A Fast Strength Pareto Evolutionary Algorithm Incorporating Predefined Preference Information

Shouyong Jiang
Centre for Computational Intelligence (CCI)
School of Computer Science and Informatics
De Montfort University
The Gateway, Leicester LE1 9BH, U.K.
Email: shouyong.jiang@email.dmu.ac.uk

Shengxiang Yang
Centre for Computational Intelligence (CCI)
School of Computer Science and Informatics
De Montfort University
The Gateway, Leicester LE1 9BH, U.K.
Email: syang@dmu.ac.uk

***Abstract*—Strength Pareto Evolutionary Algorithm 2 (SPEA2) has achieved great success for handling multiobjective optimization problems. However, it has been widely reported that SPEA2 gets subjected to a huge amount of computational effort while pursuing a good distribution of approximated solutions. This paper explores a new way to keep the good properties of SPEA2 and reduce its high computational burden simultaneously, with the aid of predefined preference information. By incorporating preference information, the proposed fast SPEA (FSPEA) can efficiently perform individuals' density estimation and environmental selection, thus speeding up the whole running time of the evolution process. Empirical studies show that the proposed FSPEA algorithm can obtain very competitive performance on a number of multiobjective test problems considered in this paper.**

I. Introduction

After decades of research, evolutionary algorithms (EAs) have demonstrated their effectiveness for solving various mathematical and real-world optimization problems [3]. One important class of those optimization problems in which they are involved is multiobjective optimization problems (MOPs), which have at least two objectives that conflict one another, and the corresponding algorithms are generally called multiobjective optimization EAs (MOEAs). Compared with conventional mathematical programming approaches, MOEAs do not require any gradient information of problems to be optimized. In contrast, they employ a population of candidate individuals and evolve them simultaneously, producing a set of tradeoff solutions, known as Pareto-optimal set (POS), in a single run. Then, the POS provides decision-makers with a wide range of options for balancing objectives, according to problem-specific requirements. Thus, attainment of a set of well-converged and well-diversified solutions is the fundamental goal of MOEAs.

Most early-developed MOEAs use Pareto-dominance relations [5] to induce discrimination between two solutions in a population. For example, for two solutions x and y, x is said to dominate y if and only if x is not worse than y on all objectives and better than y on at least one objective, and this is denoted as $x \succeq y$. Without loss of generality, minimization is considered throughout the paper. On the basis of such relations, a number of Pareto-based techniques were proposed in the mid-nineties, e.g., multiobjective optimization genetic algorithm (MOGA) [6], niched Pareto genetic algorithm (NPGA) [7], and nondominated sorting genetic algorithm (NSGA) [8]. These foundational techniques established the utility of MOEAs for multiobjecitve optimization. A few years later, MOEA developers began to recognize the importance of elitism and subsequently proposed a couple of elitist MOEAs, e.g., strength Pareto EA (SPEA) [9], Pareto-envelope based selection algorithm (PESA) [11], and Pareto archived evolution strategy (PAES) [10]. In addition to elitism, these evolutionary algorithms also introduce diversity maintenance and external archiving and have shown good performance in various studies. Coincidentally, further contributions have been made by adapting the early version of MOEAs, for example, an improved NSGA, i.e., NSGA-II [15], takes into account elitism, computational burden, and enhanced diversity preservation, and SPEA2 [18] uses an improved fitness assignment scheme, a new density estimator, and an improved archive truncation method, to augment algorithm's performance.

Amongst Pareto-based MOEAs, NSGA-II and SPEA2 are two most popular methods for multiobjective optimization and have been regarded as benchmarks for algorithm comparison in various studies. The main difference between SPEA2 and NSGA-II lies in diversity maintenance. NSGA-II uses crowding distance to maintain a well-spread nondominated set whereas SPEA2 employs a k-nearest neighbour density estimation technique. A number of studies have shown that there is no much significant difference between the performance of NSGA-II and that of SPEA2 on a wide range of MOPs [18], [17], [2], although the former has better approximation capability while the latter presents better diversity performance. In high-dimensional problems, SPEA2 provides a better distribution than NSGA-II [1], [16], [2], [4]. Despite that, the former is yet to receive as much research interest as the latter over the past years. One reason for this favouritism is that SPEA2 [18] employs a k-th nearest neighbour method to estimate individuals' density, leading to a notable increase in the computational load and thus inhibiting its application.

