IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 10, 2020, accepted February 24, 2020, date of publication March 23, 2020, date of current version April 13, 2020.
Digital Object Identifier 10.1109/ACCESS.2020.2981489

A Configuration Optimization Method of Integrating Concentrating & Migration Genetic Algorithm for Cellular Reconfigurable Spacecrafts

JUN ZHOU, BOSONG WEI, AND YINGYING LIU
Institute of Precision Guidance and Control, Northwestern Polytechnical University, Xi'an 710072, China
Corresponding author: Yingying Liu (liuyingying@nwpu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61761136001, and in part by the Science and Technology on Space Intelligent Control Laboratory under Grant KGJZDSYS-2018-04.

ABSTRACT The cellular reconfigurable spacecraft has been highly concerned as a flexible and cheaper space system. To obtain a suitable and optimal configuration scheme for our work, a novel configuration optimization method is proposed in this paper. Firstly, three-level module management architecture is built for the configuration design of cellular reconfigurable spacecraft, which can realize the free definition of modules. Afterwards, by designing the performance index suitable for our task requirements, the configuration design issue is modeled as an optimal problem. Due to the limitation of existing solutions, a novel genetic algorithm is developed by drawing inspiration from the concentrating & migration behavior in wildebeest population. And then, based on all above works, a configuration optimization method based on the concentrating & migration genetic algorithm is presented. The simulation result shows that the presented method can obtain an assembly configuration with better moment of inertia and smaller envelope filling ratio. That is to say, the overall performance of configuration is better.

INDEX TERMS Cellular reconfigurable spacecraft, in-orbit autonomous assembly, configuration optimal design, genetic algorithm.

NOMENCLATURE

CRS	= Cellular Reconfigurable Spacecraft
MMA	= Module Management Architecture
MI	= Moment of Inertia
SP	= Structural Properties
FP	= Functional Properties
CP	= Components Properties
NCA	= No-connection Area
EFR	= Envelope Filling Ratio
CER	= Cost-Effectiveness Ratio
GA	= Genetic algorithm
CMGA	= Concentrating & Migration Genetic algorithm

I. INTRODUCTION

In recent years, the modular spacecraft and their applications have been greatly concerned in the world [1]. These spacecraft are based on modular design and consisted of many modules with specific functions. Similar to the concept of cellular land robots, this type of spacecraft is called cellular reconfigurable spacecraft (CRS). Compared with the traditional integrated satellites, the potential advantages of CRS are excellent: CRS is cheaper and can be repaired in-orbit faster and easily. By replacing the disabled module, CRS can be reused and prolong its life. In addition, for other space tasks, CRS may be reconfigured into another type of CRS with a new function.

For technology research and project development about CRS, there have been some impressive achievements in the past few years. In Japan, Afreen Siddiqi *et al.* pointed out that the homogeneous design can achieve maximum reconfigurability [2], Tanaka *et al.* proposed a homogeneous cellular module called Cellsat [3], [4]; In the US, the Phoenix Project funded by DAPRA is developing a cellular spacecraft which can be fixed on the space deserted structures and recycle these structures as a new satellite [5], [6]. NovaWurks provides a homogeneous solution called HISat for this project [7], [8]; In the UK, Underwood *et al.* researched a typical case of

The associate editor coordinating the review of this manuscript and approving it for publication was Datong Liu.

