

# Chemical Reaction Optimization for the Optimal Power Flow Problem

Yi Sun

Department of Electrical and  
Electronic Engineering  
The University of Hong Kong  
Email: sunyimik@hku.hk

Albert Y.S. Lam

Department of Electrical Engineering  
and Computer Sciences  
University of California, Berkeley  
Email: albertlam@iee.ee.org

Victor O.K. Li, Jin Xu,  
and James J.Q. Yu

Department of Electrical and  
Electronic Engineering  
The University of Hong Kong  
Email: {vli, xujin, jqyu}@eee.hku.hk

**Abstract**—This paper presents an implementation of the Chemical Reaction Optimization (CRO) algorithm to solve the optimal power flow (OPF) problem in power systems with the objective of minimizing generation costs. Multiple constraints, such as the balance of the power, bus voltage magnitude limits, transmission line flow limits, transformer tap settings, etc., are considered. We adapt the CRO framework to the OPF problem by redesigning the elementary reaction operators. We perform simulations on the standard IEEE-14, -30, and -57 bus benchmark systems. We compare the perform of CRO with other reported evolutionary algorithms in the IEEE-30 test case. Simulation results show that CRO can obtain a solution with the lowest cost, when compared with other algorithms. To be more complete, we also give the average result for the IEEE-30 case, and the best and average results for the IEEE-14 and -57 test cases. The results given in this paper suggest that CRO is a better alternative for solving the OPF problem, as well as its variants for the future smart grid.

**Index Terms**—Chemical reaction optimization, metaheuristic, optimal power flow, power grid, smart grid.

## I. INTRODUCTION

Since it was first defined in 1968 by Dommel and Tinney [1], the optimal power flow (OPF) problem has been received much great attention in the research and industrial communities. Due to increasing environmental concerns such as global warming, the increasing demand for electricity, and the expansion of the power system, allocating power generation with minimum cost without violating the system and security constraints is becoming more important. Therefore the OPF problem has been widely applied in real-world power systems [2]. It also plays a significant role in electricity pricing and congestion management [3]. Moreover, the smart grid [4] makes OPF even more important.

In general, the OPF problem is a non-convex, and strictly constrained static optimization problem with both continuous and discrete variables. The nonlinear AC power flow equality constraints, also defined as the power flow (PF) problem, largely constitute the non-convexity of the problem.

Early approaches handling the OPF problem are mostly mathematical convex programming techniques, such as non-linear programming [1] [5] [6], linear programming [7], [8][9][10], quadratic programming [11][12], Newton-based algorithm [13], and interior point methods [14]. Generally, most of these optimization algorithms adopt sensitivity analysis and

gradient-based searching around an operating point along the objective function in the solution space confined by the system constraints. However, it should be noted that the OPF problem is not only non-convex, but is also a multimodal problem. In other words, it contains many local minimums and local search techniques generally fail to obtain the optimal solution [15]. Additionally, the conventional algorithms for OPF suffer from some specific drawbacks, such as complexity and insecure convergence, i.e., the possibility of failure in converging to the global optimal solution. For example, the Newton-based algorithms are very sensitive to the initial conditions which result in failure in convergence due to improper start points [16]. For the interior point methods, if the step size is not selected properly, it may lead to a solution that violates the original nonlinear domain in sublinear problems [17]. Although these drawbacks limit the performance and the effectiveness when dealing with complex objective functions, high dimensions and complicated constraints, they are still widely adopted in solving the OPF problem in practice.

With the recent advancements in computational technology, many revolutionary optimization methods have been proposed to solve the OPF problem. For example, Genetic Algorithm (GA) [17] [18], Differential Evolution (DE) [19], Particle Swarm Optimization (PSO) [20], and Biogeography-Based Optimization (BBO) [21]. They provide feasible simulation results and can generally overcome the drawbacks of the traditional methods.

Inspired by natural selection and genetics, GA has become popular since the 1970s and has been applied to solve many complex problems. GA is based on string structures (chromosomes), typically a concatenated list of binary digits representing the coding of the control parameters (phenotype) of a given problem. The chromosomes are composed of genes. GA consists of parent selection, crossover and mutation. In [18], GA is regarded to have the most important advantage of robustness in solving OPF.

DE is also based on natural evolution. Similar to GA, it is also composed of mutation, crossover, and selection to determine the optimal result. The difference is that DE creates new offsprings by generating a noisy replica of each individual of the population [19]. No coding is required to convert the variables into a binary form.

PSO is another evolutionary-based approach, which is motivated by natural behavior of organisms such as bird flocking. It combines the social psychology principles in cognition among human beings (socio-cognition human agents) and evolutionary computations [20]. Particles represent candidates of solution. In a PSO system, particles change their positions by flying around in a multi-dimensional search space until a relatively unchanging position has been encountered, or until computational limitations are exceeded.

BBO mimics the migration of species from one island to another, the raising, and the extinction of species. There are two basic mechanisms in BBO, migration and mutation. Geographical areas that are well suited for residences for biological species are said to have a high habitat suitability index and the variables that characterize habitability are called suitability index variables [21]. BBO improves its performance by replacing the worse results with the better ones. In order to improve the quality during the computation, which can be considered as a promotion for faster convergence.

Recently, a new general-purpose optimization method inspired from the molecular behaviors in chemical reactions, called Chemical Reaction Optimization (CRO), has been proposed [22]. CRO follows the natural tendency of a chemical reaction to settle in the state with the minimized free energy, i.e., the most stable state. It is not difficult to see the similarity between optimization and chemical reactions. A molecule corresponds to a solution of the optimization problem, and the free energy of the molecule corresponds to the objective function value of the solution. CRO is developed based on this discovery, mimicking the behavior of molecules in chemical reactions.

Simulations of OPF problem are conducted on the standard IEEE-14, -30, and-57 bus system [23] and the simulation results show that CRO can give good performance in solving the problem. The rest of the paper is organized as follows. We formulate the problem in Section II. In Section III, we present the algorithm design. The simulation results are discussed in Section IV and we conclude the paper in Section V.

