Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A Multi-objective Evolutionary Algorithm of Marriage in Honey Bees Optimization Based on the Local Particle Swarm Optimization

Chenguang Yang, Jie Chen, Xuyan Tu *

* *Department of Automatic Control, School of Information Science and Technology, Beijing Institute of Technology, Beijing, China; Education Ministry Key Laboratory of Complex System Intelligent Control and Decision, Beijing Institute of Technology, Beijing, China (e-mail: weiwujyang@126.com).*

Abstract: Marriage in Honey Bees Optimization (MBO) is a new swarm-intelligence method, but existing researches concentrate more on its application in single-objective optimization. In this paper, we focus on improving the algorithm to solve the multi-objective problem and increasing its convergence speed. The proposed algorithm is named as multi-objective Particle Swarm Marriage in Honey Bees Optimization (MOPSMBO). It uses non-dominated sorting strategy and crowded-comparison approach, utilizes the local Particle Swarm Optimization (PSO) to perform the local characteristic, and simpler the structure of MBO. Based on the Markov chain theory, we prove that MOPSMBO can converge with probability one to the entire set of minimal elements. Simulations are done on several multi-objective test functions and multi-objective Traveling Salesman Problem (TSP). By comparing MOPSMBO with MOGA, NPGA, NSGA and NSGA-II, simulation results show that MOPSMBO has better convergence speed and can better converge near the true Pareto-optimal front.

1. INTRODUCTION

Swarm intelligence usually studies the behavior of social insects and uses their models to solve problems. Recently, based on the marriage process of honey bees, the new algorithm of Marriage in Honey Bees Optimization (MBO) was proposed by Hussein A. Abbass and was shown to be very effective in solving the propositional satisfiability problem (Abbass [2001]). MBO is a kind of swarm-intelligence method. Mating behavior of honey-bees is also considered as a typical swarm-based optimization approach.

Recent years, several researches on MBO were done. Teo Jason et al. introduce a conventional annealing approach into MBO (Teo et al. [2001]). Omid Bozorg Haddad et al. applied MBO to minimize the total square deviation from target demands of a single reservoir with 60 periods (Haddad et al. [2006]). Hyeong Soo Chang et al. adapt MBO into "Honey-Bees Policy Iteration" (HBPI) for solving infinite horizon-discounted cost stochastic dynamic programming problems (Chang [2006]).

While, these researches all applied MBO to single-objective optimization. The objective of this paper is to improve the algorithm to solve the multi-objective problem and increasing its convergence speed.

The proposed algorithm is named as multi-objective Particle Swarm Marriage in Honey Bees Optimization (MOPSMBO). It uses non-dominated sorting strategy to get non-dominated set of solutions and uses a crowded-comparison approach to preserve diversity among solutions. It combines with the local Particle Swarm Optimization (PSO) and simplifies the structure of MBO to increase the convergence speed. Based on the Markov chain theory, we prove that MOPSMBO can converge with probability one to the entire set of minimal elements. And then, to test the proposed algorithm, simulations are done on several multi-objective test functions and also on 48 cities multi-objective Traveling Salesman Problem (TSP). By comparing MOPSMBO with MOGA, NPGA, NSGA and NSGAII, simulation results show that MOPSMBO has better convergence speed and can better converge near the true Pareto-optimal front.

The paper is organized as follows. The problem of multi-objective optimization and its basic definition is introduced in Section2. Section 3 gives the proposed algorithm and its convergence analysis is proved in Section 4. Some simulations are done in Section 5. Finally in Section 6 conclusion is given.

2. MULTI-OBJECTIVE OPTIMIZATION PROBLEM

Many multi-objective evolutionary algorithm were proposed. The Famous algorithms such as Fonseca and Fleming's MOGA (Fonseca et al. [1993]), Srinivas and Deb's NSGA (Srinivas et al. [1995]), Horn et al.'s NPGA (Horn et al. [1994]), Deb's NSGA-II (Deb et al. [2002]) and so on. The following is the basic definitions (Veldhuizen et al. [2000]).

Definition 1. (Pareto Dominance): A vector $u = (u_1, ..., u_k)$ is said to dominate $v = (v_1, ..., v_k)$ (denoted by $u \leq v$) if and only if u is partially less than v , i.e., $\forall i \in \{1, ..., k\}$, $u_i \leq v_i \wedge \exists i \in \{1, ..., k\} : u_i < v_i$.

