An Improved Multi-objective PSO Algorithm with Swarm Energy Conservation

Yaoyu Xue, Liqiang Zhao, Jiahuan Wu, and Jianlin Wang

Abstract—**An improved algorithm—multi-objective particle swarm optimization with swarm energy conservation (SEC-MOPSO) is proposed, which is aimed to solve the problem of convergence and distribution in multi-objective particle swarm optimization (MOPSO) algorithm. Swarm energy conservation mechanism is used to update the velocity and position of particles. Besides, non-dominated sorting method, adaptive grid mechanism and elitism mechanism are also incorporated into SEC-MOPSO algorithm to improve searching capabilities and avoid falling into the second-best non-dominated front. The simulation results show that SEC-MOPSO has better performance than MOPSO in distribution and convergence.**

Index Terms—**multi-objective optimization, MOPSO, swarm energy conservation.**

I. Introduction

The basic idea of solving multi-objective optimization problems is to use evolutionary algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), etc. The preservation of non-dominated solutions of last generation and making them to participate in new generation of evolutionary operation is the key issue of algorithm's construction in multi-objective optimization algorithm. With the increasing number of iteration, the obtained set of all non-dominated solutions can converge to the true Pareto front during the algorithm execution.

Due to having a good distribution of solutions during the search process, GA has been widely developed in solving multi-objective optimization problems such as NSGA [1], SPEA [2] and NSGA-II [3]. While those algorithms can usually attain diversity of solutions, the issue of searching is prompted by the trend of fitness in GA will lead to low pressure of convergence and poor performance in dealing with complicated multi-objective problems.

PSO uses the best particles to guide the swarm's searching and have an obvious convergence advantage over GA. Therefore, it is also widely used for multi-objective optimization problems. Many related studies have appeared in X. Hu [4], K.E. Parsopoulos [5], C. A. C. Coello [6] and so on. One of the most representative theories is multi-objective particle swarm optimization (MOPSO) proposed by C.A.C. Coello [6], which has an excellent convergence characteristic. However, although the personal best has a certain randomness to keep the diversity of searching direction in MOPSO, the nature of one-way flow of the swarm's main information during update process leads to lack of diversity in the algorithm.

In the paper, some critical theories such as the construction of non-dominated solution set, diversity retention mechanism, elitism mechanism, the selection principle of personal best and global best are investigated based on swarm energy conservation PSO (SEC-PSO). Then the MOPSO with swarm energy conservation (SEC-MOPSO) is presented and the implementation process is given. The simulation results show that SEC-MOPSO has better convergence and distribution of solutions than MOPSO.

II. Multi-objective PSO Algorithm with Swarm Energy Conservation

A. Multi-objective Optimization

Multi-objective optimization problems often require having a balance among different interacting and conflicting objectives. The notion of an optimum solution in multi-objective is a set of tradeoff solutions called non-dominated solutions (or Pareto optimal solutions).

The multi-objective problem is described

$$\left.\begin{aligned} &\min F(X) \\ &s.t. \\ &g_i(X) \ge 0 (i = 1,2,...,k) \\ &h_i(X) = 0 (i = 1,2,...,l) \end{aligned}\right\} \quad (1)$$

where $X = (x_1, x_2, ... x_n)$ is known as a decision vector, $F(X) = (f_1(X), f_2(X), ..., f_m(X))$ is an objective vector, n is the number of decision variables, m is the number of objectives, $g_i(X)$ and $h_i(X)$ are constraints. Multi-objective optimization entails finding $X^* = (x_1^*, x_2^*, .. x_n^*)$ that optimizes $F(X^*)$ simultaneously while $g_i(X)$ and $h_i(X)$ must be satisfied.

B. Principle of SEC-MOPSO

Convergence and diversity are two conflicting problems

Manuscript received March 10, 2011; revised August 8, 2011.

This work was supported in part by the National Nature Science Foundation of China (No. 20676013) .

Y. Xue is with College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, 100029 China (e-mail: yaoyu225@126.com).

L. Q. Zhao is with College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, 100029 China (e-mail: zhaolq@mail.buct.edu.cn).

J. H. Wu is with College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, 100029 China (e-mail: wujiahuan_buct@126.com).

J. L. Wang is with College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, 100029 China (corresponding author, phone:+86-01064433803 e-mail: wangjl@mail.buct.edu.cn).

which are unavoidable issues in multi-objective optimization algorithm. Improving diversity will inevitably result in deterioration of convergence, and vice versa. Therefore, it is important to consider the trade-off between convergence and diversity in multi-objective optimization algorithm.

