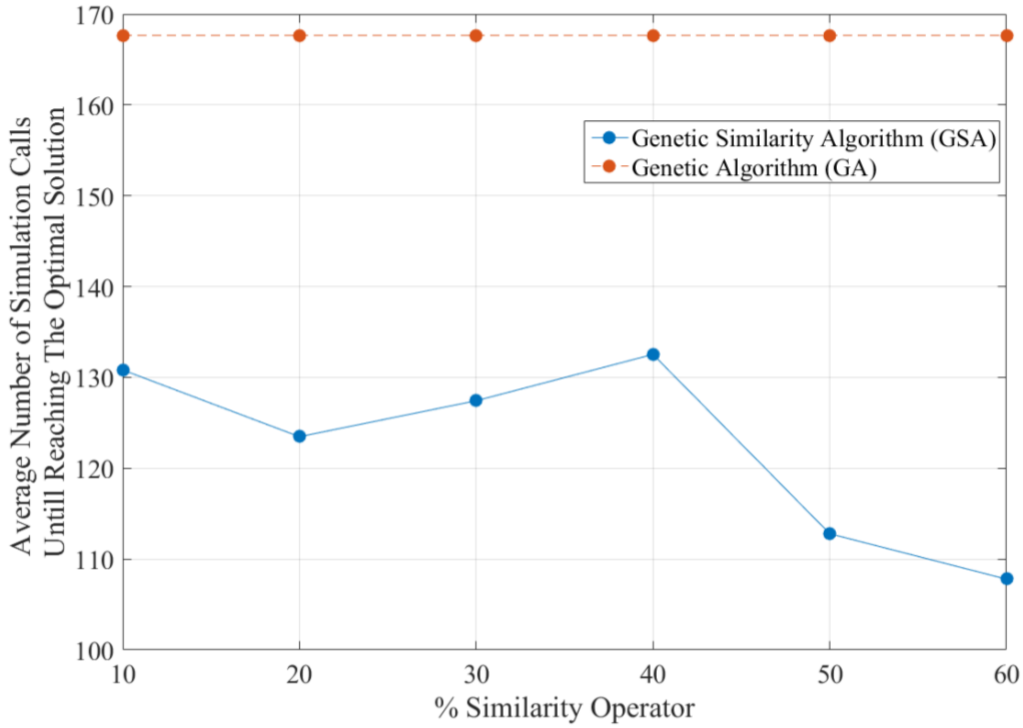


Figure 10: Genetic Algorithm vs. Genetic Similarity Algorithm with different fractions of Similarity Operator for Pop. Size = 10

