Artificial Bee Colony Algorithm Based on Clustering Method and Its Application for Optimal Power Flow Problem

Liling Sun and Hanning Chen(✉)

Tianjin Polytechnic University, Tianjin 300387, China
sll198257@163.com, perfect_chn@hotmail.com

Abstract. In this paper, an improved multi-objective ABC algorithm based on k-means clustering, called CMOABC, is proposed. For keeping the population diversity, the multi-swarm technology based on k-means clustering is employed to decompose the population into many clusters. Due to each subcomponent evolving separately, after every specific iterations, the population will be re-clustered to facilitate information exchange among different clusters. CMOABC is applied to solve the real-world Optimal Power Flow (OPF) problem that considers the cost, loss, and emission impacts as the objective functions. The simulation results demonstrate that, compared to NSGA-II, MOPSO, and MOABC, the proposed CMOABC is superior for solving OPF problem, in terms of optimization accuracy.

Keywords: K-means clustering · Artificial bee colony algorithm (ABC) · Multi-objective optimization problems (MOPs) · Optimal power flow (OPF)

1 Introduction

Swarm intelligence (SI) is an innovative artificial intelligence technique for solving complex multi-objective optimization problems (MOPs), such as non-dominated sorting genetic algorithm II [1], multi-objective particle swarm optimization [2], multi-objective evolutionary algorithm based on Decomposition [3]. Artificial bee colony (ABC) algorithm is a powerful search technique that drew inspiration from the biological foraging behaviors observed in bee colony [4]. Many researchers have presented several existing multi-objective ABC algorithms [5]. However, these proposed algorithms still suffer from low convergence rate and lacking the diversity of swarm.

To conquer the weakness of initial MOABC, an improved multi-objective ABC algorithm based on k-means clustering, named CMOABC, is proposed. The population is partitioned into several sub-populations based on k-means clustering. Information communication between the sub-populations depends on re-clustering the population after each specific iterations. To further enhance the

M. Gong et al. (Eds.): BIC-TA 2016, Part II, CCIS 682, pp. 101–106, 2016.
DOI: 10.1007/978-981-10-3614-9_13

population diversity, a number of individuals with worse performance re-generate in the re-clustering process.

Optimal Power Flow (OPF) is a classical multi-objective problem. Traditionally, the basic objective of OPF is to schedule the committed generating units to meet the system load demand at minimum operating cost while satisfying the various system equality and inequality constraints [6]. But the passage of clean air act amendments in 1990 forced the utilities to reduce the emission from fossil fuel fired thermal station [7–10]. Therefore, in addition to fuel cost, emission must also be considered as an objective. OPF problem is a non-linear, constrained optimization problem where many competing objectives are present. CMOABC is utilized to solve OPF problem. Compared with MOABC, MOPSO and NSGAII, CMOABC can accommodate considerable potential for solving OPF problem.

2 Optimal Power Flow Problem Formulation

2.1 Minimization of Total Fuel Cost

The fuel cost curves of the thermal generators are modeled as a quadratic cost curves and can be represented as follows:

$$f_{cost} = \sum_{i=1}^{N_g} f_i(a_i {P_{Gi}}^2 + b_i P_{Gi} + c_i) \tag{1}$$

where a_i, b_i and c_i are the the fuel cost coefficients of the ith generator, P_{Gi} is real power output of the ith generator.

2.2 Minimization of Total Power Losses

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

$$f_{lost} = \sum_{k=1}^{N_l} g_k(V_i^2 + V_j^2 - 2V_iV_j\cos(\delta_i - \delta_j)) \tag{2}$$

where N_l is the number of transmission lines, V_i and V_j are the voltage magnitudes at the ith bus and jth bus, respectively; δ_i and δ_j are the voltage angles at the ith bus and the jth bus, respectively.

2.3 Total Emission Cost Minimization

In this paper, two important types of emission gasses, namely, sulpher oxides SOx and nitrogen oxides NOx, are taken as the pollutant gasses. Here, the total emission cost is defined as bellow:

$$f_{emission} = \sum_{i=1}^{N_g} (\alpha_i + \beta_i P_{Gi} + \gamma_i {P_{Gi}}^2) \tag{3}$$

where $f_{emission}$ is the total emission cost (ton/h) and α_i, β_i and γ_i are the emission coefficients of the ith unit.

