I.J. Education and Management Engineering **2011, 3, 79-86**
Published Online September 2011 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2011.03.12

Available online at http://www.mecs-press.net/ijeme

An Improved Multi-objective Evolutionary Algorithm with the Hybrid Strategies

Gao Guibing[a], Huang Gang[b], Zhang Guojun[c]

[a, b, c]*State Key Laboratory of Digital Manufacturing Equipment & Technology, HUST; Wuhan, China*
[a]*School of Mechanical Engineering, HBUT; Wuhan, China*

Abstract

An improved multi-objective evolutionary algorithm with the hybrid strategies is presented in this paper for multi-objective optimization problems. The evolution process is divided into initial exploration stage, the middle feedback stage and the accelerating convergence stage by the amount of non-dominated individuals in the population. The hybrid strategies and adaptive population structure are employed to improve the behavior of the algorithm at the different stages. The proposed algorithm is validated by 3 benchmark test problems. Compared with three other famous multi-objective algorithms by two quality indicators, the proposed algorithm achieves competitive results.

Index Terms: evolutionary algorithm; multi-objective optimization; hybrid; adaptive;non-dominated solution

1. Introduction

Multi-objective Evolutionary algorithms (MOEA) are suited for tackling multi-objective optimization problems (MOPs) because of its exploration and exploitation ability to find multiple trade-off solutions in the search space. It's well known that evolutionary strategies for improving the behavior of the algorithm e.g., the elite mechanism, the technology of maintaining the diversity of individuals and Pareto dominated all are widely used in NSGA-II[2], SPEA2[3], PESA-II[4] and other MOEAs. The common problem for most MOEAs is the unchanging evolutionary strategy which leads to premature convergence, inefficiency, low convergence speed. Since the construction of the population is different, a certain strategy might be malfunction or attenuation in the different evolutionary process. It seems that employing the adaptive methods is a realistic solution for such problems. The adaptive of algorithm is one of the hotspot in the field of the evolutionary computation. A lot of efficient adaptive mechanisms and technologies such as representation of individuals [2], the dynamic parameter encoding [3], messy genetic algorithms[4], adaptive crossover[5], adaptive probabilities of operators [8,9] and varying population size [10] are proposed. However, most of these adaptive methods and technologies are still

This work is support by the Natural Science Foundation of China (grant no: 50775089), the National High Technology Research and Development Program(863 Program) of China (Grant no:2007AA04Z190)

Corresponding author:
E-mail address: [a] gaoguibing@163.com; [b] huanggang@hust.edu.cn; [c] topsearchgao@hotmail.com

problem-dependant and rely on user's experience and aesthetic preferences [11,12] without the realization of a recursive process routine during the evolution process. An improved multi-objective evolutionary algorithm with the hybrid strategies (IMEAHS) is presented, which aims to improve the performance of the algorithm and adaptively keep the diversity of individuals in the population by using the hybrid strategies.

2. Improved multi-objective evolutionary algorithm with hybrid strategy

2.1 Three evolution stages

In this paper, the evolution process is divided into the initial exploration stage, the middle feedback stage and the accelerating convergence stage according to the value of R_P , R_{ep1} and R_{ep2} respectively. (R_P is the rate of non-dominated individuals in the population, $R_P = \frac{n_n}{n_p}$. R_{ep1} and R_{ep2} are the user setting parameters. R_{ep1} is in the range of [0.15-0.30] and R_{ep2} is in the range of [0.650-0.9]. n_n is the non-dominated individuals in the population. n_p is the size of population).

The initial exploration stage is defined as $R_P < R_{ep1}$. There are few non-dominated solutions at this stage, hence we focus on improving the global exploit capability of the algorithm and preventing the premature convergence. In order to decrease the computation time and improve the global exploit capability, all the individuals in the population including the dominated solutions and non-dominated solutions are selected to construct the next population (see also section 2.2). At this stage, we use NSGA-II that has strong global exploration ability and don't select non-dominated solutions.