## II. PROBLEM FORMULATION

We consider a transmission network with  $N$  buses, collected into a set  $\mathcal{N}$ . Each bus  $i \in \mathcal{N}$  is characterized by four variables: active power injection  $P_i \in \mathcal{R}$ , reactive power injection  $Q_i \in \mathcal{R}$ , voltage magnitude  $V_i \in \mathcal{R}$ , and a phase angle  $\theta_i \in [-\pi, \pi]$ . In general, each bus  $i$  can have both active power generation ( $P_{G_i} \geq 0$ ) and consumption ( $P_{L_i} \geq 0$ ). Its active power injection  $P_i$  refers to the net active power injected into the network from bus  $i$ , and thus we have  $P_i = P_{G_i} - P_{L_i}$  (i.e. generation minus consumption). If  $P_i > 0$ , bus  $i$  supplies power to the network. Otherwise, it demands power from the network. Similarly, the reactive power injection  $Q_i$  of bus  $i$  can be defined by its reactive power generation  $Q_{G_i} \geq 0$  and consumption  $Q_{L_i} \geq 0$ , i.e.,  $Q_i = Q_{G_i} - Q_{L_i}$ . Each bus can be classified into three types: a slack bus, a PV bus, or a PQ bus, where PV and PQ buses refer to generator and load buses, respectively. When  $P_i$  is positive, bus  $i$  is a generator (PV)

bus. Otherwise, it is a load (PQ) bus. There is only one slack bus in the network and it is considered as a special generator bus with known phase angle for the reference of other buses. We denote the sets of the slack bus, the PQ buses and the PV buses by  $\mathcal{N}_0, \mathcal{N}_{PQ}, \mathcal{N}_{PV}$ , respectively. Assume that there are  $N_{PV}$  PV buses and  $N_{PQ}$  PQ buses in the network. Hence, we have  $N = 1 + N_{PV} + N_{PQ}$ . For the PV buses and the slack bus, their voltage magnitudes ( $V$ ), active and reactive power generations ( $P$  and  $Q$ ) can only operate in certain ranges:

$$V_i^{min} \leq V_i \leq V_i^{max}, \quad \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}, \quad (1)$$

$$P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max}, \quad \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}, \quad (2)$$

$$Q_{G_i}^{min} \leq Q_{G_i} \leq Q_{G_i}^{max}, \quad \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}. \quad (3)$$

Similarly, for the PQ buses, we have

$$V_i^{min} \leq V_i \leq V_i^{max}, \quad \forall i \in \mathcal{N}_{PQ}. \quad (4)$$

For any PV or slack bus  $i$ , we impose a cost  $f_i$  on the amount of active power generation, i.e.,

$$f_i(P_{G_i}) = a_i + b_i P_{G_i} + c_i P_{G_i}^2, \quad \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}. \quad (5)$$

There are  $N_C$  shunt volt-ampere reactive (VAR) compensators, which attach to a subset of the buses, denoted by  $\mathcal{N}_C$ , to unify the power factor (i.e., to improve the power factor by making it closer to one). Attaching a compensator to a bus has the same effect of adding conductance to the bus. We denote the reactive power generated by the shunt compensator at bus  $i$  by  $Q_{C_i}$ . When bus  $i$  does not have a compensator attached (i.e.,  $i \notin \mathcal{N}_C$ ), the amount of reactive power contributed by compensators  $Q_{C_i}$  are set to zero. The shunt compensators can only produce reactive power within certain limits, specified by

$$Q_{C_i}^{min} \leq Q_{C_i} \leq Q_{C_i}^{max}, \quad i \in \mathcal{N}_C. \quad (6)$$

The transmission line connecting buses  $i$  and  $k$  is characterized by its conductance  $g_{ik}$ , its susceptance  $b_{ik}$ , and the phase angle difference  $\theta_{ik} = \theta_i - \theta_k$ . If buses  $i$  and  $k$  are not connected, then both  $g_{ik}$  and  $b_{ik}$  are equal to zero. Let  $\mathcal{E}$  be the set of transmission lines, where  $|\mathcal{E}| = N_{TL}$  and  $N_{TL}$  is the total number of transmission lines in the network. Let  $I_i, i = 1, \dots, N_{TL}$ , be the magnitude of the current flow on the  $i$ th power line. To ensure that there is no excessive power flow on a line, each power line has a current flow limitation given by

$$I_i \leq I_i^{max}, \quad i \in \mathcal{E}. \quad (7)$$

There are also  $N_T$  transformers in the system, each of which attaches to a transmission line between two buses. Let  $\mathcal{E}_T$  be the set of transmission lines with transformers attached. Consider that the  $i$ th transformer is on the transmission line between buses  $l$  and  $k$ . It is used to adjust the ratio of voltage magnitudes of buses  $l$  to  $k$ . This ratio is called the tap ratio, denoted by  $T_i$ , which is bounded by

$$T_i^{min} \leq T_i \leq T_i^{max}, \quad i \in \mathcal{E}_T. \quad (8)$$

For better illustration, an example of a five-bus system is given in Fig. 1, which is adapted from [24].

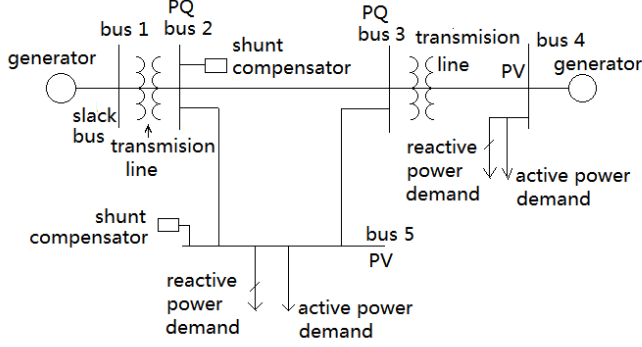


Fig. 1. A five-bus power system example

The network can be characterized by the admittance matrix  $Y_{bus} = (Y_{ik}, 1 \leq i, k \leq N)$ , where

$$Y_{ik} = \begin{cases} -(g_{ik} + jb_{ik}) & \text{if buses } i, k \text{ are connected,} \\ -\sum_{l=1, l \neq i}^N Y_{il} + jQC_i & \text{if } k = i, \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

By the Kirchhoff's Law and Ohm's Law [25], the network is said to be balanced when the following equalities hold:

$$P_{G_i} - P_{L_i} = P_i = V_i \sum_{k=1}^N V_k (G_{ik} \cos \theta_{ik} + B_{ik} \sin \theta_{ik}), \quad (10)$$

$$Q_{G_i} - Q_{L_i} = Q_i = V_i \sum_{k=1}^N V_k (G_{ik} \sin \theta_{ik} - B_{ik} \cos \theta_{ik}), \quad (11)$$

where  $G_{ik} = \text{Re}(Y_{ik})$  and  $B_{ik} = \text{Im}(Y_{ik})$ . Given the values of some variables, solving (10) and (11) for the unknowns is called the PF problem.

