

A Comparison Analysis of Intelligence Algorithms for Oil Reservoir Production Optimization

Xiang Wang^{1,*}, Zhenlong Wang¹, Lei Zhang², Zhonghui Zhang² and Yanfeng He¹

¹School of Petroleum Engineering, Changzhou University, Changzhou 213164, China

²Research Institute of Petroleum Engineering, Shengli Oilfield Company, Sinopec, Dongying 257000, China

Received 26 July 2021; Accepted 17 September 2021

Abstract

Reservoir production optimization is a large-scale constrained nonlinear programming problem. The selection of solving algorithms still mainly relies on experience, which makes it difficult to fully adapt. Moreover, the accuracy and efficiency of optimization are facing bottlenecks. Different intelligent optimization algorithms have different optimization mechanisms. To explore the performance of each algorithm in solving the reservoir production optimization problem, a systematic algorithm comparison study was conducted. Four intelligent optimization algorithms, such as generalized pattern search (GPS), particle swarm optimization (PSO), covariance matrix adaptation evolutionary strategy (CMA-ES), and multilevel coordinate search (MCS), were selected, all of which cover global/local search and deterministic/random search strategy. A series of test cases, such as benchmark functions and well control optimization problems, were made. The four intelligent algorithms were applied to solve the problems, and the performance of each algorithm was compared and analysed. Results demonstrate that for the reservoir production optimization problem, MCS performs better than the three other algorithms in the early stage of optimization, and CMA-ES performs well in the middle and late stages. As the number of variables for reservoir production optimization problem increases, PSO and MCS with global search mechanisms have a higher risk of falling into a local optimal solution than CMA-ES and GPS, which use local search mechanisms. When the computation budget is limited, global search algorithms can converge to the vicinity of optimal solution faster than local search methods. Nevertheless, local search algorithms are more accurate than the global methods with sufficient computation budget. This study provides scientific support for the selection of algorithms for solving the reservoir production optimization problem.

Keywords: Oil reservoir, Production optimization, Intelligence algorithms, Particle swarm optimization, Multilevel coordinate search

1. Introduction

As the global political and economic uncertainties increase and international oil prices continue to fluctuate, ensuring oil production capacity is becoming increasingly important for a country's security and economic development. As the characteristics of the developing reservoirs become increasingly complex, the difficulty of development continues to increase. Traditional technology has been unable to meet needs, and the tasks of improving efficiency and controlling costs are arduous, making oil companies face huge challenges [1]. At the same time, emerging information technologies, which are represented by big data and artificial intelligence, are triggering a new round of technological and industrial revolutions. Accelerating the digital transformation and building intelligent oilfields have become the strategic key to breaking through the current development dilemma and achieving high-quality development for oil industries [2].

Designing or adjusting a reservoir development plan is a complex task. How to achieve automation and intelligent optimization decision making is one of the key issues in the construction of intelligent oilfields. However, many theoretical and technical difficulties still need to be resolved. Reservoir production optimization belongs to large-scale constrained nonlinear programming problems [3,4]. The

decision-making process must consider the influence of well production history; reservoir heterogeneity; many other geological and development factors; and many constraints, such as operating strategies and technological levels. These kinds of problems cannot be solved theoretically and must use intelligent optimization algorithms to solve them through iteration. Many types of optimization algorithms have been proposed with different optimization mechanisms. Different algorithms have different performances when solving different optimization problems. To the best of our knowledge, the performance and compatibility of different optimization algorithms with different optimization mechanisms for oil reservoir production optimization still lack in-depth comparative study. Algorithm selection still mainly relies on experience, which makes it difficult to fully adapt to the problem, the optimization efficiency is not high, and falling into the local optimum is easy. Moreover, the evaluations of the effects of the candidate production plans are often performed through time-consuming numerical simulations. Hence, the oil reservoir production optimization still faces the double bottleneck of accuracy and efficiency [5,6].

Based on the above analysis, selecting representative intelligent optimization algorithms with different optimization mechanisms and comparing and analyzing the performance of each algorithm for the reservoir production optimization problem are necessary. This study can provide guidance for the selection of algorithms for such problems

*E-mail address: xiangwangdr@gmail.com

ISSN: 1791-2377 © 2021 School of Science, IHU. All rights reserved.

doi:10.25103/jestr.144.21

and improve the decision-making accuracy and efficiency of the reservoir production optimization problem. This study can also enrich and improve the theories and methods of intelligent reservoir production optimization, improve the intelligent level of reservoir production management, and provide support for the construction of intelligent oilfields.

2. State of the art

The use of optimization methods to study oilfield development decision-making problems can be traced back to 1958 [7]. Lee et al. described the oil reservoir production problem as a linear programming problem and solved it then with the continuous development and progress of theories and technologies such as reservoir numerical simulation, optimization, artificial intelligence, and computer science. Intelligent production optimization theories and methods have attracted an increasing number of scholars all over the world and have achieved a series of valuable results.

Jansen et al. [8] applied the steepest descent method to optimize the production schedule of the oil and water wells in the reservoir. This method must calculate the gradient through the adjoint matrix. Due to the complexity of the reservoir production optimization problem, the solution hyperplanes are often rough and discontinuous, making the theoretical calculation of the gradient difficult in actual reservoir problems. Zhang et al. [9] introduced the stochastic disturbance gradient approximation algorithm (SPSA) into the reservoir production optimization problem. SPSA guides the optimization by estimating the gradient from the random disturbance of the control variables. This algorithm can avoid the direct calculation of the gradient. Zhou et al. [10] further simplified and improved the SPSA algorithm to make it more robust in solving the reservoir production optimization problem. Chen et al. [11] proposed a closed-loop optimization theory for reservoir production on the basis of the EnOpt algorithm. The above algorithms are all gradient-based methods, and the first-order or even multi-order gradient information of the function are needed. In general, these kinds of algorithms have high optimization efficiency, but they easily fall into the local optimal solution.

