Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

An Spanning tree Based Method for Pruning Non-Dominated Solutions in Multi-Objective Optimization Problems

Miqing Li, Jinhua Zheng, Ke Li, Jun Wu and Guixia Xiao
Institute of Information Engineering,
Xiangtan University, 411105, Hunan, China
limit1008@126.com

***Abstract*—Diversity maintenance of solutions is a crucial part in multi-objective optimization. However, most of existing studies show a good distribution with a large computational load or a comparative bad distribution quickly. In this paper, a method for pruning a set of non-dominated solutions using a Spanning Tree is proposed. This approach defines a density estimation metric – Spanning Tree Crowding Distance (*STCD*). Moreover, information of degree of solution combined with *STCD* is employed to truncate the population. From an extensive comparative study with three other methods on a number of 2, 3 and 4 objective test problems, the proposed method indicates a good balance among uniformity, spread and execution time.**

***Keywords* — Multi-objective optimization, Evolutionary algorithms, Pruning, Density estimation**

I. INTRODUCTION

The multi-objective optimization problems (MOPs) are generally difficult to handle and have received considerable attention in operations research. Because of the conflicting nature of their objectives, MOP does not normally have a single solution, which means that usually one solution can not be said to better than another. Thus, these MOPs search for an optimal set called the Pareto front. In 1989, Goldberg [1] suggested the use of a genetic algorithm to solve MOPs and since then other researchers have been developing new methods, such as multi-objective genetic algorithm (MOGA) [2], niched Pareto genetic algorithm (NPGA) [3], non-dominated sorted genetic algorithm (NSGA [4] and NSGA-II [5]), strength Pareto evolutionary algorithm (SPEA [6] and SPEA2 [7]), Pareto archived evolution strategy (PAES [8]), Pareto envelope-based selection algorithm (PESA [19] and PESA-II [17]) and micro-genetic algorithm (Micro-GA [9]) etc..

Achieving a well-diverse Pareto front can be a time consuming computational problem, associated with multi-objective evolutionary algorithms (MOEAs). A good background review about the pruning of non-dominated solutions in MOEAs can be found in [10]. An idea is to prune a non-dominated set to have desired number of solutions in such a way that remaining solutions would be as good diversity as possible. However, the computational complexity is directly related with the level of pruning. NSGA-II [5] uses a Crowding Distance approach which has a computational complexity of O ($MN\log N$), where N is the population size and M is the objective number. SPEA [6] which has a computational complexity of O (MN^3) has proved to be little better than NSGA-II for three objective problem. Recently, several pruning methods have been studied, Kukkonen and Deb [11] proposed an improved pruning method based on Crowding Distance to truncate population, also has a computational complexity of O ($MN\log N$). It can provide a good diversity in the case of two objectives, but when the number of objectives is more than two, the obtained diversity declines drastically. The polar coordinate [12], a better method than Crowding Distance with regard to truncate non-dominated set, has a computational complexity of O (MN^2). However, it is only fit for some special problems which have a spherical Pareto front. On the other hand, some methods which achieve better results seem to have a higher computational complexity. For example, SPEA2 [7] based on finding the kth nearest neighbor of solutions may have a computational complexity of O ($MN^2\log N$). MST-MOEA [13] pruning non-dominated set by Minimum Spanning Tree could provide good spread and uniformity, however, the time complexity of the algorithm reaches O (N^3).

The aim of this study is to present a new pruning method and to compare the different diversity maintenance methods. The remaining of this paper is organized as follows: Section 2 describes the proposed method. Section 3 presents the algorithm settings, test functions and performance indices used for performance comparison and Section 4 shows the simulation results. Finally, some conclusions are drawn in Section 5.

II. THE PROPOSED METHOD

As well known, it is a very useful feature in elitism MOEAs to choose a necessary number of solutions pass to the following generation. Similarly to the NSGA-II, the proposed method is based on non-dominated fronts [5]. However, when the last allowed front is being considered, and there are more solutions in the last front than the remaining slots in the population, a maintenance method using Spanning Tree is used to pruning the non-dominated solutions from the last front. Before discussing the diversity maintenance, we define a density estimation metric.

Definition 1 (*STCD*) *Let non-dominated set* ***Q*** *be all vertex of graph* ***G****, and* ***T*** *be a spanning tree of* ***G****. For any individual (vertex)* ***i*** *of* ***G****, let edges of* ***i*** *are* $\boldsymbol{L_1}$*,* $\boldsymbol{L_2}$*, …,* $\boldsymbol{L_r}$ *respectively, the STCD of* ***i*** *is:*

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

$$T(i) = \sum_{j=1}^{r} L_j / r \qquad (1)$$

Where *r* is the number of edge joining vertex *i*. To get an estimation of the density of solutions surrounding a solution in the population, we calculate the average length of all edges which connect this solution in Spanning Tree. Next, an approach using *STCD* has been proposed to maintain diversity.