The unknown variables can be further classified into two types: control and dependent variables. The former are those we can control while the latter depends on the former via the PF problem, i.e., via (10) and (11). The control variables include voltage magnitudes of the slack and PV buses, active power generations of PV buses, reactive power generated by the shunt compensators, and the tap ratios of the transformers. The rest are dependent variables. To summarize, the constants (i.e., the values specified by the problem instance) and variables are listed as follows:

- I) Constants:  $(P_{L_i}, Q_{L_i}, \forall i \in \mathcal{N})$ ,  $(P_{G_i}, Q_{G_i}, \forall i \in \mathcal{N}_{PQ})$ ,  $(a_i, b_i, c_i, \forall i \in \mathcal{N}_{PV})$ ,  $(g_{ik}, b_{ik}, \forall i, k \in \mathcal{N}, i \neq k)$ ,  $(\theta_i, \forall i \in \mathcal{N}_0)$ ;
- II) Control variables:  $(V_i, \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV})$ ,  $(P_{G_i}, \forall i \in \mathcal{N}_{PV})$ ,  $(Q_{C_i}, \forall i \in \mathcal{N}_C)$ ,  $(T_i, \forall i \in \mathcal{N}_T)$ ;
- III) Dependent variables:  $(V_i, \forall i \in \mathcal{N}_{PQ})$ ,  $(P_{G_i}, \forall i \in \mathcal{N}_0)$ ,  $(Q_{G_i}, \forall i \in \mathcal{N}_0 \cup \mathcal{N}_{PV})$ ,  $(\theta_i, \forall i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ})$ ,  $(I_i, \forall i \in \mathcal{E})$ .

The OPF problem aims to minimize the generation costs of active power such that the network is balanced and all variables operate within their limits. Mathematically, the OPF problem is stated as

$$\begin{aligned} \text{minimize } F &= \sum_{i \in \mathcal{N}_{PV} \cup \mathcal{N}_0} f_i(P_{G_i}) \\ &= \sum_{i \in \mathcal{N}_{PV} \cup \mathcal{N}_0} (b_i \times P_{G_i} + c_i \times P_{G_i}^2), \end{aligned} \quad (12)$$

subject to

Equality constraints: (10), (11),

Inequality constraints: (1), (2), (3), (4), (6),

(7), (8),

where  $b_i$  and  $c_i$ , for all  $i$ , are cost coefficients given by the problem data.<sup>1</sup>

#### A. Power Flow Problem

In this subsection, given the control variables, we show how to determine the dependent variables by solving (10) and (11). We apply the Newton-Raphson Method (NRM) [26] to solve the PF problem. We illustrate this method briefly in the following.

NRM is an iterative method based on Taylor series with higher order terms ignored. Recall that  $P_i = P_{G_i} - P_{L_i}$  and  $Q_i = Q_{G_i} - Q_{L_i}$  are determined from the given problem data and the control variables. Thus, we know  $P_i$  and  $Q_i$  for all  $i \in \mathcal{N}$  and we also know  $V_i$  for all  $i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}$ . The phase of the slack bus  $\theta_i, i \in \mathcal{N}_0$  is also pre-defined (usually set to 0). We are determining  $V_i$  for all  $i \in \mathcal{N}_{PQ}$  and  $\theta_i$  for all  $i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ}$ .

We set the initial voltage magnitudes of the PQ buses (i.e.,  $V_i(0), \forall i \in \mathcal{N}_{PQ}$ ) to 1 p.u. and the initial phase angles (i.e.,  $\theta_i(0), \forall i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ}$ ) to zero. Let  $P_i(n)$  and  $Q_i(n)$  be the values calculated by NRM in Iteration  $n$ . We define

$$\Delta P_i(n) = P_i - P_i(n) \text{ and} \quad (13)$$

$$\Delta Q_i(n) = Q_i - Q_i(n). \quad (14)$$

Without loss of generality, assume that the buses are ordered such that we can arrange the variables of the same type into a vector following that order. Let  $\Delta P(n) = (\Delta P_i(n), i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ})^T$ ,  $\Delta Q(n) = (\Delta Q_i(n), i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ})^T$ ,  $\Delta \theta(n) = (\Delta \theta_i(n), i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ})^T$ , and  $\Delta V(n) = (\Delta V_i(n), i \in \mathcal{N}_{PQ})^T$ . Then we calculate

$$\begin{pmatrix} \Delta \theta(n) \\ \Delta V(n) \end{pmatrix} = J(n)^{-1} \begin{pmatrix} \Delta P(n) \\ \Delta Q(n) \end{pmatrix}, \quad (15)$$

where

$$J(n) = \begin{pmatrix} \frac{\partial \Delta P(n)}{\partial \theta(n)} & \frac{\partial \Delta P(n)}{\partial V(n)} \\ \frac{\partial \Delta Q(n)}{\partial \theta(n)} & \frac{\partial \Delta Q(n)}{\partial V(n)} \end{pmatrix}. \quad (16)$$

We update  $\theta_i$  and  $V_i$  by

$$\theta_i(n+1) = \theta_i(n) + \Delta \theta_i(n), \forall i \in \mathcal{N}_{PV} \cup \mathcal{N}_{PQ} \text{ and} \quad (17)$$

$$V_i(n+1) = V_i(n) + \Delta V_i(n), \forall i \in \mathcal{N}_{PQ}. \quad (18)$$

<sup>1</sup> $a_i$  can be ignored as it is a constant term given in (5).

Note that only the voltage magnitudes of the slack bus and PV buses are control variables. We determine the voltage magnitudes of the PQ buses by considering the transformer ratios. If two PQ buses are connected by a transformer, the voltage magnitude for the so-called non-standard side will also be decided by the tap ratio from the standard side voltage. Thus we have  $V_i = V_k/T_i$ , where  $(i, k) \in \mathcal{E}_T, i \in \mathcal{N}_{PQ}, k \in \mathcal{N}_0 \cup \mathcal{N}_{PV} \cup \mathcal{N}_{PQ}$ . With (10) and (11), we can determine  $P_i(n+1)$  and  $Q_i(n+1)$ . A new iteration is started until both  $\Delta P_i$  and  $\Delta Q_i$  are smaller than a certain threshold  $\xi$ , for all  $i$ . With all  $V_i$  and  $\theta_i$  known, all other unknown power injections and current flows can be determined accordingly.

