CrossMark

Nature-Inspired Chemical Reaction Optimisation Algorithms

Nazmul Siddique[1] · **Hojjat Adeli**[2]

Abstract Nature-inspired meta-heuristic algorithms have dominated the scientific literature in the areas of machine learning and cognitive computing paradigm in the last three decades. Chemical reaction optimisation (CRO) is a population-based meta-heuristic algorithm based on the principles of chemical reaction. A chemical reaction is seen as a process of transforming the reactants (or molecules) through a sequence of reactions into products. This process of transformation is implemented in the CRO algorithm to solve optimisation problems. This article starts with an overview of the chemical reactions and how it is applied to the optimisation problem. A review of CRO and its variants is presented in the paper. Guidelines from the literature on the effective choice of CRO parameters for solution of optimisation problems are summarised.

Keywords Nature-inspired computing · Biologically inspired algorithm · Physics inspired algorithms · Chemical reaction optimisation

Introduction

Machine learning is one of the most important platforms for cognitive computation paradigm. The main constituents of

✉ Nazmul Siddique
nh.siddique@ulster.ac.uk

Hojjat Adeli
adeli.1@osu.edu

1 School of Computing and Intelligent Systems, University of Ulster, Northland Road, Londonderry BT48 7JL, UK

2 College of Engineering, The Ohio State University, 470 Hitchcock Hall, 2070 Neil Avenue, Columbus, OH 43210, USA

machine learning are the meta-heuristic algorithms. Nature has always been an inspiration and source for scientific invention. Scientists have been striving for understanding the laws of nature and developing methods and computer algorithms for real-life problems. The development of three mainstream disciplines of the natural sciences, biology, physics, and chemistry have created novel problem solving paradigms [1, 2]. Biology-inspired algorithms have been in use since 1960. They include genetic algorithms [3], genetic programming [4], evolutionary computation [5], and particle swarm optimisation [6]. Physics inspired algorithms have been the subject of significant research in the last three decades [7]. They include simulated annealing [8, 9], gravitational search algorithm [10], harmony search algorithm [11], central force optimization [12], water drop algorithm [13], and spiral dynamics algorithm [14]. The chemical reaction metaphor can also be exploited for developing meta-heuristic algorithms by encoding appropriate information into molecule-like elements and performing a set of chemical reaction-like operations onto them to obtain certain kind of derivative information suitable for optimisation problems. Chemical reaction optimisation (CRO) algorithm is a recent search and optimisation algorithm inspired by chemistry, which is equally promising like biology and physics inspired algorithms.

Chemistry is the field of science that studies the chemical properties of matter and its structure. Chemical reactions break chemical bonds into molecules and form new bonds using molecules participating in reaction [15]. Energy is required for breaking chemical bonds into several molecules. Also energy is released when new bonds are formed combining several molecules. Thus, a chemical reaction is seen as a process of transformation of a set of molecules participating in a chemical reaction into a set of products with different properties. There are two types of chemical reactions, namely, unimolecular and multi-molecular elementary reactions

absorbing or releasing different level of energies during the reactions [16]. Chemical reactions uphold the laws of thermodynamics. Every chemical reaction stabilises at equilibrium, a state determined by the minimum free energy, also called Gibbs free energy [17]. The Gibbs free energy is a chemical potential that depends on the temperature, pressure, and materials involved and tends to reach its minimum at equilibrium.

A molecule consists of several atoms. The type of atoms, bond length, angle, and torsion (twisting of the structure) define the distinct structure of the molecule. Two molecules can be different even with the same set of atoms due to the difference in their structures. In other words, the structure represents the relationship between atoms in a molecule. In order to undergo a chemical reaction, molecules must acquire the necessary energy to be activated. Chemical bonds are source of energy. The energy of a molecule can be of two types: potential energy and kinetic energy. Potential energy is denoted by PE. PE is the energy that a molecule contains in the structure. In a chemical reaction (e.g. exothermic reaction), chemical bonds break and new bond are formed. That is, during the reaction molecules transform from structure of higher PE into structure of lower PE with release of energy. Kinetic energy is denoted by KE. Molecules need to collide for the chemical reaction to happen. Collision between molecules provides the KE needed to break the bonds. Sometimes there is not enough KE for collisions to happen. In such a case, energy is provided in the form of heat. Heat rises the temperature which is a measure of the average KE. Heating causes KE to increase to the required level for breaking the bonds.

If $x = \{x^1, x^2, \cdots, x^n\}$ is a molecule with x^k, $k = 1, \cdots, n$ atoms of a certain structure, then the molecule tends to change from x to a new molecule x' by changing its structure during a chemical reaction only if the following energy condition is satisfied.

$$PE(x) \geq PE\left(x'\right) \tag{1}$$

If energy Condition (1) is not satisfied, the molecules need higher energy for the reactions to happen. KE is added to achieve the required higher energy level for the reactions to occur and change to new molecules. This must satisfy the following energy condition.

$$PE(x) + KE(x) \geq PE\left(x'\right) \tag{2}$$

A molecule with higher KE has a higher possibility of transforming into a new structure with higher PE. A reactant (generally a molecule or a compound) with high energy is unstable and tends to have a reaction when it comes in contact with another molecule and goes through a sequence of elementary reaction phases dissipating energy. The process is called reaction mechanism. A chemical reaction may take more than one reaction path ensuring the maximum amounts of desired products and the minimum amounts of undesired products. The actual course of

any reaction is determined by the least energy requirement leading to an optimal reaction mechanism. At the final stage of reaction, products produced will have low-energy level and stable state. The final state is considered an optimal and stable state of the chemical reaction maintaining an optimal reaction mechanism.

Lam and Li [18] proposed the Chemical Reaction Optimization (CRO) algorithm inspired by chemical reactions. This paper presents the CRO algorithm, different variants of CRO, hybrids of CRO with other meta-heuristic methods and their applications in different domains. The rest of the paper is organised as follows: Section "Chemical Reaction Operations in CRO" presents the background on the chemical reaction operations used in CRO. Section "Implementation of CRO Operation" describes the implementation of the CRO operators. Section "Chemical Reaction Optimisation Algorithm" presents the CRO algorithm. Variants and hybrid CRO algorithms are presented in Section "Variants of CRO". Section "Conclusions" concludes the paper with some comments on the future directions of CRO algorithm.

