Multi-Objective Artificial Bee Colony with Multi-Movement Strategies for Solving Multi-objective Optimization Problems

Sheng-Ta Hsieh
Department of Communication Engineering
Oriental Institute of Technology, New Taipei City, Taiwan
Tel: (+886) 2-7738-0145 ext. 2325, Email: fo013@mail.oit.edu.tw

Chun-Ling Lin †
Department of Electrical Engineering
Ming Chi University of Technology, New Taipei City, Taiwan
Tel: (+886) 2-2908-9899 ext. 4819, Email: ginnylin@mail.mcut.edu.tw

Abstract. In this paper, a novel multi-objective Artificial Bee Colony (MOABC) is proposed for solving multi-objective problems. In order to increase optimizer's solution searching ability, three types of bees' moving behavior is introduced. Also, the strategies selector is proposed to increase bee colony's utilization and solution searching efficiency. In experiments, the IEEE CEC 2009 test functions are adopted and two recent multi-objective approaches are taken into comparison. From the results, it can be observed that the proposed method performed better than related works.

Keywords: artificial bee colony, multi-objective, optimization, Pareto front, population manager.

1. INTRODUCTION

The main different between single-object (SO) and multi-objective (MO) problems is that MO problems contain more than one conflict objective which needs to be optimized simultaneously. Unlike single-objective problems only have single optimal value (it may have more than one optimal solution); multi-objective problems have a set of optimal value which called Pareto optimal solution.

Srinivas and Deb (1994) proposed a Non-dominated Sorting Genetic Algorithm (NSGA), which ranks populations according to its non-domination characteristic, and better non-dominated solutions will have higher fitness values. Horn et al. (1994) proposed the Niched Pareto Genetic Algorithm (NPGA) which introduced a binary tournament selection without assign definite fitness value. Once problems contains too many objectives will influence its efficiency.

Zitzler and Thiele (1999) proposed the Strength Pareto Evolutionary Algorithm (SPEA), which introduced an elitism strategy to reserve extra population which contains non-dominated solutions. New found non-dominated solutions will be compared with the extra reserved population, and kept the better. An advanced version of SPEA named SPEA2 (Zitzler et al., 2001) inherited the advantages from SPEA and take both dominated and non-dominated solutions into account to improve fitness assignment. The SPEA2 also considered neighbor solutions' diversity to produce and to assign potential local guides.

Besides, an enhanced version of NSGA named NSGA-II (Deb et al., 2000; Deb et al., 2002) is proposed. The NSGA-II introduces a fast non-dominated approach to assign individuals' ranks; also the crowded tournament selection is used for density estimation. Thus, in NSGA-II the individual with a lower density count will be chosen during the selection process.

In recent years, many evolutionary algorithm based multi-objective optimizer (MOEA) methods have been developed and proposed. For example, Campelo et al. (2007) proposed the negative selection, danger theory and immune mechanisms to improve MO optimizer. Zhang and Li (2007) proposed an interested MO algorithm named MOEA/D. In MOEA/D, Multi-objective problems will be decomposed into a number of scalar sub-problems and they will be optimized simultaneously.

Since artificial bee colony (ABC) is proposed by Karaboga (2005), how to apply original or modified ABC for solving MO problems becomes a new topic of MO optimization. Medina et al. (2013) proposed decomposition based multi-objective artificial bee colony named

MOABC/D which exhibits well performance in solving MO problems.

Although, there are many MO approaches have been developed. How to increase MO optimizers' solution searching efficiency is always an import issue. In this paper, the multi movement strategies for food searching are proposed for finding better solution easier. Also, crossover for bees of colony is proposed to enhance ABC's solution searching efficiency.

The rest of the paper is organized as follows: Section 2 introduces artificial bee colony (ABC) briefly, Section 3 describes the proposed method, Section 4 presents the experimental results and the conclusions of this paper is in Section 5.

2. ARTIFICIAL BEE COLONY

Artificial bee colony (Karaboga, 2005; Karaboga and Akay, 2009) is a novel numerical optimizer which simulates bees' social behavior for foraging in solution space for finding global optimal solution and may with reasonable constraints. There are two different kinds of bees which are employed and onlooker bees in ABCs, will try to find new food source (also called solutions). The food searching process of employed bees is performed by following equation.

$$v_{i,j} = x_{i,j} + \emptyset_{i,j} \times \left(x_{i,j} - x_{k,j}\right) \quad (1)$$

where i and k is a random integer between [1, ps], i and k are two random selected bees, and i is not equal to k. The ps represents population size. The j is also a random integer between [1, D], the D denotes dimension of problems. The $\phi_{i,j}$ is a normal distribution number between [-1, 1], x *and* v are current food source and new food source, respectively.

