SIMULATION EXPERIMENTS BASED ON DIFFERENTIAL EVOLUTION MODEL IN MANUFACTURING ENGINEERING

Tian Xiaoguang, Wang Hongfu, Liu lina
College of Mechanic and Electronic Engineering, Huanghe Jiaotong University,
Jiaozuo Henan, 454950, China.
Email:41028118@qq.com

ABSTRACT*: Aiming at the problem of complex multi-objective optimization, a multi-objective evolutionary algorithm based on dynamic population multi-strategy differential evolution model and decomposition mechanism is proposed. The data computation and analysis show that the convergence and diversity of the proposed algorithm are better than MOEAD / DE and NSGA –II, algorithm is an effective way to solve complex multi-objective problems. In this paper, the dynamic multi-strategy differential evolution model is integrated into MOEA / D algorithm framework, a new multi-objective evolutionary algorithm is proposed, and according to a large number of computational experiments, the validity of the proposed algorithm is proved.*
KEYWORDS*: Multi-strategy differential evolution; multi-objective optimization; an engineering example; Optimization design*

1 INTRODUCTION

Multi-objective optimization problem is widely used in scientific research and engineering applications, and is a kind of challenging optimization problem. Relative to the single-objective problem, MOP goals conflict with each other, it is difficult to get the optimal solution, but a set of compromise Pareto optimal solution set. The traditional multi-objective optimization algorithm aggregates each sub-target into a single objective function. The common disadvantage is that only one Pareto optimal solution can be obtained in one run. Since the evolutionary algorithm can obtain a set of Pareto optimal solutions after one operation, the evolutionary algorithm is more and more in the field of multi-objective optimization. The mainstream algorithms are NSGA-II (Debet al. 2002), SPEA2 (Zitzleret al. 2002), PAES (Knowlesand Corne, 2000), MOEA/D (Zhang and Li, 2007), IBEA (Zitzler& Künzli, 2004), (Bader, Zitzler and HypE, 2011) as the representative. For the above algorithm, according to the evaluation relationship can be divided into three categories: (1) Pareto dominance or deformation of the Pareto dominance evaluation of the MOEA algorithm, such as NSGA-II, SPEA2, paє-MyDE (Hernadez-Diaz, et al. 2007) Based on the performance index of the MOEA algorithm, the use of HV performance indicators, such algorithms have high time complexity, such as IBEA, HypE, etc ; (3) decomposition mechanism based on MOEA algorithm, such as MOEA / D and so on.(Bere, Berce and Nemes 2012).

Multi-objective Optimization Evolutionary Algorithm is a new kind of MOEA framework (Zhang and Li, 2007; Li and Zhang, 2009; Zhou, Zhang and Zhang, 2014). The research of this algorithm is mainly carried out from four aspects: (1) Combine the MOEA/D algorithm with other heuristic algorithm (Li and Landa-Silva, 2011; Moubayed, Petrovski, McCall, 2010; Martinez, Coello, 2011; Wang, Jia and Zhao, 2015); (2) the new decomposition mechanism into the MOEA / D framework (Zhang et al. 2010; Ishibuchi et al. 2009; Ishibuchi et al. 2010); (3) the new weight vector method (Tan et al. 2013; Ma et al. 2014; Gu and Liu, 2010; Qi et al. 2014); (4) Add a new recombination or mutation operator to MOEA/D. (Zhou et al. 2014; Chen et al. 2009; Huang and Li, 2010; Li and Landa-Silva, 2011)

2 MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM BASED ON DYNAMIC POPULATION MULTI-STRATEGY DIFFERENTIAL MODELS

2.1 The decomposition mechanism

The decomposition mechanism is proposed by Zhang in MOEA/D to solve a multi-objective problem, which decomposes the MOP into a series of sub-problems and then uses the single-objective evolutionary algorithm to solve each sub-problem. In MOEA/D, the commonly used decomposition methods have the weight vector method, the Chebyshev law and the boundary interpolation method, in which the Chebyshev method is the most widely used, the decomposition mechanism is

$$\min g^{te}(x \mid \lambda, z^*) = \max_{1 \le i \le m}\{\lambda_i \mid f_i(x) - z_i^* \mid\} \quad (1)$$
$$subject\ to \quad x \in \Omega$$

Among $z^* = (z_1^*, z_2^*, \ldots, z_m^*)^T$ is the best reference point, generally used $z_i^* = \min\{f_i(x) \mid x \in \Omega\}$, $i = 1, 2, \ldots, m$ as an approximate reference point for the optimal reference point. For each Pareto optimal solution x^* in the MOP, a corresponding weight vector λ is given such that x^* becomes the optimal solution for the corresponding single-objective problem of $g^{te}(x \mid \lambda, z^*)$. For each optimal solution of the single objective problem $g^{te}(x \mid \lambda, z^*)$ also corresponds to a Pareto optimal solution of the MOP, the Pareto optimal solution set can be obtained by changing the weight vector. Assuming that the population NP scale is N, the number of targets is m. $\Lambda = \{\lambda^1, \lambda^2, \ldots, \lambda^N\}$ is a set of weight vectors, and $\lambda^i = \{\lambda_1^i, \lambda_2^i, \ldots, \lambda_m^i\}$ meets for $\sum_{j=1}^{m} \lambda_j^i = 1$, so the i^th su-bproblem

$$g^{te}(x \mid \lambda^i, z^*) = \max_{1 \le j \le m}\{\lambda_j^i \mid f_j(x) - z_j^* \mid\} \quad (2)$$

Each sub-problem corresponds to a weight vector, and the neighbors of the sub-problem are determined by calculating the T weight vectors of each weight vector and its lowest European distance. Each generation population consists of the current optimal solution of each sub-problem, and the evolutionary operation for each sub-problem is restricted to the neighborhood. In each generation t, the individual information of the MOEA / D saved by the Chebyshev mechanism is:

(1) the N points of the group: $x^1, x^2, \ldots, x^N \in \Omega$, among x^i is the current optimal solution of the sub-problem i;

(2) $FV^1, FV^2, \ldots, FV^N$ and $FV^i = F(x^i), i = 1, 2, \ldots, N$;

(3) $z = (z_1, z_2, \ldots, z_m)^T$,Where z_i is the optimal value found by the objective function f_i so far;

2.2 Dynamic population multi-strategy differential evolution model

In the literature, it is shown that the differential evolution strategy is beneficial to improve the performance of the MOEA / D algorithm. The multi-strategy differential evolution is helpful to improve the diversity and distribution of the algorithm. The paper analyzes the advantages and disadvantages of different evolution strategies.