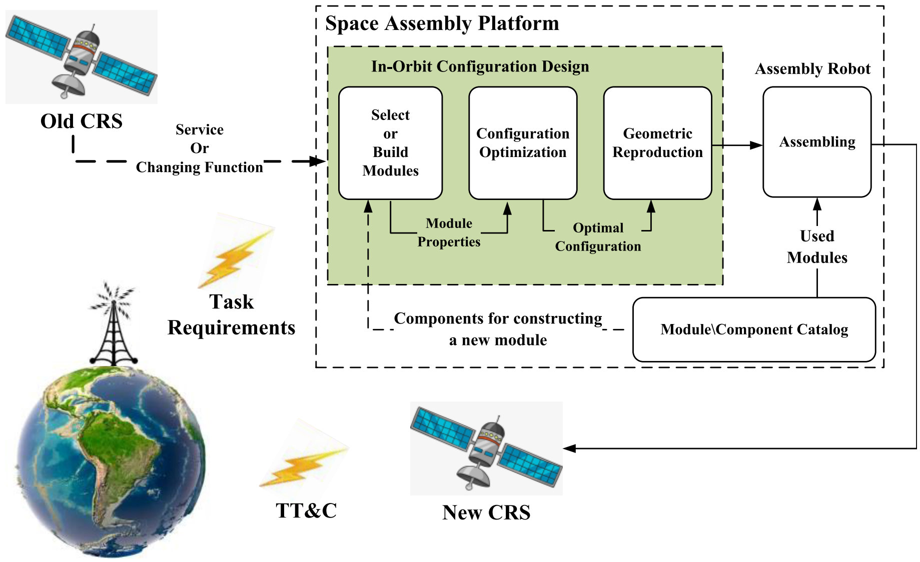

FIGURE 1. The reconfiguration process for CRS in-orbit service.

building CRS by using CubeSat components in the AAREST task [9]. In Germany, a reconfigurable spacecraft scheme based on intelligent system is proposed in the famous iBOSS project [10]. The spacecraft is assembled in-orbit using hexahedron module with intelligent interface and can be reconfigured by computer-aided design software [11], [12]; In China, some research imagines for CRS have been proposed [13], [14], and X B Cao has proposed a reconfigurable microsatellite platform based on the ''plug and play'' module [15].

In our work, we hope to develop a free assembly CRS which is built by existing CubeSat components. Therefore, we must build an in-orbit configuration design program for the CRS reconfiguration, which will be used to obtain a suitable and optimal configuration for CRS. Because existing solutions have limitations for our task requirements, a novel configuration optimization method is proposed in this paper.

The rest paper can be divided into 5 parts. Section 2 describes configuration design problem for CRS, and the CRS configuration modeling is illustrated in Section 3. And then a configuration optimization method is proposed in Section 4. Afterwards, the simulation results and analysis are given in Section 5. Finally, the conclusions are discussed in Section 6.

II. CONFIGURATION DESIGN PROBLEM FOR CELLULAR SPACECRAFT

CRS generally consists of many modules with different functions. Based on the standard structure and components of CubeSat, we construct a reconfigurable cellular module for in-orbit assembly. The module can achieve the interconnection in mechanical, electrical, data and thermal through the standard interface installed on the module surface. However, for modules with unique functions, some areas on their surfaces cannot install the interface and are forbidden to connect. These areas are called No-Connect Areas (NCA). Meanwhile, modules are required to connect to at least one other module through the interface.

For each CRS task, it is necessary to select a combination of modules with different numbers and different functions. Then, these modules will be assembled into the CRS meeting the mission requirements. The assembly relationship of these modules, including positional relationship and relative attitude, is called the configuration of CRS.

Since the modules can be interconnected by a unified bus system, the assembly relationship between the modules is not unique. Rather, as long as the task requirements are satisfied, a module can even be placed anywhere in CRS. Therefore, for a space task requiring assembly of multiple modules, a large number of possible configurations will be derived. With the number of modules increasing, the number of possible configurations will explode. Manual method will be difficult to ensure the global performance and efficiency for CRS configuration design. Thus, we have to find a method that can automatically design the optimal CRS configuration according to both the task requirements and assembly constraints. This method can be applied to the reconfiguration process for CRS in-orbit service or changing task requirements, as shown in Figure 1.

Where, the ''In-Orbit Configuration Design'' part is corresponding to the method proposed in this paper. It will receive the task requirements sent from ground station and send assembly instructions to the service robot. The assembly instructions are used to instruct the robot to take modules from the catalogue and build the new configuration.

III. MODELING OF CONFIGURATION OPTIMIZATION

A. MODELING OF CRS CONFIGURATION

For CRS configuration design, two kinds of models are needed as the module relationships model and the module

