# Effectiveness of Weighted Aggregation of Objectives for Evolutionary Multiobjective Optimization: Methods, Analysis and Applications

Yaochu Jin
Future Technology Research
Honda R&D Europe (D) GmbH
63073 Offenbach/Main, Germany
Email: yaochu.jin@ieee.org

**Abstract**

Multiobjective optimization using the conventional weighted aggregation of the objectives method is known to have several drawbacks. In this paper, multiobjective optimization using the weighted aggregation method is approached with the help of evolutionary algorithms. It is shown through a number of test functions that a Pareto front can be achieved from one single run of evolutionary optimization by changing the weights during optimization, no matter whether the Pareto-optimal front is convex or concave. To establish a theoretical background that accounts for the success of the method, the phenomenon of *global convexity* in multiobjective optimization is investigated. Global convexity means that 1) most Pareto-optimal solutions are concentrated in a very small fraction of the parameter space, and 2) the solutions that are in the neighborhood on the Pareto front are also in the neighborhood in the parameter space. It is shown that all test functions studied in this paper do exhibit the global convexity property. In fact, for the two or three dimensional problems studied in the paper, the Pareto-optimal solutions can be defined by a piecewise linear function in the parameter space. Furthermore, the mapping of a normal distribution in the parameter space onto the fitness space is investigated. It is shown that the Pareto front is sometimes a local attractor even without the influence of selection. Finally, two application examples in design optimization are given to show the effectiveness of the evolutionary dynamic weighted aggregation method.

## I. INTRODUCTION

Multiobjective optimization using weighted aggregation of the objectives is conceptually straightforward and computationally efficient. Mathematically, the weighted aggregation method can be expressed as follows:

$$F(\boldsymbol{x}) = \sum_{i=1}^{m} w_i f_i(\boldsymbol{x}), \tag{1}$$

where $f_i$ is the $i$-th objective, $m$ is the number of the objectives to be optimized, $w_i \geq 0$ with $\sum_{i=1}^{m} w_i = 1$ is the weight assigned to objective $f_i$ and $\boldsymbol{x}$ is the $n$-dimensional vector of the design parameters. It has been shown in [1] that for every Pareto-optimal solution of a convex problem, there exists a positive weight such that this solution is an optimum of $F(\boldsymbol{x})$. However, it is widely known that the conventional weighted aggregation method has three main drawbacks. First, only one Pareto-optimal solution can be

obtained from one single run of optimization. That is to say, to achieve a set of Pareto solutions, one has to run the optimization for a number of times. Second, one can get a Pareto solution using weighted aggregation method only if the solution is located in the convex region of the Pareto front. Third, a lot of Pareto fronts are not uniform, which means that the set of obtained solutions are not evenly distributed on the Pareto front given a set of evenly distributed weights.

A variety of methods have been proposed to address the first weakness of the weighted aggregation method, most of which use evolutionary algorithms. In [2], the weight of the objectives are encoded in the chromosome so that more than one Pareto-optimal solution can be obtained in one single run. To prevent the weights from converging to some particular values, a sharing method is used for the weights. In addition, mating restrictions are suggested. In [3], the weight value of a selected parent individual is generated randomly. A shortcoming of this method is that the randomly generated weights are not necessarily uniformly distributed. To fix this problem, additional measures are taken to specify the search direction [4]. To take advantage of the population-based search using evolutionary algorithms, uniformly distributed random weights are generated for the individuals within a population in [5]. In this way, the evolutionary algorithm is able to obtain a set of Pareto-optimal solutions. An interesting method, which is called dynamic weighted aggregation, has also been suggested in [5]. In this method, the weights are changed generation by generation during the evolutionary optimization. With the help of an archive, the Pareto-optimal front can be approximated. This method has been further studied and extended in [6]. It is found that the evolutionary algorithm is able to approximate the Pareto-optimal front by switching the weights between 0 and 1, if the Pareto front is concave.

The second weakness of the weighted aggregation method has also been addressed. The mathematical properties of the weighted aggregation method have been investigated to reveal the reason why the method can only achieve the Pareto-optimal solutions that are located in the convex region. In [7], a geometrical explanation for this phenomenon is presented. The concept of the admissible aggregate objective function is introduced in [8]. An aggregated objective function $F(\boldsymbol{x})$ is admissible if and only if it is a coordinate-wise increasing function of the objectives $f_i(\boldsymbol{x})$, i.e., $F(\boldsymbol{x})$ is a strict monotonically increasing function of each objective $f_i(\boldsymbol{x})$. Analytical conditions for capturing a Pareto-optimal solution using the weighted aggregation method are given. It is found that the curvature of the aggregated objective function must be increased to achieve Pareto-optimal solutions in the concave region of the Pareto front. Therefore, the weighted compromise programming method [9, 10] is able to capture Pareto-optimal solutions in the

concave region:

$$F(\boldsymbol{x}) = \sum_{i=1}^{m} w_i \{f_i(\boldsymbol{x})\}^{s_i}, \tag{2}$$

where, $s_i > 1$ is a positive integer. Usually, $f_i$ needs to be scaled so that it is always positive. Alternatively, the exponential weighted method [11] can also be used to control the curvature:

$$F(\boldsymbol{x}) = \sum_{i=1}^{m} w_i \ \exp(s_i f_i(\boldsymbol{x})). \tag{3}$$

Other compromising programming methods [12, 13] are also able to obtain Pareto-optimal solutions in the concave region. It has also been shown that the evolutionary weighted aggregation methods developed in [5, 6] are able to achieve Pareto-optimal solutions in the concave region.

To obtain a collection of evenly distributed Pareto-optimal solutions, a method called normal-boundary intersection has been proposed in [14]. It is proven that the obtained Pareto-optimal solutions are evenly distributed independent of the relative scales of the objectives.

In this paper, the evolutionary dynamic weighted aggregation (EDWA) methods proposed in [5, 6] are re-examined. Several convex or non-convex benchmark problems are tested to verify the effectiveness of the methods. To gain an insight into the theoretical background of the methods, *global convexity* of multiobjective optimization problems is investigated empirically.

Global convexity is not the convexity in the strict sense. In single objective optimization, global convexity implies that local optima are concentrated in a very small region of the parameter space [15, 16]. In [17], global convexity is studied for a combinatorial multiobjective optimization: the multiobjective traveling salesman problem. It is found that the local optima of the combined scalar objective function exhibit the global convexity. In fact, this phenomenon is also known as *connectedness* in combinatorial multiobjective optimization, where the connectedness of the efficient solutions are investigated in a topological sense [18].

In this work, we investigate the global convexity of continuous multiobjective optimization problems. In continuous multiobjective optimization, global convexity mainly has two aspects: 1) Most Pareto-optimal solutions are concentrated in a very small fraction of the parameter space. 2) If two Pareto-optimal solutions are in the neighborhood on the Pareto front, then they are also in the neighborhood in the parameter space. The first feature can be investigated by visualizing the distribution of the Pareto-optimal solutions in the parameter space for low-dimensional problems. It is shown that for the test functions studied in the paper (most of which are widely used benchmark problems), the Pareto-optimal solutions are gathered in very small regions in the parameter space, which can be defined by a simple

piecewise linear function. The second aspect, which we term it *neighborhoodness*, can be investigated by observing the relationship between the Manhattan distance on the Pareto front and that in the parameter space. It is shown that all test problems studied in the paper do exhibit the property of neighborhoodness.

The evolutionary search in the EDWA can mainly be divided into two stages. In the first stage, the exploration stage, the evolutionary algorithm tries to reach any point on the Pareto front. In the second stage, the exploitation stage, the evolutionary algorithm takes advantage of the global convexity and locates the Pareto-optimal solutions that are in the neighborhood. In this stage, the optimization algorithm has to carry out a strongly causal search. In evolutionary algorithms, strong causality means that small variations in the genotype space due to mutation imply small variations in the phenotype space. It is shown in [19] that evolution strategies (without recombination) satisfy this strong causality condition. We believe that the global convexity and the strong causality of evolution strategies are the main reasons for the success of the EDWA method. In fact, a strongly causal search is also believed to be important in single objective optimizations [20, 21].

Apart from the global convexity of multiobjective optimizations, it is also found that for some test functions, the Pareto front is a *natural local attractor* in the fitness space without selection pressure. This is investigated by mapping a normal distribution in the parameter space onto the fitness space. It is shown that for some test functions, the solutions mapped from a normal distribution in the parameter space are strongly biased toward the Pareto front, which means that the Pareto front is a local attractor in the fitness space.

The method studied in this paper does not belong to the main stream of Pareto-based approaches to evolutionary multiobjective optimization. One advantage of the EDWA method over the Pareto-based approach is that most single objective evolutionary algorithms can be directly employed for multiobjective optimization with minor changes. Nevertheless, the main purpose of the paper is to demonstrate the fact that the major weaknesses of the conventional weighted aggregation method can be overcome with the help of evolutionary algorithms. In addition, this paper intends to discuss the global convexity, which we believe is a very interesting facet of multiobjective optimization that has not been sufficiently investigated in evolutionary multiobjective optimization. It is beyond the scope of the present paper to compare the EDWA method with Pareto-based methods. Refer to [22, 23] for a comprehensive review of evolutionary multiobjective optimization.

The remainder of this paper is organized as follows. In Section II, the dynamic weighted aggregation method is briefly described. After that, the method is applied to a number of test problems to show its

effectiveness. Theoretical analysis of the method, mainly concentrating on the global convexity property, is provided in Section IV. Two application examples of multiobjective optimization are presented in Section V. Concluding remarks are provided in Section VI.

## II. EVOLUTIONARY DYNAMIC WEIGHTED AGGREGATION METHOD (EDWA)

### A. Dynamic weighting

The central idea behind the dynamic weighted aggregation method is to change the weights during the evolutionary optimization. A straightforward way of doing this is to assign a number of uniformly distributed random weights to the individuals in the population [5]. In other words, each individual has a different set of weights that are generated from a uniform random distribution. If there are two objectives and $N$ denotes the population size, the random weights can be generated as follows:

$$\begin{aligned} w_1^i(t) &= \text{random}(N)/N, \\ w_2^i(t) &= 1.0 - w_1^i(t), \\ & \quad i = 1, 2, \cdots, N, \end{aligned} \tag{4}$$

where function $random(N)$ generates a uniformly distributed random number between 0 and $N$, $i$ is the index for individuals and $t$ the generation index. This means that the uniformly distributed random weights need to be regenerated in each generation. This has been termed randomly weighted aggregation (RWA) method in [6].

