A New Differential Evolution for Multiobjective Optimization by Uniform Design and Minimum Reduce Hypervolume

Siwei Jiang and Zhihua Cai*

School of Computer Science, China University of Geosciences
Wuhan 430074, China
amosonic@gmail.com, zhcai@cug.edu.cn

Abstract. Differential evolution is a powerful and robust method to solve the Multi-Objective Problems in MOEAs. To enhance the differential evolution for MOPs, we focus on two aspects: the population initialization and acceptance rule. In this paper, we present a new differential evolution called DEMO^{UD}_{DV}, it mainly include: (1) the first population is constructed by statistical method: *Uniform Design*, which can get more evenly distributed solutions than random design, (2) a new acceptance rule is firstly presented as *Minimum Reduce Hypervolume*. Acceptance rule is a metric to decide which solution should be cut off when the archive is full to the setting size. *Crowding Distance* is frequently used to estimate the length of cuboid enclosing the solution, while *Minimum Reduce Hypervolume* is used to estimate the volume of cuboid. The new algorithm designs a fitness function *Distance/Volume* that balance the *CD* and *MRV*, which maintains the spread and hypervolume along the Pareto-front. Experiment on different multi-Objective problems include ZDTx and DTLZx by jMetal 2.0, the results show that the new algorithm gets higher hypervolume, faster convergence, better distributed solutions and needs less numbers of fitness function evolutions than NSGA-II, SPEA2 and GDE3.

Keywords: Multi-Objective Optimization, Differential Evolution, Uniform Design, Crowding Distance, Minimum Reduce Hypervolume, Genetic Distance, Inverted Genetic Distance, Spread, Hypervolume.

1 Introduction

To solve the multiple conflicting problems, Multi-Objective Evolutionary Algorithms(MOEAs) are demonstrated as useful and powerful tools to deal with complex problems such as: discontinuous, non-convex, multi-modal, and non-differentiable[1]. The main challenge for MOEAs is to obtain Pareto-optimal solutions near to true Pareto- front in terms of convergence, diversity and limited evolution times.

* The Project was supported by the Research Foundation for Outstanding Young Teachers, China University of Geosciences(Wuhan)(No:CUGQNL0911).

F. Peper et al. (Eds.): IWNC 2009, PICT 2, pp. 199–208, 2010.

The Non-dominated Sorting Genetic Algorithm-II (NSGA-II) is proposed by Deb *et al*[2], which adopts a fast non-dominated sorting approach with $O(mN^2)$ computational complexity(where m is the number of objectives and N is the population size), NSGA-II combines the parent and child populations and selects the best N solutions with respect to the *Ranking* and *Crowding Distance* metrics.

The Strength Pareto Evolutionary Algorithm (SPEA2) is proposed by Zitzler *et al*[3], which incorporates a fine-grained fitness assignment strategy, a density estimation technique, and an enhanced archive truncation method. SPEA2 uses the truncation operator based on nearest *Neighbor Density Estimation* metric.

Differential evolution(DE) is a new evolutionary algorithm, which has some advantages: simple structure, ease to use, speed and robustness[4]. The developed version of Generalized Differential Evolution(GDE3) is proposed by Kukkonen *et al*[5], which is suited for global optimization with an arbitrary number of objectives and constraints. Similar to NSGA-II, GDE3 uses *Crowding Distance* metric as its acceptance rule, it can get better distributed solutions by its powerful search ability.

For long terms, the issue of population initialization has been ignored in evolution algorithms, researchers mainly focus on regeneration method and acceptance rule. *Orthogonal design* and *Uniform Design* belong to a sophisticated branch of statistics[8], they are more powerful than the *random design*.

Leung and Wang incorporated orthogonal design and quantum technique in genetic algorithm[9], which is more robust than the classical GAs for numerical optimization problems. Zeng *et al* adopts the orthogonal design method to solve the MOPs[10], it found a precise Pareto-optimal solutions for an engineer problem which has unknown before. Cai *el al* solves the MOPs by orthogonal population and ε-dominated[11], it gets better results in terms of convergence, diversity and time consume. Leung applies the uniform design to generate a good initial population and designs a new crossover operator for searching the Pareto-optimal solutions[12], which can find the Pareto-optimal solutions scattered uniformly over the Pareto frontier.