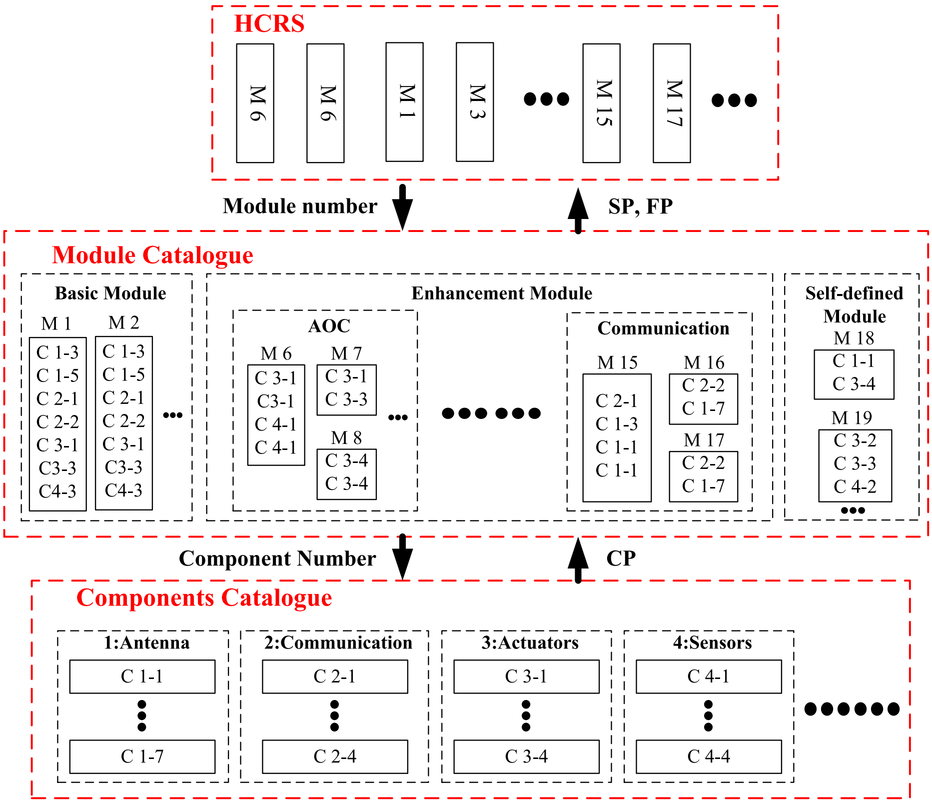

FIGURE 2. Three-level MMA allowing user-defined module.

properties management model. The module relationship is used to descript the geometry of CRS. The module properties management is used to judge the feasibility and calculate the overall performance of each CRS configuration.

Modeling of module relationships is generally considered as the relationships between the body frame *oxyz* of each module and a fixed assembly frame *OXYZ* [11], [16]. In this paper, including the information of position and attitude, the assembly relationship of each module is described by an array **c** as follows:

$$\mathbf{c} = [x_o, y_o, z_o, X, Y] \tag{1}$$

where, $[x_o, y_o, z_o]$ is the position vector of the *oxyz* origin in *OXYZ*. The directions of the axis $+ox$ and axis $+oy$ in *OXYZ* are described by X, Y. Thus, any configuration geometry can be described by matrix $\mathbf{C}_{m\times 5}$ composed of array **c** of each module, m is the number of assembly modules.

The module properties management model is commonly built as module management architecture (MMA). A two-level ontology-based MMA has been proposed in the "iBOSS" project [12]. Such two-level MMA can only manage the pre-defined modules in the module catalogue. However, during the CRS reconfiguration process, the modules undefined in advance may be sometimes needed. In this case, the two-level MMA is disabling. Thus, a new universal three-level MMA is proposed for our project as shown in Figure 2. Where array **SP** describes the module structural properties, **FP** describes the module functional properties and **CP** describes the component properties. Comparing to the two-level MMA, a component level is expanded in our model. Based on this new MMA, the module undefined in advance can be constructed in-orbit by users, which is called user-defined module, and its properties can be calculated automatically. Once an undefined module is needed in reconfiguration process, user can choose the components from the component catalogue directly and construct this module. Meanwhile, the properties of this user-defined module will be given automatically for further optimization calculation.

B. MODEL OF CRS CONFIGURATION OPTIMIZATION

CRS configuration design is a process of finding an optimal configuration from all possible configuration samples. Its mathematical essence is an optimization problem based on the configuration constraints and optimization objectives.

