Solving Three-objective Optimization Problems Using Evolutionary Dynamic Weighted Aggregation: Results and Analysis

Abstract. In this paper, evolutionary dynamic weighted aggregation methods are generalized to deal with three-objective optimization problems. Simulation results from two test problems show that the performance is quite satisfying. To take a closer look at the characteristics of the Pareto-optimal solutions in the parameter space, piecewise linear models are used to approximate the *definition function* in the parameter space that defines a Pareto-optimal front or the boundary of a Pareto-optimal surface. It is shown that such analyses are very helpful for recovering the true Pareto front.

1 Introduction

Evolutionary multiobjective optimization using the linear weighted aggregation of the objectives has been investigated in the recent years. While most weighted aggregation based methods can get only one Pareto-optimal solution by predetermining the weights using a priori knowledge or simply by trial-and-error, a few attempts have been made to obtain a set of Pareto-optimal solutions in one single run using the weighted aggregation method with the help of evolutionary optimization algorithms. The earliest effort in this direction was reported in [1]. In this approach, the weights are encoded in the chromosome together with the design parameters. Since no search direction exists for the weights in the evolutionary optimization, the weights drift randomly during optimization. A method that explicitly uses random weights during selection for genetic algorithms has been suggested in [2]:

$$w_i = \mathrm{random}_i/(\mathrm{random}_1 + \cdots + \mathrm{random}_n), i = 1, \cdots, n, \tag{1}$$

where random_i is a non-negative real number and n is the number of objectives. If N pairs of parents are selected for generating offspring, N different sets of weights will be specified. A problem in this method is that N sets of weights may not be uniformly distributed and therefore, the search direction of the individuals may overlap. To address this problem, the algorithm has been extended [3], which is known as the cellular multiobjective genetic algorithm (C-MOGA). In the C-MOGA, the weight space is divided uniformly into a certain number of cells and an individual is generated for each cell, whose weights are designated by the cell. Thus, each individual has a unique deterministic weight combination. It should be noticed that with the increase of the objective number, the population size increases exponentially.

A seemingly similar but different approach to multiobjective optimization using random weights has been proposed in [4]. The basic idea is that each individual in the population should have a separate search direction so that the advantages of population-based optimization can be exploited. Consider a two-objective optimization problem and a population of size λ, then the weights for the two objectives are determined as follows:

$$w_1^i(t) \;=\; \mathrm{random}(\lambda)/\lambda, \qquad w_2^i(t) \;=\; 1 - w_1^i(t), \tag{2}$$

where random(λ) generates uniformly distributed random weights between 0 and λ, t is the generation number. This way of weight generation has two features. First, in each generation, there are λ search directions, which are distributed uniformly but randomly in the fitness space. Second, the search direction of each individual changes in each generation. Compared to the C-MOGA algorithm, it has the merit that the population size does not increase exponentially with the number of the objective functions, although for two-objective optimization problems, it is very similar to the C-MOGA. One weakness of this method is that multiple search directions are achieved at the cost of the search efficiency. As shown in [4], the performance of the method using random weights degrades seriously, when the dimensionality of the design parameter space becomes high. This method was originally termed the random weighted aggregation (RWA), however, to distinguish this method from the one used in [2], we now call it the randomly assigned weighted aggregation (RAWA) method. A common drawback of C-MOGA and the RAWA is that they suffer from the fact that the uniform distribution of the weights does not necessarily result in uniformly distributed Pareto solutions.

A more effective weighted aggregation based method has also been suggested in [4, 5]. The method simply changes the weights gradually between 0 and 1 generation by generation. Once they reach an arbitrary point on the Pareto-front, the individuals will move along the Pareto front as the weights change. This has been termed the dynamic weighted aggregation (DWA) method [5]. It has been shown on a number of test functions as well as on real-world applications that the method is able to achieve a set of Pareto-optimal solutions successfully for both convex and concave Pareto fronts.

A special case of the DWA method is the bang-bang weighted aggregation (BWA), where the weights are switched between 0 and 1. Empirical studies have shown that the BWA method exhibits very good performance if the Pareto front is concave.

