Artif Intell Rev
https://doi.org/10.1007/s10462-018-9612-8

Application of artificial intelligence techniques in the petroleum industry: a review

Hamid Rahmanifard[1] · Tatyana Plaksina[1]

Abstract In recent years, artificial intelligence (AI) has been widely applied to optimization problems in the petroleum exploration and production industry. This survey offers a detailed literature review based on different types of AI algorithms, their application areas in the petroleum industry, publication year, and geographical regions of their development. For this purpose, we classify AI methods into four main categories including evolutionary algorithms, swarm intelligence, fuzzy logic, and artificial neural networks. Additionally, we examine these types of algorithms with respect to their applications in petroleum engineering. The review highlights the exceptional performance of AI methods in optimization of various objective functions essential for industrial decision making including minimum miscibility pressure, oil production rate, and volume of CO_2 sequestration. Furthermore, hybridization and/or combination of various AI techniques can be successfully applied to solve important optimization problems and obtain better solutions. The detailed descriptions provided in this review serve as a comprehensive reference of AI optimization techniques for further studies and research in this area.

Keywords Artificial intelligence · Genetic algorithm · Particle swarm optimization · ANN · Fuzzy logic · Differential evolution · Petroleum engineering

Contents

✉ Hamid Rahmanifard
hamid.rahmanifard@ucalgary.ca

1 Department of Chemical and Petroleum Engineering, Schulich of Engineering, University of Calgary, 2500 University Dr. NW, Calgary, AB T2N 1N4, Canada

Published online: 16 January 2018

Abbreviations

ACO	Ant colony optimization
AI	Artificial intelligence
ANFIS	Adaptive neuro-fuzzy inference system
ANN	Artificial neural networks
ARS	Adaptive random search
BP-ANN	Back propagation artificial neural networks
CCDE	Cooperative coevolutionary differential evolution
CMA-ES	Covariance matrix adaptation evolution strategy
CR	Crossover probability rate
CSOR	Cumulative steam to oil ratio
DE	Differential evolution
E¶	Exploration and production
EA	Evolutionary algorithms
F	The scaling factor
FL	Fuzzy logic
FIS	Fuzzy inference system
GA	Genetic algorithms
gbest	The other's best experiences
HDE	Hybrid differential evolution
HF	Hydraulic fracturing
ICA	Imperialist competitive algorithm
MMP	Minimum miscibility pressure
NA	Neighborhood algorithm
NAB	Neighborhood approximation Bayes
NP	The size of population
NPV	Net present value
pbest	A particle's best experience
PSO	Particle swarm optimization
SAGD	Steam assisted gravity drainage
SI	Swarm intelligence
SPSA	Simultaneous perturbation stochastic approximation
UD	Uniform design
VAPEX	Vapor extraction
WAG	Water alternative gas

1 Introduction

Optimization methods were first introduced in the petroleum exploration and production (E&P) industry in the 1940s, and since then have been utilized widely for predicting, estimating, and determining various operational parameters (e.g., asphaltene precipitation, minimum miscibility pressure (MMP), wettability, well placement, history matching, drilling operations, pipeline conditions, and etc.) (Wang 2003). These methods are mainly categorized into three groups including linear, integer, and nonlinear programming techniques.

Linear programing technique is primarily used for the cases in which both the objective function and the constraints are linear (Carroll Jr and Horne 1992). The simplex algorithm and the interior point algorithm are two examples of the linear technique. Though very popular, this approach has one considerable drawback—it requires a large number of iterations to converge (Klee and Minty 1970). Integer programming technique, on the other hand, is applicable to the problems for which all unknown components are discrete or mixed continuous and integer (e.g., coupled well control and placement optimization). To tackle these problems, the scholars usually use two approaches: the cutting plane technique and the branch and bound method (Gomory 1958; Land and Doig 2010). The main disadvantage of this approach is its high computational cost and time. The third approach, nonlinear programming technique, is utilized for the optimization problems for which either the objectives or constraints are nonlinear. This method can be divided into two main categories: gradient-based algorithms and gradient-free optimization algorithms (Mohagheghian 2016). Gradient-based optimization algorithms search for the steepest descent (or ascent, depending on the type of optimum required) direction and the function extremes using analytical or numerical objective functions including numerical finite difference methods (Taylor series expansions), the steepest descent method, Newton's method, quasi-Newton method, and sequential quadratic programming technique (Watson et al. 1980; Fujii and Horne 1995; Chen 2013). As their names suggest, gradient-based optimization algorithms require computation of the objective function derivative and its constraints. However, not all objective functions are differentiable due to the following reasons:

- The objective function or the regions defined by the constraints are non-differentiable;
- A simulation-based objective function for which the derivative computation requires access to the simulation code (especially for commercial software);
- The objective function is a result of a physical experiment for which due to the lack of a precise actuator, the derivative of the function is impossible.

Therefore, the lack of a computable derivative causes the failure of gradient-based optimizers and requires application of derivative-free techniques. Heuristic or gradient-free optimization methods tend to be fast, they provide nearly optimal solutions, and solve the problems more efficiently using knowledge of the domain. However, application of these methods cannot guarantee finding the actual value of the optimal solution (Kamrani 2010; Mohagheghian 2016). A generic classification of gradient-free approaches includes trajectory-based and population-based methods. A trajectory-based method considers only one solution, while a population-based heuristic method usually maintains a population of solutions (Kamrani 2010). Figure 1 summarizes the introduction section and shows a classification of the optimization techniques with several examples of each method.

Among population-based methods, Artificial Intelligence (AI) algorithms have been widely used for solving problems in the oil and gas industry. In general, AI is defined as the ability of intelligent agents for continuous learning in the corresponding environment and perceiving certain activities (Jang et al. 1997). AI is mainly comprised of Evolutionary Algorithms (EA), Swarm Intelligence (SI), Fuzzy Logic (FL), and Artificial Neural Networks (ANN) (Wu 2015).

Because to date no comprehensive study of applications of different AI algorithms to various problems in the oil and gas industries has been conducted, we offer this summary of the most pertinent literature on the subject. The rest of this review is organized as follows.

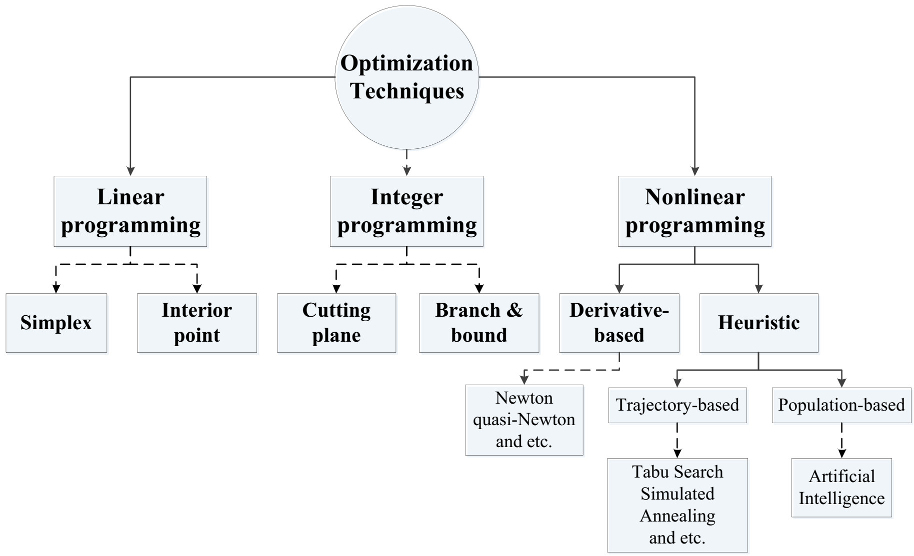


Fig. 1 Classification of optimization techniques (Mirzabozorg 2015; Mohagheghian 2016)

In the next section, we present a brief description of EA. In Sect. 3, we discuss features of SI and some relevant applications. In the subsequent Sect. 4, we offer a description of FL and complete the review of AI with a section on ANN and its applications.

2 Evolutionary algorithms

Based on the Darwinian evolutionary theory, over millions of years many species have evolved to adapt to different environments. Similarly, the same concept can be applied to numerical optimization if we consider the environment as a form of the problem and EA as an adaptation of the population to fit the best environment (Eiben and Smith 2003). The basic idea of EA is to evolve a population of candidate solutions under a selective process analogous to the natural selection, mutation, and reproduction until better solutions are obtained. Specifically, using effective search algorithms, parent solutions are combined to generate offspring solutions that can be evaluated and may themselves produce offspring (Husbands et al. 2007). Continuation of the generation cycle leads to better solutions to search, optimization, and design problems. EA includes many algorithms such as evolutionary programming, genetic algorithms, evolution strategies, and evolution programs (Mohaghegh 2000). Because of the outstanding performance of Genetic Algorithms (GA) and Differential Evolution (DE) in dealing with solutions to a variety of engineering problems, this category of techniques has become extremely popular in engineering applications and, thus, they are discussed in more detail in the next section.