Interesting in the population initialization and acceptance rule, we propose a new differential evolution called DEMO_{DV}^{UD}, it construct by three aspects:

1. The first population is constructed by *Uniform Design*. It can get well scattered solutions in feasible searching space.
2. The regeneration method of differential evolution is adopted with DE/best/1/bin strategy. It can faster the convergence of the algorithm for MOPs.
3. Acceptance rule *Minimum Reduce Hypervolume* is firstly proposed, when the archive is full, it cuts off the solution which leads to the minimum decrease for the archive's hypervolume. Then a new fitness *Distance/Volume* is designed for DE, which maintains the spread and hypervolume along the Pareto-front.

Test the bi-objective of ZDT family and tri-objective of DTLZ family problems on jMetal 2.0[13], the results show that the new algorithm gets higher hypervolume, faster convergence, better distributed solutions and needs less numbers of fitness function evolutions than NSGA-II, SPEA2 and GDE3.

2 Uniform Design

Uniform Design belongs to a sophisticated branch of statistics, experimental methods construct the candidate solutions more evenly in feasible searching space and include more information than random design[8,9,10,11].

We define the uniform array as $U_R(C)$, where Q is the level, it's primer, R, C represent the row and column of uniform array, they must satisfy to:

$$\begin{cases} R = Q > n \\ C = n \end{cases} \tag{1}$$

Where n is the number of variables. When select a proper parameters of Q, σ form table 1, uniform array can be created by Equation 2

$$U_{i,j} = (i * \sigma^{j-1} \bmod Q) + 1 \tag{2}$$

Table 1. Values of the parameter σ for different number of factors and levels

number of levels of per factor(Q)	number of factors(n)	σ
5	2-4	2
7	2-6	3
11	2-10	7
13	2	5
	3	4
	4-12	6
17	2-16	10
19	2-3	8
	4-18	14
23	2,13-14,20-22	7
	8-12	15
	3-7,15-19	17
29	2	12
	3	9
	4-7	16
	8-12,16-24	8
	13-15	14
	25-28	18
31	2,5-12,20-30	12
	3-4,13-19	22

3 Minimum Reduce Hypervolume

For Multi-Objective Optimization Problems, MOEAs often finds the optimal set including many candidate solutions which no-dominate each other. When the optimal set is full to the setting size(usually $archiveSize = 100$), we need to set an acceptance rule to decide which one should be remained or cut off from archive. It directly influences the quality of finally optimal set in convergence and spread metric.

Some acceptance rules have been presented: NSGA-II adopts *Ranking* and *Crowding Distance* metric, SPEA2 uses nearest *Neighbor Density Estimation* metric, ϵ-MOEA proposes ϵ-*dominance* concept with less time consume[6,7].

Inspired by the *Crowding Distance* metric, in this paper, we firstly present a new acceptance rule called *Minimum Reduce Hypervolume*. Hypervolume is a quality indicator proposed by Zitzler *et al*, it is adopted in jMetal 2.0[13]. Hypervolume calculates the volume covered by members of a non-dominated set of solutions (the region enclosed into the discontinuous line respect the reference point W in the figure 1 is $ADBECW$ in dashed line).

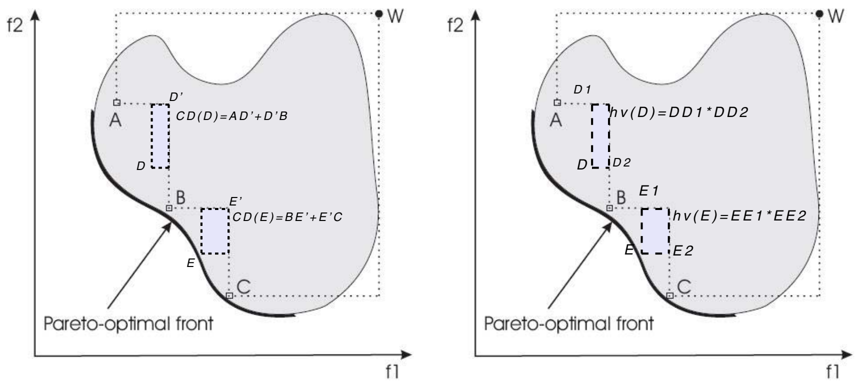

Fig. 1. The comparison of *Crowding Distance* and *Minimum Reduce Hypervolume* is described in the left and right figure