In this paper, a new version of SPEA2, called fast SPEA (FSPEA), is presented, where we propose to substitute a set of predefined preferences for the k-th nearest neighbour method during the density estimation process. That is, individuals' density is estimated by the closeness to the preferences, which allows a more precise guidance of the search process. Besides helping density estimation, the supplied preferences are also beneficial for environmental selection. When there are too many nondominated solutions in the archive population, the preferences induce truncation in a way that the most diverse solutions are preserved for the next generation. Given that the constructed preferences are well-distributed enough, the method practically provides a good spread of points along the Pareto-optimal Front (POF).

The reminder of the paper is organized as follows. Section II presents a brief review of related work in the literature. The proposed algorithm is offered in Section III, and experimental studies and comparison are presented in Section IV. Section V concludes the paper and suggests some future research directions.

II. Related Work

A. The SPEA2 Algorithm

The SPEA2 [18] algorithm is an improvement over the original SPEA [9], and we give a brief summary of the algorithm here. For a more detailed description the interested reader is referred to [18].

SPEA2 starts with an initial population P_0 and an empty archive $\overline{P_0}$. In the t-th iteration, the current population P_t and the archive population $\overline{P_t}$ are merged to form a combined population $P_t \cup \overline{P_t}$. Each member i in $P_t \cup \overline{P_t}$ is first assigned a strength value $S(i)$, representing the number of solutions it dominates:

$$S(i) = C(\{j \in P_t \cup \overline{P_t} | i \succeq j\}) \qquad (1)$$

where $C(\cdot)$ denotes the cardinality of a set. The strength value then contributes to computing the raw fitness $R(i)$ of an individual i by

$$R(i) = \sum_{j \in P_t \cup \overline{P_t}, j \succeq i} S(j) \qquad (2)$$

The raw fitness is determined by the strengths of its dominators in the combined population, and a high $R(i)$ means that i is dominated by many individuals, while $R(i) = 0$ implies i is a nondominated individual.

To discriminate individuals that have identical raw fitness values, SPEA2 considers additional density information $D(i)$ of individual i, which is estimated by an adapted k-th nearest neighbour method. SPEA2 calculates $D(i)$ as follows:

$$D(i) = \frac{1}{\sigma_i^k + 2} \qquad (3)$$

where σ_i^k is the distance to the k-th nearest neighbour (in the objective space). A small $D(i)$ value means i resides in a roomy area.

The final fitness $F(i)$ of individual i is composed of the raw fitness and additional density information, combining in a compact form:

$$F(i) = R(i) + D(i) \qquad (4)$$

Afterwards, the archive is updated by saving all nondominated solutions (whose $F(i)$ values are lower than one) from the combined population. If the updated archive size exactly equals a predefined limit, the environmental selection is completed. Otherwise, there can be two situations: the archive is either too small or too large. In the first case, the archive is filled with the best dominated solutions. In the second case, an archive truncation procedure is invoked such that the solutions with the minimum distance to other solutions are removed iteratively until the archive size is equal to the predefined value. If there are several individuals with an identical minimum distance, the tie is broken by considering the second smallest distance, and so forth.

If the stopping condition is not met, SPEA2 continues with the mating selection where individuals from the new archive are selected through binary tournaments. Finally, after recombination and mutation, the old population is replaced by the resulting offspring.

B. Work Related to SPEA2

Since its introduction, SPEA2 has received great research interest and has been successfully applied to a variety of theoretical and practical MOPs [18], [17], [16]. A major reason for its success is that SPEA2 can provide a good spread of solutions for both bi-objective and higher-dimensional problems [1], [16], [2], [4].

On the other hand, there has been some work on the improvement of SPEA2. In [19], neighbourhood crossover and mating selection are introduced to improve search ability, and two archives holding diverse solutions in the objective and variable spaces are used to promote population diversity. In [20], SPEA2 is integrated with cooperative coevolution for solving three-objective problems. Li *et al.* [21] proposed to incorporate problem-specific local search strategies into SPEA variants for a better exploiting capability. Another version of SPEA2, proposed in [22], uses adaptive crossover and mutation operators, together with a simulated annealing operation to reduce the probability of converging toward local optima when using the original SPEA2.

Recently, an interesting method to maintain the diversity of SPEA2 for many-objective optimization was presented in [23]. In that study, for fitness assignment, a shift-based density estimation (SDE) technique takes into account both the distribution and convergence information of solutions, thus nondominated solutions with poor convergence are penalised during the evolution process. It has also been reported that SPEA2 with SDE (SPEA2+SDE) can offer significantly better performance than the original SPEA2 algorithm.