A more effective approach is to change the weights gradually from generation to generation during optimization, which is called dynamic weighted aggregation (DWA) method [6]. Although there are no specific requirements on how the weights should be changed, the following two factors should be taken into account:

- The weights should be changed periodically.
- The period of the weight change should be large enough so that the evolutionary optimization is able to converge. Empirically, the weights should be allowed to change between 0 and 1 at least twice in one single run. For example, the weights can be changed in the following way for two-objective problems:

$$\begin{aligned} w_1(t) &= |\sin(2\pi t/F)| \\ w_2(t) &= 1.0 - w_1(t), \end{aligned} \tag{5}$$

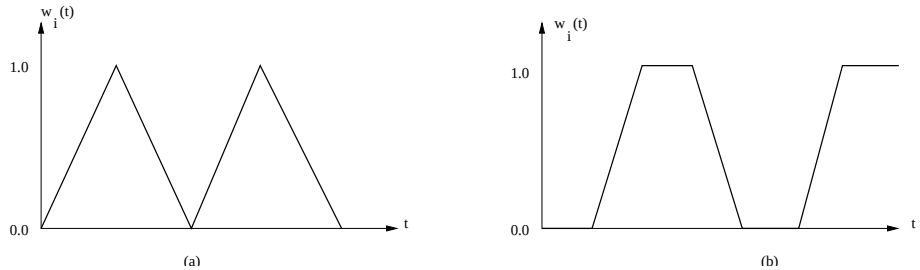



Figure 1: Continuous weight change.

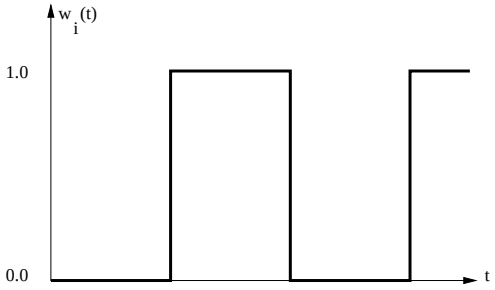



Figure 2: Switch the weights between 0 and 1.

where $t$ is the generation index, $|x|$ takes the absolute value of $x$, $F$ is the weight change frequency, usually between 50 and 200. Note that if $F = 100$, the weights can be changed twice between 0 and 1 four times within 100 generations.

Fig. 1 shows two additional possible ways of weight change.

Interestingly enough, it has also been found that a Pareto front can be approximated by switching the weights between 0 and 1 (refer to Fig. 2), if the Pareto front is concave [6]. The method has been termed bang-bang weighted aggregation (BWA) method.

In contrast to the DWA, random weight change in every generation in RWA seems to make the evolutionary search less efficient, which degrades the performance seriously when the dimensionality of the search space is high. In this paper, only the periodical weight change method (DWA) will be discussed in the following sections.

The extension of the weight change methods in equations (4) and (5) to optimization problems with more than two objectives is theoretically straightforward. For example, if there are three objectives,

equation (4) can be rewritten as:

$$\begin{aligned} w_1^i(t) &= \text{random}(N)/N, \\ w_2^i(t) &= (1.0 - w_1^i(t))\ \text{random}(N)/N, \\ w_3^i(t) &= 1.0 - w_1^i(t) - w_2^i(t), \\ & \quad i = 1, 2, \cdots, N. \end{aligned} \tag{6}$$

If there are three objectives, the DWA method in equation (5) can be modified in the following way.

Let $t_1 = 0$, $t_2 = 0$, and $t = t_1 + t_2$
**for** $t_1 = 0$ to $F/2$
  $w_1(t) = |\sin(2\pi t_1/F)|$;
  **for** $t_2 = 0$ to $F/2$
    $w_2(t) = (1.0 - w_1(t))|\ \sin(2\pi t_2/F)|$;
    $w_3(t) = 1.0 - w_1(t) - w_2(t)$;
  **end**;
**end**;

Note that all weights should be changed smoothly so that the population does not need to jump from one point to another on the Pareto front, which should be avoided if the Pareto front is convex. Furthermore, the number of generations required increases exponentially with the number of objectives if the DWA is used. This is caused by the phenomenon known as "the curse of dimensionality".

Finally, it should be pointed out that the population cannot keep all the Pareto solutions that have been found so far. Therefore, it is necessary to maintain an archive to store the Pareto solutions.

B. Evolution Strategy

A standard evolution strategy (ES) [24] has been employed for evolutionary multiobjective optimization using the dynamic weighted aggregation method. In the standard ES, mutation of the objective parameters is carried out by adding a normally distributed random number denoted as $N(0, \sigma_i^2)$ with variance $\sigma_i^2$. The $\sigma_i$ are also known as the step-sizes and are encoded in the genotype and are subject to mutations. A standard ES can be described as follows:

$$\sigma_i(t) = \sigma_i(t-1)\exp(\tau' z)\exp(\tau z_i) \tag{7}$$

$$\mathbf{x}(t) = \mathbf{x}(t-1) + \tilde{\mathbf{z}}, \tag{8}$$

where $\mathbf{x}$ is an $n$-dimensional parameter vector to be optimized, $\tilde{\mathbf{z}}$ is an $n$-dimensional random number

vector with $\tilde{\mathbf{z}} \sim N(\mathbf{0}, \boldsymbol{\sigma}(t)^2)$, $z$ and $z_i$ are normally distributed random numbers with $z, z_i \sim N(0,1)$. The parameters $\tau$, $\tau'$ and $\sigma_i$ are the strategy parameters, where $\sigma_i$ are mutated as in equation (7) and $\tau$, $\tau'$ are usually chosen as follows:

$$\tau = \left(\sqrt{2\sqrt{n}}\right)^{-1}; \ \tau' = \left(\sqrt{2n}\right)^{-1} \tag{9}$$

Several recombination methods have been proposed for the ES. The discrete recombination, the intermediate recombination and the panmictic recombination are three widely used methods. In this work, no recombination operation is implemented.

In the ES, usually a deterministic selection method is used. The best $\mu$ individuals are either selected from the $\lambda$ offspring, which is known as the comma selection, or from a combination of the $\mu$ parent and the $\lambda$ offspring, which is known as plus selection.

## III. EMPIRICAL EVALUATION OF THE METHOD

### A. Test functions

To show the effectiveness of the method, the following four test functions have been used.

- $F_1$ is the Schaffer's function [25]. Its Pareto front is convex and uniform.

$$f_1 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 \tag{10}$$

$$f_2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - 2.0)^2. \tag{11}$$

- $F_2$ is one of the test function used in [26]. Its Pareto front is concave and non-uniform.

$$f_1 = x_1 \tag{12}$$

$$g(x_2, ..., x_n) = 1.0 + \frac{9}{n-1}\sum_{i=2}^{n} x_i \tag{13}$$

$$f_2 = g \times (1.0 - (f_1/g)^2). \tag{14}$$

- $F_3$ has been studied in [27], which has a concave Pareto front.

$$f_1 = 1. - \exp\left\{-\sum_{i=1}^{n}\left(x_i - \frac{1}{\sqrt{n}}\right)^2\right\} \tag{15}$$

$$f_2 = 1. - \exp\left\{-\sum_{i=1}^{n}\left(x_i + \frac{1}{\sqrt{n}}\right)^2\right\}. \tag{16}$$

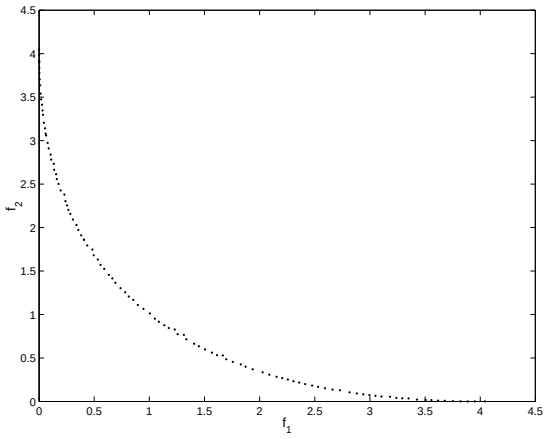



Figure 3: Achieved Pareto front for test function $F_1$.

- $F_4$ is an extension of the test function in [28]. The original problem is one-dimensional and bi-objective, and it is very easy to get the Pareto front using evolutionary algorithms, because every point is Pareto optimal. The modified problem is two-dimensional, and the resulting Pareto front is non-convex. Furthermore, although the Pareto front is continuous, the region that defines the Pareto front in the parameter space is disconnected, which makes it non-trivial for the EDWA method. This will be discussed further in Section IV.

$$f_1 = \exp(-x_1) + 1.4\,\exp(-x_1^2) + \exp(-x_2) + 1.4\,\exp(-x_2^2) \quad (17)$$

$$f_2 = \exp(x_1) + 1.4\,\exp(-x_1^2) + \exp(x_2) + 1.4\,\exp(-x_2^2) \quad (18)$$

For the convenience of analysis, the dimension of all test problems is set to 2 in the following study. Problems with a higher dimensionality are studied in the application examples.

B. Simulation results

A standard ES [24] with comma selection is adopted in the simulations. The parent and offspring population sizes are set to 15 and 100, respectively. The initial step-sizes are set to 0.1 and 400 generations are run for each test function. The weights are changed periodically according to equation (5), where $F$ is set to 200.

The Pareto fronts obtained for the four test functions are shown in Figures 3, 4, 5 and 6. It can be seen from the figures that the EDWA method has successfully obtained the Pareto front for all test problems, no matter whether their Pareto front is convex, concave or non-convex.

C. Observations

The following two important observations can be made from the optimization processes using the EDWA

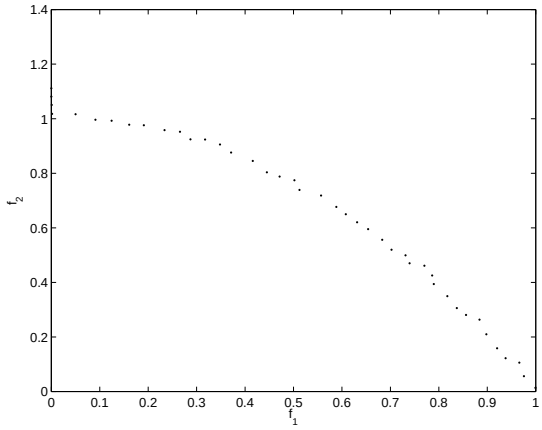



Figure 4: Achieved Pareto front for test function $F_2$.