2.1 Genetic algorithm (GA)

Holland (1992) introduced GAs as an evolution of biological species in a natural environment based on the principle of the "survival of the fittest." This stochastic optimization algorithm is

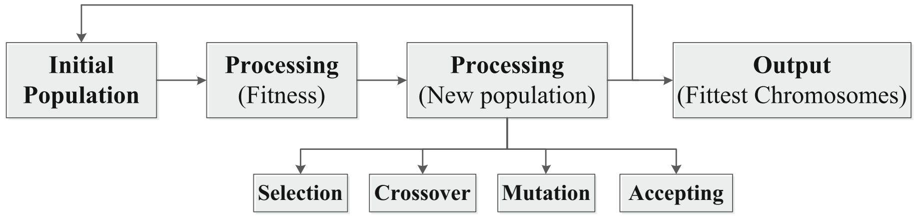


Fig. 2 Optimization process in GA terminology (Velez-Langs 2005; Chen 2013)

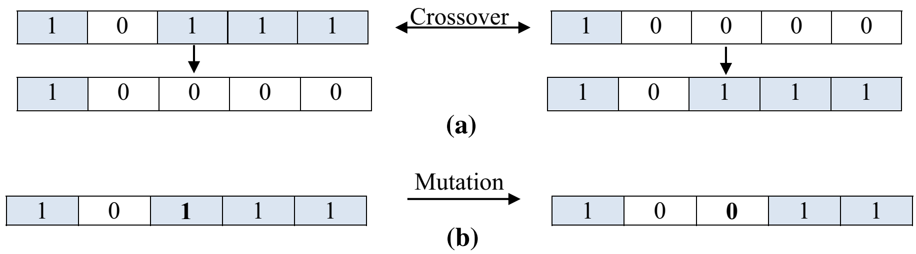


Fig. 3 **a** One point crossover example and **b** mutation example (Lin et al. 2011)

highly efficient, versatile, and suitable for multi-objective optimization problems. Although GA was initially suggested as an academic tool for investigation of biological processes, nowadays it is applied in many engineering fields due to its ability to handle multiple conflicting objectives (Güyagüler et al. 2002; Chen et al. 2010).

Application of GA to any problem requires the definition of three key parameters (Velez-Langs 2005):

(a) Chromosomes (representations of control vectors with n unknowns) that contain encoded variable strings (or sometimes called genes). Each gene represents a parameter (an unknown) and each chromosome represents a trial (or a possible solution).
(b) A large number of chromosomes (genotype), which represents the individuals of a GA population,
(c) The operations of selection, mutation, and crossover to produce a population from one generation (parents) to the next (offspring).

The mechanisms of a typical GA and the overall optimization process in the genetic terminology are shown in Fig. 2.

The first step of GA optimization is to produce the initial population or genotype as a partial space solution. The next step which is performed in each evolution cycle (iteration) is a modification of the population and calculation of the fitness of each chromosome (Algosayir 2012). Then, based on the fitness values of chromosomes, two parents with the highest values are chosen (selection) to exchange parts of their genetic information and produce an offspring or child (crossover) (Fathinasab and Ayatollahi 2016). According to the number of positions at which crossover occurs, there are two common types of crossover: one-point crossover and two-point crossover (Fig. 3a). Next, among the newly generated offspring those chromosomes with the best calculated fitness values are returned to the initial population (accepting). Note that before inserting the selected offspring back into the original population, they are mutated by changing some of their binary digits (genes) to span the search space more thoroughly and maintain variety in the population as shown in Fig. 3b (mutation) (Filgueiras et al. 2016).

GA advantages include the following:

- It requires limited parameter settings and its initialization starts with a population of parameters rather than a single parameter (Ab Wahab et al. 2015; Mohagheghian 2016);
- Probabilistic transition rules can be used instead of the deterministic ones (Algosayir 2012);
- A chromosome or a control vector is considered entirely rather than dealing with each individual parameter (Algosayir 2012);
- Direct function evaluations can be used instead of derivative calculations (Bittencourt and Horne 1997);
- The ability to be combined with other algorithms to improve optimization performance (Güyagüler et al. 2002);
- Easy parallelization can be applied for efficient calculation time (Mariajayaprakash et al. 2015).

Some disadvantages of GA include:

- Randomly selected regions among an initial population may lead to selection of inappropriate regions. Therefore, the evolution process is strongly dependent on the values of the initial members (Bittencourt and Horne 1997);
- GA tends to have slow convergence speed for complex optimization problems (Ballester and Carter 2007).

To improve the performance of GA and enhance the quality of its solutions, several alternative approaches for crossover and mutation have been proposed. Üçoluk (2002) and Jong and Spears (1992) recommended segmented and N-point crossover in which two breaking points and several random breaking points are used, respectively. For mutation, on the other hand, bitwise inversion has been utilized during which the genes in a chromosome are mutated using a random mutation rather than some assigned probability (uniform mutation) (Üçoluk 2002).

GA has been applied to multiple problems in the oil and gas E&P industry. Below, we provide some of the applications of GA in their chronological order.

Saemi et al. (2007) proposed a new algorithm for the auto-design of neural networks based on GA to predict permeability from well logs in South Pars gas field in the Persian Gulf. Farshi (2008) used continuous GA with the advantages of binary GA to optimize vertical and directional well placement problems. The results demonstrated that the modified GA model obtained higher objective function values in shorter time. To optimize the input variables of reservoir simulation models, Andersen (2009) used GA, ANN, and a combination of them with a commercial reservoir simulator software. Comparison of the model output with the results of other approaches including Matlab *fmincon* optimization and Hooke-Jeeves optimization, demonstrated promising performance of the proposed model.

Chen et al. (2010) combined GA with the Tabu search method to optimize the controlling variables such as water alternative gas (WAG) ratio, cycle time, injection rates and bottomhole pressures of the oil producers. The proposed model significantly improved the convergence speed, increased the recovery factor, and the Net Present Value (NPV). Edmunds et al. (2010) optimized a steam and solvent cycling process using GA method, which reduced the physical cumulative steam to oil ratio (CSOR) to the value close to one.

In the work of Algosayir (2012), GA, simulated annealing, and their hybridization with the orthogonal arrays and response surface proxy techniques were considered for steam and solvent applications in oil sands and fractured carbonate reservoirs. The results indicated that using a proxy saved 95% of the computational time, while utilizing the orthogonal arrays (with minimax criterion) improved model convergencey behavior for finding the yoptimaly solution. Chen (2013) designed an optimization tool using GA with the advantages of binary and continuous encoding and coupled it with a reservoir simulator to optimize the steam injection rates of a steam-assisted gravity drainage (SAGD) process in a saturated oil reservoir. Salmachi et al. (2013) built an integrated framework including a reservoir simulator, an optimization method (GA), and an economic objective function (NPV) to obtain the optimal locations of infill wells in coal bed methane reservoirs.

Later, Guria et al. (2014) developed a model using binary coded GA to find optimal variables in a drilling operation in one of the Louisiana offshore fields with abnormal formation pressure. They used objective functions such as drilling depth, drilling time, and the cost of drilling, whereas equivalent circulation density, rotary speed of the drill bit, weight on drill bit, and the Reynolds number in drill bit nozzles were considered as the control variables. The model output showed that using the optimal values of the control vector minimized drilling cost and time while maximizing drilling depth.

In the work of Xu et al. (2015), a modified GA with altered crossover and mutation rates was developed to history-match the simulation data with the experimental results of the vapor extraction (VAPEX) heavy oil recovery process. The modified approach resulted in 71% reduction of the computational time and an excellent match with the error less than 1% in comparison to conventional GA. Various novel optimization approaches using simultaneous perturbation stochastic approximation (SPSA), GA, and covariance matrix adaptation evolution strategy (CMA-ES) were proposed by Ma et al. (2015) for placement of horizontal wells and hydraulic fracturing (HF) stages in shale gas reservoirs. Their results showed the ability of SPSA, GA, and CMA-ES to handle various HF stage spacing intervals in geologic systems with homogeneous and heterogeneous petrophysical properties, whereas in large dimensional problems, GA, and CMA-ES had better performance than SPSA. Bian et al. (2016) recommended a support vector regression model with GA to predict pure and impure CO_2-crude oil MMP.

2.2 Differential evolution (DE)

DE is a population-based method that uses a real-coded GA with an adaptive random search (ARS) and a normal random generator to find the global minimum of the objective function (Boender and Romeijn 1995; Maria 1998). The main difference between GA and DE is that GA relies on crossover operation to find the optimal solution, while DE is mostly based on mutation operation (Ab Wahab et al. 2015). Like other evolutionary algorithms, DE consists of four stages: initialization, mutation, crossover, and selection. During the initialization step, a population/generation with a fixed number of candidate solutions (NP) using minimum and maximum values for each defined variable and a uniform random value in the range from 0 to 1 is created (Storn and Price 1995). The next step is to evolve the initial population in which every solution is mutated by adding the difference of two random solutions from the current population to a different selected random solution scaled by a factor F. Then, during the crossover process, diversity is created in the newly generated candidate solutions by applying the crossover probability rate (CR). There are two main crossover variants for DE: exponential and binomial. Finally, in the selection step, every solution vector in the trial population is compared with the corresponding vector in the initial population and depending on the nature of the problem (minimization or maximization), the one with the lower or higher objective function value is moved to the next generation. The four-step process repeated until the stopping criteria (reaching to the maximum number of generations or obtaining the defined different tolerance between the objective function values in the current generation and the previous one) are met. Figure 4 shows different steps of the DE algorithm.

