

Research Article

An Evolutionary Multiagent Framework for Multiobjective Optimization

Zihui Zhang,^{1,2} Qiaomei Han,² Yanqiang Li,¹ Yong Wang,¹ and Yanjun Shi² 

¹*Institute of Automation, Qilu University of Technology (Shandong Academy of Sciences), Shandong Provincial Key Laboratory of Automotive Electronics Technology, Jinan, CO 250014, China*

²*Mechanical Engineering Department, Dalian University of Technology, Dalian, CO 116024, China*

Correspondence should be addressed to Yanjun Shi; syj@ieee.org

Received 17 September 2019; Revised 1 February 2020; Accepted 2 March 2020; Published 21 April 2020

Academic Editor: Emilio Insfran Pelozo

Copyright © 2020 Zihui Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article proposes an evolutionary multiagent framework of the co-operative co-evolutionary multiobjective model (CCMO-EMAS), specifically for equipment layout optimization in engineering. In this framework, each agent is set in a multiobjective cooperative co-evolutionary mode along with the algorithms and corresponding settings. In each iteration, agents are executed in turn, and each agent optimizes a subpopulation from system decomposition. Additionally, the collaboration mechanism is addressed to build complete solutions and evaluate individuals in the co-operative co-evolutionary algorithm. Each subpopulation is optimized once, and the corresponding agent is evaluated based on the improvement of the system memory. Moreover, the agent team is also evolved through an elite genetic algorithm. Finally, the proposed CCMO-EMAS framework is verified in a multimodule satellite equipment layout problem.

1. Introduction

On the conception of the agent, researchers presented diverse viewpoints according to their research fields, and there is still no uniform definition. Currently, using an agent-based model to solve optimization problems attracts the attentions of an increasing number of researchers [1]. The multiagent system (MAS) is the set of several agent elements which interact with each other under some defined rules or given conditions. Through combining multiagent with the evolutionary algorithms, several researchers proposed agent-based evolutionary algorithms [2].

There are mainly three patterns for agent-based evolutionary algorithm. (1) The MAS is used as the management layer to control the evolutionary procedure. In such methods, agents have their functions, and the algorithm optimization is achieved through coordination and interaction among agent systems. Therefore, when solving different problems, it is necessary to reestablish the multiagent model according to different situations. For instance, Cardon et al. [3] used GA to solve multiobjective job shop

scheduling problems with a multiagent system. (2) The MAS represents the population of the algorithm. The typical method in this field is the multiagent genetic algorithm. As an example, Zhong et al. [4] took each solution in the population as one agent, and all the agents are distributed within the grid in a two-dimension space, with each node only linking with four neighbor nodes. It is equivalent to decompose the solution and searching space, which has a good effect on improving the diversity of the algorithm. This approach is more likely to be regarded as an algorithm rather than a framework. (3) Each agent in MAS represents an algorithm model, where the representative work is the research of Hanna and Cagan [5]. This evolutionary multiagent system (EMAS) is a framework that can integrate many kinds of evolutionary algorithms in MAS and changes the algorithm model by the evolution of agents. EMAS framework adjusts the algorithm model and parameter settings through collaboration and interaction between agents, and obtains a better solution.

In the past decades, the EMAS has been successful in solving engineering design problems, especially in terms of

cooperation and functional combination of algorithms. There have been many representative systems are proposed, such as A-Team [6] and A-Design [7]. However, the main problem of these systems is that an algorithm and its parameters represented by the agent are usually unchanged during the optimization process, so the full potential of each agent cannot be played out. Based on the above research, Landry and Cagan provided a universal framework of EMAS [8] which enhanced the cooperation of agents and further proposed three levels of cooperation between agents, including full cooperation, partial cooperation, and no cooperation. The algorithm parameters are also encoded into the genes of the agent and change with the evolution of the agent. Landry and Cagan [9] made a further study on the EMAS, they proposed a protocol-based EMAS method, which encoded the optional operators of different algorithms into the genes of the agent. Egan et al. [10] applied the EMAS to solve an intricate design in bio-nanotechnology and realized the cooperation between human and evolutionary agent systems by a graphical interface. As referred above, the EMAS has already got excellent achievements for single-objective problems. However, it has some deficiencies in solving multiobjective combinational optimization such as Pareto nondominated ranking and low selecting pressure. The relative references of EMAS for the multiobjective evolutionary model are insufficient, especially for some multiobjective combinational problems that need divide and conquer [11, 12]. Taking multicabin satellite equipment layout problem as an example, the conventional method is to decompose the original problems into several subproblems based on the principle of divide-and-conquer and then to solve them with a multiobjective cooperative co-evolutionary algorithm [13].

The main work of this paper is proposing an evolutionary multiagent framework of the multiobjective cooperative co-evolutionary model (CCMO-EMAS), which is compared with two original multiobjective cooperative co-evolutionary algorithms for an engineering problem, i.e., the multicabin satellite equipment layout problem (MLSELP). The paper is organized as follows. Section 2 provides a brief introduction to EMAS, and the CCMO-EMAS framework is proposed in Section 3. The model of MLSELP is introduced in Section 4. The framework is validated by MLSELP, and optimization results of all the considered algorithms are analyzed and discussed in Section 5. Finally, our future work and recommendation are concluded in Section 6.

2. Related Work

In the description of Cagan [8], EMAS is composed of multiple agents, which is also known as an agent team. The gene of each agent in team represents or describes an algorithm model, including configuration and parameter. The agent team will use multiple algorithm models to solve the same problem, and the solutions will be shared and refined through shared memory and search space of the problem. The structure of the EMAS is shown in Figure 1. It is noteworthy that the agent team is also evolving to improve the algorithm model. Agents in EMAS are encoded with a

binary string, which builds different algorithm models according to the various gene positions of the chromosome. The operation process of EMAS is as follows:

Firstly, each agent needs to be activated before evolution, so that the algorithm represented by the agent is executed. In the process of activation, the algorithm is assigned initial solutions and run a certain number of iterations to generate new solutions based on the parameters in the agent. The initial solutions can be selected from the shared memory or randomly generated. Based on these two sources of initial solution, EMAS defines three levels of cooperation between agents: (1) full cooperation: all the input population will be acquired from the shared memory; (2) partial cooperation: the input population has 50% chance of using initial solutions from the shared memory; and (3) no cooperation: all the input population will all use randomly generated solutions. In EMAS, the shared memory not only provides the initial populations for agents but also plays an external archival role in storing the optimal solution search so far. The objective values of the new solutions will be used to evaluate the agent. After each activation, each agent's fitness value is proportional to the difference between the optimal solution of the new solutions and the shared memory, and then the optimal solution in shared memory is updated. In order to keep the population size constant in the shared memory, EMAS removes the exceeded solutions based on their objective values. If the optimal solution in newly generated solutions is worse than that in shared memory, the agent does not change the fitness value and continues to run a certain number of iterations. Besides, each agent has an attribute called "age" whose initial value is zero. Whenever the algorithm in the agent runs one time, the age of this agent increases one. If the age of the agent reaches specified value, this agent will go through the mutation operation, and its fitness value will decrease at the same time.

Secondly, the agent team will evolve based on their fitness values. In EMAS, the evolutionary operation of agents is similar to that of the binary genetic algorithm, including reproduction, crossover, mutation and selection. The agents with better fitness values are selected as the parent agents and allowed to preserve beneficial genes. Each parent agent is only permitted to reproduce once in each iteration. In the reproduction, the two-parent agents are duplicated, and a single-point crossover is executed to generate two offspring agents.

Meanwhile, the new offspring agents also have 50% probability of mutation to become completely different agents. As referred above, the agent might also mutate inactively when the age of the agent reaches the specified value. After crossover and mutation, the offspring agents are activated and their fitness values are calculated. Agents with the same fitness are further sorted by age, and the smallest one is chosen. This process will repeat until termination conditions are satisfied. In this EMAS, the three levels of cooperative method between agents achieve excellent effects for solving benchmark test functions, but it is easy to fall into local optimum when solving combinational problems with complicated constraints.

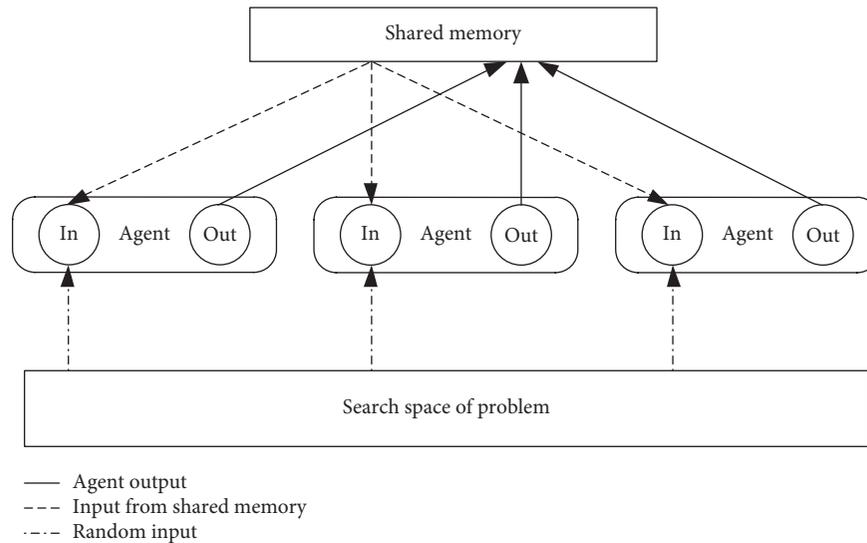


FIGURE 1: Structure of EMAS.

As mentioned above, EMAS is mostly used to solve single-objective optimization problems, but rarely solve multiobjective optimization problems. Also, the framework of EMAS does not consider co-evolutionary algorithms and has difficulty to solve combinational problems with a large number of variables.