III. Proposed FSPEA Algorithm

A. Description of the Algorithm

The basic framework of FSPEA remains the same as that of SPEA2, except that a new density estimation method with the aid of predefined preference directions for fitness assignment and environmental selection is introduced. Algorithm 1 presents the general procedure of FSPEA, and the detailed information regarding the proposed method will be interpreted step by step as follows.

To begin with, an initial population of N candidate solutions is created by uniformly sampling from the decision variable space, followed by the generation of a set of preference directions. The preference direction set is called preference information, guiding the search along desired directions. To make the resulting approximated POF well-diversified, which

Algorithm 1 Framework of FSPEA

```
1:  Input: $N$ (population size)
2:  Output: approximated Pareto-optimal front
3:  Create an initial parent population $P$;
4:  Generate a set of preference directions $W$;
5:  while stopping criterion not met do
6:      Apply genetic operators on $P$ to generate offspring population $\overline{P}$;
7:      $Q = P \cup \overline{P}$;
8:      Associate each member of $Q$ with a preference direction;
9:      Calculate fitness values of members in $Q$;
10:     Perform environmental selection on $Q$ to update $P$;
11: end while
```

is also a main goal of multiobjective optimization, one must ensure that the predefined preference directions are diverse enough. For this purpose, we employ the systematic approach proposed by Das and Dennis [12] to generate a set W of evenly-distributed preference directions. Considering p divisions along each objective coordinate, the approach generates $H = \binom{p+M-1}{M-1}$ preference directions on a unit simplex for M objectives, each satisfying $\sum_{j=1}^{M} w_j^i = 1$ where w_j^i is the j-th component of the preference direction w^i.

In the main-loop (lines 6 to 10 in Algorithm 1), binary tournaments are carried out on the parent population P to fill the mating pool, on which combination and mutation operators are applied to produce the offspring population Q. In principle, any genetic operator can fulfil this goal. In this paper, the simulated binary crossover (SBX) [13] and polynomial mutation [14], which were also used in SPEA2 [18] and SPEA2+SDE [23], are used as our recombination and mutation operators, respectively. After that, the parent population and offspring population are merged for further consideration.

Afterwards, each member i of the combined population Q is associated with a preference direction. The associated preference direction $\hat{i}$ is calculated by $\hat{i} = j$:argmin$_{w^j \in W}\langle F(x^i), w^j\rangle$, where $F(x^i)$ is the corresponding objective vector of solution x^i, and $\langle a, b\rangle$ denotes the acute angle between the vectors a and b. This means that i is associated with its closest preference direction.

As for fitness assignment, the only difference between FSPEA and SPEA2 lies in density estimation. First, similar to SPEA2, each member i is assigned a raw fitness value $R(i)$ according to Eq. (2). Then, the density information $D(i)$ of i is estimated by its proximity to the associated preference direction $w^{\hat{i}}$, in the following form:

$$D(i) = \frac{\theta_i}{\theta_i + \pi/2} \tag{5}$$

where $\theta_i = \langle F(x^i), w^{\hat{i}}\rangle$ and $\pi/2$ is added to ensure that $D(i)$ is in the range $[0, 1]$. The final fitness value of i consists of the raw fitness value and density information by Eq. (4).

For environmental selection, FSPEA uses general operators similar to SPEA2, but it differs in the truncation operation if there are too many nondominated solutions. When the size of nondominated solutions in Q exceeds the population size N, FSPEA reasonably picks N elitists from the nondominated set, with the aid of the predefined preference directions, to replace the parent population. Specifically, for nondominated members associated with each preference direction, if any, the one having the lowest fitness value is preserved. In case that a preference direction does not have any associated nondominated solution, this preference direction is abandoned for further discussion. This way, H preference directions take turns at selecting the most diverse solutions until the number of the selected solutions is equal to N, satisfying the diversity requirement during the evolution.