Algorithm 1

Parameters: a non-dominated set **Q,** the size **N** of a desired pruned set

Step 1: For each individual in **Q**, calculate the Euclidean distance to other individuals. Then, generate a Minimum Spanning Tree (MST) **T** for all individuals in **Q** by Prim algorithm. Meanwhile calculate the *STCD* and the degree of individuals in **T**.

Step 2: Find the shortest edge $\mathbf{L_{i,j}}$ in **T,** where **i** and **j** are the endpoints of $\mathbf{L_{i,j}}$.

Step 3: If $\mathbf{d_i} = \mathbf{d_j}$, then go to Step 4; else if $\mathbf{d_i} > \mathbf{d_j}$, eliminate **i**, else eliminate **j**, go to Step5, where $\mathbf{d_i}$ is the degree of **i**.

Step 4: Compare the *STCD* of **i**, **j**, if T(i) > T(j), eliminate **j**, else eliminate **i**.

Step 5: |**Q**| = |**Q**| -1, if |**Q**| = N, stop.

Step 6: Assume **i** is the individual has been eliminated, $\mathbf{i_0}$, $\mathbf{i_1}$,…, $\mathbf{i_k}$ are the individuals have ever being connected to **i** in **T**. Thus, k trees were emerged after **i** was removed. Generate a Minimum Spanning Tree for $\mathbf{i_0}$, $\mathbf{i_1}$,…, $\mathbf{i_k}$. Obviously, it would be resumed a new tree **T'** for the whole remaining non-dominated solutions.

Step 7: **T** = **T'**, update the *STCD* and the degree of individuals in **T** and go to Step 2.

Figure 1 illustrates this process, assuming that **N** = 3. At first, calculate the distance between arbitrary two individuals and generate the MST for all individuals (showed in Figure 1(a)), then find the shortest edge $\mathbf{L_{AB}}$, considering $\mathbf{d_A = 1 < d_B = 2}$, **B** is eliminated. Since **A** and **C** are the individuals connected **B**, generate the MST for **A** and **C** (showed in Figure 1(b)). (Of course, when there are only two individuals considered, generating their MST is simply connecting them). And again, find the shortest edge $\mathbf{L_{CD}}$, and then eliminate **C** considering $\mathbf{d_D = 1 < d_C = 4}$. Generate the MST for **A**, **D**, **F** and **H** (showed in Figure 1(c)). And the next, find the shortest edge $\mathbf{L_{DF}}$ in 1(c), eliminate **D** for $\mathbf{d_F = d_D = 3}$, T(**D**) < T(**F**). Repeat until there are only 3(**N** = 3) individuals left in **Q**, and they are **A**, **E**, **G**.

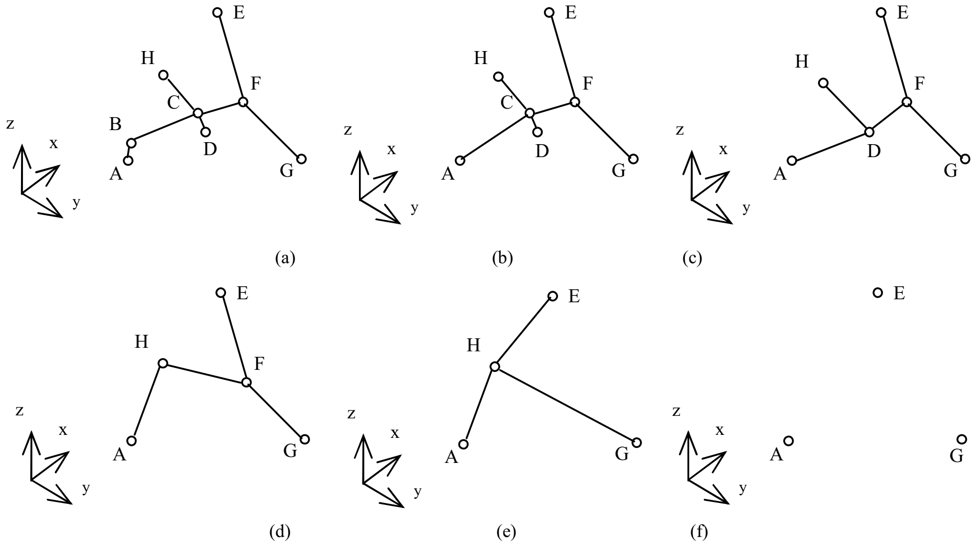


Figure 1(a)-(f). Illustration of the approximation set pruning method. Where |Q| = 8, N = 3, A, B, C, D, E, F, G, H are individuals in Q