To avoid the problems caused by gradient calculation in the process of reservoir production optimization, a large number of derivative-free algorithms have also been introduced into this field. Foroud et al. [12] regarded the reservoir as a black box and used the generalized pattern search (GPS) algorithm to optimize reservoir production. GPS only uses the sampling information of the objective function value during the optimization process. Rostamian et al. [13] applied the genetic algorithm (GA) to study the multi-objective optimization design of directional well placements in reservoirs. Erfan et al. [14] used the GA algorithm to optimize the water and gas alternate injection process schedule for the oil reservoir. The GA algorithm simulates the biological genetic evolution process and approaches the optimal solution through repeated iterations, but GA can only be employed for single-objective problems. Allahyarzadeh et al. [15] took the multi-objective optimization problem of deep-water reservoir development as the research object and proposed the use of an improved GA (NSGA-II) to solve the problem, which verified its effectiveness. Particle swarm optimization (PSO) is similar to the GA and belongs to the evolutionary algorithm, but the algorithm principle is simpler and clearer than GA. PSO is also widely used in solving the reservoir production

optimization problem. Abdorreza et al. [16] used PSO to optimize the development plan for CO₂-assisted oil recovery in the reservoir, and Kawata et al. [17] successfully applied PSO to optimize the carbon dioxide capture problem. In addition to the above algorithms, Forouzanfar et al. [18] and Islam et al. [19] respectively explored the feasibility of covariance matrix adaptive evolutionary strategy (CMA-ES) in optimizing well control and well location for reservoir development. Zhao et al. [20] constructed a physical simplified proxy model and applied the differential evolution (DE) algorithm, which greatly reduced the time consumption of using numerical simulation to evaluate the value of the objective function of reservoir production optimization. Wang et al. [5] introduced the multilevel coordinated search (MCS) with outstanding performance in the theoretical comparison study into the field of reservoir production optimization and verified its performance in well placement optimization, well control optimization, and joint placement and control optimization problems. The optimization mechanisms of these derivative-free algorithms are often inspired by biological intelligence or natural physical laws and are called intelligent optimization algorithms [21]. This type of algorithm has lower optimization efficiency than gradient algorithms but generally does not require the continuity and convexity of the objective function and constraints. This type also has strong adaptability to nonlinear problems; hence, it is more suitable for complex reservoir production optimization problems.

In general, various types of intelligent optimization algorithms have been applied to the reservoir production optimization problem, but the performance of intelligent optimization algorithms with different optimization mechanisms in solving the reservoir production optimization problem still lacks in-depth comparative analysis. In this study, according to whether the algorithm belongs to global/local search and whether it belongs to deterministic/random search, four representative intelligent optimization algorithms are selected: GPS, PSO, CMA-ES, and MCS. A series of optimization problem examples are used to compare and analyze the performance of various intelligent optimization algorithms, providing scientific support for the selection of algorithms to solve the reservoir production optimization problem.

The remainder of this study is organized as follows: Section 3 establishes the mathematical model of the reservoir production optimization problem, categorizes and filters intelligent optimization algorithms, and proposes the basic principles of the selected algorithms. Section 4 compares and analyzes the optimization performance of each intelligent optimization algorithm through a series of optimization problem examples. Section 5 summarizes the conclusions.

3. Methodology

3.1 Mathematical modeling

The production schedule of oil and water wells affects the development yield of water flooding reservoirs significantly. The goal of oil reservoir production optimization is to find the optimal oil and water well control strategy under certain conditions, such as the number of oil and water wells, well placements, and well types, to improve the reservoir development and increase oil recovery. In reservoir development, well placement cannot be adjusted after drilling, but the injection/production rate of each well can be

adjusted at any time. Therefore, injection–production adjustment is more flexible and economical than well placement adjustment.

The mathematical model of the optimization problem includes four parts: performance indicators, optimization variables, objective functions, and constraints. For the reservoir production optimization problem, the mathematical model is described as follows:

(1) Performance indicators

The performance indicators commonly used in reservoir production optimization include net present value and cumulative oil production. In this study, the saturation standard deviation is selected as the performance index of the production optimization problem. The standard deviation of saturation is an index to evaluate the equilibrium degree of water flooding. It reflects not only the economics of development but also the efficiency of development. This factor makes it have certain advantages over net present value and cumulative oil production as performance indicators [22].

(2) Optimization variables

In general, oil and water wells require dynamic control, that is, the entire production cycle of the oil reservoir is divided into several control steps, and each control step represents a period of production time, such as half a year and one month. At the beginning of each control step, the production/injection rate of each well is reset. Therefore, in the reservoir production optimization problem, the optimization variable is the injection/production rate of each well at each control step.

For a given reservoir, assuming that the reservoir contains m wells, the entire production cycle is divided into n control steps, and the optimization variables can be expressed as

$$\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m] \quad (1)$$

$$\mathbf{q}_i = [q_{i1}, q_{i2}, \dots, q_{in}]^T \quad i = 1, 2, \dots, m \quad (2)$$

where \mathbf{q}_1 to \mathbf{q}_m is the injection/production rate vector from well 1 to m , respectively; q_{i1} to q_{in} is the injection/production rate of the i th well from control step 1 to n , respectively.

For a reservoir containing m wells and n control steps, the number of optimization variables for the dynamic control optimization problem is mn .

(3) Objective function

The goal of the production optimization problem is to determine the injection/production schedule of each well, making the crude oil be produced economically and efficiently during the water flooding development of the reservoir. Reflected in the saturation standard deviation, that is, the minimum of the saturation standard deviation in each injector–producer direction of the target reservoir and the maximum of the water flooding equilibrium degree. The objective function of this problem can be expressed as

$$\min SD = \sqrt{\sum_{i=1}^{n_s} \left[\frac{(S_{wi}(\mathbf{Q}) - \bar{S}_{wi}(\mathbf{Q}))^2}{n_s} \right]} \quad (3)$$

where SD is the saturation standard deviation; n_s is the total number of injector-producer directions; S_{wi} is the water

saturation of the i th direction; \bar{S}_{wi} is the average water saturation of all directions.

(4) Constraints

The constraints that must be considered in the production optimization problem include:

a. The bound constraints of well production/injection rates

$$q_{ij}^{\min} \leq q_{ij} \leq q_{ij}^{\max} \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n \quad (4)$$

where q_{ij}^{\min} and q_{ij}^{\max} are the minimum and maximum rates of the i th well in the j th control step, respectively; q_{ij}^{\min} is usually set to 0 in the optimization.

b. The nonlinear constraints of well bottom hole pressures

$$p_{ij}^{\min} \leq p_{ij}(q_{ij}) \leq p_{ij}^{\max} \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n \quad (5)$$

where p_{ij} is the bottom hole pressure of the i th well in the j th control step; p_{ij}^{\min} and p_{ij}^{\max} are the minimum and maximum bottom hole pressures of the i th well in the j th control step, respectively.

The number of optimization variables in the reservoir production optimization problem is large, and the constraints are complex, leading to the difficulty and complexity of solving the problem.

3.2 Intelligent optimization algorithms

More than 100 types of optimization algorithms have been proposed and studied, and their optimization mechanisms and applicable conditions are different.