The main idea of SEC-MOPSO is as follows:

1) Construction of Non-dominating Set: Create a construction set; take the first individual p_1 of the set and compare the level of domination between p_1 and the remaining individuals. The individual which is dominated by p_1 will be deleted, while p_1 will be removed if it is dominated by others. Then take the second individual p_2 and compare it with other individuals except p_1 and p_2. The rule of deletion is the same as p_2; the remaining individuals can be done in the same way. The time complexity of this process is $O(rN^2)$, where r is the number of objectives and N is the size of swarm.

2) *Adaptive Grid Mechanism: Fig.*1 shows the illustration of adaptive grid based selection scheme. The objective function space where particles locate is divided into grids, while the amount of grids is g^r, where g is the area obtained from each function's space divide and r is the number of objectives. The diversity of each particle is reflected by the number of particles in the grid where the particle locates. This mechanism can improve the diversity of MOPSO effectively.

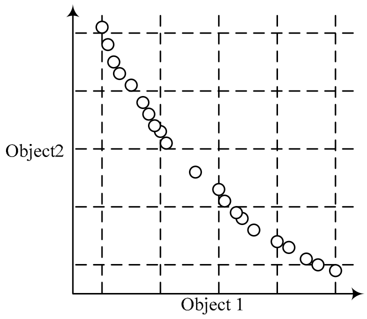

Fig 1. Illustration of adaptive grid based selection scheme

3) *Elitism Mechanism:* An external repository is created to store the newly found non-dominated solutions after each cycle. Assuming that the maximum size of external repository is M, the kth generation of non-dominated solution set is NDSet(k) and m_1 (m_1<=M) represents the number of non-dominated individuals which NDSet(k) contains. In the next generation, the newly found n_1 non-dominated individuals will be compared with the individuals in NDSet(k). The individual which is dominated by others or has same objective function value (redundant individual) will be deleted. Then the size of newly found non-dominated set TempNDSet is m_2 (m_2<=n_1+m_1). If m_2<=M, then the new generation of non-dominated solution set NDSet(k+1)= TempNDSet; If m_2>M, then one individual which locate in more crowded grid will be removed according to adaptive grid mechanism till m_2=M.

4) *Selection of Personal Best Pbest:* Personal best can be found through the comparison of objective function values in single objective problem. However, it is very difficult to determine the personal best by the same way in multi-objective optimization problem. K.E. Parsopoulos [5] presents a way of selection personal best based on the relationship of Pareto dominance. The current position of particle will be used to update Pbest if the particle dominates Pbset. Otherwise, Pbest keep being unchanged. While the method uses the idea of non-dominance, it is hard to update Pbest when there is no dominated relationship between the current particle and Pbest. On the basis of above idea, the method can be improved by the way that when there is no dominated relationship between the current particle and the Pbest, the new Pbest can be selected randomly between them. The method of calculation of crowding distance can also serve to select Pbest. The crowding distances of the current particle and Pbest in the solution space are calculated and the one that has smaller value will be selected as Pbest. Although crowding distance calculation mechanism favored the distributed uniformity of Pareto optimal solutions, it may increase the time complexity of algorithm. Therefore, random selection mechanism is a better choice because it not only solves the problem that Pbest may stop updating during the middle and later periods in the algorithm presented by K.E. Parsopoulos [5], but also avoids increasing the complexity of algorithm.

5) *Selection of Global Best Gbest:* In the iterative process, the global best Gbest is selected from NDSet in term of the principle that the particle which is located in the least crowding grid will be considered as optimum in order to keep uniform distribution of non-dominated solutions.

6) *Random Mutation Strategy:* The mutation strategy of MOPSO is that particle's regional distribution will be varied and the mutation probability will become smaller with the increasing number of iterations. Although the strategy changes particle's characteristic which favor the uniform distribution of Pareto front, it undermines the integrity of PSO and sacrifices particle's historical search information. Therefore some improvements need to be introduced as follow: New swarm POP2 which is generated by mutation takes part in the construction of non-dominated solution set but will not join the calculation in next generation. Then the algorithm can ensures diversity without compromising the historical search information of particles.