### III. ALGORITHM DESIGN

#### A. Background

CRO is a general-purpose optimization framework. Consider a number of molecules in a container with a central energy buffer attached. It defines the procedures to select candidate solutions from the solution space and the conditions to accept new solutions. CRO has been successfully applied to many research and practical problems, including channel assignment in wireless mesh networks [22], the population transition problem in peer-to-peer live streaming [27], artificial neural network training [28], and the network coding optimization problem [29]. In this paper, we apply CRO to solve the OPF problem.

First we define the algorithmic details to fit the solution structures and constraints of the problems. In the following, we will discuss the essential components to design CRO for the problem. For other details of CRO, the interested reader may refer to [22], [30].

#### B. Modified Objective Function

Given the values of the control variables, we can determine those of the dependent variables by solving the PF problem. Since we can control the values of the control variables, we can ensure that the chosen values are within their limits. However, we cannot guarantee that the dependent variables do not exceed their limits. To ensure that a dependent variable is within its limit, we add a penalty function to the objective function  $F$ . Hence, the modified objective function becomes

$$F' = F + \gamma_V \sum_{i \in \mathcal{N}_{PQ}} (V_i - V_i^{lim})^2 + \gamma_G \sum_{i \in \mathcal{N}_0} (P_{G_i} - P_{G_i}^{lim})^2 + \gamma_Q \sum_{i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}} (Q_{G_i} - Q_{G_i}^{lim})^2 + \gamma_I \sum_{i \in \mathcal{E}} (I_i - I_i^{lim})^2, \quad (19)$$

where  $\gamma_V, \gamma_G, \gamma_Q$ , and  $\gamma_I$  are penalty coefficients and

$$V_i^{lim} = \begin{cases} V_i^{min} & \text{if } V_i < V_i^{min}, \\ V_i^{max} & \text{if } V_i > V_i^{max}, \\ V_i & \text{otherwise,} \end{cases} \quad \forall i \in \mathcal{N}_{PQ}, \quad (20)$$

$$P_{G_i}^{lim} = \begin{cases} P_{G_i}^{min} & \text{if } P_{G_i} < P_{G_i}^{min}, \\ P_{G_i}^{max} & \text{if } P_{G_i} > P_{G_i}^{max}, \\ P_{G_i} & \text{otherwise,} \end{cases} \quad i \in \mathcal{N}_0, \quad (21)$$

$$Q_{G_i}^{lim} = \begin{cases} Q_{G_i}^{min} & \text{if } Q_{G_i} < Q_{G_i}^{min}, \\ Q_{G_i}^{max} & \text{if } Q_{G_i} > Q_{G_i}^{max}, \\ Q_{G_i} & \text{otherwise,} \end{cases} \quad i \in \mathcal{N}_0 \cup \mathcal{N}_{PV}, \quad (22)$$

$$I_i^{lim} = \begin{cases} I_i^{max} & \text{if } I_i > I_i^{max}, \\ I_i & \text{otherwise.} \end{cases} \quad i \in \mathcal{E}. \quad (23)$$

Eq. (19) is adapted from [21]. In [21], transmission line flows are penalized instead of the current flows. As long as no limits are violated, the computed solutions are the same irrespective of whether power flows or current flows are chosen to be penalized. From now on, we minimize  $F'$  in (19) instead of  $F$  in (12). As long as the final solution does not violate any constraints, the objective function values computed from (19) and (12) are identical.

#### C. Manipulated Agents

The manipulated agents of CRO are molecules, each of which contains a profile of attributes. These attributes include a molecular structure ( $\omega$ ), potential energy ( $PE$ ), kinetic energy ( $KE$ ), number of hit ( $numHit$ ), minimum number of hit ( $minHit$ ), and some other optional ones.  $\omega$  carries a candidate solution which only contains the control variables.<sup>2</sup> Let  $\tilde{F}(\omega)$  be a function evaluating  $F'$  in (19) when given  $\omega$ : Given  $\omega$ , we determine the dependent variables by solving PF; then we evaluate the control and dependent variables with (19). We define  $PE$  associated with  $\omega$  as

$$PE_\omega = \tilde{F}(\omega). \quad (24)$$

$KE$  represents the tolerance of the molecule having a less feasible structure (i.e. a solution with a higher objective function value). The purpose of  $KE$  will become clear in the subsequent subsections.

#### D. Elementary Reactions

There are four kinds of elementary reactions defined in CRO and they are explained in the following:

1) *On-wall Ineffective Collision*: An on-wall ineffective collision happens when a molecule hits the container wall and results in a slight change to the molecular structure. Suppose the current molecule structure is  $\omega$  and it tends to change to a new one  $\omega' = N(\omega)$ , where  $N(\cdot)$  is a neighborhood search operator and  $N(\omega)$  returns a solution in the neighborhood of  $\omega$ . We adopt a similar approach used in [31] to define  $N(\cdot)$ . Instead of one variable, we add a perturbation to *each* variable in  $\omega$  to produce  $\omega'$ . Perturbations are drawn from a Gaussian distribution with zero mean and variance  $\sigma^2$ .

However, some of variables in  $\omega'$  may violate its limits, i.e., the inequalities stated in the problem. We adopt the hybrid scheme (HS) to handle the boundary constraints [32]. With HS, we perform reflection or absorption at a violated boundary. For reflection, we obtain the new value of the variable by reflecting violated boundary with the amount of violation

<sup>2</sup>Theoretically, a candidate solution should be composed of both the control and the dependent variables. Since the latter can be calculated from the former by solving PF, the dependent variables are removed from  $\omega$  for simplicity.

while absorption sets the new value to be the boundary value. Let  $\omega'(i)$  be the  $i$ th variable in  $\omega'$  and  $u(i)$  and  $l(i)$  be its upper and lower limits. We update  $\omega'(i)$  by

$$\omega'(i) := \begin{cases} 2 \times l(i) - \omega(i)' & \text{if } \omega'(i) < l(i) \text{ AND } t \geq 0.5, \\ 2 \times u(i) - \omega(i)' & \text{if } \omega'(i) > u(i) \text{ AND } t \geq 0.5, \\ l(i) & \text{if } \omega'(i) < l(i) \text{ AND } t < 0.5, \\ u(i) & \text{if } \omega'(i) > u(i) \text{ AND } t < 0.5, \end{cases} \quad (25)$$

where  $t$  is a random number generated in  $[0, 1]$ . An example of neighborhood search where the solution has four variables is given as:

$$\underbrace{[2, 3.5, 6.3, 1.4]}_{\omega} \rightarrow \underbrace{[2.04, 3.52, 6.27, 1.38]}_{\omega'}$$