Theoretical analyses of the above methods reveal that the success of the weighted aggregation based methods can largely be attributed to the following facts:

- The change of the weights is equivalent to the rotation of the Pareto front. In this case, all Pareto solutions, whether they are located in the convex or concave region of the Pareto front, are *dynamically reachable*. In contrast, classical analyses of the weighted aggregation method usually only consider the static stability of the Pareto solutions [6, 7]. It is easy to conceive that a dynamically reachable (thus capturable) Pareto-optimal solution is not necessarily stable.
- A majority of the multiobjective optimization problems are globally convex, which means that most Pareto-optimal solutions are concentrated in a small fraction of the parameter space. Furthermore, the solutions in the neighborhood in the fitness space are also in the neighborhood in the parameter space, and vice versa. This property is also known as the connectedness. In this paper, we will discuss some additional properties of the Pareto-optimal solutions in the parameter space in Section 4.
- The locally causal search property of the evolution strategies. Once the population has reached any point on the Pareto front, the search algorithm can be regarded as "converged". Thus, local search ability is very important for the algorithms to smoothly "scan" the Pareto front point by point. The resolution of the scanning is determined by the speed of the weight change.

In this paper, the weighted aggregation method is employed to solve three-objective optimization problems. Additionally, approximation of the defining function of the Pareto front in parameter space is studied to facilitate the analysis of the true Pareto front in fitness space.

2 Weighted Aggregations for Three-objective Optimization

The three weighted aggregation methods can easily be extended to three-objective problems. For the RAWA, the weights can be generated in the following way:

$$w_1^i(t) = \mathrm{random}(\lambda)/\lambda, \tag{3}$$
$$w_2^i(t) = (1.0 - w_1^i(t))\mathrm{random}(\lambda)/\lambda, \tag{4}$$
$$w_3^i(t) = 1.0 - w_1^i(t) - w_2^i(t), \tag{5}$$

where, random(λ) generates a uniformly distributed random number between 0 and λ, $i = 1, ..., \lambda$, λ is the population size, and t is the generation index. Actually, the population size specifies the resolution in the search space. For example, if the population size $\lambda = 11$, then the weight space will be divided into 11×11 search directions and in each generation, 11 of the 121 search directions will be assigned randomly to the current individuals. In contrast, if the same resolution is used for the C-MOGA algorithm, the population size will be 121.

The dynamic weighted aggregation method can be extended to three-objective problems in a straightforward way. An example of the weight change is as follows:

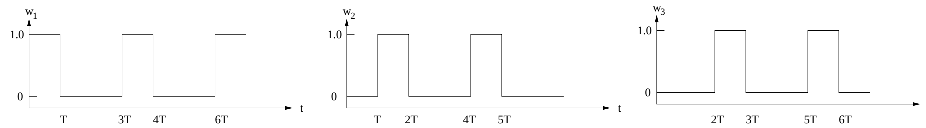


Fig. 1. Possible change of weights of BWA for three-objective optimization.

```
begin
t1=0; t2=0; t=t1+t2;
  for t1=0 to F/2
    w1(t) = |sin(2*PI*t1/F)|;
    for t2=0 to F/2
      w2(t) =(1.0-w1(t)) |sin(2*pi*t2/F)|;
      w3(t) = 1.0-w1(t)-w2(t);
    endf;
  endf;
end
```

Other change modes can of course also be adopted. In the simulation studies, a linear change of the weights is used, refer to Fig. 6.

It is easy to extend the BWA for three-objective optimization. Fig. 1 shows an example of a weight change, where T is a frequency parameter that determines the speed of the change.

