Artificial bee colony algorithm for multi-objective optimal power flow

Man Ding\textsuperscript{a}, Hanning Chen\textsuperscript{b,}\textsuperscript{*}, Na Lin\textsuperscript{c}, Shikai Jing\textsuperscript{b}, Fang Liu\textsuperscript{b}, Xiaodan Liang\textsuperscript{b}, Wei Liu\textsuperscript{d}

\textsuperscript{a} School of Architecture and Art Design, Hebei University of Technology, Tianjin 300130, China
\textsuperscript{b} School of Computer Science and Software, Tianjin Polytechnic University, Tianjin 300387, China
\textsuperscript{c} Beijing Shenzhou Aerospace Software Technology Co. Ltd., Beijing 110000, China
\textsuperscript{d} College of Information and Technology, Jilin Normal University, Siping 136000, China

Received 11 October 2016; revised 25 December 2016; accepted 7 January 2017
Available online 26 January 2017

KEYWORDS
Artificial bee colony algorithm; Life-cycle evolving model; Optimal power flow; Multi-objective optimization

Abstract This paper proposes a novel artificial bee colony algorithm with dynamic population (ABC-DP), which synergizes the idea of extended life-cycle evolving model to balance the exploration and exploitation tradeoff. The proposed ABC-DP is a more bee-colony-realistic model that the bee can reproduce and die dynamically throughout the foraging process and population size varies as the algorithm runs. ABC-DP is then used for solving the optimal power flow (OPF) problem in power systems that considers the cost, loss, and emission impacts as the objective functions. The 30-bus IEEE test system is presented to illustrate the application of the proposed algorithm. The simulation results, which are also compared to nondominated sorting genetic algorithm II (NSGAII) and multi-objective ABC (MOABC), are presented to illustrate the effectiveness and robustness of the proposed method.

© 2017 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

In many fields of science and engineering, there are always multiple conflicting objectives, which are formulated as multi-objective (MO) optimization problems in order to minimize or maximize these conflicting objective functions simultaneously. In MO optimization domain, the set of Pareto optimal solutions, namely several optimal solutions with different trade-offs in the objective space, is called the Pareto optimal front (Fonseca and Fleming, 1998; Cruz et al., 2014). Optimal power flow (OPF) is one of the most important MO problems in power system. The main goal of OPF is to find the optimal adjustments of the control variables to minimize the selected objective function while satisfying various physical and operational constraints imposed by equipment and network limitations (Kumari and Maheswarapu, 2010). Since the real power generation levels and voltage magnitudes are continuous variables whereas the transformer winding
ratios and shunt capacitors are discrete variables, the OPF problem is considered as a non-linear multi-modal optimization problem with a combination of the discrete and continuous variables (Abou El Ela et al., 2010).

Many mathematical models and conventional techniques, such as gradient-based optimization algorithms, linear programming, interior point method, and Newton method, have been applied to solve the OPF problem (Momoh et al., 1999a,b). However, these methods suffer from severe limitations in handling non-linear, discrete and continuous functions, and constraints. In order to overcome the limitations of classical optimization techniques, a wide variety of the heuristic methods have been proposed to solve the OPF problem, such as genetic algorithm (GA) (Lai et al., 1997), tabu search (TS) (Abido, 2002), differential evolution (DE) algorithm (Sayah and Zehar, 2008), and biogeography based optimization (BBO) (Roy et al., 2010). The reported results are promising and encouraging for further research in this field (Abou El Ela et al., 2010). However, all the mentioned heuristic mathematical techniques have some drawbacks such as being trapped in local optima or each of them is only suitable for solving a specific objective function in the OPF problem (AlRashidi and El-Hawary, 2007).

Swarm intelligence (SI) is an innovative artificial intelligence technique for solving complex optimization problems (Xu et al., 2013). Among them, artificial bee colony algorithm (ABC) is a relatively new optimization technique which simulates the intelligent foraging behavior of a honeybee swarm (Ma et al., 2013; Eberchart and Kennedy, 1995). Recently, two multi-objective approaches based on ABC model were proposed in (Passino, 2002; Gao and Liu, 2011). However, compared to the huge in-depth studies of other multi-objective evolutionary and swarm intelligence algorithms, such as nondominated sorting genetic algorithm II (NSGAII) (Deb et al., 2002), strength Pareto evolutionary algorithm (SPEA2) (Zitzler et al., 2001), and multi-objective particle swarm optimization (MOPSO) (Coello Coello and Pulido, 2004), how to improve the diversity of swarm or overcome the local convergence of multi-objective ABC (MOABC) is still a challenging to the researchers in MO optimization domain.

