A Differential Evolution-Based Hybrid NSGA-II for Multi-objective Optimization

Pan Xiaoying, Zhu Jing, Chen Hao, Chen Xuejing, Hu Kaikai
School of Computer Science and Technology
Xi'an University of Posts and Telecommunications
Xi'an, China, 710121
xiaoying_pan@163.com

***Abstract*—To improve the search accuracy and diversity of non-dominated sorting genetic algorithm (NSGA-II), an improved algorithm DMNSGA-II referencing to the strategy of differential evolution to strengthen local search is proposed in this paper. The algorithm uses mutation guiding operator and crossover operator of DE to replace crossover operator in NSGA-II to enhance the local search capability and improve search accuracy. while retaining the mutation operator of NSGA-II to improve diversity. We use four benchmark test problems to investigate the performance of the DMNSGA-II algorithm, and simulation results demonstrate that the proposed algorithm can achieve a good overall performance in multi-objective optimization.**

Keywords-convergence; NSGA-II; differential operator; mutation operator; multi-objective optimization

I. INTRODUCTION

In real life, the number of designing target in many complex systems is always more than one, and they are mutual restraint so as to couldn't achieve optimal at the same time[1][2]. Such problems are called multi-objective optimization problems (MOPs). MOP always has not an optimal solution but rather a set of solutions which are called Pareto set, the corresponding objective value set called Pareto front. Traditional multi-objective optimization methods, including weighting method, constraints method, mixing method, goal programming, the maximum and minimum approach[3][4][5], have the defects of subjectivity and inoperability, and have difficulties in solving complex problems. In the last two decades, a number of intelligent bionic calculation methods have been proposed and successfully applied to multi-objective optimization. The inherent parallelism property of them break the defect of traditional methods that only one single optimal solution can be get while execute one optimization process. Among them, evolutionary algorithm which is an heuristic search based on population is able to search multiple optimal solutions in parallel, making it ideal for solving multi-objective optimization problem.

During 1990s, multi-objective optimization evolutionary algorithm (MOEA) has attracted the attention of many scholars, and different MOEAs have been proposed. The first MOEA is Vector Evaluated Genetic Algorithms (VEGA) which is proposed by Schaffer in 1985 [6], marking the start of solving MOP by evolutionary algorithm. In 1993, Fonseca and Fleming proposed a multi-objective genetic algorithm (MOGA) [7]. In 2002, Deb et al. proposed a very classic fast Nondominated Sorting Genetic Algorithm NSGA-II [8]. NSGA-II makes the optimal solution set can be achieved closely to the Pareto front, and the crowding distance mechanism makes the optimal solution set has a good distribution. It has been widely applied to the actual system design because of its good overall performance. However, it should be noted, its global search capability is good, but the search accuracy is relatively poor and the diveristy remains to be enhance because of the blind spots.

In nearly a decade of research, many scholars have proposed improvement for the NSGA-II[9][10][11], but many of these papers were aiming to improve the diversity of the solutions by taking optimization strategies. A improved dynamic crowding distance as the diversity maitenance strategy to obtain a PF with high uniformity in [12]. INSGA-II were proposed by Yaping Fu et al. in [13] aiming to use interior population and external population to prevent local optimum and increase the diversity. However, they did not consider the convergence while improving the distribution of solutions[14].

Differential Evolution (DE) [15], proposed by Price and Storn, is an stochastic and parallel searching algorithm and has been successfully employed to solve a wide range of global optimization problems. In the literature, several studies revised the DE algorithm to improve the performance of standard DE. In [16], A Parato Differential Evolutional Algorithm was proposed and applied to solve MOP successfully. Xue et al. proposed MODE [17], which introduced the Pareto rank and crowding distance mechanism mechanism into DE; Meng et al. desiged a double population-based DE algorithm to solve the constrained MOP [18]. However, the above works also confronts the problem of slow convergence rate and fall in to local optimum easier. Therefor they cannot solve the MOPs effectively.