Note that the Spanning Tree renewing in Step 6 may not be a MST. For instance, the graph shown in Figure 1(b) is not a MST for $\mathbf{L_{AC}} > \mathbf{L_{AD}}$. For all that, this operation can obtain a Spanning Tree that is quite similar with the MST of the remaining solutions; however, it can largely reduce the execution time compared with the operation of generating the MST.

Obtaining an extensively-distributed set of solutions is an important target in MOEAs. Some existed algorithms maintaining the range of solutions set by preserving the extreme solutions (in this paper, the non-dominated solution which has the maximum value for one or more objectives is called as extreme solution). Generally, these methods can provide good spread in the case of 2-objective problems.

However, when the number of objectives is more than two, the results of extent of solutions set may be dubious. In our method, the degree of node in Spanning Tree is introduced to maintain the spread of solutions set, considering the individuals located in the central part of population usually have higher degrees and the individuals at the border with lower ones. In the Step 3 of Algorithm 1, the endpoint of shortest edge in Spanning Tree which has higher degree will be eliminated firstly, which maintains the range of solutions set to a great extent.

The complexity of this procedure is analyzed in this section. The first operation of the algorithm in Step 1 is governed by calculating the distance between arbitrary two solutions in population and generating the MST. The time complexity of calculating the distance between arbitrary two solutions is $O(MN^2)$. And the time complexity of generating the MST, evaluating the degree of solutions and calculating the *STCD* by Prim algorithm is $O(N^2)$. The loop in Step 2-7 is executed at most N times. Finding the shortest edge in Step 2 can be done in $O(N)$. The compare operations in Step 3-4 and the renewing of Spanning Tree in Step 6 take constant computation time, and so does the updating operation of *STCD* and degrees in Step 7. Therefore, the computation time for loop in 2-7 is bounded by $O(N^2)$. The overall complexity for the whole algorithm is $O(MN^2)$.

III. Test problems and Performance Indices

In order to validate the proposed method and compare its performance with other advanced maintaining methods, eleven test functions were tested using three existed algorithms — SPEA2 [7], NSGA-II [5], and PESA-II [17].

A. Genetic Settings

All MOEAs are given real-valued decision variables. A crossover probability of $p_c = 0.9$ and a mutation probability $p_m = 1/K$ (where K is the number of decision variables) are used. The operators for crossover and mutation are simulated binary crossover (SBX) and polynomial mutation, with distribution indexes of $\eta_c = 11$, and $\eta_m = 17$ [15], respectively. For PESA-II, we have set 32 × 32 hyper-boxed for 2-objective problems, 8 × 8 × 8 for 3-objective problems, and 6 × 6 × 6 × 6 for 4-objective problems.

B. Functions Used

The test bed is formed by a total of ten functions. They are ZDT1, ZDT2, ZDT3, ZDT4, ZDT6 [14], DTLZ1, DTLZ2, DTLZ3, DTLZ5, DTLZ7 [18]. The number of decision variables N: $N_{ZDT1} = N_{ZDT2} = N_{ZDT3} = 30$, $N_{ZDT4} = N_{ZDT6} = 10$, $N_{DTLZ1} = 7$, $N_{DTLZ2} = N_{DTLZ3} = N_{DTLZ5} = 12$, and $N_{DTLZ7} = 22$. ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6 have 2 objectives, and DTLZ1, DTLZ2, DTLZ3, DTLZ5 and DTLZ7 have variable ones.

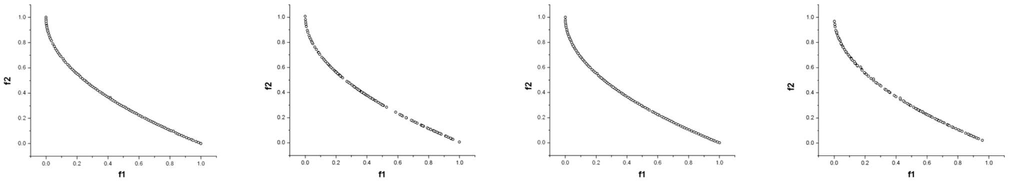
C. Performance Metrics

Usually, there are three goals that MOEAs can be identified and measured in performance [21]:

- The distance of the resulting solutions to Pareto front (PF) should be minimized.
- A uniform distribution of the solutions found is desirable.
- The extent of the solutions should be maximized.