B. Computational Complexity

FSPEA consumes the large proportion of computational resources in its main-loop. Association of $2N$ members of Q with H preference directions (line 8 of Algorithm 1) requires $O(MNH)$ computations, where M is the number of objectives. Fitness assignment (line 9 of Algorithm 1) takes $O(MN^2)$ computations. In line 10 of Algorithm 1, the time complexity of environmental selection comes from either selecting a number of best dominated solutions to fill up the population (similar to SPEA2) or truncating too many nondominated solutions. In the first case, it requires a complexity of $O(N \log N)$ when sorting the list of fitness values of dominated solutions, while in the second case, the truncation procedure can be done in $O(H)$ operations. Other procedures spend smaller computational resources. Additionally, N depends on the setting of H, i.e., $N \approx H$. Thus, the average and worst-case time complexities are identical, bounded by $O(MN^2)$.

Compared with SPEA2 and SPEA2+SDE, both of which have a computational complexity of $O(MN^3)$ in the worst case, the proposed FSPEA requires much less computational consumption, leading to notably fast run-time performance for multiobjective optimization.

IV. Experimental Study

A. Test Problems

The problem set chosen to perform the experiment is the ZDT [24] test suite plus two DTLZ [4] instances. The ZDT instances are all bi-objective problems, but they provide a series of characteristics which may appear in real-world applications, including concavity, convexity, disconnectivity, bias, and multimodality. Two three-objective DTLZ instances are used to test the performance of algorithms on higher-dimensional problems.

B. Performance Metric

In our experiment, the inverted generational distance (IGD) [26] is adopted as the performance indicator. IGD can provide reliable information on both the diversity and convergence of obtained solutions. Let PF be a set of solutions uniformly sampled from the true POF, and PF^* be the approximated solutions in the objective space, the metric measures the gap between PF^* and PF, which is calculated as follows:

$$IGD(PF^*, PF) = \frac{\sum_{p \in PF} d(p, PF^*)}{|PF|} \tag{6}$$

where $d(p, PF^*)$ is the distance between the member p of PF and the nearest member of PF^*.

TABLE I. BEST, MEDIAN AND WORST IGD VALUES OBTAINED BY THREE ALGORITHMS FOR ZDT AND DTLZ PROBLEMS

Prob.	MOEA/D	SPEA2	FSPEA
ZDT1	3.8740E-03	**3.7850E-03**	3.8680E-03
	3.8755E-03	3.9120E-03	3.8985E-03
	3.9310E-03	4.0060E-03	3.9390E-03
ZDT2	3.8020E-03	**3.7850E-03**	3.8030E-03
	3.8030E-03	3.9070E-03	3.8145E-03
	3.8080E-03	3.9580E-03	4.1030E-03
ZDT3	1.0695E-02	**4.7100E-03**	6.2800E-03
	1.0741E-02	**4.8665E-03**	6.6005E-03
	1.1457E-02	**5.0240E-03**	6.7870E-03
ZDT4	**3.8970E-03**	1.3566E-01	1.2536E-01
	4.4500E-03	3.8886E-01	3.8908E-01
	6.0630E-03	9.5859E-01	4.2744E-01
ZDT6	2.3770E-03	2.5460E-03	**2.3760E-03**
	2.6285E-03	2.6620E-03	**2.5920E-03**
	3.5050E-03	**2.8820E-03**	3.4340E-03
DTLZ1	1.7010E-02	9.7340E-03	**3.7800E-03**
	1.7080E-02	1.0375E-02	**1.0188E-02**
	1.7138E-02	1.1474E-02	**1.1286E-02**
DTLZ2	4.0714E-02	2.8683E-02	**4.9030E-03**
	4.1156E-02	3.0077E-02	**6.4965E-03**
	4.1373E-02	3.1819E-02	**7.9530E-03**

C. *Experimental Settings*

In the experiment, the original SPEA2 [18] and MOEA/D [25] are tested for comparison. The widely-used Chebyshev method is chosen as the decomposition function for MOEA/D. The population size N was set to 100 for bi-objective problems and 300 for three-objective problems, respectively. This setting exactly matches the condition involved in the generation of preference directions, i.e., $N = H = \binom{p+M-1}{M-1}$. For SPEA2, MOEA/D and FSPEA, the crossover probability is $p_c = 1.0$ and its distribution index is $\eta_c = 20$. The mutation probability is $p_m = 1/n$ and its distribution $\eta_m = 20$, where n is the number of decision variables. The archive size used in SPEA2 equalled the population size, and the neighbourhood size used in MOEA/D was set to 20. Other parameters were set according to their description in the literature.

All the compared algorithms use the same stopping condition, that is, they terminate after a pre-specified number of generations. The maximum number of generations was 500 for all the test instances. For statistical verification, each algorithm was independently executed 31 runs and optimization results were recorded.