For onlooker bees, food source selection is according to probability which is obtained by equation (2). Except that, onlooker bees performs food searching process is the same as employed bees.

$$p_i = \frac{fit_i}{\sum_{n=1}^{ps} fit_n} \quad (2)$$

where fit_i is fitness value and i denotes the i_{th} bee. The fitness value will be updated by following equation.

$$fit_i = \begin{cases} \frac{1}{(1+f_i)}, & if\ f_i \geq 0 \\ 1 + \text{abs}(f_i), & if\ f_i < 0 \end{cases} \quad (3)$$

where f_i represents objective value of i_{th} bee. If there is no better food source can be found within g generations, the scout bees will then be activated for new phase of food search process and new food source will be random produced by following equation.

$$x_i^{rand} = lb + rand(0,1)(ub - lb) \quad (4)$$

where x^{rand} denotes a new random produced bee, and lb and ub are the search range's the lower and upper bound respectively.

The steps of ABC are listed as follows.

Step 1: Initialization bee colony and generated food source randomly.

Step 2: Fitness Evaluations.

Step 3: Search for new food source by (1) and select better food source by evaluate x and v. (Employed bees' phase)

Step 4: Calculate probability by (2).

Step 5: Select food source by using roulette wheel and keep searching for better food source by (1). (Onlooker bees' phase)

Step 6: Fitness Evaluations.

Step 7: Record the best food source of the colony.

Step 8: If there is no better can be found within limited iterations, scout bee will then be activated and try to search new food source. (Scout bee's phase)

Step 10: Repeat step 3 to 9, until meet termination condition.

3. PROPOSED METHOD

According to original ABC's food searching behavior, the current food source x_i will be move according to x_k. Due to the x_k is not always better or worth than x_i, thus the $\phi_{i,j}$ is set as a normal distribution number between [-1, 1]. It is waste too much time for finding right direction for better food source. In order to overcome the disadvantage, in this paper, multi movement strategies are proposed for employee bees.

3.1 Multi-Movement Strategies for Employee Bees'

Since single food searching equation may not always suitable for all kinds of solution searching situation. Thus, in this paper, three kind of food searching process are proposed, which will be applied according to previous solution searching status. The food searching process of employed bees is performed by following equation.

$$v_{i,j} = x_{i,j} + \emptyset_{i,j} \times \left(x_{best,j} - x_{i,j}\right) \quad (5)$$

where $x_{best,j}$ denotes best food source of the bee colony and $x_{best,j} \neq x_{i,j}$. The j is represents denotes current dimension between [1, D], and D denotes dimension of the objective functions. In this paper, the $\phi_{i,j}$ will be generated by following equation.

$$\emptyset_{i,j} = \begin{cases} \emptyset_l + rand \times \emptyset_u & if\ rand < 0.1 \\ \emptyset_{i,j} & otherwise \end{cases} \quad (6)$$

where ϕ_l and ϕ_u are lower and upper boundary of ϕ, respectively. In this paper, ϕ_l is set as 0.1 and ϕ_u is set as 0.9. Thus, the $\phi_{i,j+1}$ will be set between 0.1 and 1. Thus, the current food source x_i will be move and toward to potential solution via better food source x_{best}.

$$v_{i,j} = x_{i,j} + \emptyset_{i,j} \times (x_{best,j} - x_{rand,j}) \quad (6)$$

where $x_{rand,\ j}$ denotes best food source of the bee colony and $x_{rand,j} \neq x_{i,j}$. Also, the $\phi_{i,j}$ will be generated by equation (6).

$$v_{i,j} = x_{i,j} + \emptyset_{i,j} \times (x_{rr,j} - x_{i,j}) \quad (7)$$

where x_{rr} is random generated food source of solution space. It will be generated by following equation.

$$x_{rr} = x_{lb} + rand(0,1)(x_{ub} - x_{lb}) \quad (8)$$

The one of the three strategies will be applied to current food search process according to following conditions:

1. the bees of the colony can find one or more better solutions in current generation, and the current population size doesn't equal or less than the lower boundary. The existing bees may capable to deal with current solution searching procedures. The redundant bees should be expelled from the colony to reduce their evolution time for speeding up the solution searching progress. Thus, a pair of bees with poor information in the colony will be removed from the population. For next iteration, the bee number will be two less previous iteration. Also, in order to perform deep search, equation (5) will be adopted for next iteration.
2. If bees of the colony cannot find any better solution in previous iteration, and the current population size doesn't equal or exceed the upper boundary. A pair of new bee, which combining the information of two randomly selected bees of the colony, through a crossover-like information combination to provide useful information, will be added into the population. That is, the two newborn bees will be placed at a beneficial position to the population and involved in the solution searching process in the following generation. Also, in order to perform deep search, equation (7) will be adopted for next iteration.
3. If bees of the colony cannot find any better solution in previous iteration, and the current population size equal to the upper boundary of colony. The bees of colony may be trapped into the local optimum during the searching process or need a capable guide to lead them toward the potential area. Thus, the equation (8) will then be adopted.
4. Once current colony reaches the lower or upper boundary, even the bees can find any better solution in current generation or not, the colony size will not be changed.

In this paper, the initial colony size is set as 100, the lower and upper boundary of the colony is set as 50 and 200, respectively.

3.2 Crossover

In order to guide bees toward to potential solution space for searching better food source, in this paper, the crossover is involved. It can prevent convergent prematurely.

The current food source and a randomly selected food source will perform crossover to produce elite bee by follows:

$$e_{i,j} = \begin{cases} v_{i,j} & if\ rand < C_{r1,j} \\ x_{i,j} & otherwise \end{cases} \quad (9)$$

where $e_{i,j}$ denotes potential food source, $v_{i,j}$ represents new food sources and $x_{i,j}$ is current food sources. The $C_{r1,j}$ denotes crossover rate of ith food source. Thus, crossover will be activated by crossover rate but not applied all the time. Similar to equation (6), the crossover rate is set as follows.

$$C_{r1,j}(t+1) = \begin{cases} rand_3 & if\ rand_4 < 0.1 \\ C_{r1,j} & otherwise \end{cases} \quad (10)$$

where t denotes current iteration. If the potential food source ($p_{i,\ j}$) obtained by crossover-stage is better than current food source ($x_{i,\ j}$), the current food source will be replaced in next iteration. Otherwise, current food source will be kept for next iteration. The food source selection is performed after crossover by following equation.

$$x_{i,j}(t+1) = \begin{cases} p_{i,j} & if\ f\left(p_{i,j}(t)\right) \leq f\left(x_{i,j}(t)\right) \\ x_{i,j} & otherwise \end{cases} \quad (11)$$

3.3 External Repository

The function of the external repository controller is to make decision for adding certain solutions into the archive or not. The decision-making process is stated as follows.

1. If the archive is empty, any new solution N_S found will always be accepted and stored in external repository (*Case* 1, in Fig. 1).
2. If the new solution is dominated by any solution in the external repository, then the dominated solution will be discarded (*Case* 2, in Fig. 1).
3. If all the solutions contained in the archive are dominated by new solution, then the new solution will be stored in the archive (*Case* 3, in Fig. 1).
4. Otherwise, if there is any solution in the archive that are dominated by the new solution, then the dominated solutions will be removed from the archive (*Case* 4, in Fig. 1).

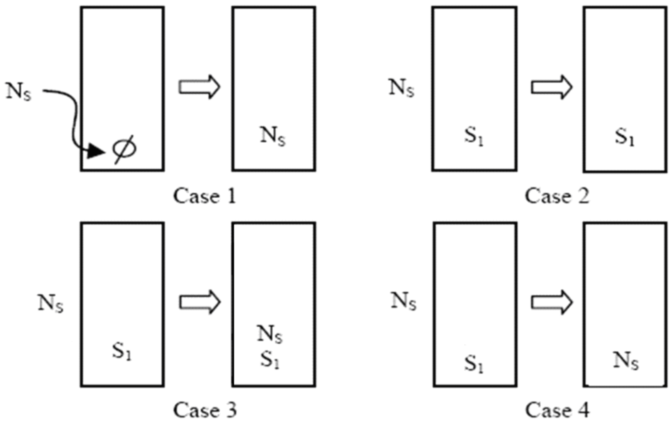

Fig.1. Possible Cases for External Repository

4. EXPERIMENT RESULTS

In the experiments, all the MO methods were implemented by MATLAB 2012b and executed on Intel Xeon E5-2650 processor with 128GB RAM on Windows 7 professional. The maximal number of function evaluations (FEs) was set as 300,000 for all problems. Each algorithm was executed for 30 independent runs. The results of mean values and standard deviation were recorded.