3. An Evolutionary Multiagent Framework of the Multiobjective Cooperative Co-Evolutionary Model (CCMO-EMAS)

Compared with conventional heuristic algorithms, cooperative co-evolutionary algorithms (CCEAs) are generally used to solve complex combinational problems with more variables. CCEA is based on the idea of divide and conquer, that is, decomposing the complex problem into several subproblems. In CCEAs, the initial population of system solutions is divided into many populations of subsystem solution based on problem decomposition firstly. At each iteration, a heuristic algorithm optimizes the population of subsystems one after another. To reduce the effects of coupling, the fitness function of each subsystem is the same as that of the system. Therefore, a significant problem occurs about building complete solutions for evaluation of each sub-system, which we also call it a collaboration mechanism. Especially in multiobjective cooperative co-evolutionary algorithms (such as NSCCGA [14], MOCCA [15]), the collaboration mechanism is the critical factor to the performance of algorithm. Meanwhile, the setting of parameters is another problem and plays an essential role in combinational problems. To address these problems, an evolutionary multiagent framework of multiobjective cooperative co-evolutionary algorithms (CCMO-EMAS) is proposed in this study.

3.1. Structure of the CCMO-EMAS. Similar to the structure of EMAS, CCMO-EMAS contains an agent team with many

agents. The structure of CCMO-EMAS is shown in Figure 2. In CCMO-EMAS, each agent can be defined as a multi-objective cooperative co-evolutionary model with the algorithms and their settings. The initial population of the system is decomposed into several subpopulations (i.e., population of subsystems) based on the characteristics of the combinational problem. CCMO-EMAS uses several subsystem memories to store subpopulations and provides the initial solutions for agents. The number of subsystem memories depends on the system decomposition and determines the number of agents. Furthermore, system memory is employed as an external archive to preserve the excellent solutions and provides the representative system solution for collaboration mechanism. In multiobjective combinational problems, nondominated solutions of system are usually selected as excellent solutions.

The gene of agent represents the algorithm type and related parameters, mainly including the setting of collaboration mechanism. With the iteration of CCMO-EMAS, the algorithms and settings will evolve along with the evolution of the agent team, where each agent optimizes a subsystem. In each iterating of CCMO-EMAS, the agents are activated firstly, which is running the algorithm model in the agent. The initial solutions of algorithm have come from different subsystems, and each agent corresponds to a subsystem. Therefore, it is impossible that several agents have the same initial population.

After the activation, the agents are evaluated by their scores and ages, which depends on the degree of improvement solutions of their algorithms. In this study, the degree of improvement solutions is calculated by updating the nondominated solutions in the system memory. After evaluation, the agent team is evolved by the crossover and mutation. Finally, an elite strategy is used for selecting the newly generated agents for realizing the improvement in the multiobjective cooperative co-evolutionary algorithms. The flowchart of CCMO-EMAS is shown in Figure 3.

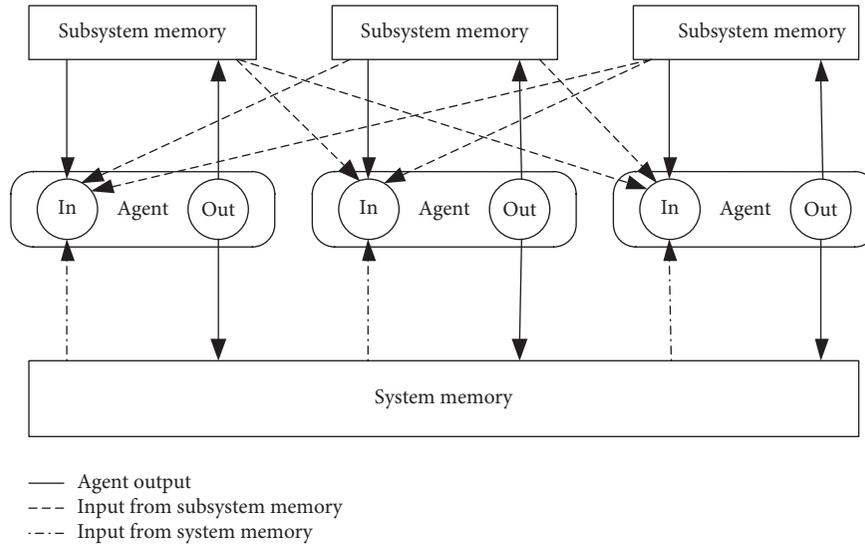


FIGURE 2: Structure of CCMO-EMAS.

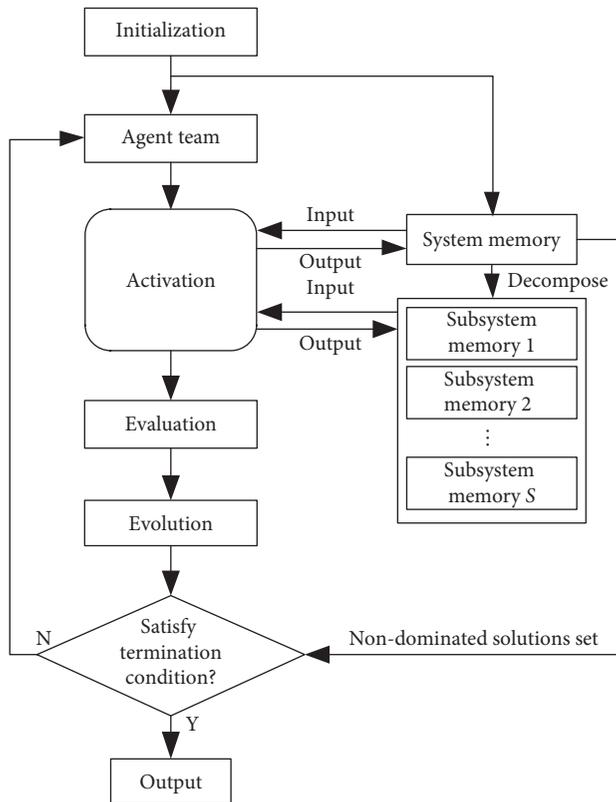


FIGURE 3: Flowchart of CCMO-EMAS.

It is worth noting that the purpose of CCMO-EMAS is to obtain the most suitable multiobjective cooperative co-evolutionary algorithms for a specific type of combinational problems, and the optimal solution of the system is generated to evaluate the agent.

The pseudo code of CCMO-EMAS is shown in the following, as Algorithm 1.

3.2. Agent Definition. In CCMO-EMAS, the agent adopts real number coding to denote the multiobjective cooperative

co-evolutionary model. The chromosome of agent contains five gene positions, which are represented by five real numbers. Therein, the former two positions denote the collaboration mechanism, which contains the subsystem memory selection and the representative selection. As described above, the collaboration mechanism in this framework is addressed to build complete solutions for fitness calculation, which combines the subpopulation with representatives of other subpopulations or system memory. The latter three positions represent the setting of the algorithm in the cooperative co-evolutionary algorithm, which contains algorithm type, crossover factor and mutation factor. The structure of the agent chromosome can be seen in Figure 4.

The specific meaning of each gene position is as follows:

- (i) *Selection of sub-system memory*: this determines which sub-system memory provides the initial population for one agent. When an agent determines one sub-system memory, other agents cannot choose this memory.
- (ii) *Selection of representative*: this determines how to select a representative form for the complete solution. The details will be introduced in Section 3.4.
- (iii) *Algorithm*: it determines which kind of algorithm is chosen as the basic algorithm for the multi-objective cooperative co-evolutionary model. Considering the requirements of the algorithm comparison and the selection constraints of the latter parameter, this paper only gives two optional multi-objective evolutionary algorithms, NSGA-II and NSDE.
- (iv) *Crossover operator*: this assigns the value of the crossover operator (CR) in the algorithm.
- (v) *Mutation operator*: it assigns the value of the mutation operator (F) in the algorithm.

3.3. Activation and Evolution. In CCMO-EMAS, every agent is activated at the beginning of each iteration. In the

Input: $F_u(X)$: multiobjective function; Pop^0 : initial solutions set; S : number of agent team; A^0 : initial agent team; T_{\max} : number of iterations;

Output: optimal agent team A^* ; solutions in system memory Pop^* ;

- (1) Initial $T=0$ and $i=0$;
- (2) Decompose population Pop^0 into Pop_S^T
- (3) Copy Pop^0 into Pop^* ;
- (4) **while** $T < T_{\max}$ **do**
- (5) **for each** $i \in [1, S]$ **do**
- (6) Decode an agent A_S^T into a multi-objective cooperative co-evolutionary algorithm A ;
- (7) Choose a sub-population in Pop_S^T as initial population of algorithm A ;
- (8) Use algorithm A to optimize Pop_S^T based on the collaboration mechanism in A_S^T ;
- (9) Update Pop^* through Pop_S^T ;
- (10) Obtain score and age of agent A_S^T ;
- (11) **end for**
- (12) Use elite genetic algorithm to evolve the agent team A^T based on the score and age;
- (13) Update A^* through A^T ;
- (14) $T = T + 1$;
- (15) **end while**

ALGORITHM 1: CCMO-EMAS.

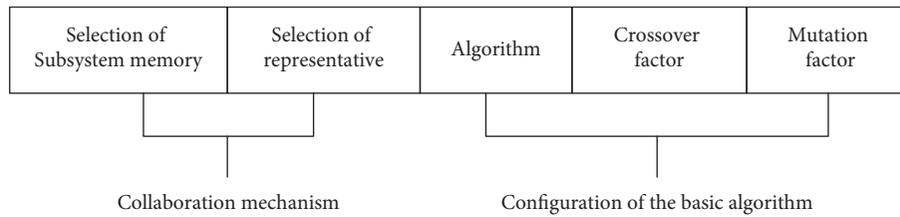


FIGURE 4: Structure of the agent chromosome in CCMO-EMAS.

activation stage, the algorithm represented by the agent is running and optimizing the sub-population in the corresponding sub-systems. The primary function of the activation process is to generate new solutions and update the sub-population of sub-systems. According to the collaboration mechanism, the new solutions and the representative are combined into a complete solution of the system, which is stored in the system memory. Afterwards, the system memory updates and removes the solutions based on non-dominated rank and crowded distance. In this study, the score of the agent is proportional to the ability to make positive changes of non-dominated solutions set in the system memory. The number of replaced solutions in the system memory is the score of an agent.