3 Simulation Results

The following two test functions have been used in the simulation study [8]. The first test function can be described as follows:

$$f_1 = x_1^2 + (x_2 - 1)^2, \quad f_2 = x_1^2 + (x_2 + 1)^2 + 1, \quad f_3 = (x_1 - 1)^2 + x_2^2 + 2, \tag{6}$$

where $-2 \leq x_1, x_2 \leq 2$. The Pareto solutions of the test function form a convex Pareto surface. The second test function is:

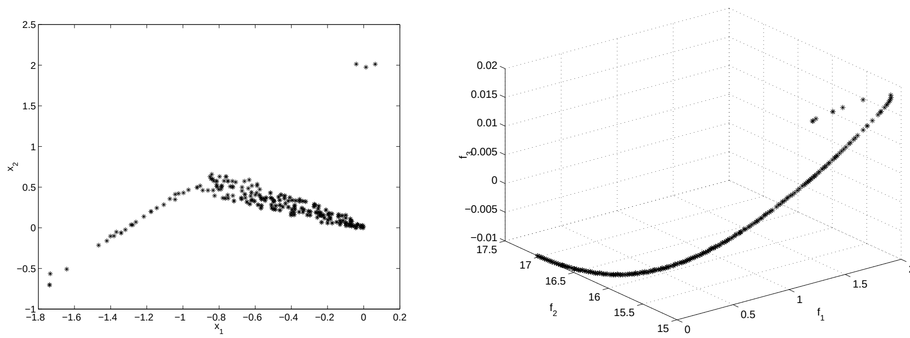
$$f_1 = 0.5(x_1^2 + x_2^2) + \sin(x_1^2 + x_2^2), \tag{7}$$

$$f_2 = \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15, \tag{8}$$

$$f_3 = \frac{1}{x_1^2 + x_2^2 + 1} - 1.1 \ \exp(-x_1^2 - x_2^2), \tag{9}$$

where $-3 \leq x_1, x_2 \leq 3$. The Pareto front of this test function consists of separate pieces of curves, which are hard to distinguish without exact analyses of the true definition function. More details are provided in the next section.

A standard (15,100)-ES with intermediate recombination [9] has been employed for evolutionary optimization. The step-sizes are randomly initialized between 0.1 and 1.0. During optimization, the step-sizes are re-initialized if they are smaller than a prescribed value. This is particularly important when employing the BWA for solving concave Pareto-optimal problems. The reason is that when the population has converged to a stable minimum, the step-sizes will converge to zero very quickly. To assure that the population is able to move from one stable minimum to another after the weights are switched, the convergence of the step-sizes to zero must be avoided. Thus, a relatively large lower bound should be used to prevent the ES from getting trapped in one stable minimum. In this paper, the BWA is employed to solve the second test function. In the simulations, the lower bounds were set to 0.0001 for RAWA and DWA and to 0.01 for BWA. The optimization was run for 500 generations. The RAWA has



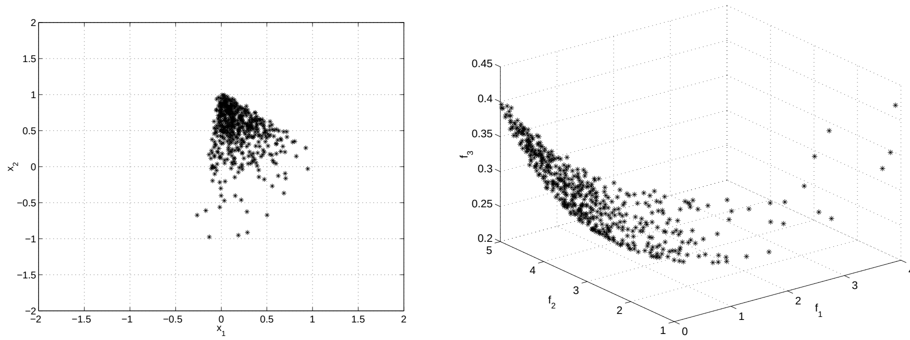
(a) (b)

Fig. 2. Results for test function 1 using RAWA. (a) Parameter space; (b) fitness space.

been employed first to solve the two test functions. As expected, the results are not satisfactory. For the first test function, part of the Pareto set is missing as can be seen in Fig. 2. This also happens for the second test function, refer to Fig. 3.