D. *Experimental Results and Analysis*

Table I presents the experimental results of three algorithms on the test problems. The best values for each instance are marked in boldface. As ZDT1 and ZDT2 are very simple continuous convex and concave test problems, respectively, the IGD results obtained by the three algorithms are very similar, implying all these algorithms can easily approximate the POF with a good spread. For the discontinuous ZDT3 problem, SPEA2 provides the best results in terms of the IGD metric, followed by FSPEA, whose IGD values are slightly worse than those of SPEA2. MOEA/D performs the worst on this problem. ZDT4 is a multimodal problem with a large number of local POFs, so it is not easy for algorithms to converge toward the global POF. On this problem, MOEA/D converges well, while both SPEA2 and FSPEA seem to have difficulties in finding the true POF, as indicated by the large IGD values. The ZDT6 problem introduces several potential difficulties for algorithms. It has a non-convex POF shape and features a bias in the solution distribution. All the tested algorithms show good performance on this problem, and FSPEA is more likely to give slightly better IGD values compared with the other two algorithms.

The POF of DTLZ1 is a linear hyperplane, satisfying $\sum_{i=1}^{M} f_i = 0.5$, but the presence of many local optima in the search space causes difficulties for algorithms to converge to the global POF. It can be clearly seen from Table I that FSPEA produces much better IGD values than MOEA/D and SPEA2, while the performance of MOEA/D is not as good as SPEA2. On DTLZ2 whose POF is a hypersphere surface, FSPEA again outperforms the other two algorithms by a clear margin. This means, with the help of preference directions, FSPEA is very promising for solving higher-dimensional problems.

For a visual understanding of these algorithms' performance, we plot their approximations in terms of the best IGD metric over 31 runs. The approximations of ZDT1 to ZDT4 are displayed in Fig. 1 and the others in Fig. 2. We can observe that, MOEA/D, SPEA2, and FSPEA perform similarly on ZDT1, ZDT2 and ZDT6. Both SPEA2 and FSPEA offer a better distribution than MOEA/D on ZDT3, where the approximation obtained by MOEA/D is overcrowded toward the x-axes. The ZDT4 approximations clearly show that SPEA2 and FSPEA do not converge to the global POF as what MOEA/D does. Nevertheless, FSPEA converges much better than SPEA2 in this case. For the two three-objective problems, both SPEA2 and FSPEA provide well-distributed approximations whereas MOEA/D does not. If we take a close look at the approximations obtained by SPEA2 and FSPEA on DTLZ1 and DTLZ2, the distribution of FSPEA follows a regular pattern and seems to be more uniform. In contrast, there are outlier points (DTLZ1) and some sparse subregions (DTLZ2) in the approximation of SPEA2.

From the aspects of the IGD metric and graphical visualization, FSPEA keeps SPEA-related advantages, e.g., discriminating solutions by Pareto-dominance relations, and provides uniformly-distributed solutions while pursuing lower computational consumption than the original SPEA2 algorithm. It shows competitive performance in comparison with the well-known MOEA/D algorithm for the bi-objective cases and even outperforms it for the three-objective instances considered in this paper. Although both FSPEA and MOEA/D use a similar idea to direct/guide the search, the results are different, especially on the three-objective problems. In MOEA/D, even if a set of uniformly-distributed weight vectors cannot result in an even spread of POF points since decomposition approaches control the search. However, with the aid of evenly-distributed preference directions, FSPEA can guide the search along these preference directions, finally generating an approximation as uniform as the given preference direction set. Thus, FSPEA provides a better distribution than MOEA/D on the tested three-objective problems.

V. CONCLUSIONS AND FUTURE WORK

This paper has presented a fast version of SPEA2, denoted FSPEA, which employs predefined preference information for individuals' density estimation and environmental selection. In FSPEA, a number of preferred search directions are specified