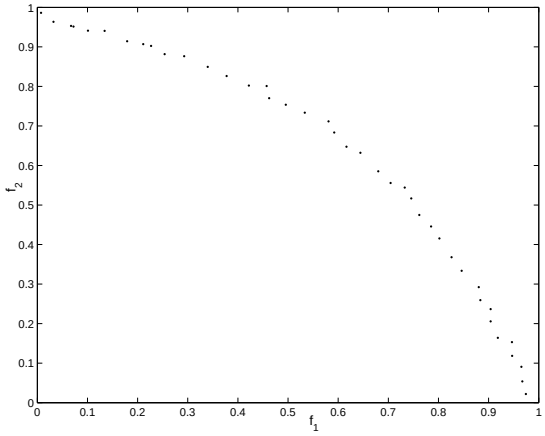



Figure 5: Achieved Pareto front for test function $F_3$.

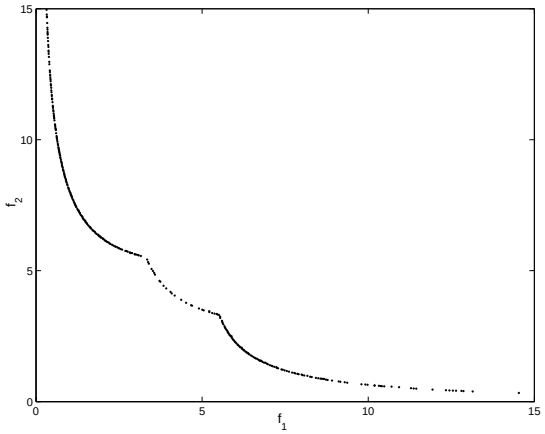
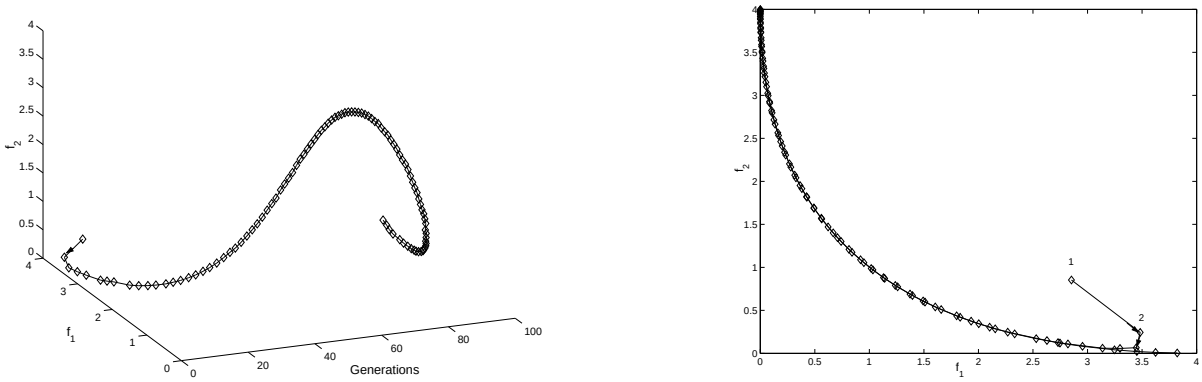



Figure 6: Achieved Pareto front for test function $F_4$.

method.

- If the Pareto front is convex, the population will first converge to any point on the Pareto front. To which point the population converges depends on the problem difficulty (convergence speed) and on the speed of weight change. Once the population has reached the Pareto front, it will keep moving on it as the weights change. The speed of movement can be controlled by the frequency of the weights change. Fig.7 shows a typical trace of the parent population during optimization of the test function $F_1$ in the first 100 generations represented by the mean of the population. Using different strategy parameters, only minor changes will occur in the starting phase of the trace.

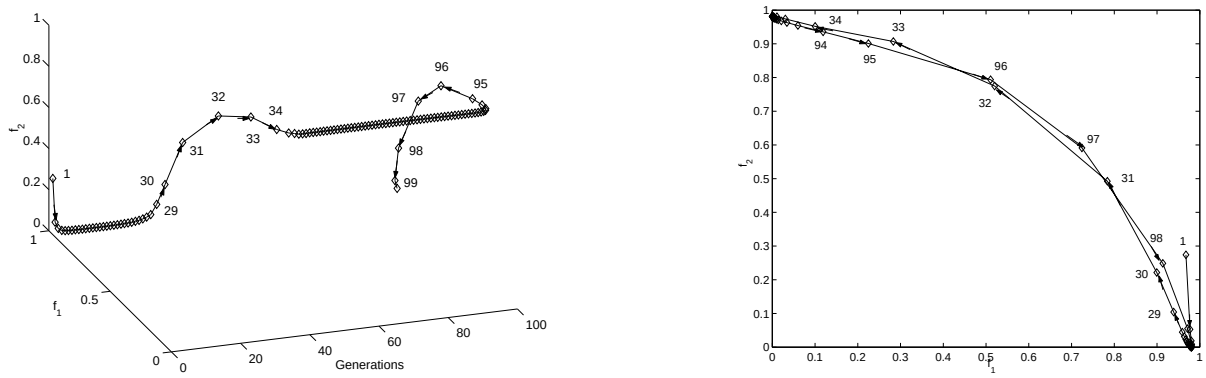



(a) (b)

Figure 7: Trace of the population for a convex problem ($F_1$). (a) Moves of the population with regard to time (generation); (b) Moves of the population in the fitness space. The arrows denote the direction of the movement and the numbers the generation index.

- If the Pareto front is concave, the population will converge to one of the two ends of the concave Pareto front. Thereafter, the population will remain on this Pareto optimal point until the weights are changed to a threshold value, which is determined by the characteristics of the Pareto front. Then, the population will move along or close to the Pareto front very quickly to the other end of the Pareto front. To illustrate this, the first 100 moves of the parent population during optimization of test function $F_3$ are presented in Fig. 8. From the figure, it can be seen that the population converges to the Pareto point ($f_1$, $f_2$)=(0.98,0) in a few generation. As the weights change, the population remains on the same point until generation 29. Then, the population moves along the Pareto front quickly to the other end of the Pareto front, where ($f_1$, $f_2$)=(0,0.98). Again, the population remains on this point until generation 91.

## IV. Theoretical Analysis




(a) (b)

Figure 8: Trace of the population for a concave problem ($F_3$). (a) Moves of the population with time (generation); (b) Moves of the population in the fitness space. The arrows denote the direction of the movement and the numbers the generation index.

Through a variety of test functions, it has been shown that the EDWA method works very well for approximation of both convex and concave Pareto fronts. However, an analytical proof of the method may be difficult, if not impossible. Nevertheless, it is still of sufficient significance to gain a deeper insight into the working mechanisms of the evolutionary weighted aggregation method. In the following, we first show schematically the reasons why the conventional weighted method has several weaknesses and why the EDWA method is able to overcome these shortcomings. Then, we show the global convexity empirically for all the test functions. Thereafter, we present the results from some test functions whose Pareto-optimal front is a natural local attractor in the fitness space. Finally, some remarks are made concerning the implementation of the EDWA method.

A. Stable Pareto-optimal Solutions

To explain the reasons why the EDWA method is able to overcome the weaknesses of the conventional weighted aggregation method, we first discuss schematically when a Pareto-optimal solution is stable using the conventional weighted aggregation method. In this paper, a Pareto optimal solution is defined as stable if this solution is achievable using the conventional weighted aggregation method for a given weight combination. We use the word "stable" instead of "achievable" because we want to explain this property of a solution using the stability of a ball on a surface under the influence of the gravity as an analogue. Therefore, a solution on the Pareto-optimal front is treated as a ball on the surface, that is, the Pareto front. The selection pressure takes up the role of the gravity, and the change of the weights is equivalent to the rotation of the surface. For the sake of clarity, we discuss bi-objective problems. We first consider a convex problem. In Fig. 9(a), point $A$ is a stable solution on the convex Pareto front

corresponding to a weight combination, say $(w_1,w_2) = (w_1^A,w_2^A)$. For this weight combination, solution $B$ is not stable. Therefore, it cannot be obtained with the weighted aggregation method using this weight combination. However, if the Pareto front rotates $\theta_B$ degrees, see Fig. 9(b), which now corresponds to a weight combination of $(w_1,w_2) = (w_1^B,w_2^B)$, point $B$ becomes stable and point $A$ unstable. To gradually decrease the weight for $f_1$ and increase the weight for $f_2$ is the same as to rotate the Pareto front counter-clockwise. Therefore, if the Pareto front is not rotated, which corresponds to a fixed weight combination in the conventional weighted aggregation method, only one solution can be obtained in one run of optimization. On the contrary, if the Pareto front is rotated gradually during optimization, which corresponds to the DWA method, the ball will roll through every point on the Pareto front. This means that all solutions on a convex Pareto front can be obtained in one run of optimization, although the found solutions need to be stored in an archive [5].

If the Pareto front is concave, the situation is much different. It is seen that given the weight combination $(w_1,w_2) = (w_1^A,w_2^A)$, point $A$ is stable and point $B$ is unstable, refer to Fig. 10(a). When the Pareto front rotates for $\theta_B$ degrees, see Fig. 10(b), neither point $A$ nor point $B$ is stable. Instead, point $C$ becomes stable. Therefore, point $C$ instead of point $B$ is obtained if the weights are changed from $(w_1,w_2) = (w_1^A,w_2^A)$ to $(w_1,w_2) = (w_1^B,w_2^B)$. It is straightforward to see that if the Pareto front is concave, only one of its two ends (point $A$ and $C$ in Fig. 10) can be a stable solution, no matter how the weights are changed. This is the reason why the conventional weighted aggregation method can obtain only one of the two stable solutions for any given weight combinations. Nevertheless, if the Pareto front is rotated, the ball is still able to move along the Pareto front from one stable point to the other, which means that the DWA method is able to get a number of the Pareto-optimal solutions even if the Pareto front is concave.

The phenomena observed in Section III(C) can be easily explained according to the above discussion. That is, once the population reaches any point on a convex Pareto front, it will keep moving on the Pareto front when the weights change gradually. In contrast, when a concave Pareto front rotates, the population will remain on the stable solution until this solution becomes unstable. Then the population will move along or close to the Pareto front from the unstable solution to the stable one.

It should be emphasized that our analysis does not conflict with the conclusions drawn in [7]. The reason is that although the points in the concave region of a Pareto front are not stable solutions and thus cannot be obtained by the conventional weighted aggregation method, they are reachable.