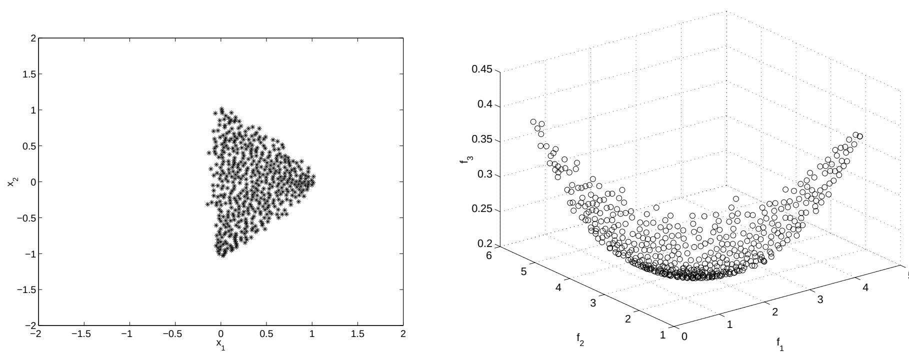
The results obtained for test function 2 using DWA are provided in Fig. 4. The distributions of the solutions in the parameter space as well as in the fitness space are given. The performance of the algorithms is difficult to evaluate from the three-dimensional Pareto surface. Therefore, the three projections of the Pareto surface onto the corresponding two-dimensional planes are also provided in Fig. 5.

The change of the weights are shown in Fig. 6. In the first 20 generations, w_1 and w_2 are fixed to 0 and w_3 is set to 1. From the 21 generation, the weights begin to change. In the first phase, w_1 increases from 0 to 1 by 0.05 each time. Once w_1 is increased, it is fixed temporarily and w_2 changes from 0 to $1.0 - w_1$ and again from $1 - w_1$ to 0, each time by 0.05 in one generation. When this completes, w_1 increases again by 0.05 and the process repeats until w_1 reaches 1. The weights are fixed for 20 generations for $w_1 = 1.0$ and $w_2 = 0$, $w_3 = 0$. In the second phase, w_1 decrease from 1 to 0 and w_2 changes in a similar way as in the first phase. In this way, the Pareto surface is “scanned”. The optimization



(a) (b)

Fig. 3. Results for test function 2 using RAWA. (a) Parameter space; (b) fitness space.



(a) (b)

Fig. 4. Results for test function 2 using DWA. (a) Parameter space; (b) fitness space.

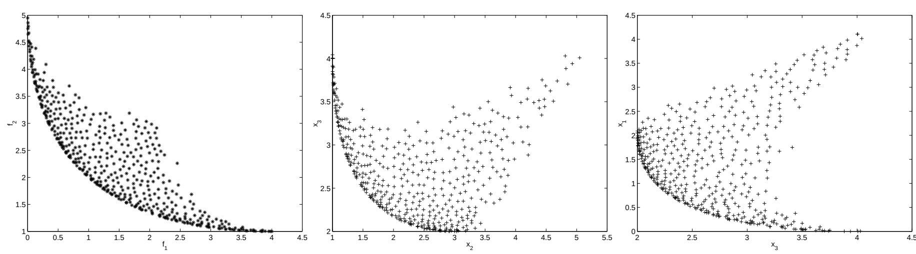

(a) (b) (c)

Fig. 5. Two dimensional projections of Fig. 4(b): (a) f_1-f_2 plane, (b) f_2-f_3 plane and (c) f_3-f_1 plane.

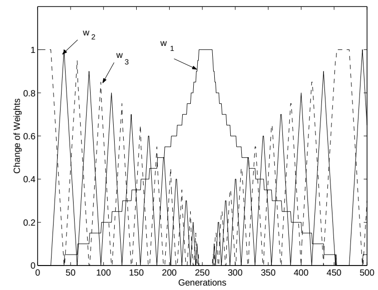
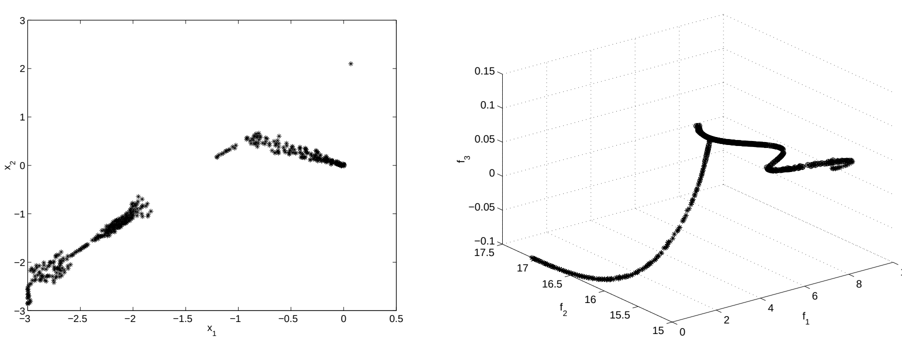


