Evolutionary Multi-tasking Single-objective Optimization based on Cooperative Co-evolutionary Memetic Algorithm

Qunjian Chen, Xiaoliang Ma, and Zexuan Zhu
College of Computer Science and Software Engineering
Shenzhen University
Shenzhen, China
maxiaoliang@yeah.net, zhuzx@szu.edu.cn

Yiwen Sun
School of Medicine
Shenzhen University
Shenzhen, China

***Abstract*—Evolutionary multi-tasking optimization has recently emerged as a promising new topic in the field of evolutionary computation. It is a promising framework for solving different optimization problems simultaneously. Compared with the classic evolutionary algorithms, evolutionary multi-tasking optimization (MTO) can take advantage of implicit genetic transfer in the optimization process and get better performance. Distinct tasks are solved simultaneously by utilizing similarities and differences across different tasks. In this paper, an evolutionary multi-tasking single-objective optimization based on cooperative co-evolutionary memetic algorithm (EMTSO-CCMA) is proposed. A local search method based on quasi-Newton is proposed to accelerate the convergence of the proposed algorithm. The effectiveness of the proposed algorithm is shown in this paper by comparing with the multifactorial evolutionary algorithm.**

Keywords-evolutionary multitasking; multifactorial optimization; memetic algorithm; cooperative co-evolutionary genetic algorithm

I. INTRODUCTION

The idea of evolutionary algorithms (EAs) stems from the Darwinian's theory of "survival of the fittest" [1]. In EAs, the process of searching for the optimal solution begins with an initial population. Offspring are generated by crossover and variation operators. In each generation, fitter individuals are more likely to be selected into the next generation and participate in the breeding process.

In traditional EAs, different optimization problems are usually solved separately. Inspired by human beings' ability to deal with multiple tasks simultaneously, an evolutionary multitasking paradigm namely multifactorial optimization (MFO) has been proposed in [2] to solve multiple problems at the same time. MFO utilizes the correlation of different optimization problems to facilitate the solving of these problems simultaneously, instead of solving them separately. In MFO, each optimization problem possesses a unique function landscape, and provides a particular factor influencing the evolution of population. The solving process of one problem can help the solving process of other problems if they have something in common.

Some related numerical experiments have shown the superiority of multifactorial evolutionary algorithm (MFEA) over traditional EAs [2]. However, MFEA still suffers from issues like slow convergence in local region and impotence of handling high-dimensional problems. In this paper, an evolutionary multi-tasking single-objective optimization based on cooperative co-evolutionary memetic algorithm (EMTSO-CCMA) is proposed to deal with these issues. Firstly, EMTSO-CCMA uses knowledge exchange of optimization problems to speed up the optimization process. A local search based on quasi-Newton method is applied resulted in a memetic algorithm [9-11, 16, 17]. To deal with high-dimensional problems, the cooperative co-evolutionary framework is considered. Cooperative co-evolution is inspired by the ecological relationship of symbiosis [13], which can be interpreted as various species cohabit in a mutually beneficial way. These species influence each other's evolution process through multiple ecological interactions. In cooperation coevolution framework, large-scale/high-dimensional global optimization problems are decomposed into a set of lower-dimensional sub-problems [14, 15]. EMTSO-CCMA is tested on benchmark problems and shown to obtain superior or comparable performance to MFEA.

II. BASIC ALGORITHMS

A. Multifactorial Optimization (MFO)

To solve K different optimization problems, say minimization problems, simultaneously, MFO builds on the implicit parallelism of population-based search:

$$\{x_1, x_2, \dots, x_{K-1}, x_K\} = \mathrm{argmin}\{f_1(x), f_2(x), \dots, f_{K-1}(x), f_K(x)\}$$

Each optimization problem $f_i(\cdot)$ has a unique search space X_j and objective function $f_j : X_j \rightarrow R$, and x_j is a solution in X_j. Each optimization problem $f_j(\cdot)$ in the MFO devotes a particular factor to promote the evolutionary process. Therefore, the composite optimization problem can be defined as a K-factorial problem [2]. All individuals are encoded in a unified search space Y encompassing $X_1, X_2, \dots, X_k$. Each individual p_i in a population P possesses a set of properties:

- *Factorial Cost*: $\psi_{ij} = \lambda\delta_{ij} + f_{ij}$ where f_{ij} is the objective value of individual p_i on optimization problem $f_j(\cdot)$, λ and δ_{ij} are the large penalizing multiplier and the total constraint violation, respectively.
- *Factorial Rank:* For a given optimization problem $f_j(\cdot)$, all the individuals are sorted in ascending order with

0-7695-6341-4/17/31.00 ©2017 IEEE
DOI 10.1109/CIS.2017.00050

respect to the *factorial cost*. The rank r_{ij} of individual p_i is the index of p_i in the *j-th* optimization problem $f_j(\cdot)$.

- *Skill Factor:* The *skill factor* τ_i of p_i represents an optimization problem, on which the individual p_i is the most effective solution, i.e., $\tau_i = \min\{r_{ij}\}, j \in \{1, 2, \ldots, K\}$.
- *Scalar Fitness:* The *scalar fitness* is defined as $\varphi_{ij} = 1/r_{ij}$ for individual p_i on problem $f_j(\cdot)$.
- *Multifactorial Optimality:* A solution is optimal when it is globally optimal in at least one optimization problem.

B. Cooperative Co-evolution Genetic Algorithm (CCGA)

Cooperative co-evolutionary genetic algorithm (CCGA) was first proposed by Potter and De Jong in 1994 [13]. CCGA works by firstly decomposing the original high-dimensional problem into a number of lower-dimensional sub-problems. Secondly, a context vector $\boldsymbol{b} = (b_1, b_2, \ldots, b_n)$ is generated randomly. Thirdly, each subcomponent is optimized by a separated GA. The fitness of the *j-th* offspring in the *i-th* subpopulation is defined as C_{ij}, which is evaluated by calculating the fitness of the context vector with the corresponding position replaced by C_{ij}, and the context vector is updated accordingly. Finally, a new subpopulation is selected from the current subpopulation. The procedure is repeated until some terminal conditions are satisfied. The process is illustrated with an example shown in Fig.1. The details of CCGA are provided in [12-16].

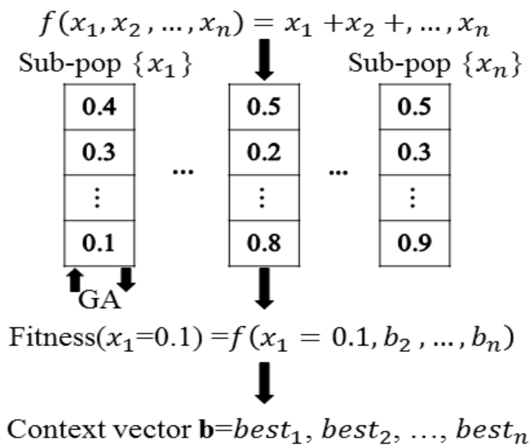


Figure 1. The general process of the CCGA

III. EMTSO-CCMA

A. Basic Framework of EMTSO-CCMA

Based on the MFO framework and CCGA, we proposed EMTSO-CCMA which is outlined in Algorithm 1. In Algorithm 1, lines 1-7 can be regarded as the initialization stage. Firstly, an initial population is generated randomly in line 1. Each individual contains an *n*-dimensional decision vector and a set of properties. In order to calculate these properties, every individual is evaluated for all tasks in line 3. The *factorial rank* r_{ij} and *skill factor* τ_i of individual p_i are assigned for each individual in lines 4-5.

A typical multi-task evolution process is described in lines 8-16. First of all, the current population generates the same amount of offspring by crossover and mutation. Secondly, the algorithm evaluates each offspring in lines 10-12. Thirdly, in line 13-16, the proposed algorithm selects the fitter individuals from the current population and puts them to the next generation. Note that the *skill factor* of offspring is inherited from its parents. To reduce the computational cost, each offspring is evaluated for only one optimization problem, as shown in line 11. Moreover, the best individuals for every task in the current generation are recorded in the line 15. These individuals are used as input in the Algorithm 2 and Algorithm 3.

In line 17, a learning strategy is proposed to improve the learning effectiveness between optimization problems. It is introduced in the Algorithm 2. Line 18 presents Algorithm 3 which introduces a local decomposition method based on the CCGA.