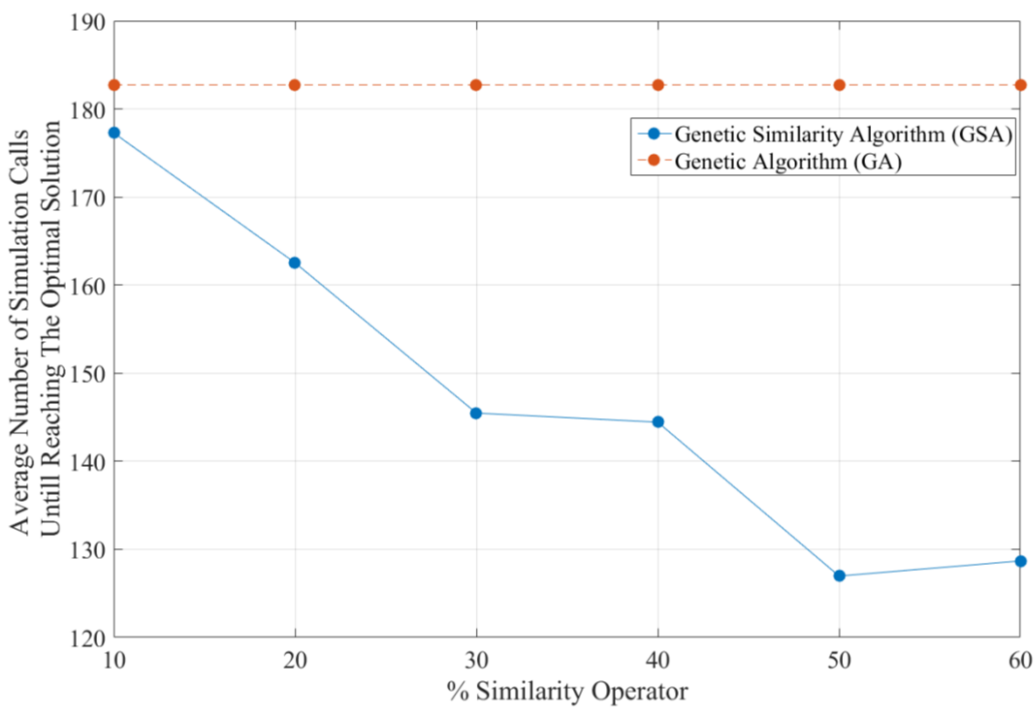
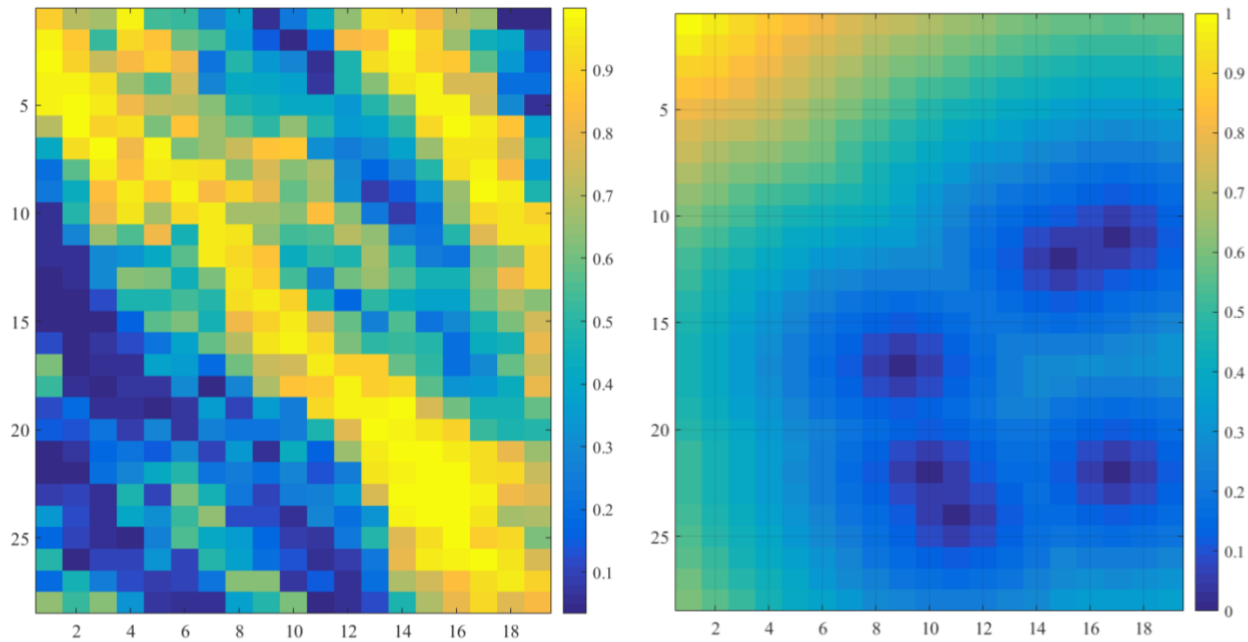


Figure 11: Genetic Algorithm vs. Genetic Similarity Algorithm with different fractions of Similarity Operator for Pop. Size = 20

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 12a and 12b show the two selected features representing each cell in PUNQ-S3 model.



(a) (b)

Figure 12: (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 12a) with the distance raster (Figure 12b) through a weighted sum will result into a new raster shown in Figure 13. 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 13) in the raster and thereafter will search and find locations having similar values (i.e. locations S1 and S2) in the raster.