In this paper, a novel artificial bee colony algorithm with dynamic population (ABC-DP) is proposed to synergize the idea of extended life-cycle evolving model, which can balance the exploration and exploitation tradeoff in artificial bee colony foraging process. The proposed ABC-DP is a more bee-colony-realistic model that the bee can reproduce and die dynamically throughout the foraging process and population size varies as the algorithm runs. By incorporating this new degree of complexity, ABC-DP can accommodate a considerable potential for solving complex MO problems. Then we applied ABC-DP to solve two and three objective OPF cases considering the cost, loss, and emission impacts as the objective functions respectively on the 30-bus IEEE test system. The simulation results, on both benchmarks and OPF cases, prove that ABC-DP has better optimization performance than the NSGA-II and MOABC algorithms.

The rest of the paper is organized as follows. Section 2 first gives a review of the original ABC algorithm. Section 3 proposes the novel ABC-DP algorithm with the life-cycle model. In Section 4, the multi-objective OPF problem is formulated, and then the implementation of the ABC-DP on OPF is presented. Simulation results and comparison with other algorithms are given in Section 5. Finally, Section 6 outlines the conclusions.

2. The original artificial bee colony algorithm

From Fig. 1, we can understand the basic behavior characteristics of bee colony foraging behaviors better. Assume that there are two discovered food sources: A and B. At the very beginning, a potential bee forager will start as an unemployed bee. That bee will have no knowledge about the food sources around the nest.

There are two possible options for such a bee:

i. It can be a scout and starts searching around the nest spontaneously for a food due to some internal motivation or possible external clue (‘S’ in Fig. 1).

ii. It can be a recruit after watching the waggle dances and starts searching for a food source (‘R’ in Fig. 1).

After finding the food source, the bee utilizes its own capability to memorize the location and then immediately starts exploiting it. Hence, the bee will become an “employed forager”. The foraging bee takes a load of nectar from the source and returns to the hive, unloading the nectar to a food store. After unloading the food, the bee has the following options:

iii. It might become an uncommitted follower after abandoning the food source (UF).

iv. It might dance and then recruit nest mates before returning to the same food source (EF1).

v. It might continue to forage at the food source without recruiting after bees (EF2).

It is important to note that not all bees start foraging simultaneously. The experiments confirmed that new bees begin after surviving from the hive.
foraging at a rate proportional to the difference between the eventual total number of bees and the number presently foraging. In mathematical terms, the original ABC algorithm can be formulated as follows.

In the initialization phase, the ABC algorithm generates a randomly distributed initial food source positions of $SN$ solutions, where $SN$ denotes the size of employed bees or onlooker bees. Each solution $x_i$ ($i = 1, 2, \ldots, SN$) is a $D$-dimensional vector. Here, $D$ is the number of optimization parameters. And then evaluate each nectar amount $fit_i$. In ABC model, nectar amount is the solution value of benchmark function or real-world problem.

In the employed bees’ phase, each employed bee finds a new food source $j$ in the neighborhood of its current source $x_i$. The new food source is calculated using the following expression:

$$V_i = x_i + \phi_i (x_j - x_i)$$

(1)

where $k \in (1, 2, \ldots, SN)$ and $j \in (1, 2, \ldots, D)$ are randomly chosen indexes, and $k$ has to be different from $i$. $\phi_i$ is a random number between $[-1, 1]$. And then employed bee compares the new one against the current solution and memorizes the better one by means of a greedy selection mechanism.

In the onlooker bees’ phase, each onlooker chooses a food source with a probability which related to the nectar amount (fitness) of a food source shared by employed bees. Probability is calculated using the following expression:

$$P_i = \frac{fit_i^{\sum_{j=1}^{SN} fit_j}}{\sum_{j=1}^{SN} fit_j}$$

(2)

In the scout bee phase, if a food source cannot be improved through a predetermined cycles, called “limit”, it is removed from the population and the employed bee of that food source becomes scout. The scout bee finds a new random food source position using the equation below:

$$x_i' = x_i^{min} + rand [0, 1] (x_i^{max} - x_i^{min})$$

(3)

where $x_i^{min}$ and $x_i^{max}$ are lower and upper bounds of parameter $j$, respectively.