7) *Swarm Energy Conservation:* In SEC-PSO [7], the principle of particle clustering is based on the sequence of objective function value according to the clustering coefficient. However, the strategy is not suitable for multi-objective optimization algorithm. In SEC-MOPSO, non-dominated sorting method can be applied to divide the swarm into to sub-swarms. Particles in the first sub-swarm update their velocity and position according to the standard PSO equations. Assuming that there are n_1 particles in the first sub-swarm, then its energy is

$$G_1 = \sum_{i=1}^{n_1} v_i^T v_i \tag{2}$$

If the number of particles is m, then the jth sub-swarm contains n_j particles of which velocity weighted coefficient is λ_j. Particles in the jth sub-swarm update their velocity according to standard PSO equation and update their position as follow

$$x_i^{k+1} = x_i^k + \lambda_j v_i^{k+1} \tag{3}$$

$$\lambda_j v_{i+1}^T v_{i+1} = \frac{n_j (G - G_1)}{m - n_1} \tag{4}$$

where G is the energy of the whole swarm. The aim of multi-objective optimization algorithm is to find the final non-dominated solution set which can approximate the Pareto front. The advantage of swarm energy conservation mechanism is to prevent optimal individuals which consist of non-dominated solution set from falling into local optimum and preserve the diversity of solutions.

The performance of multi-objective optimization algorithm is improved using the above mechanism. Generally, the computation time of the test function taken as the objective function is far less than the computation time of the algorithm itself. In the condition of no complex computations of objective functions, the time cost of SEC-MOPSO consists of the following three parts: ① Non-dominated sorting: $O(rN^2)$; ② Non-dominated comparison: $O(rN^2)$; ③ Adaptive grid based selection: $O(r'N^2)$; Thus, the whole time complexity of this algorithm is $O(r'N^2)$.

C. Implementation

Fig.2 shows the structure of SEC-MOPSO. In the kth generation, the personal best and the global best expressed as Pbest(k) and Gbest(k), respectively. POP(k) represents the current population and NDSet(k) represents external non-dominated set. Firstly, the provisional population TempPOP1 is generated by non-dominated sorting of the current population. Then the velocity and position of each particle is updated in accordance with the swarm energy conservation mechanism and the new population POP(k+1) is generated. In the update process, the new personal best Pbest(k+1) is obtained through the non-dominated comparsion between the current particles and historical personal best. Random mutation is applied to the new population to generate the provisional population TempPOP2. After that, POP(k+1), TempPOP2 and NDSet(k) are combined and the new external non-dominated set NDSet(k+1) is generated through non-dominated comparison and adaptive grid based selection. Finally, the new global best Gbest(k+1) is obtained through adaptive grid based selection from the new external non-dominated set NDSet(k+1).

SEC-MOPSO can be implemented according to the following steps:

Step 1. Initialize the impact factors (c_1, c_2, u_1 and u_2), the inertia weight (ω) and the upper limit of evolutionary generations as *MaxDT*. Initialize the external non-dominated set NDSet and it is set to null. Initialize population POP(1) with m particles randomly in the defined space. Initialize the velocity of each particle as v and compute the whole swarm energy G.

Step 2. Evaluate the swarm POP and compute the fitness of each particle in each dimension.

Step 3. TempPOP1 is generated by non-dominated sorting of the current population.

Step 4. Operate to the particles in the first sub-swarm of TempPOP1. Update the velocity and the position of the particle according to the standard PSO equations. And compute the swarm energy G_1 of this layer. Update the personal best Pbest according to the relationship of Pareto dominance between the current particles and historical personal best.

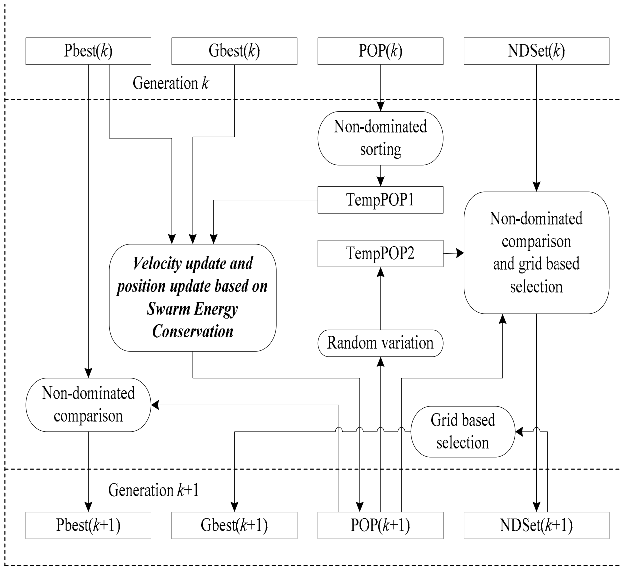

Fig 2. Structure of SEC-MOPSO

Step 5. Operate to the particles in the rest of sub-swarms of TempPOP1 except for the first sub-swarm. Update the velocity and the position of the particle according to the swarm energy conservation mechanism. The energy loss of the first sub-swarm is compensated by the particles in the rest sub-swarms through the weighted coefficient in the velocity to keep the energy conservation of the whole swarm. Update the Pbest according to the relationship of Pareto dominance between the current particles and historical personal best.