One remaining question is: when does a stable solution on a concave Pareto front become unstable

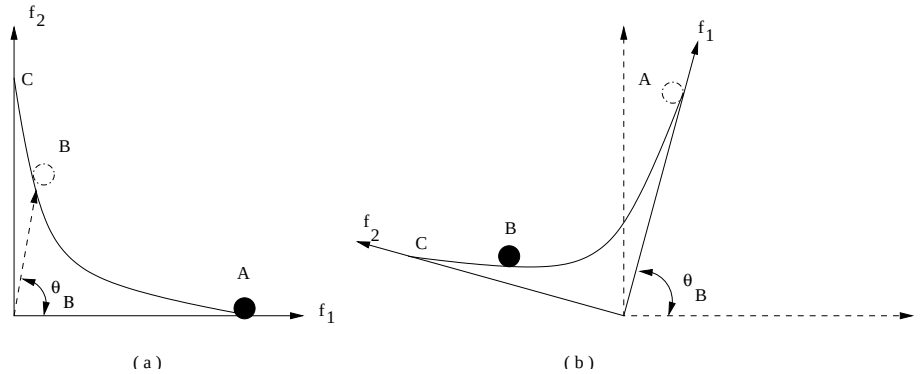



Figure 9: The case of a convex Pareto front. (a) The stable point is $A$; (b) After the Pareto front rotates $\theta_B$ degrees, the stable point becomes $B$.

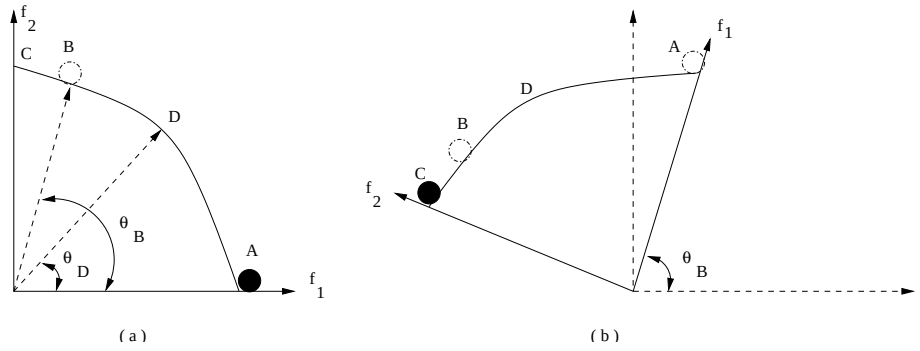



Figure 10: The case of a concave Pareto front. (a) The stable point is $A$; (b) After the Pareto front rotates $\theta_B$ degrees, the stable point is $C$, instead of point $B$. Point $D$ is the dividing point, which means that point $A$ becomes unstable until the Pareto front rotates $\theta_D$ degrees.

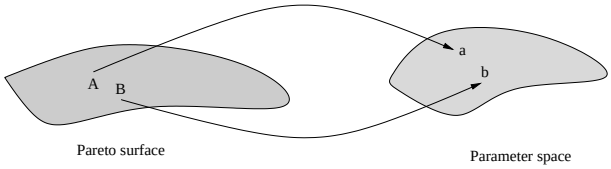



Figure 11: Mapping from the fitness space to the parameter space.

and when does the population begin to move? Obviously, there is a dividing point (denoted as point $D$ in Fig. 10) on any concave Pareto fronts. The stable solution in Fig. 10(a) becomes unstable only when the rotation angle is larger than $\theta_D$. This agrees well with the movement of the population shown in Fig. 8(a).

B. Global Convexity

To make the mechanism explained in Section IV(A) practical, one very important condition must be satisfied. That is, if an individual intends to move from one Pareto point ($A$) to a neighboring point ($B$) in a few steps, the solutions in the parameter space ($a$ and $b$) should also be in the neighborhood, as illustrated in Fig. 11. Meanwhile, since the Pareto-optimal solutions are rather concentrated in the fitness space, i.e., the Pareto front, the solutions in the parameter space should also locate in a very small region of the parameter space. These are actually the two main aspects of the global convexity in continuous multiobjective optimization problems.

It is very interesting and surprising to verify that for all the test functions studied in the paper, the solutions in the parameter space are really concentrated in very small regions that can be described by a piecewise linear function. We call it the *definition function* of the Pareto front, which is a function in the parameter space that defines the Pareto front in the fitness space.

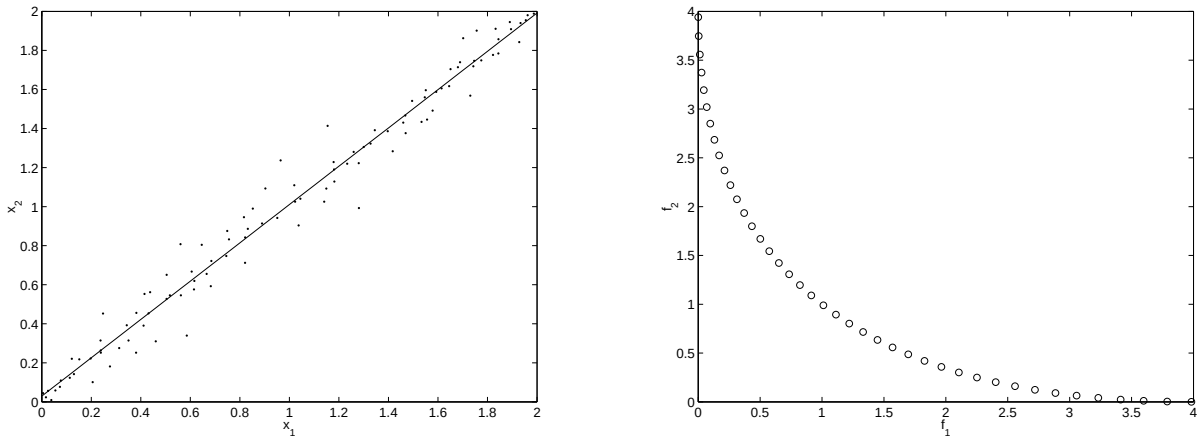
The solutions of function $F_1$ obtained in the simulation in Section III(B) are shown in Fig. 12 (a). These solutions can be described by a linear definition function. Using the least square method, the following linear model can be obtained, which is also illustrated in Fig. 12 (a):

$$x_2 = 0.0296 + 0.981x_1. \tag{19}$$

It means that the Pareto front can be directly obtained using the definition function in equation (19). This is verified in Fig. 12 (b).

Similarly, the following definition function can be obtained for $F_2$:

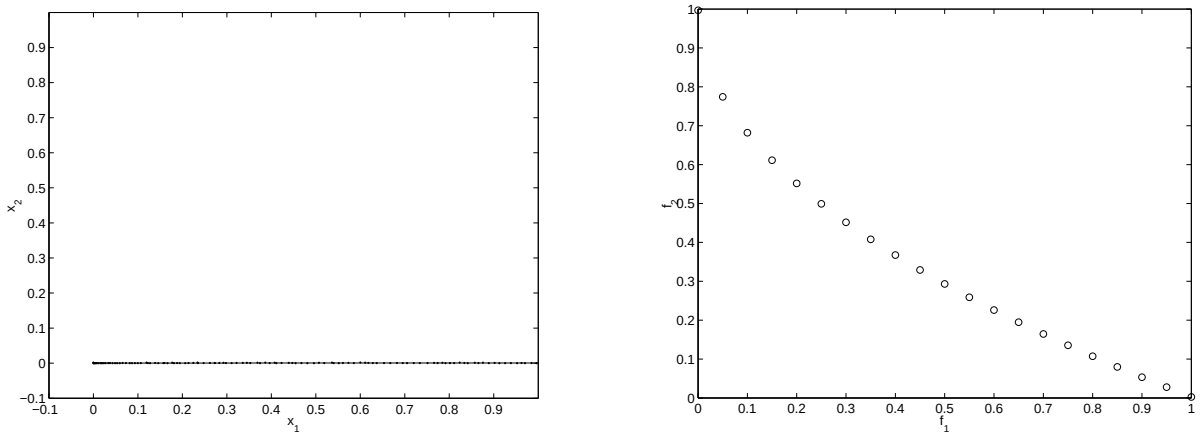
$$x_2 = 0. \tag{20}$$




(a) (b)

Figure 12: Test function $F_1$. (a) Distribution of the solutions in the parameter space and the estimated definition function; (b) Pareto front reconstructed from the estimated definition function.

The distributions of the solutions, the estimated definition function and the Pareto front reconstructed from the definition function are shown in Fig. 13.



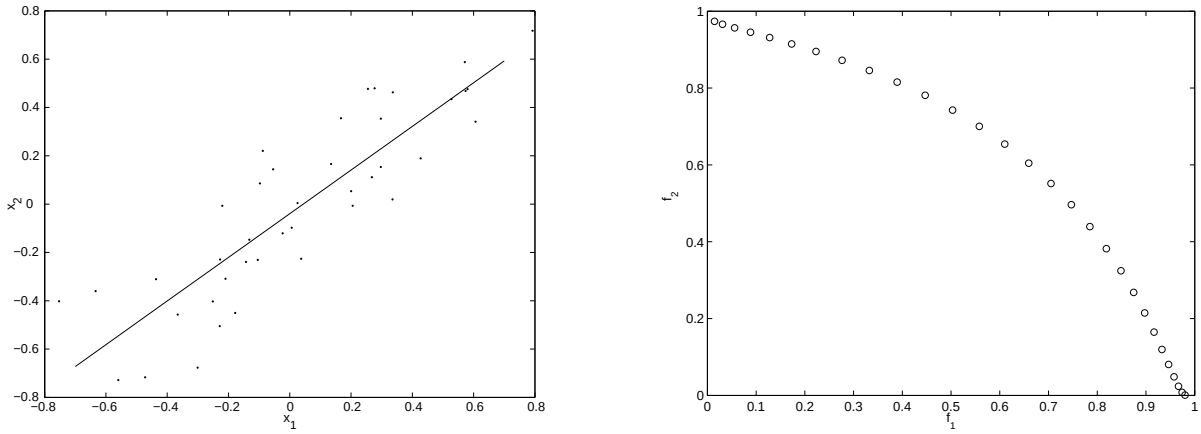

(a) (b)

Figure 13: Test function $F_2$. (a) Distribution of the solutions in the parameter space and the estimated definition function; (b) Pareto front reconstructed from the estimated definition function.

Now we try to construct an approximate model for the definition function of the Pareto front for test functions $F_3$ and $F_4$. Using the solutions obtained in optimization, the following linear model has been obtained for $F_3$:

$$x_2 = -0.0397 + 0.9035x_1, -0.70 \leq x_1 \leq 0.70. \tag{21}$$

The distribution of the solutions in the parameter space and the estimated definition function are shown in Fig. 14(a), whereas the Pareto front reconstructed from the estimated definition function is




(a) (b)

Figure 14: Test function $F_3$. (a) Distribution of the solutions in the parameter space and its estimated model; (b) Pareto front reconstructed from the estimated definition function.

illustrated in Fig. 14(b). It can be seen that the Pareto front reconstructed from the estimated definition function is qualitively comparable to the one obtained in the optimization (Fig. 5).