Chemical Reaction Operations in CRO

Molecules with different levels of energy take part in chemical reactions, undergo a sequence of elementary reactions and are transformed into products with minimum energy. The elementary reactions are the operators of CRO. These reactions are grouped into uni-molecular and multi-molecular reactions, classified into four types:

Uni-molecular reactions

(i)   On-wall ineffective collision
(ii) Decomposition
Multi-molecular reactions

(iii) Inter-molecular ineffective collision
(iv) Synthesis

On-Wall Ineffective Collision Operation Only one molecule is involved in this type of operation. The molecule does not take part in chemical reaction with another molecule. A molecule x hits on the wall and bounces back resulting in a change in PE and KE. The new value is denoted as x'. It is, therefore, called on-wall ineffective collision. The on-wall ineffective collision of a molecule is illustrated in Fig. 1. The change of the molecule is described by

$$x' = x + \Delta \tag{3}$$

where Δ is a perturbation of the molecular structure caused by the collision. The perturbation Δ can be modelled as

probability distribution over a finite interval, e.g. Gaussian, Cauchy, lognormal, and exponential distribution. Due to the change in molecule, $PE(x)$ changes to $PE(x')$ and $KE(x)$ changes to $KE(x')$, which must satisfy the energy Eq. (2). Otherwise, a change in the molecule x will not happen without release (or loss) of some amount of kinetic energy $KE(x)$. According to the law of conservation of energy, energy cannot be destroyed. Therefore, the released energy $KE(x')$ is stored in an energy buffer called central buffer after the reaction. In CRO, the released energy $KE(x')$ is modelled using a random number $\rho_1 \in [KE_{LossRate}, 1]$, where $KE_{LossRate}$ is a parameter of the chemical reaction and represents the maximum percentage of KE lost in the environment at a time. $(1 - \rho_1)$ is the fraction of KE that is lost in the environment when the molecule hits the wall. This lost energy is stored in the central buffer. $KE(x')$ and energy buffer update are described by

$$KE\left(x'\right) = \left[PE(x) - PE\left(x'\right) + KE(x)\right] \times \rho_1 \quad (4)$$

$$buffer = buffer + \left[PE(x) - PE\left(x'\right) + KE(x)\right] \times (1 - \rho_1) \quad (5)$$

Decomposition Operation One molecule is involved in this type of operation. The molecule does not take part in chemical reaction with another molecule. A molecule x hits the wall and decomposes into two molecules x_1 and x_2. The molecule can also decompose into more than two molecules. The decomposition of molecule x into molecules x_1 and x_2 is illustrated in Fig. 2. Due to the change in the molecule structure, $PE(x)$ changes to $PE(x_1)$ and $PE(x_2)$ and $KE(x)$ changes to $KE(x_1)$ and $KE(x_2)$ which must satisfy the energy conditions described by

$$PE(x) + KE(x) \geq PE\left(x_1'\right) + PE\left(x_2'\right) \quad (6)$$

$$KE\left(x_1'\right) = \left[PE(x) + KE(x) - PE\left(x_1'\right) - PE\left(x_2'\right)\right] \times \rho_1 \quad (7)$$
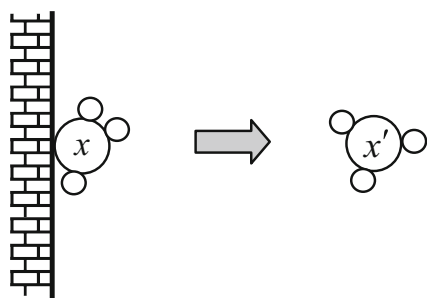

Fig. 1 Illustration of on-wall ineffective collision operator. The structure of molecule is shown by the *big circles* with attached *small circles* around representing the change in the structure. A molecule hits on the wall and bounces back with perturbed structure. The perturbed structure is shown with changed positions of the *small circles*
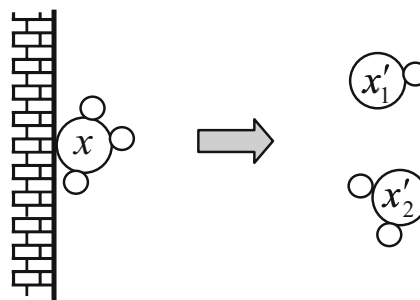

Fig. 2 Illustration of decomposition operation. A molecule hits on the wall and decomposes into two molecules with change in structures

$$KE\left(x_2'\right) = \left[PE(x) + KE(x) - PE\left(x_1'\right) - PE\left(x_2'\right)\right] \times (1 - \rho_1) \quad (8)$$

where $\rho_1 \in [0, 1]$ is a random number representing the released energy.

Sometimes molecule x does not have enough energy meaning that Condition (6) does not hold for the reaction to happen and decompose into x_1 and x_2. It can only happen when energy $KE(x)$ is large enough. According to the law of conservation of energy, energy cannot be created. The extra energy comes from the central energy buffer to decompose the molecule. The process is described by the following modified conditions:

$$PE(x) + KE(x) + buffer \geq PE\left(x_1'\right) + PE\left(x_2'\right) \quad (9)$$

$$KE\left(x_1'\right) = \left[\left\{PE(x) + KE(x) - PE\left(x_1'\right) - PE\left(x_2'\right)\right\} + buffer\right] \times (\rho_1 \times \rho_2) \quad (10)$$

$$KE\left(x_2'\right) = \left[\left\{PE(x) + KE(x) - PE\left(x_1'\right) - PE\left(x_2'\right)\right\} + buffer\right] \times (\rho_3 \times \rho_4) \quad (11)$$

where $\{\rho_1, \rho_2, \rho_3, \rho_4\} \in [0, 1]$ are random numbers. To ensure a small amount of energy for $KE(x_1')$ and $KE(x_2')$ from the buffer, multiplication of two random numbers $\{\rho_1 \times \rho_2\}$ and $\{\rho_3 \times \rho_4\}$ are used. The energy buffer update is described by

$$buffer = buffer + \left[PE(x) + KE(x) - PE\left(x_1'\right) - PE\left(x_2'\right)\right] - KE\left(x_1'\right) - KE\left(x_2'\right) \quad (12)$$

If Conditions in (6) and (9) do not hold, the decomposition will not take place.