According to whether each iteration of the algorithm optimization process is determined, the optimization algorithm can be divided into two categories: deterministic and random algorithms.

In a deterministic algorithm, the calculation rule for each iteration step is determined. Given the same input, the output result obtained by running the algorithm every time is exactly the same. Typical deterministic algorithms include Newton’s method, steepest descent method, simplex method, and GPS. Random algorithm involves random decision making in a certain step or a certain step in the operation of the algorithm, that is, some of the search decisions rely on random events. Therefore, given the same input, the random algorithm obtains different results each time. Typical random algorithms include GA and DE.

According to the searching range of the algorithm, it can be divided into local and global algorithms.

The optimization mechanism of local algorithms often relies on searching the solution space by neighborhood. Once a local algorithm falls into a local optimum, the algorithm ends at that point. Direct search algorithms, such as GPS and simplex methods, are local algorithms. Global algorithms focus on their ability to find the approximate locations of global optimal solutions in algorithm designs. This type introduces an escape mechanism to jump out of the local optimal solution and search the entire solution space. Typical global search algorithms include GA and PSO.

On the basis of the above classification, four algorithms are selected for further discussion: GPS, PSO, CMA-ES, and MCS. The classification of the four algorithms is shown in

Fig. 1. The four algorithms have different optimization mechanisms and are representative.

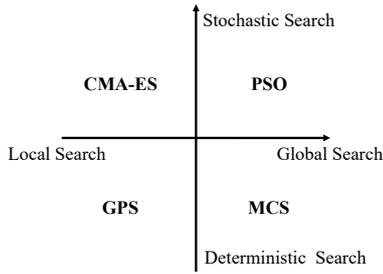


Fig. 1. Classification of the four optimization algorithms

(1) GPS

GPS is a direct search algorithm. It estimates the objective function from a set of directions and finds the descending direction by comparing the value of the objective function to solve the problem. The algorithm does not need to calculate or approximate any derivative, and it has a wide range of applications in the field of nonlinear programming and non-smooth optimization [23].

The GPS algorithm determines the set of next-generation search points according to the current search point and search mode. The specific calculation formula is

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta^k \mathbf{D} \quad (6)$$

where \mathbf{x}^{k+1} is the search point set of generation $k+1$; \mathbf{x}^k is the optimal point of generation k ; Δ^k is the search step size of generation k ; and \mathbf{D} is the search pattern. The search pattern generally uses a series of vector representations of orthogonal basis vectors $\{\mathbf{v}_i \in \mathbb{R}^n : i=1, \dots, r\}$. For an n -dimensional vector, the search basis vector contains at most $2n$ vectors and at least $n+1$ vectors. Both orthogonal basis vectors are also the two most commonly used search patterns in GPS, which are called the maximum basis and the minimum basis, respectively. Taking a two-dimensional problem as an example, the diagrams of the two search patterns are displayed in Fig. 2.

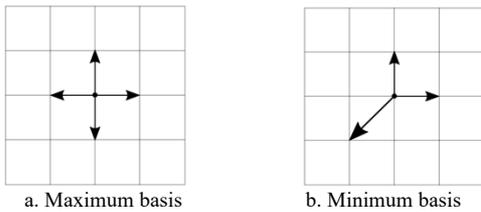


Fig. 2. Two patterns for GPS

The search step size is updated for each generation according to the following formula

$$\Delta^{k+1} = \begin{cases} \tau_\omega \Delta^k, & \tau_\omega \in [1, \infty) \\ \tau_r \Delta^k, & \tau_r \in (0, 1] \end{cases} \quad (7)$$

where Δ^{k+1} is the search step of the $k+1$ generation; τ_ω is the expansion factor; τ_r is the reduction factor.

(2) PSO

PSO, similar to GA, SA, and other algorithms, belongs to the evolutionary algorithm. PSO starts from a series of randomly generated or artificially given initial solutions and

finds the optimal solution through iterative calculations. Compared with GA, the rules for PSO to generate new individuals are simpler. PSO has high accuracy and fast convergence speed. At the same time, it can be implemented easily for parallel computing [24].

For an optimization problem containing D optimization variables, the particle swarm algorithm first generates a series of points in the D -dimensional search space called particles. The i th particle is denoted as $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$. Each particle has a velocity, and the velocity of the i th particle is expressed as $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. The particle records the information of the best position that it flies through, which is noted as $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. The best position that all particles have experienced in flight is recorded as $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. In each iteration, the velocity and position of the i th particle in the d th dimension ($1 \leq d \leq D$) are calculated according to the following equation

$$v_{id} = wv_{id} + c_1r_1(p_{id} - x_{id}) + c_2r_2(p_{gd} - x_{id}) \quad (8)$$

$$x_{id} = x_{id} + v_{id} \quad (9)$$

where w , c_1 , and c_2 are weighting parameters; r_1 and r_2 are stochastic vectors which are generated from the uniform distribution on $(0, 1)$ during each iteration.

(3) CMA-ES

CMA-ES is a random search algorithm based on population. Unlike GA and PSO, the distribution of individuals in the population of CMA-ES obeys a specific probability distribution, and the iterative optimization process mainly focuses on the adjustments of the probability distribution [25].

In iterative step k , CMA-ES first samples λ individuals to form a population according to the following formula

$$\mathbf{x}_i^k = N(\mathbf{m}^k, (\sigma^k)^2 \mathbf{C}^k), \quad i=1, \dots, \lambda \quad (10)$$

where $N(\dots, \dots)$ is a random vector of the multivariate normal distribution; \mathbf{m}^k is the mean vector; \mathbf{C}^k is the covariance matrix; σ is the step factor.

Mean vector \mathbf{m}^k represents the current optimal solution; covariance matrix \mathbf{C}^k is a symmetric positive definite matrix used to describe the geometric characteristics of the distribution; step size factor σ is used to enlarge or reduce covariance matrix \mathbf{C}^k globally to achieve rapid convergence and avoid premature phenomena. In the iterative process of CMA-ES, the above three parameters must be continuously updated.

Fig. 3 illustrates an example of CMA-ES solving an optimization problem. The objective function is $f(\mathbf{x}) = x_1^2 + x_2^2$, and the upper and lower bound constraints are $x_1, x_2 \in [-800, 800]$. The dotted line in the figure is the contour of the objective function. The initial point is $(200, 200)$, and the optimal solution is $(0, 0)$, which is marked with "x" in the figure. The red and black ellipses represent the contours of the probability distribution density of this generation and the previous generation, respectively.