The on-wall collision is allowed if

$$PE_{\omega} + KE_{\omega} \geq PE_{\omega'}, \quad (26)$$

where  $PE_{\omega'}$  is computed according to (24). In this case,  $KE_{\omega'}$  of the new structure  $\omega'$  is obtained by

$$KE_{\omega'} = (PE_{\omega} + KE_{\omega} - PE_{\omega'}) \times q, \quad (27)$$

where  $q$  is a random number generated in  $[KELossRate, 1]$  and  $KELossRate$  is a system parameter of CRO.  $numHit_{\omega}$  is increased by one and  $minHit_{\omega}$  will be equal to  $numHit_{\omega}$  if the new solution has better objective function value it has experienced. The remaining  $(1 - q)$  portion of energy is transferred to a central energy buffer (*buffer*). If (26) is not met, the change will be forbidden and the molecule will retain its original  $\omega$ ,  $PE_{\omega}$ , and  $KE_{\omega}$ .

2) *Decomposition*: A decomposition happens when a molecule hits a wall and splits into two smaller parts. This imposes a vigorous change to the molecule and the resultant molecules have substantially different molecular structures from the original one. Suppose that  $\omega$  is transformed to  $\omega'_1$  and  $\omega'_2$ . We apply the previously defined neighborhood search operator to compute  $\omega'_1$  and  $\omega'_2$ , i.e.,  $\omega'_1 = N(\omega)$  and  $\omega'_2 = N(\omega)$ .

In general, a decomposition happens if

$$PE_{\omega} + KE_{\omega} \geq PE_{\omega'_1} + PE_{\omega'_2}. \quad (28)$$

Let  $temp_1 = PE_{\omega} + KE_{\omega} - PE_{\omega'_1} - PE_{\omega'_2}$ . If (28) holds, we get

$$\begin{aligned} KE_{\omega'_1} &= temp_1 \times k \quad \text{and} \\ KE_{\omega'_2} &= temp_1 \times (1 - k), \end{aligned} \quad (29)$$

where  $k$  is a random number in  $[0, 1]$ . Note that normally  $PE_{\omega}$ ,  $PE_{\omega'_1}$  and  $PE_{\omega'_2}$  have similar values and  $KE_{\omega}$  may not be large enough to satisfy (28). In this case, energy stored in *buffer* will assist the decomposition. We consider

$$PE_{\omega} + KE_{\omega} + (buffer \times m_1 \times m_2) \geq PE_{\omega'_1} + PE_{\omega'_2}, \quad (30)$$

where  $m_1$  and  $m_2$  are random numbers independently and uniformly generated between  $[0, 1]$ .

Let  $temp_2 = PE_{\omega} + KE_{\omega} + (buffer \times m_1 \times m_2) - (PE_{\omega'_1} + PE_{\omega'_2})$ . If (30) is satisfied, the change is allowed and  $KE_{\omega'_1}$ ,  $KE_{\omega'_2}$  are updated as

$$\begin{aligned} KE_{\omega'_1} &= temp_2 \times p \\ KE_{\omega'_2} &= temp_2 \times (1 - p), \end{aligned}$$

where  $p$  is a random number in  $[0, 1]$ .

The aim of using  $m_1$  and  $m_2$  is to restrict the  $KE_{\omega'_1}$  and  $KE_{\omega'_2}$  from being too large, for the *buffer* is normally large. The *buffer* is then updated by  $temp_2 + buffer - KE_{\omega'_1} - KE_{\omega'_2}$ . The  $numHit_{\omega'_1}$ ,  $minHit_{\omega'_1}$ ,  $numHit_{\omega'_2}$  and  $minHit_{\omega'_2}$  are all set to zero. If both (28) and (30) are not satisfied, the decomposition will not happen and the original molecule structure,  $PE$ , and  $KE$  will be retained.

3) *Inter-molecular Ineffective Collision*: An inter-molecular ineffective collision occurs when two molecules hit each other and then bounce back. The energy requirement for accepting the change is similar to that of an on-wall ineffective collision except that the involved number of molecules in the reaction is more than one and there is no interaction with *buffer*.

Assume the original molecular structures are  $\omega_1$  and  $\omega_2$ . Two resultant molecular structures  $\omega'_1$  and  $\omega'_2$  are produced from the neighborhoods of  $\omega_1$  and  $\omega_2$ , respectively, i.e.  $\omega'_1 = N(\omega_1)$  and  $\omega'_2 = N(\omega_2)$ . The change is allowed if

$$PE_{\omega_1} + PE_{\omega_2} + KE_{\omega_1} + KE_{\omega_2} \geq PE_{\omega'_1} + PE_{\omega'_2}. \quad (31)$$

Set  $temp_3 = (PE_{\omega_1} + PE_{\omega_2} + KE_{\omega_1} + KE_{\omega_2}) - (PE_{\omega'_1} + PE_{\omega'_2})$ . If (31) holds,  $KE_{\omega'_1}$  and  $KE_{\omega'_2}$  are updated respectively by

$$\begin{aligned} KE_{\omega'_1} &= temp_3 \times r \quad \text{and} \\ KE_{\omega'_2} &= temp_3 \times (1 - r), \end{aligned}$$

where  $r$  is a random number generated uniformly between  $[0, 1]$ .  $numHit_{\omega_1}$  and  $numHit_{\omega_2}$  are increased by one, respectively.  $minHit_{\omega_1}$  will be equal to  $numHit_{\omega_1}$  if the new solution has better objective function value  $\omega_1$  has experienced.  $minHit_{\omega_2}$  is updated similarly. If (31) is not satisfied, the original molecular structures  $\omega_1$ ,  $\omega_2$  and  $PE_{\omega_1}$ ,  $PE_{\omega_2}$ ,  $KE_{\omega_1}$ ,  $KE_{\omega_2}$  are all kept.