Step 6. Generate the provisional population TempPOP2 through random mutation strategy.

Step 7. Combine NDSet, POP and TempPOP2, and update NDSet through non-dominated comparison and adaptive grid based selection.

Step 8. Obtain Gbest from the NDSet.

Step 9. Check the termination condition and end the loop if the condition is satisfied; or set $k = k$+1 and go to Step 2. The termination condition is the number of evolutionary generations reaches the upper limit of evolutionary generations (*MaxDT*).

III. Simulation and Result

A. Test functions and parameters

Theories research of multi-objective optimization problem still has some aspects to be further studied, but the performance of the proposed algorithm can be analyzed through some simulation experiments. E. Zitzle [8] presents several test functions which are appropriate for multi-objective optimization problem and have been widely used. This paper selects Schaffer2 [9], Kursawe [10], ZDT1, ZDT3, ZDT6 [8] as test functions.

The performance comparison of MOPSO with NSGA2, micro-GA and PAES was discussed by C.A.C. Coello [6]. In the paper, SEC-MOPSO and MOPSO which are both based on PSO are compared with same parameters such as learning factor and inertia constant in order to verify the performance of the algorithms more accurate.

Convergence and diversity are two conflicting problems which are unavoidable issues in multi-objective optimization algorithm. Improving diversity will inevitably result in deterioration of convergence, and vice versa. Therefore, K. Deb [11] argues that the effectiveness of solutions of multi-objective optimization algorithm can be estimated through the following two aspects:

1) Convergence of Performance Index GD (Generational Distance)

The solutions generated by algorithm can approximate the true Pareto front.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^{\,2}}}{n} \tag{5}$$

where n is the number of individuals in non-dominated set, d_i is a distance between the individual i in non-dominated set and the individual which is the nearest one from i on Pareto front.

2) Distribution of Performance Index SP (Spacing)

It is better for the description of the true Pareto front that the diversity of population is preservation.

$$SP = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d} - d_i)^2} \tag{6}$$

Here, n is the number of individuals in non-dominated set, the index d_i which is in the objective function space is the distance between individual i in non-dominate set and other individual which is nearest from i, and $\bar{d}$ is the mean value of d_i.

The algorithm parameters and initial conditions are listed in TABLE I. Besides, in PSO update equations, the parameters can be set as follow: $c_1=c_2=0.5$, ω linearly decreases from 0.8 to 0.4. The algorithm runs 10 times to obtain the mean value and variance.

TABLE I. VALUES OF PARAMETERS

Test Function	Population Size	Number of Iterations
Sch	100	100
Kur	200	300
ZDT1	100	100
ZDT3	100	200
ZDT6	100	500

B. Results and discussion

Fig.3, Fig.4, Fig.5, Fig.6 and Fig.7 show the comparison of Pareto front produced by MOPSO and SEC-MOPSO for different test functions, respectively. In these figures, the value of objective function f1 is the abscissa and the value of objective function f2 is the ordinate. Icon"+" and icon" · " represent the final Pareto front produced by MOPSO and SEC-MOPSO, separately.

As shown in Fig.3, the optimal Pareto fronts got by SEC-MOPSO and MOPSO for test function Schaffer2 are all have good distribution. In Fig.4, the optimal Pareto fronts got by SEC-MOPSO and MOPSO for test function Kursawe are have good distribution in the range of f1>-17, while the optimal Pareto front got by SEC-MOPSO is under the front got by MOPSO obviously and has better performance in distribution in the range of f1<-17. The optimal Pareto fronts got by SEC-MOPSO and MOPSO for test function ZDT1 are all have good distribution as shown in Fig.5. In Fig.6, the optimal Pareto front got by SEC-MOPSO is under the front got by MOPSO obviously and has better performance in distribution for test function ZDT3. In Fig.7, the optimal Pareto front got by SEC-MOPSO is under the front got by MOPSO obviously for test function ZDT6 and the optimal results of SEC-MOPSO have better performance in uniform distribution of solution space for target function on the whole.