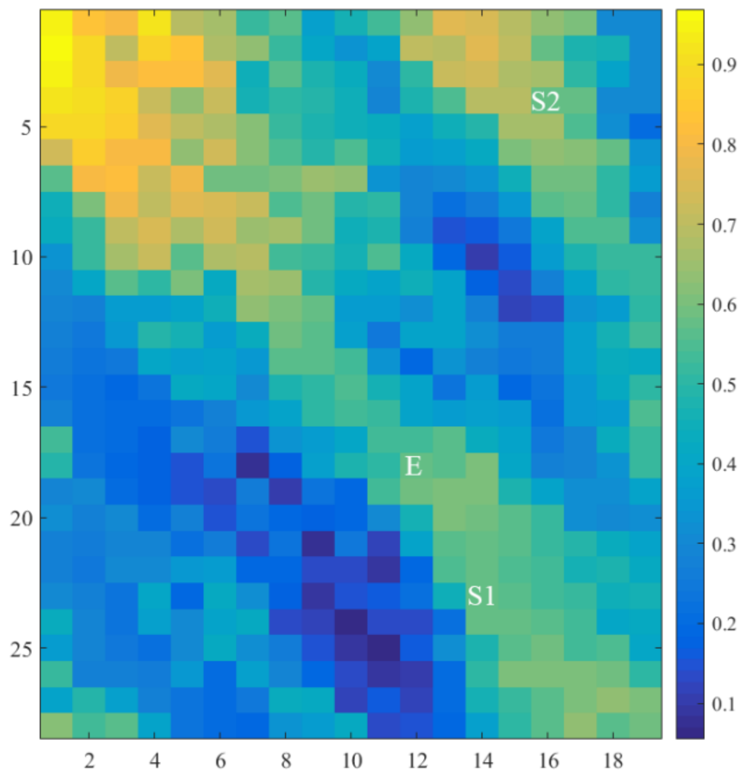


Figure 13: Similarity raster approximated from combining (0.5 NP + 0.5 ND)

It can be noted from Figure 13 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.

***4.1.2. 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 14. The algorithm has reached an optimal solution (on average) after 1760 fitness function evaluations (simulation calls) corresponding to COP = 6.016 ξ 10^6 m^3.

On the other hand, considering the GSA average performance shown in Figure 15, 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 $\xi 10^6$ 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	6.059×10^6	15,640	6.074×10^6	6.032×10^6
GA	6.016×10^6	34,231	6.069×10^6	5.959×10^6