Similarly, approximate models for the definition function of $F_4$ can also be constructed. From the solutions, the following piece-wise linear model has been obtained, which consists of four sections:

$$\text{section 1:} \quad x_2 = -0.002 + 0.999x_1; \quad 0.45 \leq x_1 \leq 2.0 \tag{22}$$

$$\text{section 2:} \quad x_2 = 1.684 + 0.80x_1; \quad 1.35 \leq x_1 \leq -0.55 \tag{23}$$

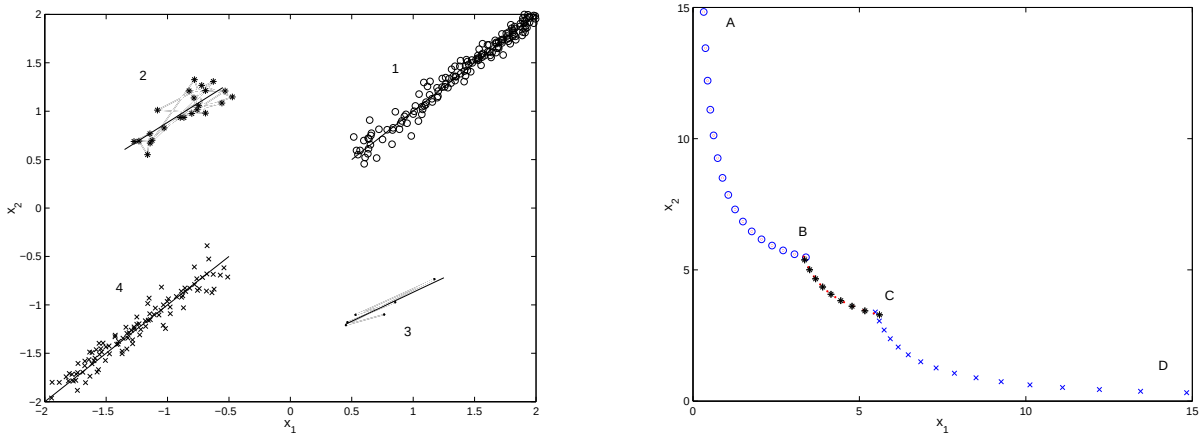
$$\text{section 3:} \quad x_2 = -1.47 + 0.60x_1; \quad 0.45 \leq x_1 \leq 1.35 \tag{24}$$

$$\text{section 4:} \quad x_2 = -0.002 + 0.999x_1; \quad -2.0 \leq x_1 \leq -0.55. \tag{25}$$

The estimated definition model and the reconstructed Pareto front are given in Fig. 15 (a) and (b), respectively. It is noticed that section $BC$ on the Pareto optimal front can be reconstructed either from section 2 or from section 3 of the definition function.

Until now, we have shown that the Pareto-optimal solutions of all the test functions are gathered in a very small fraction of the parameter space. Moreover, they can be described by a piecewise linear function. In the following, we will show empirically that all the test functions exhibit the second aspect of global convexity, i.e., if two solutions are in the neighborhood in the fitness space, then they are also in the neighborhood in the parameter space.

The neighborhoodness can be measured by calculating the Manhattan distance between the two points. Given two vectors $\boldsymbol{u} = \{u_1, .., u_n\}$ and $\boldsymbol{v} = \{v_1, .., v_n\}$, the Manhattan distance between them is




(a) (b)

Figure 15: Test function $F_4$. (a) Distribution of the solutions in the parameter space and its estimated model; (b) Pareto front reconstructed from the estimated definition function.

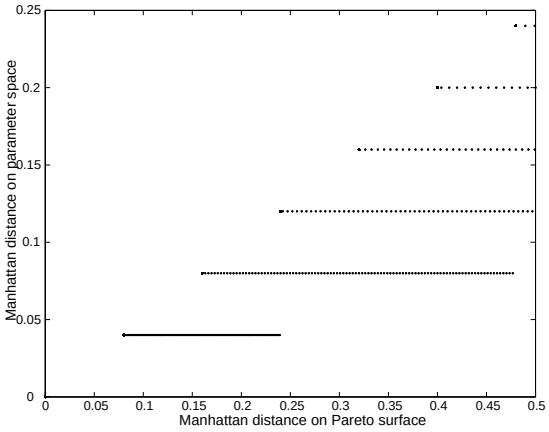



Figure 16: Manhattan distance for test function $F_1$.

defined by:

$$d(\boldsymbol{u}, \boldsymbol{v}) = \sum_{l=1}^{n} |u_l - v_l|. \tag{26}$$

With the help of the definiton function of the Pareto front, it is straightforward to calculate the Manhattan distance between two Pareto-optimal solutions in the parameter space, as well as in the fitness space. The results from the four test functions are presented in Figures 16, 17, 18 and 19, respectively. Since we are investigating the neighborhoodness of multiobjective optimization, the maximal distance between two Pareto points shown in the figures is 0.5 for $F_1$ and $F_4$, and 0.1 for $F_2$ and $F_3$, which are scaled according to the range of the Pareto fronts.

From the figures, it is confirmed that for any two neighboring points on the Pareto front, their

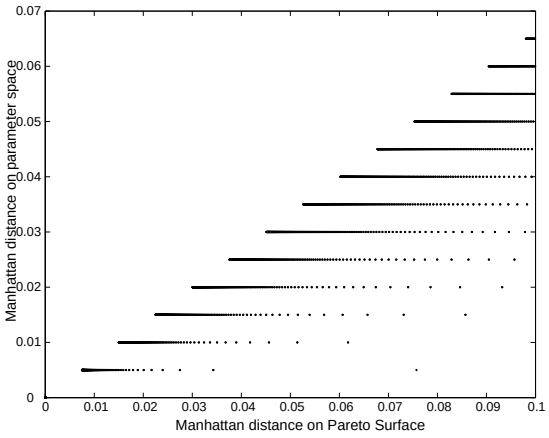



Figure 17: Manhattan distance for test function $F_2$.

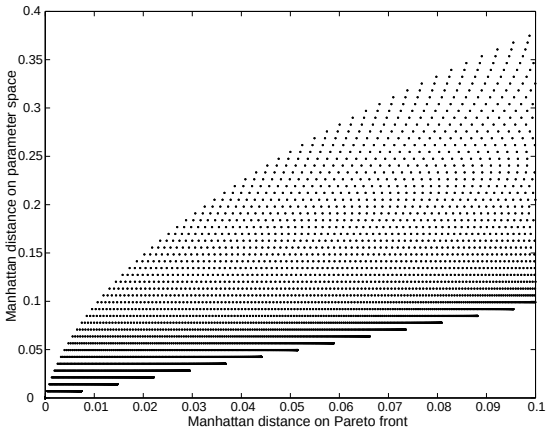
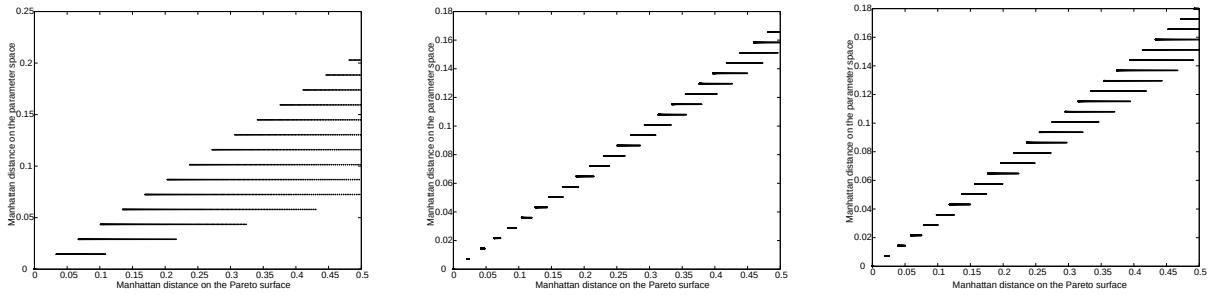



Figure 18: Manhattan distance for test function $F_3$.




(a) (b) (c)

Figure 19: Manhattan distance for test function $F_4$. (a) Section 1 and 4; (b) Section 2; (c) Section 3.

corresponding solutions in the parameter space are also in the neighborhood. Generally speaking, the smaller the distance on the Pareto front, the smaller the distance in parameter space, although this relationship is not monotonous, i.e., given the same distance on the Pareto front, the distance on the parameter space may vary a little.[1]

The global convexity of multiobjective optimization and the strongly causal search of evolution strategies are essential for the success of the EDWA. Only with these conditions being satisfied, the population is able to move along ( or close to) the Pareto front when the weights change during the optimization, once the population reaches any point on the Pareto front.

C. Local Attractor

The property of the global convexity is an important prerequisite for the success of the EDWA method. Besides, another phenomenon, namely, the Pareto front is sometimes a natural attractor in the fitness space, is also observed. This phenomenon is helpful in achieving the Pareto front using the EDWA, although it is not a necessary condition for the success of the EDWA.

