An Empirical Analysis of Fitness Assignment and Diversity-Preserving in Evolutionary Multi-Objective Optimization

Youyun Ao
School of Computer and Information, Anqing Normal University
PO box 246011, Anqing, Anhui, China
Tel: 86-150-5564-5226 E-mail: youyun.ao@gmail.com

Received: December 22, 2011 Accepted: January 9, 2012 Published: March 1, 2012
doi:10.5539/cis.v5n2p111 URL: http://dx.doi.org/10.5539/cis.v5n2p111

Abstract

Evolutionary algorithms have (EAs) been an alternative class of powerful search techniques. They have been widely applied to solve multi-objective optimization problems from scientific community and engineering fields. The aim of designing EAs for multi-objective optimization is to obtain a well-converged and well-distributed set involving multiple Pareto-optimal solutions in a single simulation run. Accordingly, improving the convergence speed and preserving the diversity of solutions are identically important during the search of EAs. In EAs, an effective fitness assignment approach is beneficial to improve the convergence speed and simultaneously guide the search of EAs towards optimal regions; an effective fitness sharing technique can improve the diversity of solutions in order to avoid the premature convergence. Additionally, the search capability of evolving operators themselves plays an important role in solving multi-objective optimization problems. This paper introduces two alternative fitness assignment approaches based on Pareto ranking to guide the search towards optimal regions, develops three alternative pruning techniques (i.e., specific fitness sharing techniques), and incorporates a dynamic mutation operator into EAs in order to enrich the diversity of solutions. Experimental results show that these approaches are effective. The purpose of this study is to gain a specific and important insight into well-established techniques and encourage their usage in further empirical studies.

Keywords: Evolutionary algorithm, Multi-objective optimization, Differential evolution, Fitness assignment, Diversity-preserving, Convergence

1. Introduction

Many real-world optimization problems often involve multiple conflicting objectives, namely multi-objective optimization problems (David A. *et al*., 2000). Without loss of generality, a single-objective optimization problem has a particular global optimum. For a multi-objective optimization problem, there exists a set of solutions. These solutions are also called Pareto-optimal solutions or trade-off solutions (David A. *et al*., 2000). Without further knowledge, they are identically important. Usually, it is more difficult to solve a multi-objective optimization problem than to solve a single-objective optimization problem. This is because the search space of a multi-objective optimization problem is often more complex and larger than that of a single-objective optimization problem. Compared with the traditional methods, evolutionary algorithms (EAs) can exhibit more advantages such as gaining a solution set in a single simulation run and often being less susceptible to the characteristics of problems when applied to solve multi-objective optimization problems (Carlos M. *et al*., 1995; E. Mezura-Montes, *et al*., 2008). Currently, EAs have been widely applied to solve multi-objective optimization problems from scientific community and engineering fields (C. A. Coello Coello, 2006).

The aim of designing EAs is to gain a well-converged and well-distributed set approximation to the true Pareto-optimal front of problem in a single simulation run (C. A. Coello Coello, 2006). Hence, it requires to improve the convergence speed and perverse the diversity of solutions during the search of EAs. Accordingly, EAs employ corresponding fitness assignment approaches (E. Zitzler, *et al*., 1999; 2002) and diversity-preserving mechanisms (J. Horn, *et al*., 1994; N. Srinivas, *et al*., 1995; J. Horn, *et al*., 1993; C. Igel, *et al*., 2007; D. E. Goldberg, *et al*., 1987) to achieve their two intentions, respectively. In EAs, the fitness assignment approaches are utilized to guide the search of EAs towards optimal regions, and the representatives of these approaches involve Goldberg's Pareto ranking (D. E. Goldberg, 1989) and Fonseca and Fleming's

Pareto ranking (C. M. Fonseca, *et al*., 1993); while diversity-preserving mechanisms are utilized to enrich the population and avoid the premature convergence, and in the diversity-preserving mechanisms, the effective fitness sharing techniques can improve the diversity of solutions. This paper first introduces two alternative fitness assignment approaches, i.e., Goldberg's Pareto ranking and Fonseca and Fleming's Pareto ranking, and develops three alternative pruning techniques. In order to implement these pruning techniques and prune excess individuals more effectively, a fast and effective density-estimating technique without using parameters is employed (K. Deb, 2002). Then, through combining these approaches and techniques, several differential evolution (R. Storn, *et al*., 1997) algorithms for multi-objective optimization are given. Numerical experiments are conducted on a test set involving seven scalable multi-objective optimization problems to be minimized. The experimental results indicate the relative performances between two alternative Pareto ranking approaches and among three alternative pruning techniques, respectively. Finally, in order to further enrich the diversity of solutions, a dynamic mutation operator (Z. Michalewicz, 1996) is introduced.

This paper is organized as follows: First, we introduce two alternative fitness assignment approaches based on Pareto ranking in Section 2. Then, we develop three pruning techniques in Section 3 and introduce a fast and effective technique for estimating the individual crowded density in Section 4. Next, we present the experimental results, compare different fitness assignment approaches, and contrast different pruning techniques in Section 5. Finally, some conclusions and future work are given in Section 6.

2. Two Alternative Fitness Assignment Approaches

2.1 Goldberg's Pareto Ranking

Based on the concept of Pareto optimality (David A., 2000), Goldberg (D. E. Goldberg, 1989) first proposed the basic idea of Pareto ranking to determine the individual rank for each individual in the current population. According to these ranks and a certain selection strategy, some individuals with better ranks have the relatively bigger probabilities to be selected into the next generation population. The procedure of Goldberg's Pareto ranking is described as follows (D. E. Goldberg, 1989): First, the current population is sorted by the concept of Pareto optimality, some non-dominated individuals are obtained and assigned rank 1, then remove these individuals from the current population and the remaining individuals constitute the current population; Repeatedly, first, the current population is sorted by the concept of Pareto optimality, some non-dominated individuals are obtained and assigned rank 2, then remove these individuals from the current population and the remaining individuals constitute the current population; Continue to the above steps till the current population becomes empty when all individuals have certain ranks. The basic idea of Pareto ranking can be illustrated in Figure 1.

2.2 Fonseca and Fleming's Pareto Ranking

Based on the concept of Pareto optimality, Fonseca and Fleming (Fonseca & Fleming, 1993) proposed another alternative Pareto ranking. According to the basic idea of Fonseca and Fleming's Pareto ranking, the individual $x_i(t)$ from the tth generation population $P(t)$ is assigned rank $R_i(t)$, where the rank function $R_i(t)$ is defined by $R_i(t) = 1 + p_i(t)$, $p_i(t)$ is the number of individuals which are dominated by the individual $x_i(t)$ in the population $P(t)$. The basic idea of the Fonseca and Fleming's Pareto ranking can be illustrated in Figure 2.

3. Three Alternative Pruning Techniques

Let $P(t)$ and $Q(t)$ be the tth generation population with size N to be pruned and its resulting population, respectively. Before implementing three alternative pruning techniques (i.e., three alternative fitness sharing techniques), first, the tth generation population $P(t)$ is sorted according to Pareto ranking, and we can obtain K subpopulations $P_{n_1}(t)$, $P_{n_2}(t)$,…, $P_{n_K}(t)$, where $P(t) = \bigcup_{i=1}^{K} P_{n_i}(t)$, $P_{n_i}(t)$ is the ith subpopulation with the rank n_i. Then let $Q(t) \leftarrow \phi$ and $i \leftarrow 1$, repeat the following loop: while ($|Q(t)| + |P_{n_i}(t)| < N$) { $Q(t) \leftarrow Q(t) \bigcup P_{n_i}(t)$ $i++$;}. After exiting the above loop, if ($|Q(t)| + |P_{n_i}(t)| > N$) then ($|Q(t)| + |P_{n_i}(t)| - N$) individuals from the ith subpopulation $P_{n_i}(t)$ will be pruned, and the resulting population $Q(t)$ with size N will be generated by $Q(t) \leftarrow Q(t) \bigcup P_{n_i}(t)$. In order to prune the ($|Q(t)| + |P_{n_i}(t)| - N$) individuals from the ith subpopulation $P_{n_i}(t)$, three alternative pruning techniques are introduced as follows: 1) We directly first calculate the individual density for each individual in the ith subpopulation $P_{n_i}(t)$, then remove ($|Q(t)|+|P_{n_i}(t)|-N$) individuals from the ith subpopulation $P_{n_i}(t)$. The basic idea of the first pruning technique is illustrated in Figure 3; 2) We first calculate the individual density for each individual from the union of the ith subpopulation $P_{n_i}(t)$ and the $(i-1)$th subpopulation $P_{n_{i-1}}(t)$, then remove ($|Q(t)|+|P_{n_i}(t)|-N$) more crowded individuals from the ith subpopulation $P_{n_i}(t)$. The basic idea of the second pruning technique is illustrated in Figure 4; 3) We calculate the individual density for each individual from the union of the ith subpopulation $P_{n_i}(t)$ and the intermediate population $Q(t)$, then remove ($|Q(t)|+|P_{n_i}(t)|-N$) more crowded individuals from the ith subpopulation $P_{n_i}(t)$. The basic idea of the third pruning technique is illustrated in Figure 5.