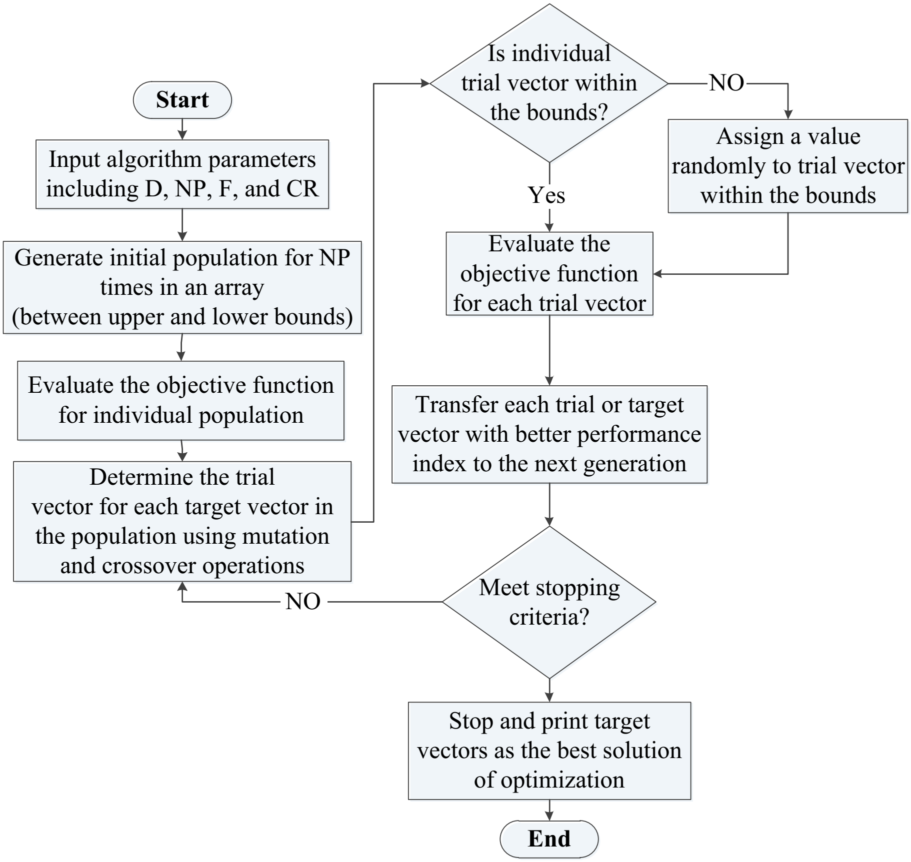


Fig. 4 The flow chart of DE algorithm (Khademi et al. 2010; Ab Wahab et al. 2015)

As mentioned above, there are three parameters that control DE performance: the size of population (NP), the crossover constant (CR), and the scaling factor (F). According to Price et al. (2006), 5–10 times the number of variables in a problem is usually suitable for NP, while a bound of 0.1 to 1 and 0.4 to 1 are recommended for CR and F, respectively.

Although DE has many advantages including robustness, simplicity in its structure, and enhanced ability for a local search, the algorithm has relatively slow convergence rate (Wu et al. 2011). Other advantages and disadvantages of DE algorithm are listed below (Khademi et al. 2010; Ab Wahab et al. 2015; Mirzabozorg 2015):

Advantages

- Ability to handle non-differentiable, nonlinear and multimodal functions.
- Ease of use, i.e. few control variables to steer the optimization.
- Good convergence properties to the global optimal point in consecutive independent trials.

Disadvantages

- Parameter tuning is necessary.

In general, the improvement of DE performance could be achieved by increasing the population size and introducing the elitism, which avoids the destruction of the best solution during creation of next generation.

Although DE method has been very successful in tackling various engineering problems, only a few applications of this technique in petroleum engineering are available in the literature as discussed below.

In their history matching study, Wang and Buckley (2006) used DE algorithm to match capillary pressures and relative permeabilities with core flooding data. Considering a cost function as the objective function, Decker and Mauldon (2006) optimized fracture shapes and sizes by applying this method. Their work indicated that DE algorithm is an appropriate tool for estimation of fracture characteristics obtained by geological analyses. In the work of Jahangiri (2007), the inflow control devices and DE approach were applied to optimize production constraints and maximize oil production from smart wells.

DE algorithm had been also successfully utilized to find optimal solutions of variogram properties in geostatistical systems (Zhang et al. 2009). In this study, it was concluded that DE method is more efficient and stable in comparison to GA technique for matching the key variogram variables with experimental data. Furthermore, Hajizadeh (2009) performed a comparison between DE method and the neighborhood algorithm (NA) for history matching of a black oil model. The results showed that DE technique performed better than NA method in matching the field data. Hajizadeh et al. (2010) also investigated convergence and robustness of different stochastic population-based optimization algorithms including DE, NA, and ant colony optimization (ACO) as a part of a history matching study. The results showed that DE algorithm had the fastest convergence rate and the lowest misfit value, whereas NA technique did not perform as well for a problem with a large number of unknown parameters.

To enhance capabilities of DE algorithm, Wang et al. (2011) introduced a new method called co-operative co-evolutionary differential evolution (CCDE) and applied it for a waveform inversion of cross-well data. Their results showed better performance of CCDE technique in comparison to that of conventional DE method.

Zhang et al. (2014) applied DE algorithm to a micro-seismic study to develop a robust velocity model for more accurate data interpretation. Awotunde and Mutasiem (2014) minimized the cost of drilling operation using DE algorithm by optimizing drilling operational parameters and reducing drilling time. In another work, Mirzabozorg (2015) compared the performance of DE method with that of PSO algorithm in a history matching study in a homogeneous 2D and a heterogeneous 3D SAGD reservoir models. The results confirmed more efficient performance of DE technique in comparison to that of PSO scheme from various perspectives. Santhosh and Sangwai (2016) coupled a novel algorithm called hybrid differential evolution (HDE) with the neighborhood approximation Bayes (NAB) algorithm in a history matching study to predict reservoir production with minimal uncertainty and simulation runs.

3 Swarm intelligence

Swarm intelligence (SI) is an innovative intelligent optimization technique that mimics social and collective behavior of swarms of ants, bees, fish schools, and insects when they are searching for food, communicating with each other, and mingling in their colonies (Abraham et al. 2006; Engelbrecht 2006). Main characteristics of SI models are their self-organization, decentralization, communication, and cooperation behaviors between individuals within the group

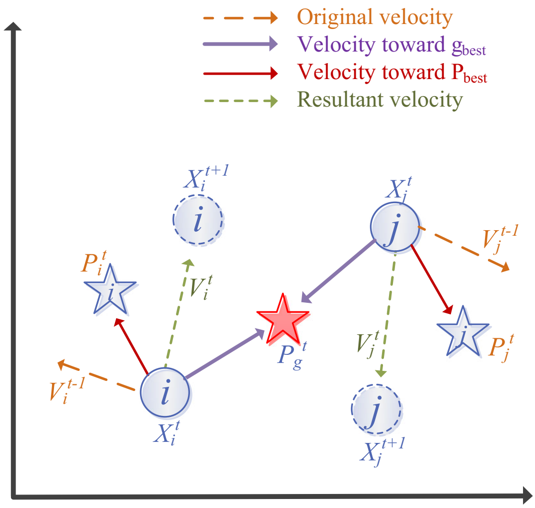


Fig. 5 Inertia weight particle trajectory (Mohagheghian 2016)

in absence of a central controlling system. Although these individual interactions are simple, eventually they lead to a complex global behavior, which is the core of SI (Bonabeau and Meyer 2001). In recent years, a lot of SI-based techniques have been proposed, which cover various research areas (Edelen 2003; Kamrani 2010; Ganesan et al. 2013; Alam et al. 2014; Senthilkumar 2014). Among those techniques, two methods became particularly popular and widely used for solving discrete and continuous optimization problems: Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) methods (Liu and Passino 2000; Trelea 2003). Here, we discuss the theory of PSO followed by its chronological use in the petroleum industry.

3.1 Particle swarm optimization (PSO)

Eberhart and Kennedy (1995) proposed PSO technique based on a natural pattern of bird flocking or fish schooling. Like GA technique, PSO algorithm starts with a randomly generated population, uses a fitness function value to evaluate the population, as well as updates the population and the search with random techniques. However, PSO technique does not use crossover and mutation operators. It considers particles with two main parameters: a vector corresponding to a unique position in the search space and a velocity for the motion of the particle (Kaewkamnerdpong and Bentley 2005). As its starting point, the algorithm randomly produces the particles' positions and velocities. Then, each particle updates its position and the associated velocity until satisfactory solutions are achieved. In other words, using the velocity function (regularly updated), each particle goes iteratively through the search space according to its best position and the entire group's best position (Mohamed et al. 2011). Therefore, the particle's experience would be tracking and memorizing the best encountered positions and the best population size, which is called a swarm. PSO combines a particle's best experience (*pbest*) and the other's best experiences (*gbest*) to update the particle's position in each iteration (Mirzabozorg 2015). Figure 5 shows a typical trajectory of a particle with respect to each term explained above.

A typical workflow of PSO method is presented in Fig. 6.

