International Journal of Cognitive Informatics and Natural Intelligence
Volume 17 • Issue 1

A Coevolution Algorithm Based on Spatial Division and Hybrid Matching Strategy

Hong-Bo Wang, University of Science and Technology, Beijing, China*
https://orcid.org/0000-0002-4765-4584

Wei Huang, University of Science and Technology, Beijing, China

ABSTRACT

With the rapid development of social economy, people's demand for diversified and precise goals is increasingly prominent. In the face of a specific engineering application practice, how to find a satisfactory equilibrium solution among multiple objectives has been the focus of researchers at home and abroad. Aiming at the convergence and diversity imbalance in the current high-dimensional multi-objective evolutionary algorithm based on reference points, this article suggests a constrained evolutionary algorithm based on spatial division, angle culling, and hybrid matching selection strategy. Experimental practices show that the proposed algorithm has better performance compared with other related variants on DTLZ/WFG benchmark functions and in solving the problem of electricity market price.

KEYWORDS

Integrative Strategy, Multi-Objective Optimization, Power Dispatching

INTRODUCTION

Many practices need consider multiple objective problem (MOP) (Cohon, 1978) at the same time to optimize the overall effect in recent years. Typical work includes the second generation non-dominant sequencing genetic algorithm (NSGA-II) proposed by Deb et al. Furthermore, Zitzler et al. put forward the second-generation strength Pareto evolutionary (SPEA2) (Deb et al., 2002). NSGA-II and SPEA2 perform well in solving 2-3 objective problems with high operating efficiency and good distribution of solutions. However, when they face with higher dimensions (more than 4 targets), their disadvantages of low efficiency and poor diversity will occur, just like works in (Ikeda et al., 2001, & Khare et al., 2003) and (Purshouse et al., 2003).

Therefore, a high-dimensional multi-objective evolutionary algorithm has become a hotspot in this field. The latest MOEA/D-M2M (Liu et al., 2014) can overcome two shortcomings of MOEA/D (Zhang et al., 2007). A new improved algorithm based on MOEA(Deb et al., 2003, & Ghoreishi et al., 2015), as well as the high dimensional multi-objective evolutionary algorithm based on corner point sorting are proposed with non-dominant sorting and etc. Due to it is more difficult to calculate

DOI: 10.4018/IJCINI.326752
*Corresponding Author

the performance index, the following problems exist. (1) The inefficiency of Pareto dominance may lead to density-based diversity methods according to the pressure of environmental selection. (2) The recombination operator may be invalid. (3) The visualization of Pareto's optimal front is very difficult.

In order to trade off the relationship between convergence and diversity in high-dimensional evolutionary algorithms based on reference points and constrained multi-objective optimization problems, a Many-Objective Optimization Algorithm based on Space-Partition and Angle-based culling strategy (MaOEA-SDAC) is proposed in this paper. To meet the requirements of high-dimensional multi-objective problems with constraints, a Constrained Many-Objective Evolutionary Algorithms based on Hybrid Mating Selection (CMaOEA-HMS) is suggested in this article, which is integrated an approach of reference-point with non-dominated sorting.

The remainder of this paper is organized in the following. In the second section, two coevolution strategies and their corresponding implementation are proposed. Section III and IV design some experiments and compare the two new variants (MaOEA-SDAC and CMaOEA-HMS) with practicable strategies with the related algorithms, and summarizes the experimental results. Section V discusses that MaOEA-SDAC is applied into a joint calculation problem of residential ladder and peak-to-valley time-of-use electricity price. Conclusions are made in Section VI.

TWO IMPLEMENTATION STRATEGIES

The Framework of MaOEA-SDAC

Algorithm 1 in Table 1 is the overall pseudo code of Many-Objective Optimization Algorithm based on Space-Partition and Angle-based culling strategy (MaOEA-SDAC). In Table 1, λ represents a vector of reference points, P_0 represents an initial population, t represents an iterator, P_t represents the current generation t of a population, Q_t represents its offspring population generated by the recombination operation, R_t represents a population generated after the merger of P_t and Q_t, P_{t+1} represents the next generation produced by P_t environmental selection.

In Table 1, lines 01-03 in algorithm MaOEA-SDAC initialize some operations for a population. Lines 05-21 are an iterative process of the population, which is also its core part. Lines 08-20 run some actions in its environmental selection stage of the population.

The specific process of MaOEA-SDAC is as follows. The first step generates reference points λ, initialize the population P_0 and set the number of iterations t=0. The second step enters a loop, and the condition of the loop judgment is whether the maximum number of iterations is reached. If the related condition is met, the solution set is output; otherwise, the loop is entered. In the cycle, P_t is first matched and is selected to generate $P_t^{'}$, then $P_t^{'}$ is cross-mutated to generate Q_t, and R_t is generated by combining P_t and Q_t. Do non-dominated sorting on R_t, and merge the sorted result with P_{t+1} to generate new P_{t+1}. Then, a judgement condition will be entered, which is to generate the next population through environmental selection operation on R_t. Lines 12 and 18 are two the strategies of spatial partitioning and angle-based Culling introduced by this algorithm MaOEA-SDAC.

The Framework of CMaOEA-HMS

Algorithm 2 in Table 2 is the overall pseudo code for CMaOEA-HMS. Lines 01-03 include some initial operations. Lines 05-25 are population iterations, and lines 05-19 are its core part in this stage, which carries out matching and selection operations on a population. Lines 20-21 run the crossover mutation, and lines 22-25 are the environmental selection stage of the population. Among them, $CV\left(x\right)$ represents a degree of constraint violation of an individual, $d_{i,j}\left(x_i, x_j\right)$ is the Euclidean distance between individuals, $d\left(x, \lambda\right)$ represents the distance between an individual and its reference vector.