If two solutions D and E no-dominate each other and one should be cut off, NSGA-II chooses the solution D remained in archive if $CD(D) > CD(E)$, it maintains the spread along the Pareto-front.

$$\begin{cases} CD(D) = AD' + D'B \\ CD(E) = BE' + E'C \end{cases} \tag{3}$$

If one solution is deleted, it will lead to a hypervolume decrease, because the higher hypervolume means the better quality of optimal set, we will delete the solution which reduces the hypervolume minimum. *Minimum Reduce Hypervolume* chooses the solution E remained in archive if $hv(D) < hv(E)$(then the hypervolume is $ABECW$ rather than $ADBCW$), it maintains the hyervolume along the Pareto-front.

$$\begin{cases} hv(D) = DD1 * DD2 \\ hv(E) = EE1 * EE2 \end{cases} \tag{4}$$

4 Differential Evolution for MOPs by Uniform Design and Minimum Reduce Hypervolume

In this paper, we design a new hybrid acceptance rule. *Crowding distance* maintains the spread and expand solutions to feasible searching place, and *Minimum*

Reduce Hypervolume maintains the hypevolume and forces solutions near to the Pareto-front, so we combine the two properties, a new fitness assignment for solution s is designed as follows(called *Distance/Volume* fitness):

$$\begin{cases} DV(s) = CD(s) + scale * hv(s) \\ scale = \dfrac{\sum_{i=1}^{n-1} CD(s_i)}{\sum_{i=1}^{n-1} hv(s_i)} \end{cases} \tag{5}$$

The factor $scale$ is designed to equal the influence of crowding distance and sub-hypervolume, but if it is too large, we set $scale = 1$ when $scale > 1000$.

To enhance the differential evolution for MOPs, we focus on two aspects: population initialization and acceptance rule. The new differential evolution based on *Uniform Design* and *Distance/Volume* fitness function is called DEMO_{DV}^{UD}. Population size and archive size is setting to 100, the first population is constructed by uniform design($Q = 31$) and random design($100 - Q = 69$); when archive is full, we discard the solution which has small fitness by equation 5.

Algorithm 1. Main procedure of the proposed DEMO_{DV}^{UD}

```
Construct first population with 31 uniform solutions(Q = 31) by equation 2
Add 69 random solutions to the first population
Add the population to archive
while eval < Max_eval do
  for i = 1 to NP do
    Random choose two solutions from archive
    select the solution with larger DV fitness as DE base solution best
    Produce child c with DE/best/1/bin scheme
    Evaluate the child c and eval++
    if the child c dominates the one of the parent population P_t^r then
      P_t^r = c
    else if c is non-dominated by all the parent population then
      Rand replace one parent with probability 0.5
    else
      Discard c
    end if
    Add child c to archive.
    if archiveSize > 100 then
      Remove the candidate solution with minimum DV(s) by equation 5
    end if
  end for
end while
```

5 Experiment Results

Experiment is based on jMetal 2.0[13], which is a Java-based framework aimed at facilitating the development of metaheuristics for solving MOPs, it provides large block reusing code and fair comparison for different MOEAs.

The test problems are choose from ZDTx, DTLZx problem family including five bi-objective problems: ZDT1, ZDT2, ZDT3, ZDT4, ZDT6 and six tri-objective problems: DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ5, DTLZ6. Each algorithm independent runs for 50 times and maximum evolution times is $25,000$.

The performance metrics can be classified into five categories:

Hypervolume. This quality indicator calculates the volume (in the objective space) covered by members of a nondominated set of solutions with a reference point. The Hypervolume (HV) is calculated by:

$$HV = volume(\bigcup_{i=1}^{|Q|} v_i) \tag{6}$$

Generational Distance. The metric is to measure how far the elements are in the set of non-dominated vectors found from those in the true Pareto-optimal set. It is defined as:

$$GD = \sqrt{\frac{\sum_{i=1}^{|n|} d_i^2}{n}} \tag{7}$$

Inverted Generational Distance. The metric is to measure how far the elements are in the true Pareto-optimal set from those in the set of non-dominated vectors found. It is defined as:

$$IGD = \sqrt{\frac{\sum_{i=1}^{|N|} d_i^2}{N}} \tag{8}$$

Spread. The Spread indicator is a diversity metric that measures the extent of spread achieved among the obtained solutions. This metric is defined as:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{n-1} |d_i - \overline{d}|}{d_f + d_l + (n-1)\overline{d}} \tag{9}$$