In this paper, four metrics (*GD* [22], *SP* [20], *D* [14], and *Hypervolume* [16]) are used to assess the performance of the algorithms. *GD* assesses the algorithm on the first goal, *SP* appraises the algorithm on the second goal, and *D* scores the algorithm on the third goal. *Hypervolume* measures comprehensive performance of algorithms, containing convergence, uniformity and spread. Next, we introduce these metrics briefly.

The generational distance (*GD*) represents the average distance of the solutions in an approximation to PF. The spacing measures (*SP*) measures the standard deviation of distances from each solution to nearest solution in the obtained non-dominated front. A smaller value of *SP* is better and for an ideal distribution *SP* = 0. The maximum spread (*D*) measures the length of the diagonal of a minimal hyper-box, which encloses the obtained non-dominated set, and a larger value tells about a larger extent between extreme solutions. The *Hypervolume* calculates the volume of the objective space between the non-dominated solutions and a reference point, and a larger value is better. The reference point used with all ZDT problems is {2.0, 2.0}. The reference point used with the 3-objective DTLZ problems is {2.0, 2.0, 2.0} except the reference point of {1.0, 1.0, 1.0} is used with DTLZ1 and the reference point of {2.0, 2.0, 7.0} is used with DTLZ7. The reference point used with 4-objective DTLZ2 problem is {2.0, 2.0, 2.0, 2.0}. The hardware used is a PC with 1.7GHz Pentium 4 CPU with a Memory of 256 MB, and the operating system is Windows XP. The code is written in C++ and compiled using VC++ 6.0.



(a) Proposed method (b) NSGA-II (c) SPEA2 (d) PESA-II

Figure 2. The final solutions obtained by four methods on ZDT1

Table 1. Performance comparison of the four methods for two-objective test. Best metric values are shown bold

Problem	Method	*SP*	*D*	*GD*	*Hypervolume*	Time (s)
ZDT1	Proposed	**3.2223E-03** ± 3.5124E-04	**1.4151E+00** ± 3.3323E-04	1.9218E-04 ± 1.8418E-05	**3.6610E+00** ± 4.6312E-04	2.1718E+00 ± 5.8146E-02
	NSGA-II	7.5588E-03 ± 1.1264E-03	1.4134E+00 ± 8.4719E-04	1.8415E-04 ± 4.4811E-05	3.6598E+00 ± 5.8557E-04	**1.4017E+00** ± 3.1219E-02
	SPEA2	3.3599E-03 ± 3.1526E-04	1.4140E+00 ± 1.0859E-03	3.6246E-04 ± 3.2286E-05	3.6609E+00 ± 9.3635E-04	8.8484E+00 ± 1.5339E+00
	PESA-II	8.2116E-03 ± 1.1264E-03	1.3842E+00 ± 2.1365E-02	**1.7485E-04** ± 3.5109E-05	3.6424E-01 ± 3.1464E-03	3.8786E+00 ± 7.8469E-02
ZDT2	Proposed	3.3859E-03 ± 3.6483E-04	**1.4163E+00** ± 3.9197E-03	1.4650E-04 ± 3.2514E-05	**3.3289E+00** ± 1.2152E-04	2.1489E+00 ± 1.3949E-01
	NSGA-II	8.1965E-03 ± 1.2146E-03	1.4141E+00 ± 3.8041E-03	**1.2701E-04** ± 1.0129E-05	3.3267E+00 ± 3.3991E-04	**1.4644E+00** ± 1.1284E-01
	SPEA2	**3.3848E-03** ± 3.6575E-04	1.4123E+00 ± 7.0245E-04	3.4501E-04 ± 6.3560E-05	3.3271E+00 ± 3.5771E-04	8.4405E+00 ± 1.7342E-01
	PESA-II	8.5658E-03 ± 7.6748E-04	1.3917E+00 ± 7.6960E-03	1.5614E-04 ± 2.7188E-05	3.3039E+00 ± 4.4492E-04	3.2582E+00 ± 2.1271E-01
ZDT3	Proposed	**3.854E-03** ± 4.7131E-04	**1.9717E+00** ± 6.6882E-03	**5.4731E-04** ± 2.4366E-05	**4.8155E+00** ± 1.9348E-04	2.3532E+00 ± 7.5910E-02
	NSGA-II	7.9885E-03 ± 1.1776E-03	1.9305E+00 ± 1.1230E-02	5.5457E-04 ± 3.4053E-05	4.8145E+00 ± 2.0794E-04	**1.4609E+00** ± 3.9710E-02
	SPEA2	4.1186E-03 ± 4.1561E-04	1.9691E+00 ± 3.1101E-03	6.4167E-04 ± 2.3815E-05	4.8151E+00 ± 2.3406E-04	8.9965E+00 ± 1.2581E-01
	PESA-II	1.0926E-02 ± 1.7085E-03	1.9245E+00 ± 2.4549E-02	5.5321E-04 ± 4.8147E-05	4.7864E+00 ± 8.0124E-04	3.8841E+00 ± 3.2668E-02
ZDT4	Proposed	**3.4159E-03** ± 2.9703E-04	1.4512E+00 ± 4.3425E-02	2.5459E-04 ± 1.5966E-05	**3.6625E+00** ± 2.3954E-04	2.0314E+00 ± 1.5045E-01
	NSGA-II	7.9517E-03 ± 1.2036E-03	**1.4635E+00** ± 1.7407E-01	1.9362E-04 ± 4.8588E-05	3.6621E+00 ± 3.6922E-03	**1.0813E+00** ± 4.4824E-02
	SPEA2	4.1902E-03 ± 5.0794E-04	1.4443E+00 ± 1.1482E-01	2.6578E-04 ± 3.6683E-05	3.6614E+00 ± 4.7154E-04	9.1454E+00 ± 2.0044E-01
	PESA-II	1.0011E-02 ± 2.3287E-03	1.2295E+00 ± 8.9895E-02	**1.8171E-04** ± 3.9425E-05	3.6538E+00 ± 3.5476E-04	2.5426E+00 ± 1.8514E-01
ZDT6	Proposed	**3.0396E-03** ± 2.7072E-04	**1.0567E+00** ± 1.5927E-03	**7.6904E-04** ± 5.7785E-05	**2.9267E+00** ± 2.6614E-04	1.3264E+00 ± 9.6551E-02
	NSGA-II	5.1495E-03 ± 4.6986E-04	1.0461E+00 ± 6.9547E-03	7.8896E-04 ± 2.0253E-05	2.9167E+00 ± 2.4024E-03	**1.1469E+00** ± 9.0027E-02
	SPEA2	3.2875E-03 ± 3.0328E-04	1.0404E+00 ± 4.0961E-03	1.8267E-03 ± 1.8422E-04	2.8795E+00 ± 4.5450E-03	7.7724E+00 ± 7.6176E-02
	PESA-II	8.1591E-03 ± 1.0517E-03	1.0456E+00 ± 3.9879E-03	1.6974E-03 ± 1.0284E-04	2.8614E+00 ± 3.6510E-03	2.8130E+00 ± 4.8158E-02