Sometimes, it is difficult to adequately reflect the pros and cons of agent-based only on the score. As each agent only optimizes the sub-problem in a sub-system, the score also depends mostly on the collaboration mechanism. In multi-objective cooperative co-evolutionary algorithms, the quality of non-dominated solutions in the system memory is also closely related to the representative. Especially when the score of the agent is zero, the agent can reselect a representative to form complete solutions and update the system memory again. The age of the agent is initialized to zero in each iteration. Then the number of age increases one when the agent reselects a representative each time. Once the age

of the agent reaches three, this agent ends activation and gets a bad evaluation. In this case, the multi-objective cooperative co-evolutionary algorithm in agent is considered to be less effective, and will directly perform mutation operation. According to the ages, the agent will increase opportunities to use different representatives, which can improve the cooperative ability of the multi-objective cooperative co-evolutionary algorithm. The flowchart of agent activation is shown in Figure 5.

As referred above, for the score and age of the agent, we need to calculate the nondominated ranking and crowded distance more than once for multiobjective combinational problems. To improve the computational efficiency, an elitist strategy is adopted in CCMO-EMAS, which is to retain a certain number of excellent agents. Meanwhile, the parent individuals are selected from other agents based on their scores and ages, and the selection operator adopts roulette selection. The higher the score and the younger age, the greater the probability of being selected as the parent agent. After that, the parent agents execute the single-point crossover and Gaussian mutation to generate the new agents, which are composed of the next-generation agent team with the elite agents. The next-generation agent team is composed of the new generated agents and the elite agents after crossover and mutation operations.

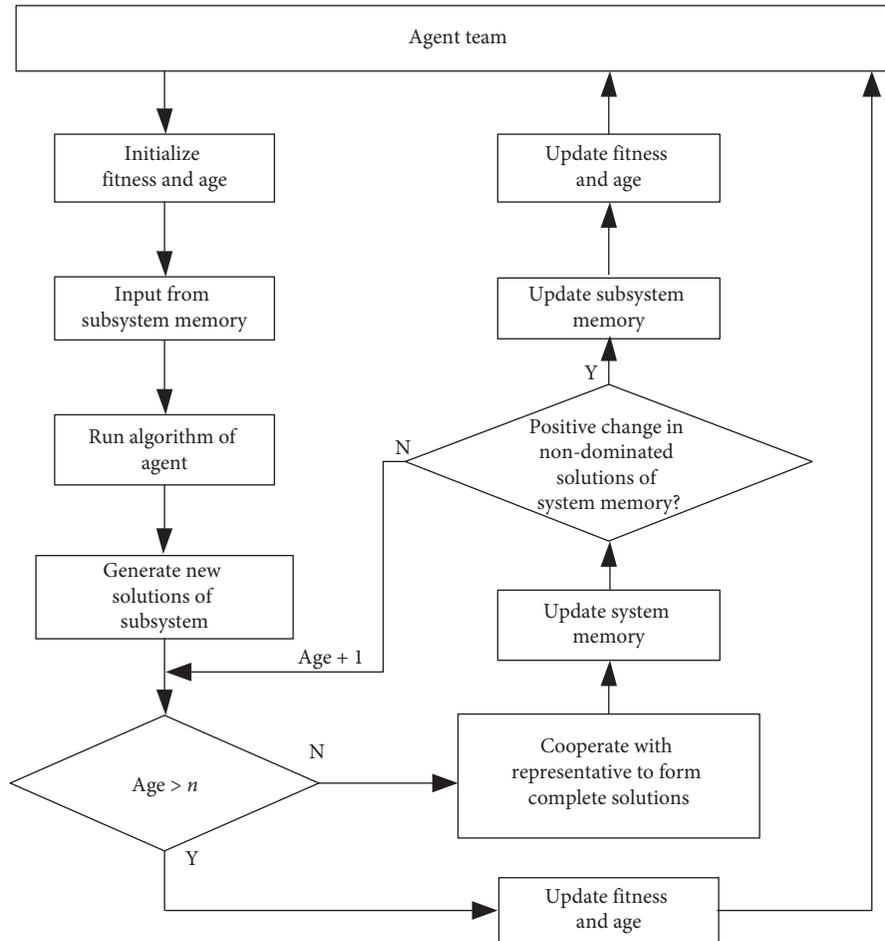


FIGURE 5: Flowchart of activation of the agent.

These steps are repeated until the termination conditions are reached. The repeating iteration for the agent team can continually explore the algorithms space of the multiobjective co-operative co-evolutionary model, so it is beneficial to obtain more applicable multiobjective cooperative co-evolutionary algorithms for multiobjective combinational problems.

3.4. Collaboration Mechanism. As described above, EMAS defined three levels of cooperation between agents, notably including no cooperation and randomly generating solutions in each iteration. For the combinational problems with performance constraints, full cooperation is the better choice than the two other cooperation levels. Especially after the solution has evolved to a certain stage, it is difficult to satisfy the performance constraints by using randomly initialized populations based on partial cooperation or no cooperation. Without the full cooperation between the agents, the qualities of nondominated solutions in the system memory are hard to improve, and the evaluation of the agent will be influenced.

Different from EMAS, multiobjective co-operative co-evolutionary algorithms must employ collaboration mechanism to evaluate the solutions in subsystem. The collaboration mechanism can also be called the collaboration method of complete solutions, which combines the subsystem solution with the representative of excellent solutions

to form system solutions. In CCMO-EMAS, a new cooperation mode is constructed to realize by the collaboration mechanism in a cooperative co-evolution framework.

Agents employ two collaboration methods: the first method is to combine the respective optimal solutions of each subsystem memory into the representative [16], and the second method is to select the representative directly from the system memory [17]. No matter which kind of collaboration methods are, each agent has to cooperate with other agents. In CCMO-EMAS, each collaboration method also has two selections modes, which are best selection and random selection, respectively. Thus, there are four kinds of collaborators selections modes of representative for agents. The four selections modes are listed as follows:

- (i) Best collaboration of subsystem memory (BC): this refers to the random selection of the representative from the Pareto optimal solutions in each subsystem memory to form a complete solution.
- (ii) Random collaboration of subsystem memory (RC): this refers to the random selection operation of the representative from each subsystem memory to form a complete solution.
- (iii) Best selection of system memory (BA): this refers to the random selection of the complete representative

from the Pareto optimal solutions in system memory.

- (iv) Random selection of system memory (RA): it refers to the random selection of the complete representative in system memory.

As the research's emphasis of multiobjective cooperative co-evolutionary algorithms, each kind of selection of representative has its characteristics at different stages of evolution.

4. Multicabin Satellite Equipment Layout Case

This section describes the details about the multicabin satellite equipment layout problem and related configurations. The mathematical model is given in Section 4.1. The Engineering requirements and of the algorithm configurations are proposed in Section 4.2. The metrics of performance are detailed in Section 4.3.

4.1. Problem Description. The multimodule satellite equipment layout problem is a multiobjective engineering problem with the features of nonlinear, noncontinuous, and multimodal. This case is cited from [18], which updated the situation of simplified international communication satellite [19]. In this problem, a set of independent components $X = \{X_i | i \in I = 1, 2, \dots, N\}$ will be layout on the support surfaces of the satellite, where N is the number of components. The design variable of the i th component is $X_i = \{x_i, y_i, z_i, \alpha_i | S\}$, where x_i, y_i, z_i are centroid coordinates, α_i is the angle between the long side of the cuboid's bottom surface and x -axis, S is the serial number of

supporting surfaces. In this case, $\alpha_i = 0$, namely, all the cuboids are the orthogonal layout. A simplified layout scheme of the multicabin satellite is shown in Figure 6, where $oxyz$ is the reference coordinates system, $o'x'y'z'$ is the coordinate system of the multicabin satellite, and $o''x''y''z''$ is the coordinate system of the components. The transformation matrix of these three coordinate systems can be obtained in [20].