Evolution Counts. This metric is to measure how many numbers of fitness function evolutions($NFFEs$) is needed, when the hypervolume of non-dominated set is larger than the 98% of hypervolume of the true Pareto-front set(Success Rate means the percentage of get the $HV_{paretoSet} \geq 0.98 * HV_{truePF}$ in 50 independent run, NA means cannot get 100% success).

$$Evolution\ counts = \{NFFEs | HV_{paretoSet} \geq 0.98 * HV_{truePF}\} \tag{10}$$

The higher Hypervolume and lower GD, IGD, Spread, Evolution Counts mean the algorithm is better. The result are compared with mean and variance, the better result between GDE3 and DEMO$_{DV}^{UD}$ is bold font.

From table 2, in term of Hypervolume, the $w/t/l$(win/tie/lost) value between DEMO$_{DV}^{UD}$ and GDE3 is 10/0/1, the new algorithm is only worse in problem

Table 2. The hypervolume metric of mean and variance for NSGA-II, SPEA2, GDE3 and DEMO_{DV}^{UD} in 50 independent run

probs	NSGA-II		SPEA2		GDE3		DEMO_{DV}^{UD}	
	mean	std	mean	std	mean	std	mean	std
Hypervolume								
ZDT1	0.659343	0.000295	0.659885	0.000370	0.661925	0.000022	**0.662084**	0.000013
ZDT2	0.326077	0.000256	0.326136	0.001615	0.328631	0.000024	**0.328796**	0.000018
ZDT3	0.514758	0.000472	0.514010	0.000676	0.515936	0.000042	**0.516024**	0.000025
ZDT4	0.654138	0.005719	0.644951	0.020486	0.661999	0.000024	**0.662086**	0.000016
ZDT6	0.388331	0.001691	0.378258	0.002821	0.401343	0.000017	**0.401482**	0.000019
DTLZ1	0.647919	0.186030	0.749499	0.082703	0.757721	0.095956	**0.770371**	0.003150
DTLZ2	0.375092	0.005691	0.404556	0.001746	0.388390	0.003685	**0.388688**	0.003002
DTLZ3	0.000000	0.000000	0.000000	0.000000	0.322214	0.140052	**0.389571**	0.004504
DTLZ4	0.372841	0.005571	0.332877	0.115285	**0.384835**	0.003979	0.384668	0.004476
DTLZ5	0.092796	0.000184	0.093194	0.000139	0.094023	0.000032	**0.094092**	0.000028
DTLZ6	0.000000	0.000000	0.000000	0.000000	0.094901	0.000026	**0.094968**	0.000034

DTLZ4; the $w/t/l$ value between DEMO_{DV}^{UD} and NSGA-II is 11/0/0, NSGA-II gets zero hypervolme in DTLZ3 and DTLZ6; the $w/t/l$ value between DEMO_{DV}^{UD} and SPEA2 is 10/0/1, the new algorithm is only worse in problem DTLZ2, and SPEA2 gets zero hypervolme in DTLZ3 and DTLZ6.

From table 3, in term of Genetic Distance, the $w/t/l$(win/tie/lost) value between DEMO_{DV}^{UD} and GDE3 is 9/0/2, the new algorithm is only worse in problem ZDT6, DTLZ5; the $w/t/l$ value between DEMO_{DV}^{UD} and NSGA-II is 11/0/0; the $w/t/l$ value between DEMO_{DV}^{UD} and SPEA2 is 11/0/0.

From table 4, in term of Inverted Genetic Distance, the $w/t/l$(win/tie/lost) value between DEMO_{DV}^{UD} and GDE3 is 8/0/3, the new algorithm is only worse in problem ZDT2, DTLZ2, DTLZ4; the $w/t/l$ value between DEMO_{DV}^{UD} and NSGA-II is 10/0/1, the new algorithm is only worse in problem DTLZ4; the $w/t/l$ value between DEMO_{DV}^{UD} and SPEA2 is 10/0/1, the new algorithm is only worse in problem DTLZ2.