In this paper, we choose three kinds of evolutionary modes: DE / rand / 1 / bin, DE / best / 1 / bin and DE / rand-to-best / 1 / bin, three evolutionary patterns of evolutionary evolution, among which three evolution modes the recombination formula is as follows

(1) DE / rand / 1 / bin mode, the reorganization formula is: $V_i = X_{r_1} + F(X_{r_1} - X_{r_2})$

(2) DE / best / 1 / bin mode, the reorganization formula is: $V_i = X_{best} + F(X_{r_1} - X_{r_2})$

(3) DE / rand-to-best/ 1 / bin mode, the reorganization formula is: $V_i = X_i + F(X_{best} - X_i) + F(X_{r_1} - X_{r_2})$

In the DE / rand / 1 / bin mode, randomly select an individual X_{r_1} as the base of the individual, and by X_{r_1} and random difference vector through the reorganization of the production of individual V_i, The characteristic is the global search ability, which has strong global convergence performance and It is not easy to fall into local convergence, but its convergence rate is slower; In the DE / best / 1 / bin mode, The benchmark individual is the optimal individual X_{best} in the current population, and the individual V_i is reconstructed by X_{best} and the random differential vector.. The global search ability of the model is weak, the local search ability and the inheritance characteristics are strong, the convergence speed is fast but easy to fall into the local optimum; In the DE / rand-to-best / 1 / bin mode, it generates a fixed differential vector $(X_{best} - X_i)$ and a random difference vector $(X_{r_1} - X_{r_2})$ with X_i as the base, and then linearly combines it to try the individual V_i, which is characterized by which can keep the balance between global search and local optimization well, and has good adaptability to all kinds of optimization problems, but the robustness is poor.

There are some differences in search performance in each differential evolution mode, but there is a common pattern, that is, the reorganization of the individuals who produce the experiment is basically the same, and the new vector is obtained by linearly combining the reference individual and the difference vector. But the various patterns in the structure and evolution of the common characteristics and search performance on the characteristics of differences, so that common and differences can be co-evolution between.

Based on this, Randomly generated size N population NP, and the population NP is divided into three sub-populations IA,IB and IC, so that ξ1 individuals in the sub-population IA, ξ2 individuals in the sub-population IB, ξ3individuals in the sub-population In the IC, ξ1=ξ2=ξ3=N/3. Each sub-population is assigned an evolutionary model for co-evolution, and its multi-strategy differential evolution process is

$$for\ i = 1 : N \quad do$$

if $i \in I_A$

DE / rand / 1 / bin operation produces offspring individual y^i

else if $i \in I_B$

DE /best / 1 / bin operation produces offspring individual y^i

else

DE /rand-to-best / 1 / bin operation produces offspring individual y^i

end for

If the size of IA,IB,IC does not change, there will be an unfavorable algorithm evolution phenomenon, that is, in the evolutionary process, if certain evolution strategy stagnation will lead to the overall performance and efficiency of the algorithm, so this paper uses dynamic sub the method of population, the main process is

(1) Multi-strategy differential evolution, resulting in new offspring individual y^i.

(2) update the reference point, If $z_j < f_j(y^i)$, then $z_j = f_j(y^i),\ j = 1, 2, \ldots, m$; If $g^{te}(y^i \mid \lambda^k, z) \le g^{te}(x^k \mid \lambda^k, z)$, then $x^k = y^i$ and $F(x^k) = F(y^i)$, and $k \in B(i) = \{i_1, i_2, \ldots, i_T\}$, $B(i)$ is the neighborhood of individual i

(3) Calculate the evolutionary success rate for each strategy

$$\tau_1 = \frac{\kappa_1 / \xi_1}{\kappa_1/\xi_1 + \kappa_2/\xi_2 + \kappa_3/\xi_3}$$

$$\tau_2 = \frac{\kappa_2 / \xi_2}{\kappa_1/\xi_1 + \kappa_2/\xi_2 + \kappa_3/\xi_3}$$

$$\tau_3 = \frac{\kappa_3 / \xi_3}{\kappa_1/\xi_1 + \kappa_2/\xi_2 + \kappa_3/\xi_3}$$

Where κ_i is the number of times the progeny generated by the i -th strategy in the i -th sub-population can update at least one individual among T individuals.

Recalculate the size of the operand population, Where ξ1=N×τ1, ξ2=N×τ2, ξ1=N-ξ1-ξ2 and thenξ1、ξ2 and ξ3 are updated.

Based on the contribution of the differential evolution strategy to the evolutionary process, this kind of dynamic population is used to adjust the size of the subpopulations. This evolutionary method improves the efficiency of the algorithm and ensures the convergence of the algorithm. At the same time, Differential evolution strategies are involved in evolution, but also conducive to the diversity of algorithms. In order to avoid a certain difference in the evolutionary process of evolutionary strategy prevail over the other two strategies, then set a range of τ_1, τ_2, τ_3. If $\tau < \tau_{min}$, take $\tau = \tau_{min}$; If $\tau > \tau_{max}$, take $\tau = \tau_{max}$,and $\tau_{max} = 0.8$, $\tau_{min} = 0.15$

2.3 Algorithm flow

MOEA / D-DPMD algorithm

Enter:

(1) Multi-objective optimization problem;

(2) stop criteria;

(3) N: MOEA / D-DPMD decomposition of the number of sub-problems;

(4) $\lambda^1, \lambda^2, \ldots, \lambda^N$: uniformly distributed N weight vectors;

(5) T: the size of the neighbor of the weight vector;

(6) The size of population NP is N;

Step1 Initialization

(1) Set $EP = \varnothing$

(2) Calculate the Euclidean distance of any two weight vectors, and select the nearest T vector as its neighbor for each weight vector. Let $B(i) = \{i_1, i_2, \ldots, i_T\}$, 6 $i = 1, 2, \ldots, N$, where $\lambda^{i_1}, \lambda^{i_2}, \ldots, \lambda^{i_T}$ is the T weight vector closest to distance λ^i.