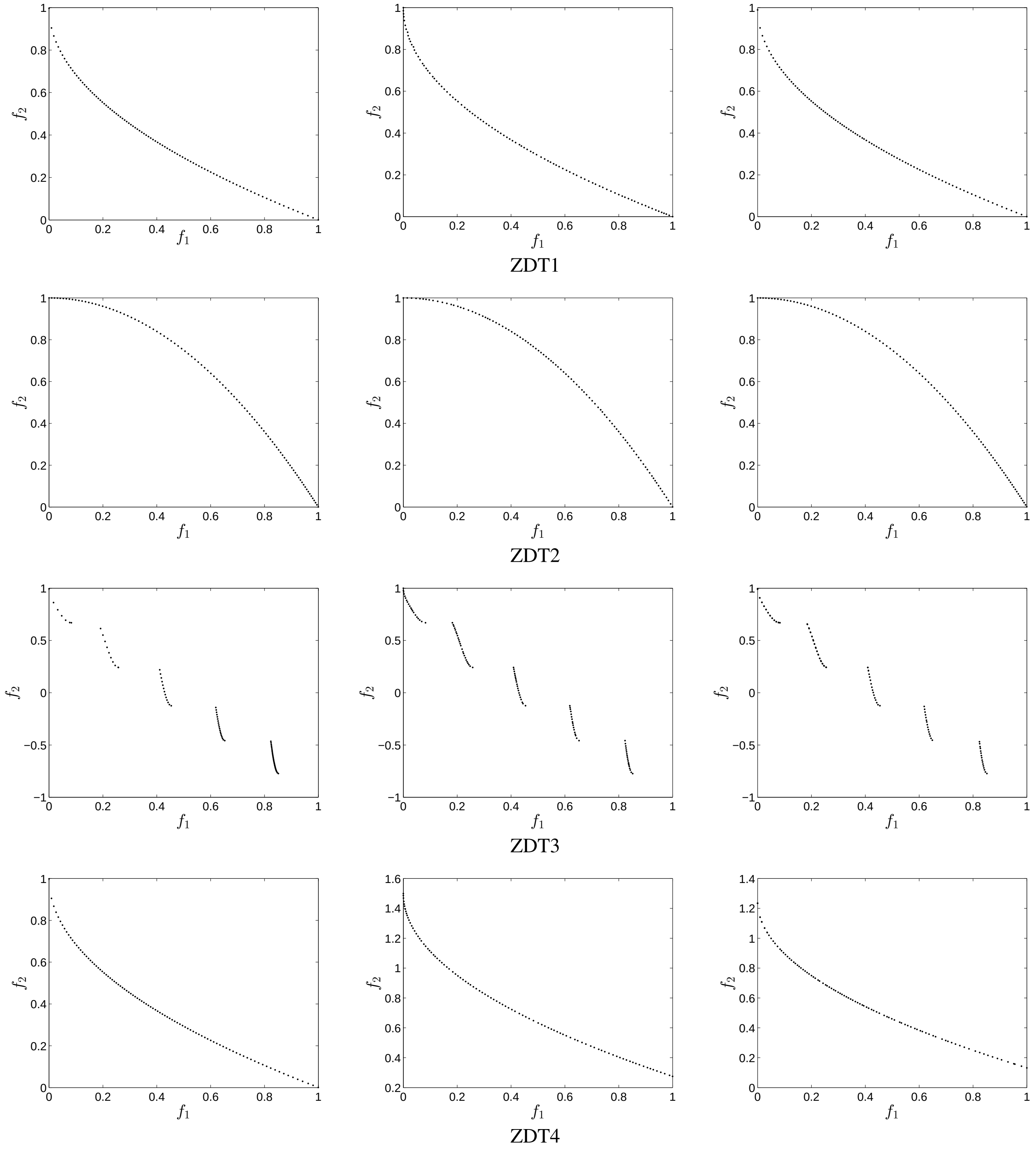
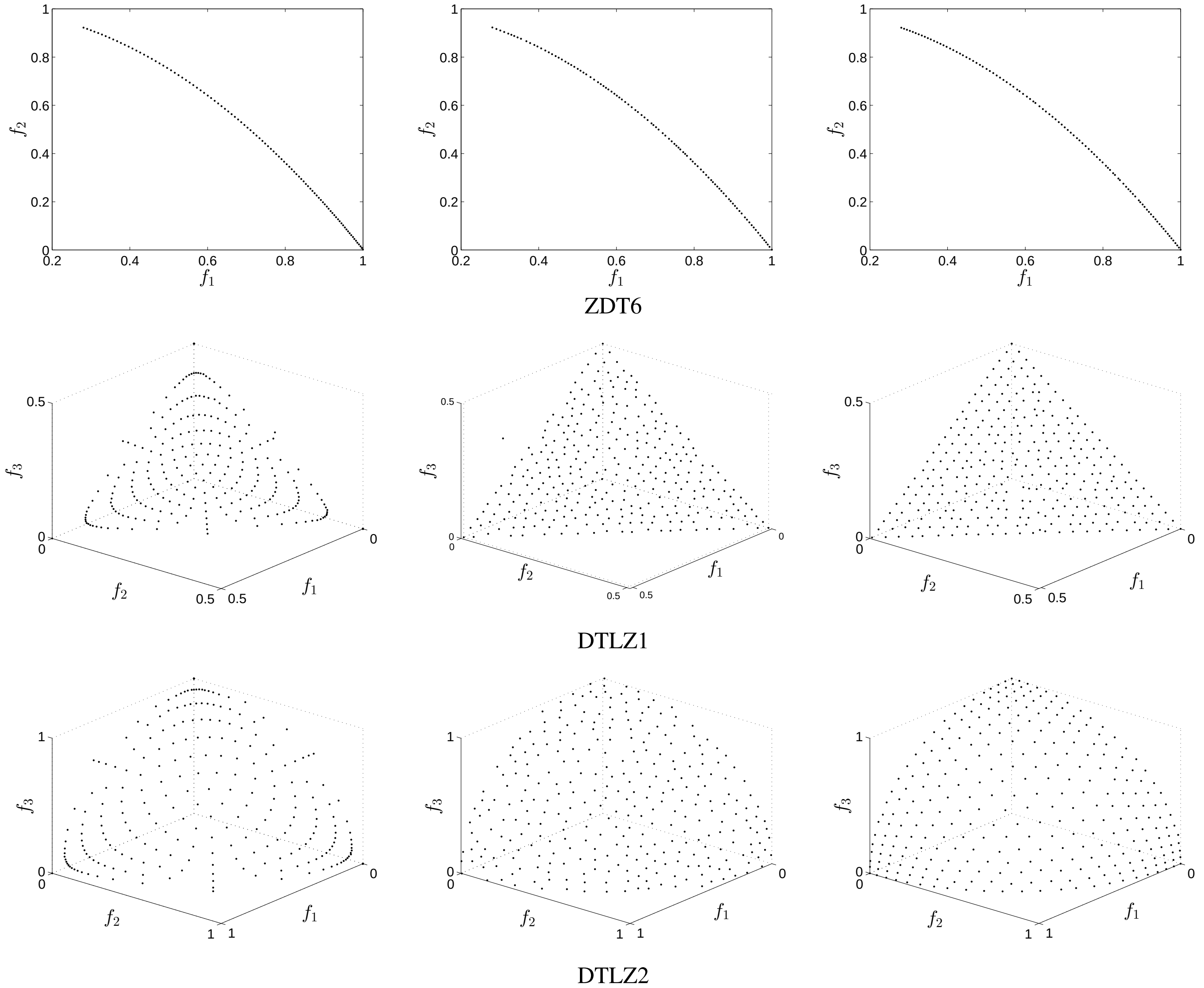