IV. Simulation Results

To compare the algorithms performance several simulations were conducted involving 50 runs each. For the results analysis are considered average and standard deviation.

A. Two-Objective Problems

The algorithms were used to solve the two-objective test problems ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6, using population size 100 and 200 generations. A set of solutions for ZDT1 is shown in Figures 2. As it can be seen, the distribution of solutions obtained by the proposed method and SPEA2 are similar, but all have better diversity compared to NSGA-II and PESA-II.

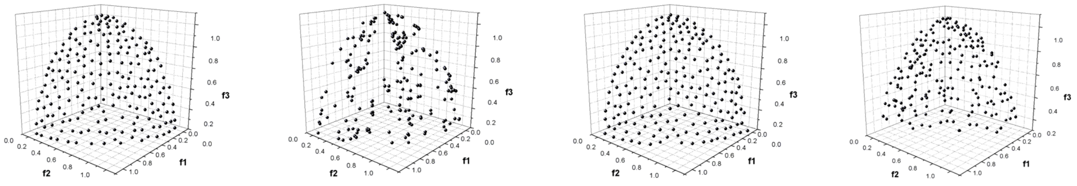
The numerical results on bi-objective optimization problems are reported in Table 1. As it can be observed for *SP*, the proposed method and SPEA2 provide better results compared to NSGA-II and PESA-II, furthermore the proposed method has slightly better than SPEA2 in all test problems except ZDT2. Difference in *D* is slight among four methods. For test problem ZDT4, NSGA-II obtains the best *D*, and for others, the proposed method has the best result. For *GD*, the proposed method, NSGA-II, and PESA2 have similar results, better than SPEA2. On test problems ZDT3 and ZDT6, the proposed method achieves best results. According to the *Hypervolume* value, the proposed method also demonstrates the best comprehensive performance in all 2-objective test problems. The total CPU time of the proposed method always takes the second place, much lower than SPEA2, and slightly lower than PESA-II.