Definition 2. (Pareto Optimality): A solution $z \in Z$ is said to be Pareto optimal with respect to Z if and only if there is no $z' \in Z$ for which $v = F(z') = (f_1(z'), ..., f_k(z'))$ dominates $u = F(z) = (f_1(z), ..., f_k(z))$.

Definition 3. (Pareto Optimal Set): For a given MOP $F(z)$, the Pareto optimal set (P^*) is defined as $P^* := \{z \in Z \,|\, \neg\exists z' \in Z : F(z') \leq F(z)\}$.

Definition 4. (Pareto Front): For a given MOP $F(z)$ and Pareto optimal set (P^*), the Pareto front (PF^*) is defined as $PF^* := \{u = F(z) = (f_1(z), ..., f_k(z)) \,|\, z \in P^*\}$

3. MULTI-OBJECTIVE PARTICLE SWARM MARRIAGE IN HONEY BEES OPTIMIZATION

3.1 Local Particle Swarm Optimization

PSO was originally developed by a social-psychologist (James Kennedy) and an electrical engineer (Russell Eberhart) in 1995 and emerged from earlier experiments with algorithms that modeled the "flocking behavior" seen in many species of birds.(Chen et al. [2003], Pan et al. [2006])

The scheme for updating the position and velocity of each particle is shown below:

$$\begin{cases} V_i(t+1) = w \cdot V(t) + c_1 \cdot r_1 \cdot (P_i - X_i(t)) \\ \qquad\qquad +c_2 \cdot r_2 \cdot (P_g - X_i(t)) \\ X_i(t+1) = X_i(t) + V_i(t+1) \end{cases} \tag{1}$$

where V_i is the velocity of the particle, and X_i is the its current poison. P_i is the best position found by the particle i by far and P_g is the best position in the swarm at that time. r_1 and r_2 are random numbers between $[0, 1]$. c_1 and c_2 are called learning parameters. w is the weighted parameter between $[0.1, 0.9]$.

We found that, when w equal zero, PSO will convergence to local optimization (Zeng et al. [2004]). Then (1) becomes

$$\begin{aligned} X_i(t+1) = X_i(t) + c_1 \cdot r_1 \cdot (P_i - X_i(t)) \\ +c_2 \cdot r_2 \cdot (P_g - X_i(t)) \end{aligned}. \tag{2}$$

We can get $X_i(t) = \frac{c_1 r_1 P_i + c_2 r_2 P_g}{c_1 r_1 + c_2 r_2} + \left(X_i(0) - \frac{c_1 r_1 P_i + c_2 r_2 P_g}{c_1 r_1 + c_2 r_2}\right) \cdot (1 - c_1 r_1 - c_2 r_2)^t$. If $|1 - c_1 r_1 - c_2 r_2| \prec 1$, then $\lim_{t \to \infty} X_i(t) = \frac{c_1 r_1 P_i + c_2 r_2 P_g}{c_1 r_1 + c_2 r_2}$. When[$t \to \infty$, $X_i(t+1) = X_i(t)$ and for equation (2), we can get $c_1 r_1 P_i + c_2 r_2 P_g = (c_1 r_1 + c_2 r_2) X_i(t)$. Because c_1, c_2, r_1 and r_2 are statistic variablesso $\lim_{t \to \infty} X_i(t) = P_i = P_g$.

3.2 Proposed Algorithm

MBO have five main processes. (a) The mating-flight of the queen bees with drones encounter at some probabilistically. (b) Creating new broods by the queen bees. (c) Improving the broods' fitness by workers. (d) Updating the workers' fitness. (e) Replacing the worst queen(s) with the fittest brood(s).

By using a fast non-dominated sorting strategy and a crowded-comparison approach, utilizing the local PSO as the Worker and adopting simplified structure of MBO, we propose the MOPSMBO.

```
Define Q: the number of queens
       D: the number of drone
       M: the sperm theca size
Initialize each worker with a unique heuristic
Initialize each queen's genotype at random
Apply local PSO method to improve the queen's genotype
While the stopping criteria is not satisfied (Cycle Times
bigger than Max Cycle Number or result is good enough)
  for queen = 1 to Q
    for i=1 to M
       generate a drone randomly;
       add spermatozoa to the queens' spermtheca;
       fast non-dominated sort;
       crowding distance assignment;
       crowded-comparison operator;
       generate a brood by crossovering the queen's genome
         with the selected sperm;
       mutate the generated brood's genotype;
       use local PSO to improve the drone's genotype ;
       if the new brood is better than the worst queen
         replace the least-fittest queen with the new brood;
         refresh the queen list depending on fitness;
       end if
    end for
  end for
end while
```

Fig. 1. Mean f2 Results of evaluation function 1

In MOPSMBO, we define four operators: Selection, Crossover, Mutation and Heuristic.