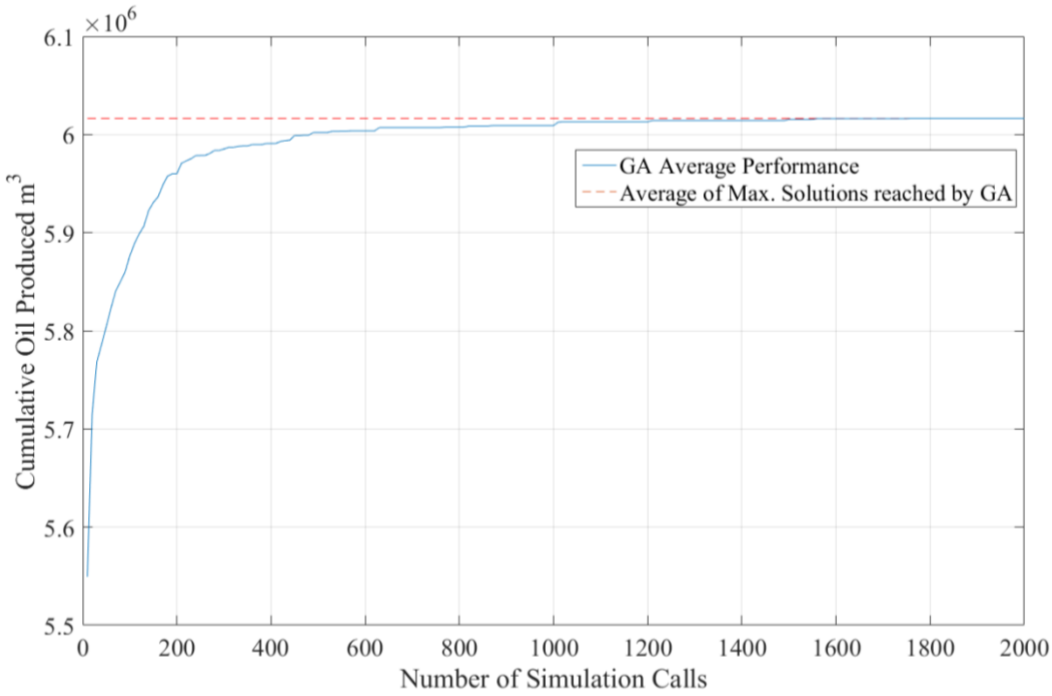


Figure 14: GA average performance over 10 optimization runs for optimal well placement of 3 production wells in PUNQ-S3 model

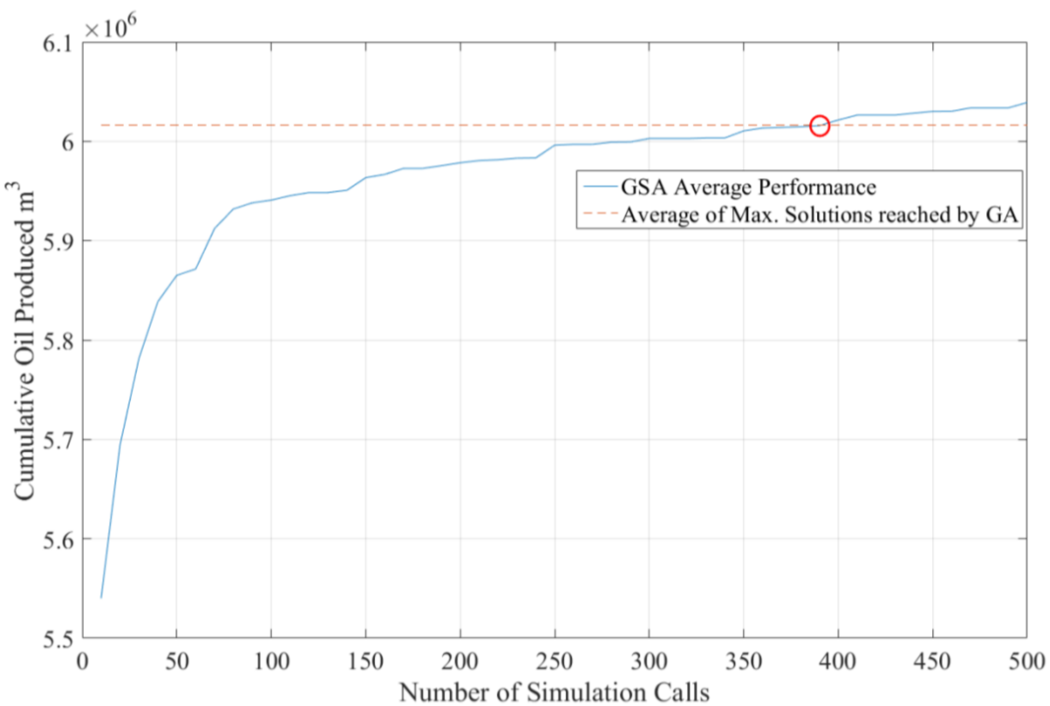


Figure 15: GSA algorithm average performance over 10 optimization runs for optimal well placement of 3 production wells in PUNQ-S3 model compared with the average of max solutions reached by GA

Figure 16 shows the optimal solution obtained by GSA for 3 added production wells under the experiment conditions.

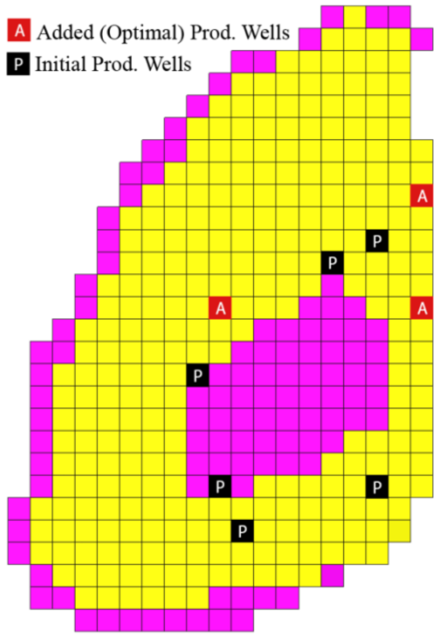


Figure 16: Optimal solution obtained by GSA for 3 added production wells under the experiment conditions

4.2. Brugge Model

The Brugge field (Peters, et al., 2010) is a (139ξ48ξ9) 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 10ξ3 km. Figure 17 shows the 3D model of Brugge field, and Figure 18 shows the porosity distribution in the field.