The mathematical model and relative formulas of this multiobjective optimization are shown as follows:

$$\begin{aligned} \min F(X) &= \{F_u(X) | u = 1, 2, 3\}, \\ \text{s.t. } g(X) &\leq 0, \end{aligned} \quad (1)$$

where $F(X)$ indicates three objective functions of the multimodule satellite equipment layout, $F_u(X)$ is the u th objective function. $u = 1, 2, 3$, which represents the moment of inertia, the inertia angle, and centroid distance, respectively.

$$\begin{aligned} F_1(X) &= |J_{x'}(X)| + |J_{y'}(X)| + |J_{z'}(X)|, \\ F_2(X) &= |\theta_{x'}(X)| + |\theta_{y'}(X)| + |\theta_{z'}(X)|, \\ F_3(X) &= |x_c - x_h|(X) + |y_c - y_h|(X), \end{aligned} \quad (2)$$

where $J_{x'}(X), J_{y'}(X)$ and $J_{z'}(X)$ are the moment of inertia of each axis (x, y, z) in the reference coordinate system, respectively. Also, $\theta_{x'}(X), \theta_{y'}(X)$ and $\theta_{z'}(X)$ are the angles between the principal axis of inertia and the axes of the cabin. Besides, $X_c = \{x_c, y_c, z_c\}$ and $X_h = \{x_h, y_h, z_h\}$ are the calculating centroid and expectation centroid of the satellite module system. The formulas of the moment of inertia are shown as follows:

$$\begin{aligned} J_{x'}(X) &= \sum_{i=0}^N (J_{x''_i} \cdot \cos^2 \alpha_i^2 + J_{y''_i} \cdot \sin^2 \alpha_i^2) + \sum_{i=0}^N m_i (y_i^2 + z_i^2) - (y_c^2 + z_c^2) \sum_{i=0}^N m_i, \\ J_{y'}(X) &= \sum_{i=0}^N (J_{y''_i} \cdot \cos^2 \alpha_i^2 + J_{x''_i} \cdot \sin^2 \alpha_i^2) + \sum_{i=0}^N m_i (x_i^2 + z_i^2) - (x_c^2 + z_c^2) \sum_{i=0}^N m_i, \\ J_{z'}(X) &= \sum_{i=0}^N J_{z''_i} + \sum_{i=0}^N m_i (x_i^2 + y_i^2) - (x_c^2 + y_c^2) \sum_{i=0}^N m_i, \end{aligned} \quad (3)$$

where $J_{x''_i}, J_{y''_i}$, and $J_{z''_i}$ are the moments of inertia of each component. Noteworthy, the formulas for cylinder and cuboid are different. The formulas of the cylinder are shown below:

$$\begin{aligned} J_{x''_i} &= \frac{m_i}{12} (3r_{i1}^2 + h_i^2), \\ J_{y''_i} &= \frac{m_i}{12} (3r_i^2 + h_i^2), \\ J_{z''_i} &= \frac{1}{2} m_i r_i^2. \end{aligned} \quad (4)$$

The formulas of the cuboid are similar as follows:

$$\begin{aligned} J_{x''_i} &= \frac{1}{12} m_i (b_i^2 + h_i^2), \\ J_{y''_i} &= \frac{1}{12} m_i (a_i^2 + h_i^2), \\ J_{z''_i} &= \frac{1}{12} m_i (a_i^2 + b_i^2). \end{aligned} \quad (5)$$

The formulas of the calculating centroid are shown as below:

$$\begin{aligned} x_c &= \frac{\sum_{i=0}^N m_i x_i}{\sum_{i=0}^N m_i}, \\ y_c &= \frac{\sum_{i=0}^N m_i y_i}{\sum_{i=0}^N m_i}, \\ z_c &= \frac{\sum_{i=0}^N m_i z_i}{\sum_{i=0}^N m_i}. \end{aligned} \quad (6)$$

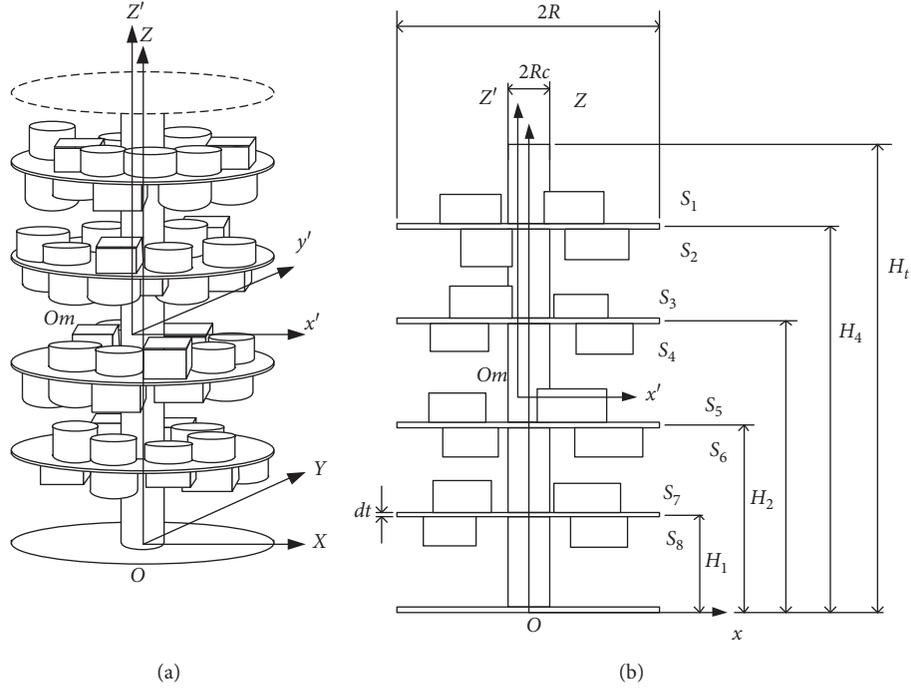


FIGURE 6: Schematic diagram of multicabin satellite equipment. (a) Schematic diagram of the 3D layout. (b) Schematic diagram of the 2D layout.

The formulas of the inertia angle are shown below:

$$\begin{aligned}\theta_{x'}(X) &= \frac{\arctan(2J_{x'y'}(X)/(J_{x'}(X) - J_{y'}(X)))}{2}, \\ \theta_{y'}(X) &= \frac{\arctan(2J_{x'z'}(X)/J_{x'}(X) - J_{z'}(X))}{2}, \\ \theta_{z'}(X) &= \frac{\arctan(2J_{y'z'}(X)/J_{y'}(X) - J_{z'}(X))}{2},\end{aligned}\quad (7)$$

where $J_{x'y'}(X)$, $J_{x'z'}(X)$, and $J_{y'z'}(X)$ are generated by the inertia, and the formulas are shown as follows:

$$\begin{aligned}J_{x'y'}(X) &= \sum_{i=0}^N \left[m_i x_i y_i + \frac{J_{x''i} + m_i (y_i^2 + z_i^2) - J_{y''i} - m_i (x_i^2 + z_i^2)}{2} \cdot \sin 2\alpha_i \right] - x_c y_c \sum_{i=0}^N m_i, \\ J_{x'z'}(X) &= \sum_{i=0}^N m_i x_i z_i - x_c z_c \sum_{i=0}^N m_i, \\ J_{y'z'}(X) &= \sum_{i=0}^N m_i y_i z_i - y_c z_c \sum_{i=0}^N m_i.\end{aligned}\quad (8)$$

The nonoverlapping constraint is shown as follows:

$$g(X) = \sum_{i=0}^{N-1} \sum_{j=i+1}^N \Delta V_{ij}(X), \quad (9)$$

where $\Delta V_{ij}(X)$ is the overlap volume of any two components i and j , as well as component and the bulkhead of the cabin, and $i \neq j$, $i, j \in N$.

The detailed geometry of the satellite cabin and components can be seen in [19].

4.2. Engineering Requirements and Algorithm Configurations. The multicabin satellite is composed of two cabins and eight bearing surfaces, which need to contain 120 components. The components are simplified to 72 cylinders and 48

cuboids in order to reduce the calculation. All components are required to be placed on each bearing surface, and cuboids are placed orthogonally.

The technical specifications for the satellite cabin system are shown as below:

- (i) The sum of the moment of inertia $I = |J_{x'}(X)| + |J_{y'}(X)| + |J_{z'}(X)|$ should be less than the allowable values, which is 2030 kg·m².
- (ii) The sum of the inertial angle $\theta = |\theta_{x'}(X)| + |\theta_{y'}(X)| + |\theta_{z'}(X)|$ should be less than the allowable values, which is 0.09 rad.
- (iii) The distance between the expected centroid and calculating centroid (centroid distance) $\Delta X = |X_h - X_c|$ should be less than the allowable value, which is 6 mm.
- (iv) To ensure that none of the components (including bulkhead of the satellite) interferes $\Delta V = 0$ mm.

The setting of the CCMO-EMAS is shown as follows. The number of system memories is one, and the number of subsystem memories is eight which is the same as the agents. The size of all memories is 200. CCMO-EMAS uses NSDE and NSGA-II as two basic algorithms in the multiobjective cooperative co-evolutionary model. The iteration number of the CCMO-EMAS is 25, while the algorithm in each agent runs 50 generations, and the total generations are 10000. The proportion of elite agents is 10%.

Two multiobjective cooperative co-evolutionary algorithms named NSCCGA and NSCCDE are used to compare with CCMO-EMAS, and the algorithm configurations are similar. The algorithm population is decomposed into eight subpopulations, and the population size is 200 too. The iteration number of the NSCCDE and NSCCGA is also 25, while the algorithm in each subpopulation runs 50 generations, and the total generations are 10000 too.

Furthermore, a non-co-operative algorithm named NSGA-II is also used to solve these problems. The population size is 200, and the total generations are 10000.

4.3. Metrics of Performance. In this paper, the multiobjective combinational problem of satellite cabin has three objectives, so at least three performance indicators are required to evaluate the performance of algorithms. It is a common practice at present, which is widely applied in the multi-objective optimization literature. These three performance indicators are shown as follows [21]:

- (i) *Convergence Performance Indicator.* Generation Distance (GD) is the average distance from the obtained Pareto front by the algorithms to the real Pareto optimal front. The smaller the value of GD, the better the convergence of the Pareto optimal solutions. GD reflects the exploitation ability of the algorithms.

$$GD = \left(\frac{1}{n} \sum_{i=1}^n d_i^2 \right)^{1/2}, \quad (10)$$

where n is the number of solutions of the obtained Pareto front, d_i is the minimum value of the Euclidean distance between the obtained Pareto front and the real Pareto front.

- (ii) *Distributed Performance Indicator.* Spacing (SP) is used to evaluate the distribution of the Pareto optimal solution set. The smaller the value of SP, the more uniform the solution on the obtained Pareto front. SP indicates the ability to maintain the diversity of the algorithms.

$$SP = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\bar{d} - d_i)^2}, \quad (11)$$

where $d_i = \min_{k=1}^m |f_k^i - f_k^j|$, $i, j = 1, 2, \dots, n$, n is the number of the Pareto solution set, m is the number of objective functions, and \bar{d} is the mean of d_i .