(a)Selection: The queens keep the optimal ones in each generation, which is considered as the strategy of selecting elites.

(b)Crossover: Crossover operator exchanges the pieces of genes between chromosomes.

(c)Mutation: Mutation operation alters individual alleles at random locations of random chromosomes at a very probability.

(d)Heuristic: For the good local convergence performance, we use local PSO method as the heuristic operator.

4. CONVERGENCE ANALYSIS OF MOPSMBO ALGORITHM

In this section, we use Markov chain to analysis the convergence of the MOPSMBO algorithm.

4.1 Markov Chain and Some Basic Definitions

Markov chains have been used extensively to study convergence characteristic.

Definition 5. A square matrix is $A = [a_{ij}]_{n \times n}$.

$$\begin{aligned}
&(a)\mathrm{if}\forall i, j \in \{1, \ldots n\} : a_{ij} > 0, \\
&\qquad\qquad A \mathrm{is positive}(A > 0); \\
&(b)\mathrm{if}\forall i, j \in \{1, \ldots n\} : a_{ij} \geq 0, \\
&\qquad\qquad A \mathrm{is nonnegative}(A \geq 0); \\
&(c)\mathrm{if} A \geq 0 \ \ \mathrm{and}\exists m \in \mathrm{N} : A^m > 0, \\
&\qquad\qquad A \mathrm{is primitive}; \\
&(d)\mathrm{if} A \geq 0 \ \ \mathrm{and}\forall i \in \{1, \ldots n\} : \sum_{j=1}^{n} a_{ij} = 1, \\
&\qquad\qquad A \mathrm{is stochastic}; \\
&(e)\mathrm{if}\forall i \in \{1, \ldots n\} : a_{ii} \succ 0, \\
&\qquad\qquad A \mathrm{is diagonal-positive}; \\
&(f)\mathrm{if}\forall j \in \{1, \ldots n\}, \exists i \in \{1, \ldots n\}, a_{ij} \succ 0, \\
&\qquad\qquad A \mathrm{is column-allowable}.
\end{aligned} \tag{3}$$

Definition 6. If the state space S is finite ($|S| = n$), and the transition probability $p_{ij}(t)$ are independent from t,

$$\exists i, j \in S, \exists u, v \in \mathrm{N}, p_{ij}(u) = p_{ij}(v) \tag{4}$$

the Markov chain is said to be finite and homogeneous. $p_{ij}(t)$ is the probability of transitioning from state $i \in S$ to state $j \in S$ at step t .

Definition 7. (G Rudolph et al. 2001): An element $z^* \in F$ is called a minimal element of the poset $(F, \leq)$, if there is no $z \in F$ such that $z \prec z^*$. The set of all minimal elements, denoted $M(F, \leq)$, is said to be complete if for each $z \in F$ there is at least one $z^* \in M(F, \leq)$ such that $z \leq z^*$.

Theorem 8. For a homogeneous finite Markov chain, with the transition matrix, if

$$\exists m \in \mathrm{N} : P^m > 0, \tag{5}$$

then this Markov chain is ergodic and with finite distribution. $\lim\limits_{t\to\infty} p_{ij}(t) = \overline{p_j}, i, j \in S$ is the steady distribution of the homogeneous finite Markov chain.

Theorem 9. (G. Rudolph 1994, The basic limit theorem of Markov chain) If P is a primitive homogeneous Markov chain's transition matrix, then

$$\begin{array}{l} \text{(a)}\exists!\omega^T > 0 : \omega^T P = \omega^T, \omega\text{: a probability vector.} \\ \text{(b)}\forall\varphi_i \in S\ (\varphi_i \text{ is the start state and it's} \\ \quad \text{probability vector is } g_i^T) : \lim\limits_{k\to\infty} g_i^T P^k = \omega^T \\ \text{(c)From } \lim\limits_{k\to\infty} P^k = \overline{P}, \overline{P}\text{: limit probability matrix} \\ \ ,n \times n, \text{it's all rows are same to}\omega^T. \end{array} \tag{6}$$

Theorem 10. (RudolphA et al. [2001]) Let I, D, C, P, A be stochastic matrices where I is irreducible, D is diagonal-positive, C is column-allowable, P is positive, and A arbitrary. Then

(a) AP and PC are positive,

(b) ID and DI are irreducible.