4. Estimating the Individual Crowded Density

In order to maintain the diversity of solutions more effectively, some individuals which are located in the more crowded regions will be removed when the number of individuals in the population is larger than a predefined constant value. The crowding-distance estimating approach provides a fast and effective measuring technique for estimating the individual crowded density for each individual in the population. The basic idea of estimating the individual crowding-distance is described as follows (K. Deb, *et al*., 2002): First, along the Kth dimensional direction in the objective space, the Kth distance between the individual x_i and the individual x_j from the population P is calculated as $d_K(x_i,x_j)=|f_K(x_i)-f_K(x_j)|\big/(f_K^{\max}-f_K^{\min})$; Then along the Kth dimensional direction in the objective space, the shortest distance sum $D_K(x_i,P)=\min\limits_{x_{il}\in P_l}\{d_K(x_i,x_{il})\}+\min\limits_{x_{ir}\in P_r}\{d_K(x_i,x_{ir})\}$ of the individual x_i near to two individuals located at each side of the individual x_i is calculated, where $P_l\bigcup\{x_i\}\bigcup P_r=P$, $P_l\bigcup P_r=\phi$, $x_i\notin P_l$ and $x_i\notin P_r$, $\forall x\in P_l$, $f_K(x_i)\geq f_K(x)$ and $\forall x\in P_r$, $f_K(x_i)\leq f_K(x)$. In order to further enrich the diversity of solutions and retain the individuals located at the extreme positions, if ($f_K(x_i)=f_K^{\max}$ or $f_K(x_i)=f_K^{\min}$) then let $D_K(x_i,P)=+\infty$; Finally, the crowding-distance $D(x_i,P)=\sum_{K=1}^{M}D_K(x_i,P)$ of the individual x_i in the population P is obtained, where M is the number of objectives, $f_K^{\max}$ and $f_K^{\min}$ are the maximum objective value and the minimum objective value along the Kth dimensional direction in the objective space, respectively. The basic idea of estimating the individual crowding-distance is illustrated in Figure 6.

5. Experimental Studies

5.1 Benchmark Test Functions and Performance Metrics

In order to gain a specific and important insight into existing approaches, through combining these approaches, we give several differential evolution algorithms for multi-objective optimization. Numerical experiments are conducted on a test set of multi-objective optimization problems (K. Deb, *et al*., 2001; S. Huband, *et al*., 2006). These multi-objective optimization problems are scalable and to be minimized. They have been widely used to valid the performances of evolutionary algorithms for multi-objective optimization. The details of these multi-objective optimization problems are described in Appendix A.

In order to measure the performances including convergence, diversity as well as convergence and diversity of obtained solution set, let Z and PF^* be the obtained solution set and the true Pareto-optimal front (typically, PF^* is produced by uniformly sampling), then convergence metric γ , diversity metric Δ as well as convergence and diversity metric IGD can be correspondingly defined in the following.

Definition 1 (convergence metric γ) Let the convergence metric $\gamma(Z,PF^*)$ be used to measure the closeness of the obtained solution set Z to the true Pareto-optimal front PF^*. This metric can be calculated by

$$\gamma(Z,PF^*)=\tfrac{1}{|Z|}\sum\nolimits_{z\in Z}\min\{\|z-z'\|,z'\in PF^*\} \tag{1}$$

Obviously, the smaller the value of $\gamma(Z,PF^*)$ is, the closer the distance from Z to PF^* is, and the value of $\gamma(Z,PF^*)$ is equal to zero, if and only if $Z\subseteq PF^*$.

Definition 2 (diversity metric Δ) (R. Storn, and K. Price, 1997; T. Okabe, *et al*., 2003) Let the diversity metric Δ be used to measure the distribution of diversity of Z and the spread along the true Pareto-optimal front PF^*. This metric can be calculated by

$$\Delta(Z, PF^*) = \frac{\sum_{e\in E} d(e,Z) + \sum_{z\in Z} | d(z,Z) - \overline{d} |}{\sum_{e\in E} d(d,Z) + | Z | \overline{d}} \tag{2}$$

where $E \subseteq PF^*$ is the set involving all extreme points in PF^*, the function $d(x,S)$ can be defined by $d(x,S) = \min_{y\notin S} \|x - y\|$, where $\overline{d}$ is the arithmetic mean of all $d(z,Z)$, $z \in Z$. $\|x - y\|$ denotes the Euclidean distance between two points x and y. A smaller value of $\Delta(Z, PF^*)$ represents a well-distributed diversity of Z along the true Pareto-optimal front PF^*. If the obtained solution set Z is well distributed and involves the extreme points in PF^*, then $\Delta(Z, PF^*)$ is equal to zero.

Definition 3 (convergence and diversity metric IGD) (T. Okabe, *et al*., 2003; Q. Zhang, *et al*., 2008) Let the convergence and diversity metric IGD be used to measure the average distance from PF^* to Z. This metric can be calculated by

$$IGD(Z, PF^*) = \frac{\sum_{z\in PF^*} d(z,Z)}{| PF^* |} \tag{3}$$

where $d(z,Z)$ is the minimum Euclidean distance from the point z in PF^* to all points in Z. If $| PF^* |$ is large enough to represent the true Pareto-optimal front, $IGD(Z, PF^*)$ is used to measure the convergence and diversity of the obtained solution set Z. A smaller value of $IGD(Z, PF^*)$ indicates a significant quality of Z.

5.2 Investigating Two Alternative Pareto Ranking Approaches

In this study, we first investigate the relative contributions of two alternative Pareto ranking approaches to the quality of obtained solutions. According to Goldberg's Pareto ranking and Fonseca and Fleming's Pareto ranking, we give two corresponding differential evolution algorithms called PR and SP. The parameter settings for all problems (Set A) are listed in Table 1. Experimental Parameters utilized by differential evolution algorithms are given in Table 2. Table 3 gives the convergence results over 20 runs; Table 4 gives the diversity results over 20 runs; Table 5 gives the convergence and diversity results over 20 runs.

According to Table 1, we give the corresponding Pareto-optimal set (PS) and Pareto-optimal front (PF) for each problem in Figure 7 to Figure 13. For DTLZ4, it is not easy to sample the true Pareto-optimal set and the true Pareto-optimal front. Figure 10 might only stand for the part of the true Pareto-optimal set and the true Pareto-optimal front of DTLZ4.

In order to more intuitively observe the experimental results, we present the obtained Pareto-optimal front for each problem in terms of two alternative differential evolution algorithms PR and SP in Figure 14 to Figure 20. According to Figure 14 to Figure 20, we can observe that SP and PR can find an approximation set to the true Pareto-optimal front for each problem above in a single simulation run.

According to Table 3, we can find that for all problems, the best results obtained by SP are slightly better than those obtained by PR in terms of convergence. For DTLZ1, DTLZ5, and DTLZ6, the mean results obtained by SP are slightly better than those obtained by PR, while for DTLZ2, and DTLZ7, the mean results obtained by SP are slightly worse than those obtained by PR. It is interesting to note that for DTLZ3, SP can find an approximation set for each run out of 20 runs. Unfortunately, PR failed to find an approximation set for each run out of 20 runs. At the same time, for most of problems DTLZ1, DTLZ2, DTLZ5, and DTLZ6, the worst results obtained by SP are slightly worse than those obtained by PR, while for DTLZ3 and DTLZ7, the worst results obtained by SP are slightly superior to those obtained by PR. It signifies that SP outperforms PR when applied to solve more complex problems.

According to Table 4, we can observe that for all problems, SP and PR can obtain the approximate mean, best, worst, std (standard deviation) results. It is worthy of noting that for DTLZ3, although PR failed to find an approximation set for each run out of 20 runs, the obtained approximation set keeps the better diversity of solutions.

According to Table 5, for DTLZ1, DTLZ3, DTLZ5, and DTLZ7, the mean results produced by SP are slightly superior to those produced by PR, while for other problems, the mean results produced by SP are worse than those produced by PR. For DTLZ1-DTLZ5, we can find that the best results produced by SP are slightly worse than those produced by PR, while for other problems, the best results produced by SP are slightly better than those produced by PR. For DTLZ1~ DTLZ2, and DTLZ4~ DTLZ7, the worst results obtained by SP are approximate to those obtained by PR. In general, the mean performance of SP is superior to that of PR.

5.3 Investigating Three Alternative Pruning Techniques

In this section, we continue to further investigate the relative contributions of three alternative pruning techniques to the quality of obtained solutions. According to the first pruning technique, the second pruning technique, and the third pruning technique, we create three corresponding differential evolution algorithms called DPA, DPB, and DPC. They all employ Fonseca and Fleming's Pareto ranking approach. The parameter settings for all problems (Set B) are listed in Table 6. Experimental Parameters utilized by DPA, DPB, and DPC are given in Table 7. Table 8 gives the convergence results over 10 runs; Table 9 gives the diversity results over 10 runs; Table 10 gives the convergence and diversity results over 10 runs.