PSO algorithm has several advantages including the need for a small number of parameters as the input data, simplicity for implementation, high efficiency in a global optimum

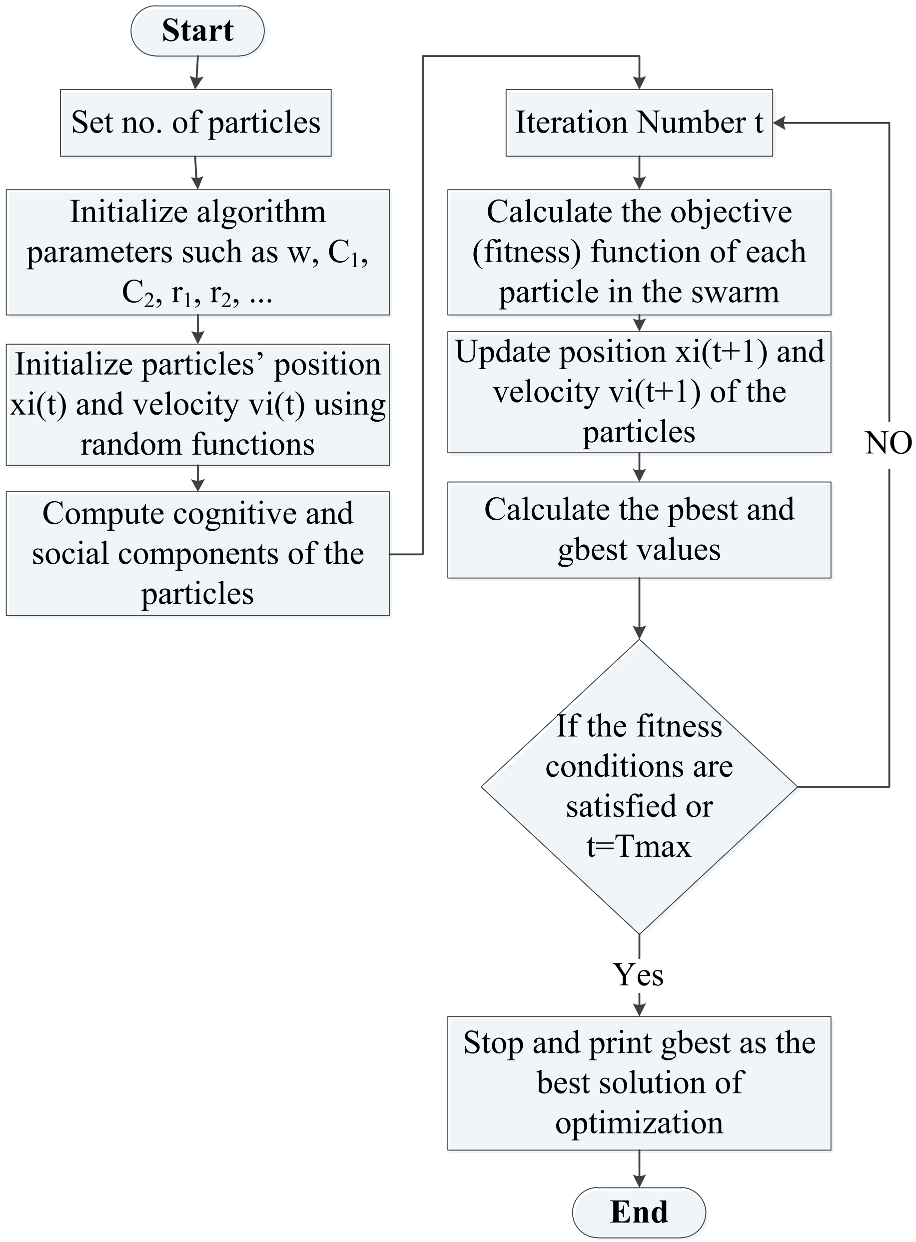


Fig. 6 PSO diagram (Kathrada 2009; Assareh et al. 2010)

search, and flexibility in scaling of design search. However, it has a tendency to converge slowly (in other words, weak local search ability). Increasing the population size, dynamic velocity adjustment, and sub-swarm approach are various strategies that have been proposed to increase the convergence probability towards a global optimum (Atyabi and Powers 2013; Chang and Yu 2013; Ab Wahab et al. 2015).

Recently, PSO has been successfully applied in many areas such as continuous function optimization (Eberhart and Kennedy 1995), neural network training (Shen et al. 2006), static function optimization (Shi and Eberhart 1999), dynamic function optimization (Blackwell and Branke 2004), multimodal optimization (Brits et al. 2002), and data clustering (Cohen and de Castro 2006). However, its application in oil industry goes back for about six years.

Kathrada (2009) studied PSO implementation for history matching of finite difference simulation models. He showed that even with a highly efficient algorithm like “Flexi-PSO”, there was no guarantee that the obtained predictions would cover the true reservoir behavior range. Based on socio-economic indicators including population, GDP (gross domestic product), import and export data and the NPV as the objective function, Onwunalu and Durlofsky (2010) applied GA and PSO methods to determine optimal well type (including vertical, directional, and dual-lateral) and location. Onwunalu and Durlofsky (2010) developed PSO and GA demand estimation models (PSO-DEM and GA-DEM) in exponential and linear forms to estimate future oil demand values up to year 2030. They reported that the models based on PSO algorithm had lower average relative errors.

To optimize oil recovery of a heavy oil reservoir, Wang and Qiu (2013) investigated convergence behavior and performance of three different PSO algorithms. The results indicated that conventional PSO yielded the highest objective function. Humphries et al. (2014) linked PSO and the generalized pattern search (GPS) in a simultaneous and sequential manner to optimize well placement and control problems. The results showed better performance of the sequential approach in comparison to that of the simultaneous approach. By integrating uniform design (UD) into the initialization process of PSO, Zhou et al. (2016) developed a hybrid method to maximize the NPV of a cyclic steam stimulation project. The results showed that integration of the initialization process of PSO with UD increased the quality of the initial population and, consequently, the convergence rate. Additionally, utilizing the recommended hybrid techniques led to the improvement of technical and economic scenarios for heavy oil reservoirs development. In another study, considering squeeze lifetime, total injected squeeze volume, and injected water volume as the objective functions, Vazquez et al. (2016) utilized PSO algorithm to determine the most effective chemical scale inhibitor squeeze designs in two field cases. The proposed model was successful in identifying the most cost-effective scenario among all chemical scale inhibitor squeeze cases.

Mohagheghian (2016) used GA and PSO algorithms to optimize hydrocarbon WAG performance in the E-segment of Norne field. In comparison to the reference cases, the results showed that the best overall values of NPV found by GA and PSO were 13.8 and 14.2% higher, respectively, while for incremental recovery factor as the objective function, he obtained an increase of 14.2% in the case of GA and 16.2% in the case of PSO.

4 Fuzzy logic

Fuzzy logic (FL) is a powerful mathematical tool for modeling the uncertainty of information in the real world by generalizing any specific theory from a crisp (discrete) to a continuous (fuzzy) form (Zadeh 1965). Each variable of FL commonly consists of a truth value that ranges in a degree between 0 and 1 and between completely true and completely false (Novák et al. 2012). As shown in Fig. 7, FL generally consists of three essential components including Fuzzification unit, Knowledge base (database and rule base) and Reasoning mechanism, and Defuzzification (Kar et al. 2014).

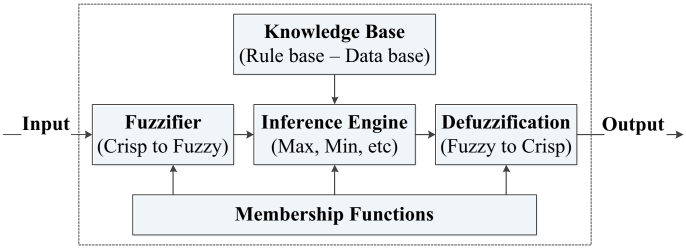
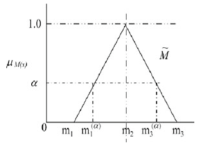
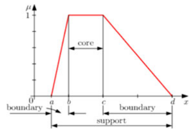
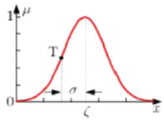
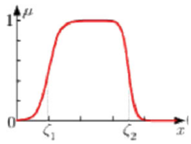
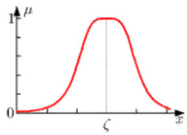


Fig. 7 Fuzzy inference systems (Zoveidavianpoor et al. 2012; Wu 2015)

Table 1 The major function used in fuzzy logic system (Zoveidavianpoor et al. 2012)

Function name	Parametric representation	Graphical representation
Triangular function	$\mu\left(x,a,b,c\right)=\left\{\begin{matrix} 0 & x\le a \\ \frac{(x-a)}{(b-a)} & x\in(a,b) \\ \frac{(c-x)}{(c-b)} & x\in(b,c) \\ 0 & x\ge c \end{matrix}\right\}$	
Trapezoidal function	$A=\left\{\begin{matrix} 0 & x\le a \\ \frac{(x-a)}{(b-a)} & x\in(a,b) \\ 1 & x\in(b,c) \\ \frac{(d-x)}{(d-c)} & x\in(c,d) \end{matrix}\right\}$	
Normalized Gaussian function	$\mu\left(x,\sigma_i,c_i\right)=e^{-\frac{(x-c_i)^2}{2\sigma_i^2}}$	
Two Sigmoidal functions	$\mu\left(x,\alpha_1,\xi_1,\alpha_2,\xi_2\right)=\left[1+e^{-\alpha_1(x-\xi_1)}\right]^{-1}-\left[1+e^{-\alpha_2(x-\xi_2)}\right]^{-1}$	
Generalized Bell function	$\mu\left(x,\alpha_1,\beta,\xi\right)=\left[1+\left\vert\frac{x-\xi}{\alpha}\right\vert^{2\beta}\right]^{-1}$	

In Fuzzification step, crisp inputs are converted into fuzzy (continuous) values using fuzzy sets. The fuzzy set is a set of ordered pairs for which the membership function allows each input element to have its membership grade between 0 and 1 (Jang et al. 1997). In general, the membership function is used to characterize its corresponding fuzzy set. There are five basic membership functions as presented in Table 1.