4.2 Convergence of MOPSMBO

Definition 11. The state space of MOPSMBO is

$$\begin{array}{l} X = \{X_t = [x_1, x_2, \ldots, x_M] \,|x_i \in S, i = 1, \ldots, M\} \\ S = \{x = [q_1, q_2, \ldots, q_N] \,|q_i \in \{0, 1\}, i = 1, \ldots, N\} \end{array}, \tag{7}$$

where $[q_1, q_2, \ldots, q_N]$ is the binary bit cluster listed in turn. N is the dimension of a population member. M is the number of population members in one generation. X_t represents the population of iteration t, and it is a big binary bit cluster composed by that's of all population members. X is the state space of MOPSMBO.

Theorem 12. MOPSMBO can be described as a Markov chain, and the Markov chain is finite and homogeneous.

Proof. With $\rho_i, \rho_j \in X$, the probability of transformation from the state ρ_i to the state ρ_j at step t only depends on ρ_i and is independent of time. So the MOPSMBO can be described as a Markov chain. From the whole evolution process of MOPSMBO, we can see that all four operators (Selection, Crossover, Mutate and Heuristic) don't change with time. Thus the Markov chain of the MOPSMBO algorithm is homogeneous. And then, the number of the population members (M) is finite, we can know easily that the Markov chain of the MOPSMBO algorithm is finite.

We use four transition matrix S, C, M and H to describe their infections respectively. Finally, we can get

$$Tr = S \cdot C \cdot M \cdot H, \tag{8}$$

where Tr is the transition matrix of the Markov chain of the MOPSMBO algorithm.

Theorem 13. The transition matrixes of the selection probability (S) in the MOPSMBO algorithm is column-allowable.

Proof. The selection operator is a deterministic operator. Every generation, the best populations are saved in the queens, and the worst ones are rejected.

The square matrix S is $S = [s_{ij}]_{n\times n}$. Then

$$\forall j \in \{1, \ldots n\} : \exists i \in \{1, \ldots n\}, s_{ij} > 0. \tag{9}$$

So S is column-allowable.

Theorem 14. The transition matrixes of the crossover probability (C) and Heuristic probability (H) in the MOPSMBO algorithm are all stochastic.

Proof. The square matrix C is $C = [c_{ij}]_{n\times n}$. Then

$$\begin{cases} \forall i, j \in \{1, \ldots n\} : c_{ij} \geq 0 \\ \forall i \in \{1, \ldots n\} : \sum\limits_{j=1}^{n} c_{ij} = 1 \end{cases} \tag{10}$$

So C is stochastic.

The square matrix H is $H = [h_{ij}]_{n\times n}$. Then,

$$\begin{cases} \forall i, j \in \{1, \ldots n\} : h_{ij} \geq 0 \\ \forall i \in \{1, \ldots n\} : \sum\limits_{j=1}^{n} h_{ij} = 1 \end{cases} \tag{11}$$

So H is stochastic.

Theorem 15. The transition matrix of the MOPSMBO with mutation probability (M) is stochastic and positive.

Proof. $M = [m_{ij}]_{n\times n}$ is a square matrix. Then

$$\begin{cases} \forall i, j \in \{1, \ldots n\} : m_{ij} \geq 0 \\ \forall i \in \{1, \ldots n\} : \sum\limits_{j=1}^{n} m_{ij} = 1 \end{cases} \tag{12}$$

So M is stochastic.

And the mutation has an influence on every position of a state vector. We can easily know $\forall x_i, x_j \in X$. Each position of x_i can mutate to the value of x_j . So the probability of x_i mutate to x_j is positive. So M is positive.

Theorem 16. The Markov chain of the MOPSMBO (Tr) is ergodic and with finite distribution, $\lim\limits_{t\to\infty} tr_{ij}(t) = \overline{tr_j} > 0, i, j \in X$.