B. Multitask Learning Strategy

Algorithm 1. Pseudocode of the EMTSO-CCMA

Numerical value: *n*: the dimension of problems. *N*: the population size. *K*: the number of tasks.

1. Randomly generate a population P_0 of *N* individuals.
2. **for each** p_i in P_0 **do**
3.   Evaluate p_i on all optimization problems
4.   Calculate *factorial rank* r_{ij} for p_i, $i=1,2,\ldots,K$
5.   Assign *skill factor* τ_i of p_i
6. **end for**
7. Set t=1
8. **While** stopping criterions are not met **do**
9. C_t=Crossover + Mutate(P_t)
10.   **for each** c_i in C_t **do**
11.   Evaluate c_i for optimization problem τ_i only
12.   **end for**
13. R_t=C_t + P_t
14.   Calculate *scalar fitness* φ_{ij} for all individuals
15.   Record best individuals for every optimization problem in current generation: $b_1, b_2, .., b_k$
16.   Select *N* elite individuals from R_t as P_{t+1}
17.   Multitask learning strategy – (see Algorithm 2)
18.   Local decomposition evolution – (see Algorithm 3)
19.   Set t=t+1
20. **end while**

As shown in Algorithm 2, an effective learning strategy between optimization problems is introduced. The input of the Algorithm 2 is a set of individuals which obtain the best fitness for at least one task in current generation. Suppose there are K optimization problems, the number of input individuals is K. In line 2, a quasi-Newton [6] method is implemented for the K individuals, each of them is the best individual for a single optimization problem. Line 3 exchanges their *skill factor* with ring topology. Through such adjustment, K new individuals are generated and evaluated based on their corresponding *skill factor* in line 4.

Algorithm 2 is inspired by the learning of the human body. For example, if an operation can be mastered by right hand, the left hand can do the same operation as well to a certain extent. In the multitasking environment, each individual may inherit different genetic materials from different parents. This is a unique advantage of multifactorial evolution. It is not difficult to speculate that an excellent individual may be effective to different optimization problems. If an optimization problem obtains a good solution, other optimization problems can benefit from it and receive a lasting influence [3-5]. The transfer of genetic material can be reflected in step 3, i.e. the transfer of *skill factor*. To avoid possible damage and achieve considerable performance, this adjustment is applied only for minority individuals. The results of learning can be inspected in step 4.

Algorithm 2. Multitask learning strategy

Input: best individuals for every task in current generation $b_1, b_2, .., b_k$
Output: K individuals.
1. **for each** $b_j, j=1,...,K$
2. Apply a quasi-Newton method for b_j
3. Exchange the *skill factor* with b_r, $r=j+1$
4. Evaluate b_j on optimization problem τ_j only
5. **end for**

C. Local decomposition evolution

In Algorithm 3, a simplified CCGA is suggested. The inputs of Algorithm 3 are two sets, i.e., one (denoted as set 1) contain the individuals that obtain the best fitness for one task in the current generation, and the other (set 2) contains global optimal individuals corresponding to each task. First, the individuals of set 1 are divided into n 1-dimensional subcomponents in line 1. Second, one global optimal individual is chosen randomly from set 2 in line 2. The role of the individual is similar to the context vector in the cooperative co-evolutionary genetic algorithm. Third, the algorithm calculates the fitness of the B_r with the corresponding part replace by the b_k. Finally, the algorithm updates the B_r and put it into the next generation.

In the multitasking optimization, the solver is supposed to acquire the capacity to solve complex problems. Inspired by the CCGA, the problems in the multitasking environment can be decomposed into multiple sub-problems [7, 8]. In MFO, each sub-problem can use a number of evolutionary factors. Therefore, in such environment, different tasks can promote each other's evolution process through the influences of multiple factors.