Fig. 1. Approximated POFs for ZDT1-ZDT4. The left column is with MOEA/D, the middle with SPEA2 and the right with FSPEA.



ZDT6

DTLZ1

DTLZ2

Fig. 2. Approximated POFs for ZDT6, DTLZ1 and DTLZ2. The left column is with MOEA/D, the middle with SPEA2 and the right with FSPEA.

to guide the search. Specifically, this preference information is used to estimate individuals' density, based on which FSPEA can easily discriminate nondominated individuals and preserve the most promising individuals in a new parent population for the next generation. This way, the preference based density estimation method can significantly alleviate the huge amount of computational effort required in the original SPEA2, leading to a fast optimization manner in terms of the running time for solving multiobjective problems. Experimental results have shown that FSPEA can achieve very competitive performance in comparison with SPEA2 and MOEA/D on a number of test problems considered in this paper.

Encouraged by the promising performance provided by FSPEA on multiojective optimization, in future research, we will extend the FSPEA framework to handling many-objective problems that have at least four conflicting objectives. Besides, it is also interesting to investigate the performance of FSPEA for dynamic multiobjective problems.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of UK under Grant EP/K001310/1.

References

[1] S. B. D. V. P. S. Anauth, R. T. F. Ah King, "Comparative application of multiobjective evolutionary algorithms to the voltage and reactive power optimization problem in power systems," in *Proc. 8th Conf. Simulated Evolution and Learning*, LNCS, vol. 6457, 2010, pp. 424–434.