Proof. Let $Tg = C \cdot M \cdot H$, We have $Tr = S \cdot Tg$. According to Theorem 14, Theorem 15, Tg is positive. According to Theorem 13, and Theorem 10, Tr is positive. And according to Theorem 8, this proposition is proved.

Definition 17. (G Rudolph et al. 2001): Let X_t be the population of MOPSMBO algorithm at iteration $t \geq 0$ and $F_t = f(X_t)$ its associated image set. The evolutionary algorithm is said to converge with probability 1 to the entire set of minimal elements if $d(F_t, F^*) \to 0$ with probability 1 to the set of minimal elements, Here, F^* denotes the set of minimal elements, $d(F_t, F^*) = |A \cup B| - |A \cap B|$.

Theorem 18. The MOPSMBO converges with probability 1 to the entire set of minimal elements.

Proof. S_{now} denotes the set of selected population member (the queens). $F^* = M(F, \leq)$ denotes the complete set of minimal elements (Pareto optimal set). $F_t = f(X_t)$ denotes the image set of X_t.

Let $x_1 \in S_{now}$ and $f(x_1) \notin F^*$. Because $F^* = M(F, \leq)$ is a complete set of all minimal elements. Depending on the Definition 7, $\exists x_2 \in X, f(x_2) < f(x_1)$. Theorem 16 tell us that MOPSMBO can reach every element in state space X infinitely often and the selection operator use elite strategy (save in queens). So if at iteration t, $\exists x_2 \in X_t, f(x_2) < f(x_1)$, then x_1 will be replaced by x_1. This process will go on until $\forall x_1 \in S_{now}, f(x_1) \in F^*$. Therefore non-optimal elements will be eliminated after a finite number of iterations with probability one.

The selection operator use elite strategy. So if $x_3 \in S_{now}$ and at iteration t, $\exists x_4 \in X_t, f(x_4) < f(x_3)$, then x_3 will be deleted from S_{now} . But if $f(x_3) \in F^*$ according to Definition 7 that $\neg\exists x_4 \in X_t, f(x_4) < f(x_3)$. So if $f(x_3) \in F^*$ and $x_3 \in S_{now}$, then x_3 will not be deleted from S_{now}.

To sum up, in finite time, all non-optimal elements will be discarded and all optimal ones will go into S_{now} with probability one. At the same time optimal ones do not go out of S_{now}. So the MOPSMBO algorithm converges with probability 1 to the entire set of minimal elements.

5. SIMULATIONS

To test the convergence performance of MOPSMBO, we choose MOGA, NPGA, NSGA, NSGA-II for comparison. We did the simulation on two parts, one using some popular complex multi-objective Evaluation Functions and the other using multi-objective TSP.

5.1 Comparison on multi-objective Evaluation Functions

The initial value is generated randomly, and population number is 30, generation number is 50.

Evaluation Function 1 : Schaffer

$$\begin{aligned} f_1(x) &= x^2 \\ f_2(x) &= (x-2)^2 \end{aligned} \tag{13}$$

Evaluation Function 2 : Fonseca

$$\begin{aligned} f_1(x_1,...,x_n) &= 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i - 1/\sqrt{n}\right)^2\right) \\ f_2(x_1,...,x_n) &= 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i + 1/\sqrt{n}\right)^2\right) \end{aligned} \tag{14}$$

5.2 Comparison on multi-objective Travelling Salesman Problem

A Multi-objective Traveling Salesman Problem (TSP) has been an interesting problem. Here it is solved by MOPSMBO, MOGA, NPGA, NSGA and NSGA-II respectively.