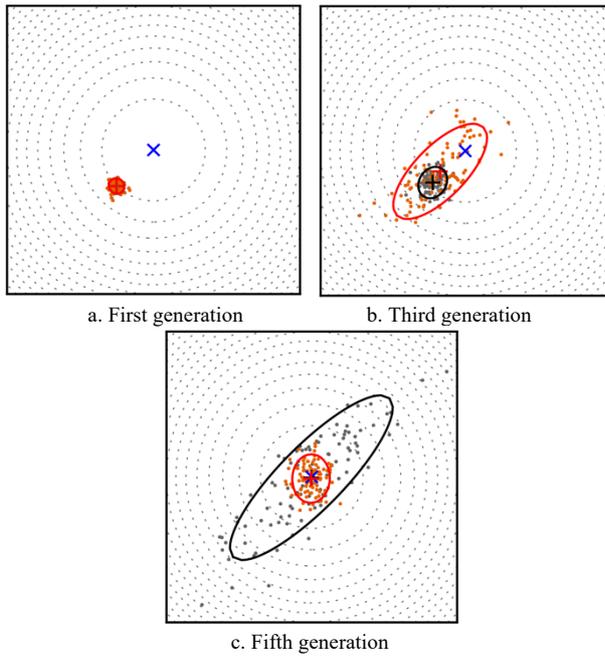


Fig. 3. An optimization case using CMA-ES

(4) MCS

MCS was proposed by Huyer and Neumaier, and its basic idea was inspired by the DIRECT algorithm, which also belongs to the derivative-free algorithm [26]. MCS combines global and local searches. The optimization process is mainly to subdivide the entire search space into small search spaces. For subspaces that have not been fully explored, global algorithms are used for optimization; for subspaces that are sufficiently small, local algorithms are used for further precise optimization.

To further describe the MCS optimization process, an upper and lower bound optimization problem is defined as follows

$$\min f(\mathbf{x}), \quad \mathbf{x} \in [\mathbf{u}, \mathbf{v}] \quad (11)$$

where \mathbf{u} and \mathbf{v} are n -dimensional vectors.

MCS divides the initial search space into a series of subspaces B through the preset initialization list. Each subspace contains three characteristic parameters: base point, pair point, and level s . All the subspaces are sorted according to level and included in the sweep list, and the subspace with the smallest objective function value in each level is marked. The marked subspace enters the next splitting step for further splitting. Splitting includes two strategies: sorting splitting and expecting splitting. Sorting splitting is used for subspaces that have been split enough times and further splitting along the direction of the least number of splits of the subspace. Expecting splitting is used for subspaces that have not been fully split. First, we estimate the expectation that the subspace will be split in various directions to obtain a better solution and then split along the direction of the maximum expectation. The new subspace generated by the split is added to the sweep list, which is updated. After the scan is completed, the base points of all subspaces that reach the maximum number of levels s_{\max} are put into the shopping basket according to the value of the objective function. The points in the shopping basket are judged whether they are in a new local minimum area, and the points with better objective function values in

each local minimum area are selected to start the local search. Local search combines the ideas of coordinated search and sequential quadratic programming. First, a partial quadratic form is constructed through a series of coordinated searches. Second, the search direction and search step length are determined according to the sequential quadratic programming. After the search is completed, the quadratic form is updated.

4. Result Analysis and Discussion

Different intelligent optimization algorithms have different solving mechanisms and are applicable to different optimization problems. The comparison of the pros and cons of various optimization algorithms is mainly investigated and evaluated from three aspects: reliability, effectiveness, and simplicity. The focus here is on the efficiency of the optimization algorithm. The main computational effort when solving the reservoir production optimization problem is the calculation of the objective function value. The calculation budget required by commonly used numerical simulation methods to predict the value of the objective function under different production plans is much greater than that of the optimization algorithm itself. Therefore, when examining the calculation efficiency here, the attention is paid to the relationship between the number of calculations of the objective function value and the optimal value found.

4.1 Benchmark test

4.1.1 Problem description

Benchmark test is widely conducted to test algorithm performance and comparatively analyze optimization algorithms. It often uses a series of standardized optimization problems with different characteristics and scales. The benchmark test set used here comes from CEC 2005 and contains a total of 25 test functions, including five unimodal functions and 20 multimodal functions. Among the multimodal functions, seven basic, two extended, and 11 mixed functions are included. Some functions add noise terms to further increase the difficulty of solving the problem. Due to space limitations, only a few representative functions are presented for the introduction. For a more detailed introduction of each function, refer to the report of P. N. Suganthan et al. [27].

(1) Shifted sphere function

$$f(\mathbf{x}) = \sum_{i=1}^D z_i^2 + C, \quad \mathbf{z} = \mathbf{x} - \mathbf{o}, \quad \mathbf{x} = [x_1, x_2, \dots, x_D] \quad (12)$$

where D is the number of dimensions; $\mathbf{o} = [o_1, o_2, \dots, o_D]$ is the translational global optimal solution.

This function is unimodal, which is obtained by translating the sphere function, and has the characteristics of decomposability and scalability. The upper and lower bound constraints are $\mathbf{x} \in [-100, 100]^D$, the global optimal solution is $\mathbf{x}^* = \mathbf{o}$, and the optimal value is $f(\mathbf{x}^*) = C = -450$. When $D = 2$, the 3D image of this function is displayed in Fig. 4.

(2) Shifted Schwefel function with noise

$$f(\mathbf{x}) = \left[\sum_{i=1}^D \left(\sum_{j=1}^i z_j \right)^2 \right] [1 + 0.4|N(0,1)|] + C, \quad \mathbf{z} = \mathbf{x} - \mathbf{o}, \quad \mathbf{x} = [x_1, x_2, \dots, x_D] \quad (13)$$

where D is the number of dimensions; $\mathbf{o}=[o_1, o_2, \dots, o_D]$ is the translational global optimal solution.

This function is unimodal, which is obtained by shifting the Schwefel function. The optimized variables are expandable but not decomposable, and noise terms are added to the objective function value, resulting in a rough solution surface. The upper and lower bound constraints are $\mathbf{x} \in [-100, 100]^D$, the global optimal solution is $\mathbf{x}^* = \mathbf{o}$, and the optimal value is $f(\mathbf{x}^*) = C = -450$. When $D = 2$, the 3D image of this function is shown in Fig. 5.