(3) Initialize the population $x^1, x^2, \ldots, x^N$, set $FV^i = F(x^i), i = 1, 2, \ldots, N$;

(4) The population $NP = \{1, 2, \ldots, N\}$ was randomly divided into three subpopulations of I_A, I_B, I_C such that ξ_1 individuals were in subpopulation I_A , ξ_2 individuals in subpopulation I_B , ξ_3 individuals in subpopulation I_C , initially set $\xi_1 = \xi_2 = \xi_3 = [N/3]$.

Step2 Dynamic cooperative differential evolution

(1) Synchronous differential operation, as described in Section 3.2;

(2) Update the reference point. If $z_j < f_j(y^i)$, then $z_j = f_j(y^i),\ j = 1, 2, \ldots, m$;

(3) Update sub-problems. If $g^{te}(y^i \mid \lambda^k, z) \le g^{te}(x^k \mid \lambda^k, z)$ then $x^k = y^i$ and $F(x^k) = F(y^i)$, where $k \in B(i) = \{i_1, i_2, \ldots, i_T\}$;

(4) The contribution rate of the operator population, as described in Section 3.2;

(5) Update ξ_1, ξ_2, ξ_3.

Step3 Stop judging

If G> Gmax, the algorithm is stopped and the result is output, otherwise it returns to Step2.

2.4 Time complexity analysis

In order to evaluate the computational efficiency of the MOEA / D-DPMD algorithm, it is compared with MOEA / D-DE and NSGA-II. MOEA / D-DE always uses a single DE operator in the process of population evolution, while MOEA / D-DPMD dynamically divides the population into three sub-populations and then uses different DE operators for different populations, both of which have The same computational framework, so the time complexity is the same. Assuming that the population size of the three algorithms is N, the number of targets is m. In a single iteration, when the new operator is transformed by the DE operator, it is necessary to update the reference point Z * and the neighbor of scale T, so the time complexity of MOEA / D-DE and MOEA / D-DPMD is O (mNT). The time cost of NSGA-II is mainly used for its non-dominated sorting operation. The non-dominated ordering needs to compare the individuals in the population to determine the dominance relation, so the time complexity is O(mN2). MOEA / D-DPMD and MOEA / D-DE have the same time complexity, but less than NSGA-II, because the neighbor scale T is less than the population size (T is about 0.1N to 0.2N).

3 EXPERIMENTAL SIMULATIONS AND ANALYSIS

In order to test the validity of the MOEA / D-DPMD algorithm, we use the LZ09_F (1-9) series reference function proposed in, which has complex PS, where F6 and F9 are non-convex PF, others Function is convex function, F7 and F8 are multi-peak problem, F1-F5 and F7-F9 are 2 objective functions, F6 is 3 objective function.

3.1 performance indicators

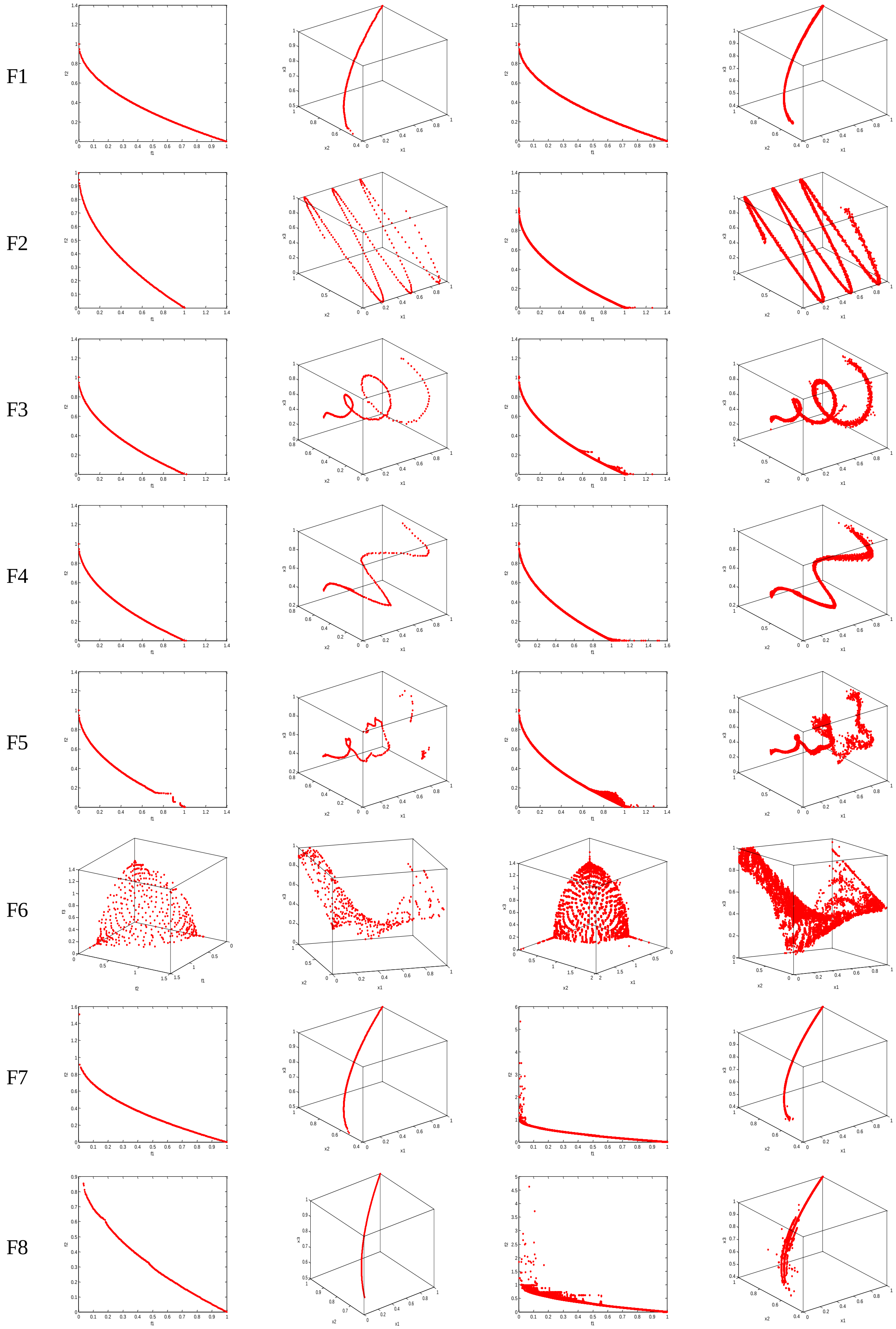
As a result of the test function can be used to obtain the theoretical optimal value, this paper selects HV and IGD two evaluation indicators to evaluate the performance of the algorithm.