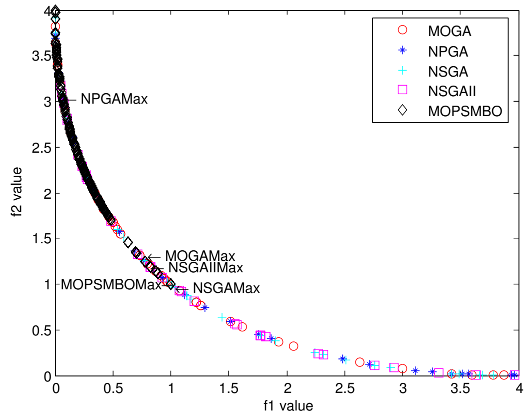

Fig. 2. Pareto-set Results of evaluation function 1

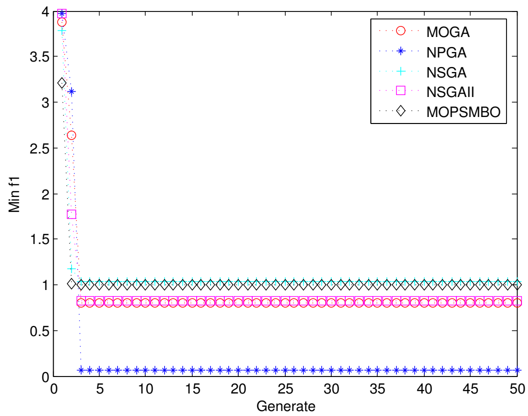

Fig. 3. Min f1 Results of evaluation function 1

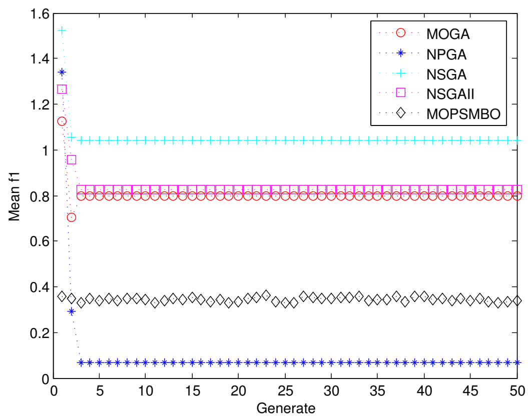

Fig. 4. Mean f1 Results of evaluation function 1

5.3 Some Remarks

From the above, we can see that:

(a) MOPSMBO is convergent and keeps good performance for all these test functions, though these test function are more complex than the normal ones and may have many local optimization points.

(b) MOPSMBO performs better than MOGA, NPGA, NSGA and NSGA-II. MOPSMBO converges more quickly, especially at initial part. Particularly, even if the ini-

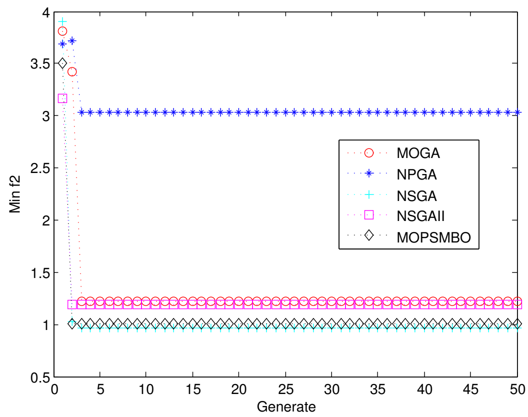

Fig. 5. Min fs Results of evaluation function 1

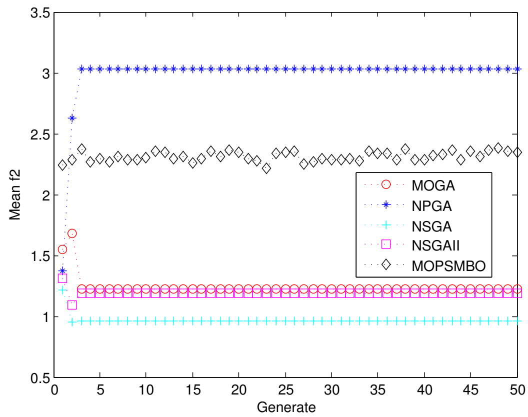

Fig. 6. Mean f2 Results of evaluation function 1

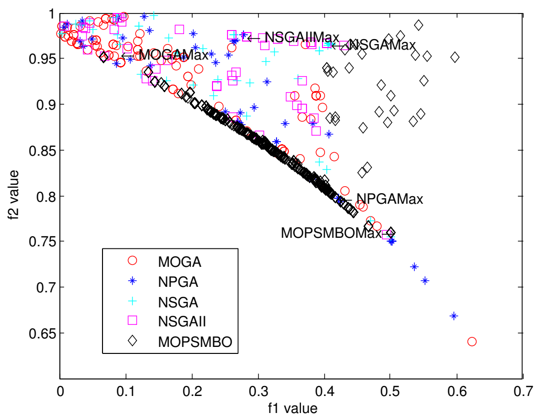

Fig. 7. Pareto-set Results of evaluation function 2

tial condition is worse than MOGA, NPGA, NSGA and NSGA-II, MOPSMBO can show finer result.