Table 1. Pseudo code of MaOEA-SDAC

Algorithm 1 MaOEA-SDAC pseudo code

```
01: λ = UniformPoint(); /* Randomly generate an initial reference vector*/
02: P_0 = Initialization(); /* Randomly generate a population */
03: t = 0;
04: while t ≤ T_max do
05:   P'_t = MatingSelection(P_t); /* Implement matching selective operation */
06:   Q_t = SBXCrossover(P'_t);   /* Implement Simulated Binary Crossover */
07:   Q_t = PolyMutation(Q_t);   /* Implement polynomial mutation operation */
08:   R_t = P_t ∪ Q_t;            /* merger of P_t and Q_t */
09:   {F_1, F_2, ... F_l, ...} =NDSort(R_t); /*Implement non dominated sorting operations*/
10:   P_{t+1} = P_{t+1} ∪ {F_1, F_2, ... F_{l-1}}; /*Implement competition to choose a new generation*/
11:   if |P_{t+1}| < N then
12:     C = Space-Partition(F_l);   /* Implement space division strategy */
13:     for i=1:N
14:       Q = min PBI(C_i);     /*Convergence and diversity measures*/
15:       P_{t+1} = P_{t+1} ∪ Q;     /*Implement optimization strategy*/
16:     end
17:    if |P_{t+1}| > N then
18:       P_{t+1} =Angle-based-culling(P_{t+1}); /*Implement the culling strategy of angle*/
19:    end
20:   end
21:   t=t+1;
22: end while
```

The specific chart is as follows. Firstly, the constraint violation degree $CV\left(x\right)$ of individuals is calculated, the Euclidean distance $d_{i,j}\left(x_i, x_j\right)$ between individuals is calculated, and the distance $d\left(x, \lambda\right)$ between individuals and reference vectors are calculated. Then in lines 8-19, if $t \leq maxgen^{*}\tau$ is satisfied, that is, in the early stage of population evolution, *T* solutions are randomly selected, and then an individual with the closest Euclidean distance to the *i*-th current individual joins the mating pool. Otherwise, if the number of feasible solutions is greater than or equals 1, the distance between feasible solutions and the reference point is calculated, and the smaller solution is selected. If there is no feasible solution, the infeasible solution with a lower constraint violation is selected into the matching pool.

EXPERIMENT AND ANALYSIS OF MAOEA-SDAC

Test Function Set and Performance Metrics

DTLZ (Huband et al., 2006) test function: for test functions $DTLZ_1$-$DTLZ_3$, the target dimension is 3, 5, 8, 10 and 15, respectively. And the number of decision variables is $n = n + k - 1$ and *m* is the target dimension. WFG (Deb et al., 2002) test function: for test functions WFG_1-WFG_3, the target dimension is 3, 5, 8, 10, and 15, the number of decision variables is *2*(m- 1)+20*, *m* is the target dimension, and the *k* and *l* in the WFG problem are set to *2*(m-1)* and 20, respectively.

Table 2. Pseudo code of CMaOEA-HMS

Algorithm 2 CMaOEA-HMS pseudo code

$\lambda = UniformPoint()$; /* *Randomly generate an initial reference vector**/
$P_0 = Initialization()$; /* *Randomly generate a population* */
$t = 0$;
while $t \leq T_{max}$ **do**
$CV(x) = \sum_{j=1}^{k}\langle g_j(x)\rangle + \sum_{k+1}^{k+l}|h_k(x)|$; /**Calculation of constraint violation degree* $CV(x)$*/
$d_{i,j}(x_i, x_j) = \sqrt{\sum_{k=1}^{M}(x_{i,k} - x_{j,k})}$; /* *Calculate the Euclidean distance between individuals* */
$\mathrm{d}(\mathrm{x}, \lambda) = \mathrm{d}_{j,1}(x, \lambda) + \theta * d_{j,2}(x, \lambda)$; /* *Calculate the distance between an individual and its reference vector* */
for *i = 1:N*
if $t \leq maxgen * \tau$;
*/*In the early stage of population evolution, feasible solutions and infeasible solutions are not distinguished in the mating pool. Randomly select t solutions, and then select the individual with the nearest Euclidean distance from the current i-th individual in the T solutions to join the mating pool*/*
In *T* candidates, find the closest toX_i^t, which isX_j^t;
MatingPool(i)=min($d_{i,j}(x_i, x_j)$*);*
else
/ When* $t > maxgen * \tau$*, we need to try to select feasible solutions to join the mating pool. The constraint violation degree* $CV(x) = 0$ *of the feasible solution. If there is only one feasible solution, then it is directly added to the mating pool. If there are more than one feasible solution, two feasible solutions will be randomly selected. */*
if *sum(CV==0) > 0*
MatingPool(i)=min($d(x, \lambda)$*);*
else
MatingPool(i)=min($CV(x)$*);*
end
end
end
$Q_t = SBXCrossover$(MatingPool); /* *Implement Simulated Binary Crossover* */
$Q_t = PolyMutation(Q_t)$; /* *Implement polynomial mutation operation* */
$R_t = P_t \cup Q_t$; /* *merger of* P_t *and* Q_t */
$P_{t+1} = EnvironmentalSelection(R_t)$; /**Implement environmental selection strategy to produce the next generation**/
λ = *Adaptive(*P_{t+1}*)*; /**Adaptive calculation operation**/
t=t+1;
end while

Performance Indicators

Two comprehensive performance evaluation indexes are used to simultaneously verify the convergence and diversity of the algorithm. Inverse generational distance (IGD) (Veldhuizen et al., 1999) is the retrograde distance index, and Hyper-volume Measure (HV) (Emmerich et al., 2005) is the super volume index. The IGD value is obtained by computing the Euclidean distance from the final solution set to the true Pareto front surface. The smaller the IGD value, the better its convergence and diversity of an algorithm is. The value of HV is obtained by calculating the space enclosed between its final solution set and reference points. The greater the HV value, the better the convergence and diversity of an algorithm is.

Results and Analysis on DTLZ Test Functions

Cells with a bold font in Table 3 represent the optimal IGD value obtained from the six algorithms. It can be seen from Table 3 that for the 8 target DTLZ2 problem, MaOEA-SDAC has the smallest IGD value and that the method in this paper, which can obtain a smaller IGD value in most problems compared with the other five methods. For $DTLZ_1$-$DTLZ_3$ problems, MaOEA-SDAC performs better than others. The second is algorithm MOEAD, which obtains two minimum IGD values on $DTLZ_1$ and one on $DTLZ_2$ and $DTLZ_3$, respectively. In general, MaOEA-SDAC can obtain a good IGD value. It can be seen from Table 4 that for $DTLZ_1$-$DTLZ_3$ problems, the MaOEA-SDAC performs well and obtains most of the highest HV values. The second is algorithm IBEA, which obtains the three highest HV values.

Six algorithms were used to solve the change of IGD values of $DTLZ_{1\text{-}3}$ test set of 8 targets with the number of assessments. The relevant data results are plotted in Figure 1, Figure 2 and Figure 3.

As can be seen from Figure 1, Figure 2 and Figure 3, Algorithm NSGA-III can obtain a good solution set for all the $DTLZ_1$-$DTLZ_3$ problems of 8 targets, but its convergence speed is slow, and there is a little fluctuation for the $DTLZ_1$ problem of 8 targets. Algorithm MOEA/D has good performance on the $DTLZ_1$-$DTLZ_3$ problem of 8 targets, and a good solution set is obtained with good convergence speed. Algorithms MOEA/D-DE and IBEA have better performance on the $DTLZ_1$ and $DTLZ_3$ problems of 8 targets, and can converge quickly to obtain a better solution set, but they cannot obtain a better solution set for the $DTLZ_2$ problems of 8 targets. Algorithm θ-DEA can converge quickly on the $DTLZ_1$-$DTLZ_3$ of 8 targets.