3.2 Effects of different evolutionary algebra

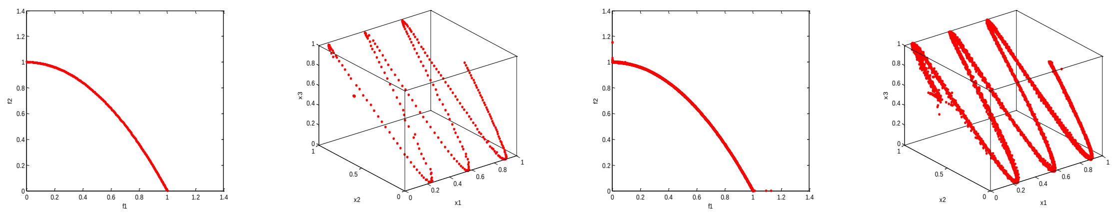
Table 1 Performance Analysis of MOEA / D-DPMD with Different Evolutionary Algebra G

	100	150	200	250	300
F1	$1.47\times10^{-4}(1.7\times10^{-5})$	$1.10\times10^{-4}(1.2\times10^{-5})$	$8.51\times10^{-5}(1.1\times10^{-5})$	$8.79\times10^{-5}(1.6\times10^{-5})$	$8.00\times10^{-5}(1.3\times10^{-5})$
F2	$1.17\times10^{-2}(2.6\times10^{-3})$	$6.70\times10^{-3}(1.7\times10^{-3})$	$3.95\times10^{-3}(1.3\times10^{-3})$	$3.51\times10^{-3}(7.3\times10^{-4})$	$3.23\times10^{-3}(1.4\times10^{-3})$
F3	$5.52\times10^{-3}(1.2\times10^{-3})$	$5.55\times10^{-3}(2.1\times10^{-3})$	$3.59\times10^{-3}(2.2\times10^{-3})$	$4.09\times10^{-4}(2.6\times10^{-3})$	$2.40\times10^{-3}(1.5\times10^{-3})$
F4	$4.10\times10^{-3}(6.6\times10^{-4})$	$3.44\times10^{-3}(7.3\times10^{-4})$	$2.51\times10^{-3}(5.3\times10^{-4})$	$2.39\times10^{-3}(6.7\times10^{-4})$	$2.39\times10^{-3}(1.2\times10^{-4})$
F5	$4.24\times10^{-3}(1.1\times10^{-3})$	$3.67\times10^{-3}(1.0\times10^{-3})$	$2.73\times10^{-3}(1.5\times10^{-3})$	$2.44\times10^{-3}(6.7\times10^{-4})$	$2.20\times10^{-3}(7.5\times10^{-4})$
F6	$5.12\times10^{-3}(1.6\times10^{-3})$	$3.42\times10^{-3}(3.3\times10^{-4})$	$3.06\times10^{-3}(4.2\times10^{-4})$	$2.96\times10^{-3}(3.8\times10^{-4})$	$2.79\times10^{-3}(4.8\times10^{-4})$
F7	$2.45\times10^{-3}(3.4\times10^{-3})$	$2.08\times10^{-3}(4.9\times10^{-3})$	$1.89\times10^{-3}(6.6\times10^{-3})$	$1.65\times10^{-3}(7.4\times10^{-3})$	$1.31\times10^{-3}(6.0\times10^{-3})$
F8	$2.13\times10^{-3}(4.0\times10^{-3})$	$1.51\times10^{-3}(3.2\times10^{-3})$	$1.18\times10^{-3}(3.4\times10^{-3})$	$1.04\times10^{-3}(1.8\times10^{-3})$	$9.68\times10^{-4}(1.8\times10^{-3})$
F9	$9.64\times10^{-3}(2.0\times10^{-3})$	$6.94\times10^{-3}(1.5\times10^{-5})$	$4.67\times10^{-3}(1.8\times10^{-3})$	$3.99\times10^{-3}(1.6\times10^{-3})$	$3.90\times10^{-3}(1.7\times10^{-3})$

Table 1 is the MOEA / D-DPMD on the nine test function operation results IGD pointer statistics. As can be seen from Table 1, with the increase of computational algebra, the IGD metric is significantly reduced, especially between 100 and 200 generations, but the gap between the generation of 250 and 300 is smaller. After 300 generations of calculations, the test set LZ09 has a complex PS and is difficult to obtain a uniform PF. Figure 1 shows the final solution set for the MOEA / D-DPMD algorithm. From 1.a, for the F1 ~ F4, F7 and F9 problems, the optimal solution PF obtained by the algorithm is very close to the real PF solution set, there are a small number of intermittent parts on the F5 and F8 problems, F6 problem The resulting PF is more uniform, but at the end there is a little point that does not converge to the endpoint. From the 1.b, it is difficult to optimize the algorithm because of the complex PS of the LZ09 problem, but the MOEA / D-DPMD algorithm can effectively approximate the real PS. Figure 1.c and 1.d for the MOEA / D-DPMD algorithm run 30 times to obtain all the PF and PS values, we can see that the algorithm can be obtained by solving all the problems of PF and PS, and the convergence and diversity Are better, but in the F8 problem PS convergence slightly worse.


F1
F2
F3
F4
F5
F6
F7
F8

F9



(a) The best available PF (b) The best available PS (c) 30 groups of PFs (d) 30 groups of PSs

Fig.1MOEA / D-DPMD algorithm in solving F1 ~ F9 problem on the final solution set

3.3 Comparison of neighborhood size

For the MOEA/D-DPMD algorithm, the size of the neighborhood size T will have a certain influence on the convergence rate and diversity of the algorithm. In order to verify the performance of different neighborhood sizes on the performance of the algorithm, the neighborhood size is divided into 10,15,20,25,30 for comparative analysis. The parameters of the MOEA / D-DPMD are set to 500 when the population size is 300 and the target is 500 when the target is 2, the maximum computational algebra G is 250, and the control parameters of the three DE strategies are CR = 1.0, F = 0.5; neighborhood search probability δ = 0.9; polynomial variation operand parameter η = 20, Pm = 1 / Var, where Var is the length of decision variable.