Fig. 6. The change of weights of the DWA method for the first test function.

results for the first test function using the BWA method are illustrated in Fig. 7. The weights have been changed as shown in Fig. 1 and T has been set to 50. It can be seen that the Pareto-optimal solutions are composed of four pieces of linear curves and a separate isolated point in the parameter space, while the three-dimensional Pareto front appears to be one continuous curve. However, a closer look at the solutions reveals that this impression is wrong. This will be discussed further in the next section.



(a) (b)

Fig. 7. Results for test function 1 using BWA. (a) Parameter space; (b) fitness space.

Through the two three-objective test functions, we show that the weighted aggregation methods are capable of solving three-objective optimization problems.

4 Approximation of Definition Functions

One of the important aspects of global convexity is that the Pareto-optimal solutions are concentrated in a very small region of the parameter space. In addition, the distribution of the solutions in the parameter space often shows high regularity. Actually, it is found that in many cases the Pareto solutions in parameters space can be defined by piecewise linear functions, which we have termed definition functions.

In this section, we will analyse the properties of the definition function and the relation between the order of the definition function(s) and the function(s) describing the Pareto surface in fitness space for the two three-objective test functions. We will start by formulating the following two conjectures:

- If the Pareto solutions form a curve in the fitness space, then the definition function is also a curve. If the Pareto solutions form a surface in the fitness space, then the definition function is also a surface.
- The order of the definition function in the parameter space is equal to or lower than that of the function describing the Pareto front. Thus, if the Pareto front consists of lower order curves, the solutions in the parameter space can most probably be described by piecewise linear functions.

If the above conjectures hold for an unknown optimization task, it will be very helpful in getting the true Pareto front from the approximated solutions obtained using multiobjective optimization algorithms. In the following, approximate models for the definition function will be constructed for both test problems on the basis of the previous conjectures. It will be shown that in both cases, the accuracy of the solutions will be improved significantly. Furthermore, missing solutions can be found through such analysis.

4.1 Test Function 1

The regularity of the distribution of the solutions in the parameter space can easily be observed. We find that the boundary of the solution region can be defined by the following three lines:

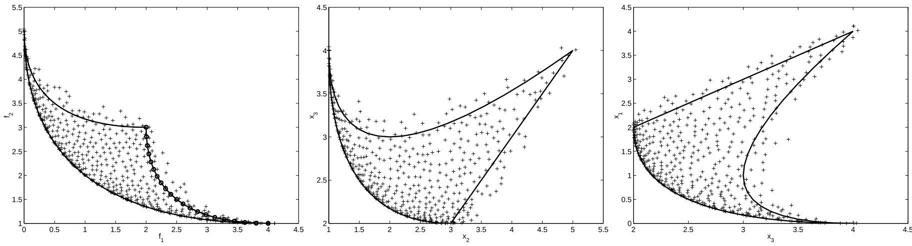
$$\begin{array}{lll} x_2 = x_1 - 1.0, & x_2 = -x_1 + 1.0, & x_1 = 0, \\ 0 \le x_1 \le 1, & 0 \le x_1 \le 1, & -1 \le x_2 \le 1. \end{array} \tag{10}$$

Using these linear curves, it is straightforward to get the boundary of the true Pareto surface, as shown in Fig. 8, where the approximated solutions are also provided.

In this way, it is easy to get rid of the solutions that are not Pareto-optimal. Meanwhile, it is interesting to find out that the boundary of the region in the parameter space corresponds nicely to the boundary of the Pareto surface.