The middle feedback stage is defined as $R_{ep1} < R_P < R_{ep2}$. This stage is to improve the boundary exploration ability and the efficiency of selecting the non-dominated individuals. A feedback mechanism which used in the MOCell is employed. The feedback here means some individuals in the external population are selected to construct a new population. With the feedback mechanism, the search ability of the algorithm is enhanced because of the diversity of the population. The population is constructed by (1):

$$N_{Pt+1} = N_{Pt} + N_{Qt} \qquad (1)$$

Where N_{Qt} is the number of individuals randomly selected from the external population, N_{Pt} is the number of individuals selected from the evolution population according to the way of II.B.

The accelerating convergence stage is defined as $R_P > R_{ep2}$. Due to the principal of approximate best[10], the good solutions are similar to each other. Therefore, it is reasonable to use the existing good solutions to generate new solutions. We use the elitism mechanism at this stage because we hope the elite individuals imposing the high selection pressure to speed up the convergence of algorithm. On the other hand, as the amount of non-dominated individuals in the population is mushroom, the relationships among the non-dominated individuals is more complex, it difficult to laminate the population and result the non-dominated sorting method used in NSGA-II[2] is attenuation or malfunction, therefore, we decide to use a fast selection algorithm (see section 2.3) to choose the non-dominated individuals.

Moreover, in order to prevent the degeneration (the non-dominated solutions are cancelled by the evolution operations), the external population is adopted to keep the already obtained non-dominated solutions at the feedback stage and the accelerating convergence stage. If the external population is saturated by the non-dominated individuals, the crown distance mechanism used in NSGA-II is adopted to obliterate the redundant individuals in the population.

2.2 The adaptive of the population

Non-dominated sorting mechanism [2] partitions the population into several levels based on the Pareto dominated relationships among the individuals. Only n individuals located in the upper levels are chose to constitute the population. This method ensures elite individuals with high probability entering the next generation, but it may result in the problem of premature as the elite individuals are excessively used. An improved method for solving this problem is employed in Controlled NSGA-II[3], in which individuals are chose to construct the evolution population according to the attenuation coefficient r from different levels. This method improves the diversity of the population with the fixed attenuation rate r. For the different applications, it may difficult to select the attenuation rate. The improved method is employed that adaptively controls the population by adaptive attenuation coefficient r.

The parent population P_t generated children population Q_t by the evolution operators, $P^{'}_{t+1} = P_t \cup Q_t$, $P^{'}_{t+1}$ is divided into k different subsets $P^{1'}_{t+1}$, $P^{2'}_{t+1}, \cdots$, $P^{k'}_{t+1}$ by the non-dominated sorting method, defines $P^{1'}_{t+1}$ as the top one, the others decreased progressively, and $n^{1'}_{t+1}, n^{2'}_{t+1} \cdots n^{k'}_{t+1}$ is the number of individuals respectively, $n^{1'}_{t+1} + n^{2'}_{t+1} + \cdots + n^{k'}_{t+1} = 2N$,then n_i individuals (calculated by (3)) are selected randomly from the $P^{i'}_{t+1}$ to construct the parent population P_{t+1} .

$$r_t = \frac{n^{1'}_{t+1}}{2 N} \qquad (2)$$

$$n_i = N * \frac{1-r_t}{1-r_t^k} * r_t^{i-1} \; i = 1 \cdots k \quad r_t \neq 1 \qquad (3)$$

In doing so, as the non-dominated individuals in the population is changed in different evolution, the attenuation coefficient adaptively changed. In the initial phase, there are few non-dominated individuals and the value of r_t is small, as a consequence more dominated individuals will be selected from the bottom levels. This procedure improves the global exploration ability of the algorithm. While in the later phase, the non-dominated individuals are mushroom in the population, the value of r_t becomes larger and the few individuals will be selected form the bottom levels, this procedure let the algorithm converge to the Pareto Front.

2.3 the quick select non-dominated individuals and the construct the evolution population

Base on the three-radix quick sorting algorithm[11] and the relationships among the individuals in population, a quick select algorithm is designed to select the non-dominated individuals and construct the parent population.