These steps are repeated through a predetermined number of cycles, or until a termination criterion is satisfied. The pseudo code of original ABC algorithm is illustrated in Fig. 2.

3. The dynamic population ABC algorithm with life-cycle model

In biology, the term life-cycle refers to the various phases an individual passes through from birth to maturity, reproduction, and death. This process often leads to drastic transformations of the individuals with stage-specific adaptations to a particular environment. Inspired by this phenomenon, this work assumes that the computational life-cycle model of bee colony has five major stages, namely the born, forage, reproduction, death, and migration. The bee state transition diagram is shown in Fig. 3.

$$N_i(t + 1) = \begin{cases} N_i(t) + 1 & \text{if } fit(X_i^{t+1}) < fit(X_i^t) \\ N_i(t) - 1 & \text{else} \end{cases}$$

(4)

where $fit(X_i^t)$ is the fitness of the $i$th bee $x_i$ at time $t$ for a minimum problem, $N_i(t)$ is the nutrient obtained by the $i$th bee $x_i$ at time $t$. In initialization stage, nutrients of all bees are zero. For each $x_i$ at onlooker phase, if the new position is better than the last one, it is regarded that the bee will gain nutrient from the environment and the nutrient is added by one. Otherwise, it loses nutrient in the foraging process and its nutrient is reduced by one. Then the information rate $P_i$ deciding to reproduce or die for each bee $X_i$ at time $t$ is computed as:

$$H_i(t) = \frac{fit(X_i^t) - fit^{new}_i}{fit^{best}_i - fit^{worst}_i}$$

(5)

$$P_i = \eta \frac{H_i(t)}{\sum_{j=1}^{SN} H_i(t)} + (1 - \eta) \frac{N_i(t)}{\sum_{j=1}^{SN} N_i(t)}, \eta \in [0, 1]$$

(6)

where $fit^{worst}_i$ and $fit^{best}_i$ are the current worst and best fitness of the whole bee colony at time $t$.

In the foraging process, if the bee $X_i$ converts enough information rate $P_i$ as:

$$F_i^t \geq \max \left( F_{reproduce}, F_{reproduce} + \frac{(S - S)}{F_{adapt}} \right)$$

(7)

it will reproduce an offspring by using best-so-far solution information in search equation of employed and onlooker bees steps based on the works of:

$$x_{new} = x_i + \phi (x_{best} - x_i)$$

(8)

where $x_{new}$ is the new offspring, $x_i$ is the $i$th bee, $x_{best}$ is best individual of current colony, $j$ is a randomly chosen indexes; $\phi$ is a random number in range $[-1, 1]$.

If the bee enters bad environment, and its information rate drops to a certain threshold as:

$$F_i^t < \min \left( 0, \frac{(S - S)}{F_{adapt}} \right)$$

(9)

The pseudo-code of the proposed ABC-DP is listed in Table 1.

It will die and be eliminated from the population. Here $S$ is the initial population size and $S'$ is the current colony size, $F_{split}$ and $F_{adapt}$ are two control parameters used to adjust the bee reproduction and death criterions.

It should be noticed that the population size will increase by one if a bee reproduces and reduce by one if it dies. As a result, the population size dynamically varies in the foraging process. At the beginning of the foraging process, the bee will reproduce when its information rate is larger than $F_{reproduce}$. In the course of bee foraging, in order to avoid the population size becoming too large or too small, the reproduction and death criteria, namely Eqs. (7) and (9), are delicately designed: if $S'$ is larger than $S$, for each $F_{adapt}$ of their differences, the reproduce threshold value will increase by one; if $S'$ is smaller than $S$, for each $F_{adapt}$ of their differences, the death threshold value will decrease by one. The strategy is also consistent with the natural law: if the population is too crowded, the competition between the individuals will increase and death becomes common; if the population is small, the individuals are easier to survive and reproduce. When the nutrient of a bee is less than zero, but it has not died yet, it could migrate with a probability as a scout bee. A random number is generated and if the number is less than migration probability $P_m$, it will migrate and move to a randomly produced position. Then nutrient of this bee will be reset to zero.
4. The optimal power flow problem formulation