In the process of experimental simulation, we find that for two problems DTLZ1 and DTLZ3, each algorithm DPA, DPB, or DPC has a certain possibility of getting struck in local Pareto-optimal fronts in a single simulation run. In order to more intuitively observe the shapes of local Pareto-optimal fronts, Figure 21 illustrates two different local Pareto-optimal fronts obtained in two independent simulation runs with respect to DTLZ1; Figure 22 illustrates two different local Pareto-optimal fronts obtained in two independent simulation runs with respect to DTLZ3. Seen from Figure 21 and Figure 22, these local Pareto-optimal fronts still exhibit better diversity and profile. It indicates that diversity-preserving mechanisms employed in these algorithms are effective.

According to Table 8, for four problems DTLZ2, DTLZ5, DTLZ6, and DTLZ7, the mean and best results obtained by three algorithms DPA, DPB, and DPC are approximate. For DTLZ1, The mean performance of DPA is superior to those of DPB and DPC, the best and std (standard deviation) results obtained by these algorithms are slightly different, and the worst result of DPB is the worst among three algorithms. For DTLZ3, three algorithms DPA, DPB, and DPC failed to find an approximation set for each run out of 10 runs, and they perform approximately.

According to Table 9, for each problem, each algorithm can obtain the approximate performances with respect to the mean, best, worst, and std experimental results.

According to Table 10, for most of problems DTLZ1, DTLZ2, DTL3, DTLZ5, and DTLZ7, the mean performance of DPC is superior to that of DPA or DPB. The initial results indicate that the third pruning technique is more effective than the first pruning technique, and the second pruning technique.

5.4 Improving the Search Abilities by Incorporating Other Search Techniques

In this section, in order to improve the search capability and avoid the premature convergence, we introduce a dynamic mutation operator, which can be described as follows (Storn and Price, 1997):

$$x_i'(t) = \begin{cases} x_i(t) + \chi(U - x_i(t), t, T, b) \text{ if } \xi \le 0.5 \\ x_i(t) - \chi(L - x_i(t), t, T, b) \text{ otherwise} \end{cases} \tag{4}$$

where the mutated individual vector $x_i'(t)$ is generated by the target individual vector $x_i(t)$, ξ is a random number from the interval [0, 1], U and L are the upper and lower bounds of the target individual vector $x_i(t)$, respectively. The function χ is defined as $\chi(y,t,T,b) = \xi \cdot y \cdot (1 - t/T)^b$, where t and T are the generation number and the maximum generation number, respectively. The parameter b is often predefined as 2 or 3.

Next, let $Q(t)$ denote the offspring population of the tth generation population $P(t)$; let $R(t)$ denote the mutated population of the tth generation population $P(t)$ according to Equation (4). After generating the offspring population $Q(t)$ and the mutated population $R(t)$ of the tth generation population $P(t)$, let $P(t) \leftarrow P(t) \bigcup Q(t) \bigcup R(t)$, and select N better individuals from the union population $P(t)$ into the $(t+1)$th population $P(t+1)$, where N is the population size. We call this modified version H-DPC of DPC.

The parameter settings for two problems DTLZ1 and DTLZ3 are referred to Table 6; Experimental parameters for two differential evolution algorithms DPC and H-DPC are referred to Table 7. Finally, Table 11 gives the convergence results over 10 runs with respect to DTLZ1 and DTLZ3; Table 12 gives the diversity results over 10 runs with respect to DTLZ1 and DTLZ3; Table 13 gives the convergence and diversity results over 10 runs with respect to DTLZ1 and DTLZ3.

For DTLZ1 and DTLZ3, Table 11 shows that the convergence performances of H-DPC are obviously superior to those of DPC; Table 12 shows that the diversity performances of H-DPC are approximate to those of DPC; Table 13 shows that the convergence and diversity performances of H-DPC are obviously better than those of DPC. It indicates that an effective search operator can improve the performance of an evolutionary algorithm.

6. Conclusions and Future Work

This paper introduces two alternative fitness assignment approaches based on Pareto ranking to guide the search of EAs towards optimal regions, three alternative diversity-preserving pruning techniques (i.e., three alternative fitness sharing techniques), and incorporates a dynamic mutation operator into differential evolution algorithms in order to enrich the diversity of solutions to premature convergence. Through combining these approaches and/or techniques, we present several specific differential evolution algorithms for multi-objective optimization, and compare and contrast these algorithms.

As to future work, some research directions will be considered as follows: On the one hand, during the search, how to identify better individuals more effectively is still our research work. As shown in Figure 23, the individual (white circle) is easier to be rejected than any other individuals (back circles) in the population. In fact, the individual is a potential solution. On the other hand, we will also further introduce more effective fitness assignment approaches to improve the convergence speed, and more effective fitness sharing techniques and diversity-preserving mechanisms to enrich the diversity of solutions.

References

A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, & E. Tsang. (2006). Combining Model-Based and Genetic-Based Offspring Generation for Multi-Objective Optimization Using a Convergence Criterion. In: *2006 IEEE Congress on Evolutionary Computation (CEC'2006)*, 3234-3241. http://dx.doi.org/10.1109/CEC.2006.1688406

C. A. Coello Coello. (2006). Evolutionary Multiobjective Optimization: A Historical View of the Field. *IEEE Computational Intelligence Magazine*, 1(1), 28-36. http://dx.doi.org/10.1109/MCI.2006.1597059

C. Igel, N. Hansen, & S. Roth. (2007). Covariance Matrix Adaptation for Multi-Objective Optimization. *Evolutionary Computation*, 15(1), 1-28. http://dx.doi.org/10.1162/evco.2007.15.1.1

C. M. Fonseca, & P. J. Fleming. (1993). Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion, and Generalization. In: Forrest, S., editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 416-423, Morgan Kaufmann, San Mateo, California.

Carlos M. Fonseca, & Peter J. Fleming. (1995). An Overview of Evolutionary Algorithms in Multiobjective Optimization. *Evolutionary Computation*, 3(1), 1-16. http://dx.doi.org/10.1162/evco.1995.3.1.1

D. E. Goldberg, & J. Richardson. (1987). Genetic Algorithms with Sharing for Multimodal Function Optimization. In *Proceedings of the Second International Conference on Genetic Algorithms and Their Application*, 41-49.

D. E. Goldberg. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, Massachusetts. ISBN: 0201157675

David A. Van Veldhuizen, & Gary B. Lamont. (2000). Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art. *Evolutionary Computation*, 8(2), 125-147. http://dx.doi.org/10.1162/106365600568158

E. Mezura-Montes, M. Reyes-Sierra, & C. A. Coello Coello. (2008). Multi-Objective Optimization Using Differential Evolution: A Survey of the State-of-the-Art. *Advances in Differential Evolution, Studies in Computational Intelligence*, 143, 173-196. Berlin: Springer. http://dx.doi.org/10.1007/978-3-540-68830-3_7

E. Zitzler, & L. Thiele. (1999). Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4), 257-271. http://dx.doi.org/10.1109/4235.797969

E. Zitzler, M. Laumanns, & L. Thiele. (2002). SPEA2: Improving the Strength Pareto Evolutionary Algorithm. In: *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems (EUROGEN 2001)*, 95-100.

J. Horn, & N. Nafpliotis. (1993). Multiobjective Optimization Using the Niched Pareto Genetic Algorithm. *Technical Report 930005*, Illinois Genetic Algorithms Laboratory (IlliGAL), University of Illinois, Urbana, Illinois.

J. Horn, N. Nafpliotis, & D. E. Goldberg. (1994). A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In: Michalewicz, Z., editor, *Proceedings of the First IEEE Conference on Evolutionary Computation*, 82-87. IEEE Press, Piscataway, New Jersey. http://dx.doi.org/10.1109/ICEC.1994.350037

K. Deb, A. Pratap, S. Agarwal, & T. Meyarivan. (2002). A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182-197. http://dx.doi.org/10.1109/4235.996017

K. Deb, L. Thiele, M. Laumanns, & E. Zitzler. (2001). Scalable Test Problems for Evolutionary Multi-Objective Optimization. *TIK-Report 112*, Swiss Federal Institute of Technology, Zurich.