In fuzzy reasoning step, using fuzzy values from previous step and standard operations, a fuzzy engine executes a fuzzy inference procedure according to the pre-determined rule base and data base to obtain a reasonable output or conclusion. Rule base and data base, which are categorized in knowledge base section, are the core of Fuzzy Inference System (Wu 2015).

Rule base contains a reasonable and careful selection of fuzzy rules and accomplishes the procedure of mapping fuzzy rules of the inputs to fuzzy values of the outputs, while data base defines the membership function of input and output elements (Freuder and Wallace 2005). Table 2 presents three types of standard operations in FL systems including intersection, union, and complement where their selection depends solely on a problem. Furthermore, the intersection and union operations are based on min/max operations, while the complement is an algebraic operation (Zoveidavianpoor et al. 2012).

In Defuzzification step, consequents given by fuzzy reasoning step are converted into crisp values (Mardani et al. 2015).

The benefits of FL can be summarized as follows (De Reus 1994; Kar et al. 2014):

- FL is simple, fast, robust, and insensitive to changing environments
- FL describes systems as a combination of numeric and symbolic
- FL addresses the problems with very restricted conditions or without exact solutions.
- The algorithm could be described with little and/or approximate data.

Although the advantages of FL are numerous, there are certain situations in which FL does not perform well (De Reus 1994):

- Cases for which good mathematical descriptions and solutions exist, the use of FL might be sensible only when computing power restrictions are too severe for a complete mathematical implementation.
- It is difficult to prove the characteristics of fuzzy systems in most cases due to the lack mathematical descriptions (e.g., in the area of stability of control systems).

Nowadays, there are numerous applications of FL in the oil and gas industry. Using a fuzzy neural network and a backpropagation neural network, Ouenes (2000) evaluated factors affecting rock fracturing and measured their relationship with fracture intensity. The author found that the use of FL method and a stochastic framework can minimize the risks associated with data driven techniques and assist the interpretation process. Some studies such as Yang (2009) and Yin and Wu (2009), described the use of FL for stimulation candidate-well selection. In Yang's study, to determine fuzzy variables, the author conducted an analysis over the factors affecting oil well fracturing, and then selected target wells and formations for hydraulic fracturing using a fuzzy mathematics model (Yang 2009). To choose a candidate well for fracturing, Yin and Wu (2009) designed a fuzzy judging mathematical model by analyzing different effective parameters, determining the relation between fracturing effects and the parameters, and dividing the grade intervals of each influencing parameter.

Yetilmezsoy et al. (2011) developed an approach using adaptive neuro-fuzzy system (ANFIS) for modeling water-in-oil emulsion formation in terms of percentage of SARA (saturates, aromatics, resins, asphaltene), viscosity, and density data. Attia et al. (2013) described a model using FL and neural networks to predict the multiphase flow pressure drop in surface pipelines for oil fields. The results showed that ANFIS outperformed Beggs and Brill, Dukler Flannigan, Dukler Eaton Flannigan correlations, and neural networks.

Afshar et al. (2014) employed FL and neural network models to calculate bubble point pressure as a function of gas specific gravity, oil gravity, solution gas oil ratio, and reservoir temperature. To avoid trapping in local minima and increase the accuracy, they utilized models optimized with GA. The results showed that optimization with GA could prevent their neural network and neuro-fuzzy models from trapping in local minima, which is a common case for a back-propagation algorithm. Ravandi et al. (2014) generated two new optimized models using GA and ANFIS for porosity calculation and water saturation determination. Comparison between the obtained results and the outputs of other methods showed that the

Table 2 Standard operations in FL system (Zoveidavianpoor et al. 2012)

Fuzzy operator name	Algebraic operator name	Symbol	Description	Equation based on	
				Algebraic	Fuzzy
Intersection	AND	$\cap$	Applied to two fuzzy sets A and B with the membership functions $\mu_A(x)$and $\mu_B(x)$	$\mu_{A\cap B} = \{\mu_A(x), \mu_B(x)\}$, x $\in$ X	$\mu_{A\cap B} = \min\{\mu_A(x), \mu_B(x)\}$, x $\in$ X
Union	OR	$\cup$	Applied to two fuzzy sets A and B with the membership functions $\mu_A(x)$and $\mu_B(x)$	$\mu_{A\cup B} = \{\mu_A(x), \mu_B(x)\}$, x $\in$ X	$\mu_{A\cup B} = \max\{\mu_A(x), \mu_B(x)\}$, x $\in$ X
Complement	NOT	NOT	Applied to fuzzy sets A with the membership functions $\mu_A(x)$	$\mu_A = 1 - \mu_A(x)$, x $\in$ X	$\mu_A = 1 - \mu_A(x)$, x $\in$ X

accuracy of the recommended models for estimating porosity and water saturation improved considerably (mean standard error of 0.00007 and 0.00033, respectively). In another work, Ahmadi and Ebadi (2014) designed FL method with different types of membership functions (e.g., curve shaped, triangular, and trapezoidal shape) to specify MMP of injected gas and reservoir oil. They concluded that a curve-shaped membership function demonstrated a better match with experimental results unlike other tested types of membership function. Rammay and Abdulraheem (2014) coupled ANFIS with DE method to reproduce production data of an arbitrary reservoir model.

Wu (2015) showed the feasibility of Fuzzy Inference System (FIS) and ANN techniques in pipeline risk assessment by developing two hybrid pipeline risk assessment systems including FIS and ANN with an expert risk assessment methodology. Olatunji et al. (2015) proposed a hybrid system through a combination of FL systems and a sensitivity-based linear learning method to model both permeability and PVT properties of oil and gas reservoirs. In their work, FL system was used to manage uncertainties in the reservoir data and training the output. In another research work, conducting different natural depletion tests at various temperatures, Mohammadi et al. (2015) developed a FL model to predict asphaltene precipitation in the range of experiment temperatures. Comparative studies were carried out between the model results and the output of the WinProp module (computer modelling group (CMG) software), which showed an acceptable performance of their FL model.

Bakyani et al. (2016) designed a model based on ANFIS and optimized it by PSO algorithm to forecast carbon dioxide diffusivity in oils at different reservoir temperatures and pressures. Zhou et al. (2016) designed an analytic model using FL to assess corrosion failure likelihood (CFL) in a natural gas pipeline. They considered corrosion thinning factor, corrosion cracking factor, inspection effectiveness, and inspection times as the key factors and determined the fuzzy rules between the key factors and CFL. The results demonstrated success in application of the proposed model and feasibility of its use as a reference for pipeline inspection and maintenance plans. Later in 2016, Jalalnezhad and Kamali (2016) presented a novel method using an experimental dataset and ANFIS to model the thickness of wax precipitation in single-phase turbulent flow. According to their results, ANFIS model demonstrated a promising performance for oil production optimization and wax deposition thickness prediction in single-phase turbulent flow.

5 Artificial neural network (ANN)

An artificial neural network (ANN) consists of a pool of simple processing units, which communicate by sending signals to each other over a large number of weighted connections. A list of its characteristics can be summarized as follows (Rumelhart et al. 1986; Cybenko 1989):

- a set of processing units (neurons),
- a state of activation y_k for every unit which is equivalent to the output of the unit,
- connections between units; generally, each connection is defined by a weight w_{jk} which determines the effect that the signal of unit j has on unit k,
- a propagation rule that determines the effective input s_k of a unit,
- an activation or transfer function F_k, that determines the new level of activation based on the effective input $s_k(t)$,
- an external input (bias, offset) θ_k for each unit used for a better match of the neural network model to the real one.

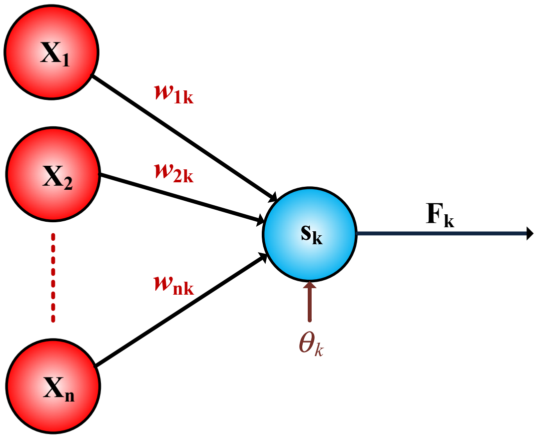


Fig. 8 A simple neuron model (Mohaghegh 2000)

Figure 8 illustrates the basics for a neuron. Each unit performs a relatively simple job: receiving input from its neighbors or external sources and using this input to compute an output signal which is then propagated to other units. Apart from this processing, the second task is adjustment of the weights. The system is inherently parallel in the sense that many units can carry out their computations at the same time. In most cases, it is assumed that each unit provides an additive contribution to the input of the unit with which it is connected. As shown in Fig. 8, the total input ($s_k(t)$) to the unit k is a simple weighted sum of the separate outputs from each of the connected units plus a bias or offset term θ_k (Ramadhas et al. 2006):

Then, a rule that calculates the effect of the total input on the output of the unit is required. Hence, a function (F_k) that takes the total input $s_k(t)$ and produces a new value of the activation of the unit k (activation or transfer function) is needed (Mohanty 2005). The transfer functions mainly serve as a type of filter or a gate that allows some signals to move forward and to stop others as they progress from the input nodes to the output ones. The most commonly used activation functions are logarithmic sigmoid, hyperbolic tangent sigmoid, and linear functions (Mohanty 2005).