(c) On the multi-objective TSP test, MOPSMBO show better performance than MOGA, NPGA, NSGA and NSGA-II and can keep converge faster with both distance and money.

6. CONCLUSIONS

In this paper, we proposed a multi-objective Particle Swarm Marriage in Honey Bees Optimization (MOPSMBO) algorithm. It uses non-dominated sorting strategy and

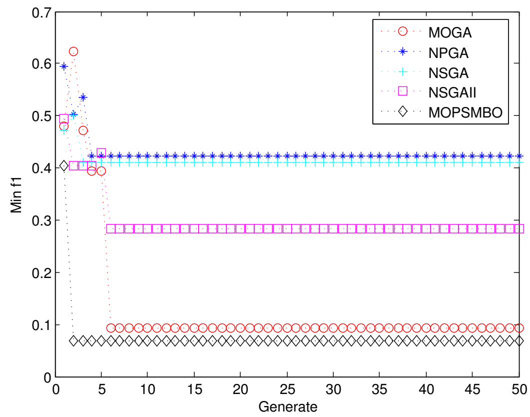

Fig. 8. Min f1 of evaluation function 2

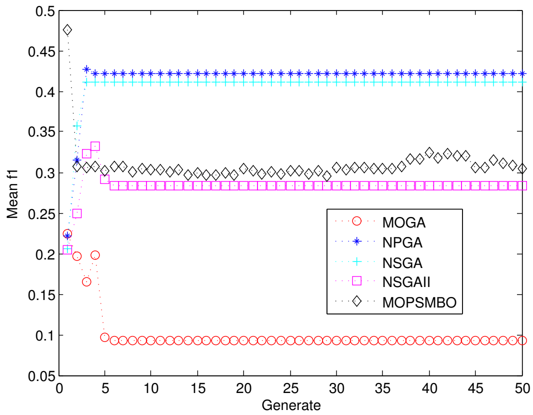

Fig. 9. Mean f1 of evaluation function 2

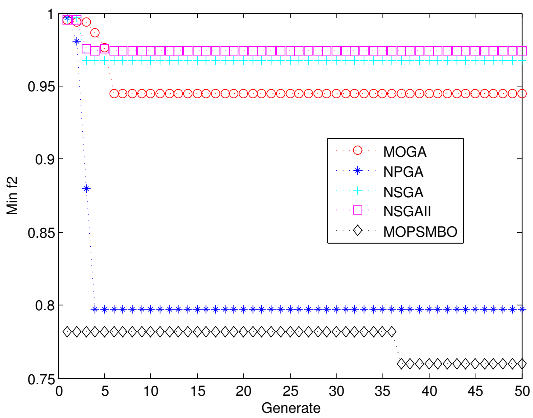

Fig. 10. Min f2 of evaluation function 2

crowded-comparison approach, utilizes the local Particle Swarm Optimization (PSO) to perform the local characteristic, and simpler the structure of MBO.

Based on the Markov chain theory, we prove that MOPSMBO can converge with probability one to the entire set of minimal elements.

Simulating with multi-objective evaluation functions and multi-objective TSP, MOPSMBO shows better performance than MOGA, NPGA, NSGA and NSGA-II. And it has better convergence speed and can better converge near the true Pareto-optimal front.

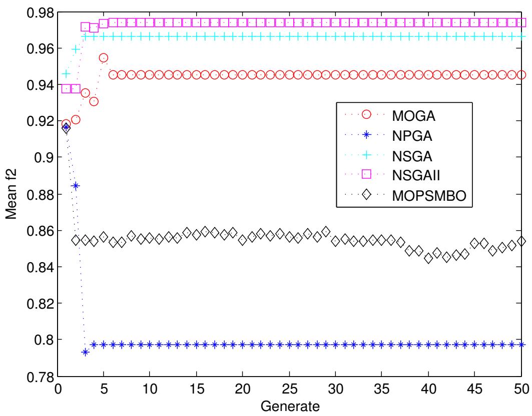

Fig. 11. Mean f2 of evaluation function 2

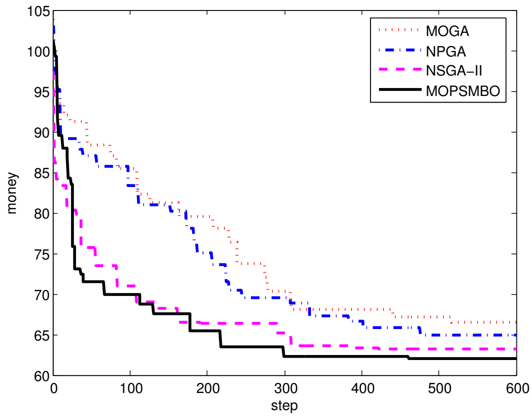

Fig. 12. The distance of TSP with 48 nodes solved by MOGA, NPGA, NSGA-II and MOPSMBO