Inter-Molecular Ineffective Collision Operation Two molecules are involved in this type of operation. A molecule x_1 collides with another molecule x_2 and the two molecules x_1 and x_2 are perturbed and change to x_1' and x_2'. Thus, the

collision causes change in potential energies {PE(x_1), PE(x_2)} and kinetic energies {KE(x_1), KE(x_2)}. The inter-molecular collision operation is illustrated in Fig. 3. The potential energies {PE(x_1), PE(x_2)} change to {PE(x_1'), PE(x_2') } and the kinetic energies {KE(x_1),KE(x_2)} change to {KE(x_1'), KE(x_2') }, respectively. The inter-molecular ineffective collision must hold the energy conditions:

$$\text{PE}(x_1) + \text{PE}(x_2) + \text{KE}(x_1) + \text{KE}(x_2) \geq \text{PE}\left(x_1'\right)$$
$$+ \text{PE}\left(x_2'\right) \tag{13}$$

$$\text{KE}\left(x_1'\right)$$
$$= \left[\text{PE}(x_1) + \text{PE}(x_2) + \text{KE}(x_1) + \text{KE}(x_2) - \text{PE}\left(x_1'\right) - \text{PE}\left(x_2'\right)\right] \times \rho_1 \tag{14}$$

$$\text{KE}\left(x_2'\right)$$
$$= \left[\text{PE}(x_1) + \text{PE}(x_2) + \text{KE}(x_1) + \text{KE}(x_2) - \text{PE}\left(x_1'\right) - \text{PE}\left(x_2'\right)\right] \times (1-\rho_1) \tag{15}$$

If Condition (13) does not hold, the reaction will not take place.

Synthesis Operation Two or more molecules are involved in this type of operation. Two molecules x_1 and x_2 collide together and fuse into a molecule x'. Fusion releases a large amount of energy. Therefore, a large energy change occurs during the synthesis operation. The synthesis operation is illustrated in Fig. 4. The potential energies PE(x_1) and PE(x_2) change to PE(x') and the kinetic energies KE(x_1) and KE(x_2) change to KE(x'). The energy balance of synthesis operation must hold the energy conditions:

$$\text{PE}(x_1) + \text{PE}(x_2) + \text{KE}(x_1) + \text{KE}(x_2) \geq \text{PE}\left(x'\right) \tag{16}$$

$$\text{KE}\left(x'\right) = \left[\text{PE}(x_1) + \text{PE}(x_2) + \text{KE}(x_1) + \text{KE}(x_2) - \text{PE}\left(x'\right)\right] \tag{17}$$

If Condition (16) does not hold, the synthesis reaction will not take place.

In a chemical reaction process, a population of reactants with high-energy level undergoes a sequence of elementary chemical reactions, transforms through different energy levels and produces certain products with new molecular structures of low energy and stable states at the final stage. Every chemical reaction seeks to achieve equilibrium after which no further reactions take place. The chemical reaction process
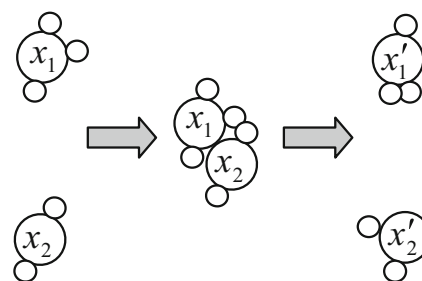

Fig. 3 Illustration of inter-molecular ineffective collision operation. Two molecules x_1 and x_2 collide against each other and change to new molecules x_1' and x_2^1

continues until it satisfies the aforementioned energy equations. The process of the chemical reaction is seen as an optimisation process where the set of parameters to be optimised are reactants that take part in the transformation process.

Implementation of CRO Operation

CRO is a population-based optimisation algorithm. Chemical reaction operations discussed in earlier section are the inspiration behind the operators of CRO. Four types of operators are applied to the population of solutions. The total number of solutions kept by the algorithm may change from time to time as the decomposition and synthesis operators increase and decrease the number of molecules in the reaction pool, respectively. The computational implementation of these four operators is discussed in this section.

Ineffective Collision Operator

Some small change occurs in the molecular attributes during this operation and the molecule x obtains a new structure x' in the neighbourhood of x which is expressed as:

$$x' = N(x) \tag{18}$$

where $N(\cdot)$ is the neighbourhood operator.

Yu et al. [19] used a neighbourhood operator $N(x)$ to generate a new solution x' by perturbing one element of x chosen randomly. The perturbation is done by adding a Gaussian perturbation $\rho(m, \sigma)$ to the randomly chosen element m, where m is the mean and σ is the variance. σ is a parameter and chosen arbitrarily. The neighbourhood operator is implemented using the following pseudo-code.

$$i = \text{random integer } (1, n)$$
Find the i-th element m in x, i.e. $m = x(i)$
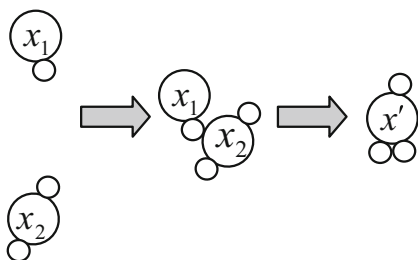Calculate $m = m + \rho(m, \sigma)$

Fig. 4 Illustration of synthesis operator. Two molecules x_1 and x_2 collide together and fuse into one molecule x'

Decomposition Operator

Decomposition operator breaks a molecule into two or more molecules. Firstly, the solution x is copied onto x_1 and x_2. Then half of the variables (i.e. $n/2$ variables where n is the total number of variables) of the solution x are perturbed by adding random variations and creating new solutions. The following pseudo-code is used to implement the decomposition operator.

> Copy $x(i)$ to $x_1(i)$ and $x_2(i)$
> For $i = 1\, to\, n$ do
>   If ($i \leq \frac{n}{2}$)
> $x_1(i) = x(i) + rand[0,1]$
>   Else
> $x_2(i) = x(i) + rand[0,1]$
> Endfor

Inter-Molecular Ineffective Collision Operator

Two molecules x_1 and x_2 collide with each other and change to new molecules x_1' and x_2' in the neighbourhood of x_1 and x_2. x_1' and x_2' are expressed as.

$$\begin{cases} x_1' = N(x_1) \\ x_2' = N(x_2) \end{cases} \tag{19}$$

Implementation of the neighbourhood operator is presented in Section "Ineffective collision operator".

Synthesis Operator

Synthesis operator combines multiple molecules into one. Two solutions x_1 and x_2 are combined by applying a 'probabilistic select' to implement synthesis into a new solution x'. Each component of x' in the same position is chosen either

from x_1 or x_2 randomly. The following pseudo-code is used to implement the synthesis operator.

> For $i = 1\, to\, n$ do
> $r = rand[0,1]$
>   If ($r > 0.5$)
> $x'(i) = x_1(i)$
>   Else
> $x'(i) = x_2(i)$
>   Endif
> Endfor

The effect of synthesis operator is similar to the recombination (or crossover) operation used by other evolutionary algorithms (EA)[1] [21, 22].

In CRO, decomposition and synthesis are exploration mechanisms providing the effect of global search while ineffective collision and inter-molecular ineffective collision are exploitation mechanisms providing the effect of local search.