In this paper, the OPF problem is to minimize three competing objective functions, fuel cost, emission, and real power loss, while satisfying several equality and inequality constraints. Generally the problem is formulated as follows.

\[
\begin{align*}
\min f(x, u) \\
\text{s.t. } g(x, u) &= 0 \\
h(x, u) &= 0
\end{align*}
\]

where \(f\) is the optimization objective function, \(g\) is a set of constrain equations, and \(h\) is a set of formulated constrain in equations, \(u\) is a set of the control variables such as the generator real power output \(P_{G1}\), \(x\) is the vector of dependent variables such as the slack bus power \(P_{G1}\), the load bus voltage \(V_{L}\), generator reactive power outputs \(Q_{G}\), and the apparent power flow \(S_k\). \(x\) can be expressed as:

\[
x^T = [P_{G1}, V_{L1}, ..., V_{LN}, Q_{G1}, ..., Q_{GN}, S_1, ..., S_N]
\]
The objective function is generalized as follows:

\[ F = f + \sum_{i \in N_G} \lambda_{P_i}(V_i - V_{i_{\text{min}}})^2 + \sum_{i \in N_C} \lambda_{Q_i}(Q_i - Q_{i_{\text{lim}}})^2 + \sum_{i \in N_E} \lambda_S(|S_i| - S_{i_{\text{lim}}})^2 \]

where \( \lambda_{P_i}, \lambda_{Q_i}, \) and \( \lambda_S \) are the penalty factors. \( V_{i_{\text{lim}}} \) and \( Q_{i_{\text{lim}}} \) are defined as:

\[ V_{i_{\text{lim}}} \begin{cases} V_i^{\text{max}} & \text{if } V_i > V_i^{\text{max}} \\ V_i^{\text{min}} & \text{if } V_i < V_i^{\text{min}} \end{cases} \]

\[ Q_{i_{\text{lim}}} \begin{cases} Q_i^{\text{max}} & \text{if } Q_i > Q_i^{\text{max}} \\ Q_i^{\text{min}} & \text{if } Q_i < Q_i^{\text{min}} \end{cases} \]

The quality constraints \( g(x, u) \) are the nonlinear power flow equations which are formulated as below:

\[ 0 = P_G - P_D - \sum_{j \in N} V_j(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad i \in N_0 \]

\[ 0 = Q_G - Q_D - \sum_{j \in N} V_j(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad i \in N_{pq} \]

The equality constraints \( h(x, u) \) are limits of control variables and state variables which can be formulated as:

\[ P_{G_i}^{\text{min}} \leq P_{G_i} \leq P_{G_i}^{\text{max}} \quad i \in N_G \]

\[ Q_{G_i}^{\text{min}} \leq Q_{G_i} \leq Q_{G_i}^{\text{max}} \quad i \in N_G \]

\[ T_{k}^{\text{min}} \leq T_k \leq T_{k}^{\text{max}} \quad k \in N_T \]

\[ V_{i_{\text{lim}}}^{\text{min}} \leq V_i \leq V_{i_{\text{lim}}}^{\text{max}} \quad i \in N_B \]

\[ |S_k| \leq S_{k_{\text{lim}}}^{\text{max}} \quad k \in N_E \]

To solve non-linear constrained optimization problems, the most common method uses penalty function to transform a constrained optimization problem into an unconstrained one. The objective function is generalized as follows:

\[ f_{\text{cost}} = \sum_{i = 1}^{N_j} f_i(a_iP_{G_i}^2 + b_iP_{G_i} + c_i) \]

where \( a_i, b_i, \) and \( c_i \) are the fuel cost coefficients of the \( i \)th generator, \( P_{G_i} \) is real power output of the \( i \)th generator.

### 4.1. Minimization of total fuel cost

This objective function is to minimize the total fuel cost \( f_{\text{cost}} \) of the system. The fuel cost curves of the thermal generators are modeled as a quadratic cost curves and can be represented as follows:

\[ f_{\text{cost}} = \sum_{i = 1}^{N_j} f_i(a_iP_{G_i}^2 + b_iP_{G_i} + c_i) \]

where \( a_i, b_i, \) and \( c_i \) are the fuel cost coefficients of the \( i \)th generator, \( P_{G_i} \) is real power output of the \( i \)th generator.