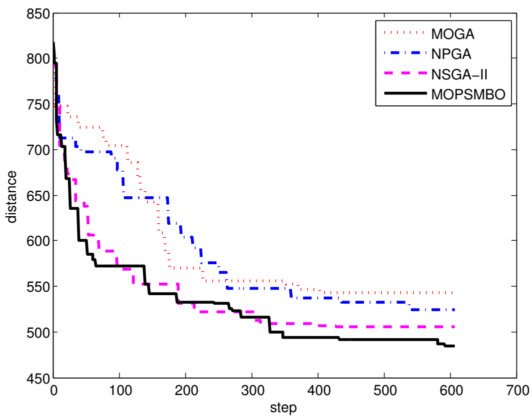

Fig. 13. The money of TSP with 48 nodes solved by MOGA, NPGA, NSGA-II and MOPSMBO

The algorithm still deserves deep study. And the research about MOPSMBO will be carried out and will be tested and improved with high dimension (> 2) practical cases in the future.

REFERENCES

H.A. Abbass. Marriage in Honey Bees Optimization (MBO): a haplometrosis polygynous swarming approach. *Congress on Evolutionary Computation, CEC2001*, Seoul, Korea, pages 207–214. 2001.

H.A. Abbass. A single queen single worker honey-bees approach to 3-SAT. *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO2001*, San Francisco, USA, pages 804–817. 2001.

Jason Teo and H.A. Abbass. An annealing approach to the mating-flight trajectories in the marriage in honey bees optimization algorithm. In A.F. Round, editor, *Technical Report CS04/01.* Academic Press, School of Computer Science, University of New South Wales at ADFA, 2001.

Omid Bozorg Haddad, Abbas Afshar and Miguel A. Marino. Honey-bees mating optimization (HBMO) algorithm: a new heuristic approach for water resources optimization. *Water Resources Management*, volume 20, pages 661–680. 2006.

Hyeong Soo Chang. Converging marriage in honey-bees optimization and application to stochastic dynamic programming. *Journal of Global Optimization*, volume 35, pages 423–441. 2006.

C. M. Fonseca and P. J. Fleming. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. *Proceedings of the Fifth International Conference on Genetic Algorithms*, S. Forrest, Ed. San Mateo, CA: Morgan Kauffman, pages 416–423. 1993.

N. Srinivas and K. Deb. Multiobjective function optimization using nondominated sorting genetic algorithms. *Evol. Comput*, volume 2(3), pages 221–248. 1995.

J. Horn, N. Nafploitis, and D. E. Goldberg. A niched Pareto genetic algorithm for multiobjective optimization. *Proceedings of the First IEEE Conference on Evolutionary Computation*, Z. Michalewicz, Ed. Piscataway, pages 82–87. 1994.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computaion*, volume 6(2), pages 182–197. 2002.

DA Van Veldhuizen, GB Lamont. Evolutionary Computation. *Evolutionary Computation*, volume 8(2), pages 125–147. 2000.

Jie Chen, Feng Pan, Tao Cai, Xuyan Tu. Stability analysis of particle swarm optimization without Lipschitz constraint. *Journal of Control Theory and Applications*, volume 1, pages 86–90. 2003.

Feng Pan, Jie Chen, Minggang Gan, Tao Cai, Xuyan Tu. Model Analysis of Particle Swarm Optimizer. *ACTA AUTOMATICA SINICA*, volume 32(3), pages 368–377. 2006.

Zeng Jian Chao, Jie Qing, Cui Zhi Hua. Particle Swarm Optimization Algorithm. *Evolutionary Computation.*Science Press, Beijing, 2004.

G RudolphA Agapie. Convergence Properties of Some Multi-objective Evolutionary Algorithm. *Proceedings of the 2001 IEEE Congress on Evolutionary Computation*, Seoul, Korea. pages 27–30. 2001.

G. Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transaction Neural Networks*, volume 5(1), pages 96–101. 1994.