Algorithm MAOEA-SDAC can obtain a good solution set for the $DTLZ_1$-$DTLZ_3$ problems of 8 targets, and its performance is relatively stable. The convergence speed of $DTLZ_1$-$DTLZ_3$ for 8 targets is faster than that of NSGA-III.

Results and Analysis on WFG Test Functions

As you can see from Table 3, for the 3 target WFG_1 problem, MaOEA-SDAC has the smallest IGD value 1.9501E-01. For the WFG_1/WFG_3 problems, the MaOEA-SDAC has better performance and has obtained most of the optimal IGD values. In general, MaOEA-SDAC can obtain a good IGD value. The second is algorithm IBEA, which obtains the four smaller IGD values.

It can be seen from Table 4 that algorithm MaOEA-SDAC still maintains a good performance with the optimal HV Mean Values on $DTLZ_{1\text{-}3}$ and $WFG_{1\text{-}3.}$ In the designed 30 independent experimental competitions, the best running performance reached as many as 16 times, which is much higher than that of the second place. The second is algorithm IBEA (7/30).

EXPERIMENT AND ANALYSIS ON CMAOEA-HMS

Test Function Set and Performance Metrics

In order to verify the performance of algorithm CMaOEA-HMS when dealing with constrained multi-objective optimization problems, CMaOEA-HMS were compared with the results of the three current related algorithms A-NSGAIII, C-MOEA/DD and C-RVEA on two constrained test sets (C1_DTLZ1/C2_DTLZ2), synthetic indicators IGD and HV are still used to measure the performance of evolutionary algorithms.

Results and Analysis on Two Test Sets

Table 5 collects the IGD mean and standard deviation obtained by the four candidate algorithms running independently for 20 times to solve the C1_DTLZ1 and C2_DTLZ2 problem of 3-15 targets.

For the 3-C1_DTLZ1 problem, the mean IGD of A-NSGAIII, C-RVEA, C-MOEA/DD and CMaOEA-HMS are 2.6283e-02, 2.1455e-02, 2.1179e-02 and 2.0783e-02 respectively. It can be seen that the mean IGD of CMaOEA-HMS algorithm is the smallest. It can be seen from the table

Table 3. Six algorithms with different IGD mean values on DTLZ 1-3 and WFG 1-3

Func.	M	NSGAIII	θ-DEA	MOEAD	MOEA/D-DE	IBEA	MAOEA-SDAC
	3	4.3434E-02	3.3297E-01	3.5145E-02	**8.4872E-02**	2.7908E-01	2.0568E-02
	5	1.4164E-01	8.8452E-01	6.9738E-02	8.4145E-01	1.7474E-01	**5.2648E-02**
$DTLZ_1$	8	1.3065E-01	9.7138E-02	**9.5869E-02**	1.5285E-01	2.2558E-01	9.6915E-02
	10	8.2992E-01	1.1479E-01	1.0270E-01	1.0550E+00	2.0662E-01	**1.0254E-01**
	15	5.7087E-01	2.6658E-01	**1.2661E-01**	7.1230E-01	3.2896E-01	1.7368E-01
	3	5.4895E-02	5.4764E-02	5.5215E-02	7.8489E-02	8.8112E-02	**5.4258E-02**
	5	1.8635E-01	1.7673E-01	1.6807E-01	3.9770E-01	1.9010E-01	**1.6511E-01**
$DTLZ_2$	8	3.5414E-01	3.3927E-01	3.3035E-01	6.8246E-01	3.5866E-01	**3.1491E-01**
	10	6.4476E-01	4.2999E-01	**4.0966E-01**	6.5341E-01	4.1436E-01	4.2066E-01
	15	7.4452E-01	5.9962E-01	7.9902E-01	9.9308E-01	**5.9116E-01**	6.19884E-01
	3	5.4888E-02	6.0634E-02	**5.4725E-02**	8.4937E-02	4.8033E-01	5.4929E-02
	5	2.5773E-01	1.7325E-01	2.4707E+00	1.3405E+01	5.9713E-01	**1.6517E-01**
$DTLZ_3$	8	7.2283E+00	1.3457E+00	3.2724E-01	7.0179E-01	6.7194E-01	**3.1545E-01**
	10	1.6557E+01	1.2675E+00	1.1856E+00	8.1746E-01	7.2565E-01	**4.1992E-01**
	15	1.5869E+01	**6.3717E-01**	6.0167E+00	3.5578E+01	1.8965E+00	6.3826E-01
	3	3.2911E-01	5.1326E-01	5.0016E-01	1.5519E+00	1.9651E-01	**1.9501E-01**
	5	1.6013E+00	1.1488E+00	1.5143E+00	2.4432E+00	6.9438E-01	**4.9492E-01**
WFG_1	8	1.9003E+00	1.4281E+00	2.5868E+00	3.0926E+00	**1.2111E+00**	1.5951E+00
	10	2.6914E+00	2.2287E+00	3.0726E+00	3.6808E+00	1.6293E+00	**1.4861E+00**
	15	2.8252E+00	2.3945E+00	3.6050E+00	3.9701E+00	**1.9955E+00**	2.5579E+00
	3	**1.7460E-01**	2.3385E-01	1.0521E+00	5.7052E-01	2.6003E-01	1.8116E-01
	5	**7.2487E-01**	7.8452E-01	5.7509E+00	1.4033E+00	1.2610E+00	1.2068E+00
WFG_2	8	1.8284E+00	1.7290E+00	8.9508E+00	4.0032E+00	2.7540E+00	**1.5890E+00**
	10	6.2121E+00	2.8397E+00	1.7098E+01	4.9536E+00	8.0355E+00	**1.5152E+00**
	15	1.3097E+01	1.8361E+01	2.7713E+01	**8.0311E+00**	1.6457E+01	1.6040E+01
	3	9.7709E-02	1.3383E-01	1.5021E-01	1.4698E-01	4.2233E-02	**3.2977E-02**
	5	4.7450E-01	5.0530E-01	1.2201E+00	1.7917E+00	**1.3117E-01**	1.5469E-01
WFG_3	8	9.4931E-01	1.1597E+00	3.8427E+00	2.4929E+00	4.6631E-01	**3.7686E-01**
	10	8.5321E-01	9.9318E-01	6.3293E+00	3.0275E+00	7.3962E-01	**5.5198E-01**
	15	4.0532E+00	2.2846E+00	9.9090E+00	3.7987E+00	**8.7455E-01**	3.2482E+00

that CMaOEA-HMS obtains almost all the optimal values in solving C1_DTLZ1 problem and has a good performance. In the designed 10 independent experimental competitions on C1_DTLZ1 problem, the best running performance reached 9 times. According to the standard variance value of IGD, CMaOEA-HMS has good stability. Secondly, algorithm C-MOEA/DD performs better than A-NSGAIII and C-RVEA. Compared with A-NSGAIII, C-RVEA obtains the best IGD values, but those invalid values appear, which shows that algorithm C-RVEA does not solve the real solution set of the problem when dealing with the C1_DTLZ1 problem of 10 targets.