4) *Synthesis*: A synthesis takes place when two molecules collide and fuse together. The energy change is generally vigorous. Assume that two existing molecules have  $\omega_1$  and  $\omega_2$ . They are combined to form  $\omega'$  by the following mechanism: For the  $i$ th variable in the  $\omega'$ , we introduce a random number  $u_i$  generated between  $[0, 1]$ . The  $i$ th variable of  $\omega'$  is updated by

$$\omega'(i) = \begin{cases} \omega_1(i) & \text{if } u_i > 0.5, \\ \omega_2(i) & \text{otherwise.} \end{cases} \quad (32)$$

Consider an example of four variables:

$$\underbrace{[2, 3.5, 6.3, 1.4]}_{\omega_1} + \underbrace{[1.8, 3.2, 6.6, 1.1]}_{\omega_2} \rightarrow \underbrace{[2, 3.5, 6.6, 1.4]}_{\omega'}$$

The change is accepted if

$$PE_{\omega_1} + PE_{\omega_2} + KE_{\omega_1} + KE_{\omega_2} \geq PE_{\omega'}, \quad (33)$$

is satisfied. Then  $KE_{\omega'}$  is updated by

$$KE_{\omega'} = PE_{\omega_1} + PE_{\omega_2} + KE_{\omega'_1} + KE_{\omega'_2} - PE_{\omega'}$$

and  $numHit'_{\omega}$  and  $minHit'_{\omega}$  are both set to zero. If (31) not satisfied, no change will happen.

#### E. The Overall Algorithm

The overall algorithm includes three stages: initialization, iterations, and the final stage. In each simulation run, we begin with the initialization, execute a preset number of iterations, and stop to output the best result.

In initialization, we set the values of algorithmic parameters, i.e.,  $PopSize$ ,  $InitialKE$ ,  $MoleColl$ ,  $KELossRate$ ,  $\alpha$ , and  $\beta$ . We create  $PopSize$  number of molecules randomly in the solution space and each of them are assigned with  $KE$  equal to  $InitialKE$ . We go through a certain number of iterations until a predefined number of function evaluations (FEs) is reached.

In each iteration, one elementary reaction will take place. We first decide whether it is a unimolecular or an inter-molecular collision by generating a random number  $z$  in  $[0, 1]$ . If  $z$  is larger than  $MoleColl$ , a unimolecular (i.e. on-wall ineffective collision or decomposition) will be triggered. Otherwise an inter-molecular collision (i.e. inter-molecular ineffective collision or synthesis) will happen. For a unimolecular reaction, we randomly select one molecule (e.g.  $\omega$ ) and further check the decomposition criterion  $minHit_{\omega} - numHit_{\omega} \geq \alpha$ , where  $\alpha$  is the preset system parameter. If the criterion is satisfied, then we have a decomposition. Otherwise, it will result in an on-wall ineffective collision. For the inter-molecular collision, we randomly select two molecules (e.g.  $\omega_1$  and  $\omega_2$ ) and then examine the synthesis criterion:  $KE_{\omega_1} \leq \beta$  and  $KE_{\omega_2} \leq \beta$ , where  $\beta$  is a preset system parameter constant. If so, we have a synthesis with  $\omega_1$  and  $\omega_2$ . Otherwise, an inter-molecular ineffective collision will happen.

At the end of each iteration, we record any newly found solution with better objective function value. When the maximum FEs have been reached, we output the best found result. More details of implementing CRO can be found in [22], [30], [31].

## IV. SIMULATION RESULTS

The simulation is implemented in the Visual C++ 6.0 environment executed on a 2.4 GHz Intel i5-2430M dual core personal computer with 4GB RAM. To illustrate the effectiveness of CRO, we test the performance of CRO on the IEEE 14-, 30-, and 57- bus standard systems [23].

The parameter values for different test cases are determined by trial-and-error and they are given in Table I. Note that the penalty coefficient  $\gamma_G$  for the power generated by the slack bus is set much larger than other penalty coefficients because we try to suppress the power produced at the slack bus. According to our experience, if all the coefficients have similar values, e.g. all set to one, the power generation at the slack bus will likely violate its limit. If the final solution computed by the algorithm does not violate any limits, the

TABLE I  
CRO PARAMETER SETTINGS

Variables	Bus 14	Bus 30	Bus 57
$PopSize$	3	5	3
$InitialKE$	500	1000	1000
$KELossRate$	0.2	0.2	0.2
$MoleColl$	0.2	0.2	0.2
$\alpha$	300	1000	300
$\beta$	0.005	0.0005	0.001
$\sigma^2$ for $Q_c$	0.0005	0.0005	0.0005
$\sigma^2$ for others	0.05	0.003	0.003
$\gamma_V$	1	1	1
$\gamma_G$	100000	100000	100000
$\gamma_Q$	1	1	1
$\gamma_I$	1	1	1

values of the coefficients will not affect the final result and this is true for the solutions generated by CRO shown later.

In the IEEE 14-bus test case, there are 14 buses including Bus 1 as the slack bus, 2, 3, 6, 8 as PV buses and the rest as PQ buses. There are three transformers between Buses 7-4, 9-4 and 6-5 and Buses 3, 6, 8 have shunt VAR compensators connected [23]. In the IEEE 30-bus test case, there are 30 buses: Bus 1 as the slack bus, Buses 2, 5, 8, 11, 13 as PV buses, and the rest as PQ buses. There are four transformers between Buses 9-6, 10-6, 12-4, and 27-28 and there are shunt VAR compenstors attached to Buses 10, 12, 15, 17, 20, 21, 23, 24, and 29. For the IEEE-57 bus system, there are 57 buses including Bus 1 as the slack bus, Bus 2, 3, 6, 8, 9, 12 as PV buses and the rest as PQ buses. Buses 18, 25, 53 have a shunt VAR compensator connected. Detailed data for the constants and various limitations can be found in [23].

We first discuss the results for the IEEE-30 test case. To the best of our knowledge, for the same settings and objective function for the OPF problem, there are only results comparing the best performance among different algorithms in the literature. We compare the best results with three other metaheuristics including BBO [21], PSO [20], and GA [21]. We repeat each simulation test 50 times. The algorithms used for comparison adopt the following FE limits as the stopping criterion: 16 000 for GA [18], 10 000 for PSO [20], and 2500 for BBO [21]. To show the effectiveness of CRO, the stopping criterion adopted for CRO is 2500 FEs. The best solution found is reported in Table II and those computed by other algorithms are obtained from [20]. The result generated by CRO does not violate any physical limits and constraints so that all the penalties are zero. Although CRO uses much fewer FEs than GA and PSO, CRO can produce the solution with the lowest cost among the four algorithms. The convergence curve of the corresponding simulation run is shown in Fig 2. We can see that CRO converges very fast and it can almost reach the best values after 1600 FEs (for all test cases). To be more complete, we also give the results of CRO for the average and the standard deviation in Table III.