4.1 Test Functions

In the experiments, five unconstrained (bound constrained) MO problems of CEC 2009 technic report (Zhang et al, 2009) were adopted for testing the proposed method with the results compared to related works. The selected problems are listed as follows:

1. Problem 1

$$f_1 = x_1 + \frac{2}{|J_1|}\sum_{j\in J_1}\left[x_j - \sin(6\pi x_1 + \frac{j\pi}{n})\right]^2$$

$$f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|}\sum_{j\in J_2}\left[x_j - \sin(6\pi x_1 + \frac{j\pi}{n})\right]^2$$

$J_1 = \{j \mid j \text{ is odd and } 2 \le j \le n\}$ *and* $J_2 = \{j \mid j \text{ is even and } 2 \le j \le n\}$

search space is $[0,1]x[-1,1]^{n-1}$

$n = 30$

2. Problem 2

$$f_1 = x_1 + \frac{2}{|J_1|}\sum_{j\in J_1} y_j^2$$

$$f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|}\sum_{j\in J_2} y_j^2$$

$J_1 = \{j \mid j \text{ is odd and } 2 \le j \le n\}$ *and* $J_2 = \{j \mid j \text{ is even and } 2 \le j \le n\}$

$$y_j = \begin{cases} x_j - [0.3x_1^2\cos(24\pi x_1 + \frac{4j\pi}{n}) + 0.6x_1]\cos(6\pi x_1 + \frac{j\pi}{n}) & j \in J_1 \\ x_j - [0.3x_1^2\cos(24\pi x_1 + \frac{4j\pi}{n}) + 0.6x_1]\sin(6\pi x_1 + \frac{j\pi}{n}) & j \in J_2 \end{cases}$$

search space is $[0,1]\mathrm{x}[-1,1]^{n-1}$

$n = 30$

3. Problem 3

$$f_1 = x_1 + \frac{2}{|J_1|}(4\sum_{j\in J_1} y_j^2 - 2\prod_{j\in J_1}\cos(\frac{20y_j\pi}{\sqrt{j}}) + 2)$$

$$f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|}(4\sum_{j\in J_2} y_j^2 - 2\prod_{j\in J_2}\cos(\frac{20y_j\pi}{\sqrt{j}}) + 2)$$

$J_1 = \{j \mid j \text{ is odd and } 2 \le j \le n\}$ *and* $J_2 = \{j \mid j \text{ is even and } 2 \le j \le n\}$

$y_j = x_j - x_1^{0.5(1.0 + \frac{3(j-2)}{n-2})}, \ j = 2,3,4,5,\ldots,n$

search space is $[0,1]^n$

$n = 30$

4. Problem 4

$$f_1 = x_1 + \frac{2}{|J_1|}\sum_{j\in J_1} h(y_j)$$

$$f_2 = 1 - x_1^2 + \frac{2}{|J_2|}\sum_{j\in J_2} h(y_j)$$

$J_1 = \{j \mid j \text{ is odd and } 2 \le j \le n\}$ *and* $J_2 = \{j \mid j \text{ is even and } 2 \le j \le n\}$

$y_j = x_j - \sin(6\pi x_1 + \frac{j\pi}{n}), \ j = 2,\ldots,n$

$h(t) = \frac{|t|}{1 + e^{2|t|}}$

search space is $[0,1]\mathrm{x}[-2,2]^{n-1}$

$n = 30$

5. Problem 5

$$f_1 = x_1 + (\frac{1}{2N} + \varepsilon)|\sin(2N\pi x_1)| + \frac{2}{|J_1|}\sum_{j\in J_1} h(y_j)$$
$$f_2 = 1 - x_1 + (\frac{1}{2N} + \varepsilon)|\sin(2N\pi x_1)| + \frac{2}{|J_2|}\sum_{j\in J_2} h(y_j)$$
$$J_1 = \{ j \mid j \text{ is odd and } 2 \le j \le n\}$$
and
$J_2 = \{ j \mid j \text{ is even and } 2 \le j \le n\}$. N is an integer, ε>0

$$y_j = x_j - \sin(6\pi x_1 + \frac{j\pi}{n}),\ j = 2,\ldots,n$$
$$h(t) = 2t^2 - \cos(4\pi t) + 1$$
search space is $[0,1]\text{x}[-1,1]^{n-1}$
$N = 10, \varepsilon = 0.1, n = 30$

4.2 Performance Metric

In order to fair comparison, the performance metric of CEC 2009, which called IGD [16], is also adopted. Let P^* be a set of uniformly distributed points along the PF (in the objective space). Let A be an approximate set to the PF, the average distance from P^* to A is defined as:

$$IGD(A, P^*) = \frac{\sum_{v\in P^*} d(v, A)}{|P^*|} \quad (16)$$

where $d(v, A)$ is the minimum Euclidean distance between v and the points in A. If $|P^*|$ is large enough to represent the PF very well, $IGD(A, P^*)$ could measure both the diversity and convergence of A in a sense. To have a low value of $D(A, P^*)$, The set A must be very close to the *PF* and cannot miss any part of the whole *PF*.