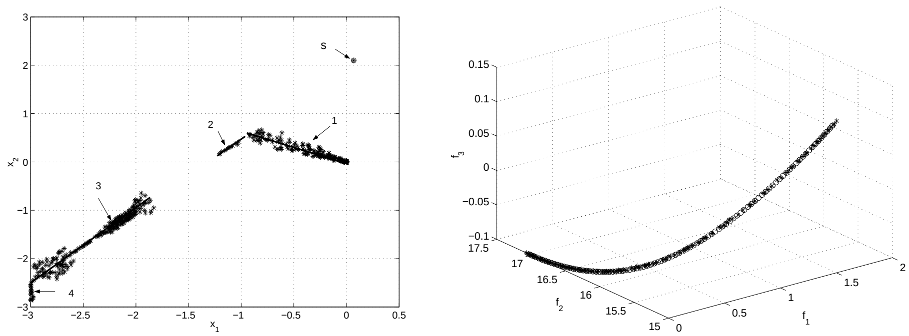
4.2 Test Function 2

From the distribution of the obtained solutions in the parameter space, as found in [10–12], the definition function of this test problem consists of more than one linear section plus a separate point, while the Pareto front appears to be a continuous curve (albeit it can be seen in [11], that the Pareto front is composed of two disconnected pieces, however, one piece of the Pareto optimal section is



(a) (b) (c)

Fig. 8. Boundary of the true Pareto surface. (a) f_1-f_2. (b) f_2-f_3 and (c) f_3-f_1.



(a) (b)

Fig. 9. (a) Distribution of the solutions and the approximation in the parameter space and (b) Pareto front generated by line section 1.

missing). The four sections together with their approximation shown in Fig. 9(a) can be described by:

$$\text{S1: } x_2 = 0.001 - 0.62x_1,\ -0.95 \leq x_1 \leq 0, \tag{11}$$

$$\text{S2: } x_2 = 1.94 + 1.47x_1,\ -1.2 \leq x_1 \leq -0.95, \tag{12}$$

$$\text{S3: } x_2 = 1.94 + 1.47x_1,\ -3 \leq x_1 \leq -1.95, \tag{13}$$

$$\text{S4: } x_2 = 30.8 + 11.19x_1,\ -3 \leq x_1 \leq -2.89. \tag{14}$$

Now we try to reproduce the Pareto front using the approximate definition functions S1-S4. The Pareto front generated by line section 1 is shown in Fig. 9(b) and the two sections of the Pareto front generated by line sections 2 and 3 are illustrated in Fig. 11(a)-(c).

If we take a closer look at the solutions reconstructed from the approximate definition functions, we find out that the Pareto front has much richer features than what has been obtained in existing work [11, 10, 12]. The Pareto solutions generated by line section 2 are so close to those generated by line section 3 that it is difficult to distinguish them without zooming in.

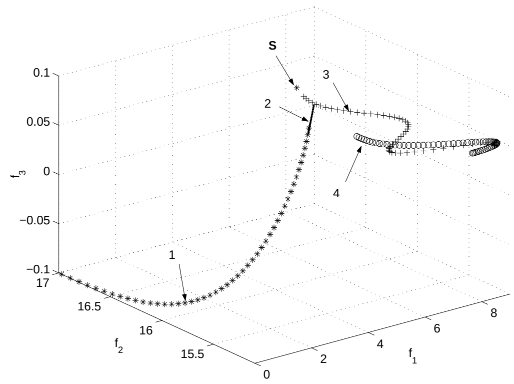


Fig. 10. The Pareto front reconstructed from the approximate models of the definition function.

The solutions generated by line section 4 are shown in Fig. 11(d)-(f). To get a better idea of the curve, it is shown together with the solutions generated by line section 2 and 3. It should be pointed out that in the f_3-f_1 projection, there is an overlap of the Pareto front (between point A and B).