B. Three-Objective Problems

The algorithms were used to solve the three-objective test problems DTLZ1, DTLZ2, DTLZ3, DTLZ5 and DTLZ7, using population size 200 and 500 generations. A set of solutions for DTLZ3 is shown in Figures 3. As it can be seen, the proposed method provides similar uniformity with SPEA2. The numerical results are shown in Table 2. According to *SP*, the proposed method provides similar results compared to SPEA2. In DTLZ3 and DTLZ5, the proposed method has the best results, and SPEA2 achieve the best values in remaining problems. As in the case of bi-objective problems, the difference in *D* is slight among four methods. The proposed method provides best value except DTLZ2. For *GD*, it is observed that the convergence of the proposed method is relatively better than that of the other three MOEAs in all test problems except DTLZ3 and DTLZ5. *Hypervolume* measure shown in column 6 of the table indicates that the proposed method and SPEA2 provide better comprehensive performance compared to NSGA-II and PESA-II. In DTLZ1 and DTLZ2, SPEA2 has the best results, and the proposed method performs the best in DTLZ3, DTLZ5 and DTLZ7. Considering the time of four methods, the proposed method is much faster than SPEA2, slightly faster than PESA-II for the majority of cases (except DTLZ5), and at worst about three times slower than the NSGA-II.

Table 2. Performance comparison of the four methods for three-objective test. Best metric values are shown bold

Problem	Method	*SP*	*D*	*GD*	*Hypervolume*	Time (s)
DTLZ1	Proposed	7.1364E-03 ± 4.0706E-04	**8.6872E-01** ± 1.7229E-03	**1.3767E-04** ± 6.0228E-05	9.7506E-01 ± 1.0424E-04	1.6756E+01 ± 9.5145E-02
	NSGA-II	3.1992E-02 ± 2.1719E-03	8.6132E-01 ± 1.9690E-03	1.4072E-04 ± 8.0928E-05	9.6713E-01 ± 1.3068E-03	**8.1047E+00** ± 8.7312E-02
	SPEA2	**7.1253E-03** ± 1.5848E-03	8.6652E-01 ± 1.0014E-02	1.4714E-04 ± 7.0782E-05	**9.7540E-01** ± 4.2907E-04	1.1409E+02 ± 1.0962E+00
	PESA-II	1.9875E-02 ± 2.7006E-03	8.5088E-01 ± 3.3827E-02	1.6971E-04 ± 7.0284E-05	9.5987E-01 ± 5.3518E-03	2.2078E+01 ± 7.9047E-01
DTLZ2	Proposed	1.7901E-02 ± 1.0748E-03	1.7575E+00 ± 1.0321E-02	**5.6384E-04** ± 4.9549E-05	7.4067E+00 ± 4.2657E-03	2.7609E+01 ± 1.3364E-01
	NSGA-II	4.1831E-02 ± 2.1634E-03	1.7455E+00 ± 6.7949E-03	7.3687E-04 ± 4.7045E-05	7.3516E+00 ± 1.7937E-02	**1.0600E+01** ± 4.0608E-02
	SPEA2	**1.6225E-02** ± 1.1705E-03	**1.7712E+00** ± 1.0455E-02	6.3572E-04 ± 9.5673E-05	**7.4091E+00** ± 5.5528E-03	1.5123E+02 ± 5.8966E-01
	PESA-II	4.1732E-02 ± 1.7522E-03	1.7550E+00 ± 2.2388E-02	5.7618E-04 ± 3.2708E-05	7.2456E+00 ± 2.6794E-02	4.0177E+01 ± 8.6841E-02
DTLZ3	Proposed	**1.5583E-02** ± 6.9112E-04	**1.7546E+00** ± 1.4512E-03	7.9354E-04 ± 1.3542E-04	**7.4166E+00** ± 3.7059E-03	6.1126E+00 ± 8.0098E-02
	NSGA-II	4.0491E-02 ± 2.9231E-03	1.7473E+00 ± 4.3894E-03	8.2892E-04 ± 3.5751E-04	7.3196E+00 ± 2.9307E-02	**5.4751E+00** ± 3.4995E-02
	SPEA2	1.5645E-02 ± 7.8496E-04	1.7514E+00 ± 2.6927E-02	6.9622E-04 ± 1.8681E-04	7.4071E+00 ± 6.2594E-03	6.0565E+01 ± 1.1957E+00
	PESA-II	3.5264E-02 ± 5.0803E-04	1.6139E+00 ± 5.5185E-02	**5.9541E-04** ± 1.2591E-04	7.2657E+00 ± 6.2547E-03	1.2457E+01 ± 6.3547E-01
DTLZ5	Proposed	**2.3036E-03** ± 2.0529E-04	**1.4161E+00** ± 1.8362E-03	6.3699E-05 ± 1.5865E-05	**6.1072E+00** ± 4.2487E-04	2.5416E+01 ± 1.8362E-03
	NSGA-II	5.3861E-03 ± 8.1431E-04	1.4146E+00 ± 8.4298E-04	4.7077E-05 ± 2.5158E-06	6.1062E+00 ± 3.8869E-04	**8.9062E+00** ± 1.9923E-02
	SPEA2	2.3502E-03 ± 1.9935E-04	1.4147E+00 ± 3.8444E-04	6.9791E-05 ± 1.0872E-05	6.1070E+00 ± 5.0740E-04	1.2008E+02 ± 1.5941E-01
	PESA-II	8.5424E-03 ± 1.1689E-03	1.4027E+00 ± 1.8890E-02	**3.2651E-05** ± 3.7266E-06	6.0854E+00 ± 9.2347E-04	1.5984E+01 ± 2.5794E-01
DTLZ7	Proposed	2.4295E-02 ± 2.6566E-03	**3.9751E+00** ± 2.6120E-01	**6.6635E-03** ± 3.1988E-04	**1.3512E+01** ± 3.0445E-02	2.2097E+01 ± 1.5777E-01
	NSGA-II	3.2491E-02 ± 1.6232E-02	3.5301E+00 ± 4.6207E-02	7.0801E-03 ± 4.4494E-04	1.2579E+01 ± 1.0634E-01	**8.8854E+00** ± 3.0742E-02
	SPEA2	**2.1404E-02** ± 2.1603E-03	3.7593E+00 ± 6.0109E-02	7.7735E-03 ± 9.1835E-04	1.3506E+01 ± 2.2885E-02	1.2093E+02 ± 1.2815E-01
	PESA-II	4.7411E-02 ± 4.5493E-03	3.3987E+00 ± 1.0835E-01	7.3251E-03 ± 6.5421E-04	1.2741E+01 ± 7.5461E-02	3.0925E+01 ± 7.9898E-02