1) CONSTRAINTS

For CRS configuration design, constraints are used to avoid the incorrect connection state. These incorrect states include the non-connection state, overlap state and violation of NCA as shown in Figure 3 b, c, d. According to these necessary assembly rules, the constraints g_1, g_2 and g_3 can be deduced as follows:

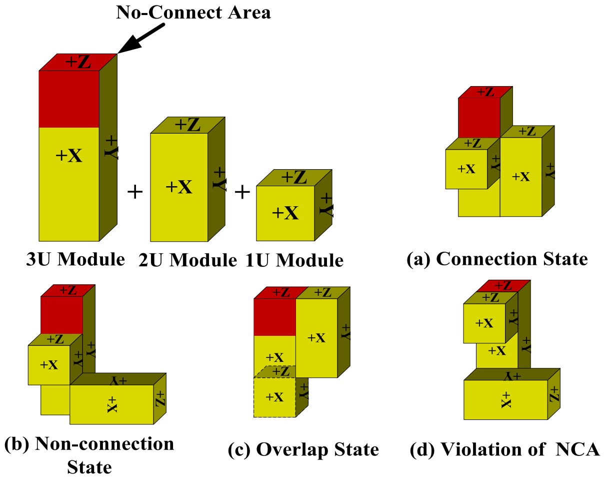

FIGURE 3. Possible relationships between modules of a 6U-class CRS case. (a) A correct connection; (b) An incorrect connection. 2U module is not connected; (c) An incorrect connection. 1U module is overlapped with 3U module; (d) An incorrect connection. The NCA of 3U module is violated by 1U module.

$g_1 : \sum_{j=1}^{m} a_{ij} \neq 0$. Each module must be connected to at least one another module.

$g_2 : a_{ij} \geq 0$. Modules are not allowed to overlap each other.

Where, the connection matrix $\mathbf{A} = [a_{ij}]_{m\times m}$ is built to descript the connection relationship between each two modules. The value of a_{ij} is defined as follows:

$$a_{ij} = \begin{cases} 1, & \text{“}Connection\ State\text{”} \quad and \quad i \neq j \\ -1, & \text{“}Overlap\ State\text{”} \quad and \quad i \neq j \\ 0, & \text{“}Non-Connection\ State\text{”} \quad or \quad i = j \end{cases} \tag{2}$$

$g_3 : d_{ij} \neq 0$. The NCA of each module must be satisfied.

To judge g_3, the connection matrix $\mathbf{A}$ is augmented as matrix $\overline{\mathbf{A}} = [\bar{a}_{ij}]_{m\times m}$ which records the connection areas $\mathbf{Con}_{ij}$ between each two modules. And the matrix $\mathbf{D_{ban}} = [d_{ij}]_{m\times m}$ is used to record the result of NCAs checking. The value of d_{ij} is defined as follows:

$$\bar{a}_{ij} = \begin{cases} \mathbf{Con}_{ij}, & \text{“}Connection\ State\text{”} \quad and \quad i \neq j \\ 0, & \text{“}Overlap\ State\text{”} \quad and \quad i \neq j \\ 0, & \text{“}Suspension\ State\text{”} \quad or \quad i = j \end{cases}$$
$$d_{ij} = \begin{cases} 0, & \overline{\mathbf{Ban}}_i \cap \bar{a}_{ij} \neq \emptyset \\ 1, & else \end{cases} \tag{3}$$

2) OPTIMIZATION OBJECTIVES

According to our requirements for CRS configuration design, the optimization objectives can be proposed from three aspects:

$j_1 = 1 - \sigma_I/\hat{I}$. This sub-index indicates that the standard deviation of MI is expected to be minimized. σ_I is the standard deviation of MI. $\hat{I}$ is normalization constant.

$j_2 = 1 - \bar{I}/\hat{I}$. This sub-index indicates that the value of MI is expected to be minimized. $\bar{I}$ is the average value of MI. $\hat{I}$ is normalization constant.

$j_3 = \eta = V_{all}/V_{env}$ is called “Envelope Filling Ratio”(EFR). It indicates that the geometrical of the CRS is expected to be regular and the space occupation is expected to be minimized. V_{all} is the total volume of the module used in CRS. V_{env} is the volume of CRS envelope hexahedron. $V_{all} \leq V_{env}$.