- (iii) *Dispersion Performance Indicator.* Maximum Spread (MS) is used to evaluate the coverage performance of obtaining nondominated solutions in the real Pareto optimal front. The larger the value of MS, the better the spread of obtained nondominated solutions. MS reveals the scope of exploration in the nondominated solutions.

$$MS = \sqrt{\frac{1}{n} \sum_{i=1}^n \left\{ \frac{[\min(f_i^{\max}, F_i^{\max}) - \max(f_i^{\min}, F_i^{\min})]}{(F_i^{\max} - F_i^{\min})} \right\}^2}, \quad (12)$$

where n is the number of objectives, f_i^{\max} and f_i^{\min} are the maximum and minimum values of the i th objective in the obtained Pareto front, respectively. F_i^{\max} and F_i^{\min} are the maximum and minimum values of the i th objective of the real Pareto front, respectively.

Because the real Pareto optimal front of the practical engineering problems is unknown, we integrate all the Pareto fronts of the various algorithms to obtain an approximate optimal front as the reference Pareto front (right Pareto front) for evaluation [22].

5. Results and Discussions

In this section, the multi-cabin satellite equipment layout problem is solved by CCMO-EMAS and other three algorithms, and then the results are analyzed and discussed.

5.1. Result Analysis of CCMO-EMAS. To show the position of Pareto front more clearly, a reference basis point (RBS) is taken as a basis point in the solution space. RBS is the solution that meets the noninterference constraints and the allowable values of the design objectives, which are moment of inertia, inertia angle, and centroid distance (i.e., F_1 equals 2030 kg·m², F_2 equals 0.09 rad, and F_3 equals 6 mm, respectively). As shown in Figure 7, the Pareto front obtained by CCMO-EMAS is far from the RBS, so it meets the

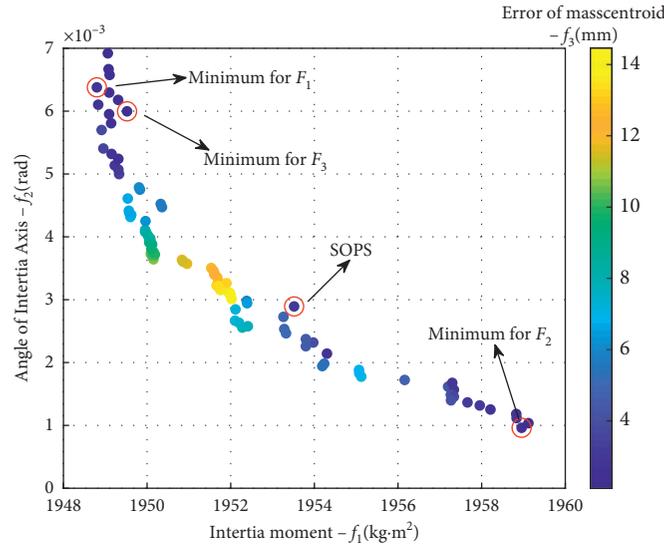


FIGURE 7: 2D diagram of the Pareto front.

engineering allowable value. In summary, the Pareto solutions meet the allowable value should be better than the RBS.

To further analyze the shape and the distribution of the Pareto front obtained by CCMO-EMAS, a 2D figure of the Pareto front is obtained in Figure 7. Therein, the value of centroid distance of F_3 is smaller, and the color becomes deeper when it moves into the downward position. The three objectives of the Pareto front can be seen in Table 1 and Figure 8. It can be seen in Figure 7 that the minimum F_1 and F_2 in the Pareto optimal solutions are placed at both ends of the Pareto front. Besides, the minimum F_3 in the Pareto front has the deepest color. It illustrates that the Pareto front obtained by CCMO-EMAS has a balanced distribution and accords with the character of the Pareto front. A balanced solution of the three objectives is selected based on practical experiences (selected optimal Pareto solution, SOPS), and the 2D layout scheme of the SOPS solutions can be seen in Figure 9.

5.2. Comparison and Analysis of Different Algorithms. The computing results of CCMO-EMAS, NSCCDE, NSCCGA, and NSGA-II are compared in the following contents. It is shown as Figure 10 and Table 2; the mean value of convergence performance of CCMO-EMAS ranks first in the four algorithms. NSCCDE, NSCCGA, and NSGA-II rank the second, third, and the last, respectively. To be specific, the mean value of convergence performance of CCMO-EMAS is 14.34% higher than NSCCDE, 300% higher than NSCCGA and 500% higher than NSGA-II. The mean values of the distributed performance are listed in order: CCMO-EMAS, NSCCGA, NSGA-II, and NSCCDE. The mean of the distributed performance of CCMO-EMAS is 53.16% higher than NSCCGA, 97.18% higher than NSGA-II, and 188.64% higher than NSCCDE. The mean of dispersion performance of CCMO-EMAS is 26.82% higher than NSGA-II, 56.33% higher than NSCCGA, and 106.28% higher than NSCCDE.

As mention above, the convergence, distributed, and dispersion performance of CCMO-EMAS are all ranked first in the four algorithms, which show that CCMO-EMAS has advantages in the following aspects. Firstly, CCMO-EMAS can use multiple algorithms, and their parameters are adaptive. According to no free lunch theory, various algorithms have their advantages. Therefore, the approaches simultaneously use two algorithms have a better performance than that using one of the algorithms. Secondly, CCMO-EMAS employs a system memory as an external archive to save the nondominated solutions that have been searched. It continually updates this archive, which is beneficial for improving the performance of the Pareto front. Thirdly, CCMO-EMAS adopts two kinds of collaboration mechanisms consisting of four selections of representative. It can also reselect the representative in the activation process of the agent, which equates to using several representatives and helps to improve the dispersion and distributed performance.

However, CCMO-EMAS still has some shortcomings. The main shortcoming is that computing time is much larger than the other algorithms. The average computing time of CCMO-EMAS is 845% longer than NSGA-II, 600% longer than NSCCDE, and 373% longer than NSCCGA. This problem needs to be solved in future research.

5.3. Comparison and Analysis of Collaboration Methods. In order to study the effect of different collaboration methods, each kind of representative selection of collaboration methods in CCMO-EMAS is compared and analyzed. The four representative selections of collaboration methods are the best selection (BA), best collaboration (BC), random selection (RA), and random collaboration (RC), respectively. Moreover, CCMO-EMAS uses one of the collaboration methods to solve the problem and compares and analyzes the results.

As seen in Figure 11 and Table 3, the convergence performance GD of BA ranks the first, BC ranks the second,

TABLE 1: The comparison between RBS and the SOPS of CCOMS-EMAS.

Objective function	Objective function value of representative points in Pareto front				
	RBS	SOPS	Optimal F_1	Optimal F_2	Optimal F_3
Rotational inertias $F_1/\text{kg}\cdot\text{m}^2$	2030.000	1953.515	1948.800	1958.955	1949.521
Inertia angle F_2/rad	0.090	0.014	0.064	0.009	0.059
Centroid distance F_3/mm	6.000	3.036	2.812	2.598	2.104
Noninterference constraint $\Delta V/\text{mm}^3$	0	0	0	0	0

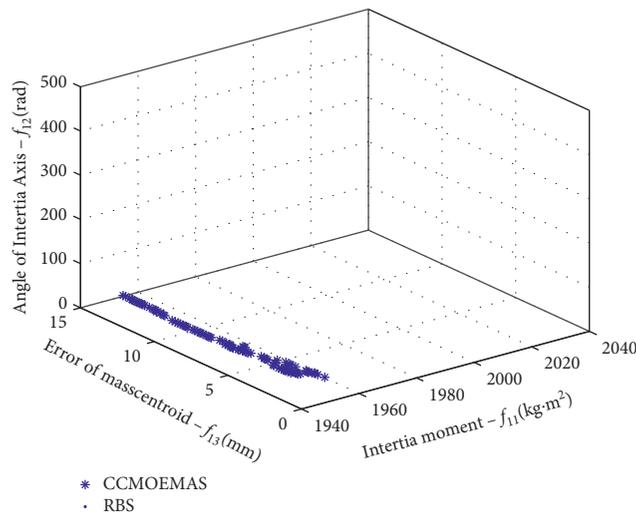


FIGURE 8: 3D diagram of the Pareto front.

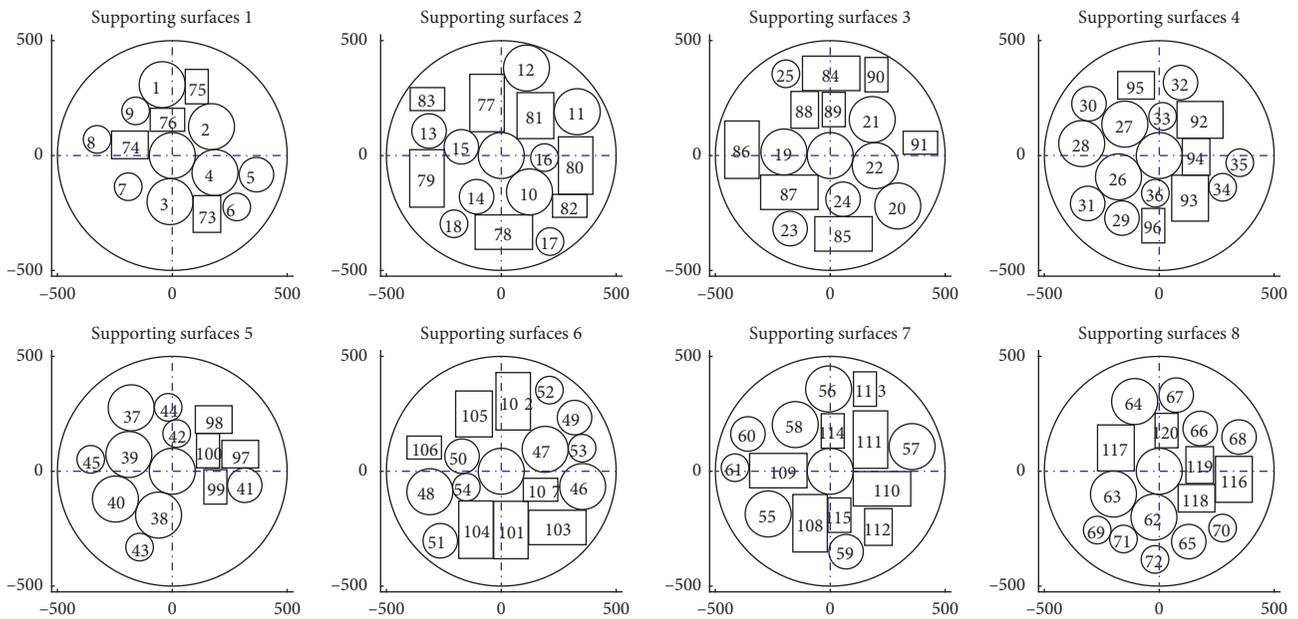


FIGURE 9: 2D Layout scheme of SOPS.