From table 5, in term of Spread, the $w/t/l$(win/tie/lost) value between DEMO_{DV}^{UD} and GDE3 is 9/0/2, the new algorithm is only worse in problem ZDT3, DTLZ1; the $w/t/l$ value between DEMO_{DV}^{UD} and NSGA-II is 11/0/0; the

Table 3. The Generation Distance metric of mean and variance for NSGA-II, SPEA2, GDE3 and DEMO_{DV}^{UD} in 50 independent run

probs	NSGA-II		SPEA2		GDE3		DEMO_{DV}^{UD}	
	mean	std	mean	std	mean	std	mean	std
Generation Distance								
ZDT1	2.199E-4	3.623E-5	2.241E-4	3.127E-5	9.930E-5	3.210E-5	**7.252E-5**	1.874E-5
ZDT2	1.806E-4	6.895E-5	1.773E-4	3.921E-5	4.607E-5	2.202E-6	**4.588E-5**	2.258E-6
ZDT3	2.116E-4	1.322E-5	2.299E-4	1.534E-5	1.738E-4	1.243E-5	**1.700E-4**	1.165E-5
ZDT4	4.888E-4	2.358E-4	9.792E-4	1.840E-3	9.099E-5	3.139E-5	**8.127E-5**	2.494E-5
ZDT6	1.033E-3	1.108E-4	1.799E-3	2.997E-4	**5.277E-4**	1.564E-5	5.293E-4	1.617E-5
DTLZ1	1.890E-1	5.973E-1	5.181E-1	1.021E0	2.004E-3	9.746E-3	**5.976E-4**	2.763E-5
DTLZ2	1.428E-3	3.207E-4	1.363E-3	3.058E-4	6.463E-4	2.579E-5	**6.414E-4**	2.177E-5
DTLZ3	1.073E0	7.782E-1	2.353E0	1.478E0	2.196E-2	7.415E-2	**1.076E-3**	4.797E-5
DTLZ4	5.098E-3	2.786E-4	4.495E-3	1.291E-3	4.829E-3	1.693E-4	**4.737E-3**	2.294E-4
DTLZ5	3.707E-4	6.887E-5	3.770E-4	4.705E-5	**2.520E-4**	2.311E-5	2.592E-4	2.051E-5
DTLZ6	1.759E-1	2.111E-2	1.626E-1	1.424E-2	5.681E-4	2.373E-5	**5.655E-4**	2.648E-5

Table 4. The Inverted Generation Distance metric of mean and variance for NSGA-II, SPEA2, GDE3 and DEMO$_{DV}^{UD}$ in 50 independent run

probs	NSGA-II		SPEA2		GDE3		DEMO$_{DV}^{UD}$	
	mean	std	mean	std	mean	std	mean	std
Inverted Generation Distance								
ZDT1	1.893E-4	8.636E-6	1.521E-4	4.140E-6	1.398E-4	1.144E-6	**1.351E-4**	1.100E-6
ZDT2	1.928E-4	1.012E-5	1.843E-4	2.047E-4	**1.457E-4**	1.733E-6	1.467E-4	2.583E-6
ZDT3	3.519E-4	6.766E-4	4.323E-4	9.520E-4	2.007E-4	3.445E-6	**1.924E-4**	5.210E-6
ZDT4	3.099E-4	5.045E-4	1.338E-3	1.346E-3	1.378E-4	1.088E-6	**1.354E-4**	1.314E-6
ZDT6	3.533E-4	5.251E-5	6.832E-4	9.836E-5	1.170E-4	4.315E-6	**1.088E-4**	3.969E-6
DTLZ1	1.409E-3	1.533E-3	6.567E-4	5.810E-4	6.345E-4	7.597E-4	**5.302E-4**	1.973E-5
DTLZ2	7.747E-4	3.662E-5	5.906E-4	1.104E-5	**6.802E-4**	2.263E-5	6.843E-4	2.227E-5
DTLZ3	9.306E-2	5.289E-2	7.400E-2	3.446E-2	4.244E-3	1.150E-2	**1.089E-3**	4.322E-5
DTLZ4	1.196E-3	1.134E-4	2.317E-3	2.592E-3	**1.211E-3**	9.7101E-5	1.225E-3	1.132E-4
DTLZ5	2.006E-5	1.063E-6	1.581E-5	4.951E-7	1.445E-5	2.630E-7	**1.387E-5**	3.131E-7
DTLZ6	7.254E-3	7.102E-4	6.726E-3	4.950E-4	3.518E-5	9.993E-7	**3.390E-5**	9.281E-7