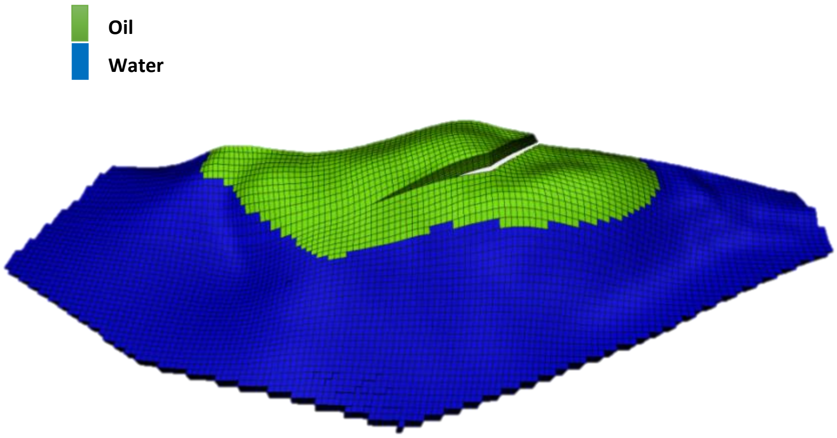


Figure 17: Brugge Oil Field Model

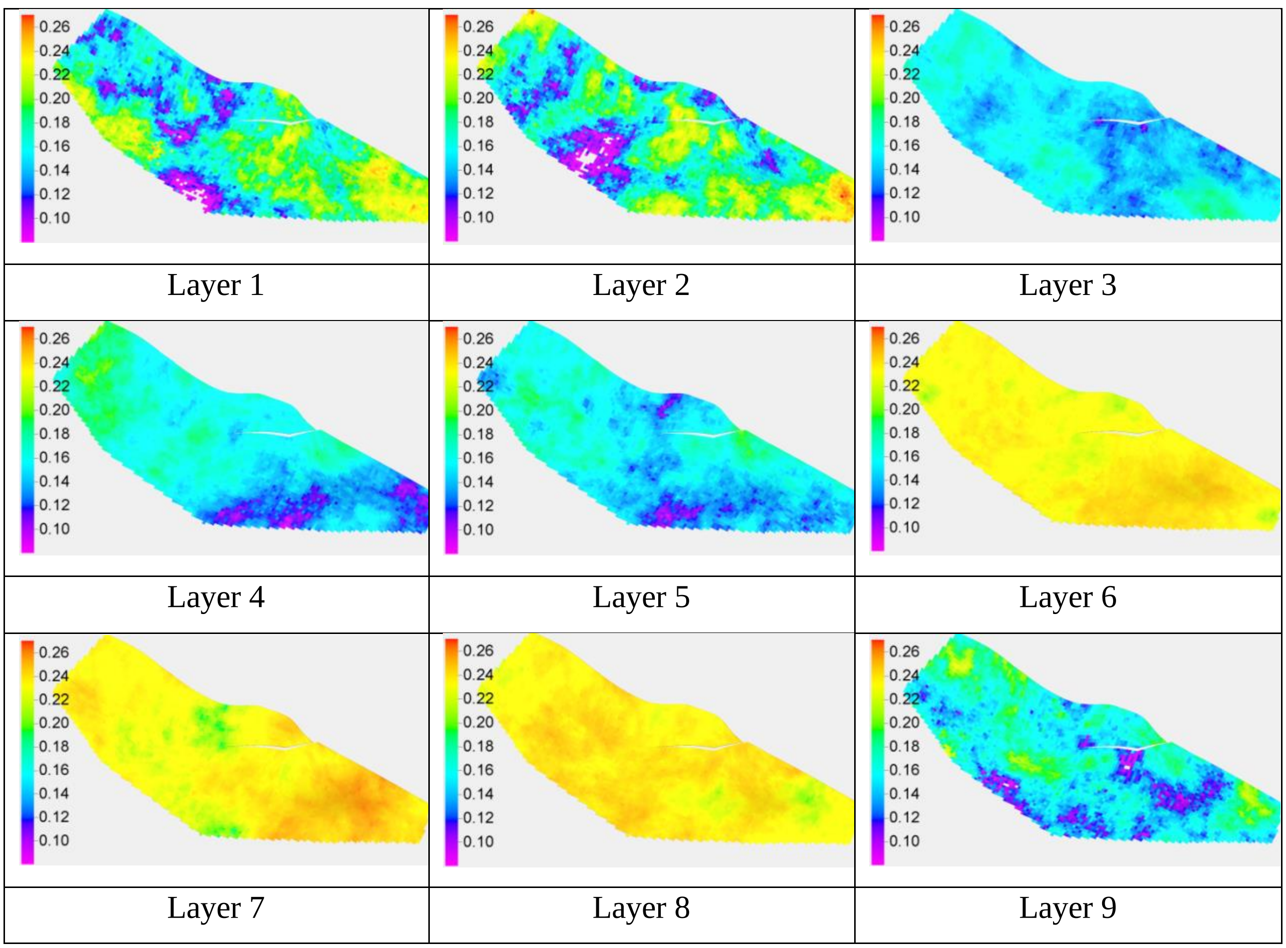


Figure 18: Porosity Distribution in Brugge model

***4.2.1 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:

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

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%).

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 17 