[2] L. Bradstreet, L. Barone, L. While, S. Huband, and P. Hingston, "Use of the WFG toolkit and PISA for comparison of MOEAs" in *Proc. 2007 IEEE Symp. Comput. Intell. Multicriteria Decision Making (MCDM 2007)*, 2007, pp. 382–389.

[3] C. A. C Coello, G. B. Lamont, and D. A. V. Van, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York, NY, USA: Springer, 2007.

[4] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scable test problems for evolutionary multi-objective optimization," in *Evolutionary Multi-objective Optimization*. London: Springer, 2005, pp. 105–145.

[5] K. Miettnen, *Nonlinear Multiobjective Optimization*. New York, NY, USA: Springer, vol. 12, 1999.

[6] C. M. Fonseca and P. J. Fleming, “Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization,” in *Proc. 5th Int. Conf. Genetic Algorithms*, 1993, pp. 456–423.

[7] J. Horn, N. Nafpliotis, and D. E. Goldberg, “A nched Pareto genetic algorithm for multiobjective optimization,” in *Proc. 1st IEEE Conf. Evol. Comput.*, vol. 1, 1994, pp. 82–87.

[8] N. Srinivas and K. Deb, “Multiobjective optimization using nondominated sorting in genetic algorithms, *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1994.

[9] E. Zitzler and Thiele, “Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach,” *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, 1999.

[10] J. D. Knowles and D. W. Corne, “Approximating the nondominated front using the Pareto archived evolution strategy,” *Evol. Comput.*, vol. 8, pp. 149–172, 1999.

[11] D. W. Corne and J. D. Knowles, “The Pareto-envolope based selection algorithm for multiobjective optimization,” in *Proc. 6th Int. Conf. Parallel Problem Solving from Nature (PPSN VI)*, LNCS, vol. 1917, 2000, pp. 839–848.

[12] I. Das and J. Dennis, “Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems,” *SIAM J. Optimiz.*, vol. 8, no. 3, pp. 631–657, 1998.

[13] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester: John Wiley and Sons, 2001.

[14] K. Deb and M. Goyal, “A combined genetic adaptive search (GeneAS) for engineering design,” *Comput. Sci. and Inform.*, vol. 26, pp. 30–45, 1996.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[16] J. M. Lanza-Gutierrez, J. A. Gomez-Pulido, M. A. Vega-Rodriguez, “A trajectory-based heuristic to solve a three-objective optimization problem for wireless sensor network deployment,” in *EvoApplications 2014: Appl. of Evolut. Comput.*, LNCS, vol. 8602, 2014, pp. 27–38.

[17] J. Yu, M. Kirley and R. Buyya, “Multi-objective planning for workflow execution on grids,” in Proc. 8th IEEE/ACM Int. Conf. Grid Computing, 2007, pp. 10–17.

[18] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization,” in *Proc. Evol. Methods Des. Optimiz. Contr. Applicat. Indust. Problems*, EUROGEN, vol. 3242, no. 103, 2002, pp. 95-100.

[19] M. Kim, T. Hiroyasu, M. Miki and S. Watanabe, “SPEA2+: Improving the performance of the strength Pareto evolutionary algorithm 2,” in *Proc. 8th Int. Conf. Parallel Problem Solving from Nature (PPSN VIII)*, LNCS, vol. 3242, 2004, pp. 742–751.

[20] T. G. Tan, H. K. Lau and J. Teo, “Cooperative coevolution for Pareto multiobjective optimization: An empirical study using SPEA2,” in *Proc. 2007 IEEE Region 10 Conf.*, 2007, pp. 1–4.

[21] Z. Li, Y. Li and X. Duan, “Improved strength Pareto evolutionary algorithm with local search strategies for optimal reactive power flow,” *Inform. Tech. Journal*, vol. 9, no. 4, pp. 749–757.

[22] W. Sheng, Y. Liu, X. Meng and T. Zhang, “An improved strength Pareto evolutionary algorithm 2 with application to the optimization of distributed generations,” *Comput. and Math. with Appl.*, vol. 64, pp. 944–955, 2012.

[23] M. Li, S. Yang, X. Liu, “Shift-based density estimation for Pareto-based algorithms in many-objective optimization,” *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 348–365, Jun. 2014.

[24] E. Zitzler, K. Deb, and L. Thiele, “Comparison of multiobjective evolutionary algorithms: empirical results,” *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.

[25] Q. Zhang and H. Li, “MOEA/D: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, 2007.

[26] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, 2003.