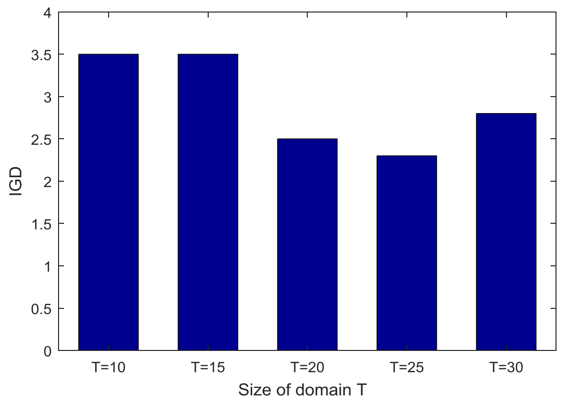


Fig. 2 IGD values for different neighborhoods T

The nine test functions were run for 30 times and evaluated using IGD. For a more accurate understanding, the use of Friedman statistical analysis of the size of the neighborhood T on the impact of algorithm performance, the specific performance shown in Figure 2. It can be seen from Fig. 2 that the IGD of the algorithm is the smallest when T = 25, so the neighborhood T = 25 is considered.

3.4 Analysis of different population size

Population size is an important parameter for multi-objective evolutionary algorithm. In order to obtain as many non-dominated solutions, it is usually necessary to use a larger population size, but increasing the size of the population will increase the computational overhead. In order to detect the effect of population size on MOEA / D-DPMD, the population size was set to 100, 200, 300, 500, 600 and other parameters were consistent. The effect of population size on the performance of the algorithm was observed. Table 2 compares the mean and standard deviation of the IGD indicators for 30 run results, where LZ09 issues other than F6 are selected for comparison. It can be observed from the table that increasing the size of the population does help to improve the performance of the algorithm, but the degree of improvement is small, which also indicates that MOEA / D-DPMD is not sensitive to N. When the population size is 300,500,600, the average of IGD is basically the same order of magnitude. In order to improve the performance of the algorithm, but also because of the increase in population and cause a substantial increase in computing costs, the population N is set to 300 is an ideal choice. For the F6 problem, due to the increase in the target one, so consider the choice of population size of 500.

Table 2. The effect of population size N on MOEA / D-DPMD

	100	200	300	500	600
F1	$2.29\times10^{-4}(3.5\times10^{-6})$	$1.10\times10^{-4}(8.1\times10^{-7})$	$7.34\times10^{-5}(1.3\times10^{-6})$	$5.29\times10^{-5}(1.0\times10^{-5})$	$4.92\times10^{-5}(1.9\times10^{-5})$
F2	$4.22\times10^{-3}(1.7\times10^{-3})$	$2.21\times10^{-3}(6.0\times10^{-4})$	$1.30\times10^{-3}(9.4\times10^{-4})$	$2.69\times10^{-4}(1.5\times10^{-4})$	$1.81\times10^{-4}(1.3\times10^{-4})$
F3	$4.74\times10^{-3}(3.0\times10^{-3})$	$2.75\times10^{-3}(2.5\times10^{-3})$	$2.01\times10^{-3}(1.6\times10^{-3})$	$4.40\times10^{-4}(4.4\times10^{-4})$	$2.10\times10^{-4}(1.6\times10^{-4})$
F4	$3.13\times10^{-3}(7.4\times10^{-4})$	$1.44\times10^{-3}(4.2\times10^{-4})$	$8.60\times10^{-4}(4.1\times10^{-4})$	$5.67\times10^{-4}(3.4\times10^{-4})$	$3.73\times10^{-4}(2.5\times10^{-4})$
F5	$3.00\times10^{-3}(8.5\times10^{-4})$	$2.54\times10^{-3}(2.3\times10^{-3})$	$1.98\times10^{-3}(1.3\times10^{-3})$	$1.19\times10^{-3}(4.7\times10^{-4})$	$1.05\times10^{-3}(2.0\times10^{-4})$
F7	$9.46\times10^{-3}(5.7\times10^{-3})$	$5.07\times10^{-3}(4.4\times10^{-3})$	$4.98\times10^{-3}(3.8\times10^{-3})$	$3.36\times10^{-3}(3.0\times10^{-3})$	$2.71\times10^{-3}(2.7\times10^{-3})$
F8	$1.01\times10^{-2}(1.7\times10^{-3})$	$8.75\times10^{-3}(1.6\times10^{-3})$	$8.73\times10^{-3}(1.7\times10^{-3})$	$7.66\times10^{-3}(2.7\times10^{-3})$	$7.03\times10^{-3}(1.6\times10^{-3})$
F9	$4.99\times10^{-3}(1.5\times10^{-3})$	$3.62\times10^{-3}(1.8\times10^{-3})$	$2.59\times10^{-3}(2.4\times10^{-3})$	$7.37\times10^{-4}(7.1\times10^{-4})$	$3.71\times10^{-4}(2.1\times10^{-4})$

3.5 Different differential evolution strategies

MOEA / D-DPMD algorithm uses three kinds of differential evolution strategies to co-evolution. In order to compare the influence of different differential evolution modes on the algorithm, different differential evolution modes are integrated into the multi-objective evolutionary algorithm framework based on decomposition mechanism. 1, 2, 3, where algorithm 1 is a separate DE / rand / 1 / bin strategy into the MOEA / D algorithm framework, algorithm 2 is a separate DE / best / 1 / bin strategy, algorithm 3 will be DE / rand -to-best / 1 / bin strategy, algorithm 4 is MOEA / D-DPMD algorithm.