N. Srinivas, & K. Deb. (1995). Multiobjective Optimization Using Nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3), 221-248. http://dx.doi.org/10.1162/evco.1994.2.3.221

Q. Zhang, A. Zhou, S. Z. Zhao, P. N. Suganthan, W. Liu, & S. Tiwari. (2008). Multiobjective Optimization Test Instances for the CEC 2009 Special Session and Competition. *Technical Report CES-487*, University of Essex and Nanyang Technological University. http://dces.essex.ac.uk/staff/qzhang/moeacompetition09.htm

R. Storn, & K. Price. (1997). Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4), 341-359. http://dx.doi.org/10.1023/A:1008202821328

S. Huband, P. Hingston, L. Barone, & L. While. (2006). A Review of Multi-Objective Test Problems and Scalable Test Problem Toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5), 477-506. http://dx.doi.org/10.1109/TEVC.2005.861417

T. Okabe, Y. Jin, & B. Sendhoff. (2003). A Critical Survey of Performance Indices for Multi-Objective Optimization. In: *Proceedings of the IEEE Congress on Evolutionary Computation (CEC'2003)*, 2, 878-885. http://dx.doi.org/10.1109/CEC.2003.1299759

Z. Michalewicz. (1996). *Genetic Algorithms + Data Structures = Evolution Programs (3rd Edition)*. Springer-Verlag, New York.

Table 1. Parameter settings for all problems (Set A)

function	k	M	n
DTLZ1	5	3	7
DTLZ2	10	3	12
DTLZ3	10	3	12
DTLZ4	10	3	12
DTLZ5	10	3	12
DTLZ6	10	3	12
DTLZ7	20	3	22

Table 2. Parameter settings for all algorithms (Set A)

population size	generation number	scaling factor	crossover probability
200	200	0.5	0.2

Table 3. Experimental results (over 20 runs) with respect to convergence γ (Set A)

function	method	mean	best	worst	Std
DTLZ1	SP	2.031838e-03	1.863285e-03	2.181295e-03	6.637889e-05
	PR	2.044266e-03	1.994832e-03	2.131832e-03	3.951647e-05
DTLZ2	SP	5.302748e-03	5.014488e-03	5.597901e-03	1.440265e-04
	PR	5.282749e-03	5.017556e-03	5.442548e-03	1.264624e-04
DTLZ3	SP	6.517009e-03	5.536080e-03	8.646826e-03	7.061516e-04
	PR	5.649878e-02	5.775278e-03	1.002519e+00	2.226712e-01
DTLZ5	SP	3.825585e-03	3.629720e-03	4.041557e-03	9.096097e-05
	PR	3.838241e-03	3.640095e-03	3.983132e-03	9.261637e-05
DTLZ6	SP	3.853767e-03	3.630669e-03	4.041242e-03	8.849952e-05
	PR	3.854559e-03	3.703397e-03	3.978090e-03	6.777661e-05
DTLZ7	SP	1.702500e-02	1.454956e-02	1.909859e-02	1.321462e-03
	PR	1.698686e-02	1.517265e-02	2.215507e-02	1.665180e-03

Table 4. Experimental results (over 20 runs) with respect to diversity Δ (Set A)

function	method	mean	best	worst	std
DTLZ1	SP	5.628245e-01	5.164370e-01	6.309867e-01	2.977525e-02
	PR	5.698309e-01	5.202605e-01	6.146444e-01	2.722818e-02
DTLZ2	SP	4.593718e-01	3.945783e-01	5.550712e-01	4.088541e-02
	PR	4.500276e-01	3.685823e-01	5.054371e-01	3.789337e-02
DTLZ3	SP	4.579975e-01	3.597210e-01	5.151380e-01	4.486129e-02
	PR	4.664997e-01	3.893994e-01	5.467750e-01	4.116880e-02
DTLZ5	SP	2.971913e-01	2.587800e-01	3.335140e-01	2.117458e-02
	PR	2.987648e-01	2.507607e-01	3.271258e-01	1.898881e-02
DTLZ6	SP	2.573789e-01	2.245677e-01	2.881078e-01	1.506331e-02
	PR	2.501997e-01	2.223664e-01	2.818551e-01	1.608989e-02
DTLZ7	SP	6.185258e-01	5.000880e-01	7.281582e-01	6.870801e-02
	PR	6.094301e-01	4.828956e-01	7.533552e-01	6.973806e-02

Table 5. Experimental results (over 20 runs) with respect to IGD (Set A)

function	method	mean	best	worst	std
DTLZ1	SP	1.571187e-02	1.535013e-02	1.627591e-02	2.925840e-04
	PR	1.592578e-02	1.513805e-02	1.662548e-02	3.659530e-04
DTLZ2	SP	4.312730e-02	4.190519e-02	4.593635e-02	1.116948e-03
	PR	4.270037e-02	4.107839e-02	4.369294e-02	8.510555e-04
DTLZ3	SP	4.296872e-02	4.138649e-02	4.461358e-02	8.296723e-04
	PR	9.129606e-02	4.095235e-02	1.004348e+00	2.149135e-01
DTLZ5	SP	2.223213e-03	2.033885e-03	2.519202e-03	1.399678e-04
	PR	2.256304e-03	1.995968e-03	2.446413e-03	1.197079e-04
DTLZ6	SP	2.209367e-03	1.879349e-03	2.459724e-03	1.462624e-04
	PR	2.198635e-03	1.981355e-03	2.385968e-03	1.078765e-04
DTLZ7	SP	5.332914e-02	4.986150e-02	5.970970e-02	2.642792e-03
	PR	5.502353e-02	5.035829e-02	6.624393e-02	3.781681e-03

Table 6. Parameter settings for all problems (Set B)

function	k	M	n
DTLZ1	25	3	27
DTLZ2	50	3	52
DTLZ3	50	3	52
DTLZ4	50	3	52
DTLZ5	50	3	52
DTLZ6	50	3	52
DTLZ7	100	3	102

Table 7. Parameter Settings for all algorithms (Set B)

population size	generation number	scaling factor	crossover probability
200	5000	0.5	0.2

Table 8. Experimental results (over 10 runs) with respect to convergence γ (Set B)

function	method	mean	best	worst	std
DTLZ1	DPA	3.552232e-02	1.918474e-03	3.370166e-01	1.059343e-01
	DPB	1.752185e-01	1.892351e-03	1.064024e+00	3.419406e-01
	DPC	1.723846e-01	1.976512e-03	6.988795e-01	2.433958e-01
DTLZ2	DPA	5.294251e-03	4.929414e-03	5.508555e-03	1.764598e-04
	DPB	5.343427e-03	5.111571e-03	5.725184e-03	1.711851e-04
	DPC	5.246948e-03	5.014469e-03	5.456907e-03	1.367266e-04
DTLZ3	DPA	1.209321e+01	2.998505e+00	4.371778e+01	1.184372e+01
	DPB	1.049958e+01	1.999012e+00	3.199647e+01	8.827131e+00
	DPC	9.462961e+00	2.387961e+00	2.899799e+01	8.186345e+00
DTLZ5	DPA	3.795862e-03	3.694258e-03	3.968720e-03	9.612101e-05
	DPB	3.797401e-03	3.711054e-03	3.886041e-03	6.180404e-05
	DPC	3.787637e-03	3.677604e-03	3.872392e-03	6.411970e-05
DTLZ6	DPA	3.834440e-03	3.739174e-03	3.965775e-03	6.633829e-05
	DPB	3.810799e-03	3.722204e-03	3.893788e-03	4.598090e-05
	DPC	3.875848e-03	3.773779e-03	4.016359e-03	9.165353e-05
DTLZ7	DPA	1.664319e-02	1.486735e-02	1.808948e-02	1.026120e-03
	DPB	1.666665e-02	1.551426e-02	1.734761e-02	5.933660e-04
	DPC	1.664920e-02	1.491465e-02	1.806811e-02	8.770764e-04

Table 9. Experimental results (over 10 runs) with respect to diversity Δ (Set B)

function	method	mean	best	worst	std
DTLZ1	DPA	5.566587e-01	5.108032e-01	5.824428e-01	2.455997e-02
	DPB	5.558897e-01	5.045916e-01	5.976209e-01	3.756563e-02
	DPC	5.380164e-01	4.872395e-01	6.010286e-01	3.407660e-02
DTLZ2	DPA	4.772971e-01	4.237429e-01	5.058848e-01	2.250187e-02
	DPB	4.743576e-01	4.285477e-01	5.064148e-01	2.186255e-02
	DPC	4.609011e-01	4.044574e-01	5.553018e-01	4.102087e-02
DTLZ3	DPA	4.971516e-01	4.719587e-01	5.368933e-01	2.540926e-02
	DPB	5.185695e-01	4.844420e-01	5.654886e-01	2.181079e-02
	DPC	5.215294e-01	4.602714e-01	5.553033e-01	2.878796e-02
DTLZ5	DPA	2.695051e-01	2.457427e-01	2.882001e-01	1.288009e-02
	DPB	2.715688e-01	2.488244e-01	2.933755e-01	1.491894e-02
	DPC	2.757213e-01	2.569387e-01	2.973063e-01	1.372642e-02
DTLZ6	DPA	2.476024e-01	2.308133e-01	2.748538e-01	1.286597e-02
	DPB	2.460156e-01	2.326828e-01	2.669019e-01	1.147593e-02
	DPC	2.583052e-01	2.417108e-01	2.793733e-01	1.172258e-02
DTLZ7	DPA	6.301640e-01	5.438537e-01	6.967074e-01	6.211428e-02
	DPB	6.389495e-01	5.964780e-01	7.413489e-01	4.192206e-02
	DPC	6.444901e-01	5.515701e-01	7.427199e-01	5.680151e-02