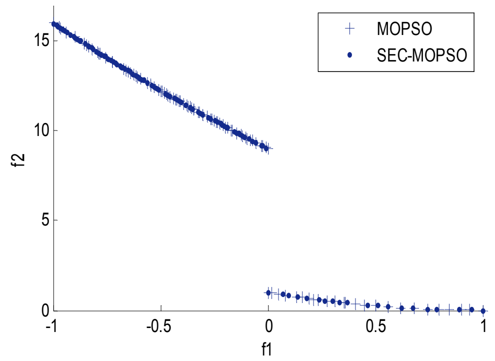

Fig 3. Pareto front for the test function Schaffer2

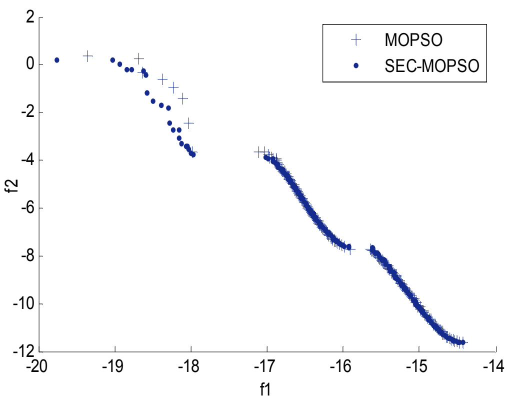

Fig 4. Pareto front for the test function Kursawe

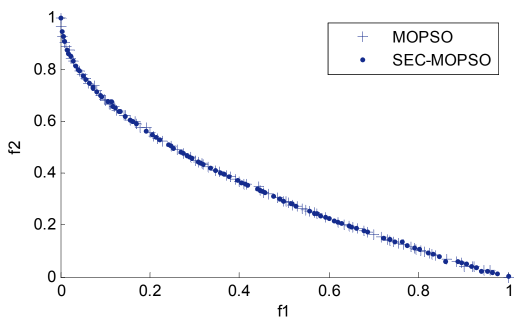

Fig 5. Pareto front for the test function ZDT1

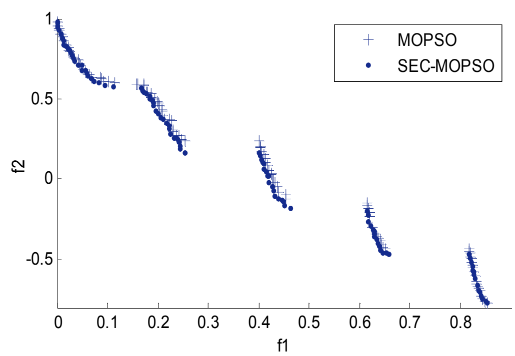

Fig 6. Pareto front for the test function ZDT3

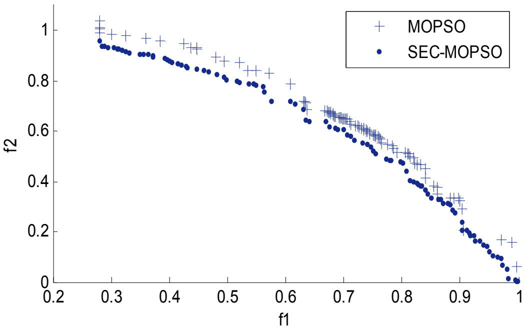

Fig 7. Pareto front for the test function ZDT6

In order to further illustrate the effectiveness and feasibility of the algorithm, the convergence of performance index *GD* and the distribution of performance index *SP* on different test functions are given in TABLE II and TABLE III.

TABLE II. CONVERGENCE OF PERFORMANCE INDEX

Test Function	Mean Value		Variance	
	MOPSO	***SEC-MOPSO***	***MOPSO***	***SEC-MOPSO***
Sch	0.00074	0.00049	0.00000	0.00000
Kur	0.02163	0.01352	0.00042	0.00021
ZDT1	0.00150	0.00050	0.00000	0.00000
ZDT3	0.02078	0.01189	0.00000	0.00000
ZDT6	0.00408	0.00341	0.00000	0.00000

TABLE III. DISTRIBUTION OF PERFORMANCE INDEX

Test Function	Mean Value		Variance	
	MOPSO	***SEC-MOPSO***	***MOPSO***	***SEC-MOPSO***
Sch	0.05818	0.04023	0.00046	0.00009
Kur	0.09131	0.09455	0.00098	0.00072
ZDT1	0.01043	0.00571	0.00001	0.00000
ZDT3	0.00515	0.00445	0.00000	0.00000
ZDT6	0.00848	0.00776	0.00002	0.00000

As one can see, the algorithm performance of MOPSO is little worse than SEC-MOPSO when using the Schaffer, Kursawe and ZDT1 functions. While for the functions of ZDT3 and ZDT6, the experimental results show that SEC-MOPSO compared to MOPSO has much better performance in convergence and distribution.