TABLE II  
SOLUTIONS FOR THE IEEE 30-BUS TEST CASE

Variables	CRO	BBO	PSO	GA
$P_1$	1.76767	1.7758	1.7696	1.77594
$P_2$	0.48746	0.48701	0.4898	0.48722
$P_5$	0.21054	0.2128	0.213	0.21454
$P_8$	0.21743	0.2122	0.2119	0.20954
$P_{11}$	0.11711	0.1157	0.1197	0.11768
$P_{13}$	0.12067	0.12048	0.12	0.12052
$V_1$	1.1	1.1	1.0855	1.081
$V_2$	1.08717	1.071	1.0653	1.063
$V_5$	1.06173	1.04	1.0333	1.034
$V_8$	1.06833	1.045	1.0386	1.038
$V_{11}$	1.1	1.1	1.0848	1.1
$V_{13}$	1.1	1.0895	1.0512	1.055
$T_1$	1.05534	1.07	1.0233	1
$T_2$	0.915	0.97	0.9557	0.975
$T_3$	0.97619	1.08	0.9724	0.975
$T_4$	0.96139	1	0.9728	1
$Q_{10}$	0.0098	0.03	0.0335	0.001
$Q_{12}$	0.0441	0.02	0.022	0.007
$Q_{15}$	0.0353	0.05	0.0198	0.019
$Q_{17}$	0.0376	0.03	0.0315	0.024
$Q_{20}$	0.05	0.04	0.0454	0.015
$Q_{21}$	0.0365	0.05	0.0381	0.022
$Q_{23}$	0.0123	0.046	0.0398	0.047
$Q_{24}$	0.0381	0.045	0.05	0.047
$Q_{29}$	0.0063	0.03	0.0251	0.024
Cost (\$/hr.)	799.365	800.2803	800.41	800.805

TABLE IV  
SOLUTIONS FOR THE IEEE-14 AND IEEE-57 TEST CASES

14-Bus		57-Bus	
Variable	Value	Variable	Value
$P_1$	1	$P_1$	4.0799
$P_2$	0.5	$P_2$	1
$P_3$	0.35255	$P_3$	0.4
$P_6$	0.45	$P_6$	2.0024
$P_8$	0.32564	$P_8$	4.335
$V_1$	1.08647	$P_9$	0.468231
$V_2$	1.07084	$P_{12}$	0.6014
$V_3$	1.06085	$V_1$	1.1
$V_6$	1.05671	$V_2$	1.1
$V_8$	1.0565	$V_3$	1.09123
$T_1$	0.93111	$V_6$	1.1
$T_2$	1.09987	$V_8$	1.1
$T_3$	1.1	$V_9$	1.08472
$Q_3$	0.05	$V_{12}$	1.0678
$Q_6$	0.0196	$Q_{18}$	0.01287
$Q_8$	0.0055	$Q_{25}$	0.03891
		$Q_{53}$	0.05
Cost (\$/hr.)	763.137	Cost (\$/hr.)	4953.9

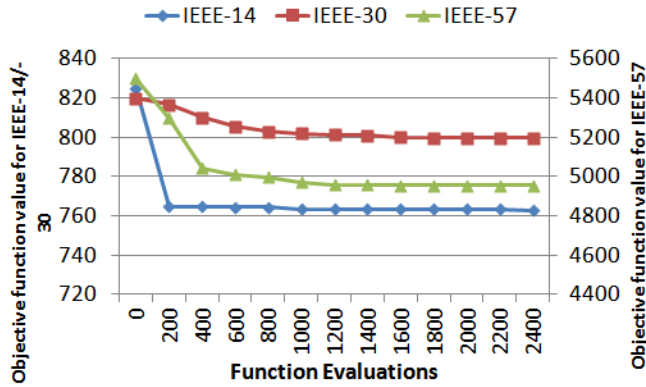


Fig. 2. Convergence Curve of the three test cases

TABLE III  
AVERAGE RESULTS AND THE STANDARD DEVIATIONS

	Average Cost (\$/hr.)	Standard deviation
IEEE-14	764.1737	1.14286
IEEE-30	799.8655	0.28366
IEEE-57	4956.547	4.87183

As there are no similar results computed by other algorithms for the IEEE-14 and -57 test cases in the literature, we give the results obtained by CRO for reference only. For these two test cases, the simulation settings for CRO are similar to those used in the IEEE-30 test case. The best results for the IEEE-14 and -57 test cases are given in Tables IV. None of these results violate any physical limits and constraints of the systems. The corresponding convergence curves are shown in Fig. 2. The average results and the standard deviations are also provided in Table III. With the small standard deviations, we can see that the results produced by CRO are very consistent in different simulation runs for all test cases.

## V. CONCLUSION

The OPF problem is one of the fundamental problems in power engineering. It is frequently used to determine the lowest cost generation settings in real-world power systems. This problem is in general non-convex and there is no known effective algorithm to solve the general OPF problem. Thus metaheuristics are usually employed to determine the global minimum. In this paper, we propose to use CRO to solve the OPF problem. CRO is a recently proposed general-purpose metaheuristic for optimization. It mimics the interactions of molecules with elementary reactions to locate the global minimum. CRO relies on the conservation of energy principle and the natural tendency that a reacting system tries to stabilize itself by transforming the resultants to the state with the lowest potential energy. We perform simulations on the standard IEEE-14, -30, and -57 benchmark test systems. We compare the performance of CRO with other algorithms on the IEEE-30 test case. Simulation results show that CRO can produce

the result with the lowest cost among all the tested algorithms. To be more complete, we also give the average and the best results for all three test cases. In the future, we will study the performance of CRO on solving other variants of the OPF problem, e.g. multiobjective function considering the generation cost and voltage deviations (differences between the resultant voltage and the mean value), and the security-constrained and contingency-constrained OPF problems.

#### ACKNOWLEDGMENT

This research is supported in part by the Collaborative Research Fund of the Research Grants Council of Hong Kong, under Grant No. HKU10/CRF/10. A.Y.S. Lam is also supported in part by the Croucher Foundation Research Fellowship.