Table 10. Experimental results (over 10 runs) with respect to IGD (Set B)

function	method	mean	best	worst	std
DTLZ1	DPA	4.343832e-02	1.490991e-02	2.910798e-01	8.701401e-02
	DPB	1.563178e-01	1.572475e-02	8.683551e-01	2.751596e-01
	DPC	1.547967e-01	1.526067e-02	5.790862e-01	1.977196e-01
DTLZ2	DPA	4.353963e-02	4.164419e-02	4.510119e-02	1.198702e-03
	DPB	4.350591e-02	4.182151e-02	4.583749e-02	1.269912e-03
	DPC	4.287316e-02	4.190607e-02	4.468166e-02	9.172470e-04
DTLZ3	DPA	1.209452e+01	2.999916e+00	4.371899e+01	1.184367e+01
	DPB	1.050092e+01	2.000648e+00	3.199767e+01	8.827054e+00
	DPC	9.463143e+00	2.377364e+00	2.899937e+01	8.187449e+00
DTLZ5	DPA	2.192472e-03	2.044215e-03	2.394160e-03	1.241114e-04
	DPB	2.241719e-03	1.998765e-03	2.450419e-03	1.523540e-04
	DPC	2.157124e-03	2.021991e-03	2.297052e-03	7.612698e-05
DTLZ6	DPA	2.159979e-03	1.976232e-03	2.429575e-03	1.212622e-04
	DPB	2.161228e-03	1.928455e-03	2.321165e-03	1.084079e-04
	DPC	2.250094e-03	2.028662e-03	2.499558e-03	1.607380e-04
DTLZ7	DPA	5.377720e-02	5.013285e-02	5.810569e-02	2.837050e-03
	DPB	5.551391e-02	4.991557e-02	6.214196e-02	3.621382e-03
	DPC	5.268983e-02	4.952584e-02	5.693206e-02	2.681996e-03

Table 11. Experimental results (over 10 runs) with respect to convergence γ (for DTLZ1 and DTLZ3 in Set B)

function	method	mean	Best	Worst	std
DTLZ1	DPC	1.723846e-01	1.976512e-03	6.988795e-01	2.433958e-01
	H-DPC	2.047708e-03	1.889407e-03	2.126142e-03	6.480516e-05
DTLZ3	DPC	9.462961e+00	2.387961e+00	2.899799e+01	8.186345e+00
	H-DPC	5.280089e-03	4.861896e-03	5.432715e-03	1.851809e-04

Table 12. Experimental results (over 10 runs) with respect to diversity Δ (for DTLZ1 and DTLZ3 in Set B)

function	method	mean	Best	worst	std
DTLZ1	DPC	5.380164e-01	4.872395e-01	6.010286e-01	3.407660e-02
	H-DPC	5.861069e-01	5.289829e-01	6.659675e-01	4.112187e-02
DTLZ3	DPC	5.215294e-01	4.602714e-01	5.553033e-01	2.878796e-02
	H-DPC	4.729079e-01	4.228091e-01	5.414467e-01	3.814277e-02

Table 13. Experimental results (over 10 runs) with respect to convergence and diversity IGD (for DTLZ1 and DTLZ3 in Set B)

function	method	mean	best	worst	std
DTLZ1	DPC	1.547967e-01	1.526067e-02	5.790862e-01	1.977196e-01
	H-DPC	1.608508e-02	1.558558e-02	1.645429e-02	2.549915e-04
DTLZ3	DPC	9.463143e+00	2.377364e+00	2.899937e+01	8.187449e+00
	H-DPC	4.356007e-02	4.241451e-02	4.467135e-02	7.600018e-04