### 4.2. Minimization of total power losses

The power flow solution gives all bus voltage magnitudes and angles. Then, the total MW active power loss in a transmission network can be described as follows:

\[ f_{\text{loss}} = \sum_{k = 1}^{N_l} g_k(V_i^2 + V_j^2 - 2V_iV_j\cos(\delta_i - \delta_j)) \]

where \( N_l \) is the number of transmission lines, \( V_i \) and \( V_j \) are the voltage magnitudes at the \( i \)th bus and \( j \)th bus, respectively; \( \delta_i \) and \( \delta_j \) are the voltage angles at the \( i \)th bus and the \( j \)th bus, respectively.
Table 2  Characteristics of the generation units.

|   | G1       | G2       | G3       | G4       | G5       | G6       |
|---|----------|----------|----------|----------|----------|----------|
| PCmax(MW) | 150      | 150      | 150      | 150      | 150      | 150      |
| PCmin(MW)  | 5        | 5        | 5        | 5        | 5        | 5        |
| Cost coefficients |  |  |  |  |  |  |
| a         | 10       | 10       | 20       | 10       | 20       | 10       |
| b         | 200      | 150      | 180      | 100      | 180      | 150      |
| c         | 100      | 120      | 40       | 60       | 40       | 100      |
| Emission coefficients |  |  |  |  |  |  |
| a         | 4.091    | 2.543    | 4.258    | 5.326    | 4.258    | 6.131    |
| b         | −5.554   | −6.047   | −5.094   | −3.550   | −5.094   | −5.555   |
| c         | 6.490    | 5.638    | 4.586    | 3.380    | 4.586    | 5.151    |
| k         | 2.0e−4   | 5.0e−4   | 1.0e−6   | 2.0e−3   | 1.0e−6   | 1.0e−5   |
| λ         | 2.857    | 3.333    | 8.000    | 2.000    | 8.000    | 6.667    |

Figure 4  Pareto fronts obtained by ABC-DP, MOABC, and NSGA-II on Fuel cost – Emission-Loss ($f_1$–$f_2$–$f_3$). (a) ABC-DP, (b) MOABC and (c) NSGA-II.

Table 2  Characteristics of the generation units.

|   | G1       | G2       | G3       | G4       | G5       | G6       |
|---|----------|----------|----------|----------|----------|----------|
| Generator limits |  |  |  |  |  |  |
| PCmax(MW) | 150      | 150      | 150      | 150      | 150      | 150      |
| PCmin(MW)  | 5        | 5        | 5        | 5        | 5        | 5        |
| Cost coefficients |  |  |  |  |  |  |
| a         | 10       | 10       | 20       | 10       | 20       | 10       |
| b         | 200      | 150      | 180      | 100      | 180      | 150      |
| c         | 100      | 120      | 40       | 60       | 40       | 100      |
| Emission coefficients |  |  |  |  |  |  |
| a         | 4.091    | 2.543    | 4.258    | 5.326    | 4.258    | 6.131    |
| b         | −5.554   | −6.047   | −5.094   | −3.550   | −5.094   | −5.555   |
| c         | 6.490    | 5.638    | 4.586    | 3.380    | 4.586    | 5.151    |
| k         | 2.0e−4   | 5.0e−4   | 1.0e−6   | 2.0e−3   | 1.0e−6   | 1.0e−5   |
| λ         | 2.857    | 3.333    | 8.000    | 2.000    | 8.000    | 6.667    |

Figure 4  Pareto fronts obtained by ABC-DP, MOABC, and NSGA-II on Fuel cost – Emission-Loss ($f_1$–$f_2$–$f_3$). (a) ABC-DP, (b) MOABC and (c) NSGA-II.
4.3. Total emission cost minimization

In this paper, two important types of emission gasses, namely, sulfur oxides SOx and nitrogen oxides NOx, are taken as the pollutant gasses. The emission gasses generated by each generating unit may be approximated by a combination of a quadratic and an exponential function of the generator active power output. Here, the total emission cost is defined as below:

\[ f_{\text{emission}} = \sum_{i=1}^{N_g} (a_i + b_i P_{Gi} + c_i P_{Gi}^2) \]

where \( f_{\text{emission}} \) is the total emission cost (ton/h) and \( a_i, b_i \) and \( c_i \) are the emission coefficients of the \( i \)th unit.