Algorithm 3. Local decomposition evolution

Input: best individuals for every task in current generation: $b_1, b_2, .., b_k$ and global optimal individuals for every task: $B_1, B_2, .., B_k$
Output: Global optimal individual for one problem
1. $b_1, b_2, .., b_k$ are divided into n 1-dimensional subcomponents
2. Randomly choose B_r, $r \in 1,...,K$
3. **for** $i = 1$ to n **do**
4. **for** $j = 1$ to K **do**
5. Evaluate $O_r = (B_r(i), \ldots Br(i-1), b_k(i), Br(i+1), \ldots Br(n))$
6. **If** $f_j(O_r) < f_j(B_r)$, **then** $B_r(i) = b_k(i)$
7. **end for**
8. **end for**
9. put B_r into P_{t+1}

IV. Numerical Experiment

In this section, the proposed algorithm is compared with MFEA on nine benchmark problems [18]. We adopt the same crossover operator, mutation operator, population size in MFEA and the proposed algorithm. Each algorithm is executed for 30 runs. The maximal number of function evaluations is set to 300,000.

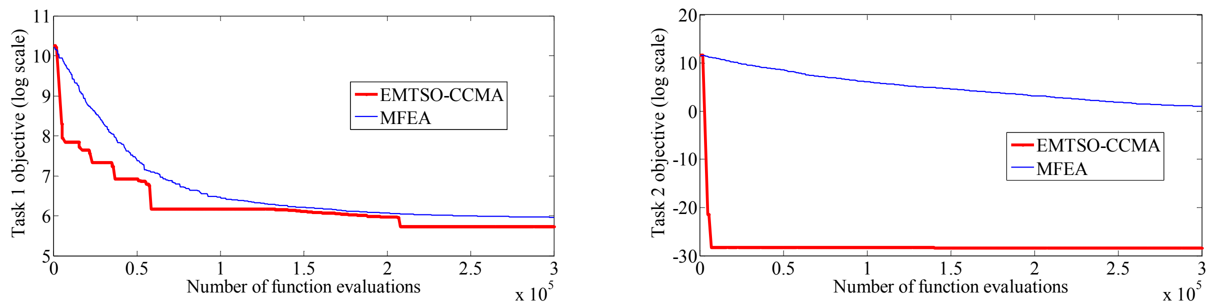
Figs. 2-4 provide the convergence trends of MFEA and EMTSO-CCMA on three representative benchmark problems. Each of the figures represents a benchmark problem consisting of two optimization problems. Generally, EMTSO-CCMA is shown to obtain better performance than MFEA.The statistical analysis is provided for the experiment results. As shown in Table 1, for all the benchmark problems, it presents the mean and the best objective in 30 runs on MFEA and the proposed algorithm. We can see that the proposed algorithm can get better performance in majority of benchmark problems. Specially, the proposed algorithm has substantial improvements in problem 1, problem 5, and problem 7.

V. Conclusion

In this paper, two new methods have been presented to improve the MFEA, which based on cooperative co-evolutionary memetic algorithm. Experimental results of comparative tests are presented for demonstrating the effectiveness of EMTSO-CCMA. Finally, the work that deserves our attention in the future is relieving the issues of MFEA by using the cooperative co-evolutionary algorithm.

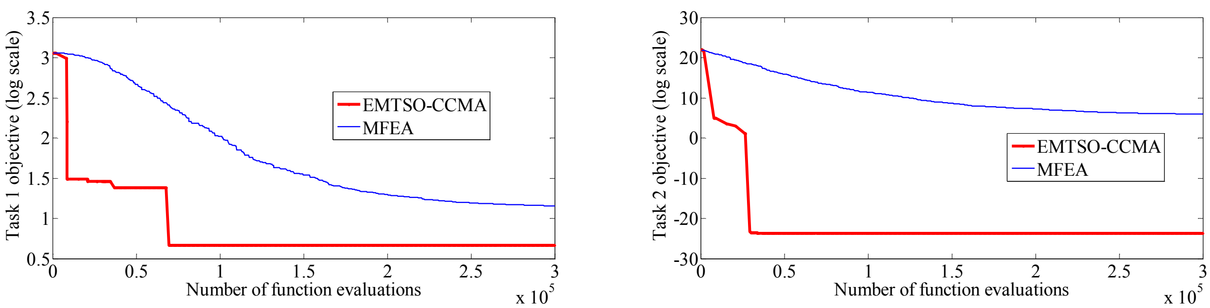
ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China [61471246, 61603259, 61575125], Guangdong Special Support Program of Top-notch Young Professionals [2014TQ01X273, 2015TQ01R453], Guangdong Foundation of Outstanding Young Teachers in Higher Education Institutions [Yq2015141], China Postdoctoral Science Foundation [2016M592536], and Shenzhen Fundamental Research Program [JCYJ20150324141711587, JCYJ20170302154328155].