RA ranks the third, and RC ranks the last. The mean value of convergence performance GD of BA is 13.18% higher than BC, 17.41% higher than RA, and 78.03% higher than RC. It can be seen that the convergence performances of BA and BC are better, which illustrates that the optimal selection of

representative is helpful to the convergence performance of the algorithm. The convergence performance of RC has fallen far behind the other three collaboration methods.

The distributed performance SP of these four collaboration methods is listed in the following order: RA, BA, BC,

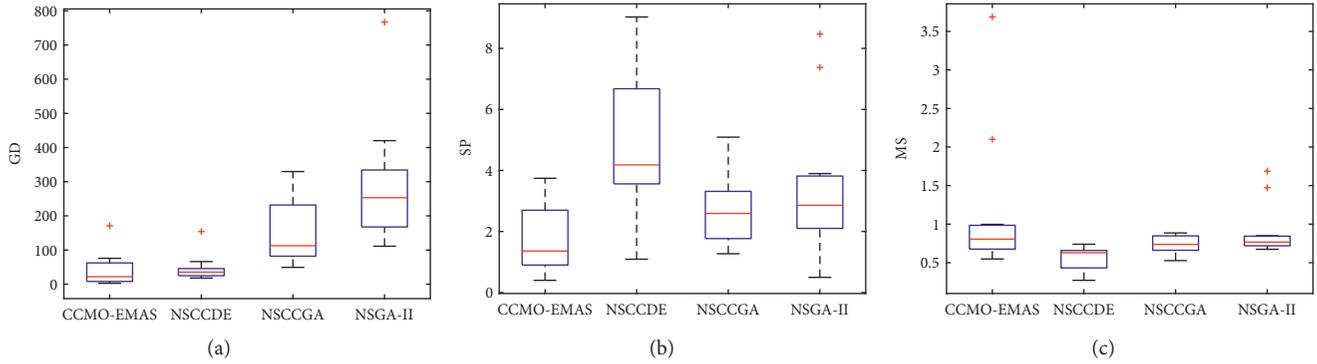


FIGURE 10: Box plots of four algorithms concerning the metrics of (a) GD, (b) SP and (c) MS.

TABLE 2: Mean value of performance indicators of the four algorithms.

	CCMO-EMAS	NSCCDE	NSCCGA	NSGA-II
Convergence performance indicator GD	41.022	46.905	164.500	286.284
Distributed performance indicator SP	1.770	5.109	2.711	3.490
Dispersion performance indicator MS	1.149	0.557	0.735	0.906
Computing time (s)	$9.265e + 3$	$1.530e + 3$	$2.482e + 3$	$1.096e + 3$

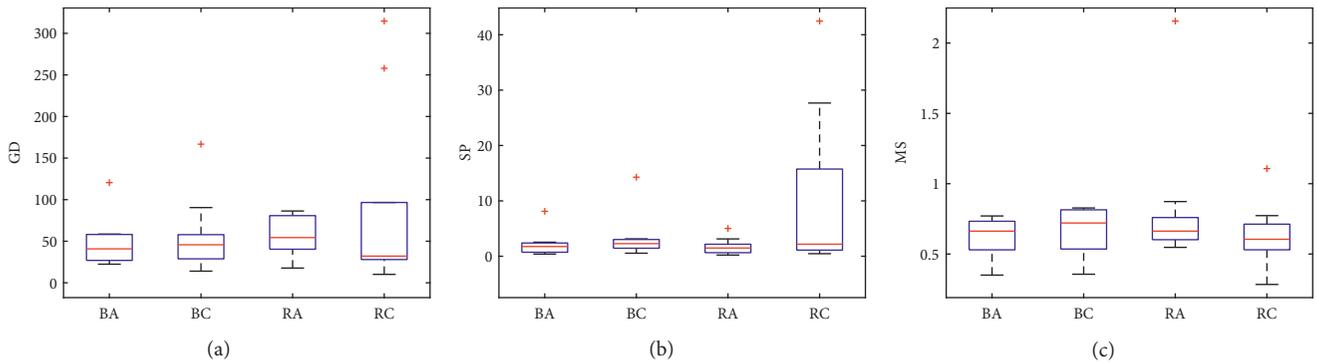


FIGURE 11: Box plots of four representatives concerning the metrics of (a) GD, (b) SP, and (c) MS.

TABLE 3: Mean value of performance indicator of the four selections of the representatives.

	BA	BC	RA	RC
Convergence performance indicator GD	48.394	54.772	56.820	86.156
Distributed performance indicator SP	2.152	3.262	1.723	10.776
Dispersion performance indicator MS	0.628	0.822	0.677	0.624
Computing time (s)	$1.906e + 003$	$2.078e + 003$	$2.119e + 003$	$2.496e + 003$

and RC. The average value of distributed performance SP of RA is 25% higher than BA, 89.32% higher than BC, and 525% higher than RC. It can be seen that the distributed performances of RA and BA outperform others so that the complete representative selection from the system memory has a positive effect on the distributed performance of algorithm. Meanwhile, the distributed performance of RC is far worse than mainly those of other three collaboration methods.

In terms of the dispersion performance MS, BC ranks the first, RA ranks the second, BA ranks the third, and RC ranks the last. The mean value of the dispersion performance MS of BC is 21.42% higher than that of RA, 30.89% higher than BA,

and 31.73% higher than RC. It can be indicated that the dispersion performance of BC is the best. The dispersion performances of the other three selections are nearly the same.

The computing times of all four methods are also very close to each other.

5.4. Analysis of Problem Characteristics. The previous contents have already compared and analyzed the different algorithms and collaboration methods in multiobjective cooperative co-evolutionary models of CCMO-EMAS. To further study the characteristics of multiobjective

cooperative co-evolutionary models in a specific multi-objective combinational problem; this section analyzes the occurrence frequency of different basic algorithms and collaboration methods in different iteration stages. The purpose is to build the targeted multiobjective cooperative co-evolutionary model for a specific problem in the future.

As can be seen in Figure 12, in the earlier evolutionary stage of the agents (the former five iterations of CCMO-EMAS), NSDE is used more often than NSGA-II. This situation illustrates that NSDE has more advantages in the earlier stage for this problem. It can be concluded from the computing results of the previous contents that NSDE performs better than NSGA-II in improving convergence. In the earlier stage of evolution, convergence is a general requirement, and the agent tends to use NSDE as the underlying algorithm more frequently. In the middle and later evolutionary stages of the agents (the last five iterations of CCMO-EMAS), NSGA-II used more often than NSDE. In particular, the agents rarely use the NSDE algorithm in the last 5–10 generations, because the convergence rate of the algorithm began to drop sharply when it converged to a certain degree. To better explore the solution space, the agent requires algorithms with better dispersion and distributed performance. The NSGA-II is also mainly used in the 10–25 generations of agents' evolution. The algorithm selection of CCMO-EMAS is relatively simple because only two basic algorithms are used. In future research, more algorithms are planned to add.

Another research emphasis is the occurrence frequency of collaboration methods. CCMO-EMAS employs different representative selections of collaboration methods in different stages and makes various effects. As can be seen in Figure 13, in the earlier evolutionary stage of the agents (the former five iterations of CCMO-EMAS), BC and RA are used more often than BA. In the earlier iteration of the agents, the algorithms do not converge to a particular degree, so it is necessary to explore each direction of solution space. At this stage, the quality of the nondominated solutions in system memory is poor and is still in the accumulation stage. BA is beneficial to the convergence of algorithms, which is dependent on the solution quality in the system memory. In the middle evolutionary stage of the agents (5–20 iterations of CCMO-EMAS), BA and RA are the two most frequently used because they are archive-based collaboration methods. The BC is also used to improve the dispersion of algorithms. RC is still rarely used in this period. In the later evolutionary stage of the agents (20–25 iterations of CCMO-EMAS), BC and RA are the two most used. It proves that the dispersion and distributed performance have been the main objectives rather than the convergence performance in this stage. BA and RC are less chosen in this period. RC is the worst in the four collaboration methods and rarely used.

To sum up, it is appropriate to use different basic algorithms and collaboration methods in different iteration stage for a specific class of multiobjective combinational problem.