IV. CONCLUSION

This paper has presented an approach called SEC-MOPSO which extends the swarm energy conservation PSO (SEC-PSO) algorithm to handle multi-objective optimization problems. The mechanism of swarm energy conservation is used to update the velocity and position of particles. In SEC-MOPSO, on the one hand, the strategy that search oriented in PSO is used to increase the convergence pressure and rate; on the other hand, non-dominated sorting method, adaptive grid mechanism, elitism mechanism and random mutation strategy are incorporated into the algorithm to preserve the diversity of solutions. The performance of SEC-MOPSO and MOPSO is evaluated on different test functions, respectively. The results show that SEC-MOPSO has better performance than MOPSO in convergence and distribution.

REFERENCES

[1] N. Srinivas and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," Evolutionary Computation, 1994, 2, pp. 221-248.

[2] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," IEEE Transactions on Evolutionary Computation, 1999, 3, pp. 257-271.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II," IEEE Transactions on Evolutionary Computation, 2002, 6, pp. 182-197.

[4] X. Hu and R. Eberhart, "Multiobject Optimization Using Dynamic Neighborhood Particle Swarm Optimization," Proc. Of the 2002 Congress on Evolutionary Computation (CEC 02), IEEE Press, 2002, pp. 1677-1681.

[5] K. E. Parsopoulos and M. N. Vrahatis, "Particle Swarm Optimization Method in Multiobjective Problems," Proc. Of the 2002 ACM Symp. on Applied Computing (SAC 02), ACM Press, 2002, pp. 603-607.

[6] C.A.C. Coello, G. T. Pulido, and M. S. Lechuga, "Handing Multiple Objectives with Particle Swarm Optimization," IEEE Transactions on Evolutionary Computation, 2004, 8, pp. 256-279.

[7] J. L. Wang, Y. Y. Xue, T. Yu, and L. Q. Zhao, "Run-to-run Optimization for Fed-batch Fermentation Process with Swarm Energy Conservation Particle Swarm Optimization Algorithm," Chinese Journal of Chemical Engineering, 2010, 18(5), pp. 787-794.

[8] E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," Evolutionary Computation, 2000, 8(2), pp. 173-195.

[9] J. D. Schaffer, "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms," Proc. Of the 1st International Conference on Genetic Algorithms, ACM Press, 1985, pp. 93-100.

[10] F. Kursawe, "A Variant of Evolution Strategies for Vector Optimization," Lecture Notes in Computer Science, 1991, 496, pp. 193-197.

[11] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II," Lecture Notes in Computer Science, 2000, 1917, pp. 849-858.

Yaoyu Xue is from P.R. China, got B.S. degree from Henan University of Science and Technology in the year 2003. Currently, he is a Ph.D. candidate of Beijing University of Chemical Technology.
His main research fields are intelligent detection and control of complex industrial processes; safety monitoring and alarms of industrial processes.

Liqiang Zhao is from P.R. China, was born in June 1982. He received his B.S. degress from Beijing University of Chemical Technology in the year 2004 in the field of Measurement and Control. He also received Ph.D. degree in Control Theory and Control Engineering from Beijing University of Chemical Technology in 2010. He is currently working in the College of Information Science and Technology, Beijing University of Chemical Technology as a lecturer.
His area of research includes intelligent measurement and optimal modeling and control.

Jiahuan Wu is from P.R. China. He is currently pursuing the Ph.D. degree in the field of Control Theory and Control Engineering from Beijing University of Chemical Technology.
His research fields mainly include optimization algorithm and its applications in fermentation process.

Jianlin Wang is from P.R. China, was born in January 1965. He received the Ph.D. degree from the Tianjin University in 1997. Now he is a professor of Beijing University of Chemical Technology. His research interests include intelligent measurement & control technology of industrial process, and intelligent measurement & control systems.