Rakesh Angria and B.V. Babu introduced Nondominated Sorting Differential Evolution (NSDE) which combines DE and the nondominated selection of NSGA-II [19]. NSDE uses DE as the main framework, and uses nondominated selection operator to get better solution. This approach can improves the convergence speed, but it is easy to fall into local optimum and less effective.

How to enhance the local search capability and improve the convergence, while enjoying the benefits of optimal solution set approximate to Pareto front and well-balanced

978-1-4673-7338-8/15/$31.00

distribution by NSGA-II strategy, is an important but under-explored research problem. In this paper, we introduce the differential evolution (DE) and NSGA-II based genetic algorithm and propose a new differential evolution-based and mutation-preserved nondominated sorting genetic algorithm (DMNSGA-II), with the aim of improving the search accuracy and convergence. The major contributions of this paper are two-fold: 1) using the directional mutation operator of DE to intervene and disturb evolutionary direction of a solution; 2) preserving the mutation operator of NSGA-II to improve local search and the ability to escape from local optimum.

II. MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

Multi-objective optimization problem also known as multiple criteria optimization problem, which is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. In mathematical terms, a MOP with a decision vectors set X and m objective variables can be formulated as

$$\begin{cases} \min \quad y = F(x) = \left(f_1(x), f_2(x), \ldots f_m(x)\right)^T \\ s.t. \quad g_i(x) \le 0, \quad i = 1,2,\ldots,q \\ \qquad h_j(x) = 0, \quad j = 1,2,\ldots,p \end{cases} \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n) \in X \subset R^n$ is an n-dimensional decision vector, and X is the feasible set of decision vectors. $y = (y, y, \ldots, y) \in Y \subset R$ is m-dimensional objective vector, and Y is m-dimensional objective space. The objective function $F(x)$ defines m mapping functions from decision space to the objective space; $g_i(x) \le 0 (i = 1,2,\ldots,q)$ presents the number of inequality constraints is q; $h_j(x) = 0 (j = 1,2,\ldots,p)$ represents the number of equality constraints is p. On this basis, the following several important definitions are given.

Definition 1(Feasible Solution Set). For certain $x \in X$, if satisfying the constraints $g_i(x) \le 0$ and $h_j(x) = 0$ in (1), x is called a feasible solution, all of the feasible solution constitute feasible solution set, denoted as X_f, $X_f \subseteq X$.

Definition 2(Pareto Dominant). Suppose $x_A, x_B \in X_f$ are two feasible solutions of the MOP as the formula (1) shown, compared to x_B , x_A is Pareto dominant , iff

$$\begin{aligned} &\forall i = 1,2,\ldots,m.\ f_i(x_A) \le f_i(x_B) \\ &\wedge \exists j = 1,2,\ldots,m.\ f_j(x_A) < f_j(x_B) \end{aligned} \tag{2}$$

denoted as x_A f x_B , also known as x_A dominate x_B .

Definition 3(Pareto Optimal Solution and Pareto Optimal Solution Set) A solution $x^* \in X_f$ is called Pareto optimal solution (or nondominated solutions), if and only if meet the following conditions:

$$\neg \exists x \in X_f : x \text{ f } x^* \tag{3}$$

The set of all Pareto optimal solutions is called Pareto optimal solution set.

Definition 4 (Pareto Front) The surface consisted by objective vectors which correspond to optimal solutions of Pareto optimal solution set is called Pareto front P^*:

$$PF^* \text{ @} \left\{ F(x^*) = \left(f_1(x^*), f_2(x^*), \ldots, f(x^*)\right)^T \mid x^* \in P^* \right\} \tag{4}$$