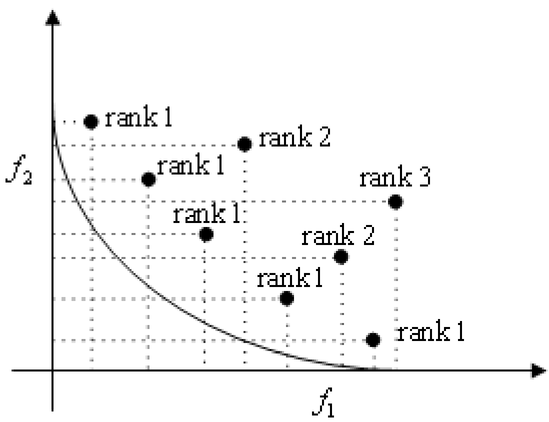

Figure 1. Goldberg's Pareto ranking

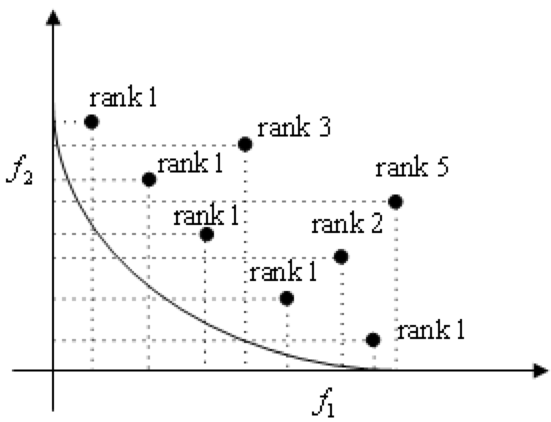

Figure 2. Fonseca and Fleming's Pareto ranking

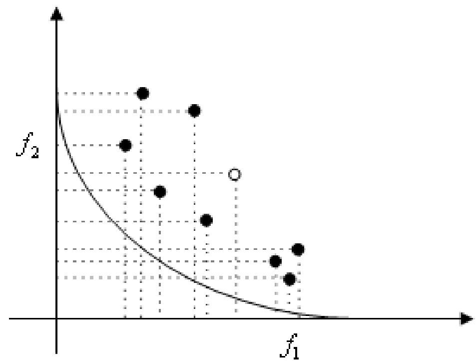

Figure 3. The first pruning technique

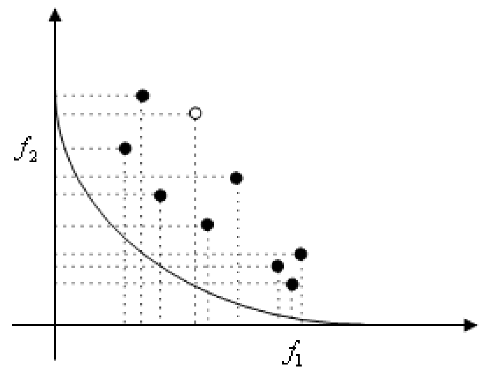

Figure 4. The second pruning technique

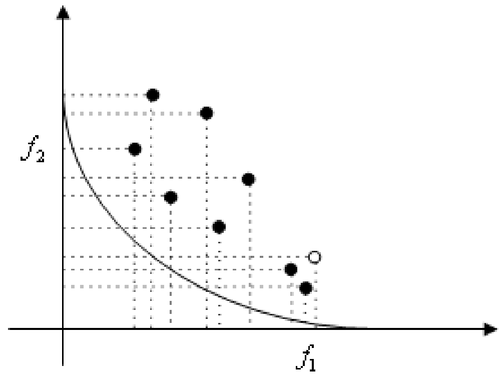

Figure 5. The third pruning technique

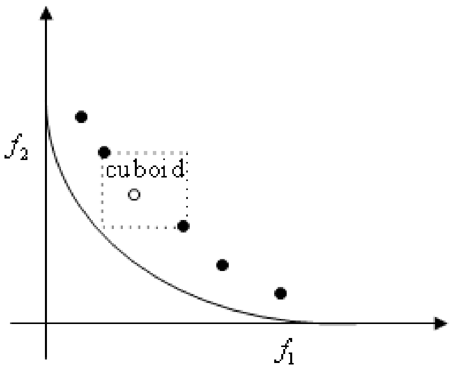

Figure 6. The density-estimating technique

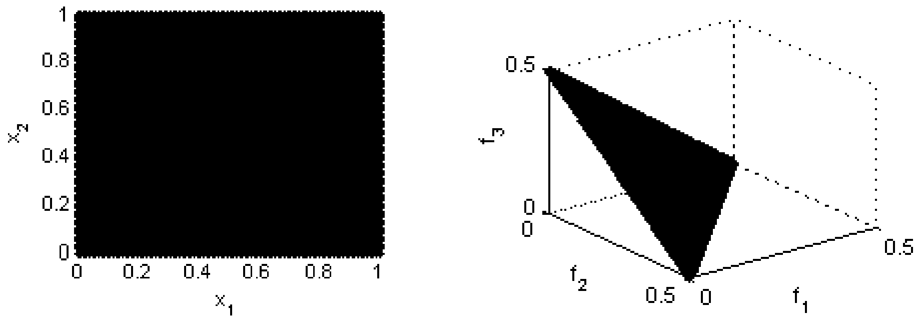

Figure 7. PS (left) and PF (right) of DTLZ1

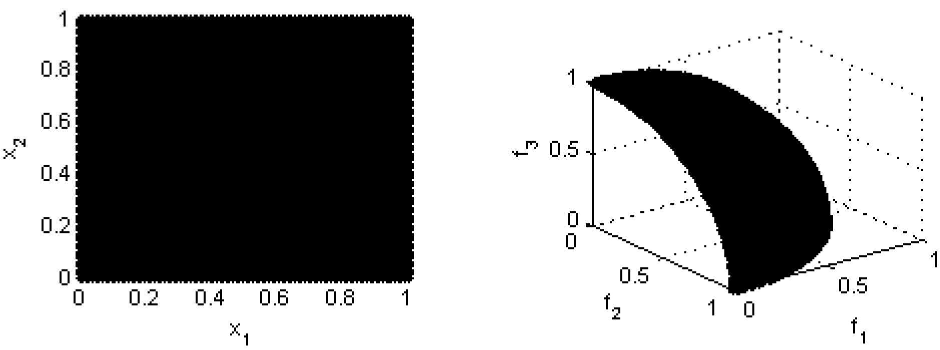

Figure 8. PS (left) and PF (right) of DTLZ2

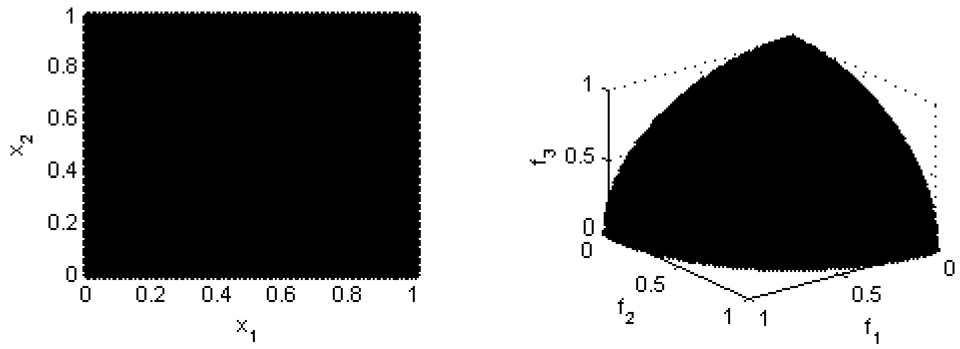

Figure 9. PS (left) and PF (right) of DTLZ3

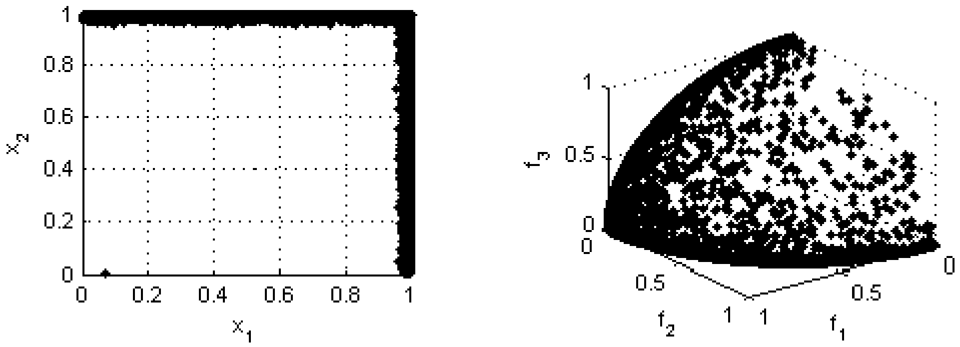

Figure 10. PS (left) and PF (right) of DTLZ4

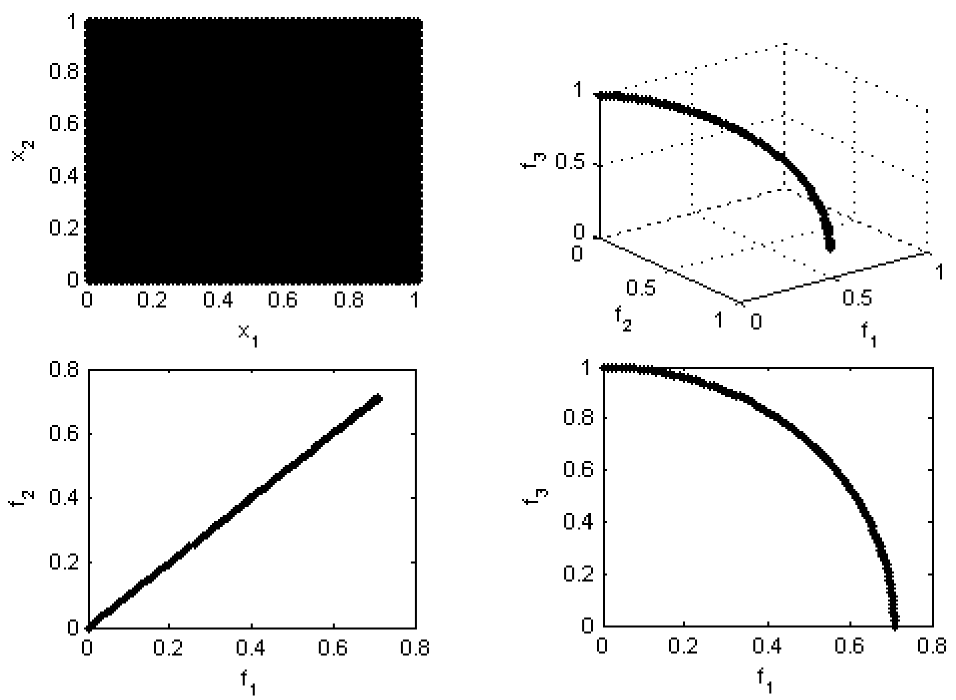

Figure 11. PS (left) and PF (right) of DTLZ5

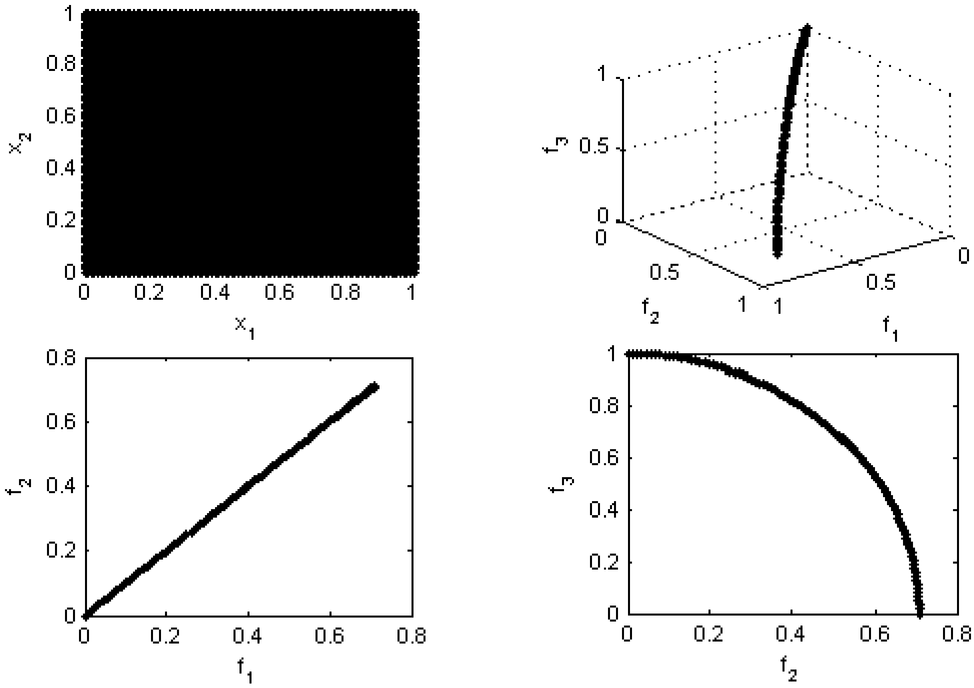