Chemical Reaction Optimisation Algorithm

In terms of optimisation, a molecule with its atoms in a particular structure can be thought of a solution of a problem. If the feasible solution of a problem can be defined as a set of positive real numbers R^{n+} in n-dimensional space, then any vector $x_i \in R^{n+}$ with $x_i = \{x_i^1, x_i^2, \cdots, x_i^n\}$, $i = 1, \cdots, N$, is a valid molecule representing a valid solution of the problem and x_i^k, $k = 1, \cdots, n$, are thought of atoms representing the decision variables. The representation of molecule $x \in R^{n+}$ can be in the form of numbers, an array similar to chromosomes in genetic algorithms [23, 24], a matrix, or a graph similar to tree structure in genetic programming [25, 26]. A change in a molecule $x \in R^{n+}$ with higher energy state can only occur by means of chemical reaction. The chemical reaction changes the energy state of the molecule and results in a new molecule $x' \in R^{n+}$ with a lower energy state. Therefore, minimisation of energy is considered as the objective in CRO and hence PE is defined as the objective function when evaluating a solution. PE is the energy responsible for a stable structure whereas KE is the energy needed for the movement of molecules and collision between molecules such that the reaction can happen. KE of the molecule helps it escape from the local minimum if the solution is stuck at a local minimum. If x represents the

---

[1] Recombination or crossover operation in EA generally selects two solutions randomly from the population and combines them to produce one or two offspring by choosing individual elements from each of them. A detailed account of discussion on EA crossover operations can be found in [20].

molecule with a certain structure (i.e. configuration of atoms), then an objective function, $f(\cdot)$, is defined as equal to PE:

$$PE(x) = f(x) \qquad (20)$$

Lam and Li [18] presented the CRO meta-heuristic algorithm inspired by chemical reactions. In CRO, decision variables are like atoms that form a molecule and a molecule is a representation of solution of a problem. A population of molecules is generated randomly within the search space. The molecules undergo chemical reaction-like transformation. Four types of chemical reaction operations are used: on-wall ineffective collision, decomposition, inter-molecular ineffective collision, and synthesis as discussed earlier. The parameter MoleColl decides on the fraction of all elementary reactions that involve more than one molecule, i.e. inter-molecular reactions in CRO. The process continues until a minimum of energy is reached as defined by the objective function, Eq. (20). This objective function is problem-dependent. The CRO algorithm is illustrated by the flow diagram in Fig. 5.

The operators of the CRO algorithm are the decomposition and synthesis operations as mechanisms for generating new solutions for exploring the search space, and the on-wall ineffective and inter-molecular ineffective collisions operations as the mechanisms for generating solutions from the neighbourhood structure for exploiting the search space. The decomposition and synthesis operators act as diversification and ineffective and inter-molecular ineffective collisions operators act as intensification for the algorithm [27].

There are mainly four parameters in CRO. These are population size, KE loss rate, fraction of uni-molecular reaction, and initial KE. Population size, denoted as PopSize, is the initial number of solutions generated randomly in the solution space. $KE_{LossRate}$ is the loss rate of KE during reaction, i.e. it is the upper limit of percentage of KE lost to the environment during on-wall ineffective collisions. Fraction of uni-molecular reaction, denoted as MoleColl, is the fraction of molecules that undergo uni-molecular or inter-molecular reactions. If MoleColl is less than a random number, ρ, it will result in a uni-molecular collision. Otherwise, an inter-molecular collision will take place. It is to be noted that a uni-molecular collision will always take place when there remains only one molecule in the population. Initial KE, denoted as KE_0, is the initial value assigned to each element of KE in the initialization stage.

Variants of CRO

CRO algorithm is a recent addition to the meta-heuristic algorithm family. Researchers made simple

modifications while applying the algorithm to different application domains which later became known as variants of CRO algorithms such as real-coded CRO, opposition-based CRO, and orthogonal CRO. Some researchers attempted to improve the operators borrowing ideas from other meta-heuristic algorithms such as differential evolution (DE) [28, 29] and particle swarm optimisation (PSO) [30, 31] which led to hybrid CRO algorithms. These variants are discussed briefly in the sequel.

Real-Coded CRO (RCCRO)

The original version of CRO [18, 24] is designed for discrete optimisation problems. An extension of CRO algorithm is proposed for continuous problems by Lam et al. [32], which has become known as real-coded CRO. Three modifications are introduced: solution representation, neighbourhood operator, and boundary constraint handling.

A solution of an optimisation problem in standard CRO [18, 24] is represented by a molecular structure $x = \{x_1, x_2, \cdots, x_n\}$ where each individual x_i, $i = 1, 2, \cdots, n$ can be of binary or integer type. In real-coded CRO, each x_i is implemented using a floating-point number. To deal with continuity, continuous search ability is incorporated into the neighbourhood search operator $N(\cdot)$. If the problem does not impose any constraints on relations between solution variables, x_i can be treated independently and a perturbation is defined as

$$x_i' = N(x_i) = x_i + \delta_i \qquad (21)$$

where δ_i is a probabilistic perturbation in the i-th element, e.g. Gaussian, Cauchy, Lévy etc. For example, it may defined by Gaussian distribution of the form $\delta_i = N(\mu, \sigma^2)$ with mean μ and variance σ^2.

The perturbation depends mainly on the starting point for the solution x_i, direction from the mean μ and step-size based on the spread σ. In general σ is fixed in real-coded CRO during the execution of the algorithm. Too large or too small value for σ will make the algorithm inefficient. Lam et al. [32], therefore, proposed an adaptive scheme for σ where the initial value of σ is set equal to solution space $(u - l)$ (upper bound minus lower bound values) and decreased gradually by a factor θ.

The perturbation of a solution x_i may go out of bound defined by the lower (l_i) and upper (u_i) bounds. The boundary constraints can be handled by bringing the solution x_i back within boundary [32]. A simple technique such as reflecting