3) OPTIMIZATION MODEL

Overall, the problem of CRS configuration design can be modeled as a constrained multi-objective optimization problem. There are still many methods for utilizing an optimization approach with such multiple objectives, which are widely used for many similar engineering problems [17]–[20]. The optimization problem can be described as follows:

$$\begin{aligned} &\max_{\mathbf{C}_i \in \mathbf{S}} J = f(j_1, j_2, j_3) \\ &s.t. \quad g_1, g_2, g_3 \end{aligned} \tag{4}$$

According to whether all the constraints g_1, g_2, g_3 are satisfied, the sample space $\mathbf{S}$ is divided into the feasible region $\mathbf{D}$ and infeasible region $\overline{\mathbf{D}}$. And it can be transformed into an unconstrained single-objective optimization problem by setting the weight coefficients and penalty coefficients:

$$\max_{\mathbf{C}_i \in \mathbf{S}} J_1 = (k_1 j_1 + k_2 j_2 + k_3 j_3) \cdot l_1 g_1 \cdot l_2 g_2 \cdot l_3 g_3 \tag{5}$$

where the objective functions j_1, j_2 and j_3 are weighted by $k_1, k_2, k_3 \in R^+$. The value of k_1, k_2 and k_3 is considered to be determined according to the task requirements. And the two-valued penalty coefficients $l_1, l_2, l_3 \in \{0, 1\}$ are used to ensure that the infeasible solutions violating the constraint conditions will be kept away from the maximum value. Thus, under the effect of penalty coefficients, the value of index function J_1 of the new unconstrained single-objective optimization problem is as follows:

$$J_1 = \begin{cases} k_1 j_1 + k_2 j_2 + k_3 j_3, & \mathbf{C}_i \in \mathbf{D} \\ 0, & \mathbf{C}_i \in \overline{\mathbf{D}} \end{cases} \tag{6}$$

IV. CONFIGURATION OPTIMIZATION METHOD BASED ON CMGA

The optimization problem shown in Eq.4 is a NP-hard problem. By applying J_1 as the fitness, the genetic algorithm (GA) can be used to obtain its optimal solution. For reference, GA has been used to solve a similar problem in “iBOSS” project [11]. However, according to our several experiments, we find the “iBOSS” solving method based on traditional GA cannot obtain a suitable solution for some tasks. This may because the influence of low fitness individuals is not eliminated effectively in traditional GA. Strongly inspired by the great migration of wildebeest, an initialization modified GA based on small population migration and merging behavior is developed, which can effectively overcome that limitation.

In nature, several small wildebeest populations will concentrate from everywhere in June to prepare the annual great migration. About 50% of the individuals die in this process, and the remaining good individuals will return to the habitat to breed in December. Although the concentrating and migration are non-genetic behaviors, they still have great influence for the evolution of wildebeests. Concentrating enables the migration population to have sufficient genetic diversity. Migration allows the wildebeest to retain enough good individuals for breeding offspring.

The genetic algorithm can be simulated by the great migration process. The initialization operation and genetic operation of GA respectively correspond to the concentrating stage and the great migration stage of the great migration. We believe that the quality of the initial migration population is improved by the migration and merging of several small populations during the concentrating stage, and then the population quality after the large migration is also improved. Inspired by this, we modified the initialization operation of GA by adding the migration and merging behavior of several small random populations to improve the fitness of the optimal solution, which is shown in Figure 4. We call this modified genetic algorithm as the concentrating & migration genetic algorithm (CMGA).

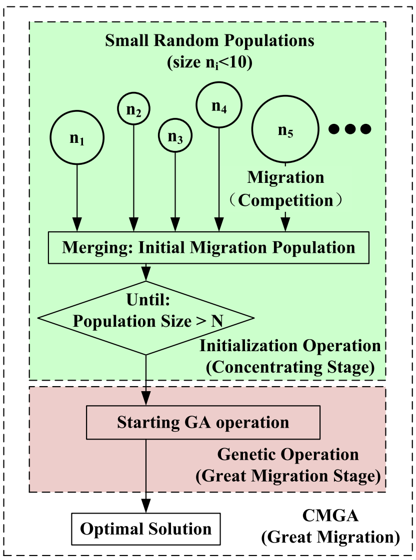

FIGURE 4. Flow of the CMGA.