Figure 12. PS (left) and PF (right) of DTLZ6

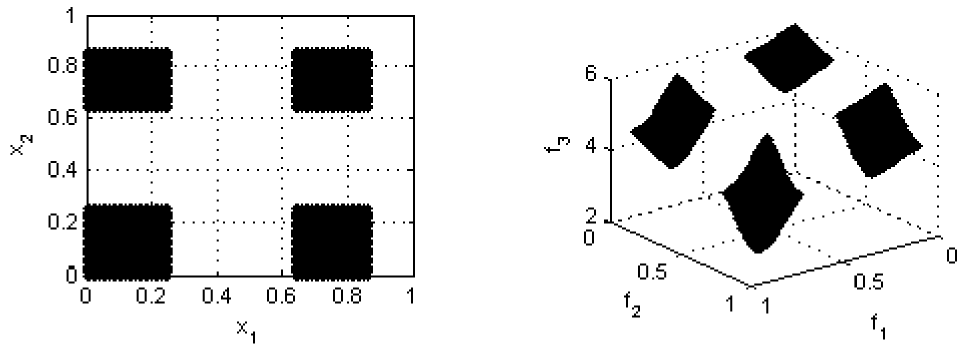

Figure 13. PS (left) and PF (right) of DTLZ7

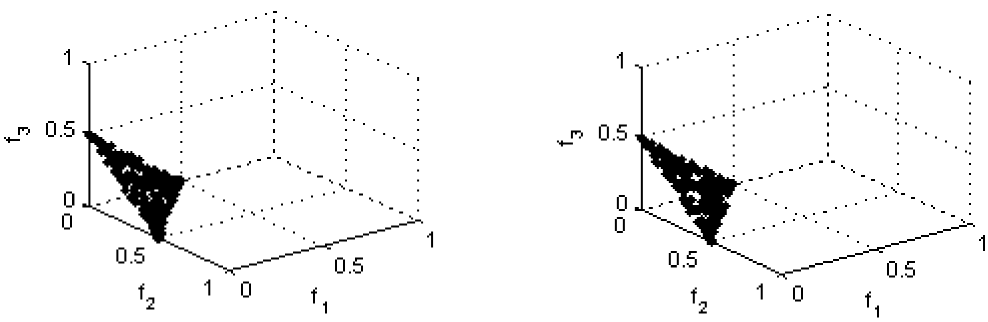

Figure 14. Pareto-optimal fronts of DTLZ1 obtained by SP (left) and PR (right)

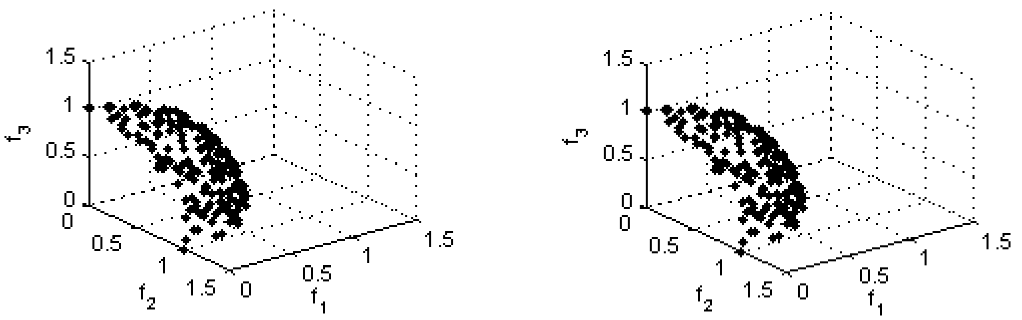

Figure 15. Pareto-optimal fronts of DTLZ2 obtained by SP (left) and PR (right)

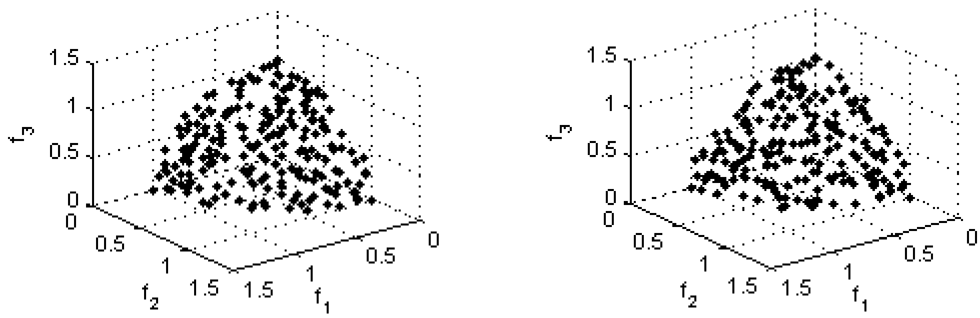

Figure 16. Pareto-optimal fronts of DTLZ3 obtained by SP (left) and PR (right)

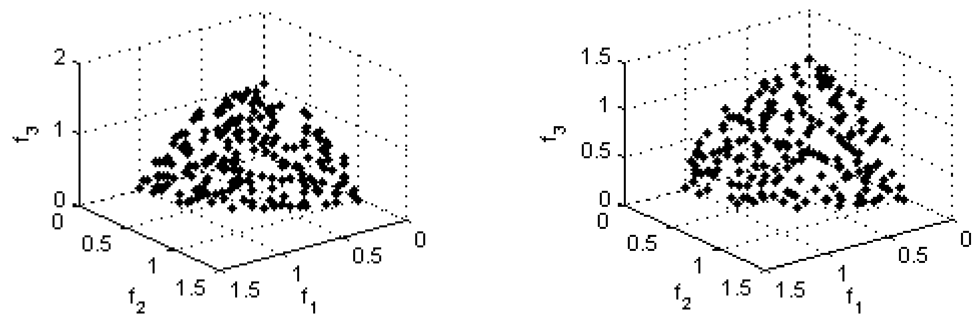

Figure 17. Pareto-optimal fronts of DTLZ4 obtained by SP (left) and PR (right)

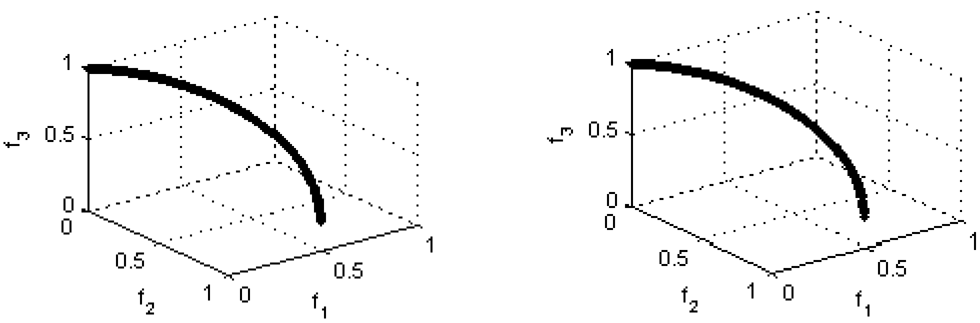

Figure 18. Pareto-optimal fronts of DTLZ5 obtained by SP (left) and PR (right)

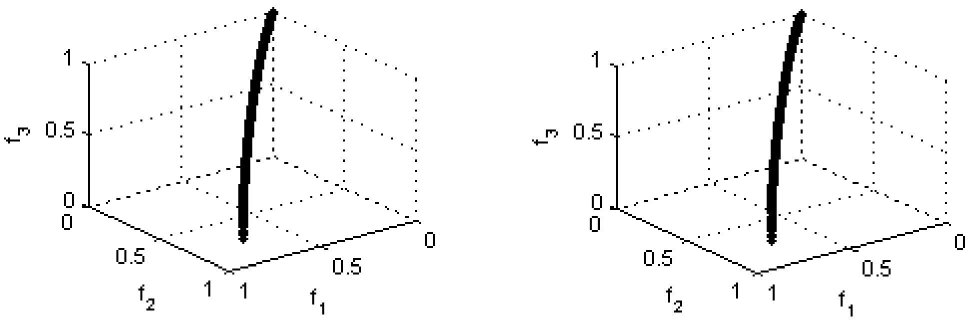

Figure 19. Pareto-optimal fronts of DTLZ6 obtained by SP (left) and PR (right)

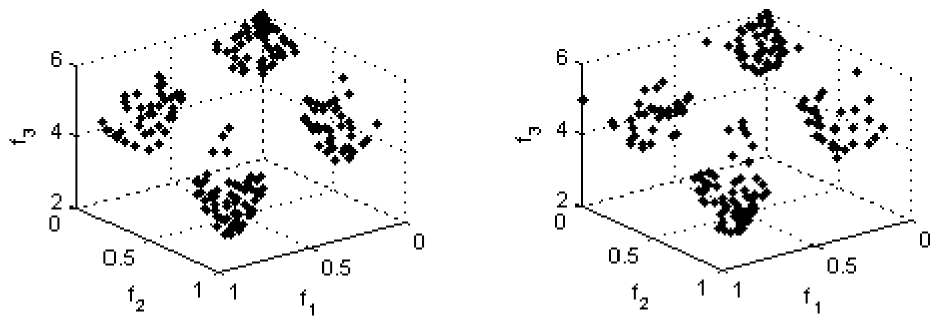

Figure 20. Pareto-optimal fronts of DTLZ7 obtained by SP (left) and PR (right)