III. HYBRID MULTI-OBJECTIVE OPTIMIZATION ALGORITHM DMNSGA-II

A. Fast Nondominated Sorting Genetic Algorithm

For real-coded NSGA-II, recombination and mutation operators adopt Simulated Binary Crossover (SBX) and polynomial mutation operator respectively [20]. For SBX operator, it selects two parent individuals $x_1 = (x_{11}, x_{12}, \ldots, x_{1n})$ and $x_2 = (x_{21}, x_{22}, \ldots, x_{2n})$ randomly, and the randomly selected junctions change as follows:

$$\begin{aligned} y_{i1} &= 0.5[(1-\beta)x_{1i} + (1+\beta)x_{2i}] \\ y_{i2} &= 0.5[(1+\beta)x_{1i} + (1-\beta)x_{2i}] \end{aligned} \tag{5}$$

where $\beta(u) = \begin{cases} (2u)^{\frac{1}{P_c+1}}, if\ u \le 0.5 \\ (2(1-u))^{\frac{1}{P_c+1}}, orelse \end{cases}$

u is a uniformly distributed random number of [0, 1]. The offspring individual can be obtained by y and x in a certain formula.

In polynomial mutation operator, for the parent individual $x = (x_1, x_2, \ldots, x_n)$, select the mutation point randomly, and generate a new gene according to the following formula:

$$x_i^{'} = x_i + (x_{UB,i} - x_{LB,i})\delta \tag{6}$$

where, $x_{UB,i}$, $x_{LB,i}$ are respectively upper and lower boundaries of the i-th decision variables, δ is determined according to the formula:

$\delta(u) = \begin{cases} (2u)^{\frac{1}{P_m+1}} - 1, & u < 0.5 \\ 1 - [2(1-u)]^{\frac{1}{P_m+1}}, & u \ge 0.5 \end{cases}$, where u is a uniformly distributed random number of [0, 1].

Crossover operator in evolutionary algorithms plays a crucial role. However NSGA-II's crossover operator shows poor search accuracy and global search capability, and polynomial mutation operators tends to escape from local optimum. Therefore it needs to strengthen the capacity of local search of NSGA-II.

B. Differential Evolution Algorithm(DE)

DE algorithm guides the search process intelligently through the cooperation and competition between individuals of the population. Its special mutation operator effects the individual evolutionary direction. Therefor DE has both global and local search capability, simple mutation operator, and lower complexity of genetic operation. The key process of mutation, crossover, and selection operators is shown in Fig. 1:

For each individual x_i:

1 Randomly select three solutions, conduct the mutation operation, and obtain a temporary solution t;

2 Conduct binomial crossover operation on the temporary t and the original individual x_i to get a new solution x_i';

3 Compare the obtained x_i' and original individual x_i, choose a better solution into the next generation population.

Figure 1. Differential Operator flowchart

DE algorithm has better local search capability and accuracy, but due to the randomly selected solution set, its convergence is relatively slow, and it is difficult to escape from local optima.

C. Proposed DMNSGA-II Algorithm

Since the poor search accuracy and global search capability of NSGA-II, in this section, we present the DMNSGA-II. The objective of this algorithm is to enhance search accuracy and keep the ability to escape from local optimum. To enhance the search accuracy, DMNSGA-II adopts NSGA-II as the main framework, preserving the elitism, fast nondominated sorting approach, crowding distance mechanism and binary tournament selection, and uses crossover and mutation operator of DE to replace NSGA-II crossover operator. Meanwhile, to keeping the local search capability and the ability to jump out of local optimum, DMNSGA-II preserves polynomial mutation operator of NSGA-II. The detailed procedure of this algorithm is as follow and its pseudo code of the proposed DMNSGA-II is shown in Fig. 2.

Initially, a random parent population P_0 is created. The population is sorted based on the fast nondominated sorting approach. Each solution is assigned a rank equal to its non-domination level. At first, the binary tournament selection is used to choose population P'_t. For each individual p in P'_t, with a differential probability Pd, it generates temporary individual q' by the mutation model of DE/best/2 and binomial crossover; with a mutation probability Pm, it generates new variation individual q'' according to the NSGA-II polynomial mutation operator. Then Q_t is consisted by q' and q''. The middle population R_t is constituted by the parent population P_t and the offspring Q_t. The size of R_t is 2N which has enlarged, so we have to truncate it to prepare it for the next step of the algorithm. We use the fast nondominated sorting approach and crowding distance mechanism in NSGA-II to truncate R_t .