As shown in Figure 4, CMGA is an initialization modified GA to overcome the influence of low fitness individuals at initialization operation. The modified initialization operation is mainly divided into two parts:

Migration: This behavior simulates the elimination process of several small wildebeest populations with geographical isolation concentrating towards the initial migration population. It is used to improve the quality of the initial migration population, thereby helping to produce the individual with better fitness after the large migration. The governing parameters are individual fitness J_1 and a fitness threshold $\widetilde{J}_1$:

$$J_1 = \begin{cases} J_1 > \widetilde{J}_1, & Accept \\ J_1 < \widetilde{J}_1, & Reject \end{cases} \tag{7}$$

The value of fitness threshold $\widetilde{J}_1$ used in this paper is 0.

Merging: This behavior simulates the process of small populations merge into the initial migration population. Its governing parameters include the maximum size of the small random populations n and the maximum size of the initial migration population N. The numerical relation between real-time size N^* and maximum size N is the condition to stop the merging behavior:

$$\begin{cases} \mathrm{N}^* \geq N, & Stop \\ \mathrm{N}^* < N, & Continue \end{cases} \tag{8}$$

The maximum size of the small random populations n used in this paper is 10.

After all the work above, we need to use the configuration optimization method based on CMGA to find the optimal configuration for assembly tasks. The application of this method to the configuration optimization problem is as follows:

The configuration sample is an individual in the CMGA population, and is encoded as a corresponding configuration matrix $\mathbf{C}$ according to the configuration model. All the possible configuration samples constitute the search space $\mathbf{S}$.

When initializing the CMGA population, several sample sets with a size of $n_i < 10$ are generated first, and then the fitness of each configuration sample is calculated by the objective function J_1 in Eq.5. The samples meeting the fitness threshold $\widetilde{J}_1$ are accepted into the initial population until the stopping condition in Eq.8 is satisfied.

In the genetic operation, the configuration matrix $\mathbf{C}$ of each individual is operated according to their fitness in the parent population, and the new configuration matrix after genetic operation will be stored into the offspring population as a new individual and wait for the next genetic operation. Finally, the individual with the best fitness in population P_{k+1} satisfying the stopping condition is regarded as the optimal configuration. The detailed process of optimizing the configuration by using the method based on CMGA is shown in Figure 5:

V. EXPERIMENT RESULTS AND ANALYSIS

In this section, the performance of the configuration optimization method presented in this paper is verified based on two simulation cases. Case 1 with 3 modules is designed to show clearly the difference in optimizing performance between the proposed method and the iBOSS method. Case 2 with 27 modules is designed to check the actual performance for our CRS application.

The choice of k_1, k_2 and k_3 is considered to be determined by the task requirements in our work since they are using to descript the decision weight between three optimization objectives j_1, j_2 and j_3. In this manuscript they are selected subjectively as $k_1 = 0.2$, $k_2 = 0.1$ and $k_3 = 1$ since we want to ensure the assembly configuration with the most regular geometry will be first chosen, then the configuration with the same envelope filling ratio(EFR) would be further decided by value of moment of inertia(MI).

TABLE 1. Module configuration of 6U-Class CRS Case1.

Module	Configuration	The Structural Properties $\mathbf{SP}=[s_1,s_2,s_3,s_4,s_5,s_6]$
basic module	3U structure, U/V band antenna, U/V transceiver, CPU, EPS and battery pack, solar panels, magnetic torquer, 3-axis CubeWheel, GNSS, sun sensors, radiator	$s_1:2.16$; $s_2:[1,1,3]$ $s_3:\begin{bmatrix}[0,1],[0,1],3;1,[0,1],[2,3];0,[0,1],[2,3]\\0,[0,1],[0,1];[0,1],0,[0,1];1,[0,1],[0,1]\end{bmatrix}$ $s_4:1$; $s_6:3$ $s_5:\mathbf{I}=[0.0231,0.0231,0.0052],\mathbf{r}=[0.5,0.5,1.5]$,
image payload module	2U structure, SCS Gecko imager	$s_1:1.556$; $s_2:[1,1,2]$; $s_3:[[0,1],[0,1],2]$ $s_4:1$; $s_6:2$ $s_5:\mathbf{I}=[0.0083,0.0083,0.0033],\mathbf{r}=[0.5,0.5,1]$
communication enhancing module	1U structure, S-band antenna, S-band transceiver	$s_1:0.687$; $s_2:[1,1,1]$; $s_3:[0,0,0]$ $s_4:0$; $s_6:1$ $s_5:\mathbf{I}=[0.0015,0.0015,0.0015],\mathbf{r}=[0.5,0.5,0.5]$