Table 5. The Spread metric of mean and variance for NSGA-II, SPEA2, GDE3 and DEMO$_{DV}^{UD}$ in 50 independent run

probs	NSGA-II		SPEA2		GDE3		DEMO$_{DV}^{UD}$	
	mean	std	mean	std	mean	std	mean	std
Spread								
ZDT1	0.380309	0.028535	0.150063	0.014035	0.146912	0.012623	**0.123448**	0.014250
ZDT2	0.379900	0.028633	0.157648	0.027377	0.135998	0.013377	**0.121396**	0.012272
ZDT3	0.747304	0.014352	0.710772	0.005718	**0.710740**	0.004612	0.716190	0.011558
ZDT4	0.388983	0.040338	0.322715	0.148944	0.139899	0.013194	**0.115616**	0.013786
ZDT6	0.358147	0.029767	0.233547	0.036104	0.127141	0.010248	**0.085469**	0.013509
DTLZ1	0.925933	0.236171	0.904105	0.316227	**0.733342**	0.043591	0.745113	0.039776
DTLZ2	0.705338	0.053090	0.530350	0.031301	0.642637	0.027290	**0.635581**	0.045237
DTLZ3	1.052288	0.128959	1.258939	0.139970	0.639076	0.040248	**0.632920**	0.045891
DTLZ4	0.672005	0.045341	0.488522	0.182093	0.644641	0.037459	**0.643950**	0.132165
DTLZ5	0.457166	0.050954	0.238807	0.037331	0.161843	0.012136	**0.110917**	0.015109
DTLZ6	0.805454	0.051306	0.578990	0.041086	0.164002	0.012606	**0.111253**	0.015057

$w/t/l$ value between DEMO$_{DV}^{UD}$ and SPEA2 is 8/0/3, the new algorithm is only worse in problem ZDT3,DTLZ2, DTLZ4.

From table 6, in term of Evolution Counts, the $w/t/l$(win/tie/lost) value between DEMO$_{DV}^{UD}$ and GDE3 is 7/4/0, in problem DTLZ1, DTLZ2, DTLZ3 and DTLZ4, both the new algorithm and GDE3 gets 0% Success Rate; the $w/t/l$ value between DEMO$_{DV}^{UD}$ and NSGA-II is 7/4/0; the $w/t/l$ value between DEMO$_{DV}^{UD}$ and SPEA2 is 7/4/0. NSGA-II and SPEA2 can not get 100% success Rate in problem ZDT2, they get 0% success Rate in problem ZDT6 and all tri-objective problems.

The comparison for four algorithms in five categories : Hypervolume, Generation Distance, Inverted Genetic Distance, Spread and Evolution Count/Success Rate, it shows that the new algorithm is more efficient to solve the MOPs. Now, we summarize the highlight as follows:

1. population initialization is constructed by *Uniform Design*, which can get more evenly scatter solutions in feasible space than *Random Design*, it will provide good guide information for next offspring and speed the convergence.

Table 6. The Evolution times metric of mean and variance for NSGA-II, SPEA2, GDE3 and DEMO_{DV}^{UD} in 50 independent run. Success Rate means the percentage of get the $HV_{paretoSet} \geq 0.98 * HV_{truePF}$ in 50 independent run, NA means cannot get 100% success.