#### REFERENCES

- [1] H. W. Dommel and W. F. Tinney, "Optimal power flow solutions," *IEEE Trans. Power App. Syst.*, vol. PAS-87, no. 10, pp. 1866–1876, Oct. 1968.
- [2] J. A. Momoh, R. J. Koessler, M. S. Bond, B. Stott, D. Sun, A. Papalexopoulos, and P. Ristanovic, "Challenges to optimal power flow," *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 444–455, Feb. 1997.
- [3] R. D. Christie, B. F. Wollenberg, and I. Wangenstein, "Transmission management in the deregulated environment," *Proc. IEEE*, vol. 88, no. 2, pp. 170–195, Feb. 2000.
- [4] P. P. Varaiya, F. F. Wu, and J. W. Bialek, "Smart operation of smart grid: Risk-limiting dispatch," *Proc. IEEE*, vol. 99, pp. 40–57, 2011.
- [5] O. Alsac and B. Stott, "Optimal load flow with steady state security," *IEEE Trans. Power App. Syst.*, vol. PAS-93, no. 3, pp. 745–751, May/June 1974.
- [6] R. R. Shoults and D. T. Sun, "Optimal power flow based upon p-q decomposition," *IEEE Trans. Power App. Syst.*, vol. PAS-101, no. 2, pp. 397–405, Feb. 1982.
- [7] B. Stott and E. Hobson, "Power system security control calculation using linear programming, part i," *IEEE Trans. Power App. Syst.*, vol. PAS-97, no. 5, pp. 1713–1720, Sep/Oct. 1978.
- [8] —, "Power system security control calculation using linear programming, part ii," *IEEE Trans. Power App. Syst.*, vol. PAS-97, no. 5, pp. 1721–1731, Sep/Oct. 1978.
- [9] B. Stott and J. L. Marinho, "Linear programming for power system network security applications," *IEEE Trans. Power App. Syst.*, vol. PAS-98, no. 3, pp. 837–848, May/June 1979.
- [10] R. Mota-Palomino and V. H. Quintana, "A penalty function-linear programming method for solving power system constrained economic operation problems," *IEEE Trans. Power App. Syst.*, vol. PAS-103, no. 6, pp. 1414–1442, Jun. 1984.
- [11] R. C. Burchett, H. H. Happ, and D. R. Vierath, "Quadratically convergent optimal power flow," *IEEE Trans. Power App. Syst.*, vol. PAS-103, no. 11, pp. 3267–3276, Nov. 1984.
- [12] K. Aoki, A. Nishikori, and R. T. Yokoyama, "Constrained load flow using recursive quadratic programming," *IEEE Trans. Power Syst.*, vol. 2, no. 1, pp. 8–16, Feb. 1987.
- [13] D. I. Sun, B. Ashley, B. Brewer, A. Hughes, and W. Tinney, "Optimal power flow by newton approach," *IEEE Trans. Power App. Syst.*, vol. PAS-103, no. 10, pp. 2864–2875, Oct. 1984.
- [14] X. Yan and V. H. Quintana, "Improving an interior point based opf by dynamic adjustments of step sizes and tolerances," *IEEE Trans. Power Syst.*, vol. 14, no. 2, pp. 709–717, May 1999.
- [15] J. Kennedy, "The particle swarm: social adaptation of knowledge," in *Proc. IEEE Int. Conf. Evol. Comput.*, 1997, pp. 303–308.
- [16] M. J. A., R. Adapa, and M. E. El-Hawary, "A review of selected optimal power flow literature to 1993. i. nonlinear and quadratic programming approaches," *IEEE Trans. Power Syst.*, vol. 14, no. 1, pp. 96–104, Feb. 1999.
- [17] Q. H. Wu, Y. J. Cao, and J. Y. Wen, "Optimal reactive power dispatch using an adaptive genetic algorithm," *Int. J. Elect. Power Energy Syst.*, vol. 20, no. 8, pp. 563–569, Nov. 1998.
- [18] A. G. Bakirtzis, P. N. Biskas, C. E. Zoumas, and V. Petridis, "Optimal power flow by enhanced genetic algorithm," *IEEE Trans. Power Syst.*, vol. 17, no. 2, pp. 229–236, May 2002.
- [19] K. Vaisakh and L. R. Srinivas, "Differential evolution approach for optimal power flow solution," *J. Theor. Appl. Inf. Technol.*, vol. 4, pp. 261–268, 2008.
- [20] M. A. Abido, "Optimal power flow using particle swarm optimization," *Electrical Power and Energy Systems*, vol. 24, pp. 563–571, 2002.
- [21] A. Bhattacharya and P. K. Chattopadhyay, "Biogeography-based optimization for solution of optimal power flow problem," in *Proc. International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology*, Chaing Mai, May 2010, pp. 435–439.
- [22] A. Y. S. Lam and V. O. K. Li, "Chemical-reaction-inspired metaheuristic for optimization," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 381–399, Jun. 2010.
- [23] University of washington, power systems test case archive. [Online]. Available: <http://www.ee.washington.edu/research/pstca>
- [24] J. D. Glover, M. S. Sarma, and T. J. Overbye, *Power System Analysis and Design*, 5th ed. Stamford, CT: Cengage Learning, 2012.
- [25] C. R. Paul, Ed., *Fundamentals of electric circuit analysis*. New York, NY: John Wiley & Sons, 2001.
- [26] A. Wood and B. Wollenberg, *Power Generation, Operation, and Control*, 2nd ed. New York, NY: John Wiley & Sons, 1996.
- [27] A. Y. S. Lam, J. Xu, and V. O. K. Li, "Chemical reaction optimization for population transition in peer-to-peer live streaming," in *Proc. IEEE Congress on Evol. Comput.*, Barcelona, Spain, Jul. 2010.
- [28] J. J. Q. Yu, A. Y. S. Lam, and V. O. K. Li, "Evolutionary artificial neural network based on chemical reaction optimization," in *Proc. IEEE Congress on Evol. Comput.*, New Orleans, LA, Jun. 2011.
- [29] B. Pan, A. Y. S. Lam, and V. O. K. Li, "Network coding optimization based on chemical reaction optimization," in *Proc. IEEE Global Comm. Conf.*, Houston, TX, Dec. 2011.
- [30] A. Y. S. Lam and V. O. K. Li, "Chemical reaction optimization: A tutorial," vol. 4, no. 1, pp. 3–17, Mar. 2012.
- [31] A. Y. S. Lam, V. O. K. Li, and J. J. Q. Yu, "Real-coded chemical reaction optimization," *IEEE Trans. Evol. Comput.*, To appear.
- [32] J. Ronkkonen, S. Kukkonen, and K. V. Price, "Real-parameter optimization with differential evolution," in *Proc. IEEE Congress on Evol. Comput.*, Sep. 2005, pp. 506–513.