(a) Proposed method (b) NSGA-II (c) SPEA2 (d) PESA-II

Figure 3. The final solutions obtained by four methods on DTLZ3

Table 3. Performance comparison of the four methods for four-objective test. Best metric values are shown bold

Problem	Method	*SP*	*D*	*GD*	*Hypervolume*	Time (s)
DTLZ2	Proposed	**3.3213E-02** ± 2.3650E-03	2.3397E+00 ± 4.0236E-02	1.9276E-03 ± 1.6531E-04	**1.3137E+01** ± 5.1092E-02	2.4816E+02 ± 6.0027E-01
	NSGA-II	7.6069E-02 ± 4.4408E-03	2.1926E+00 ± 5.7378E-02	**1.8468E-03** ± 1.2481E-04	1.3129E+01 ± 7.3289E-02	**6.4635E+01** ± 6.3518E-01
	SPEA2	3.4801E-02 ± 2.0229E-03	**2.3869E+00** ± 5.7339E-02	3.9369E-03 ± 2.1361E-04	1.2955E+01 ± 2.8990E-01	6.4842E+02 ± 3.4001E+00
	PESA-II	5.0725E-02 ± 2.6547E-03	1.8038E+00 ± 5.2401E-02	2.1592E-03 ± 1.7614E-04	1.3041E+01 ± 6.3284E-02	3.1242E+02 ± 7.5241E-01

C. Four-Objective Problem

The maintaining algorithms were used to solve a four-objective test problem DTLZ2, using population size 300 and 1000 generations. The numerical results are shown in Table 3. According to the *SP* and *D* the proposed method provides similar results compared to SPEA2. In which the proposed method has the best value on *SP*, while for *D*, SPEA2 performs the best. Considering *GD* of four methods, the proposed method is slightly worse than NSGA-II, however, better than PESA-II and SPEA2. According to the *Hypervolume*, the proposed method is the best, followed by NGSA-II, PESA-II. Interestingly, SPEA2 shown good performance in uniformity and spread has worse *Hypervolume* than other algorithms. The reason for this may be that SPEA2 has difficulty in converging for 4-objective problem. The total CPU time of the proposed method takes the second place, slower than NSGA-II, however faster than PESA-II and SPEA2.

V. Conclusions

Finding a solution set of good distribution in a small computational time is a crucial part for MOEA. In this paper a method of pruning non-dominated set replacing Crowding Distance in NSGA-II has been proposed. The method defines *STCD* to estimate the density of individual, and truncate population by *STCD* and degree of individual.