Table 6 collects the HV mean and standard deviation obtained by the four candidate algorithms running independently for 20 times to solve the C1_DTLZ1 and C2_DTLZ2 problem of 3-15 targets.

It can be seen from Table 5 that algorithm CMaOEA-HMS has the smallest IGD mean value for C2_DTLZ2 of 8 targets. In addition, CMaOEA-HMS has a good performance in dealing with C2_DTLZ2 of 3 targets, obtaining a better IGD mean value and standard variance. In terms of C2_DTLZ2 of 5 and 8 targets, CMaOEA-HMS has better performance and stability. On the C2_DTLZ2 problem of targets 10 and 15, CMaOEA-HMS obtained the optimal standard variance of IGD and had better

Figure 1. IGD mean values curves of six algorithms on DTLZ 1-8

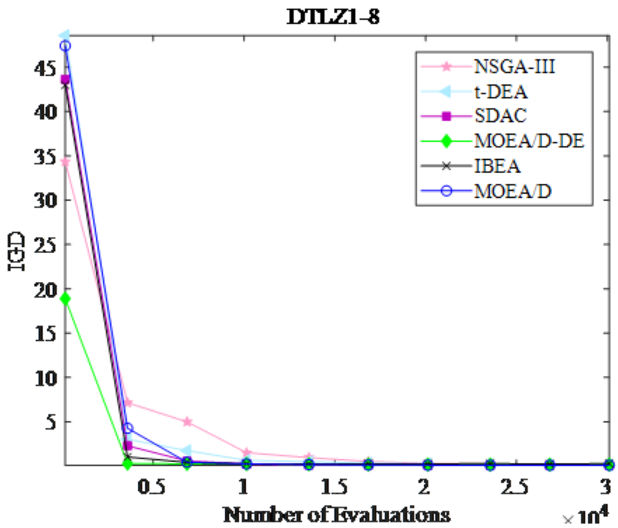
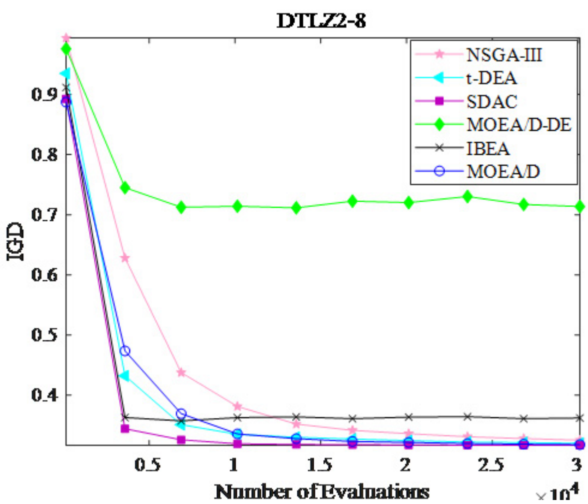


Figure 2. IGD mean values convergence curves of six algorithms on DTLZ 2-8

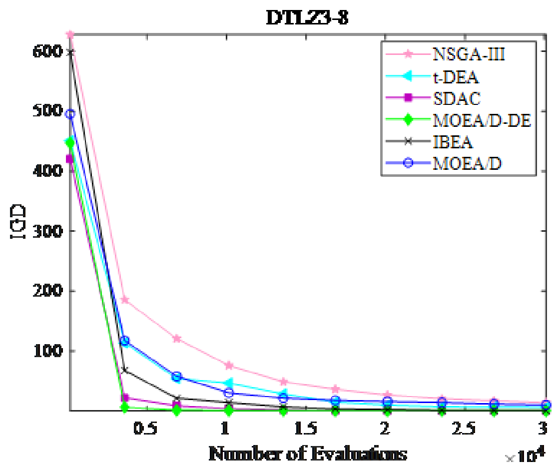


stability. From Table 6, for the C2_DTLZ2 problem of 8 targets, the HV mean of algorithm CMaOEA-HMS is the largest, namely, CMaOEA-HMS algorithm has a good performance in C2_DTLZ2 of 3, 5 and 8 targets, and has obtained the optimal HV mean and good stability, on the C2_DTLZ2 of 10 targets, there is a good HV standard variance, indicating a relatively good stability.

Comparative Analysis of Running Time

Figures 4 and 5 record the average time taken by the four algorithms to process two constraint problems of 3, 5, 8, 10 and 15 targets for 20 times.

Figure 3. IGD mean values convergence curves of six algorithms on DTLZ 3-8



For the C1-DTLZ1 problem of 3 targets, the required time of A-NSGAII, C-RVEA, CMAOEA-HMS, and C-MOEA/DD were 23.2, 24.7, 36.4, and 350.4 (seconds), respectively, in Figure 4. As can be seen from the figure, the cost of time used by the three algorithms A-NSGAII, CMAOEA-HMS and C-RVEA is not much different. Algorithm CMaOEA-HMS is slightly running time longer, due to its matching selection and computing the Euclidean distance between an individual and its reference point. C-MOEA/DD takes the longest time, which has a large gap compared with the other three algorithms. It can be seen that different targets correspond to different population sizes. When the population size is large, the time taken by the four algorithms will increase. In addition, problem C2_DTLZ2 in Figure 5 is complex and its Pareto front is discontinuous. Therefore, the time required is relatively long.

SIMULATION APPLICATION

From the power optimization scheduling (Deb et al., 2002, & Huband et al., 2006), how to guide the industrial residents, businesses and users of electricity, and to improve the tense situation of energy, is becoming a hot topic.

Model Solving Process Based on MaOEA-SDAC

Step 1: Initialize parameters, generate the reference points, set the upper and lower limits of the decision variables according to the scope of $\alpha, \beta, \gamma (0.5 \le \alpha \le 1, -0.1 \le \beta \le 0.1, -0.7 \le \gamma \le 0)$ and then initialize the population.