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 19, 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 20 graphically presents the optimal solution obtained by GSA for 5 added injection wells under the experiment conditions

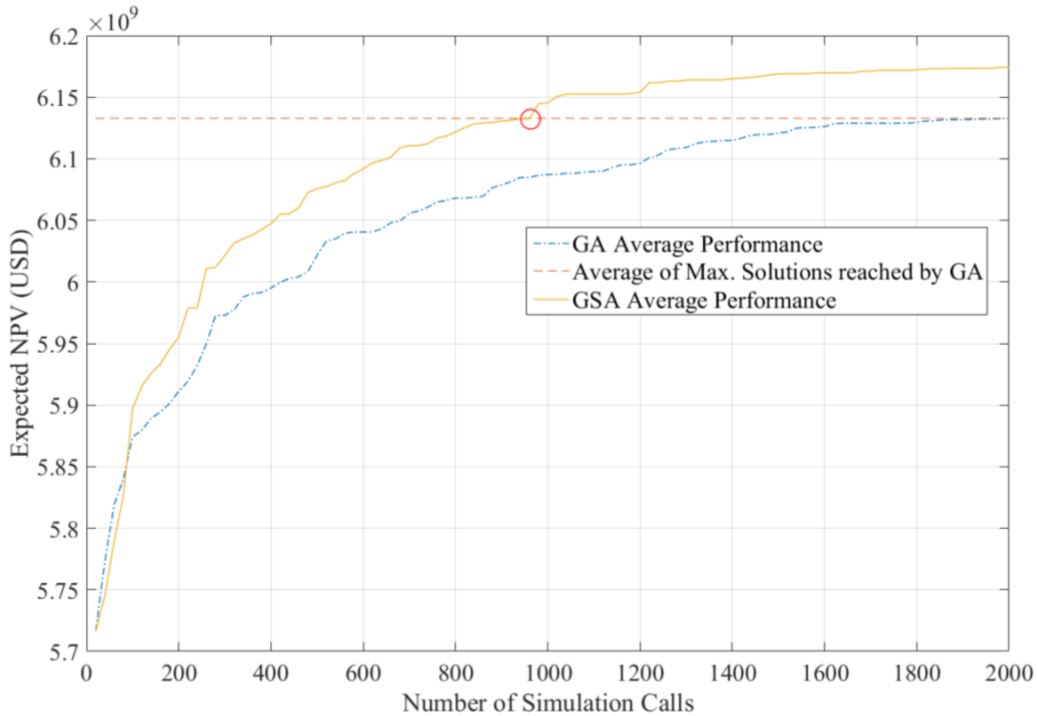


Figure 19: GSA and GA algorithms average performance over 10 optimization runs for optimal well placement of 5 injection wells in Brugge model