Here, for the DE/best/2 differential mutation model, we choose $t = x_{bset} + F(x_{r2} - x_{r1}) + F(x_{r4} - x_{r3})$. It is an mutation scheme, which selects the current best individual x_{best} as the base of mutation, x_{r1} , x_{r2} , x_{r3} and x_{r4} are randomly selected, F is a scaling factor. The crossover operators in DE algorithm which combine the existing population information into evolution, makes DMNSGA-II has both the global search and local search capabilities, and improves DMNSGA-II search accuracy.

```
P0={x1, x2, ..., xN}
t=0,Pt=fast-nondominated-sort(P0)
repeat
    P't=binary-tournament-selection(Pt)
    Qt=∅
    for each p∈P't do
        if rand1<Pd then
            t=DE/best/2(p)
            q'=binomial crossover(p,t)
            Qt=Qt∪q'
        end if
        if rand2<Pm then
            q''=polynomial mutation(p)
            Qt=Qt∪q''
        end if
    end for
    Rt=Pt∪Qt
    F=fast-nondominated-sort(Rt)
    Pt+1=∅ and i=1
    until |Pt+1|+|Fi|≤N
        crowding-distance-assignment(Fi)
        Pt+1=Pt+1∪Fi
        i=i+1
    Sort(Fi, <n)
    Pt+1=Pt+1∪Fi[1:(N-|Pt+1|)]
    t=t+1
until t > gen
```

Figure 2. Implementation of DMNSGA-II algorithm

IV. Simulation with Instances and Performance Comparison

A. Test Problems

Test problems are suggested by Zitzler et al.[21], with aiming at multi-objective optimizing. We use four test problems here ZDT1, ZDT3, ZDT4 and ZDT6. These problems have two objective functions and are used as benchmark in many literatures to measure the performance of an algorithm. They are described in detail in Table I.

B. Performance Measures

In generally, to evaluating the multi-objective optimization algorithm, two goals should be considered: 1)whether the optimal solution set could converge to the true Pareto optimal solution set and 2) maintenance of diversity in solutions of the Pareto optimal solution set. Here,we use two performance metrics γ and Δ [22] to evaluate the algorithm's convergence and uniformity of distribution.. It is noted that the smaller metrics indicate a better performance.

Problem	n	Variable bounds	Objective functions	Optimal solutions	Comments
ZDT1	30	[0,1]	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x(1)/g(x)}]$ $g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	$x_1 \in (0,1)$ $x_i = 0$ $i = 2,\ldots,n$	convex
ZDT3	30	[0,1]	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x(1)/g(x)} - \frac{x_1}{g(x)}\sin(10\pi x_1)]$ $g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	$x_1 \in (0,1)$ $x_i = 0$ $i = 2,\ldots,n$	convex, disconnected
ZDT4	10	$x_1 \in [0,1]$ $x_i \in [-5,5]$ $i = 2,\ldots,n$	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x(1)/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}[x_i^2 - 10\cos(4\pi x_i)]$	$x_1 \in (0,1)$ $x_i = 0$ $i = 2,\ldots,n$	nonconvex
ZDT6	10	[0,1]	$f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9\left[\left(\sum_{i=2}^{n} x_i\right)/(n-1)\right]^{0.25}$	$x_1 \in (0,1)$ $x_i = 0$ $i = 2,\ldots,n$	nonconvex, nonuniforml space

TABLE I. DESCRIPTION OF THE TEST PROBLEMS ZDT1, ZDT3, ZDT4 AND ZDT6

- γ (Convergence metric) It measures the distance between the nondominated front Q and true Pareto optimal solution set P^* :

$$\gamma = \frac{\sum_{i=1}^{|Q|} d_i}{|Q|}$$

Where, d_i denotes the Euclidean distance (in the objective space) between the solution $i \in Q$ and the nearest member of P^*.