Fig. 5 Flow diagram of CRO
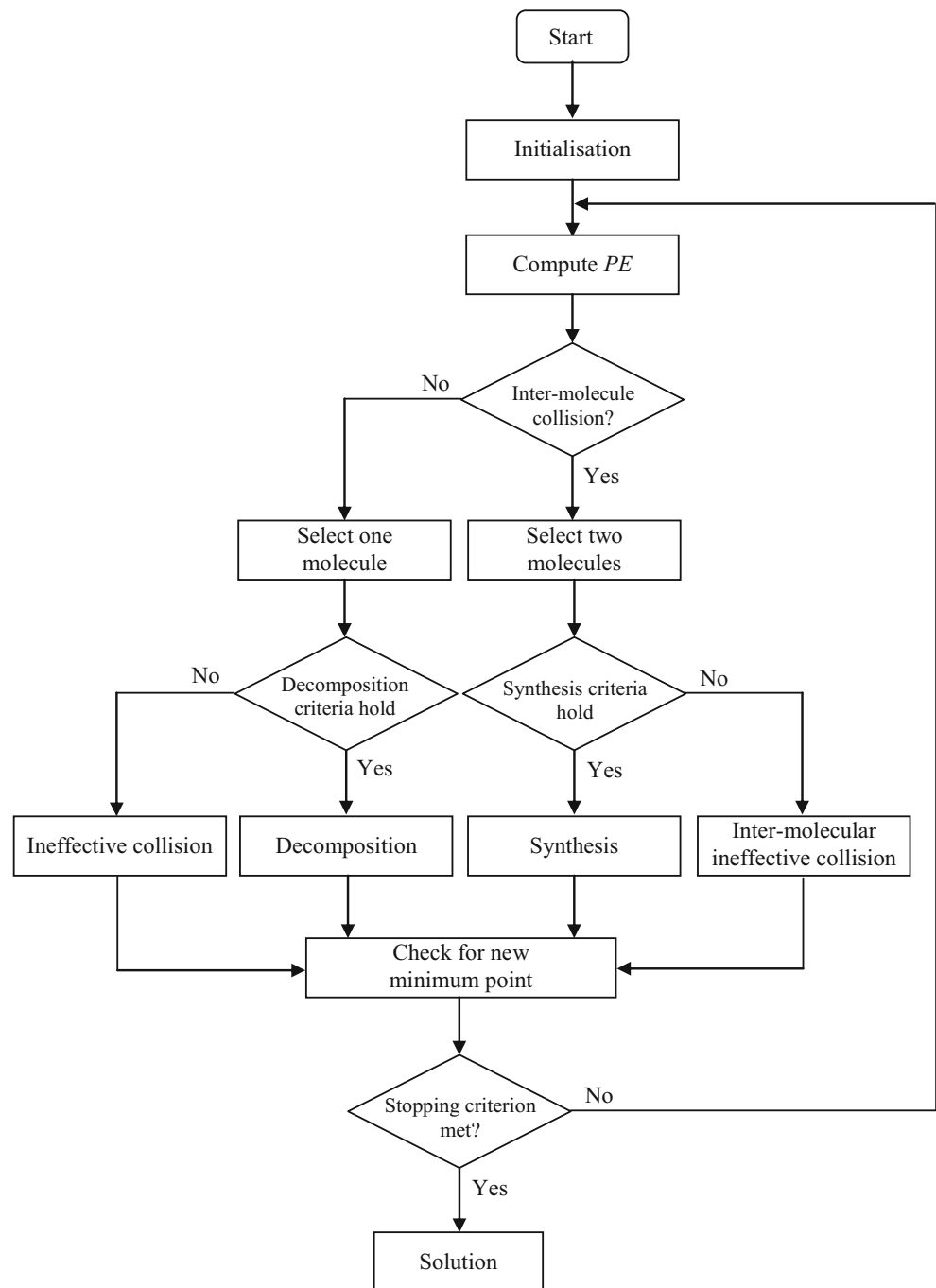
scheme, where x_i reflects back by the same amount from the boundary, can be used. $x_i^{'}$ is then defined by

$$x_i^{'} = \begin{cases} 2 \times l_i - x_i & if\, x_i < l_i \\ 2 \times u_i - x_i & if\, x_i < l_i \end{cases} \tag{22}$$

The effectiveness and performance of the real-coded CRO algorithm has been verified on a number of unimodal, high-dimensional multi-modal, and low-dimensional multi-modal benchmark functions.

Bhattacharjee et al. [33] applied real-coded CRO to minimisation of total power generation cost by scheduling different power plants for certain intervals of time.

Opposition-Based CRO

Opposition-based approach in learning was proposed by Tizhoosh [34] with the assumption that if a set of randomly generated numbers can not satisfy a criterion, then the opposite set of numbers may a have higher possibly to satisfy

the criterion. It is found that opposition-based approach improves computational efficiency. Many researchers have used opposition-based approach in optimisation problems [35–37] where they used the current population and its opposite population. Bhattacharjee et al. [38] used opposite and quasi-opposite numbers in one-dimensional space in the CRO algorithm. An opposite number $\breve{x}$ of any real number $x \in [a, b]$ is defined by

$$\breve{x} = a + b - x \tag{23}$$

Bhattacharjee et al. [38] defined the quasi-opposite number $\breve{x}_q$ as

$$\breve{x}_q = \text{rand}\left(c, \breve{x}\right) \tag{24}$$

where c is the centre of the interval $[a, b]$. c can be estimated as the mean of the interval $[a, b]$, i.e. $c = (a + b)/2$. Similarly, reflected quasi-opposite number $\breve{x}_{qr}$ is defined by

$$\breve{x}_{qr} = rand(c, x) \tag{25}$$

Bhattacharjee et al. [38] extended one-dimensional representation of x to two-dimensional representation. A population of molecular set is generated and then a quasi-opposite molecular matrix, denoted as QOM, is formed from the molecular set using a parameter $J_r \in [0, 1]$ called jumping rate as follows:
If $rand < J_r$
$QOM(i, j) = rand(c, \breve{x})$ with $i = 1, \cdots, PopSize$ and $j = 1, \cdots, n$
where $PopSize$ is the population size and n is the number of variables in the optimisation problem.

The performance of the opposition-based real-coded CRO has been verified on short-term hydrothermal scheduling problem. The total minimum, maximum, and average system costs are obtained within 25 trials demonstrating the algorithm has good exploration and exploitation ability [38]. It is found that opposition-based approach helps improve convergence speed in optimisation algorithms.

Orthogonal CRO

CRO algorithm traverses through search space in a random manner [39] which eventually limits the search scope and slows down the convergence speed. A meta-heuristic algorithm needs to explore promising regions. The exploration becomes difficult when an optimisation problem has a large number of decision variables to be optimised within a limited number of iterations. The orthogonal experimental design is an approach to find the best combination of different factors within a small number of trials. Such orthogonal array [40] has been applied in simulated annealing [41, 42], PSO [43], and genetic algorithm [44] with better results. Li et al. [45] proposed an orthogonal CRO algorithm by introducing a quantisation orthogonal crossover (QOX) operator, where the decision variables are quantised into different levels and the variables are divided into groups. The groups are treated as factors in orthogonal CRO. Then, an individual molecule is created using this information. The QOX operator is used for the synthesis operation in the CRO algorithm as follows:

If ($r <$ MoleColl) Select randomly two molecules x_1, x_2 from the population If (synthesis criteria hold) Create new molecule x' using orthogonal crossover operation QOX(x_1, x_2)

The molecules $\{x_1, x_2\}$ represent molecular structures described earlier. The effectiveness and performance of the orthogonal CRO have been verified on 23 well-known uni-modal, high-dimensional multi-modal, and low-dimensional multi-modal benchmark functions. The approach was showed to be less efficient for low-dimensional functions [45]. Duan and Gan [46] use an orthogonal multi-objective CRO for optimal design of a brushless DC motor.