5. Results

In order to verify the proposed approach, the IEEE 30-bus system is used as the test systems with ABC-DP, MOABC, and NSGA-II algorithms. The IEEE 30 bus system data are given in (Alsac and Stott, 1974). The active power generation limits are listed in Table 2. The limits of generator buses and load buses are between 0.95–1.1 p.u, and 0.9–1.05 p.u, respectively. The lower and upper limits of transformer taps are 0.9 p.u. and 1.05 p.u., respectively, and the step size is 0.01 p.u.

Experiments were conducted with HMOABC, MOABC, and the nondominated sorting genetic algorithm II (NSGA-II). The NSGA-II algorithm uses Simulated Binary Crossover (SBX) and Polynomial crossover (Deb et al., 2002). We use a population size of 100. Crossover probability \( pc = 0.9 \) and mutation probability is \( pm = 1/n \), where \( n \) is the number of decision variables.

For the MOABC, as described in (Zou et al., 2011), a colony size of 50, archive size \( A = 100 \) was adopted. The ABC-DP algorithm parameters were set as follows: the number of species \( K \) is set at 5, the colony size and archive size is \( N = 10 \), CR = 0.1, and \( A = 40 \), respectively. In the experiment, in order to compare the different algorithms with a fair time measure, the number of function evaluations (FEs) is used for the termination criterion.

In this simulation, three competing objectives are optimized simultaneously by the proposed algorithm and the obtained Pareto-optimal fronts are shown in Fig. 4. Table 3 shows the minimum values for each objective in the three-dimensional Pareto front (\( f_1-f_2-f_3 \)) (Table 4).

It is clear that cost, emission and loss cannot be further improved without degrading the other two related optimized objectives. Fig. 4 clearly shows the relationships among all presented objective functions. Between the obtained Pareto-optimal solutions, it is necessary to choose one of them as a best compromise for implementation.

To directly analysis the population distribution of ABC-DP, MOABC and NSGA-II, the diversity metric is employed, which measures the extent of spread achieved among the obtained solutions. This metric is defined as:

\[
\eta = \frac{d_i + d_i + \sum_{i=1}^{N-1} d_i - \bar{d}}{d_i + d_i + (N - 1)\bar{d}} \tag{22}
\]

where \( d_i \) is the Euclidean distance between consecutive solutions in the obtained non-dominated set of solutions and \( N \) is the number of non-dominated solutions obtained by an
algorithm. $d$ is the average value of these distances. $d_f$ and $d_l$ are the Euclidean distances between the extreme solutions and the boundary solutions of the obtained non-dominated set.

It can be proved that the proposed method is giving well distributed Pareto-optimal front for the three-objective OPF optimization. The results confirm that the ABC-DP algorithm is an impressive tool for solving the complex multi-objective optimization problems where multiple Pareto-optimal solutions can be obtained in a single run.

Here, we give a brief analysis on algorithm complexity of the proposed algorithm and other two compared algorithms. Assuming that the computation cost of one individual in the ABC-DP is $Cost_a$, $S'$ is the current population size. $D$ is the problem dimension, then, the total computation cost of ABC-DP for one generation is $S' * Cost_a$. However, due to its dynamical population, the time complexity of this procedure can be roughly estimated as $O(S)$. Through directly evaluating the algorithmic time response on different combinations of objectives as shown in Fig. 5, we can observe that ABC-DP takes the less computing time in most cases. This can be explained that by the life-cycle strategy, the population size of the ABC-DP can be dynamically adaptive. Hence, the proposed ABC-DP algorithms have the potential to solve complex real-world problems.

### 6. Conclusion

In this paper, different multi-objectives, which consider the cost, loss, and emission impacts, for OPF problem were formed. These multi-objectives were solved by the proposed ABC-DP. The proposed ABC-DP model extends original artificial bee colony (ABC) algorithm to synergize the idea of extended life-cycle evolving model, which can balance the exploration and exploitation tradeoff in artificial bee colony foraging process. The simulation studies, which conduct on 30-bus IEEE test system, show that the ABC-DP obtains better distributed Pareto optimal solutions than NSGA-II method in terms of optimization accuracy and convergence robust.

In this paper, we only apply the proposed ABC-DP on the optimal power flow problem of the 30-bus IEEE test system, which is a widely adopted simulation model. In the future work, we will analyze the more complex simulation model (e.g. 118-bus IEEE test system) and some virtual power flow system.