Step 2: Generate an offspring population: the parent population is matched and selected, and then the crossover mutation is carried out.

Step 3: Merge the parent with the offspring population, and then make environmental selection for the next iteration.

Step 4: Judge whether the terminal condition has been reached or not. If not, loop step 2. If the termination condition is reached, Pareto optimal solution set is generated.

Step 5: Select the optimal solution from Pareto optimal set, the specific steps are as follows:

a. Calculate the membership function u of the i-th Pareto solution to the j-th target value, as shown in Equation (1):

Table 4. Six algorithms with different HV mean values on DTLZ1-3 and WFG1-3

Func.	M	NSGAIII	θ-DEA	MOEAD	MOEA/D-DE	IBEA	MAOEA-SDAC
	3	1.3021E-01	1.7802E-02	2.2846E-02	4.3993E-03	6.7075E-02	**1.4003E-01**
	5	0.0000E+00	1.8435E-03	4.1477E-02	2.8473E-04	**4.0473E-02**	4.0316E-02
$DTLZ_1$	8	4.7564E-03	3.3679E-07	6.3511E-03	7.1124E-03	5.7919E-03	**8.3529E-03**
	10	1.5487E-03	1.8254E-03	**2.7412E-03**	1.2548E-03	1.5349E-04	2.5313E-03
	15	1.0543E-04	2.8840E-05	7.8255E-05	4.6764E-05	3.5739-05	**1.2682E-04**
	3	7.3929E-01	7.4009E-01	7.3912E-01	6.9656E-01	7.3851E-01	**7.4490E-01**
	5	1.2173E+00	1.2267E+00	1.2607E+00	6.6797E-01	1.2992E+00	**1.3092E+00**
$DTLZ_2$	8	1.7807E+00	1.9136E+00	1.8462E+00	1.0639E+00	**1.9941E+00**	1.9804E+00
	10	1.4565E+00	2.3043E+00	2.0811E+00	1.3458E+00	**2.4947E+00**	2.4153E+00
	15	2.7965E+00	3.9217E+00	3.7217E+00	1.4723E+00	4.0068E+00	**4.1389E+00**
	3	7.3795E-01	7.1602E-01	7.3852E-01	6.5040E-01	3.3036E-01	**7.4249E-01**
	5	7.2316E-01	8.2536E-01	1.2054E+00	1.01124E+00	6.9654E-01	**1.3065E+00**
$DTLZ_3$	8	1.5242E+00	9.0516E-01	**1.9955E+00**	7.8541E-01	1.8015E+00	1.9791E+00
	10	1.7215E+00	1.5246E+00	**2.5486E+00**	1.1873E+00	1.8512E+00	2.5140E+00
	15	0.0000E+00	1.1249E+00	2.2900E+00	1.3328E+00	3.1065E+00	**4.1383E+00**
	3	5.2358E+01	4.8673E+01	4.4838E+01	1.7506E+01	5.9366E+01	**5.9414E+01**
	5	2.4666E+03	3.2839E+03	4.4176E+03	1.0408E+03	4.8862E+03	**5.9719E+03**
WFG_1	8	9.5722E+06	1.1510E+07	9.0080E+06	3.9816E+06	**2.7420E+07**	2.0548E+07
	10	2.6818E+09	3.7153E+09	2.9204E+09	1.8509E+09	5.6012E+09	**8.6339E+09**
	15	6.7032E+16	9.5666E+16	5.0831E+16	5.3341E+16	**1.3823E+17**	1.3811E+17
	3	5.9273E+01	5.9464E+01	5.6858E+01	5.6091E+01	5.9356E+01	**5.9692E+01**
	5	5.9731E+03	6.0150E+03	5.7596E+03	5.8592E+03	6.0546E+03	**6.0689E+03**
WFG_2	8	2.1530E+07	**2.1677E+07**	1.9900E+07	2.1444E+07	2.1562E+07	2.1319E+07
	10	9.5227E+09	9.4710E+09	8.5929E+09	**9.5565E+09**	9.4928E+09	9.5142E+09
	15	**1.7680E+17**	1.4309E+17	1.6026E+17	1.7434E+17	1.7490E+17	1.6993E+17
	3	6.1292E+00	6.1138E+00	5.6349E+00	5.6722E+00	6.5264E+00	**2.5752E+01**
	5	1.4835E+03	1.5210E+03	0.0000E+00	1.0858E+03	**5.9080E+03**	5.6094E+03
WFG_3	8	6.1434E+05	1.3470E+07	0.0000E+00	9.8138E+06	9.4258E+06	**2.8418E+07**
	10	0.0000E+00	0.0000E+00	0.0000E+00	**1.8081E+05**	0.0000E+00	0.0000E+00
	15	**1.6028E+17**	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

$$u_{ij} = \begin{cases} \dfrac{f_{j\max} - f_{ij}}{f_{j\max} - f_{j\min}}, & j = 1,2 \\ \dfrac{f_{ij} - f_{j\min}}{f_{j\max} - f_{j\min}}, & j = 3 \end{cases} \quad (1)$$

Table 5. Four algorithms on C1_DTLZ1 and C2_DTLZ2 with IGD mean and standard variance values

Func.	M	A-NSGAIII	C-RVEA	C-MOEA/DD	CMaOEA-HMS
C1-DTLZ1	3	2.6283E-02	2.1455E-02	2.1179E-02	**2.0783E-02**
		4.9900E-03	6.9900E-04	1.0700E-03	**1.6800E-04**
	5	7.0094E-02	5.7608E-02	5.4964E-02	**5.1704E-02**
		2.2400E-02	9.8200E-03	6.1600E-03	**1.4900E-04**
	8	1.6400E-01	1.0348E-01	9.8047E-02	**9.3455E-02**
		5.7900E-02	1.4700E-02	7.8700E-03	**8.2300E-04**
	10	2.0469E-01	NaN	1.1417E-01	**1.0471E-01**
		5.2500E-02	NaN	1.8800E-02	**4.1200E-04**
	15	2.0807E-01	**1.5431E-01**	1.7580E-01	1.5759E-01
		3.3600E-02	1.0500E-02	1.6400E-02	**3.8900E-03**
C2-DTLZ2	3	**4.4150E-02**	5.3803E-02	5.0109E-02	4.4770E-02
		2.7800E-04	1.6600E-03	4.0600E-04	5.1700E-04
	5	1.3369E-01	1.4588E-01	1.4079E-01	**1.3362E-01**
		1.2000E-03	1.5800E-03	**7.7200E-04**	8.0900E-04
	8	4.7154E-01	2.8482E-01	3.1509E-01	**2.3647E-01**
		2.9000E-01	3.4100E-03	7.3400E-02	**3.3900E-03**
	10	5.8618E-01	**4.3584E-01**	4.5752E-01	4.6151E-01
		4.8500E-02	3.6300E-02	3.4800E-02	**1.4600E-03**
	15	5.5870E-01	**5.4083E-01**	5.5420E-01	6.3328E-01
		2.5800E-01	**3.8000E-02**	4.2300E-02	4.5000E-02