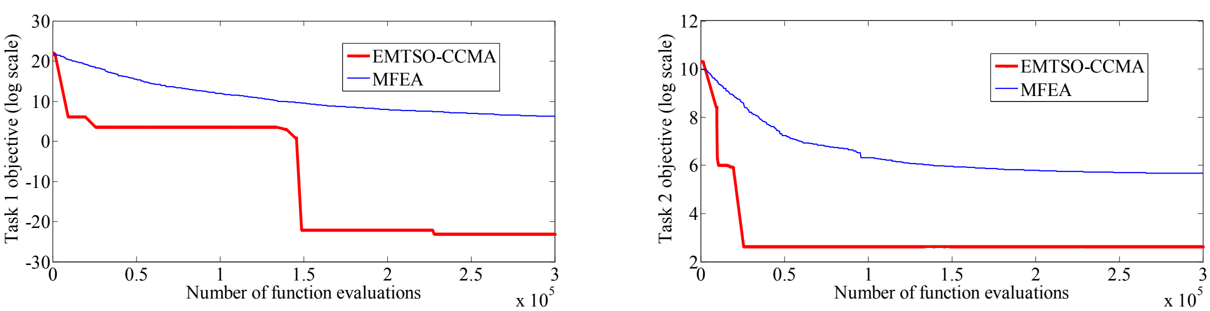
(a) Rastrigin (b) Sphere

Figure 2. Convergence trends for MFEA and EMTSO-CCMA on Rastrigin and Sphere



(a) Ackley (b) Rosenbrock

Figure 3. Convergence trends for MFEA and EMTSO-CCMA on Ackley and Rosenbrock



(a) Rosenbrock (b) Rastrigin

Figure 4. Convergence trends for MFEA and EMTSO-CCMA on Rosenbrock and Rastrigin

TABLE I. PERFORMANCES OF MFEA AND EMTSO-CCMA ON DIFFERENT TASKS

Problem	**Task**	**MFEA**		**EMTSO-CCMA**	
		Best	**Mean**	**Best**	**Mean**
Problem 1	Griewank	0.1552	0.2203-	**0**	**5.0500e-13**
	Rastrigin	78.0593	166.3511-	**0**	**1.2183e-09**
Problem 2	Ackley	2.5295	**3.4302**	**2.01331**	4.73612≈
	Rastrigin	87.5465	**163.8335**	**47.75800**	240.68179≈
Problem 3	Ackley	20.0179	20.0826-	**19.99417**	**20.04930**
	Schwefel	**1997.6938**	**2841.1228**	2405.41330	3962.36350-
Problem 4	Rastrigin	266.0987	405.1913-	**242.76869**	**350.18942**
	Sphere	2.5521	3.7426-	**9.7800e-14**	**3.7480e-13**
Problem 5	Ackley	2.0382	2.8807-	**1.64620**	**2.504153**
	Rosenbrock	219.8812	370.9291-	**4.4192e-12**	**0.38588**
Problem 6	Ackley	**3.2114**	18.8466≈	4.47339	**13.54511**
	Weierstrass	**3.4792**	15.6661-	4.914200	**12.85608**
Problem 7	Rosenbrock	246.9659	414.5092-	**5.4031e-11**	**13.59726**
	Rastrigin	93.6166	241.7448-	**3.0200e-14**	**53.95717**
Problem 8	Griewank	0.1624	0.2546-	**1.3589e-12**	**2.6192e-11**
	Weierstrass	18.6733	24.5647≈	**15.69599**	**19.62791**
Problem 9	Rastrigin	**260.9278**	**421.1149**≈	262.66788	434.32731
	Schwefel	**2166.1976**	**2909.1464**	3183.29350	3905.12568≈

REFERENCES

[1] T. Back, U. Hammel, and H. P. Schwefel, Evolutionary computation: comments on the history and current state: IEEE Press, 1997.