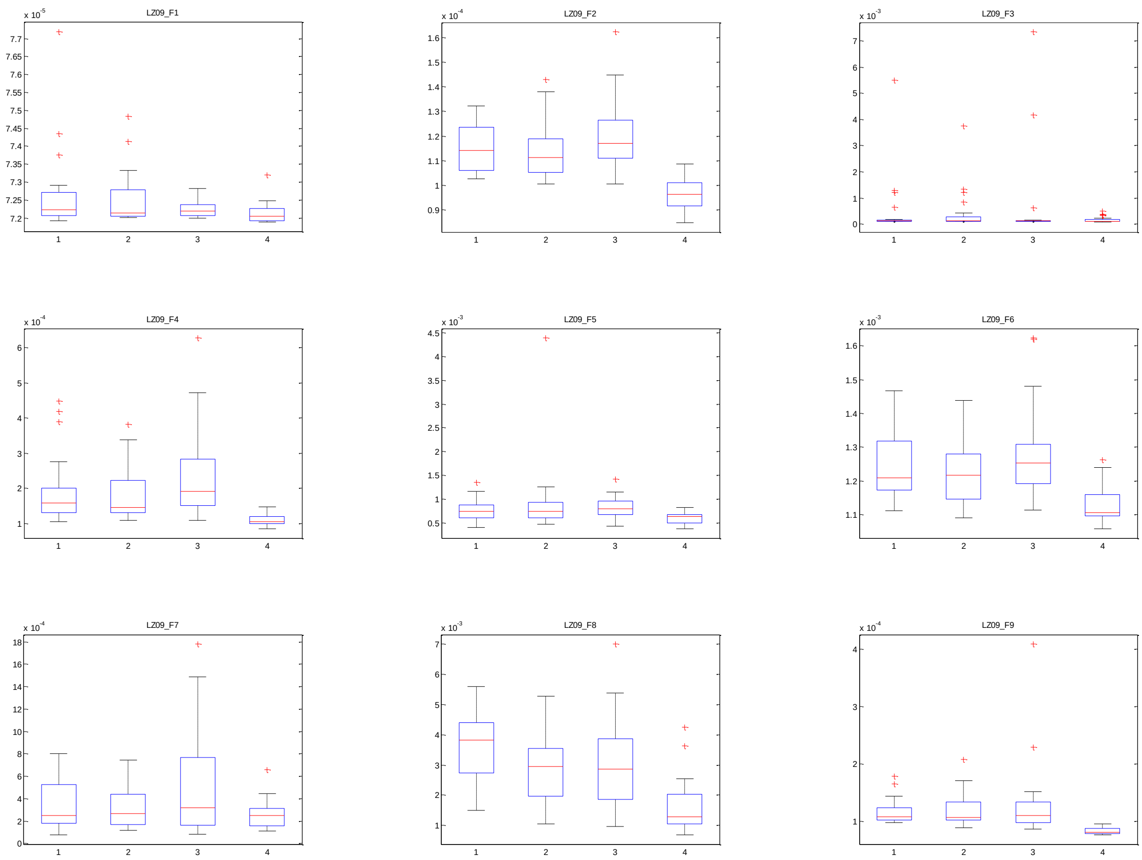


Fig. 3 Different differential evolution mode of the box diagram comparative analysis

The parameters of the four algorithms are set to: for a 2-target problem, the population size is 300, the population size is 500 for the 3-target problem, and the number of iterations is 250. In the three differential modes, CR and F are set to 1.0 and 0.5, respectively. The polynomial variant operand parameter $\eta = 20$, Pm = 1/Var, where Var is the length of the decision variable. The four algorithms run independently for 30 times for each problem, and use the box graph to represent the experimental results of the algorithm for each test problem. From Fig. 3, the PF obtained by MOEA / D-DPMD was the most concentrated on F1 ~ 9, indicating that the results obtained were very stable in 30 independent runs. Moreover, the median value of the data obtained by MOEA / D-DPMD is smaller than that of the other three strategies, which shows that the convergence and coverage are better than the other three algorithms, especially in F (1, 2, 4, 6, 8, 9) on the advantages of more obvious. In the further analysis, we can see that the median value of the algorithm is close to that of the upper and lower quartiles, and the performance of the three algorithms is similar, but the performance of MOEA / D-DPMD with co-evolution mechanism is a greater degree of promotion. It can be explained that: 1) MOEA / D-DPMD with co-evolutionary mechanism Compared with the algorithm using single-difference strategy, the Pareto front end is more close to the real Pareto front end 9 and evenly distributed, and its performance is larger The degree of improvement; 2) Co-evolutionary MOEA / D-DPMD is more robust and can solve all kinds of complex optimization problems with different PS

3.6 Algorithm comparison experiment

In this section, we compare the MOEA/ D-DPMD algorithm with the NSGA-II and MOEA / D-DE algorithms, where the calculated algebra of the three algorithms is 250 generations. For the 2-target population, the NP size is 300, When the target problem is 500, the control parameters of all DE strategies are CR = 1.0, F = 0.5, the polynomial variation operand parameter η = 20, Pm = 1/Var, where Var is the length of the decision variable. MOEA/D-DPMD algorithm and MOEA/D other parameters are: neighborhood size T = 25, neighborhood search probability δ = 0.9; NSGA-II algorithm other parameters: SBX crossover probability Pc = 0.9. For each test function are run independently 30 times, and then statistical indicators HV, and IGD mean and standard deviation, the results shown in Table 3 and 4.

Table 3. HV standard mean and variance

Function	NSGA-II		MOEA/D-DE		MOEA/D-DPMD	
	Average value	Standard deviation	Average value	Standard deviation	Average value	Standard deviation
F1	0.662	1.0×10^{-4}	**0.665***	1.1×10^{-5}	0.665	1.6×10^{-5}
F2	0.555	2.5×10^{-2}	0.661	9.5×10^{-4}	**0.662***	4.3×10^{-4}
F3	0.626	8.7×10^{-3}	0.652	1.8×10^{-2}	**0.654***	1.8×10^{-2}
F4	0.636	3.6×10^{-3}	**0.660***	2.1×10^{-3}	0.659	3.1×10^{-3}
F5	0.634	5.1×10^{-3}	0.648	8.6×10^{-3}	**0.651***	3.8×10^{-3}
F6	0.318	1.5×10^{-2}	**0.421***	1.8×10^{-3}	0.421	2.2×10^{-3}
F7	0.508	4.0×10^{-2}	0.643	2.6×10^{-2}	**0.649***	2.7×10^{-2}
F8	0.502	1.8×10^{-2}	0.495	5.0×10^{-2}	**0.509***	4.5×10^{-2}
F9	0.199	4.5×10^{-2}	0.325	4.4×10^{-3}	**0.327***	1.6×10^{-3}