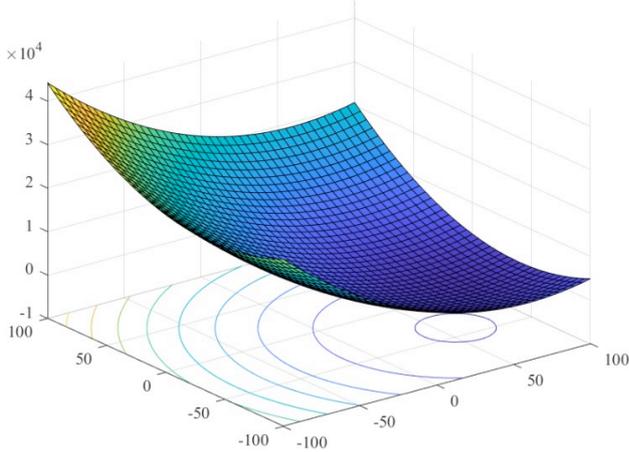


Fig. 4. 3D map for the 2D-shifted sphere function

(3) Shifted rotated Weierstrass function

$$f(\mathbf{x}) = \sum_{i=1}^D \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k (z_i + 0.5))] - D \sum_{k=0}^{k_{\max}} [a^k \cos(\pi b^k)] + C, \quad (14)$$

$$a = 0.5, b = 3, k_{\max} = 20, \mathbf{z} = (\mathbf{x} - \mathbf{o})\mathbf{M}, \mathbf{x} = [x_1, x_2, \dots, x_D]$$

where D is the number of dimensions; $\mathbf{o}=[o_1, o_2, \dots, o_D]$ is the global optimal solution for translation; \mathbf{M} is the linear transformation matrix.

This function is multimodal, which is obtained by shifting and rotating the Weierstrass function. The optimized variable is expandable but not decomposable. Although the function is continuous but can only be derived from a series of points, the function contains multiple local optimal solutions. The upper and lower bound constraints are $\mathbf{x} \in [-0.5, 0.5]^D$, the global optimal solution is $\mathbf{x}^* = \mathbf{o}$, and the optimal value is $f(\mathbf{x}^*) = C = 90$. When $D = 2$, the 3D image of this function is illustrated in Fig. 6.

(4) Non-continuous rotated hybrid composition function

This function is a mixture of rotating extended Scaffer, Rastrigin, F8F2, Weierstrass, and Griewank functions. The specific function expression is complicated. Refer to the report of P. N. Suganthan et al. [27], which is not listed here. This function is multimodal, and the optimization variable is expandable but not decomposable. It also contains a large number of local optimal solutions. The characteristics of various functions are mixed, and the function is discontinuous and non-differentiable. The optimal value is located on the boundary of the optimization space. The upper and lower bound constraints $\mathbf{x} \in [-5, 5]^D$, the global optimal solution $\mathbf{x}^* = \mathbf{o}$, and the optimal value $f(\mathbf{x}^*) = 360$.

When $D = 2$, the 3D image of this function is shown in Fig. 7.

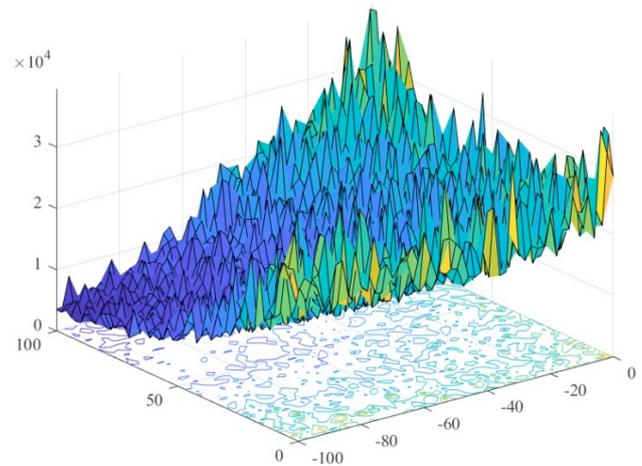


Fig. 5. 3D map for the 2D-shifted Schwefel function with noise

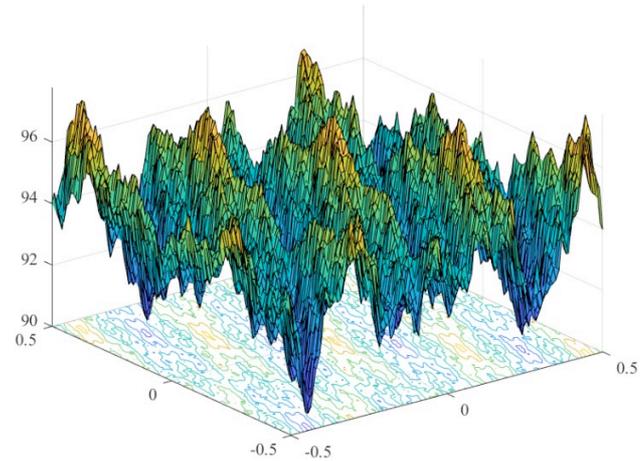


Fig. 6. 3D map for the 2D-shifted rotated Weierstrass function

When using the optimization algorithm introduced in this study to optimize the above benchmark functions, consider the four cases of dimension $D = 2, 10, 30$, and 50 and set the maximum objective function evaluation times to $500D$.

4.1.2 Algorithm performance comparison

GPS, PSO, CMA-ES, and MCS are used to solve the 25 benchmark optimization problems. Each problem is optimized when the number of optimization variables is equal to 2, 10, 30, and 50, resulting in a total of 100 optimization problems. By analyzing the evaluation number of the objective function of each algorithm in each test problem and its corresponding optimal objective function value, the performance of each algorithm in solving each problem can be obtained. Due to a large number of problems, comparisons are made according to the number of optimization variables. In this case, each algorithm still contains 25 problems, and the initial and optimal values of each problem are different. To further facilitate the comparison, the results of each algorithm are normalized, and the optimal objective function value in the optimization process of each algorithm is converted to a value between 0 and 1, where 0 represents the global optimal value of the problem, and 1 represents its initial value. For each algorithm, we use the median value of its performance in 25 problems under different optimization variables as its average performance. Then, we plot the average

performance comparison of each algorithm under different optimization variables. The results are presented in Figs. 8–11.