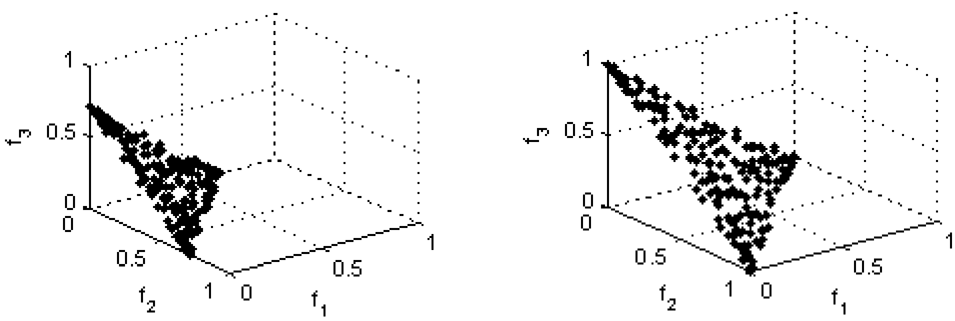

Figure 21. Local Pareto-optimal front of DTLZ1 obtained in a single simulation run

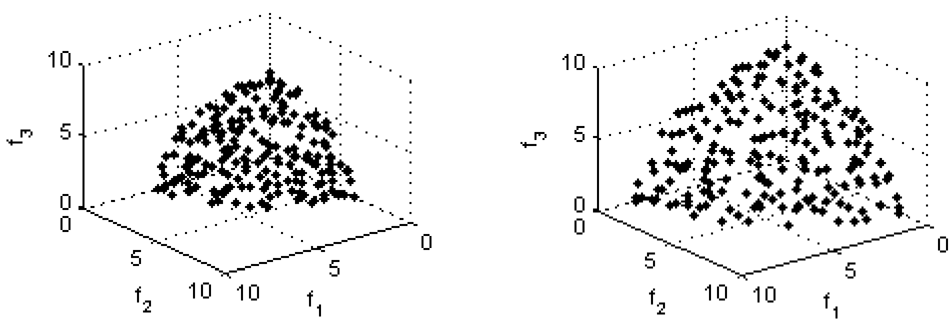

Figure 22. Local Pareto-optimal front of DTLZ3 obtained in a single simulation run

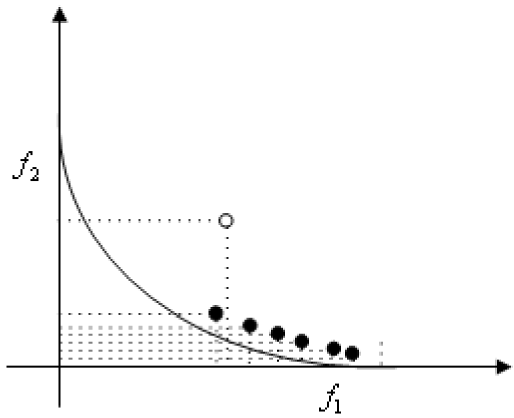

Figure 23. The individual (white circle) is easier to be rejected than any other individuals (black circles)

Appendix A: Benchmark test functions

1) DTLZ1 Function:

Minimize $f_1(x) = 0.5(1+g(x))\prod_{i=1}^{M-1} x_i$

Minimize $f_{m=2:M-1}(x) = 0.5(1+g(x))(1-x_{M-m+1})\prod_{i=1}^{M-m} x_i$

Minimize $f_M(x) = 0.5(1+g(x))(1-x_1)$

Subject to $0 \le x_i \le 1, i = 1,2,...,n$

Where $g(x) = 100[k + \sum_{i=n-k+1}^{n}((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$. The total number of variables is $n = M + k - 1$. A value of $k = 5$ is suggested. The Pareto-optimal solution corresponds to $x_i = 0.5$, $i = n-k+1, n-k+2,....,n$, and the objective function values lie on the linear hyper-plane $\sum_{m=1}^{M} f_m = 0.5$. The search space has $(11^k - 1)$ local Pareto-optimal fronts.

2) DTLZ2 Function:

Minimize $f_1(x) = (1+g(x))\prod_{i=1}^{M-1}\cos(0.5\pi x_i)$

Minimize $f_{m=2:M-1}(x) = (1+g(x))\sin(0.5\pi x_{M-m+1})\prod_{i=1}^{M-m}\cos(0.5\pi x_i)$

Minimize $f_M(x) = (1+g(x))\sin(0.5\pi x_1)$

Subject to $0 \le x_i \le 1, i = 1,2,...n$

Where $g(x) = \sum_{i=n-k+1}^{n}(x - 0.5)^2$. The total number of variables is $n = M + k - 1$. A value of $k = 10$ is suggested. The Pareto-optimal solution corresponds to $x_i = 0.5$, $i = n-k+1, n-k+2,....,n$, and all objective function values satisfy $\sum_{m=1}^{M} f_m^2 = 1$.

3) DTLZ3 Function:

Minimize $f_1(x) = (1+g(x))\prod_{i=1}^{M-1}\cos(0.5\pi x_i)$

Minimize $f_{m=2:M-1}(x) = (1+g(x))\sin(0.5\pi x_{M-m+1})\prod_{i=1}^{M-m}\cos(0.5\pi x_i)$

Minimize $f_M(x) = (1+g(x))\sin(0.5\pi x_1)$

Subject to $0 \le x_i \le 1, i = 1,2,...n$

Where $g(x) = 100[k + \sum_{i=n-k+1}^{n}((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$. The total number of variables is $n = M + k - 1$. A value of $k = 10$ is suggested. The above g function has $(3^k - 1)$ local Pareto-optimal fronts, and the global Pareto-optimal front satisfies $\sum_{m=1}^{M} f_m^2 = 1$, corresponding to $x_i = 0.5$, $i = n-k+1, n-k+2, ..., n$.

4) DTLZ4 Function:
Minimize $f_1(x) = (1+g(x))\prod_{i=1}^{M-1}\cos(0.5\pi x_i^{\alpha})$
Minimize $f_{m=2:M-1}(x) = (1+g(x))\sin(0.5\pi x_{M-m+1}^{\alpha})\prod_{i=1}^{M-m}\cos(0.5\pi x_i^{\alpha})$
Minimize $f_M(x) = (1+g(x))\sin(0.5\pi x_1^{\alpha})$
Subject to $0 \le x_i \le 1, i = 1,2,...n$

Where $g(x) = \sum_{i=n-k+1}^{n}(x-0.5)^2$. The total number of variables is $n = M + k - 1$. A value of $k = 10$ is suggested. The parameter $\alpha = 100$ is suggested. All variables x_1 to x_{M-1} are varied in the range [0, 1]. The global Pareto-optimal solution corresponds to $g^* = 0$ and $\sum_{m=1}^{M} f_m^2 = 1$.

5) DTLZ5 Function:
Minimize $f_1(x) = (1+g(x))\prod_{i=1}^{M-1}\cos(0.5\pi\theta_i)$
Minimize $f_{m=2:M-1}(x) = (1+g(x))\sin(0.5\pi\theta_{M-m+1})\prod_{i=1}^{M-m}\cos(0.5\pi\theta_i)$
Minimize $f_M(x) = (1+g(x))\sin(0.5\pi\theta_1)$
Subject to $0 \le x_i \le 1, i = 1,2,...n$

Where $\theta_1 = x_1$, $\theta_i = \frac{1+2g(x)x_i}{2(1+g(x))}$, $i = 2,3,..., M-1$, and $g(x) = \sum_{i=n-k+1}^{n}(x-0.5)^2$. The total number of variables is $n = M + k - 1$. A value of $k = 10$ is suggested.

6) DTLZ6 Function:
Minimize $f_1(x) = (1+g(x))\prod_{i=1}^{M-1}\cos(0.5\pi\theta_i)$
Minimize $f_{m=2:M-1}(x) = (1+g(x))\sin(0.5\pi\theta_{M-m+1})\prod_{i=1}^{M-m}\cos(0.5\pi\theta_i)$
Minimize $f_M(x) = (1+g(x))\sin(0.5\pi\theta_1)$
Subject to $0 \le x_i \le 1, i = 1,2,...n$

Where $\theta_1 = x_1$, $\theta_i = \frac{1+2g(x)x_i}{2(1+g(x))}$, $i = 2,3,..., M-1$, and $g(x) = \sum_{i=n-k+1}^{n} x_i^{0.1}$. The total number of variables is $n = M + k - 1$. A value of $k = 10$ is suggested.

7) DTLZ7 Function:
Minimize $f_{m=1:M-1}(x) = x_m$
Minimize $f_M(x) = (1+g(x))h(f_1, f_2, ..., f_{M-1}, g(x))$
Where $g(x) = 1 + \frac{9}{k}\sum_{i=n-k+1}^{n} x_i$,
and $h(f_1, f_2, ..., f_{M-1}, g(x)) = M - \sum_{i=1}^{M}[\frac{f_i(x)}{1+g(x)}(1+\sin(3\pi f_i(x)))]$
Subject to $0 \le x_i \le 1, i = 1,2,...n$.

The total number of variables is $n = M + k - 1$. A value of $k = 20$ is suggested. This problem has 2^{M-1} disconnected Pareto-optimal regions in the search space. The global Pareto-optimal solutions correspond to $x_i = 0$, $i = n-k+1, n-k+2, ..., n$.