* Is the optimal value

Table 4. IGD standard mean and variance

Function	**NSGA-II**		**MOEA/D-DE**		**MOEA/D-DPMD**	
	Average value	**Standard deviation**	**Average value**	**Standard deviation**	**Average value**	**Standard deviation**
F1	1.29×10^{-4}	3.5×10^{-6}	7.95×10^{-5}	9.0×10^{-7}	**7.92×10^{-5}***	4.8×10^{-7}
F2	4.54×10^{-3}	1.1×10^{-3}	1.64×10^{-4}	3.7×10^{-5}	**1.52×10^{-4}***	1.7×10^{-5}
F3	2.26×10^{-3}	7.1×10^{-4}	1.39×10^{-3}	2.5×10^{-3}	**1.17×10^{-3}***	2.0×10^{-3}
F4	2.62×10^{-3}	7.4×10^{-4}	**3.83×10^{-4}***	1.6×10^{-4}	4.75×10^{-4}	2.5×10^{-4}
F5	1.82×10^{-3}	3.6×10^{-4}	1.22×10^{-3}	9.6×10^{-4}	**9.44×10^{-4}***	2.0×10^{-4}
F6	3.07×10^{-3}	3.0×10^{-4}	**1.17×10^{-3}***	9.4×10^{-5}	1.19×10^{-3}	6.6×10^{-5}
F7	8.12×10^{-3}	3.0×10^{-3}	9.55×10^{-4}	1.0×10^{-3}	**8.11×10^{-4}***	1.2×10^{-3}
F8	6.27×10^{-3}	1.7×10^{-3}	5.35×10^{-3}	1.6×10^{-3}	**4.78×10^{-3}***	1.4×10^{-3}
F9	6.35×10^{-3}	2.2×10^{-3}	3.05×10^{-4}	1.7×10^{-4}	**2.17×10^{-4}***	8.4×10^{-5}

* Is the optimal value

It can be seen from Table 3 that MOEA/ D-DPMD algorithm obtains six optimal values, MOEA/D-DE obtains three optimal values, NSGA-II algorithm does not get the optimal value. F1, F4 and F6, MOEA/ D-DE achieved better results. The performance of MOEA / D-DPMD was the best for F2, F3, F5 and F7 ~ F9 and NSGA-II did not achieve better results. Where MOEA/ D-DE is better than MOEA/D-DPMD for F1 and F4, MOEA / D-DE is better than MOEA / D-DPMD; Problem, MOEA/D-DPMD values are better than MOEA / D-DE.

It can be seen from Table 4 that the MOEA / D-DPMD algorithm obtains seven optimal values in the nine test functions; MOEA/D-DE obtains two optimal values; NSGA-II cannot achieve better value. The data of MOEA/D-DE are better than those of MOEA/D-DE, and the mean value of MOEA/D-DE is 1.24 times of that of MOEA/D-DPMD. For F3, F5 and F7 ~ F9, The results of MOEA / D-DPMD are 1.19, 1.29, 1.18, and 1.12, 1.41 times of the mean value of MOEA/D-DE, respectively. For F1, F2 and F6 problems, the results are close to each other.

Based on the above analysis, we can conclude that MOEA/D-DPMD is more competitive than MOEA/D-DE and NSGA-II. In order to analyze the performance of multiple algorithms in a more comprehensive sense, the results are analyzed by Friedman test. Figure 4 and Figure 5 show the distribution of different pointers of each algorithm intuitively. Indicating that the better the distribution, IGD indicators, the smaller the value that better performance. The MOEA/D-DPMD algorithm obtains the value of MOEA/D-DP, which is 1.2 times that of MOEA/D-DE and 2.4 times of NSGA-II. From the numerical analysis of IGD, the MOEA/D-DPMD algorithm obtains MOEA/ D -DE 1.53 times, 2.36 times that of NSGA-II. The convergence and performance of the MOEA/ D-DPMD algorithm are far superior to the other two algorithms.

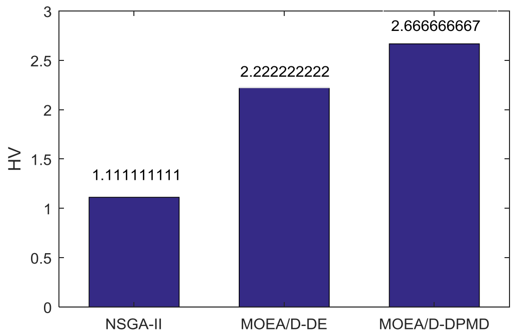


Fig.4 the HV value of the Friedman ranking histogram

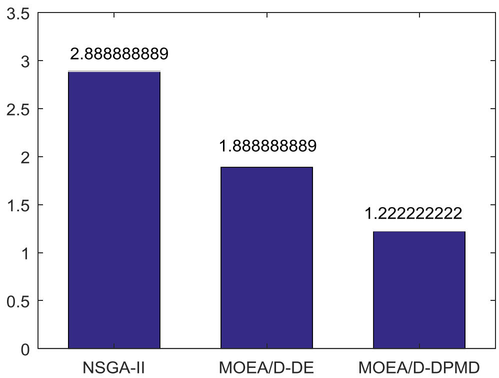


Fig.5 IGD value of the Friedman ranking histogram

4 CONCLUDING REMARKS

In this paper, a dynamic population multi-strategy differential evolution model is proposed in the framework of MOEA/D algorithm and a multi-objective evolutionary algorithm (MOEA/D-DPMD) based on dynamic population multi-strategy differential evolution model and decomposition mechanism is proposed. The algorithm divides the population into several subpopulations. Each subgroup is assigned a DE strategy. In the evolutionary process, the contribution of the next generation DE strategy is based on the contribution of different DE strategies to the population. DE strategy with each other, co-evolution. The experimental results show that:

(1) When the neighborhood size of MOEA / D-DPMD algorithm is 25, the comprehensive performance is the best.

(2) For the population size N, the larger the population size, the more PF obtained by the algorithm, but also the time complexity of the algorithm. By analyzing the size of different populations N, considering the performance improvement and calculation overhead, the population size is 300 when the target is 300, the target is 500;

(3) Different differential evolution model comparison analysis shows that the dynamic population multi-strategy differential evolution model is more close to the real PF than the single differential evolution strategy, and its performance is improved greatly.