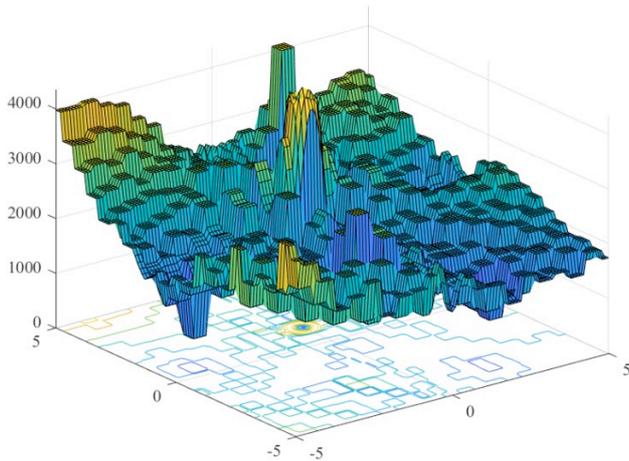


Fig. 7. 3D map for the 2D non-continuous rotated hybrid composition function

Figs. 8-11 show that the MCS algorithm obtains the closest solution to the global optima in the final stage of optimization when $D = 2$. CMA-ES outperforms the three other algorithms in the final stage of optimization when $D = 10, 30,$ and 50 . From the perspective of the optimization process, MCS performs better than the three other algorithms in the early stage of optimization, and CMA-ES performs better in the middle and late stages.

When $D = 2$, the four algorithms all converge to between 0 and 0.2. The final convergence values of the four algorithms all decrease, ranging from 0.1 to 0.4 for problems $D = 10$. When the number of optimization variables reaches 50, the four algorithms finally only converge to between 0.2 and 0.6. At the same time, as the number of optimization variables increases, the number of objective function evaluations performed by optimization also increases significantly, from 1,000 when $D = 2$ to 25,000 when $D = 50$. Therefore, as the number of optimization variables increases, the difficulty of the problem continues to increase; the difficulty and calculation budget for the optimization algorithm to find the optimal solution also continue to increase.

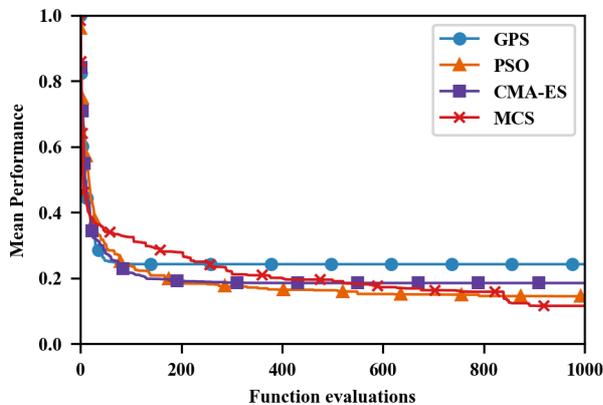


Fig. 8. Benchmark test performance comparison of the four algorithms ($D = 2$)

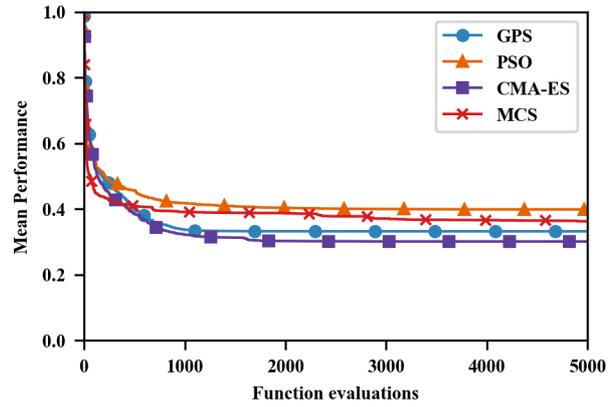


Fig. 9. Benchmark test performance comparison of the four algorithms ($D = 10$)

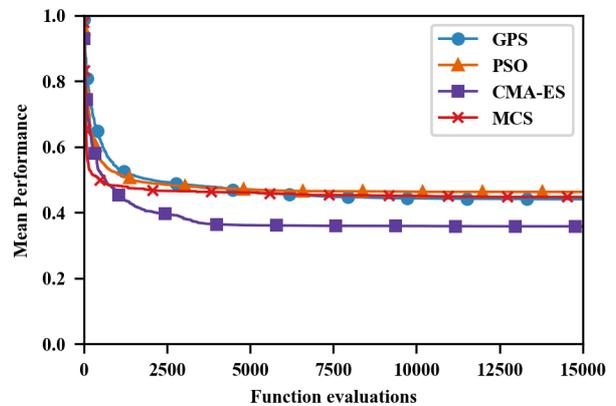


Fig. 10. Benchmark test performance comparison of the four algorithms ($D = 30$)

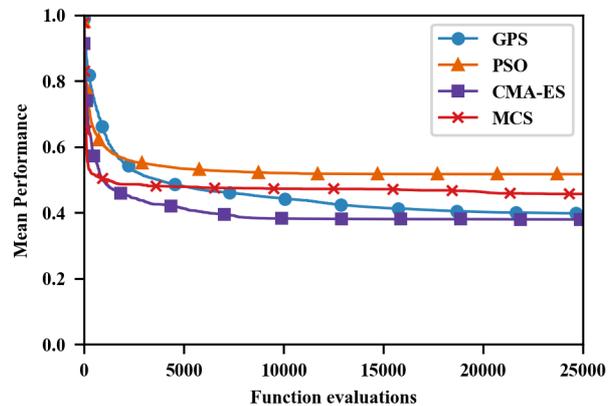


Fig. 11. Benchmark test performance comparison of the four algorithms ($D = 50$)

4.2 Well production optimization example

4.2.1 Problem description

The well control optimization example uses a single-layer reservoir model. The number of model grids is $51 \times 51 \times 1$, and the size of each grid is $\Delta x = \Delta y = 10\text{m}$ [28]. The oil reservoir contains oil and water. A five-point well pattern is placed, with a water injection well in the middle and production wells on four corners. The reservoir permeability is divided into four areas: two high-permeability and two low-permeability areas. The high-permeability areas have a

permeability of 1000 mD, whereas the low-permeability areas have a permeability of 100 mD. Fig. 12 shows the permeability distribution and well location map of the reservoir model.