The phenomenon that the Pareto front is a local attractor in the fitness space can be shown by investigating the mapping from the parameter space to the fitness space in multiobjective problems. In evolution strategies, mutation is the main search mechanism, which is usually implemented by adding a random number from a normal distribution to the objective parameters. It is therefore very important to see how a normal distribution is changed by the mapping from the parameter space onto the fitness space. Without the loss of generality, we consider two-dimensional two-objective optimization problems. The distribution in the fitness space can be derived from the distribution in the parameter space as follows,

[1]It is interesting to note that this corresponds to "small changes are allowed to lead to small or no changes" and vice versa, which is known as neutrality in the genotype-phenotype mapping in genetic programming [29].

provided that the fitness function is a one-to-one function:

$$\int_{f_1}^{f_1+\Delta f_1} \int_{f_2}^{f_2+\Delta f_2} g(f_1', f_2') df_2' df_1' = \int_{x_1}^{x_1+\Delta x_1} \int_{x_2}^{x_2+\Delta x_2} f(x_1', x_2') dx_2' dx_1', \tag{27}$$

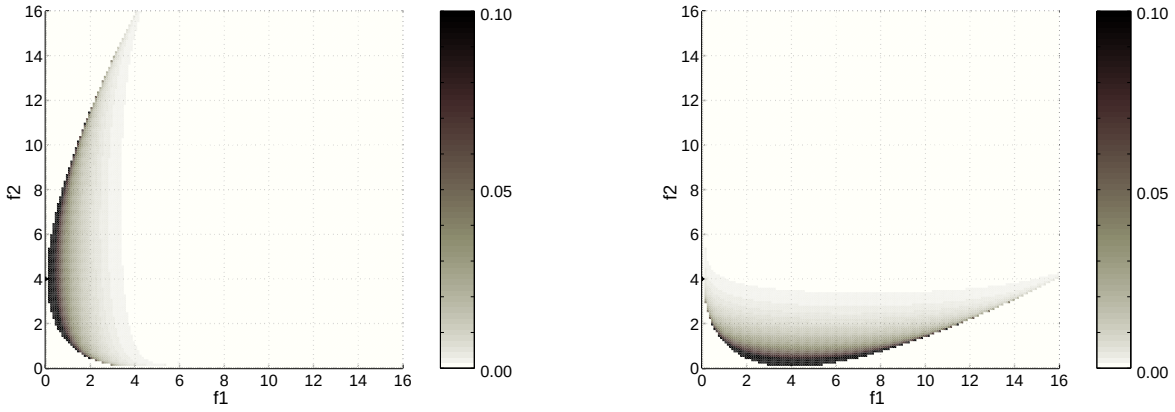
where $g(f_1, f_2)$ is the probability density distribution (pdf) in the fitness space, and $f(x_1, x_2)$ is a normal random distribution in the parameter space:

$$f(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{(x_1-\mu_1)^2}{2\sigma_1^2}} e^{-\frac{(x_2-\mu_2)^2}{2\sigma_2^2}}, \tag{28}$$

where, $\sigma_i$ and $\mu_i$ $(i = 1, 2)$ are the standard deviation and the mean, respectively. Equation 27 can be rewritten by:

$$g(f_1, f_2) = \frac{1}{|J|} f(x_1, x_2), \tag{29}$$

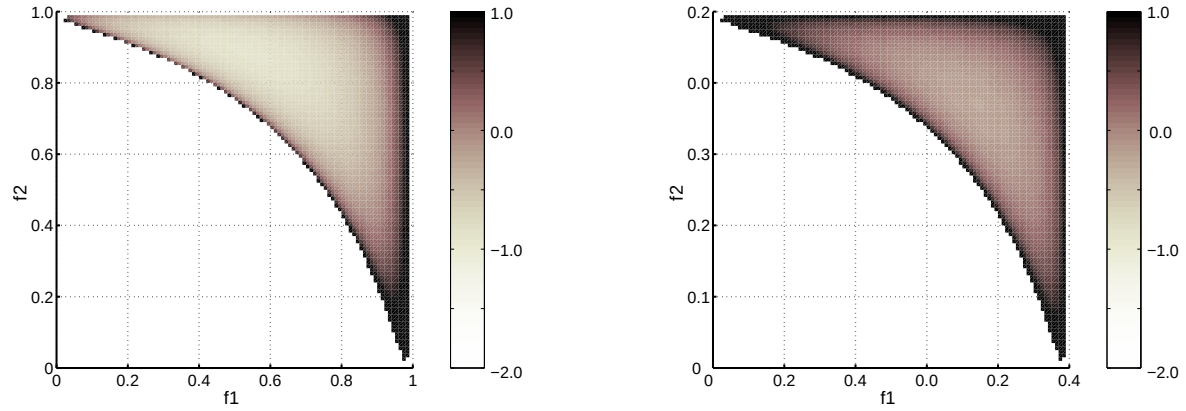
where $J$ is the Jacobian matrix. Refer to [30] for more discussions on this property of evolutionary multiobjective optimization.




(a) (b)

Figure 20: The distributions $g(f_1, f_2)$ in the fitness space for test function $F_1$ generated from a normal distribution in the parameter space on (a) $(x_1, x_2) = (0, 0)$, (b) $(x_1, x_2) = (2, 2)$. The standard deviation of distribution $f(x_1, x_2)$ is $(\sigma_1, \sigma_2) = (1, 1)$.

Fig. 20 shows the probability distributions in the fitness space for $F_1$ corresponding to a normal distribution in the parameter space, where the darker the color, the higher the probability. It can be seen that given a normal distribution in the parameter space, its corresponding distribution in the fitness space is biased toward the Pareto front, which means that the region close to the Pareto front will be explored more extensively than that far from the Pareto front. Therefore, the Pareto front is a local attractor in the fitness space in a probability sense. We call it a local attractor because this phenomenon can be observed only if the center of the distribution in the parameter space is relatively close to or on the curve of the definition function.




(a) (b)

Figure 21: The distributions $g(f_1, f_2)$ in the fitness space for test function $F_3$ generated from a normal distribution in the parameter space on (a) $(x_1,\ x_2)$=$(-\frac{1}{\sqrt{2}},-\frac{1}{\sqrt{2}})$, (b) $(x_1,\ x_2)$=(0,0). The standard deviation of distribution $f(x_1, x_2)$ is $(\sigma_1, \sigma_2) = (1, 1)$.

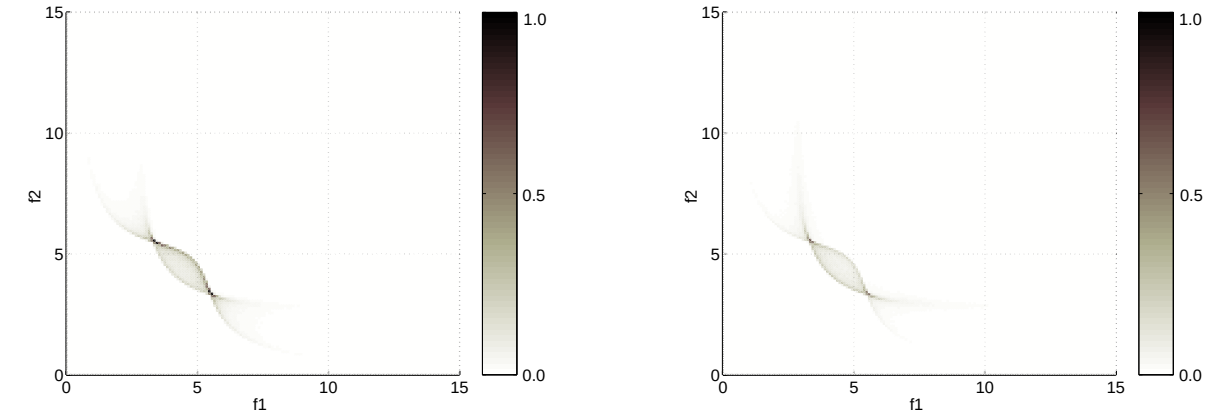
Similar phenomena have been observed for test functions $F_3$ and $F_4$, which are shown in Fig. 21 and Fig. 22.

In contrast, this phenomenon cannot be observed for test function $F_2$. Fig. 23 shows that the distributions in the fitness space almost remain to be normal ones. Nevertheless, the EDWA method has obtained the Pareto front successfully, which again implies that the local attractor property is very helpful but not a prerequisite for the EDWA method.

D. Remarks

Stability of a Pareto solution, the global convexity of multiobjective optimization, as well as the strong causality of evolution strategies are believed to be the main factors behind the success of the EDWA method. Nevertheless, the following two issues should also be taken into account in the implementation of the method.

- Notice that the population moves along or close to the Pareto front when moving from one stable point on the front to another. However, if the weight change is very large, for example, if the weights are switched between 0 and 1 and the Pareto front is convex, the population may leave the Pareto front, see Fig.24 (a). This is due to the fact that with such a large change of the weights, the population has to move to another stable solution that is not in the neighborhood. Therefore, it is important to change the weights gradually if the Pareto front is convex. In contrast, if the Pareto front is concave, the population is usually able to move along or close to the Pareto front, even when the weights change is large, refer to Fig. 24(b). It can be seen that if an individual is




(a) (b)

Figure 22: The distributions $g(f_1, f_2)$ in the fitness space for test function $F_3$ generated from a normal distribution in the parameter space on (a) ($x_1$, $x_2$)=(0,0), (b) ($x_1$, $x_2$)=(1,−0.884). The standard deviation of distribution $f(x_1, x_2)$ is $(\sigma_1, \sigma_2) = (1, 1)$.

currently at point $A$ and should move towards $B$, then the direction of the selection pressure is pointing from $A$ to $B$. However, since the region left to the Pareto front is infeasible, the population has to move along or close to the Pareto front from point $A$ to $B$.

- As discussed in Section IV(B), the global convexity is a basis for the success of the EDWA method, where it is assumed that distance in the parameter space is small. However, the step sizes, $\sigma_i$ in equation (7), of the evolution strategy should be prevented from converging to zero, especially when the Pareto front is concave. Therefore, it is necessary to check the step-size of the evolution strategy. If it is smaller than a given threshold, it should be reset to a predefined lower bound.

## V. APPLICATIONS

The EDWA method has been successful in approximating Pareto fronts for a number of test problems. In this section, we will apply the EDWA method to two non-trivial application examples. The first example is a three-bar truss optimization problem taken from [31] and the second example is a real-world aerodynamic design optimization problem.

### A. Three-bar truss optimization under static loading

The structure of the three-bar truss is illustrated in Fig.25. The target of the optimization is:

$$\min\{\delta(\boldsymbol{A}), V(\boldsymbol{A})\}, \tag{30}$$

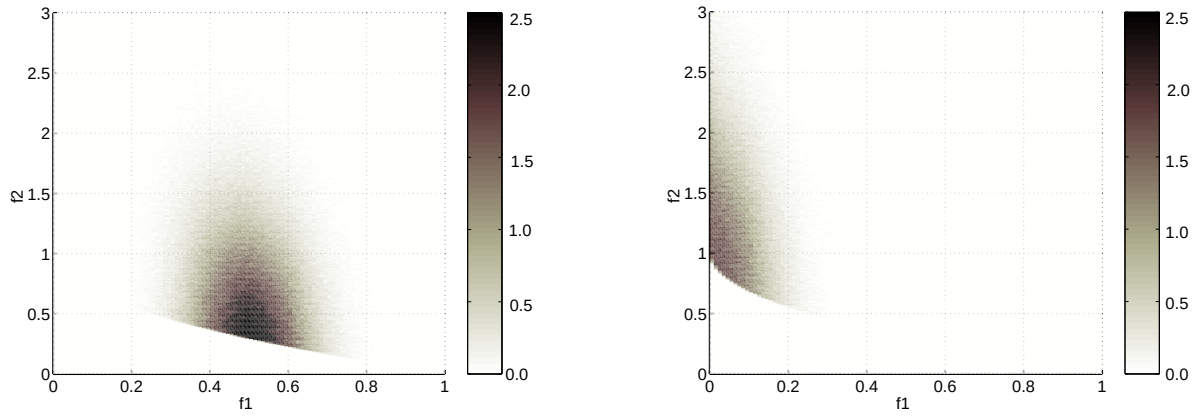



Figure 23: The distributions $g(f_1, f_2)$ in the fitness space for test function $F_2$ generated from a normal distribution in the parameter space on (a) ($x_1$, $x_2$)=(0.5,0), (b) ($x_1$, $x_2$)=(0,0). The standard deviation of distribution $f(x_1, x_2)$ is $(\sigma_1, \sigma_2) = (0.01, 0.01)$.