Adaptive Collision CRO

In the standard CRO, there is an overlap between functionalities of inter-molecular and on-wall ineffective collision operators leading to unnecessary computation times. To reduce this functional overlap, an adaptive collision scheme is introduced by Yu et al. [47, 48]. The adaptive collision consists of a new inter-molecular ineffective collision operator and an adaptive collision scheme.

In the inter-molecular ineffective collision operation, generally two on-wall ineffective collisions take place at the same time. Therefore, an inter-molecular operator is introduced to ineffective collisions such that it makes a difference between these two on-wall ineffective collisions. Two molecules $\{x_i, x_j\}$ are randomly selected and their fitness values (i.e. PE values) are calculated. Let $PE(x_i)$ be greater than $PE(x_j)$. Based on the PE values, two approaches are deployed to modify the molecules instead of neighbourhood operation usually employed in on-wall ineffective collision [47, 48].

$$\begin{cases} x_i'^d = \left(x_i^d - x_j^d\right) \times r^d + x_i^d \\ x_j'^d = \left(x_i^d - x_j^d\right) \times r^d + x_j^d \end{cases} \tag{26}$$

where $x_i'^d$ and $x_j'^d$ are the new molecules of the d-th element, x_i^d and x_j^d are old molecules of the d-th element and r^d is a random

number for each element over the interval [0, 1]. The mechanism expressed by Eq. (26) ensures that the new molecules $x_i'^d$ and $x_j'^d$ are not similar.

In the adaptive collision scheme, an adaptive collision rate (CollRate) is introduced. The CollRate is the ratio of occurrence of on-wall ineffective collision and inter-molecular reaction. The CollRate plays a critical role in the CRO performance. In standard CRO, CollRate is a user-defined parameter, which is fixed during execution and usually chosen empirically by users [49]. In adaptive collision scheme, CollRate is defined as a sigmoid function [47, 48] with two parameters count and FE$_{max}$.

$$CollRate = \frac{1}{1 + \exp\left[-6 \times \dfrac{count}{FE_{max}}\right]} \qquad (27)$$

The parameter count is the number of successful inter-molecular reaction. The value of count is incremented when a successful inter-molecular reaction occurs and it is decremented when an on-wall collision occurs. The parameter FE$_{max}$ is the maximum allowable value for function evaluations. The performance of the adaptive CRO has been verified experimentally on 16 different benchmark functions and compared with standard CRO [47, 48].

Elitist CRO

In the standard CRO algorithm, molecules are selected randomly. Though the random selection contributes to the diversity of the population, it impacts on the convergence rate. Duan and Gan [50] proposed an elitist CRO (ECRO) algorithm by introducing elitist strategies for selection, evolution, and crossover. Two new attributes of the molecule are introduced in ECRO: affinity and concentration. The affinity and concentration identify the quality of solution and similarity between solutions respectively. The efficiency of the ECRO has been verified on a contour-based target recognition problem.

Hybrid CRO and DE

Roy et al. [51] propose an improvement to CRO algorithm by introducing the mutation and crossover operators borrowed from Differential Evolution (DE) algorithm [52] and called it hybrid DE-CRO. The CRO operators such as on-wall ineffective collision operation, decompose operation, inter-molecular ineffective collision operation and synthesis operation are implemented using mutation and crossover operation of DE.

The on-wall ineffective collision operation in CRO is implemented using the mutation operation of DE. A new molecule is generated using the mutation operation as follows

$$x_{ij}' = x_{ij} + F^*\left(x_{mj} - x_{nj}\right) \qquad (28)$$

where x_{ij}' is the new j-th component of the i-th molecule, $\{x_{ij}, x_{mj}, x_{nj}\}$ are the j-th components of three different molecules chosen randomly from the current population and F is a positive control parameter.

The decompose operation is implemented using the crossover operation. To perform the crossover, one molecule x_m is selected randomly from the population and another molecule x_n is generated randomly. Two new molecules x_m' and x_n' are created by applying crossover operation on x_m and x_n.

The inter-molecular ineffective collision operation is implemented using crossover operation. Two new molecules are created by performing crossover operation on two randomly selected molecules x_m and x_n from the population.

Molecules are modified using synthesis collision operation implemented by applying conventional crossover operation from genetic algorithm.

The effectiveness and performance of the DE-CRO algorithm has been verified on four test systems of conventional static economic load dispatch problem [51]. Dutta et al. [53] applied the DE-CRO algorithm to unified power flow control problem to determine the optimal parameter setting for power system network.

Hybrid CRO and PSO

There are some good features of PSO algorithm that can be incorporated into CRO for improving exploration or exploitation. Nguyen et al. [54] combined the explorative and exploitative features of PSO and CRO. Due to low efficiency, the decomposition and synthesis operations are eliminated and a PSO-based update operation is performed instead. New molecules are created using neighbouring operations of CRO and mechanisms of PSO described by Eqs. (29)–(30). These molecules can be considered as molecules of CRO or particles of PSO. PSO and CRO use the same population generated initially. The basic operations involved in PSO algorithm are described by

$$v_i^d(k+1) = w \cdot v_i^d(k) + c_1 r_1\left(p_i^d - x_i^d\right) + c_2 r_2\left(p_{gi}^d - x_i^d\right) \qquad (29)$$

$$x_i^d(k+1) = x_i^d(k) + v_i^d(k+1) \times T \qquad (30)$$

where v_i^d is the velocity of i-th particle, x_i^d, p_i^d, and p_{gi}^d are the position, iteration best, and global best position of the d-th element respectively, w is the inertia weight, c_1 and c_2 are

cognitive and social coefficient respectively, r_1 and r_2 are random numbers generated between [0, 1] and T is the time, which is unity to convert the velocity into position. The computation of molecules (i.e. position of particles) is straightforward. If a PSO update criterion is satisfied, then molecules are updated using the PSO algorithm, i.e. using Eqs. (29)–(30), otherwise inter-molecular ineffective collision operation and on-wall ineffective collision operation are performed. Thus, the hybrid CRO-PSO algorithm repeats the PSO update, inter-molecular ineffective collision operation and on-wall ineffective collision operation until termination condition is satisfied. The CRO-PSO algorithm has been applied to well-known uni-modal and multi-modal benchmark functions.