There are several patterns of ANNs such as feed-forward networks, recurrent networks, etc. Conventional feed-forward networks are the most common networks for function approximation (Eslamloueyan and Khademi 2009). A multi-layer feed-forward network which consists of groups of interconnected nodes arranged in layers corresponding to input, hidden, and output nodes, is shown in Fig. 9.

The advantages of ANNs with respect to other models include the following (Mohaghegh 2000; Dumitru and Maria 2013):

- ANNs are a relatively simple learning algorithm;
- They have an ability to outperform other models when high-quality data are available;
- They can approximate any function, regardless of its linearity;
- ANNs can be used in problems for which it is difficult or impractical to formulate a non-linear relationship.

ANNs also have some significant drawbacks (Groth 2000; Dumitru and Maria 2013):

- It is hard to understand and interpret ANN models, because they are "black box" prediction engines; however, with the new tools on the market, this problem has been alleviated.
- ANNs are susceptible to overtraining meaning that they just memorize their training data and are not capable of generalization. Note that in recent years, commercial-grade

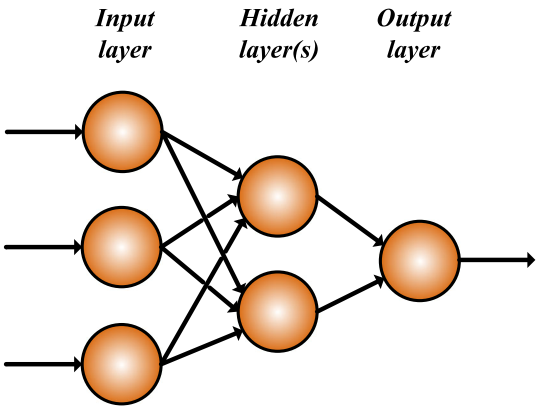


Fig. 9 Schematic of a multi layer feed-forward neural network model (Ahmadi 2011)

neural networks have eliminated overtraining by monitoring test versus training errors and through "bootstrapping holdout (test) samples;"
- Their predictions are not acceptable over small data sets.

Below we provide some applications of ANN in the oil industry reviewed in a chronological order.

Yilmaz et al. (2002) described a model based on back-propagation ANN and fractal geostatistics to solve the optimal bit selection problem in terms of real rock bit data, gamma ray and sonic log data for several wells in a carbonate field. In the work of Huang et al. (2003), an ANN model was presented to predict MMP of pure and impure CO_2 and oil systems. This approach used the molecular weight of the C_{5+} fraction, reservoir temperature, and concentrations of volatile (methane) and intermediate (C_2–C_4) fractions in the oil. Comparison between the results predicted by the ANN model and other statistical methods demonstrated that the ANN model accurately estimated CO_2 MMP in oil reservoirs. Chapoy et al. (2007) presented a feed-forward ANN model for estimating hydrate dissociation pressures of natural gases in the presence/absence of inhibitor aqueous solutions with 19 input variables.

To predict asphaltene precipitation in an oil reservoir, Ahmadi (2011) utilized a hybrid model combining a feed-forward neural network and the imperialist competitive algorithm (ICA). He evaluated the performance of this ICA-ANN model in comparison to a scaling model and a conventional ANN model and showed the effectiveness of the former model. To predict asphaltene precipitation due to natural depletion, Ahmadi and Golshadi (2012) developed a feed-forward ANN optimized by hybrid GA and PSO (denoted as HGAPSO) techniques. Zendehboudi et al. (2012) used a feed-forward ANN optimized with PSO to predict the condensate-to-gas ratio in a gas condensate reservoir. Based on their error analyses, the proposed PSO-ANN outperformed conventional ANN and empirical correlations.

Ahmadi et al. (2013) described a methodology using a feed-forward ANN model optimized by GA and PSO methods to examine real field data for forecasting reservoir permeability. Good performance of the model in comparison to those of conventional ANN validated the accuracy of the hybrid model. Zendehboudi et al. (2014) proposed a hybrid model of ANN and PSO for predicting the production performance of steam assisted gravity drainage (SAGD) in heavy-oil fractured reservoirs. The error analysis indicated a good agreement between the output of PSO-ANN model and the actual data. To determine the conditions of offshore oil and gas pipelines, El-Abbasy et al. (2014) developed an ANN model using the datasets from three existing offshore oil and gas pipelines in Qatar. The authors considered different factors including diameter, material, type of the carried product, anode wastage, support condition, joint condition, free spans, and corrosion.

In another work, Ahmadi et al. (2014) developed a correlation using multivariable regression, back propagation ANN (BP-ANN) and GA-ANN to predict the recovery rate of vapor extraction in heavy oil reservoirs. They conducted comparative studies between the model outputs and experimental data. From the statistical errors, they found that the proposed GA-ANN outperformed the conventional BP-ANN and regression correlation. They also demonstrated the ability of GA-ANN to search in different directions simultaneously, which increased the probability of finding the global optimum. Xue et al. (2014) proposed a GA-BP-ANN model to identify the fracture in terms of deep-shallow laterolog curves and micro-electrode logging curves. The model prediction was in good agreement with the reservoir production performance, which proved high accuracy of the utilized method.

The work of Chiroma et al. (2015) described the implementation of GA and neural network (GA-NN) to predict the West Texas Intermediate (WTI) crude oil price. In the proposed technique, GA was used to optimize the weights, bias, and topology of the neural network. The comparison of the model output with those of ten back-propagation algorithms indicated better performance and higher computational efficiency of the GA–NN model. Azizi et al. (2016) used ANN to estimate the water hold up in a two-phase flow vertical and inclined pipeline (90°, 75°, 60°, and 45° from horizontal). In their work, the input parameters were the pipe inclination angle and water and oil superficial velocities, whereas water holdup values of two-phase flow were considered as the output parameters. The predicted water hold-up by the proposed ANN model matched well with the experimental water holdup data. Zhang et al. (2016) predicted the wettability and modelled nonlinear relationship between wettability, rock, and fluid properties by developing a general regression neural network model containing nine influence factors. Kim et al. (2017) used ANN to address the storage efficiency of CO_2 sequestration in deep saline aquifers. The evaluation of the ANN model with the field scale data indicated a very good match (determination coefficient (R^2) of 0.96).

6 Conclusions and recommendations

This review has presented a comprehensive summary of different optimization methods in the field of AI and their applications in the petroleum E&P industry. We reviewed and categorized the pool of algorithms with respect to different types of AI, application areas, and publication year. We presented a general resource allocation for AI algorithms by dividing it into four groups, i.e., EA, SI, FL, and ANN and specified the most popular techniques for the first two categories.

The research shows that the application of AI methods has demonstrated outstanding performance in prediction, estimation, and optimization of different objective functions (e.g., minimum miscibility pressures, oil production rate, asphaltene precipitation around wellbore, well placement, and reservoir characterization). In general, faster convergence has been reported for PSO algorithm in comparison to DE and GA methods, whereas DE yielded superior optimal solutions with respect to GA and PSO approaches. On the other hand, hybridization of FL and ANN methods with other optimization algorithms (e.g., GA and neural network (GA-NN), Fuzzy Inference System and Artificial Neural Networks (FIS-ANNs), and the combinations of DE and ANFIS) has been shown to be more efficient and has obtained better solutions in comparison with the conventional FL and ANN models.

The focus of this review is on the use of GA, DE, and PSO techniques among Evolutionary computation and SI algorithms, while a future work can be expanded to include other methods such as Ant Colony Optimization.

References

Ab Wahab MN, Nefti-Meziani S, Atyabi A (2015) A comprehensive review of swarm optimization algorithms. PLoS ONE 10(5):e0122827

Abraham A, Guo H, Liu H (2006) Swarm intelligence: foundations, perspectives and applications. Swarm intelligent systems. Springer, Berlin, pp 3–25

Afshar M, Gholami A, Asoodeh M (2014) Genetic optimization of neural network and fuzzy logic for oil bubble point pressure modeling. Korean J Chem Eng 31(3):496–502

Ahmadi MA (2011) Prediction of asphaltene precipitation using artificial neural network optimized by imperialist competitive algorithm. J Pet Explor Prod Technol 1(2–4):99–106

Ahmadi M-A, Ebadi M (2014) Fuzzy modeling and experimental investigation of minimum miscible pressure in gas injection process. Fluid Phase Equilib 378:1–12

Ahmadi MA, Golshadi M (2012) Neural network based swarm concept for prediction asphaltene precipitation due to natural depletion. J Pet Sci Eng 98:40–49

Ahmadi MA, Zendehboudi S, Lohi A et al (2013) Reservoir permeability prediction by neural networks combined with hybrid genetic algorithm and particle swarm optimization. Geophys Prospect 61(3):582–598

Ahmadi MA, Zendehboudi S, Bahadori A et al (2014) Recovery rate of vapor extraction in heavy oil reservoirs? Experimental, statistical, and modeling studies. Ind Eng Chem Res 53(41):16091–16106

Alam S, Dobbie G, Koh YS et al (2014) Research on particle swarm optimization based clustering: a systematic review of literature and techniques. Swarm Evol Comput 17:1–13

Algosayir MM (2012) Optimization of steam/solvent injection methods: application of hybrid techniques with improved algorithm configuration. University of Alberta, Edmonton

Andersen MG (2009) Reservoir production optimization using genetic algorithms and artificial neural networks Master's thesis, Department of Computer Science, NTNU. http://hdl.handle.net/11250/251403

Assareh E, Behrang M, Assari M et al (2010) Application of PSO (particle swarm optimization) and GA (genetic algorithm) techniques on demand estimation of oil in Iran. Energy 35(12):5223–5229