4.3 Experimental Results

Table I presents the mean and standard deviation of 30 runs of the proposed method, MOABC/D and MOEA/D on the five test problems. The best results among the three approaches are shown in bold.

From the results, the proposed method performed better results on problems 1, 3 and 4. In problem 5, the proposed method proposed similar result to MOABC/D. The MOABC/D performed better results on functions 1 and 5.

Table 1: Results of five CEC 2009 test problems

MO Method / Problems	Proposed Method	MOABC/D	MOEA/D
P_1 Results	**2.06e-02** ± **8.34e-03**	2.32e-02 ± 5.07e-03	3.85e-01 ± 3.76e-02
P_2 Results	1.68e-02 ± 1.01e-03	**1.32e-02** ± **3.30e-03**	3.17e-01 ± 1.91e-02
P_3 Results	**2.74e-02** ± **9.32e-03**	7.00e-02 ± 1.94e-02	6.89e-01 ± 3.12e-02
P_4 Results	**4.02e-02** ± **3.56e-04**	4.15e-02 ± 1.18e-03	1.13e-01 ± 2.42e-03
P_5 Results	2.28e-01 ± 1.62e-01	**2.23e-01** ± **5.27e-02**	2.67e+00 ± 9.98e-02

5 CONCLUSIONS

In this paper, three kinds of movement strategies for food searching are proposed; they will be adopted according to previous solution searching status. It can make bees performs deep search, wild search and random search from solution space. Also, crossover is involved to enhance ABC's solution searching ability. It will make proposed MOABC to find the better solutions easier. Five unconstrained (bound constrained) MOP test problems of CEC 2009 technic report were adopted for experiments. From the results, it can find out that the proposed method can find better solution than related works.

ACKNOWLEDGMENTS

This work was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 104-2221-E-161-003-MY2.

REFERENCES

Srinivas, N. and Deb, K. (1994) Multiobjective optimization using nondominated sorting in genetic algorithms, *Evolutionary Computation*, vol. 2, no. 3, 221-248.

Horn, J., Nafpliotis, N. and Goldberg, D. (1994) A niched pareto genetic algorithm for multiobjective optimization, *Proceedings of the First IEEE Conference on Evolutionary Computation*, IEEE World Congress on Computational Intelligence, vol. 1, 82-87.

Zitzler, E. and Thiele, L. (1999) Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach, *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, 257-271.

Zitzler, E., Laumanns, M. and Thiele, L. (2001) SPEA2: Improving the strength pareto evolutionary algorithm," *TIK-Report* 103, Swiss Federal Institute of Technology (ETH) Zurich.

Deb, K., Agrawal, S., Pratab, A. and Meyarivan, T. (2000) A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, 849-858.

Deb, K. Pratap, A., Agarwal, S. and Meyarivan, T. (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, 182-197.

Zhang, Q. and Li, H. (2007) MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition, *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, 712-731.

Karaboga, D. (2005) An idea based on honeybee swarm for numerical optimization, *Technical Report TR*06, Erciyes University .

Medina, M. A., Das, S., Coello Coello, C. A. and Ramirez, J. M. (2013) Two Decomposition-based Modern Metaheuristic Algorithms for Multi-objective Optimization – A Comparative Study, *Proceedings of 2013 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making*, 9-16.

Karaboga, D. and Akay, B. (2009) Artificial bee colony (ABC), harmony search and bees algorithms on numerical optimization, *Proceedings of Innovative Production Machines and Systems, Virtual Conference*, 2009.

Zhang, Q., Zhou, A., Zhao, S., Suganthan, P. N., Liu,W. and Tiwari, S. (2009) Multiobjective optimization Test Instances for the CEC 2009 Special Session and Competition, *Technical Report CES-487*, Nanyang Technological University, Singapore. http://www.ntu.edu.sg/home/EPNSugan