Zhang and Duan [55] proposed another version of hybrid PSO-CRO approach, called PCRO, for solving the image matching problem where the best molecule is saved in each iteration. Once the on-wall ineffective collision and inter-molecular ineffective collision are performed, the molecule is updated by the distance between the original molecule and the current best molecule. In PCRO, the PSO update mechanism is simplified.

$$v_i^d(k+1) = c_1 r_1 \left(p_i^d - x_i^d \right) / T \qquad (31)$$

$$x_i^d(k+1) = x_i^d(k) + v_i^d(k+1) \times T \qquad (32)$$

where v_i^d, x_i^d, and p_i^d are the velocity, position, and iteration best position in the d-th element of the i-th particle, respectively. Li et al. [56] use a hybrid PSO-CRO algorithm for multi-object optimisation problems. The proposed algorithm balances the operators of CRO and PSO while exploring the search space effectively. The hybrid approach also improves convergence.

Other Chemistry-Based Algorithms

There are a few other chemistry-inspired algorithms reported in the literature. They are artificial chemical process algorithm (ACPA) based on the principles of artificial chemical process [57], artificial chemical reaction optimization (ACRO) [58] based on a different set of bi-molecular and uni-molecular chemical reactions different from CRO including redox (reduction-oxidation) reactions, chemical reaction algorithm (CRA) based on the principles of artificial chemistry [59], and Gases Brownian motion optimisation (GBMO) algorithm based on the laws of Brownian motion and turbulent rotation motion of gas molecules [60].

Conclusions

CRO algorithm is a recent addition to meta-heuristic family and to cognitive computation paradigm. The significant contribution of this paper is the introduction of inspiration from chemistry and the development of meta-heuristic algorithms based on the principles of chemical reactions, which are complementary to biology and physics inspired algorithms. CRO algorithm has attracted the attention of machine learning and cognitive computation community over the past few years and has been successfully applied in several real-world optimization problems.

To improve the performance of CRO and solution quality, a number of hybrid variants have been proposed. There has been little theoretical analysis done so far apart from the convergence analysis by Lam et al. [39]. Premature convergence, convergence speed, searching behaviours, and parameter selection are important issues in CRO that need further research.

References

1. Siddique N, Adeli H. Brief history of natural sciences for nature inspired computing in engineering. J Civ Eng Manag. 2016;22(3):287–301.
2. Siddique N, Adeli H. Nature-inspired computing: an overview and some future directions. Cogn Comput. 2015;7(6):706–14.
3. Holland JH. Adaptation in natural and artificial systems. Ann Arbor: University of Michigan Press; 1975.
4. Javed SG, Majid A, Ali S, Kausar N. A bio-inspired parallel-framework based multi-gene genetic programming approach to Denoise biomedical images. Cogn Comput. 2016;8(4):776–93.
5. Eiben AE, Smith JE. Introduction to evolutionary computing. Berlin: Springer-Verlag; 2003.
6. Zeng N, Wang Z, Zhang H, Alsaadi FE. A novel switching delayed PSO algorithm for estimating unknown parameters of lateral flow immunoassay. Cogn Comput. 2016;8(2):143–52.
7. Siddique N, Adeli H. Physics-based search and optimization: inspirations from nature. Expert Systems, 2016. 2016;33(6):607–23.
8. Wu Z, Pang W, Coghill GM. An integrated qualitative and quantitative biochemical model learning framework using evolutionary strategy and simulated annealing. Cogn Comput. 2015;7(6):637–51.

9. Siddique N, Adeli H. Simulated annealing, its variants and engineering applications. Int J Artif Intell Tools. 2016;25(6):1630001. 24 pages

10. Siddique N, Adeli H. Gravitational search algorithm and its variants. Int J Pattern Recognit Artif Intell. 2016;30(8):1639001. 22 pages

11. Siddique N, Adeli H. Harmony search algorithm and its variants. Int J Pattern Recognit Artif Intell. 2015;29(8):1539001. (22 pages)

12. Siddique N, Adeli H. Central force metaheuristic optimization. Scientia Iranica, − Transaction A: Civil Engineering. 2015;22(6): 1941–53. Invited Paper

13. Siddique N, Adeli H. Water drop algorithms. Int J Artif Intell Tools. 2014;23(6):1430002. (22 pages)

14. Siddique N, Adeli H. Spiral dynamics algorithm. Int J Artif Intell Tools. 2014;23(6):1430001. (24 pages)

15. Atkins PW, De Paula J. Physical chemistry. 4th Edt ed. Weinheim: Wiley-VCH; 2006.

16. Brückner R. Reaktionsmechanismen (in German). 3rd edt ed. München: Spektrum Akademischer Verlag; 2004.

17. Gibbs JW. A method of geometrical representation of the thermodynamic properties of substances by means of surfaces. Trans Connecticut Acad Arts Sci. 1873;1873(2):382–404.

18. Lam AYS, Li VOK. Chemical-reaction-inspired meta-heuristic for optimization. IEEE Trans Evol Comput. 2010;14(3):381–99.

19. Yu, J.J.Q., Lam, A.Y.S., and Li, V.O.K. (2011) Evolutionary artificial neural network based on chemical reaction optimization. Proc IEEE Congr Evol Comput (CEC), June 2011, 2011, 2083–2090.

20. Siddique N, Adeli H. Computational intelligence: synergies of fuzzy logic, neural networks and evolutionary computing. UK: Wiley; 2013.

21. Bolourchi A, Masri SF, Aldraihem OJ. Studies into computational intelligence and evolutionary approaches for model-free Identification of hysteretic systems. Computer-Aided Civil Infrastruct Eng. 2015;30(5):330–46.

22. Reyes O, Morell C, Ventura S. Evolutionary feature weighting to improve the performance of multi-label lazy algorithms. Integrated Computer-Aided Engineering. 2014;21(4):339–54.

23. Lee HG, Yi C, Lee DE, Arditi D. An advanced stochastic time-cost tradeoff analysis based on a CPM-guided multi-objective genetic algorithm. Computer-Aided Civil Infrastruct Eng. 2015;30(10): 824–42.

24. Lin DY, Ku YH. Using genetic algorithms to optimize stopping patterns for passenger rail transportation. Comput-Aid Civil Infrastruct Eng. 2014;29(4):264–78.

25. Rashidi S, Ranjitkar P. Bus dwell time modeling using gene expression programming. Comput-Aid Civil Infrastruct Eng. 2015;30(6): 478–89.

26. Paris PCD, Pedrino EC, Nicoletti MC. Automatic learning of image filters using Cartesian genetic programming. Integr Comput-Aid Eng. 2015;22(2):135–51.