Define 1: Superior individual. for two individuals x_i , x_j , if x_i dominate x_j , or x_i equal to x_j , x_i does not dominate x_i and x_j does not dominate x_i , then x_i is defined as superior individual for x_j

Define 2: Bad individual. for two individuals x_i , x_j , if x_j dominate x_i , then x_i is defined as bad individual for x_j

Define 3: Superior population. Based on define 1, for the population P_t, if x_i is the reference individual, the population constituted by the individuals that superior to x_i is defined as Superior population

Define 4: Bad population. Based on define 2, for the population P_t, if x_i is the reference individual, the population constituted by the individuals that bad to x_i is defined as bad population

Inference: based on define 1 to define 4, the population P_t could be divided into the Superior population and bad population by any reference individual in P_t.

The method for selecting the non-dominated individuals:

Step 1: Select an individual x_i randomly from the external population as the reference individual, divide the population into two sections: the superior section and the bad section, if the x_i dominated all the individuals in the population, end the operation and continue the evolution otherwise go to the step 2;

Step 2: Select another individual x_j from the superior section as another reference individual, divide the superior section into the superior section and bad sections. If the superior section is only one individual x_j, copy the x_j to the external population, otherwise continue this step until the front section(bad section) is empty, then the individuals in superior section is superior to the individual x_i, as x_i is selected from the external population, the individuals in superior are non-dominated individuals.

Step 3: copy the superior individuals to the external population. If the external population is saturated, use the crowd distance method[2] to delete the redundant individuals.

The method for constructing evolution population

Supposing the population P_t generate children population Q_t, $P'_{t+1} = P_t \cup Q_t$,selecting the N individuals from P'_{t+1} is as follows:

Step 1: a reference individual is selected from the external population randomly, divide evolution population into two sections: the bad sections and the superior sections, counts the number of individual n_{better} and n_{bad} respectively, if $n_{better} > N$ go to Step 3, otherwise go to step 2;

Step 2: select a reference individual as reference from the superior population, divide the superior section into two sections(the superior population and the bad population), count the amount of individuals n_{better} and n_{bad}, if $n_{better} > N$, continue step 2 until $n_{better} < N$, then go to Step 3, if all the individuals in back section are select as the reference individual but still to find $n_{better} > N$, then go to step 4;

Step 3: preserve the superior population obtained by step 2, and sort the bad population by the Pareto non-dominated sort, then select $N - n_{better}$ individuals by the method detailed in 2.2 to construct the population P_{t+1} altogether, end the operator and continue the next evolution.

Step 4: calculate the crowd distance for the individuals in the superior section by the method detailed in 2.3, select N individuals which have the better crowded distance to construct the evolution population P_{t+1}, end of the operation and continue the evolution.

3. Experiment

3.1 Test Instance

Three well-know benchmark multi-objective test problems are selected from the literature. They are Schaffer[12], as well as some diverse complexities problems like the ZDT1[13] problem (which is developed by Zitzler), and the DTLZ2[13] problems (which is defined by Deb et all), (noted : DTLZ2, the amount of objectives function are three).

3.2 Quality Indicator

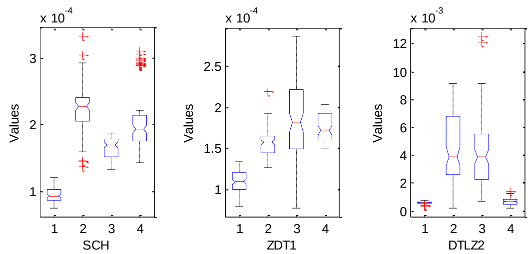
Two quality indicators one Generation Distance[14] for convergence behavior and the other spread (Δ [2])for diversity of solutions are used. Their Pareto Front is obtained by using enumeration search strategy [15].

3.3 Experimental results

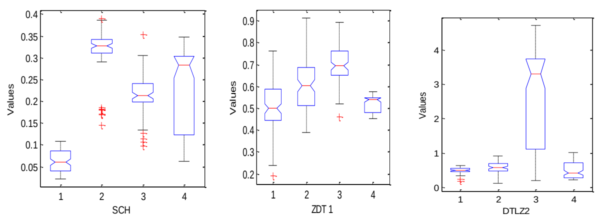
The results of the quality indicators are shown by box-plots[16], the statistical values of GD and Δ for the test samples obtained by IMEAHS, NSGAII, SPEA2, PESAII are shown in fig 1.a and fig 1.b.