In Equation (1) f_{ij} is the *j*-th target value of the *i*-th Pareto solution; f_{jmin} and f_{jmax} are the minimum and maximum values of the *j*-th target of Pareto solution, respectively.

b. Calculate the weight w_i of the *j*-th target, as shown in Equation (2):

$$w_j = \frac{\sum_{i=1}^{N}\sum_{k=1}^{N}(u_{ij} - u_{kj})^2}{\sum_{j=1}^{3}\sum_{i=1}^{N}\sum_{k=1}^{N}(u_{ij} - u_{kj})^2} \tag{2}$$

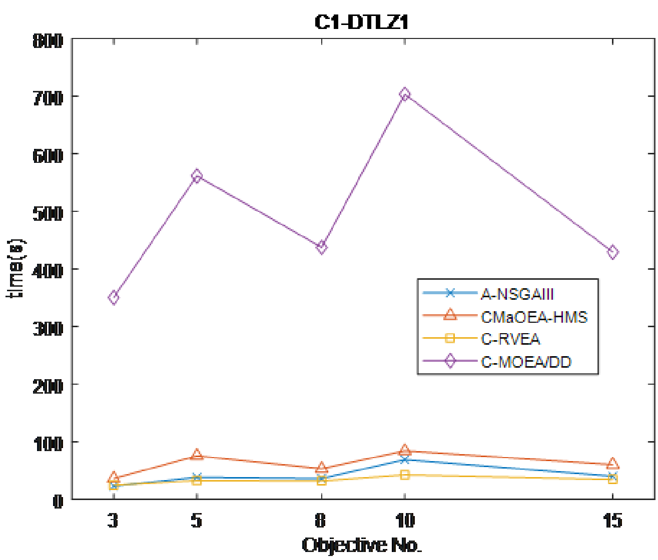
c. Calculate the selection priority F_i of the *i*-th Pareto solution, as shown in Equation (3), and select the maximum value of F_i as the optimal solution:

$$F_i = \sum_{j=1}^{3} w_j u_{ij} \tag{3}$$

Table 6. Four algorithms on 3-15 Objective C1_DTLZ1 and C2_DTLZ2 problem with different HV mean and standard variance values

Func.	M	A-NSGAIII	C-RVEA	C-MOEA/DD	CMaOEA-HMS
C1-DTLZ1	3	1.3459E-01	1.3758E-01	1.3614E-01	**1.3949E-01**
		2.5300E-03	1.6000E-03	2.6600E-03	**5.6500E-04**
	5	4.3968E-02	4.6913E-02	4.6864E-02	**4.8573E-02**
		2.5400E-03	1.6500E-03	1.4900E-03	**3.6100E-04**
	8	6.6633E-03	8.0535E-03	7.8627E-03	**8.1482E-03**
		1.1600E-03	1.0800E-04	1.8500E-04	**7.5500E-05**
	10	1.8351E-03	NaN	2.3172E-03	**2.4076E-03**
		2.8500E-04	NaN	6.5700E-05	**2.5200E-05**
	15	1.0338E-04	1.2025E-04	1.1301E-04	**1.2430E-04**
		8.3800E-06	**1.0800E-06**	6.5800E-06	1.4300E-06
C2-DTLZ2	3	6.7954E-01	6.5785E-01	6.6953E-01	**6.8211E-01**
		1.8600E-03	3.8300E-03	3.9300E-03	3.8200E-03
	5	1.1878E+00	1.1657E+00	1.1990E+00	**1.2104E+00**
		4.8600E-03	1.0600E-02	**2.3700E-03**	3.6600E-03
	8	1.3074E+00	1.7147E+00	1.5869E+00	**1.7996E+00**
		6.0900E-01	1.9000E-02	1.8900E-01	**1.5400E-02**
	10	**1.5121E+00**	1.1360E+00	1.4689E+00	1.2999E+00
		2.2100E-01	6.7500E-02	1.2300E-01	**1.0100E-02**
	15	2.8823E+00	**3.2477E+00**	2.5693E+00	2.3463E+00
		1.1500E+00	**1.2900E-01**	4.5400E-01	1.3400E-01

Figure 4. Comparison of the average running time of the four algorithms on C1-DTLZ1



EXPERIMENTAL RESULTS AND ANALYSIS

A user's actual load data is used to verify the effectiveness of the algorithm. Table 7 lists the optimal Pareto solutions α, β, γ and values obtained by the four algorithms. Before the implementation of the combined price, the price of the system was P_0 (unit: kW/h). After the implementation of the combined price, the price of 30% of the total user load was still P_0, and the price of the rest 70% of the load in peak, flat and valley periods was $(1+\alpha)P_0$, $(1+\beta)P_0$, $(1+\gamma)P_0$, respectively.

Figure 5. Comparison of the average running time of the four algorithms on C2_DTLZ2

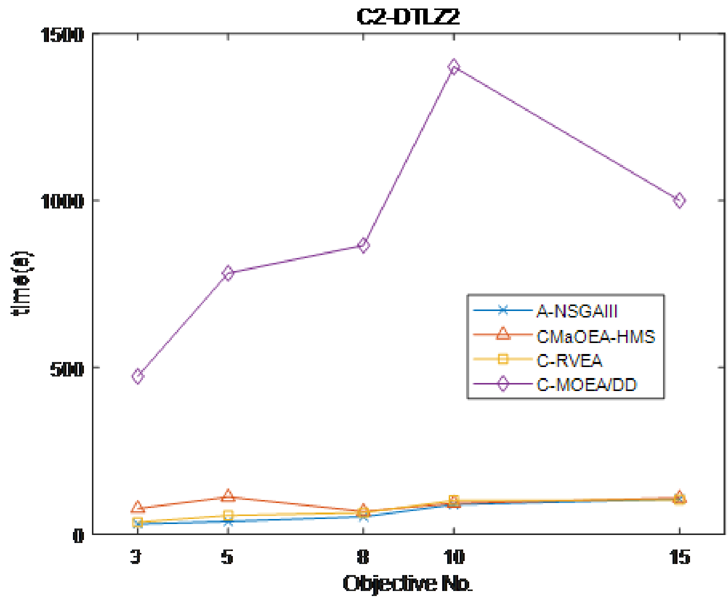


Table 7. Values *α, β, γ* of Four Algorithms

variable	NSGAIII	θ -DEA	IBEA	MaOEA-SDAC
α	0.5108	0.5359	0.5002	0.5001
β	-0.0999	-0.1000	-0.0861	-0.0983
γ	-0.7000	-0.7000	-0.7000	-0.6979

Figure 6 shows the user load distribution curve after and before the optimization of algorithm MaOEA-SDAC. It can be seen that after the optimization, the user load in the peak period decreases somewhat, while the load in the trough period increases somewhat, which can relieve the power tension and improve the load rate.