probs	NSGA-II		SPEA2		GDE3		DEMO_{DV}^{UD}	
	mean	std	mean	std	mean	std	mean	std
Evaluation Counts: $HV_{paretoSet} \geq 0.98 * HV_{truePF}$ **or Success Rate**								
ZDT1	14170.0	670.6	15998.0	912.7	9590.0	350.0	**3110.6**	450.7
ZDT2	NA(74%)	NA(74%)	NA(62%)	NA(62%)	11254.0	426.2	**4764.7**	734.9
ZDT3	12980.0	764.5	15374.0	810.9	10446.0	398.1	**4110.6**	1079.7
ZDT4	NA(80%)	NA(80%)	NA(46%)	NA(46%)	16306.0	751.6	**5136.3**	3024.5
ZDT6	NA(0%)	NA(0%)	NA(0%)	NA(0%)	4724.0	443.0	**2639.6**	478.3
DTLZ1	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)
DTLZ2	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)
DTLZ3	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)
DTLZ4	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)	NA(0%)
DTLZ5	NA(0%)	NA(0%)	NA(0%)	NA(0%)	9088.0	429.3	**3910.0**	612.5
DTLZ6	NA(0%)	NA(0%)	NA(0%)	NA(0%)	4042.0	202.1	**2326.3**	336.0

2. Acceptance rule is firstly presented as *Distance/Volume* fitness, which expands solutions to feasible searching place and forces solutions near to the Pareto-front.
3. DEMO_{DV}^{UD} can get higher hypervolume, faster convergence and well distributed solutions than NSGA-II, SPEA2 and GDE3.
4. DEMO_{DV}^{UD} need less $NFFES$ to get 98% hypervolume of true Pareto-front than NSGA-II, SPEA2 and GDE3.

6 Conclusion and Further Research

In this paper, a new differential evolution for MOPS called DEMO_{DV}^{UD} is presented. We adopt the statistical experimental method *Uniform Design* to construct the first population; we design a new fitness assignment as *Distance/Volume*. Experiment on ZDTx and DTLZx problems by jMetal 2.0, the results show that the new algorithm gets higher hypervolume, faster convergence, better distributed solutions and needs less number of fitness function numbers than NSGA-II, SPEA2 and GDE3.

The new acceptance rule *Minimum Reduce Hypervolume* is powerful to maintain the hypervolume and force the solution near to the Pareto-front, in the future, we can use this property to optimize ϵ-MOEA and $pa\epsilon$-MOEA.

References

1. Coello, C.A.C.: Evolutionary multi-objective optimization: A historical view of the Field. IEEE Computational Intelligence Magazine 1(1), 28–36 (2006)
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA - II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)

3. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm, Technical Report 103, Computer Engineering and Networks Laboratory (2001)
4. Storn, R., Price, K.: Differential evolution CA simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11, 341–359 (1997)
5. Kukkonen, S., Lampinen, J.: GDE3: The third evolution step of generalized differential evolution. Proceedings of the 2005 IEEE Congress on Evolutionary Computation (1), 443–450 (2005)
6. Deb, K., Mohan, M., Mishra, S.: Evaluating the epsilon-domination based multiobjective evolutionary algorithm for a quick computation of Paretooptimal solutions. Evolutionary Computation 13(4), 501–525 (2005)
7. Laumanns, M., Thiele, L., Deb, K., Zitzler, E.: Combining convergence and diversity in evolutionary multi-objective optimization. Evolutionary Computation 10(3), 263–282 (2002)
8. Fang, K.T., Ma, C.X.: Orthogonal and uniform design. Science Press, Beijing (2001) (in Chinese)
9. Leung, Y.W., Wang, Y.: An orthogonal genetic algorithm with quantization for global numerical optimization. IEEE Transactions on Evolutionary Computation 5(1), 41–53 (2001)
10. Zeng, S.Y., Kang, L.S., Ding, L.X.: An orthogonal multiobjective evolutionary algorithm for multi-objective optimization problems with constraints. Evolutionary Computation 12, 77–98 (2004)
11. Cai, Z.H., Gong, W.Y., Huang, Y.Q.: A novel differential evolution algorithm based on epsilon-domination and orthogonal design method for multiobjective optimization. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 286–301. Springer, Heidelberg (2007)
12. Leung, Y.-W., Wang, Y.: Multiobjective programming using uniform design and genetic algorithm. IEEE Transactions on Systems, Man, and Cybernetics, Part C 30(3), 293 (2000)
13. Durillo, J.J., Nebro, A.J., Luna, F., Dorronsoro, B., Alba, E.: jMetal: A Java Framework for Developing Multi-Objective Optimization Metaheuristics, Departamento de Lenguajes y Ciencias de la Computación, University of Málaga, E.T.S.I. Informática, Campus de Teatinos, ITI-2006-10, December (2006), http://jmetal.sourceforge.net