Table 1 The number of different algorithms respectively in figure 1

Number	1	2	3	4
Algorithm	IMEAHS	NSGA-II	PESA-II	SPEA2



(a) value of GD



(b) value of spread(Δ)

Fig 1. statistical value of GD and spread for SCH, Kur and ZDT1, ZDT3, DTLZ1, DTLZ2 obtained by IMEAHS, NSGAII , SPEA2, PESA-II and MOCell. The distributions of these samples have been illustrated by the box plots, in a notched-box, a robust estimate of the uncertainty about the medians for box-to-box comparison could be represented by the notches. symbol +denote outliers

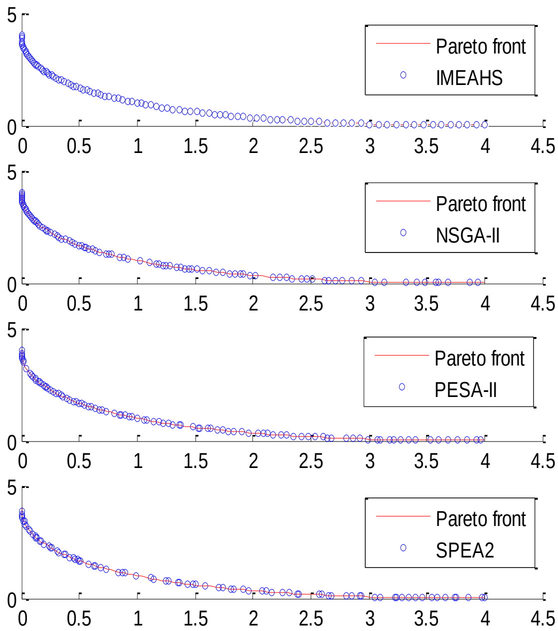
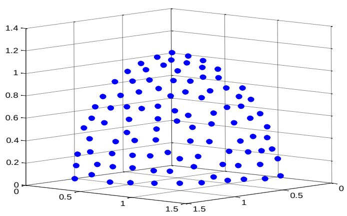


Fig 2. IMEAHS finds a better spread of solutions than SPEA2 , NSGA-II and PESA-II in the ZDT1 problem

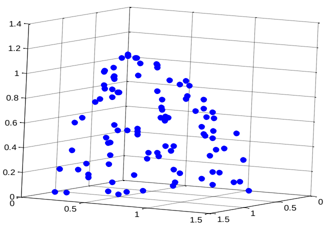
As shown in fig1.a, in terms of GD, IMEAHS has the best performance in Schaffer, ZDT1, DTLZ2, and SPEA2 do the better values in DTLZ2. With regard to Δ, IMEAHS does better than NSGA-II, PESA-II in this experiment, for the problem of DTLZ2, SPEA2 get the best values while the IMEAHS also get the competitive result.

With the aim of giving a complete graphical overview of the behavior of IMEAHS, the Pareto fronts for Schaffer and DTLZ2 are simulated in Fig. 2 and Fig. 3, it is obvious that the IMEAHS obtain the better spread than the other three algorithm on the problem of SCH; IMEAHS and SPEA2 perform better than NSGA-II and PESA-II on the problem of DTLZ2.

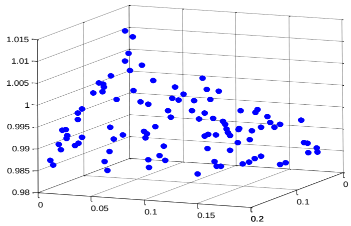
Overall, considering the results of the experiments, it is obvious that IMEAHS is an efficient algorithm in solving MOPs because IMEAHS obtained the competitive values in test problems and it performed very stable in terms of convergence and diversity.