27. Lam AYS, Li VOK. Chemical reaction optimization. A tutorial, Memetic Computing. 2012;4(1):3–17.

28. Cheng J, Zhang G, Caraffini F, Neri F. Multicriteria adaptive differential evolution for global numerical optimization. Integr Comput-Aid Eng. 2015;22(2):103–17.

29. Vincenzi L, Savoia M. Coupling response surface and differential evolution for parameter identification problems. Comput-Aid Civil Infrastruct Eng. 2015;30(5):376–93.

30. Shabbir F, Omenzetter P. Particle swarm optimization with sequential niche technique for dynamic finite element model updating. Comput-Aid Civil Infrastruct Eng. 2015;30(5):359–75.

31. Zeng Z, Xu J, Wu S, Shen M. Antithetic method-based particle swarm optimization for a queuing network problem with fuzzy data in concrete transportation systems. Comput-Aid Civil Infrastruct Eng. 2014;29(10):771–800.

32. Lam AYS, Li VOK, Yu JJQ. Real-coded chemical reaction optimization. IEEE Trans Evol Comput. 2012;16(3):339–53.

33. Bhattacharjee K, Bhattacharya A, Dey SH. Real coded chemical reaction based optimization for short-term hydrothermal scheduling. Appl Soft Comput. 2014;24:962–76.

34. Tizhoosh, H.R.. Opposition-based learning: a new scheme for machine intelligence, Proceedings of International Conference on Computational Intelligence for Modelling Control and Automation, CIMCA'2005, Vienna–Austria, 28th–30th November; 2005, I: 695–701.

35. Chatterjee A, Ghoshal SP, Mukherjee V. Solution of combined economic and emission dispatch problems of power systems by an opposition-based harmony search algorithm. Int J Electr Power Energy Syst. 2012;39(1):9–20.

36. Shaw B, Mukherjee V, Ghoshal SP. A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems. Electr Power Energy Syst. 2012;35(1):21–33.

37. Upadhyay P, Kar R, Mandal D, Ghoshal SP, Mukherjee V. A novel design method for optimal IIR system Identification using opposition-based harmony search algorithm. J Frankl Inst. 2014;2014(351):2454–88.

38. Bhattacharjee K, Bhattacharya A, Dey SH. Opposition real coded chemical reaction based optimization for short-term hydrothermal scheduling. Electr Power Energy Syst. 2014;2014(63):145–57.

39. Lam AYS, Li VOK, Xu J. On the convergence of chemical reaction optimization for combinatorial optimization. IEEE Trans Evol Comput. 2013;17(5):605–20.

40. Montgomery, D. C. (2008) Design and analysis of experiments. Wiley; 2008.

41. Ho S-J, Ho S-Y, Shu L-S. OSA: orthogonal simulated annealing algorithm and its application to designing mixed H_2/H_∞ optimal controllers. IEEE Trans Syst, Man Cybern – Part A: Syst Hum. 2004;34(5):588–600.

42. Jung J, Jayakrishnan R, Park JY. Dynamic shared-taxi dispatch algorithm with hybrid simulated annealing. Comput-Aid Civil Infrastruct Eng. 2016;31(4):275–91.

43. Zhan Z-H, Zhang J, Li Y, Shi Y-H. Orthogonal learning particle swarm optimization. IEEE Trans Evol Comput. 2011;2011(15): 832–47.

44. Hu, X., Zhang, J., and Zhong, J.. An enhanced genetic algorithm with orthogonal design. In: IEEE Congress on Evolutionary Computation, CEC; 2006. 3174–3181.

45. Li Z-Y, Li Z, Nguyen TT, Chen S-M. Orthogonal chemical reaction optimization algorithm for global numerical optimization problems. Expert Syst Appl. 2015;2015(42):3242–52.

46. Duan H, Gan L. Orthogonal multi-objective chemical reaction optimization approach for the brushless DC motor design. IEEE Trans Magn. 2015;51(1):7000207.

47. Yu, J. J. Q., Li, V. O. K. and Lam, A. Y. S.. An inter-molecular adaptive collision scheme for chemical reaction optimization. Proceedings of the IEEE congress on evolutionary computation (CEC); 2014. pp. 1998–2004.

48. Yu, J. J. Q., Lam, A. Y. S. and Li, V. O. K.. Adaptive chemical reaction optimization for global optimisation, 2015 I.E. Congress on Evolutionary Computation; 2015. 3192–3199.

49. Xu J, Lam A, Li VK. Chemical reaction optimization for task scheduling in grid computing. IEEE Trans Parallel Distrib Syst. 2011;22(10):1624–31.

50. Duan H, Gan L. Elitist chemical reaction optimization for contour-based target recognition in aerial images. IEEE Trans Geosci Remote Sens. 2015;53(5):2845–59.

51. Roy PK, Bhui S, Paul C. Solution of economic load dispatch using hybrid chemical reaction optimization approach. Appl Soft Comput. 2014;2014(24):109–25.

52. Storn R, Price KV, Lampinen J. Differential evolution—a practical approach to global optimization. Berlin: Springer; 2005.

53. Dutta S, Roy PK, Nandi D. Optimal location of UPFC controller in transmission network using hybrid chemical reaction optimization algorithm. Electr Power Energy Syst. 2015;2015(64):194–211.

54. Nguyen TT, Li Z-Y, Zhang S-W, Truong TK. A hybrid algorithm based on particle swarm and chemical reaction optimization. Expert Syst Appl. 2014;2014(41):2134–43.

55. Zhang Z, Duan H. A hybrid Particle Chemical Reaction Optimization for biological image matching based on lateral inhibition. Optik. 2014;2014(125):5757–63.

56. Li Z, Nguyen TT, Chen SM, Truong TK. A hybrid algorithm based on particle swarm and chemical reaction optimization for multi-object problems. Appl Soft Comput. 2015;2015(35):525–40.

57. Irizarry R. LARES: an artificial chemical process approach for optimization. Evol Comput J. 2004;12(4):435–59.

58. Alatas B. ACROA: artificial chemical reaction optimization algorithm for global optimization. Expert Syst Appl. 2011;2011(38):13170–80.

59. Melin P, Astudillo L, Castillo O, Valdez F, Valdez F. Optimal design of type-2 and type-1 fuzzy tracking controllers for autonomous mobile robots under perturbed torques using a new chemical optimization paradigm. Expert Syst Appl. 2013;2013(40):3185–95.

60. Abdechiri M, Meybodi MR, Bahrami H. Gases Brownian motion optimization: an algorithm for optimization (GBMO). Appl Soft Comput. 2013;2013(13):2932–46.