6. Conclusions and Future Work

The primary contribution of this study is to propose an evolutionary multiagent framework of the multiobjective cooperative co-evolutionary model. In this framework of CCMO-EMAS, each agent represents a multiobjective cooperative co-

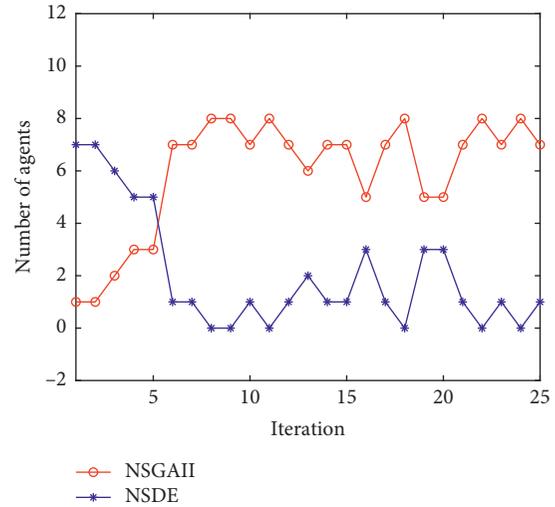


FIGURE 12: Frequency of different algorithms.

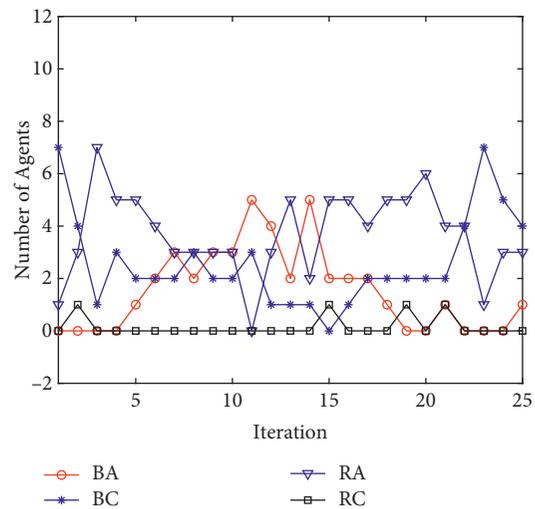


FIGURE 13: Frequency of different collaboration mechanisms.

evolutionary algorithm. With the iteration of the CCMO-EMAS, the algorithm will evolve with the evolution of agents. In the different evolutionary stage of agents, the different basic algorithms and collaboration methods are applied based on the characteristics of specific problem. CCMO-EMAS is used to solve a complex multiobjective problem of MLSELP. Through analysis and comparison, the proposed framework can obtain better results than the other three algorithms.

Through analysis and discussion, CCMO-EMAS has many advantages for multiobjective combinational problems. Firstly, CCMO-EMAS can obtain high-quality solutions that do not depend on specific algorithms or adjusting the parameters. Secondly, as a learning tool, CCMO-EMAS has the potential to guide the evolution process. Finally, CCMO-EMAS deals with flexibility and validity when handling the variable design space. The main disadvantage of CCMO-EMAS is a long calculation time. To improve computational efficiency, the parallel computing framework will be introduced into CCMO-EMAS in the next study.

TABLE 4: Dimensions and masses of components in Section 5.

No	Mass (kg)	Dimensions (mm)			Surface	
		a_i	b_i	r_i		
1	23.56	—	—	100	250	S1
2	23.56	—	—	100	250	S1
3	23.56	—	—	100	250	S1
4	23.56	—	—	100	250	S1
5	7.95	—	—	75	150	S1
6	5.09	—	—	60	150	S1
7	5.09	—	—	60	150	S1
8	5.09	—	—	60	150	S1
9	5.09	—	—	60	150	S1
10	18.85	—	—	100	200	S2
11	15.08	—	—	100	160	S2
12	15.08	—	—	100	160	S2
13	8.48	—	—	75	160	S2
14	7.95	—	—	75	150	S2
15	7.95	—	—	75	150	S2
16	5.09	—	—	60	150	S2
17	5.09	—	—	60	150	S2
18	5.09	—	—	60	150	S2
19	23.56	—	—	100	250	S3
20	18.85	—	—	100	200	S3
21	18.85	—	—	100	200	S3
22	15.08	—	—	100	160	S3
23	8.48	—	—	75	160	S3
24	7.95	—	—	75	150	S3
25	5.09	—	—	60	150	S3
26	23.56	—	—	100	250	S4
27	23.56	—	—	100	250	S4
28	23.56	—	—	100	250	S4
29	8.48	—	—	75	160	S4
30	8.48	—	—	75	160	S4
31	7.95	—	—	75	150	S4
32	7.95	—	—	75	150	S4
33	5.09	—	—	60	150	S4
34	5.09	—	—	60	150	S4
35	5.09	—	—	60	150	S4
36	5.09	—	—	60	150	S4
37	23.56	—	—	100	250	S5
38	23.56	—	—	100	250	S5
39	23.56	—	—	100	250	S5
40	23.56	—	—	100	250	S5
41	7.95	—	—	75	150	S5
42	5.09	—	—	60	150	S5
43	5.09	—	—	60	150	S5
44	5.09	—	—	60	150	S5
45	5.09	—	—	60	150	S5
46	18.85	—	—	100	200	S6
47	15.08	—	—	100	160	S6
48	15.08	—	—	100	160	S6
49	8.48	—	—	75	160	S6
50	7.95	—	—	75	150	S6
51	7.95	—	—	75	150	S6
52	5.09	—	—	60	150	S6
53	5.09	—	—	60	150	S6
54	5.09	—	—	60	150	S6
55	23.56	—	—	100	250	S7
56	18.85	—	—	100	200	S7
57	18.85	—	—	100	200	S7
58	15.08	—	—	100	160	S7
59	8.48	—	—	75	160	S7

TABLE 4: Continued.

No	Mass (kg)	Dimensions (mm)			Surface	
		a_i	b_i	r_i		
60	7.95	—	—	75	150	S7
61	5.09	—	—	60	150	S7
62	23.56	—	—	100	250	S8
63	23.56	—	—	100	250	S8
64	23.56	—	—	100	250	S8
65	8.48	—	—	75	160	S8
66	8.48	—	—	75	160	S8
67	7.95	—	—	75	150	S8
68	7.95	—	—	75	150	S8
69	5.09	—	—	60	150	S8
70	5.09	—	—	60	150	S8
71	5.09	—	—	60	150	S8
72	5.09	—	—	60	150	S8
73	8.64	160	120	—	150	S1
74	8.64	160	120	—	150	S1
75	5.4	150	100	—	120	S1
76	5.4	150	100	—	120	S1
77	28.13	250	150	—	250	S2
78	28.13	250	150	—	250	S2
79	28.13	250	150	—	250	S2
80	28.13	250	150	—	250	S2
81	19.2	200	160	—	200	S2
82	5.4	150	100	—	120	S2
83	5.4	150	100	—	120	S2
84	28.13	250	150	—	250	S3
85	28.13	250	150	—	250	S3
86	28.13	250	150	—	250	S3
87	28.13	250	150	—	250	S3
88	8.64	160	120	—	150	S3
89	5.4	150	100	—	120	S3
90	5.4	150	100	—	120	S3
91	5.4	150	100	—	120	S3
92	19.2	200	160	—	200	S4
93	19.2	200	160	—	200	S4
94	15.36	160	120	—	250	S4
95	15.36	160	120	—	250	S4
96	5.4	150	100	—	120	S4
97	8.64	160	120	—	150	S5
98	8.64	160	120	—	150	S5
99	5.4	150	100	—	120	S5
100	5.4	150	100	—	120	S5
101	28.13	250	150	—	250	S6
102	28.13	250	150	—	250	S6
103	28.13	250	150	—	250	S6
104	28.13	250	150	—	250	S6
105	19.2	200	160	—	200	S6
106	5.4	150	100	—	120	S6
107	5.4	150	100	—	120	S6
108	28.13	250	150	—	250	S7
109	28.13	250	150	—	250	S7
110	28.13	250	150	—	250	S7
111	28.13	250	150	—	250	S7
112	8.64	160	120	—	150	S7
113	5.4	150	100	—	120	S7
114	5.4	150	100	—	120	S7
115	5.4	150	100	—	120	S7
116	19.2	200	160	—	200	S8
117	19.2	200	160	—	200	S8
118	15.36	160	120	—	250	S8

TABLE 4: Continued.

No	Mass (kg)	Dimensions (mm)				Surface
		a_i	b_i	r_i	h_i	
119	15.36	160	120	—	250	S8
120	5.4	150	100	—	120	S8

The first 72 are cylinders, and the last 48 are cuboids.

TABLE 5: The data of the SOPS layout scheme by CCMO-EMAS in Section 6.

No.	x (mm)	y (mm)	α_i (rad)
1	-43.779	308.314	c
2	170.558	125.613	c
3	-9.702	-200.726	c
4	185.863	-74.234	c
5	366.329	-83.129	c
6	280.420	-223.813	c
7	-191.707	-135.696	c
8	-327.228	70.259	c
9	-159.536	194.246	c
10	122.750	-158.309	c
11	330.463	190.611	c
12	109.668	380.795	c
13	-314.250	106.689	c
14	-107.376	-178.995	c
15	-173.774	37.996	c
16	188.104	-8.936	c
17	212.042	-374.329	c
18	-205.791	-296.925	c
19	-201.576	15.537	c
20	293.819	-219.489	c
21	182.695	156.572	c
22	195.776	-44.106	c
23	-174.557	-317.978	c
24	55.683	-190.278	c
25	-193.199	355.697	c
26	-177.223	-92.859	c
27	-149.729	136.669	c
28	-337.329	51.946	c
29	-162.139	-272.197	c
30	-305.650	225.522	c
31	-311.217	-208.473	c
32	92.696	317.435	c
33	14.689	170.397	c
34	275.951	-138.141	c
35	350.734	-31.225	c
36	-16.816	-164.086	c
37	-178.766	276.015	c
38	-59.774	-190.907	c
39	-188.973	72.991	c
40	-247.372	-121.078	c
41	315.318	-60.968	c
42	19.175	163.936	c
43	-142.282	-330.165	c
44	-17.933	278.073	c
45	-354.902	51.771	c
46	354.087	-66.598	c
47	190.931	95.383	c
48	-310.844	-89.185	c
49	318.844	233.499	c
50	-170.711	65.514	c

TABLE 5: Continued.