According to the experimental results, the proposed method also provides a similar uniformity result compared to SPEA2. Moreover, the proposed method spreads over the widest range in objective space for most test problem. And, the convergence study shows this pruning approach, instead of Crowding Distance, does not break the convergence of algorithm. The execution time needed for the proposed method is more than for NSGA-II, but slightly less than for PESA-II in most cases, and significantly less than for SPEA2. In a word, the proposed method emerges out a balance technique, producing good uniformity and extent with a small computational effort.

Notice that the proposed method is more demanding in terms of computational load compare to the method in NSGA-II. It is the future research topic that how to reduce the computational expense even thought it is currently reasonable.

References

[1] David E. Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Company, Reading, Massachusetts, 1989

[2] Carlos M. Fonseca and Peter J. Fleming. An overview of evolutionary algorithms in multiobjective optimization, *Evolutionary Computation*, 3(1): 1-16, 1995

[3] Jeffrey Horn, Nicholas Nafpliotis and David E. Goldberg. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. in *Proceeding of the first IEEE Congress on Evolutionary Computation*, (*CEC'1994*), 82-87, 1994

[4] N. Srinivas and Kalyanmoy Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation, 2(3): 221-248, 1994

[5] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA--II, IEEE Transactions on Evolutionary Computation, 6(2): 182-197, 2002

[6] Eckart Zitzler and Lothar Thiele. An evolutionary algorithm for Multiobjective optimization: The strength Pareto approach. Technical Report 43, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 1998

[7] Eckart Zitzler, Marco Laumanns and Lothar Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-Report 103, 2001

[8] Joshua Knowles and David Corne. Approximating the nondominated front using the Pareto archive evolutionary strategy. Evolutionary Computation, 8(2): 149-172, 2000.

[9] Carlos A. Coello Coello and Gregorio Toscano Pulido. A Micro-Genetic Algorithm for Multiobjective Optimization. First International Conference on Evolutionary Multi-Criterion Optimization (EMO'2001), 126-140, 2001

[10] Saku Kukkonen and Kalyanmoy Deb. A Fast and Effective Method for Pruning of Non-dominated Solutions in Many-Objective Problems, in Parallel Problem Solving from Nature - PPSN IX, 9th International Conference, 553-562, 2006

[11] Saku Kukkonen and Kalyanmoy Deb. Improved Pruning of Non-Dominated Solutions Based on Crowding Distance for Bi-Objective Optimization Problems, in *Proceedings of IEEE* Congress on Evolutionary Computation (CEC'2006), 3995-4002, 2006

[12] Da Kuang and Jinhua Zheng. Strategies Based on Polar Coordinates to Keep Diversity in Multi-Objective Genetic Algorithm, in Proceedings of IEEE Congress on Evolutionary Computation (CEC'2005), 1276-1281, 2005

[13] Miqing Li, Jinhua Zheng and Guixia Xiao. An Efficient Multi-objective Evolutionary Algorithm Based on Minimum Spanning Tree, in *Proceedings of IEEE Congress on Evolutionary Computation (CEC'2008), 617-624, 2008*

[14] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation, 8(2):173-195, 2000

[15] Kalyanmoy Deb. Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, Chichester, UK, 2001, ISBN 0-471-87339-X

[16] Eckart Zitzler and Lothar Thiele. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach, IEEE Transactions on Evolutionary Computation, 3(4): 257-271, 1999

[17] David W. Corne, Nick R. Jerram, Joshua D. Knowles and Martin J. Oates, PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization, in Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001), 283-290, 2001

[18] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns and Eckart Zitzler. Scalable Test Problems for Evolutionary Multiobjective Optimization, in Ajith Abraham, Lakhmi Jain and Robert Goldberg (editors), Evolutionary Multiobjective Optimization. Theoretical Advances and Applications, 105-145, 2005

[19] David W. Corne, Joshua D. Knowles and Martin J. Oates. The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization, In Proceedings of the Parallel Problem Solving from Nature VI Conference. Springer. 2000. 839-848

[20] Jason R. Schott. Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1995

[21] Miqing Li and Jinhua Zheng. Spread Assessment for Evolutionary Multi-Objective Optimization, Evolutionary Multi-Criterion Optimization, 5th International Conference (EMO'2009), Nantes, France 216-230, 2009

[22] Van Veldhuizen David A. and Gary B. Lamont. Evolutionary Computation and Convergence to a Pareto Front, In John R. Koza, editor, Late Breaking Papers at the Genetic Programming Conference, 1998. 221-228