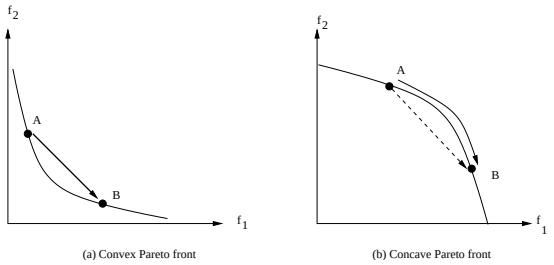



Figure 24: Direction of selection pressure. (a) Convex Pareto front; (b) Concave Pareto front.

where, $\boldsymbol{A} = \{A_1, A_2, A_3\}$ are the area of the elements, and $\delta$ is a combination of the nodal displacements:

$$\delta = 0.25\delta_1 + 0.75\delta_2. \tag{31}$$

The target of the optimization is to minimize the volume of the truss and the combined nodal displacement, subject to the following constraints:

$$\begin{aligned} A_{min} \leq A_i \leq A_{max}, \qquad & i = 1, 2, 3, \\ \sigma_{min} \leq \sigma_i \leq \sigma_{max}, \qquad & i = 1, 2, 3, \end{aligned} \tag{32}$$

where $A_{min}$ and $A_{max}$ are the constraints for the area, and $\sigma_{min}$ and $\sigma_{max}$ are the allowable stress for each element.

Using the finite element analysis method, we can reduce the truss optimization to the following two-

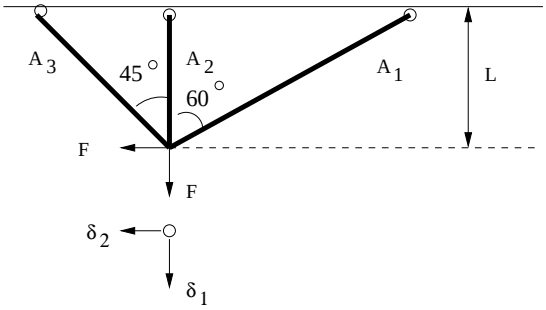



Figure 25: Three-bar truss structure.

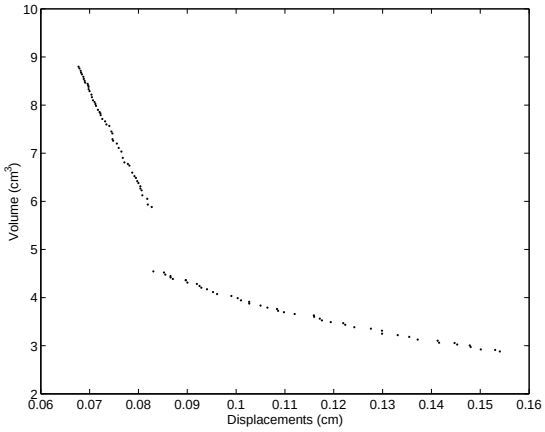



Figure 26: Achieved Pareto front for the truss problem.

objective problem:

$$f_1 = \frac{FL}{4E}\frac{(6\sqrt{2}-4\sqrt{6})A_1 + 24\sqrt{2}A_2 + 32A_3}{(\sqrt{3}+2)A_1A_3 + 3\sqrt{2}A_1A_3 + 4A_2A_3} \quad (33)$$

$$f_2 = 2A_1 + A_2 + \sqrt{3}A_3, \quad (34)$$

subject to the constraints in Equation (32). In this example, the parameters are set as follows: $F = 20kN$, $L = 100cm$, $E = 200GPa$, $A_{min} = 0.1cm$, $A_{max} = 2cm$, $\sigma_{min} = -200MPa$ and $\sigma_{max} = 200MPa$.

As shown in [31], the true Pareto front is divided into two disconnected sections. One section is concave and the other section is convex. It has also been shown in the same reference that the conventional weighted aggregation method fails to obtain the Pareto front. Using the EDWA method, the Pareto front has successfully been obtained, as shown in Fig. 26.

To take a closer look at the properties of this problem, we can also estimate the definition function

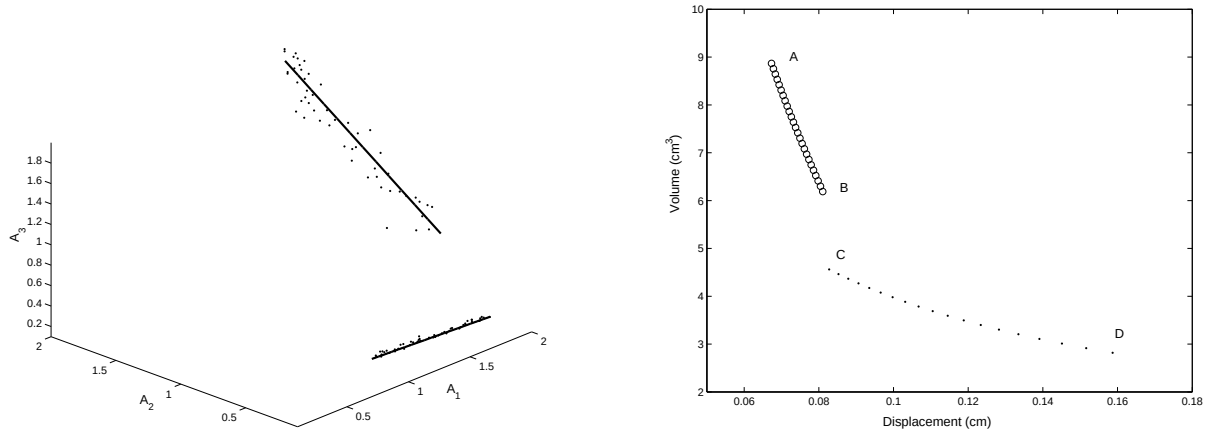
for the truss optimization problem, which consists of two separated sections:

$$\text{section 1:} \qquad A_1 = 2.0$$

$$A_3 = -0.019 + 0.946A_2, \quad 0.8 \leq A_2 \leq 2.0 \tag{35}$$

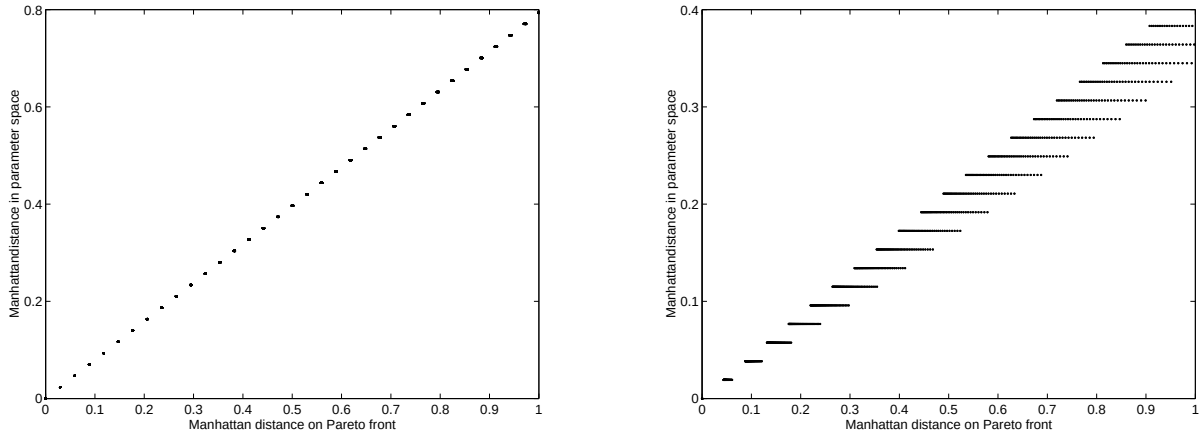
$$\text{section 2:} \quad A_2 = 0.549 - 0.0645A_1, \quad 1.1 \leq A_1 \leq 2.0$$

$$A_3 = 0.1. \tag{36}$$




(a) (b)

Figure 27: Truss optimization. (a) The distribution of the solutions in the parameter space and the estimated definition function. (b) The reconstructed Pareto front using the definition function.

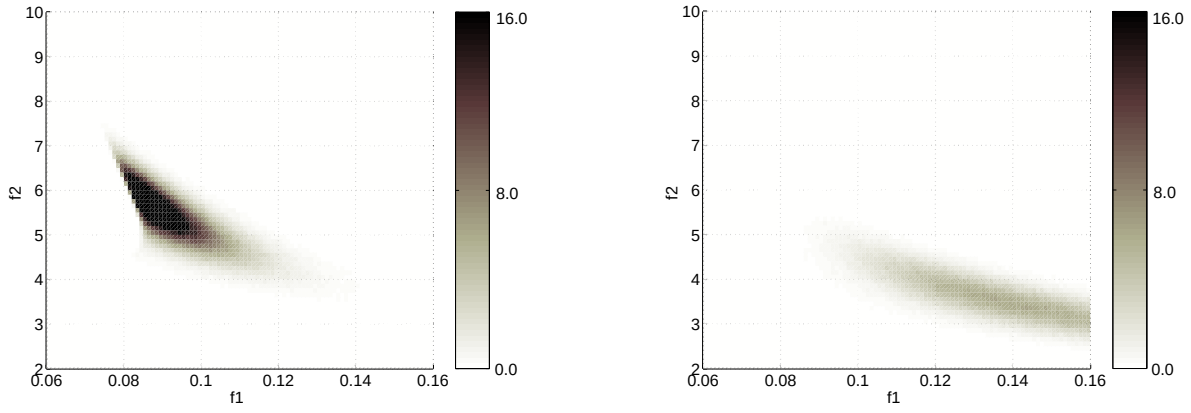



(a) (b)

Figure 28: Manhattan distance for the truss optimization problem. (a) Section 1; (b) Section 2.

The estimated definition function and the reconstructed Pareto front are shown in Fig. 27. Section $AB$ of the Pareto front is concave, which cannot be obtained using the conventional weighted aggregation method. With the estimated definition function, the Manhattan distance on the Pareto front versus the

distance in the parameter space can be calculated, which is shown in Fig 28. It can be seen that the truss optimization problem is also globally convex.