- Δ (Diversity metric) It measures the extent of spread among the obtained nondominated solution set:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{|Q|-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (|Q|-1)\bar{d}}$$

Where, d_i denotes the Euclidean distance between two adjacent solutions in nondominated front Q in the objective space, $\bar{d}$ is the average of d_i . d_f and d_l are the Euclidean distances between the boundary solution of the obtained nondominated set and extreme solution of true Pareto set.

C. Experimental results

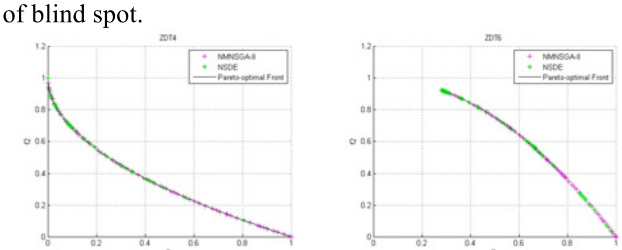
In order to evaluate the performance of our proposed algorithm, it is coded on MATLAB, running on the computer of Intel Core2 Duo CPU, 2.00GHz, 2GB memory. The encoded mode in our simulation use the real coded. The mutation model in differential operator $t = x_{bset} + F(x_{r2} - x_{r1}) + F(x_{r4} - x_{r3})$ is adopted in our algorithm and the scaling factor F is configured to 0.5. The differential probability Pd is set to 0.9. In addition, both DMNSGA-II and NSGA-II use polynomial mutation, the mutation probability can be calculated as Pm = 1/n, where n represents dimensions of the decision variables. Polynomial mutation distribution index is set to 20. In order to compare with NSDE, NSGA-II and MODE conveniently, the population size and the number of generations are set to be 100 and 250 respectively.

Table II presents the mean and variance of the convergence metric and the diversity metric obtained using four algorithms DMNSGA-II, NSDE, NSGA-II and MODE. DMNSGA-II performs better than NSDE in ZDT1 and ZDT3 in term of convergence metric. However, though its convergence metric is slightly worse in ZDT4 and ZDT6, DMNSGA-II is able to converge to true Pareto front PF^* well and distribute uniformly.

TABLE II. STATISTICS OF THE RESULTS ON TEST PROBLEMS ZDT1, ZDT3, ZDT4 AND ZDT6

Problem	Algorithm	Convergence metric	Diversity metric
ZDT1	DMNSGA-II	**0.0029 ± 0.000000**	0.5759 ± 0.005457
	NSDE	0.0521 ± 0.000031	0.6532 ± 0.0509102
	NSGA-II	0.033482 ± 0.004750	**0.390307 ± 0.001876**
	MODE	0.0058 ± 0.000000	N/A
ZDT3	DMNSGA-II	**0.0046 ± 0.000000**	**0.6590 ± 0.001596**
	NSDE	0.0385 ± 0.000731	0.7511 ± 0.023960
	NSGA-II	0.114500 ± 0.007940	0.738540 ± 0.019706
	MODE	0.021560 ± 0.000000	N/A
ZDT4	DMNSGA-II	0.0025 ± 0.00000008	**0.6198 ± 0.000558**
	NSDE	**0.0013 ± 0.000005**	0.8797 ± 0.033192
	NSGA-II	0.513053 ± 0.118460	0.702612 ± 0.064648
	MODE	0.638950 ± 0.500200	N/A
ZDT6	DMNSGA-II	0.000927 ± 0.000000	**0.6568 ± 0.005822**
	NSDE	**0.000764 ± 0.000002**	0.7261 ± 0.001998
	NSGA-II	0.296564 ± 0.013135	0.668025 ± 0.006923
	MODE	0.026230 ± 0.000861	N/A