[2] A. Gupta, Y. S. Ong, and L. Feng, "Multifactorial Evolution: Toward Evolutionary Multitasking," IEEE Transactions on Evolutionary Computation, vol. 20, pp. 343-357, June 2016.

[3] L. L. Cavallisforza and M. W. Feldman, "Cultural versus biological inheritance: phenotypic transmission from parents to children. (A theory of the effect of parental phenotypes on children's phenotypes)," American Journal of Human Genetics, vol. 25, pp. 618-637, November 1973.

[4] Feldman, W. Marc, Laland, and N. Kevin, "Gene-culture coevolutionary theory," Trends in Ecology & Evolution, vol. 11, pp. 453-457, November 1996.

[5] C. R. Cloninger, J. Rice, and T. Reich, "Multifactorial inheritance with cultural transmission and assortative mating. II. a general model of combined polygenic and cultural inheritance," American Journal of Human Genetics, vol. 31, pp. 176-198, March 1979.

[6] A. Z. Yi, C. S. Jianhong, D. Y. Xiuxia, and B. S. Pei, "Quasi-Newton iterative learning control and its application," in IEEE International Conference on Automation and Logistics, 2008, pp. 656-660.

[7] X. Ma, Q. Zhang, J. Yang, and Z. Zhu, "On Tchebycheff Decomposition Approaches for Multi-objective Evolutionary Optimization," IEEE Transactions on Evolutionary Computation, vol. PP, pp. 1-1, May 2017.

[8] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, et al., "A Multiobjective Evolutionary Algorithm Based on Decision Variable Analyses for Multiobjective Optimization Problems With Large-Scale Variables," IEEE Transactions on Evolutionary Computation, vol. 20, pp. 275-298, April 2016.

[9] Z. Zhu, Y. S. Ong, and M. Dash, "Wrapper–Filter Feature Selection Algorithm Using a Memetic Framework," IEEE Transactions on Systems Man and Cybernetics--Part B: Cybernetics vol. 37, pp. 70-76, February 2007.

[10] Z. Zhu, J. Zhou, Z. Ji, and Y. H. Shi, "DNA Sequence Compression Using Adaptive Particle Swarm Optimization-Based Memetic Algorithm," IEEE Transactions on Evolutionary Computation, vol. 15, pp. 643-658, October 2011.

[11] Z. Zhu, J. Xiao, S. He, Z. Ji, and Y. Sun, "A multi-objective memetic algorithm based on locality-sensitive hashing for one-to-many-to-one dynamic pickup-and-delivery problem," Information Sciences, vol. 329, pp. 73-89, February 2016.

[12] Y. Sun, M. Kirley, and S. K. Halgamuge, "Extended Differential Grouping for Large Scale Global Optimization with Direct and Indirect Variable Interactions," in Conference on Genetic and Evolutionary Computation, 2015

[13] M. A. Potter and K. A. D. Jong, "A Cooperative Coevolutionary Approach to Function Optimization," Lecture Notes in Computer Science, vol. 866, pp. 249--257, October 1994.

[14] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," Information Sciences, vol. 178, pp. 2985-2999, August 2008.

[15] Y. Shi, H. Teng, and Z. Li, "Cooperative Co-evolutionary Differential Evolution for Function Optimization," Lecture Notes in Computer Science, vol. 3611, pp. 428-428, July 2005.

[16] Z. Zhu, J. Xiao, J. Li, F. Wang, and Q. Zhang, "Global path planning of wheeled robots using multi-objective memetic algorithms," Integrated Computer-Aided Engineering vol. 22, pp. 387-404, 2015.

[17] Z. Zhu, S. Jia, S. He, Y. Sun, Z. Ji, and L. Shen, "Three-dimensional Gabor feature extraction for hyperspectral imagery classification using a memetic framework," Information Sciences, vol. 298, pp. 274–287, 2015.

[18] B. Da, Y. S. Ong, L. Feng, A. K. Qin, A. Gupta, Z. Zhu, C. K. Ting, K. Tang, and X. Yao, "Evolutionary Multitasking for Single-objective Continuous Optimization: Benchmark Problems, Performance Metric, and Baseline Results", arXiv preprint arXiv:1706.03470, 2017