Attia M, Mahmoud M, Abdulraheem A et al (2013) Evaluation of the pressure drop due to multi phase flow in horizontal pipes using fuzzy logic and neural networks. In: SPE Middle East oil and gas show and conference, society of petroleum engineers

Atyabi A, Powers DM (2013) Cooperative area extension of PSO-transfer learning vs. uncertainty in a simulated swarm robotics. In: ICINCO (1)

Awotunde AA, Mutasiem MA (2014) Efficient drilling time optimization with differential evolution. In: SPE Nigeria annual international conference and exhibition, society of petroleum engineers

Azizi S, Awad MM, Ahmadloo E (2016) Prediction of water holdup in vertical and inclined oil–water two-phase flow using artificial neural network. Int J Multiph Flow 80:181–187

Bakyani AE, Sahebi H, Ghiasi MM et al (2016) Prediction of CO_2–oil molecular diffusion using adaptive neuro-fuzzy inference system and particle swarm optimization technique. Fuel 181:178–187

Ballester PJ, Carter JN (2007) A parallel real-coded genetic algorithm for history matching and its application to a real petroleum reservoir. J Pet Sci Eng 59(3):157–168

Bian X-Q, Han B, Du Z-M et al (2016) Integrating support vector regression with genetic algorithm for CO_2–oil minimum miscibility pressure (MMP) in pure and impure CO_2 streams. Fuel 182:550–557

Bittencourt AC, Horne RN (1997) Reservoir development and design optimization. In: SPE annual technical conference and exhibition, society of petroleum engineers

Blackwell T, Branke J (2004) Multi-swarm optimization in dynamic environments. Workshops on applications of evolutionary computation. Springer, Berlin

Boender CGE, Romeijn HE (1995) Stochastic methods. Handbook of global optimization. Springer, Berlin, pp 829–869

Bonabeau E, Meyer C (2001) Swarm intelligence: a whole new way to think about business. Harv Bus Rev 79(5):106–115

Brits R, Engelbrecht AP, Van den Bergh F (2002) A niching particle swarm optimizer. In: Proceedings of the 4th Asia-Pacific conference on simulated evolution and learning. Orchid Country Club, Singapore

Carroll JA Jr, Horne RN (1992) Multivariate optimization of production systems. J Pet Technol 44(07):782–831

Chang Y, Yu G (2013) Multi-sub-swarm PSO classifier design and rule extraction In: International workshop on cloud computing and information security (CCIS), Atlantis Press
Chapoy A, Mohammadi AH, Richon D (2007) Predicting the hydrate stability zones of natural gases using artificial neural networks. Oil Gas Sci Technol Revue de l'IFP 62(5):701–706
Chen Z (2013) A genetic algorithm optimizer with applications to the SAGD process. University of Calgary, Calgary
Chen S, Li H, Yang D et al (2010) Optimal parametric design for water-alternating-gas (WAG) process in a CO_2-miscible flooding reservoir. J Can Petrol Technol 49(10):75–82
Chiroma H, Abdulkareem S, Herawan T (2015) Evolutionary neural network model for West Texas Intermediate crude oil price prediction. Appl Energy 142:266–273
Cohen SC, de Castro LN (2006) Data clustering with particle swarms. In: 2006 IEEE international conference on evolutionary computation. IEEE
Cybenko G (1989) Approximation by superpositions of a sigmoidal function. Math Control Signals Syst 2(4):303–314
De Jong KA, Spears WM (1992) A formal analysis of the role of multi-point crossover in genetic algorithms. Ann Math Artif Intell 5(1):1–26
De Reus N (1994) Assessment of benefits and drawbacks of using fuzzy logic, especially in fire control systems. Fysisch en Elektronisch Lab TNO the Hague (Netherlands)
Decker J, Mauldon M (2006) Determining size and shape of fractures from trace data using a differential evolution algorithm. Golden Rocks 2006, the 41st US Symposium on Rock Mechanics (USRMS), American Rock Mechanics Association
Dumitru C, Maria V (2013) Advantages and disadvantages of using neural networks for predictions. Ovidius Univ Ann Ser Econ Sci 13(1):444–449
Eberhart RC, Kennedy J (1995) A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science, New York, NY
Edelen MR (2003) Swarm intelligence and stigmergy: robotic implementation of foraging behavior Master's thesis,Faculty of the Graduate School of the University of Maryland. http://hdl.handle.net/1903/107
Edmunds N, Peterson J, Moini B (2010) Method for viscous hydrocarbon production incorporating steam and solvent cycling, Google Patents
Eiben AE, Smith JE (2003) Introduction to evolutionary computing. Springer, Berlin
El-Abbasy MS, Senouci A, Zayed T et al (2014) Artificial neural network models for predicting condition of offshore oil and gas pipelines. Autom Constr 45:50–65
Engelbrecht AP (2006) Fundamentals of computational swarm intelligence. Wiley, New York
Eslamloueyan R, Khademi M (2009) Estimation of thermal conductivity of pure gases by using artificial neural networks. Int J Therm Sci 48(6):1094–1101
Farshi MM (2008) Improving genetic algorithms for optimum well placement. Stanford University, Stanford
Fathinasab M, Ayatollahi S (2016) On the determination of CO_2-crude oil minimum miscibility pressure using genetic programming combined with constrained multivariable search methods. Fuel 173:180–188
Filgueiras PR, Portela NlA, Silva SR et al (2016) Determination of saturates, aromatics, and polars in crude oil by 13C NMR and support vector regression with variable selection by genetic algorithm. Energy Fuels 30(3):1972–1978
Freuder E, Wallace M (2005) Constraint programming. In: Burke EK, Kendall G (eds) Search methodologies: introductory tutorials in optimization and decision support techniques, Chapter 9. Springer, pp 243–265
Fujii H, Horne R (1995) Multivariate optimization of networked production systems. SPE Prod Facil 10(03):165–171
Ganesan T, Elamvazuthi I, Shaari KZK et al (2013) Swarm intelligence and gravitational search algorithm for multi-objective optimization of synthesis gas production. Appl Energy 103:368–374
Gomory RE (1958) Outline of an algorithm for integer solutions to linear programs. Bull Amer Math Soc 64(5): 275–278. https://projecteuclid.org/euclid.bams/1183522679
Groth R (2000) Data mining: building competitive advantage. Prentice Hall PTR, Englewood Cliffs
Guria C, Goli KK, Pathak AK (2014) Multi-objective optimization of oil well drilling using elitist non-dominated sorting genetic algorithm. Pet Sci 11(1):97–110
Güyagüler B, Horne RN, Rogers L et al (2002) Optimization of well placement in a Gulf of Mexico waterflooding project. SPE Reserv Eval Eng 5(03):229–236
Hajizadeh Y, Christie MA, Demyanov V (2009) Application of differential evolution as a new method for automatic history matching. In: Kuwait international petroleum conference and exhibition, society of petroleum engineers
Hajizadeh Y, Christie MA, Demyanov V (2010) Comparative study of novel population-based optimization algorithms for history matching and uncertainty quantification: PUNQ-S3 revisited. In: Abu Dhabi international petroleum exhibition and conference, society of petroleum engineers