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 (m^3)	**Std.**	**Max. (m^3)**	**Min. (m^3)**
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

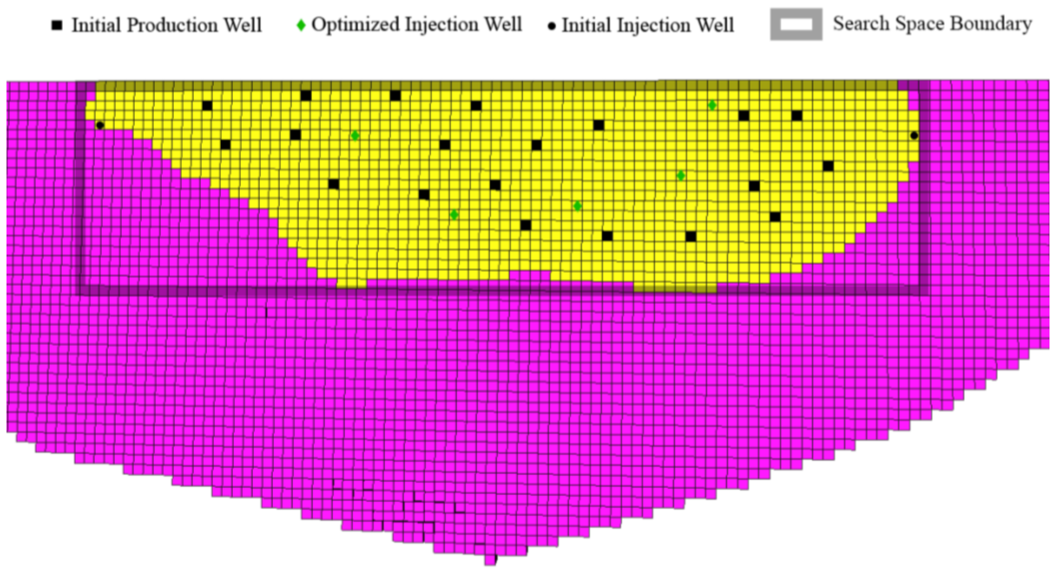


Figure 20: Optimal solution obtained by GSA for 5 added injection wells under the experiment conditions

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 are the convergence rate as well as solution robustness. A variety of example problems are investigated, involving placement of injection and production wells in two reservoir benchmark models. The first example is a simple case of optimal allocation for a single well, whereby the global optimal solution can be obtained through an exhaustive search algorithm. For that case, GSA has shown a better convergence rate compared to other approaches.

Afterwards, more difficult problems were addressed, whereby two cases for multi well placement were considered in the two different models. The GSA average performance has outperformed GA's average performance in convergence rate and solution robustness, furthermore, GSA has shown to be less prone to straying (premature convergence) which was indicated through the standard deviation values at the end of the optimization runs.

The results presented in this paper demonstrate that the addition of the proposed Similarity Operator can significantly improve convergence rate as well as solution robustness at a slight computational cost. This computational cost is initially required to compute the similarity between the features representing the cells.

Acknowledgments

The authors would like to acknowledge Schlumberger for donating the reservoir simulator used in this work. The authors also wish to acknowledge the financial support of the Munib & Angela Masri Institute of Energy and Natural Resources at the American University of Beirut (AUB) as well as the University Research Board of AUB. Furthermore, the authors would like to thank TNO for making the Brugge field data set available.

References

[1] 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.

[2] 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.

[3] A. C. Bittencourt and R. N. Horne, "Reservoir Development and Design Optimization," in *SPE Annual Technical Conference and Exhibition*, San Antonio, 1997.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[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] 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.

[12] 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.

[13] 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.

[14] 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.

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

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

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

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.