No.	x (mm)	y (mm)	α_i (rad)
51	-265.337	-302.588	c
52	209.589	352.941	c
53	351.818	100.421	c
54	-151.926	-69.308	c
55	-270.162	-185.894	c
56	-7.145	357.953	c
57	356.155	107.321	c
58	-152.224	203.019	c
59	68.969	-349.975	c
60	-358.053	162.193	c
61	-414.530	15.459	c
62	-22.996	-199.606	c
63	-199.055	-99.365	c
64	-106.949	303.349	c
65	128.580	-306.114	c
66	178.192	185.997	c
67	73.911	331.223	c
68	347.449	148.706	c
69	-269.834	-255.928	c
70	275.710	-247.057	c
71	-155.502	-294.839	c
72	-18.424	-383.747	c
73	150.756	-254.385	$\pi/2$
74	-184.382	46.341	0
75	106.597	299.879	$\pi/2$
76	-21.048	155.699	0
77	-61.971	228.618	$\pi/2$
78	11.225	-333.317	0
79	-323.521	-99.564	$\pi/2$
80	323.720	-43.636	$\pi/2$
81	148.684	173.627	$\pi/2$
82	298.125	-218.899	0
83	-321.441	245.680	0
84	4.177	357.520	0
85	57.752	-341.108	0
86	-384.239	25.367	$\pi/2$
87	-177.884	-159.943	0
88	-111.091	198.833	$\pi/2$
89	15.501	200.327	$\pi/2$
90	201.091	351.896	$\pi/2$
91	392.423	55.767	0
92	177.660	155.373	0
93	134.286	-185.560	$\pi/2$
94	160.538	-5.223	$\pi/2$
95	-100.842	305.583	0
96	-25.770	-305.227	$\pi/2$
97	295.801	74.355	0
98	180.283	224.878	0
99	187.493	-67.984	$\pi/2$
100	155.467	89.662	$\pi/2$
101	41.923	-255.837	$\pi/2$

TABLE 5: Continued.

No.	x (mm)	y (mm)	α_i (rad)
102	51.885	304.472	$\pi/2$
103	244.194	-244.790	0
104	-110.027	-254.381	$\pi/2$
105	-118.611	249.243	$\pi/2$
106	-335.324	102.960	0
107	171.027	-80.546	0
108	-87.797	-226.323	$\pi/2$
109	-225.907	2.610	0
110	225.276	-76.409	0
111	175.046	137.612	$\pi/2$
112	209.739	-242.638	$\pi/2$
113	151.126	357.577	$\pi/2$
114	10.798	175.366	$\pi/2$
115	39.836	-191.212	$\pi/2$
116	324.871	-35.222	$\pi/2$
117	-188.505	101.424	$\pi/2$
118	162.364	-117.534	0
119	176.208	26.967	$\pi/2$
120	31.956	176.982	$\pi/2$

$\alpha_i = \{0, (\pi/2)\}$ for cuboids, and $\alpha_c = 0$ for cylinder which represented with c .

In this multiobjective combinational problem, 120 components will be layout on the support surfaces of the satellite, which include 72 cylinders and 48 cuboids. It is assumed that all the components are rigid bodies and the mass are uniformly distributed. As shown in Figure 8, $R = 500$ mm, $R_C = 100$ mm, $H_1 = 300$ mm, $H_2 = 830$ mm, $H_3 = 1360$ mm, $H_4 = 1890$ mm, $H_t = 2200$ mm, $D_t = 20$ mm. The total mass of satellite cabin is 1553.068 kg. The centroid coordinate of relative reference coordinate system $Oxyz$ is $(0, 0, 1100)$ mm, and the Inertia matrix I_0 is

$$I_0 = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} = \begin{bmatrix} 1808.992 & 0 & 0 \\ 0 & 1808.992 & 0 \\ 0 & 0 & 292.819 \end{bmatrix} \text{kg} \cdot \text{m}^2. \quad (13)$$

The dimensions and masses of components are shown in Table 4.

The data of the SOPS layout scheme by CCMO-EMAS is shown in Table 5.

Data Availability

The experimental data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was also supported by the International co-operation project of Shandong Academy of Sciences

(Awards: 2019GHZD08), 2019 Shandong Province Key R&D Plan (Awards: 2019JZZY010126) and the China National Key R&D Plan (Awards: 2018YFE0197700), funded by the Key R&D Program of Shandong (Awards: 2017GGX50107). The authors are grateful for Specially Recruited Experts of Taishan Scholar. This research was funded by the National Natural Science Foundation of China, grant numbers 61472062 and 91546123.

References

- [1] M. Barbati, G. Bruno, and A. Genovese, "Applications of agent-based models for optimization problems: a literature review," *Expert Systems with Applications*, vol. 39, no. 5, pp. 6020–6028, 2012.
- [2] A. Byrski, R. Dreżewski, L. Siwik, and M. Kisiel-Dorohinicki, "Evolutionary multi-agent systems," *The Knowledge Engineering Review*, vol. 30, no. 2, pp. 171–186, 2015.
- [3] A. Cardon, T. Galinho, and J.-P. Vacher, "Genetic algorithms using multi-objectives in a multi-agent system," *Robotics and Autonomous Systems*, vol. 33, no. 2-3, pp. 179–190, 2000.
- [4] W. Zhong, J. Liu, M. Xue, and L. Jiao, "A multiagent genetic algorithm for global numerical optimization," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 2, pp. 1128–1141, 2004.
- [5] L. Hanna and J. Cagan, "Evolutionary multi-agent systems: an adaptive and dynamic approach to optimization," *Journal of Mechanical Design*, vol. 131, Article ID 011010, 2009.
- [6] S. Talukdar, L. Baerentzen, A. Gove, and P. De Souza, "Asynchronous teams: cooperation schemes for autonomous agents," *Journal of Heuristics*, vol. 4, no. 4, pp. 295–321, 1998.
- [7] M. I. Campbell, J. Cagan, and K. Kotovsky, "A-design: an agent-based approach to conceptual design in a dynamic environment," *Research in Engineering Design*, vol. 11, no. 3, pp. 172–192, 1999.
- [8] L. H. Landry and J. Cagan, "Search strategies in evolutionary multi-agent systems: the effect of cooperation and reward on solution quality," *Journal of Mechanical Design*, vol. 133, Article ID 061005, 2011.
- [9] L. H. Landry and J. Cagan, "Protocol-based multi-agent systems: examining the effect of diversity, dynamism, and cooperation in heuristic optimization approaches," *Journal of Mechanical Design*, vol. 133, Article ID 021001, 2011.
- [10] P. F. Egan, J. Cagan, C. Schunn, and P. R. LeDuc, "Cognitive-based search strategies for complex bio-nanotechnology design derived through symbiotic human and agent-based approaches," in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Buffalo, NY, USA, August 2014.
- [11] W. Zhao, S. Alam, and H. A. Abbass, "MOCCA-II: a multi-objective co-operative co-evolutionary algorithm," *Applied Soft Computing*, vol. 23, pp. 407–416, 2014.
- [12] R. Dreżewski and K. Doroz, "An agent-based co-evolutionary multi-objective algorithm for portfolio optimization," *Symmetry*, vol. 9, p. 168, 2017.
- [13] B. Dorransoro, G. Danoy, A. J. Nebro, and P. Bouvry, "Achieving super-linear performance in parallel multi-objective evolutionary algorithms by means of cooperative co-evolution," *Computers and Operations Research*, vol. 40, no. 6, pp. 1552–1563, 2013.
- [14] A. W. Iorio and X. D. Li, "A cooperative coevolutionary multiobjective algorithm using non-dominated sorting," in

- Proceedings of the Genetic and Evolutionary Computation-Gecco 2004*, pp. 537–548, Berlin, Germany, 2004.
- [15] K. C. Tan, T. H. Lee, Y. J. Yang, and D. S. Liu, “A cooperative coevolutionary algorithm for multiobjective optimization,” in *Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics*, pp. 1926–1931, Hague, Netherlands, October 2004.
- [16] R. P. Wiegand, W. C. Liles, and K. A. De Jong, “An empirical analysis of collaboration methods in cooperative coevolutionary algorithms,” in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pp. 1235–1242, San Francisco, CA, USA, July 2001.
- [17] A. Bucci and J. B. Pollack, “On identifying global optima in cooperative coevolution,” in *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, pp. 539–544, Washington, DC, USA, June 2005.
- [18] Z. H. Zhang, Y. S. Wang, H. F. Teng, and Y. J. Shi, “Parallel dual-system cooperative co-evolutionary differential evolution algorithm with human-computer cooperation for multi-cabin satellite layout optimization,” *Journal of Convergence Information Technology*, vol. 8, no. 4, pp. 711–720, 2013.
- [19] B. Zhang, H.-F. Teng, and Y.-J. Shi, “Layout optimization of satellite module using soft computing techniques,” *Applied Soft Computing*, vol. 8, no. 1, pp. 507–521, 2008.
- [20] H. F. Teng, Y. Chen, W. Zeng, Y. J. Shi, and Q. H. Hu, “A dual-system variable-grain cooperative coevolutionary algorithm: satellite-module layout design,” *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 3, pp. 438–455, 2010.
- [21] K. C. Tan, C. K. Goh, A. A. Mamun, and E. Z. Ei, “An evolutionary artificial immune system for multi-objective optimization,” *European Journal of Operational Research*, vol. 187, no. 2, pp. 371–392, 2008.
- [22] A. P. C. Cuco, F. L. De Sousa, and A. J. S. Neto, “A multi-objective methodology for spacecraft equipment layouts,” *Optimization and Engineering*, vol. 16, no. 1, pp. 165–181, 2015.