Holland J (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology. Control and artificial intelligence. MIT Press, Cambridge
Huang Y, Huang G, Dong M et al (2003) Development of an artificial neural network model for predicting minimum miscibility pressure in CO_2 flooding. J Pet Sci Eng 37(1):83–95
Humphries TD, Haynes RD, James LA (2014) Simultaneous and sequential approaches to joint optimization of well placement and control. Comput Geosci 18(3–4):433–448
Husbands P, Copley P, Eldridge A et al (2007) An introduction to evolutionary computing for musicians. Evolutionary computer music. Springer, Berlin, pp 1–27
Jahangiri H (2007) Production optimization using smart well technology with differential evolution algorithm. University of Southern California, Graduate Student Symposium
Jalalnezhad MJ, Kamali V (2016) Development of an intelligent model for wax deposition in oil pipeline. J Pet Explor Prod Technol 6(1):129–133
Jang J-SR, Sun C-T, Mizutani E (1997) Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence [Book Review]. IEEE Trans 42(10):1482–1484
Kaewkamnerdpong B, Bentley PJ (2005) Perceptive particle swarm optimisation: an investigation. In: Proceedings 2005 IEEE swarm intelligence symposium, 2005 (SIS 2005). IEEE
Kamrani E (2010) Modeling and forecasting long-term natural gas (NG) consumption in Iran, using particle swarm optimization (PSO) . Master thesis, Department of Computer Engineering of Dalarna University
Kar S, Das S, Ghosh PK (2014) Applications of neuro fuzzy systems: a brief review and future outline. Appl Soft Comput 15:243–259
Kathrada M (2009) Uncertainty evaluation of reservoir simulation models using particle swarms and hierarchical clustering. Heriot-Watt University, Edinburgh
Khademi M, Rahimpour M, Jahanmiri A (2010) Differential evolution (DE) strategy for optimization of hydrogen production, cyclohexane dehydrogenation and methanol synthesis in a hydrogen-permselective membrane thermally coupled reactor. Int J Hydrog Energy 35(5):1936–1950
Kim Y, Jang H, Kim J et al (2017) Prediction of storage efficiency on CO_2 sequestration in deep saline aquifers using artificial neural network. Appl Energy 185:916–928
Klee V, Minty GJ (1970) How good is the simplex algorithm, DTIC Document
Land AH, Doig AG (2010) An automatic method for solving discrete programming problems. 50 Years of integer programming 1958–2008. Springer, Berlin, pp 105–132
Lin C, Qing A, Feng Q (2011) A comparative study of crossover in differential evolution. J Heuristics 17(6):675–703
Liu Y, Passino KM (2000) Swarm intelligence: literature overview. Department of Electrical Engineering, the Ohio State University
Ma X, Gildin E, Plaksina T (2015) Efficient optimization framework for integrated placement of horizontal wells and hydraulic fracture stages in unconventional gas reservoirs. J Unconv Oil Gas Resour 9:1–17
Mardani A, Jusoh A, Zavadskas EK (2015) Fuzzy multiple criteria decision-making techniques and applications-two decades review from 1994 to 2014. Expert Syst Appl 42(8):4126–4148
Maria G (1998) IDENTIFICATION/DIAGNOSIS-adaptive random search and short-cut techniques for process model identification and monitoring. AIChE Symposium Series. American Institute of Chemical Engineers, New York, NY, 1971-c2002
Mariajayaprakash A, Senthilvelan T, Gnanadass R (2015) Optimization of process parameters through fuzzy logic and genetic algorithm—a case study in a process industry. Appl Soft Comput 30:94–103
Mirzabozorg A (2015) Incorporation of engineering knowledge in history matching, optimization, and uncertainty assessment frameworks with application to the SAGD process. University of Calgary, Calgary
Mohaghegh S (2000) Virtual-intelligence applications in petroleum engineering: part 1—artificial neural networks. J Pet Technol 52(09):64–73
Mohagheghian E (2016) An application of evolutionary algorithms for WAG optimisation in the Norne Field, Memorial University of Newfoundland
Mohamed L, Christie MA, Demyanov V (2011) History matching and uncertainty quantification: multiobjective particle swarm optimisation approach. Society of Petroleum Engineers, SPE EUROPEC/EAGE annual conference and exhibition
Mohammadi M, Kharrat R, Hashemi A (2015) Developing a fuzzy logic model to predict asphaltene precipitation during natural depletion based on experimental data. Iran J Oil Gas Sci Technol 4(2):40–49
Mohanty S (2005) Estimation of vapour liquid equilibria of binary systems, carbon dioxide-ethyl caproate, ethyl caprylate and ethyl caprate using artificial neural networks. Fluid Phase Equilib 235(1):92–98
Novák V, Perfilieva I, Mockor J (2012) Mathematical principles of fuzzy logic. Springer, Berlin
Olatunji SO, Selamat A, Azeez ARA (2015) Modeling permeability and PVT properties of oil and gas reservoir using hybrid model based on type-2 fuzzy logic systems. Neurocomputing 157:125–142

Onwunalu JE, Durlofsky LJ (2010) Application of a particle swarm optimization algorithm for determining optimum well location and type. Comput Geosci 14(1):183–198

Ouenes A (2000) Practical application of fuzzy logic and neural networks to fractured reservoir characterization. Comput Geosci 26(8):953–962

Price K, Storn RM, Lampinen JA (2006) Differential evolution: a practical approach to global optimization. Springer, Berlin

Ramadhas A, Jayaraj S, Muraleedharan C et al (2006) Artificial neural networks used for the prediction of the cetane number of biodiesel. Renew Energy 31(15):2524–2533

Rammay MH, Abdulraheem A (2014) Automated history matching using combination of adaptive neuro fuzzy system (ANFIS) and differential evolution algorithm. SPE Large Scale Computing and Big Data Challenges in Reservoir Simulation Conference and Exhibition, Society of Petroleum Engineers

Ravandi EG, Nezamabadi-Pour H, Monfared AF et al (2014) Reservoir characterization by a combination of fuzzy logic and genetic algorithm. Pet Sci Technol 32(7):840–847

Rumelhart DE, McClelland JL, Group PR (1986) Parallel distributed processing: explorations in the microstructure of cognition: foundations, vol. 1

Saemi M, Ahmadi H, Varjani A (2007) Design of neural networks using genetic algorithm for the permeability estimation of the reservoir. J Pet Sci Eng 59(1):97–105

Salmachi A, Sayyafzadeh M, Haghighi M (2013) Infill well placement optimization in coal bed methane reservoirs using genetic algorithm. Fuel 111:248–258

Santhosh EC, Sangwai JS (2016) A hybrid differential evolution algorithm approach towards assisted history matching and uncertainty quantification for reservoir models. J Pet Sci Eng 142:21–35

Senthilkumar S (2014) Practical applications of swarm intelligence and evolutionary computation, hybrid soft computing. In: International journal of swarm intelligence and evolutionary computation 2014

Shen Q, Shi W-M, Yang X-P et al (2006) Particle swarm algorithm trained neural network for QSAR studies of inhibitors of platelet-derived growth factor receptor phosphorylation. Eur J Pharm Sci 28(5):369–376

Shi Y, Eberhart RC (1999) Empirical study of particle swarm optimization. In: Proceedings of the 1999 congress on evolutionary computation, 1999 (CEC 99). IEEE

Storn R, Price K (1995) Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Science Institute, Berkeley

Trelea IC (2003) The particle swarm optimization algorithm: convergence analysis and parameter selection. Inf Process Lett 85(6):317–325

Üçoluk G (2002) Genetic algorithm solution of the TSP avoiding special crossover and mutation. Intell Autom Soft Comput 8(3):265–272

Vazquez O, Fursov I, Mackay E (2016) Automatic optimization of oilfield scale inhibitor squeeze treatment designs. J Pet Sci Eng 147:302–307

Velez-Langs O (2005) Genetic algorithms in oil industry: an overview. J Petrol Sci Eng 47(1):15–22

Wang P (2003) Development and applications of production optimization techniques for petroleum fields. Stanford University, Stanford

Wang J, Buckley JS (2006) Automatic history matching using differential evolution algorithm. International Symposium of the Society of Core Analysis, Trondheim

Wang X, Qiu X (2013) Application of particle swarm optimization for enhanced cyclic steam stimulation in a offshore heavy oil reservoir. arXiv preprint arXiv:1306.4092

Wang C, Gao J, Yang H et al (2011) Waveform inversion of cross-well data with cooperative coevolutionary differential evolution algorithm. SEG technical program expanded abstracts 2011, society of exploration geophysicists, pp 2752–2756

Watson A, Seinfeld J, Gavalas G et al (1980) History matching in two-phase petroleum reservoirs. Soc Pet Eng J 20(06):521–532

Wu W (2015) Oil and gas pipeline risk assessment model by fuzzy inference systems and artificial neural network. University of Regina, Faculty of Graduate Studies and Research

Wu Y-C, Lee W-P, Chien C-W (2011) Modified the performance of differential evolution algorithm with dual evolution strategy. In: International conference on machine learning and computing

Xu S, Zhang M, Zeng F et al (2015) Application of genetic algorithm (GA) in history matching of the vapour extraction (VAPEX) heavy oil recovery process. Nat Resour Res 24(2):221–237

Xue Y, Cheng L, Mou J et al (2014) A new fracture prediction method by combining genetic algorithm with neural network in low-permeability reservoirs. J Pet Sci Eng 121:159–166

Yang E (2009) Selection of target wells and layers for fracturing with fuzzy mathematics method. In: Sixth international conference on fuzzy systems and knowledge discovery, 2009 (FSKD'09). IEEE

Yetilmezsoy K, Fingas M, Fieldhouse B (2011) An adaptive neuro-fuzzy approach for modeling of water-in-oil emulsion formation. Colloids Surf Physicochem Eng Asp 389(1):50–62

Yilmaz S, Demircioglu C, Akin S (2002) Application of artificial neural networks to optimum bit selection. Comput Geosci 28(2):261–269
Yin D, Wu T (2009) Notice of retraction optimizing well for fracturing by fuzzy analysis method of applying computer. In: 2009 First international conference on information science and engineering. IEEE
Zadeh LA (1965) Fuzzy sets. Inf Control 8(3):338–353
Zendehboudi S, Ahmadi MA, James L et al (2012) Prediction of condensate-to-gas ratio for retrograde gas condensate reservoirs using artificial neural network with particle swarm optimization. Energy Fuels 26(6):3432–3447
Zendehboudi S, Rajabzadeh AR, Bahadori A et al (2014) Connectionist model to estimate performance of steam-assisted gravity drainage in fractured and unfractured petroleum reservoirs: enhanced oil recovery implications. Ind Eng Chem Res 53(4):1645–1662
Zhang D, Gong X, Peng L (2009) Estimating geostatistics variogram parameters based on hybrid orthogonal differential evolution algorithm. In: International symposium on intelligence computation and applications. Springer, Berlin
Zhang J, Zhang H, Yu J et al (2014) Fast one-dimensional velocity model determination using station-pair differential times based on the differential evolution method in microseismic monitoring. SEG Technical Program Expanded Abstracts 2014, Society of Exploration Geophysicists, pp 4832–4836
Zhang Y, Chen M, Jin Y et al (2016) Experimental study and artificial neural network simulation of the wettability of tight gas sandstone formation. J Nat Gas Sci Eng 34:387–400
Zhou Q, Wu W, Liu D et al (2016) Estimation of corrosion failure likelihood of oil and gas pipeline based on fuzzy logic approach. Eng Fail Anal 70:48–55
Zoveidavianpoor M, Samsuri A, Shadizadeh SR (2012) Fuzzy logic in candidate-well selection for hydraulic fracturing in oil and gas wells: a critical review. Int J Phys Sci 7(26):4049–4060