Figure 7 shows the user load distribution curve before and after optimization of the MaOEA-SDAC algorithms. As can be seen from the figure, MaOEA-SDAC algorithms have achieved good optimization results, such as reducing the load difference during peak and valley periods, cutting the peak and filling the valley, and reducing the load during peak period after the adjustment of electricity price.

Table 8 shows the average electricity consumption of the user before and after the implementation of the stepwise and peak-valley timesharing joint optimization of the four algorithms. In the trough period of 0, the original load was 27.2, and the load obtained by each algorithm was 28.4247, 28.4595, 28.4254 and 28.4097(unit: kW/h), all of which effectively increased the load. At the peak time of 10 o 'clock, the original load was 34.1, and the load obtained by each algorithm was 32.7115, 32.6772, 32.7404 and 32.7295(unit: kW/h), all of which effectively reduced the load rate. It can be concluded from Table 9 that the performance of each algorithm is relatively good.

Figure 6. MaOEA-SDAC before and after load curve optimization

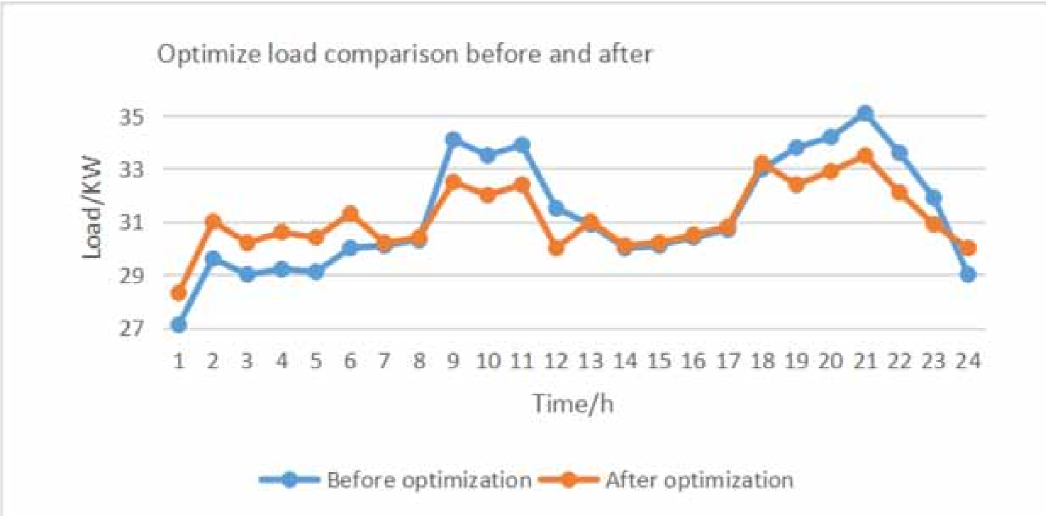


Figure 7. Load curve before and after the electricity price

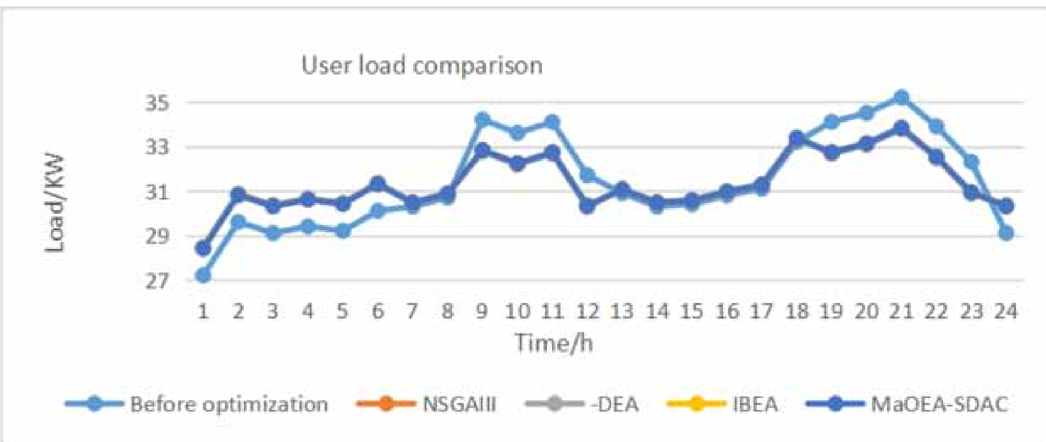


Table 9 shows the user satisfaction before and after the four algorithms optimize the price of electricity. Users' satisfaction with electricity mode *Sm*, users' satisfaction with electricity expense *Sc*, and users' comprehensive satisfaction *So*. The IBEA algorithm gives the highest *Sm*. MaOEA-SDAC obtains the better performance on *Sc* and *So*.

CONCLUSION

Two coevolution strategies are proposed, one is based on space division and Angle culling strategy for high dimensional multi-objective coevolution, the other is a constrained high-dimensional multi-objective coevolution based on hybrid matching selection strategy. The two coevolution strategies

Table 8. Changes in average load before and after electricity price optimization

Period	Actuating preload(kW)	NSGAIII	θ-DEA	IBEA	MaOEA-SDAC
0	27.2	28.4247	28.4595	28.4254	28.4097
1	29.6	30.8247	30.8595	30.8254	30.8097
2	29.1	30.3247	30.3595	30.3254	30.3097
3	29.4	30.6247	30.6595	30.6254	30.6097
4	29.2	30.4247	30.4595	30.4254	30.4097
5	30.1	31.3247	31.3595	31.3254	31.3097
6	30.3	30.4679	30.4951	30.4454	30.4564
7	30.7	30.8679	30.8951	30.8454	30.8564
8	34.2	32.8115	32.7772	32.8404	32.8295
9	33.6	32.2115	32.1772	32.2404	32.2295
10	34.1	32.7115	32.6772	32.7404	32.7295
11	31.7	30.3115	30.2772	30.3404	30.3295
12	30.9	31.0679	31.0951	31.0454	31.0564
13	30.3	30.4679	30.4951	30.4454	30.4564
14	30.4	30.5679	30.5951	30.5454	30.5564
15	30.8	30.9679	30.9951	30.9454	30.9564
16	31.1	31.2679	31.2951	31.2454	31.2564
17	33.2	33.3679	33.3951	33.3454	33.3564
18	34.1	32.7115	32.6772	32.7404	32.7295
19	34.5	33.1115	33.0772	33.1404	33.1295
20	35.2	33.8115	33.7772	33.8404	33.8295
21	33.9	32.5115	32.4772	32.5404	32.5295
22	32.3	30.9115	30.8772	30.9404	30.9295
23	29.1	30.3247	30.3595	30.3254	30.3097