Finally, the complete Pareto front reconstructed from the approximate definition function in the parameter space is presented in Fig. 10. Compared to the results shown in Fig. 7(b), it can be seen that the Pareto-optimal solutions generated by line section 4 have neither been found by the BWA optimization algorithm nor by previously published works. Additionally, due to the complexity of the Pareto front, the omission of parts would not have been detected without the approximation of the definition functions on the basis of the two conjectures.

5 Discussions and Conclusions

The main purpose of this paper is twofold: first, to demonstrate the successful extension of the weighted aggregation based approaches to three-objective problems and second, to show that the distribution of the Pareto-optimal set exhibits surprising regularity and simplicity in the parameter space. This property could be more interesting and helpful for the identification of the *complete* Pareto front than global convexity. By taking advantage of such regularities, it is possible to build simple models from the obtained Pareto-optimal solutions for the approximation of the definition function. Such an approximate model can be of great significance in the following aspects.

- It allows to get more accurate, more complete Pareto solutions from the approximate solutions as shown in Section 4.
- It alleviates many difficulties in multiobjective optimization. If the whole Pareto front can be reconstructed from a few Pareto solutions, then many requirements on the optimizer can be alleviated, e.g., a uniform distribution is no more important in approximating Pareto-optimal solutions.

In the design optimization and operations research communities, research has been reported to approximate the Pareto front [13]. However, the final target of optimization is not to get the Pareto front itself, but the Pareto-optimal set. Besides, it has been argued that the definition function in the parameter space is usually of lower complexity than the Pareto front. Of course, the approximation becomes harder when the dimension increases. Nevertheless, if a lower order polynomial is able to represent the definition function, the approximation is still practical.

The result on the approximation of the definition function described in this paper is still preliminary. Further work should be carried out to check the conjectures on multiobjective optimization problems with a higher design space.

References

1. P. Hajela and C. Y. Lin. Genetic search strategies in multicriteria optimal design. *Structural Optimization*, 4:99–107, 1992.
2. T. Murata and H. Ishibuchi. Multi-objective genetic algorithm and its application to flow-shop scheduling. *International Journal of Computers and Engineering*, 30(4):957–968, 1996.
3. T. Murata, H. Ishibuchi, and M. Gen. Specification of genetic search directions in cellular multi-objective genetic algorithms. In L. Thiele K. Deb and E. Zitzler, editors, *Proceedings of First International Conference on Evolutionary Multi-Criterion Optimization*, pages 82–95. Springer, 2001.
4. Y. Jin, T. Okabe, and B. Sendhoff. Adapting weighted aggregation for multi-objective evolution strategies. In L. Thiele K. Deb and E. Zitzler, editors, *Proceedings of First International Conference on Evolutionary Multi-Criterion Optimization*, pages 96–110. Springer, 2001.
5. Y. Jin, M. Olhofer, and B. Sendhoff. Evolutionary dynamic weighted aggregation for multiobjective optimization: Why does it work and how? In *Genetic and Evolutionary Computation Conference*, pages 1042–1049, 2001.
6. I. Das and J.E. Dennis. A closer look at drawbacks of minimizing weighted sum of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14(1):63–69, 1997.
7. A. Messac, G. Sundararaj, R. Tappeta, and J. Renoud. Ability of objective functions to generate points on non-convex Pareto frontiers. *AAAI Journal*, 38(6):1084–1091, 2000.
8. R. Viennet, C Fonteix, and I. Marc. Multicriteria optimization using genetic algorithms for determining a pareto set. *International Journal of Systems Science*, 27(2):255–260, 1996.
9. H.-P. Schwefel. *Numerical Optimization of Computer Models*. John Wiley, 1981.
10. K.C. Tan, E.F. Khor, C.M. Heng, and T.H. Lee. Exploratory multi-objective evolutionary algorithms: Performance study and comparisons. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 647–654. Morgan Kaufmann, 2001.
11. K. Deb. *Multi-objective Optimization Using Evolutionary Algorithms*, chapter 8, page 331. Wiley, 2001.
12. C. Coello Coello, D. Van Veldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*, chapter 3, page 114. Kluwer, 2002.
13. B. Schandl, K. Klamroth, and M. Wiecek. Norm-based approximation in bicriteria programming. *Computational Optimization and Applications*, 20:23–42, 2001.