$$|S_{Li}| \leq S_{Li,\max} \quad i = 1, \ldots, N_l \tag{4}$$

3 CMOABC Algorithm

The stochastically generated population is partitioned into n subpopulations based on the widely adopted k-means cluster method [8]. The number of clusters is determined by the predefined set $G = \{g_1, g_2, \ldots, g_m\}$, where $g_1 > g_2 > \ldots > g_m$. It may happen that two or more clusters come close to each other or get overlapped to a high degree. The distances between each two clusters are calculated as following equation:

$$Dis\_cluster = \left\| cluster_i^{center} - Nei\_cluster_i^{center} \right\| \tag{5}$$

where $Dis\_cluster$ is the distance between one cluster and its neighbor, $Nei\_cluster_i^{center}$ is the center of the ith cluster's neighbor. $cluster_i^{center}$ is the center of the ith cluster. If the distance is smaller than the specific distance DIS_m, one of the clusters will be removed and its non-domination solutions are store.

$$DIS_m = 0.2 * min\left(R_i, R_{i\_neighbor}\right) \tag{6}$$

where R_i is the radius of $cluster_i$ and $R_{i\_neighor}$ is the radius of the neighbors of $cluster_i$.

In order to exchange information among individuals, the whole population is re-partitioned into g_{i+1} clusters based on k-means clustering after each TI iterations, where g_i and g_{i+1} are orderly chosen from the predefined set G. The individuals in a cluster may be distributed into different new clusters when the number of the clusters is changing. To balance the exploration and exploitation, TI is not a constant.

$$TI = \begin{cases} floor\left(0.03 * iter_{max}\right) & \text{if iter} \leq 0.5 * iter_{max} \\ floor\left(0.06 * iter_{max}\right) & \text{if iter} > 0.5 * iter_{max} \end{cases} \tag{7}$$

where $iter_{max}$ is the maximum iterations; $iter$ is the current iteration.

After each TI iterations, a certain number of individuals in each cluster should be regenerated according to this cluster's contribution to the external archive. For the jth cluster, the number of solutions updating to the external archive during each TI iterations is recorded in $Num\_Update(j)$. Then, according to its position in the sort of Num_Update, the number of individuals needed to regenerate in the jth cluster is calculated in Eq. (32). The individuals which will be removed in cluster j are determined by non-domination sort.

$$Num\_regenerate(j) = \frac{Sort\_Update(j)}{g_i} * \frac{Num\_ind(j)}{2} \tag{8}$$

where $Num\_ind(j)$ is the number of individuals in the jth cluster; $Sort\_Update(j)$ indicates the jth cluster's position in the sort of Num_Update; and g_i is the current number of clusters (Table 1).

Table 1. Pseudocode of CMOABC

```
//Step 1: Initialization
Generalize a population of NP individuals in the search region randomly;
Set the numbers of clusters G; Create the external archive EA; initial the current iteration iter = 1;
//Step 2: Loop
while stopping criteria are not satisfied do
   Calculate TI according to Eq. (7)
   Decide the number g_i of clusters according to G
   Partition the whole population based on K-means clustering
   For all cluster
        Compute Dis_cluster according to Eq. (5)
        Decide whether some clusters need to be removed according to DIS_m
        Select x_up individuals based on non-domination.
   End for
   For iter=1:TI
      Update the individuals' position
      Update the external archive and update Num_Update
      iter = iter+1;
   End for
   Calculate the number of individuals in each cluster needing to be regenerated, according to Eq. (8)
   Cenerate a certain number of the new individuals in every cluster;
End while
```

4 Multi-objective Optimal Power Flow Based on CMOABC

In order to validate the robustness of the proposed CMOABC method, a standard IEEE 30 bus system has been used as the test system. The system represents a portion of the American Electric Power System (in the Midwestern US).

The three objectives are optimized simultaneously by the four algorithms and the corresponding best solutions are given in Table 2. Figure 1 also shows the result values of three competing objectives. As shown in Fig. 13, compare to other three algorithms, the Pareto-optimal solutions obtained by the CMOABC are better distributed on the front with good diversity. Among other three algorithms, the Pareto-optimal solutions obtained by the standard MOABC are also well distributed, the diversification of them is not as well as the ones obtained by the proposed CMOABC.

Furthermore, from the Table 2, CMOABC is able to discover a well-distributed and diverse solution set for three-objective problem. However, other three algorithms cannot archive the true Pareto front for three-objective OPF problem. CMOABC obtains the best emission, cost, though the system loss obtained by CMOABC is 2.1601MW, which is a little more than 2.1598MW obtained by MOABC.