### Acknowledgements

This research is partially supported by National Natural Science Foundation of China and Grant 61305082, 51378350, 5157518, 61602343 and 51607122.

### References

Abido, M.A., 2002. Optimal power flow using tabu search algorithm. Electr. Power Compon. Syst. 30, 469–483.

Abou El Ela, A.A., Abido, M.A., Spea, S.R., 2010. Optimal power flow using differential evolution algorithm. Electr. Power Syst. Res. 80, 878–885.

AlRashidi, M.R., El-Hawary, M.E., 2007. Hybrid particle swarm optimization approach for solving the discrete OPF problem considering the valve loading effects. IEEE Trans. Power Syst. 22 (4), 2030–2038.

Aliev, O., Stott, B., 1974. Optimal load flow with steady-state security. IEEE Trans. Power Apparatus Syst. 93 (3), 745–751.

Coello Coello, C.A., Pulido, G.T., 2004. Maximo salazar lechuga, handling multiple objectives with particle swarm optimization. IEEE Trans. Evol. Comput. 8 (3), 256–279.

Cruz, L., Fernandez, E., Gomez, C., Rivera, G., Perez, F., 2014. Many-objective portfolio optimization of interdependent projects with ‘a priori’ incorporation of decision-maker preferences. Appl. Math. Inf. Sci. 8 (4), 1517–1531.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6 (2), 182–197.

Eberhart, R.C., Kennedy, J., 1995. A new optimizer using particle swarm theory. In: Proceeding of the 6th international symposium on Micromachine and Human Science. Nagoya, Japan. pp. 39–43.

Fonseca, C.M., Fleming, P.J., 1998. Multiobjective optimization and multiple constraint handling with evolutionary algorithms. IEEE Trans. Syst. Man Cybern. 28, 26–37.

Gao, W., Liu, S., 2011. Improved artificial bee colony algorithm for global optimization. Inf. Process. Lett. 111, 871–882.

Kumari, M.S., Maheswarapu, S., 2010. Enhanced genetic algorithm based computation technique for multi-objective optimal power flow solution. Electr. Power Energy Syst. 32, 736–742.

Lai, L.L., Ma, J.T., Yokoyama, R., Zhao, M., 1997. Improved genetic algorithm for optimal power flow under both normal and contingent operation states. Electr. Power Energy Syst. 19, 287–292.

Ma, L.B., Hu, K.Y., Zhu, Y.L., 2013. Computational evolution of social behavior in bacterial colony optimization model. J. Pure Appl. Microbiol.

Momoh, J.A., El-Hawary, M.E., Adapa, R., 1999a. A review of selected optimal power flow literature to 1993. Part I: nonlinear and quadratic programming approaches. IEEE Trans. Power Syst. 14, 96–104.

Momoh, J.A., El-Hawary, M.E., Adapa, R., 1999b. A review of selected optimal power flow literature to 1993. Part II: Newton, linear programming and interior point methods. IEEE Trans. Power Syst. 14, 105–111.

Passino, K.M., 2002. Biomimicry of bacterial foraging for distributed optimization and control. IEEE. Control Syst. Mag. 22, 52–67.

Roy, P.K., Ghoshal, S., Thakur, P.S.S., 2010. Biogeography based optimization for multiconstraint optimal power flow with emission and non-smooth cost function. Expert Syst. Appl. 37, 8221–8228.

Sayah, S., Zehar, K., 2008. Modified differential evolution algorithm for optimal power flow with non-smooth cost functions. Energy Convers. Manage. 49, 3036–3042.

Xu, J.M., Zhang, M.J., Cai, Y.G., 2013. Cultural ant algorithm for continuous optimization problems. Appl. Math. Inf. Sci. 7 (2), 705–710.

Zitzler, E., Laumanns, M., Thiele, L., 2002. SPEA2: Improving the strength pareto evolutionary algorithm. In: Proc. EUROGEN 2001: Evolutionary Methods Design Optimization Control Appl. Ind. Problems, Athens, Greece. pp. 95–100.

Zou, W.P., Chen, H.N., Zhua, Y.L., Zhang, B.W., 2011. Solving multiobjective optimization problems using artificial bee colony algorithm. Discrete Dyn. Nat. Soc.