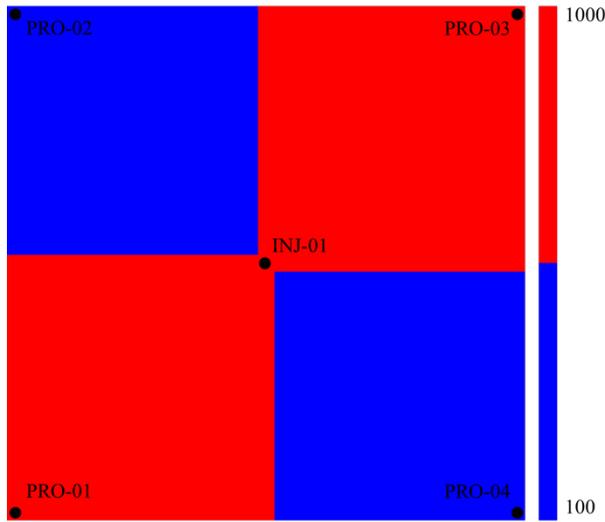


Fig. 12. Permeability distribution and well placement

The well production optimization for four production wells is performed with a constant injection rate. Well production static allocation optimization and dynamic control optimization are considered. For the static allocation optimization problem (Case 1), each well maintains the same rate to produce until the end of development, resulting in a total of four optimization variables. For the dynamic control optimization problem (Case 2), each well performs rate adjustment every 90 days and has eight control steps. The problem has a total of 32 optimization variables.

4.2.2 Algorithm performance comparison

Figs. 13 and 14 display the convergence of the four algorithms (i.e., GPS, PSO, CMA-ES, and MCS) in the well production optimization problem. The x-axis in both figures is the number of evaluations of the objective function, and the y-axis is the optimal standard deviation of the saturation. GPS, PSO, CMA-ES, and MCS all use the median value of the upper and lower limits of each variable as the initial solution. Therefore, in the injection and production optimization, the initial solutions and values of the four algorithms are exactly the same.

For the two algorithms of PSO and CMA-ES, given that they are random algorithms, the results obtained in each run are different. Both algorithms are run 10 times with the same initial solution to see the average performance. The median of the results of 10 runs is used instead of the average value to reflect the average effect and avoid the influence of extreme runs on the average effect.

Fig. 13 shows that for the static allocation optimization problem, the convergence speed of the four algorithms is not much different. Among them, MCS converges to the optimal value the fastest, and then GPS and CMA-ES converge to the optimal value. At the end of the operation of PSO, its optimal value is close to the global optimal value, but a slight gap remains. This result is consistent with the previous analysis of the characteristics of the algorithm. PSO is a global search algorithm, and its ability to converge to an accurate optimal value is worse than other algorithms.

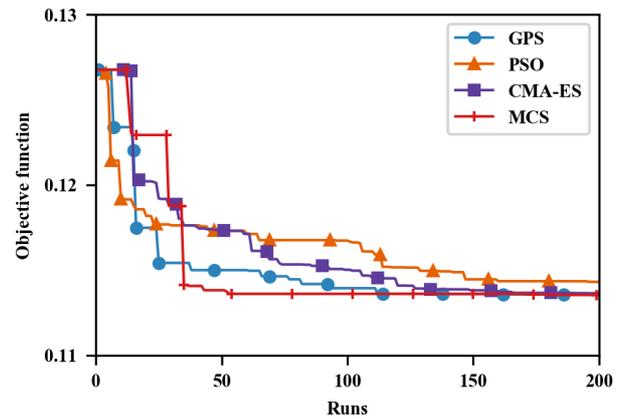


Fig. 13. Optimization performance for well production optimization using different algorithms

Fig. 14 illustrates that for the dynamic control optimization problem, due to a large number of optimization variables, the correspondingly set maximum objective function calculation times are also large, which is 3,000. In the middle and early stages of algorithm operation, MCS and CMA-ES have the fastest convergence speed, whereas GPS has the slowest convergence speed. Due to the increase in the number of variables, the optimization speed of local search algorithms, such as GPS, is becoming increasingly worse. However, after enough objective function evaluation times, GPS and CMA-ES converge to the lowest saturation standard deviation. If the calculation amount is sufficient, then local algorithms are still the most accurate ones. PSO is similar to the performance in the static deployment problem and still only converges to a value near the optimal value.

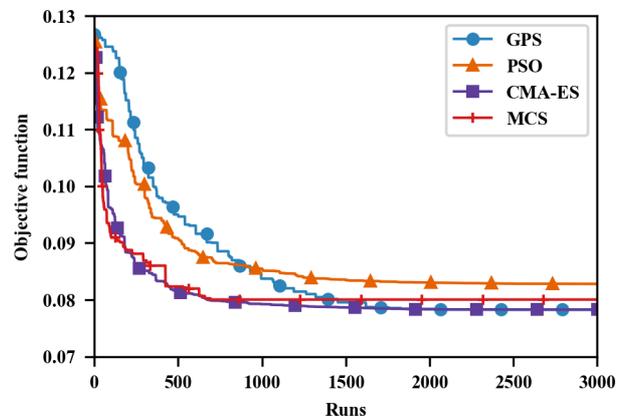


Fig. 14. Optimization performance for well control optimization using different algorithms

Comparing the performance of the four algorithms in the static allocation problem and the dynamic control problem, as the number of optimization variables increases, the number of calculations of the objective function value required by the four algorithms for optimization increases significantly. MCS performs well with different numbers of optimized variables, whereas the performance of GPS is greatly affected by the increase in the number of optimized variables. In addition, dynamic control optimization can converge to a lower objective function value than static allocation optimization. It indicates that for a reservoir, the optimization effect of dynamic development is better than that of static development.

5. Conclusion

To explore the performance of intelligent optimization algorithms with different optimization mechanisms in solving the reservoir production optimization problem, four representative intelligent optimization algorithms (i.e., GPS, PSO, CMA-ES, and MCS) are selected and evaluated through a series of optimization examples. The following conclusions can be drawn:

(1) As the number of optimization variables increases, the difficulty of the problem continues to increase. Moreover, the difficulty and number of evaluations for the four intelligent optimization algorithms to find the optimal solution continue to increase.

(2) MCS performs better than the three other algorithms in the early stage of optimization, and CMA-ES performs well in the middle and late stages.

(3) When PSO and MCS with global search mechanisms are used to solve the problem of large-scale reservoir production optimization, compared with CMA-ES and GPS using local search mechanisms, a high risk of falling into a local optimal solution is possible. When the calculation budget is limited, global algorithms can converge to the optimal solution fast. When the calculation budget is

sufficient, local algorithms are more accurate than global algorithms.

Through comparative analysis, this study clarifies the performance of representative algorithms under different optimization mechanisms, such as global/local search and deterministic/random search, and provides reference and guidance for the efficient solution of the reservoir production optimization problem. However, algorithms that can consistently outperform during the entire optimization process when solving such a problem are still lacking. Combining the advantages of different algorithms is necessary to build more efficient hybrid algorithms in the future.

Acknowledgements

This work was supported by Open Fund (PLC20190803) of State Key Laboratory of Oil and Gas Reservoir Geology and Exploitation (Chengdu University of Technology).