Table 2. The best compromise solutions for cost, emission and loss using different multi-objective algorithms

	CMOABC	MOABC	NSGAII	MOPSO
PG1	19.0732	18.9234	35.6214	22.0152
PG2	32.9875	27.9542	53.9065	15.1023
PG3	68.0549	70.8965	47.6936	90.0136
PG4	82.0132	85.9831	45.4762	84.2253
PG5	29.0385	27.0467	54.9945	7.2104
PG6	52.1248	53.0102	46.0154	65.3609
f1 fuel cost	612.0513	614.0154	622.5149	631.4003
f2 (emisson)	0.2120	0.2171	0.2225	0.2341
f3 (loss)	2.1601	2.1598	3.0301	2.8998

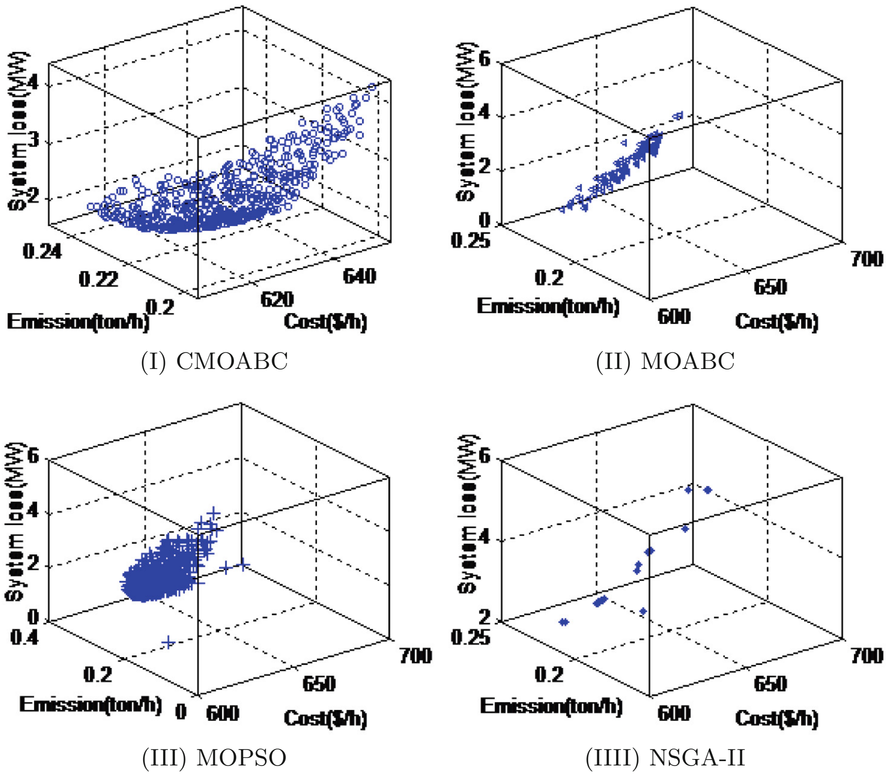

Fig. 1. Pareto fronts obtained by CMOABC, MOPSO, MOABC, and NSGA-II for fuel cost, emission and loss

5 Conclusions

An improved multi-objective ABC algorithm based on k-means cluster, called CMOABC, is proposed. CMOABC adopts k-means clustering method to partition the population into many clusters and the number of the clusters is changing to implement information exchange among the different clusters. CMOABC is used to handle multi-objective OPF problem, and 30-bus IEEE test system is adopted to test the proposed algorithm. By comparing the simulation results of CMOABC, MOABC, MOPSO and NSGAII, the proposed method is able to give well distributed Pareto optimal solutions than other three methods for OPF problem with different objectives.

References

1. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
2. Coello, C.A.C., Pulido, G.T., Lechuga, M.S.: Handling multiple objectives with particle swarm optimization. IEEE Trans. Evol. Comput. **8**(3), 256–279 (2004)
3. Zhang, Q., Liu, W., Li, H.: The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances. In: Proceedings of the Congress on Evolutionary Computation (CEC 2009), Norway, pp. 203–208 (2009)
4. Karaboga, D., Akay, B.: A comparative study of artificial bee colony algorithm. Appl. Math. Comput. **214**, 108–132 (2009)
5. Omkar, S.N., Senthilnath, J., Khandelwal, R., Naik, G.N., Gopalakrishnan, S.: Artificial bee colony (ABC) for multi-objective design optimization of composite structures. Appl. Soft Comput. **11**(1), 489–499 (2011)
6. Sivasubramani, S., Swarup, K.S.: Environmental/economic dispatch using multi-objective harmony search algorithm. Electr. Power Syst. Res. **81**, 1778–1785 (2011)
7. El-Keib, A., Ma, H., Hart, J.: Economic dispatch in view of the clean air act of 1990. IEEE Trans. Power Syst. **9**(2), 972–978 (1994)
8. Halder, U., Das, S., Maity, D.: A cluster-based differential evolution algorithm with external archive for optimization in dynamic environments. IEEE Trans. Cybern. **43**(3), 881–897 (2013)
9. Song, T., Pan, Z., Wong, D.M., Wang, X.: Design of logic gates using spiking neural P systems with homogeneous neurons and astrocytes-like control. Inf. Sci. **372**, 380–391 (2016)
10. Wang, X., Song, T., Gong, F., Pan, Z.: On the computational power of spiking neural P systems with self-organization. Sci. Rep. (2016). doi:10.1038/srep27624