Table 9. Comparison of satisfaction on electricity price

S	NSGAIII	θ-DEA	IBEA	MaOEA-SDAC
S_m	0.9703	0.9693	**0.9709**	0.9707
S_c	1.0264	1.0198	1.0258	**1.0278**
S_0	1.0040	0.9996	1.0039	**1.0054**

perform on DTLZ/WFG benchmark functions, and their IGD and HV values compare with those related competitors. The effectiveness of MAOEA-SDAC and CMaOEA-HMS in solving high-dimensional multi- objective optimization problems has been verified. Finally, the proposed MAOEA-SDAC is employed to a multi-objective model that solves the joint calculation problem of residential ladder and peak-to-valley time-of-use electricity price.

FUNDING

This work was partially supported by National Natural Science Foundation of China (61572074) and National Key Research and Development Program of China (2020YFB1712104).

COMPETING INTERESTS

The authors declare no compting financial or non-financial interests.

REFERENCES

Cheng, R., Jin, Y., Olhofer, M., & Sendhoff, B. (2016). A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, *20*(5), 773–791. doi:10.1109/TEVC.2016.2519378

Cohon, J. (1978). *Multi-Objective Programming and Planning*. Academic Press.

Deb, K., Mohan, M., & Mishra, S. (2003). Towards a quick computation of well spread pareto-optimal solutions. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 222-236). Springer-Verlag. doi:10.1007/3-540-36970-8_16

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. doi:10.1109/4235.996017

Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2002). Scalable multi-objective optimization test problems. In *2002 Congress on Evolutionary Computation* (CEC'02), Honolulu, HI, USA. doi:10.1109/CEC.2002.1007032.10.1109/CEC.2002.1007032

Emmerich, M., Beume, N., & Naujoks, B. (2005). An EMO Algorithm Using the Hyper volume Measure as Selection Criterion. In *Evolutionary Multi-Criterion Optimization* (pp. 62–76). Springer Berlin Heidelberg. doi:10.1007/978-3-540-31880-4_5

Ghoreishi, S. N., Sørensen, J. C., & Jørgensen, B. N. (2015). Comparative study of evolutionary multi-objective optimization algorithms for a non-linear Greenhouse climate control problem. In *IEEE Congress on Evolutionary Computation* (CEC '15)(pp. 1909-1917.). IEEE. doi:10.1109/CEC.2015.7257119

Hiroaki, F., & Akira, O. (2019). *Coverage Enhancement of MOEA/D-M2M for Problems with Difficult-to-Approximate Pareto Front Boundaries*. In *IEEE Congress on Evolutionary Computation* (CEC 2019) Wellington, New Zealand. doi:10.1109/CEC.2019.8790146

Huband, S., Hingston, P., Barone, L. & While, L. (2006). A review of multi-objective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, *10*(5), 477-506. doi:.2005.861417.10.1109/TEVC

Ikeda, K., Kita, H., & Kobayashi, S. (2001). Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal? In *Proceedings of the 2001 Congress on Evolutionary Computation,* Seoul, South Korea. doi:10.1109/CEC.2001.934293

Ishibuchi, H., Tsukamoto, N., & Sakane, Y. (2010). Indicator-based evolutionary algorithm with hyper volume approximation by achievement scalarizing functions. In *Proceedings of the Conference on Genetic and Evolutionary Computation* (GECCO*'10*)(pp. 527-534). doi:10.1145/1830483.1830578

Jain, H., & Deb, K. (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Transactions on Evolutionary Computation*, *18*(4), 602–622. doi:10.1109/TEVC.2013.2281534

Khare, V., Yao, X., & Deb, K. (2003). Performance Scaling of Multi-objective Evolutionary Algorithms. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp.376–390). Springer. doi:10.1007/3-540-36970-8_27

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. doi:10.1109/TEVC.2008.925798

Li, K., Deb, K., Zhang, Q., & Kwong, S. (2015). An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition. *IEEE Transactions on Evolutionary Computation*, *19*(5), 694–716. doi:10.1109/TEVC.2014.2373386

Liu, L., Gu, F., & Zhang, Q. (2014). Decomposition of a multi-objective optimization problem into a number of simple multi-objective sub-problems. *IEEE Transactions on Evolutionary Computation*, *18*(3), 450–455. doi:10.1109/TEVC.2013.2281533

Purshouse, R. C., & Fleming, P. J. (2003). Evolutionary many-objective optimization: an exploratory analysis. In The 2003 Congress on Evolutionary Computation (CEC '03) (pp.2066-2073). ACT. doi:10.1109/CEC.2003.1299927

Veldhuizen, V. D. A. (1999). Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations. *Evolutionary Computation*, *8*(2), 125–147. doi:10.1162/106365600568158 PMID:10843518

Yuan, Y., Xu, H., & Wang, B. (2014). An improved NSGA-III procedure for evolutionary many-objective optimization. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO '14)*(pp. 661–668). Association for Computing Machinery. doi:10.1145/2576768.2598342

Zhang, Q., & Li, H. (2007). MOEA/D: A Multi-objective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, *11*(6), 712–731. doi:10.1109/TEVC.2007.892759

*Wang Hongbo (corresponding author) is an associate professor and M.S. Candidate supervisor at the School of Computer and Communication Engineering, University of Science and Technology Beijing, China. He won 2015 National College Challenge Cup Smart City special competition first prize, Smart iCase Gold Medal team Coaches, the best paper reward in 15th IEEE International Conference on Cognitive Informatics and Cognitive Computing (ICC*ICC 2016, Stanford University), one of the best team winner in hAQUAthon 2017(UK) and the gold reward team Coach in Google Android Mobile Internet Innovation Challenge finals (2017, China). He is focusing on micro-services oriented cognition scheduling and evolutionary computing.*

Wei Huang received his master's degree in computer technology from University of Science and Technology Beijing in June 2022.