This is an Open Access article distributed under the terms of the Creative Commons Attribution License.



References

1. Jefferson, M., "A crude future? COVID-19s challenges for oil demand, supply and prices". *Energy Research & Social Science*, 68, 2020, pp.101669.
2. Jia, A., Guo, J., "Key technologies and understandings on the construction of smart fields". *Petroleum Exploration and Development*, 39(1), 2012, pp.118-122.
3. Isebor, O. J., Durlafsky, L. J., Ciaurri, D. E., "A derivative-free methodology with local and global search for the constrained joint optimization of well locations and controls". *Computational Geosciences*, 18(3-4), 2014, pp.463-482.
4. Hou, J., Zhou, K., Zhang, X-S., Kang, X-D., Xie, H., "A review of closed-loop reservoir management". *Petroleum Science*, 12(1), 2015, pp.114-128.
5. Wang, X., Haynes, R. D., Feng, Q., "A multilevel coordinate search algorithm for well placement, control and joint optimization". *Computers & Chemical Engineering*, 95, 2016, pp.75-96.
6. Fonseca, R. M., Della Rossa, E., Emerick, A. A., Hanea, R. G., Jansen, J. D., "Overview Of The Olympus Field Development Optimization Challenge". In: *ECMOR XVI-16th European Conference on the Mathematics of Oil Recovery*, Barcelona, Spain: European Association of Geoscientists & Engineers, 2018, pp.1-10.
7. Lee, A. S., Aronofsky, J. S., "A linear programming model for scheduling crude oil production". *Journal of Petroleum Technology*, 10(07), 1958, pp.51-54.
8. Jansen, J. D., Durlafsky, L. J., "Use of reduced-order models in well control optimization". *Optimization and Engineering*, 18(1), 2017, pp.105-132.
9. Zhang, K., Li, Y., Yao, J., Liu, J., Yan, X., "Theoretical research on production optimization of oil reservoirs". *Acta Petrolei Sinica*, 31(01), 2010, pp.86-91.
10. Zhao H., Cao L., Li Y., Yao, J., "Production optimization of oil reservoirs based on an improved simultaneous perturbation stochastic approximation algorithm". *Acta Petrolei Sinica*, 32(6), 2011, pp.1031-1036.
11. Chen, Y., Oliver, D. S., Zhang, D., "Efficient ensemble-based closed-loop production optimization". *SPE Journal*, 14(4), 2009, pp.634-645.
12. Foroud, T., Baradaran, A., Seifi, A., "A comparative evaluation of global search algorithms in black box optimization of oil production: A case study on Brugge field". *Journal of Petroleum Science and Engineering*, 167, 2018, pp.131-151.
13. Rostamian, A., Jamshidi, S., Zirbes, E., "The development of a novel multi-objective optimization framework for non-vertical well placement based on a modified non-dominated sorting genetic algorithm-II". *Computational Geosciences*, 23(5), 2019, pp.1065-1085.
14. Erfan, M., James, L. A., Haynes, R. D., "Optimization of hydrocarbon water alternating gas in the Norne field: Application of evolutionary algorithms". *Fuel*, 223, 2018, pp.86-98.
15. Allahyarzadeh, A., Dezan, D. J., Salviano, L. O., de Oliveira Junior, S., Yanagihara, J. I., "FPSO fuel consumption and hydrocarbon liquids recovery optimization over the lifetime of a deep-water oil field". *Energy*, 181, 2019, pp.927-942.
16. Karkevandi-Talkhooncheh, A., Hajirezaei, S., Hemmati-Sarapardeh, A., Husein, M. M., Karan, K., Sharifi, M., "Application of adaptive neuro fuzzy interface system optimized with evolutionary algorithms for modeling CO₂-crude oil minimum miscibility pressure". *Fuel*, 205, 2017, pp.34-45.
17. Kawata, Y., Azuma, H., Takagi, M., "Some approaches for optimizing co₂ storage system with water production in a heterogeneous reservoir using particle swarm optimization algorithm". *Energy Procedia*, 114, 2017, pp.4775-4786.
18. Forouzanfar, F., Poquioma, W. E., Reynolds, A. C., "A covariance matrix adaptation algorithm for simultaneous estimation of optimal placement and control of production and water injection wells". In: *SPE reservoir simulation symposium*, Houston, USA: SPE, 2015.
19. Islam, J., Vasant, P. M., Negash, B. M., Laruccia, M. B., Myint, M., Watada, J., "A holistic review on artificial intelligence techniques for well placement optimization problem". *Advances in Engineering Software*, 141, 2020, pp.102767.
20. Zhao, H., Xu, L., Guo, Z., Liu, W., Zhang, Q., Ning, X., Shi, L. "A new and fast waterflooding optimization workflow based on INSIM-derived injection efficiency with a field application". *Journal of Petroleum Science and Engineering*, 179, 2019, pp.1186-1200.
21. You, J., Ampomah, W., Sun, Q., "Development and application of a machine learning based multi-objective optimization workflow for CO₂-EOR projects". *Fuel*, 264, 2020, pp.116758.1-116758.13.
22. Feng, Q., Wang, X., Wang, D., Huang, Y., "Theoretical analysis on the performance of equilibrium displacement in water flooding reservoir". *Petroleum Geology and Recovery Efficiency*, 23(3), 2016, pp.83-88.
23. Abramson, M. A., Audet, C., Dennis, J. E., "Generalized pattern searches with derivative information". *Mathematical Programming*, 100(1), 2003, pp.3-25.

24. Murugesan, R. K., "Stochastic prediction with particle swarm optimization". Doctoral dissertation of Dalhousie University, Canada, 2015.
25. Bouzarkouna, Z., Ding, D. Y., Auger, A., "Well placement optimization with the covariance matrix adaptation evolution strategy and meta-models". *Computational Geosciences*, 16(1), 2012, pp.75-92.
26. Costa, M. F. P., Rocha, A. M. A., Fernandes, E. M., "Hooke and jeeves based multilevel coordinate search to globally solving nonsmooth problems". In: *11th International Conference of Numerical Analysis and Applied Mathematics*, Rhodes, Greece: American Institute of Physics, 1558(1), 2013, pp.610-613.
27. Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y. P., Auger, A., Tiwari, S., "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization". *KanGAL report*, 2005005(2005), 2005.
28. Oliveira, D. F., Reynolds, A., "An adaptive hierarchical multiscale algorithm for estimation of optimal well controls". *SPE Journal*, 19(5), 2014, pp.909-930.