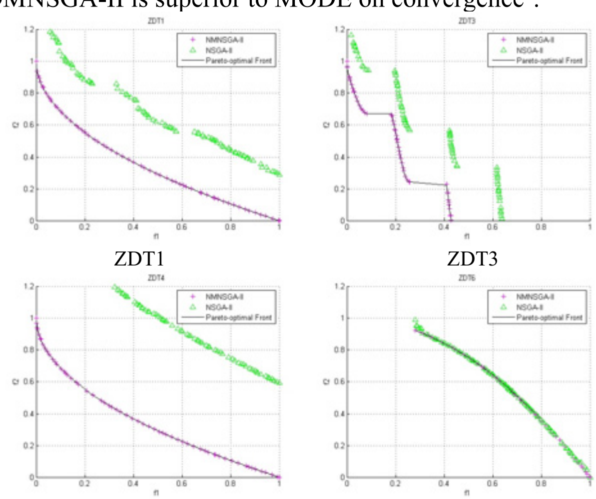
In order to demonstrate the detailed comparison between DMNSGA-II and NSDE . From Fig. 3, we can see that the distribution of diversity of DMNSGA-II is far superior to NSDE's. This is because that in the process of generating offspring population, NSDE always choose the better solution to replace the parent one without using elitist strategy. It is easy to lose the optimum solution to form part of blind spot.



ZDT4 ZDT6

Figure 3. Nondominated solutions by DMNSGA-II and NSDE on ZDT4 and ZDT6

Compared with NSGA-II, DMNSGA-II has better convergence on all test problems; in diversity, DMNSGA-II performs better than NSGA-II except ZDT1. On overall performance, DMNSGA-II has great improvement than NSGA-II. Meanwhile, on all benchmark problems, DMNSGA-II is superior to MODE on convergence[1].



ZDT1 ZDT3

ZDT4 ZDT6

Figure 4. DMNSGA-II finds better solutions than NSGA-II on four ZDT test problems

In order to understand intuitively the improvement on convergence of DMNSGA-II than NSGA-II, Fig. 4 shows

[1] The results for MODE are the average of 30 instead of 10 runs. In [17] no diversity metric was calculated.

the Pareto fronts obtained from DMNSGA-II and NSGA-II on test problems in Table I. As can be seen from the diagram, on ZDT1, ZDT3, ZDT4, the PF obtained from NSGA-II is far from PF^*, and some object place show sparse individuals and even blind spots; but the PF obtained from DMNSGA-II is more convergence to PF^*, and the distribution uniformity is obviously better than NSGA-II. The reason lies in that the guiding direction mutation operator in DE makes the solutions in population evolve towards the true Pareto front, and the local search capabilities become stronger. On ZDT6, although the two results are close to PF^*, but the population distribution of DMNSGA-II is more uniform than NSGA-II, the global search capability of it is superior to NSGA-II's.

V. CONCLUSION

In this paper, a new hybrid evolutionary algorithm is designed to improve the convergence and maintain the diversity. Testing by the benchmark problems ZDT1, ZDT3, ZDT4, ZDT6 and comparing with three mentioned algorithms, the proposed DMNSGA-II algorithm can efficiently improve convergence to solve the multi-objective optimization problem. Moreover, the spread of solutions can be well maintained. In the future, differential operator in DMNSGA-II can be improved to enhance the convergence, and use other benchmark problems to test the algorithm. In addition, the parameters need to be adjust according to actual engineering design to continuously improve their performance.

ACKNOWLEDGEMENT

This work was supported by the National Natural Science Foundation of China (Program No. 61105064, 61203311, 61373116), the Scientific Research Program Funded by Shaanxi Provincial Education Department (Program No. 14JK1667), Xiamen Science and Technology Program (No. 3502Z20141164).

REFERENCES

[1] Fan Lin, Wang Zhongmin, Wang Hai, Researches on events triggering multi-target tracking based on dynamic Cell in Wireless Sensor Networks, Journal of Xi'an University of Posts and Telecommunications, 2008(05): 76-81.

[2] Wang Song, GE Haibo, A multi-target pulsed laser range finder based on NIOS Ⅱ, Journal of Xi'an University of Posts and Telecommunications, 2011(03):49-52.

[3] L. A. Zadeh. Optimality and non-scalar-valued performance criteria[J]. IEEE Transactions on Automatic Control, 1963, 59-60.