(a) (b)

Figure 29: The distributions $g(f_1, f_2)$ in the fitness space for the truss problem generated from a normal distribution in the parameter space on (a) ($A_1$, $A_2$, $A_3$) = (2.0, 0.8, 0.7378), (b) ($A_1$, $A_2$, $A_3$)=(1.1, 0.478, 0.1). The standard deviation of distribution $f(A_1, A_2, A_3)$ is ($\sigma_1$, $\sigma_2$, $\sigma_3$)= (0.1, 0.1, 0.1).

The parameter settings of the evolution strategy is the same as those in the simulations in Section III.

It is also interesting to investigate the distribution in the fitness generated from a normal distribution in the parameter space. From Fig. 29, it is seen that the Pareto front is not a local attractor. However, as we mentioned previously, this does not prevent the algorithm from getting the entire Pareto front, although it does imply that this type of test functions may be harder for the EDWA method.

B. Aerodynamic blade optimization

In this example, the EDWA method is employed to optimize a gas turbine blade [32, 33]. To evaluate the performance of a blade, computational fluid dynamics (CFD) simulations have to be carried out. Navier-Stokes equations with the $(k-\epsilon)$ turbulence model are used for two-dimensional CFD simulations [34]. The two most important performance indices of a blade are the pressure loss and the deviation of the outflow angle from a prescribed value. Therefore, the target of this multiobjective optimization problem is to minimize the pressure loss and the deviation of the outflow angle.

In order to describe the two-dimensional cross section of a blade, a spline encoding based on the non-uniform rational B-splines is used [32, 35]. The spline is constructed from a set of $N$ four-dimensional control points $(x, y, z, w)$ that define the control polygon, where $x, y, z$ are the three-dimensional coordinates and $w$ is a weight for the point. A two-dimensional cross section is obtained by setting the

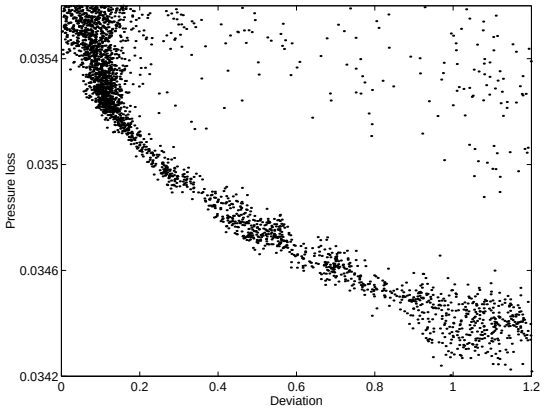



Figure 30: Solutions of multiobjective blade optimization.

$z$-coordinate to zero. To minimize the dimension of the search space, the weights of the control points are fixed so that the representation used is a regular B-spline in this example. Therefore, every control point is described by only two parameters, i.e., the $x$ and $y$ coordinate of the point. In this work, 26 control points are used in the B-spline. Since two-dimensional optimization is considered, there are 52 input parameters.

An evolution strategy with covariance matrix adaptation [36] has been combined with the dynamic weighted aggregation method. The parent population size is 2 and the offspring population size is set to 16. All solutions whose deviation is less than 1.2 degree and pressure loss is less than 0.0356 are shown in Fig.30. It can be seen that quite a complete set of Pareto solutions have successfully been obtained by the EDWA in one run of optimization.

## VI. CONCLUSIONS

The evolutionary dynamic weighted aggregation method for multiobjective optimization has been further studied in the paper. By changing the weights periodically, the method is able to approximate a Pareto front in one single run, no matter whether the Pareto front is convex or concave.

To establish the theoretical basis for the success of the EDWA method, the properties of global convexity and local attractor in multiobjective optimization have been studied. It is empirically shown that for all the test functions, the Pareto-optimal solutions are concentrated in a very small and regular region of the parameter space. The region can be defined by a piecewise linear function when the parameter space is of low dimensionality. It is also empirically verified that solutions in the neighborhood in the fitness space are also in the neighborhood in the parameter space. Besides, it is found that the

Pareto front is a natural local attractor in many multiobjective optimization problems, which is helpful in achieving the Pareto front.

The effectiveness of the EDWA method has been supported by two non-trivial application examples. In the 3-bar truss optimization, the Pareto front consists of two disconnected sections, one is convex, the other is concave. Despite of this difficulty, the EDWA method is able to achieve a quite full Pareto front. Through the blade optimization example, it has been shown that the EDWA method works also efficiently for high-dimensional, real-world problems.

ACKNOWLEDGMENT

The authors would like to thank Rajkumar Vaidyanathan, Wei Shyy and Carlos Artemio Coello Coello for sharing their knowledge on the finite element analysis of truss structures. The authors are also grateful for the support from Edgar Körner.

# References

[1] A.M. Geoffrion. Proper efficiency and the theory of vector optimization. *Journal of Mathematical Analysis and Application*, 41:491–502, 1968.

[2] P. Hajela and C. Y. Lin. Genetic search strategies in multicriteria optimal design. *Structural Optimization*, 4:99–107, 1992.

[3] 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.

[4] T. Murata, H. Ishibuchi, and M. Gen. Specification of genetic search directions in cellular multi-objective genetic algorithms. In *Proceedings of the First International Conference on Evolutionary Multi-Criteria Optimization*, volume 1993 of *Lecture Notes in Computer Science*, pages 82–95. Springer, 2001.

[5] Y. Jin, T. Okabe, and B. Sendhoff. Adapting weighted aggregation for multi-objective evolution strategies. In K. Deb, L. Thiele, and E. Zitzler, editors, *First International Conference on Evolutionary Multi-criterion Optimization*, volume 1993 of *Lecture Notes in Computer Science*, pages 96–110, Zurich, Switzerland, 2001. Springer.

[6] 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, San Francisco, CA, 2001.

[7] 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.

[8] A. Messac, E. Melachrinoudis, and C.P. Sukam. Aggregate objective functions and Pareto frontiers: Required relationships and practical implications. *Optimization and Engineering*, 1:171–188, 2000.

[9] A. Charnes and W. Cooper. Global programming using multiple objective optimization - Part I. *European Journal of Operations Research*, 1:39–54, 1977.

[10] J. Dauer and R. Krueger. An iterative approach to global programming. *Operations Research Quarterly*, 28:671–681, 1977.

[11] T.W. Athan and P.Y. Papalambros. A note on weighted criteria methods for compromise solutions in multi-objective optimization. *Engineering Optimization*, 27:155–176, 1996.

[12] M. Zeleny. Compromise programming. In J.L. Cochrane and M. Zeleny, editors, *Multiple Criteria Decision Making*, pages 262–301. University of South Carolina Press, SC, 1973.

[13] V.J. Bowman. On the relationship of the Tchebycheff norm and the efficient frontier of multiple criteria objectives. In H. Thiriez and S. Zionts, editors, *Multiple Criteria Decision Making*, volume 135 of *Lecture Notes in Economics and Mathematical Systems*, pages 76–85. Springer, 1976.

[14] I. Das and J. Dennis. Normal-boundary intersection: An alternate method for generating Pareto optimal points in multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3):631–657, 1998.

[15] T.C. Hu, V. Klee, and D. Larman. Optimization of globally convex functions. *SIAM Journal of Control and Optimization*, 27:1026–1047, 1989.

[16] K.D. Boese, A.B. Kahng, and S. Muddu. A new adaptive multi-start technique for combinatorial global optimizations. *Operations Research Letters*, 16:101–113, 1994.

[17] P.C. Borges and M.P. Hansen. A basis for future successes in multiobjective combinatorial optimization. Technical Report IMM-REP-1998-8, Department of mathematical Modeling, Technical University of Denmark, 1998.

[18] M. Ehrgott and K. Klamtoth. Connectedness of efficient solutions in multiple criteria combinatorial optimization. *European Journal of Operational Research*, 97:159–166, 1997.

[19] B. Sendhoff, M. Kreutz, and W. von Seelen. A condition for the genotype-phenotype mapping: Causality. In T. Bäck, editor, *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 73–80. Morgan Kauffman, 1997.

[20] I. Rechenberg. *Evolutionsstrategie'94*. Friedrich Frommann Holzboog, 1994.

[21] R. Lohman. Structure evolution and incomplete induction. *Biological Cybernetics*, 69(4):319–326, 1993.

[22] C.A.C. Coello. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems*, 1(3):269–308, 1999.

[23] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195, 2000.

[24] H.-P. Schwefel. *Evolution and Optimum Seeking*. Sixth-Generation Computer Technologies Series. John Wiley & Sons, Inc., 1994.

[25] J.D. Schaffer. Multiobjective optimization with vector evaluated genetic algorithm. In J.Grefenstette, editor, *Proceedings of the First International Conference on Genetic Algorithms*, pages 93–100, San Mateo, CA, 1985.

[26] K. Deb. Multiobjective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation*, 7(3):205–230, 1999.

[27] C.M. Fonseca and P.J. Fleming. Genetic algorithm for multiobjective optimization: Formulation, discussion and generalization. In S. Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 416–423, San Mateo, 1993. Morgan Kaufmann.

[28] 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.

[29] C. Igel. Causality of hierarchical variable length representations. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 324–329. IEEE Press, 1998.

[30] T. Okabe, Y. Jin, and B. Sendhoff. On the dynamics of evolutionary multiobjective optimization. In *Proceedings of Genetic and Evolutionary Computation*, pages 247–256, New York, 2002.

[31] J. Koski. Deffectiveness of weighting method in multicriterion optimization of structures. *Communications in Applied Numerical Methods*, 1:333–337, 1985.

[32] M. Olhofer, Y. Jin, and B. Sendhoff. Adaptive encoding for aerodynamic shape optimization using evolution strategies. In *Proceedings of IEEE Congress on Evolutionary Computation*, volume 2, pages 576–583, Seoul, Korea, 2001.

[33] M. Olhofer, T. Arima, Y. Jin, T. Sonoda, and B. Sendhoff. Optimization of transonic gas turbine with evolution strategies. *Honda R&D Technical Review*, 14(1):203–216, 2002.

[34] T. Arima, T. Sonoda, M. Shirotori, A. Tamura, and K. Kikuchi. A numerical investigation of transonic axial compressor rotor flow using a low-Reynolds number $\kappa$-$\varepsilon$ turbulance model. *Journal of Turbomachinery*, 121:44–58, 1999.

[35] Y. Jin, M. Olhofer, and B. Sendhoff. A framework for evolutionary optimization using approximate fitness functions. *IEEE Transactions on Evolutionary Computation*, 2002. in press.

[36] Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.