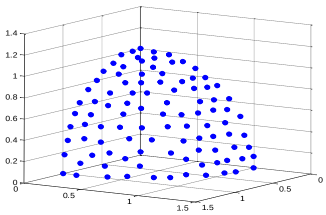
(a) IMEAHS



(b) NSGA-II



(c) PESA-II



(d) SPEA2

Fig 3. IMEAHS and SPEA2 finds a better spread of solutions than NSGA-II and PESA-II in the DTLZ2 problem

4. Conclusion

This research presents an improved multi-objective evolutionary algorithm, called IMEAHS, for dealing with MOPs. The evolution process in IMEAHS is divided into three different stages. A hybrid strategy and the adaptive population structure are introduced to enhance the performance of the algorithm in finding Pareto optimal solutions, while the three-way radix quick sort incorporated to improve the efficiency of selecting non-dominated solutions and the algorithm's convergence speed. The experimental results from three benchmark problems indicate that IMEAHS is a competitive and effective method considering the measurement of convergence and diversity.

REFERENCES

[1]. Coello C.A.C. Evolutionary multiobjective optimization: A historical view of the field [J]. IEEE Computational Intelligence Magazine, **1**; 2006.pp 28–36.
[2]. DEB, K.,A. PRATAP, S. AGARWAL, and T.MEYARIVAN. A fast and elitist multiobjective genetic algorithm: NSGA-II[C]. IEEE Transactions on Evolutionary Computation, 6, 2002a. pp:182–197.
[3]. Deb K, Goel T. Controlled elist non-dominated sorting genetic algorithms for better convergence [C]. 1st

Int Conf on Evolutionary Multi-criterion Optimization. Zurich : Springer-Verlag , 2001 : 67-81.
[4]. SHEN Xiaoning, GUO Yu, CHEN Qingwei, HU Weili. Multi-objective optimization genetic algorithm keeping diversity of population[J]. Control and Decision.Vol. 23 No. 12, 2008 pp 1435-1440
[5]. Jinhua Zheng, et all. A Multi-objective Genetic Algorithm Based on Quick Sort[C], conference on artificial intelligence CAI 2004 Canada , pp:175-186
[6]. Zitzler E., M. Laumanns and L. Thiele. SPEA2: Improving the Strength ParetoEvolutionary Algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35,CH-8092 Zurich, Switzerland. 2001
[7]. CORNE D. W., N. R. JERRAM, J. D. KNOWLES, and M. J. OATES. PESA-II: Region-based selection in evolutionary multiobjective optimization[C]. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), Morgan Kaufmann Publishers, San Francisco, CA, 2001.pp 283–290.
[8]. SHI Ruifeng , ZHOU Hong , SHANG Guan Chunxia. A Hybrid Escalating Multi-objective Evolutionary Algorithm with Its Application to Flow Shop Problems [J]. Systems engineering-Theory & Practice 2006.8 pp101-109
[9]. BAI Zhijiang, LIU Guangzhong. Recursive multiple objective genetic algorithms. Journal of ShanghaiMaritime University[J]. 2007.6 pp62-67
[10]. GLOVER FW, LAGUNA M. Tabu search [M]. Norwell, MA: Kluwer Academic Publishers, 1998.
[11]. Jon L. Bentley and Robert Sedgewick, Fast Algorithms for Sorting and Searching Strings, Proc. 8th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 1997. pp 360-369,.
[12]. J. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in First International Conference on Genetic Algorithms, J. Grefensttete, Ed., Hillsdale, NJ, 1987. pp 93–100.
[13]. E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," Evolutionary Computation, vol. 8, no. 2, 2000. pp 173–195
[14]. Van Veldhuizen, D.A., and G.B. Lamont. On Measuring Multiobjective Evolutionary Algorithm Performance, 2000 IEEE Congress on Evolutionary Computation, 1, 2000. pp 204-211.
[15]. A. J. Nebro, E. Alba, and F. Luna, "Multi-objective optimization using grid computing," Soft Computing, vol. 11, no. 6, 2007. pp 531–540
[16]. McGill R, Tukey JW, Larsen WA. Variations of boxplots. The American Statistician, 1978,32(1):12−16.