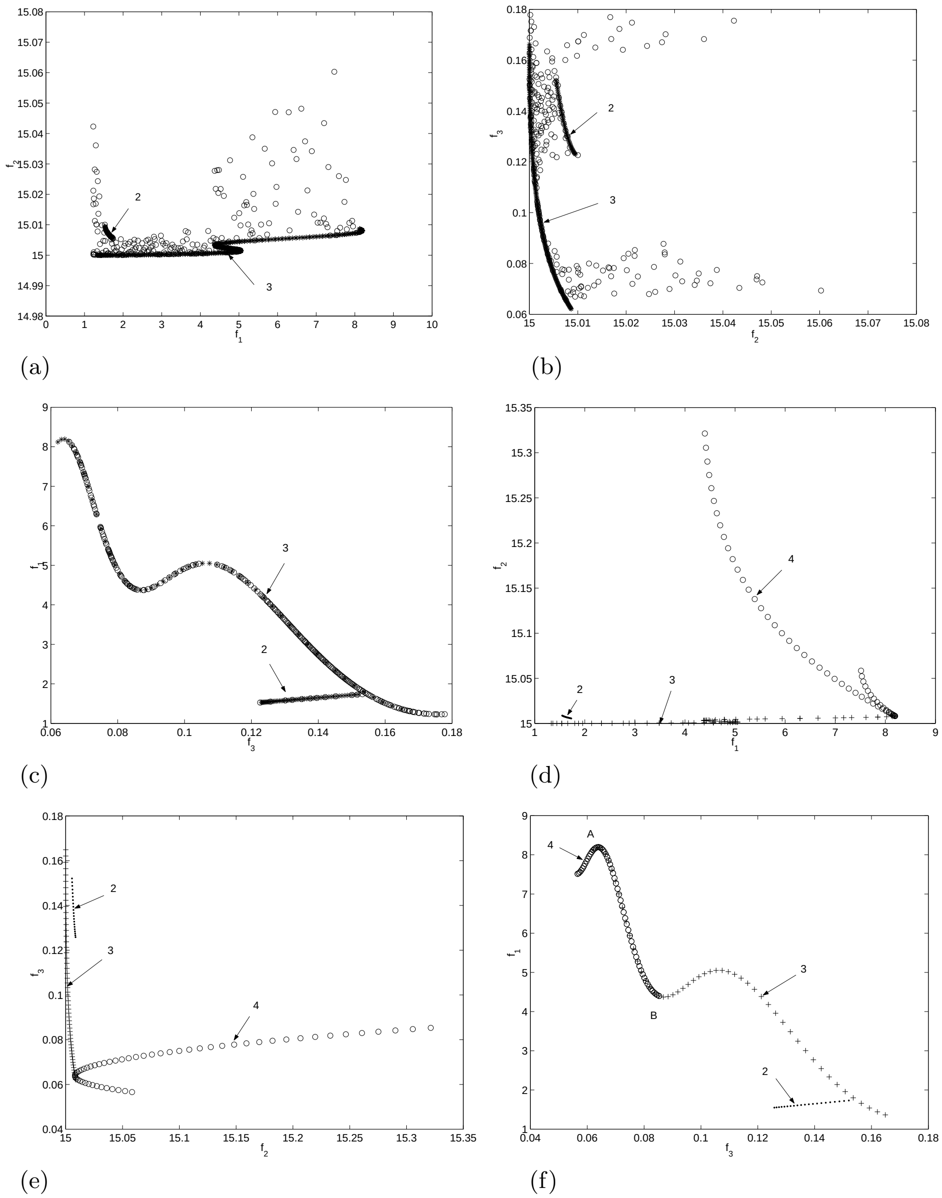


Fig. 11. Pieces of Pareto front generated by line section 2 and 3: (a) f_1-f_2, (b) f_2-f_3 and (c) f_3-f_1. Pieces of Pareto front generated by line section4 together with those generated by line sections 2 and 3: (d) f_1-f_2, (e) f_2-f_3 and (f) f_3-f_1.