(4) Compared with MOEA / D-DE and NSGA-II, MOEA / D-DPMD is superior to the other two algorithms in convergence and coverage. The average of the IGD in the evolution is compared, indicating that the number of evaluations required for MOEA / D-DPMD is lower than that of the other two. The next step is to further refine it in solving the high dimensional multiobjective optimization problem and the problem of engineering problems with constraints.

ACKNOWLEDGMENT

Training Plan for Young backbone Teachers of Colleges and Universities in Henan (2018GGJS194); Engineering Research Center on Intelligent Manufacturing Technology and Equipment in Henan.

REFERENCES

► Bere, Paul; Berce, Petru; Nemes, Ovidiu. (2012). *Phenomenological fracture model for biaxial fibre reinforced composites*, Composites Science and Technology, Vol: 43, Iss: 5, Pages: 2237-2243

►Bader, J., & Zitzler, E. (2011). *HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization.* Evolutionary Computation, 19(1), 45-76. doi:10.1162/evco_a_00009

► Chi-Ming Chen, Ying-ping Chen, & Qingfu Zhang. (2009). *Enhancing MOEA/D with guided mutation and priority update for multi-objective optimization.* 2009 IEEE Congress on Evolutionary Computation. doi:10.1109/cec.2009.4982950

► Deb K, Pratap A, Agarwal S, et al. (2002). *A fast and elitist multi-objective genetic algorithm: NSGA-II.* IEEE Transactions on Evolutionary Computation, 6(2): 182-197.

► Gu, F.-Q., & Liu, H.-L. (2010). *A Novel Weight Design in Multi-objective Evolutionary Algorithm.* 2010 International Conference on Computational Intelligence and Security. doi:10.1109/cis.2010.37

► H Ishibuchi, Y Sakane, N. Tsukamoto, Y. Nojima, (2009). *Adaptation of scalarizing functions in MOEA/D: An adaptive scalar zing function-based multi-objective evolutionary algorithm, in: International Conference on Evolutionary Multi-Criterion Optimization,* Springer Berlin Heidelberg, 438-452.

► Hernadez-Diaz AG, Santana-Quintero LV, Coello Coello CA, Molina J. (2007). *Pareto-Adaptive ε-dominance.* Evolutionary Computation, 15(4):493-517.

► Huang, W., & Li, H. (2010). *On the differential evolution schemes in MOEA/D.* 2010 Sixth International Conference on Natural Computation.doi:10.1109/icnc.2010.5583335

► Ishibuchi, H., Sakane, Y., Tsukamoto, N., & Nojima, Y. (2010). *Simultaneous use of different scalarizing functions in MOEA/D. Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation - GECCO '10.* doi:10.1145/1830483.1830577

► Knowles J. D., Corne D. W. (2000). *Approximating the non-dominated front using the Pareto archive evolution strategy.* Evolutionary Computation, 8(2):149-172.

► Li, H, Zhang, Q. (2009). *Multi-objective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II.* IEEE Transactions on Evolutionary Computation, 13(2):284-302.

► Li, H., & Landa-Silva, D. (2011). *An Adaptive Evolutionary Multi-Objective Approach Based on Simulated Annealing.* Evolutionary Computation, 19(4), 561-595.doi:10.1162/evco_a_00038

► Li, H., & Landa-Silva, D. (2011). *An Adaptive Evolutionary Multi-Objective Approach Based on Simulated Annealing.* Evolutionary Computation, 19(4), 561-595.doi:10.1162/evco_a_00038

► Ma, X., Qi, Y., Li, L., Liu, F., Jiao, L., & Wu, J. (2014). *MOEA/D with uniform decomposition measurement for many-objective problems.* Soft Computing,18(12),2541-2564. doi:10.1007/s00500-014-1234-8

► N Moubayed, A Petrovski, J. McCall. (2010). *A novel Smart multi-objective particle swarm optimization based on decomposition. In: International Conference on Parallel Problem Solving from Nature,* Springer Berlin Heidelberg, 1-10.

► Qi, Y., Ma, X., Liu, F., Jiao, L., Sun, J., & Wu, J. (2014). *MOEA/D with Adaptive Weight Adjustment. Evolutionary Computation,* 22(2), 231-264. doi:10.1162/evco_a_00109

► Tan, Y., Jiao, Y., Li, H., & Wang, X. (2013). *MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives.* Computers & Operations Research, 40(6), 1648-1660.doi:10.1016/j.cor.2012.01.001

► Wang, Y. H., Jia, C. H., Zhao, R. P. (2015). *Multi-objective bat algorithm based on decomposition.* Transactions of the Chinese Society for Agricultural Machinery, 46(4):316-324. (in Chinese)

► Zapotecas Martínez, S., & Coello Coello, C. A. (2011). *A multi-objective particle swarm optimizer based on decomposition. Proceedings of the 13th Annual Conference on Genetic and Evolutionary ComputationGECCO '11.*doi:10.1145/2001576.2001587

► Zhang Q, Li H. (2007). *MOEA/D: A multi-objective evolutionary algorithm based on decomposition.* IEEE Transactions on Evolutionary Computation, 11(6): 712-731.

► Zhang, Q., Li, H., Maringer, D., & Tsang, E. (2010). *MOEA/D with NBI-style Tchebycheff approach for portfolio management.* IEEE Congress on Evolutionary Computation.doi:10.1109/cec.2010.5586185

► Zhou, A. M., Zhang, Q. F., Zhang, G. X. (2014). *Multi-objective Evolutionary Algorithm Based on Mixture Gaussian Models.* Journal of Software, 25(5):913-928. (in Chinese)

► Zitzler E, Laumanns M, Thiele L. (2002). *SPEA2: Improving the strength Pareto evolutionary algorithm for multi-objective optimization. Proceedings of the Evolutionary Methods for Design, Optimization and Control.* Athens: International Center for Numerical Methods in Engineering, 95-100.

► Zitzler, E., & Künzli, S. (2004). *Indicator-Based Selection in Multiobjective Search.*Parallel Problem Solving from Nature - PPSN VIII, 832-842. doi:10.1007/978-3-540-30217-9_84