[4] A. Charnes, W. W. Cooper, and R. Ferguson. Optimal estimation of executive compensation by linear programming[J]. Management Science, 1955, 1:138-151.

[5] C. H. Tseng and T.W Lu. Minimax multiobjective optimization in structural design[J].International Journal for Numerical Methods in Engineering, 1990, 30:1213-1228.

[6] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: Grefenstette JJ, ed. Proc. of the Int'l Conf. on Genetic Algorithms and Their Applications. Hillsdale: L. Erlbaum Associates, Inc., 1985. 93-100.

[7] Fonseca CM, Fleming PJ. Genetic algorithm for multiobjective optimization: Formulation, discussion and generation. In: Forrest S, ed. Proc. of the 5th Int'l Conf. on Genetic Algorithms. San Mateo: Morgan Kauffman Publishers, 1993. 416-423.

[8] Deb K,Pratap A,Agarwal S,et al.A fast and elitist multiobjective genetic algorithm NSGA-II[J].IEEE Transactions on Evolutionary Computation, 2002, 6 (2):182-197.

[9] K. Sindhya, A. Sinha, K. Deb, and K. Miettinen, "Local search based evolutionary multi-objective optimization algorithm for constrained and unconstrained problems", in Proc. IEEE CEC, May 2009, pp. 2919-2926.

[10] K. Sindhya, K.Miettinen, and K. Deb, "A hybrid framework for evolutionary multi-objective optimization," IEEE Trans. Evol. Comput., vol. 17,no. 4, pp.495-511, Aug. 2013.

[11] K. Deb and J. Himanshu. An evolutionary many-objective optimization algorithm using constraints. IEEE Transactions on Evolutionary Computation, in press,2014.

[12] B. Luo, J.H. Zheng, J. L. Xie, J. Wu, Dynamic crowding distance? A new diversity maintenance strategy for MOEAs, Proceedings of 5th International Conference on Natural Computation, Vol.1, 580-585, 2008.

[13] Yaping Fu, Min Huang, Hongfeng Wang, Guanjie Jiang.An Improved NSGA-II to Solve Multi-Objective Optimization Problem[C]. IEEE, 2014: 1037-1040 Control and Decision Conference, Changsha, Chinese.

[14] Maoguo Gong, Licheng Jiao, Dong Yang, et al.Researh on evolutionar multibjective otimization algorithms[J].Journal of Software,2009.

[15] Storn R, Price K. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11(4): 341-359.

[16] Abbass H A. The self-adaptive Pareto differential evolution algorithm [A]. YaoX. Proceedings of the Congress on Evolutionary Computation[C]. NY: IEEE Press, 2002. 831-836.

[17] Xue, F., Sanderson, A.C., Graves, R.J.: Pareto-based multi-objective differential evolution. In: Proceedings of the 2003 Congress on Evolutionary Computation (CEC 2003), Canberra, Australia, vol. 2, pp. 862-869. IEEE Press, Los Alamitos (2003)

[18] Hongyun Meng, Xiaohua Zhang, Sanyang Liu, et al. Double populations for constrained multiobjective optimization problems of differential evolution algorithm [J]. Chinese Journal of Computers, 2008, 31(2): 228-235.

[19] Rakesh Angira, B.V. Babu, "Non-dominated Sorting Differetial Evolution (NSDE): An Extension of Differential Evolution for Multi-objective Optimization". Proceedings of The 2nd Indian International Conference on Artificial Intelligence (IICAI-2005), Pune, India, December 20-22,2005,1428-1443.

[20] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," in Complex Syst., Apr. 1995, vol. 9, pp. 115-148.

[21] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary Computation, 2000,8(2):173 195. [doi: 10.1162/106365600568202]

[22] Wei Xin, Fujimura, S., "Parallel Quantum Evolutionary Algorithms with Client-Server Model for Multi-Objective Optimization on Discrete Problems", in Proc. IEEE CEC, June 2012, pp. 1-8.