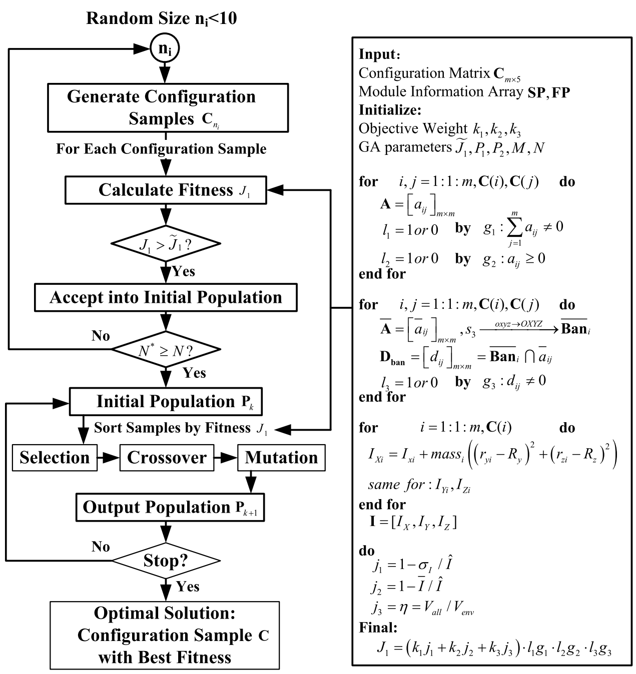

FIGURE 5. Detailed flow of CMGA method applied to configuration optimization problem.

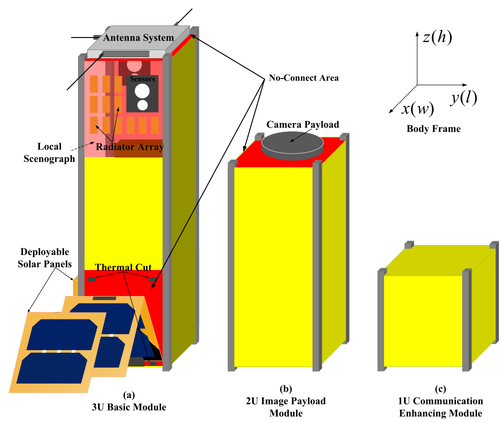

FIGURE 6. The geometry structure of 3 modules for case 1.

For the parameters in genetic algorithm, the crossover probability is selected as $P_1 = 50\%$ because the winning rate of each individual is considered to be 50% in a duel. The mutation probability is selected as $P_2 = 2\%$ to ensure the reasonable mutation probability of each individual and to enable the population to add new gene samples. The population size is selected as $N = 100$ and the genetic algebra is selected as $M = 100$.

A. CASE1 FOR COMPARISON

In case 1, three modules are used to construct a simple CRS. For these 3 modules, the iBOSS method and our method proposed are respectively used to optimize the CRS configuration. And then, the comparing analysis is given. The geometry structure of 3 modules is shown in Figure 6, and the module properties we used are shown in Table 1.

For comparing the performance of our presented method with the iBOSS method, the performance of two methods can be compared from three comparison criteria as follow:

Cost-effectiveness ratio (CER): $N_{sum}/\overline{J_1}$. This criterion is defined as the ratio of the number of the configuration samples searched to the average fitness value. Because the CMGA modifies the initialization operation, it can achieve a higher fitness benefits but increase the time cost of searching configuration samples. Thus, CER is used to calculate the ratio of time cost to fitness benefit as the absolute benefit, which shows the necessity of CMGA in fitness benefit.

Optimal fitness: $J_{1\max}$. This criterion is defined as the fitness of the best individual given by two methods after evolution. It is used to reflect the influence of the modified initialization operation in CMGA on the optimal solution, which can further show the searching efficiency of CMGA.

TABLE 2. CER comparison.

Performance Value	[500,1000)	[1000,1500)	[1500,2000)	[2000,2500)	[2500,3000)	[3000,3500)
Presented Method	0%	2%	82%	16%	0%	0%
iBOSS Method	40%	49%	9%	1.5%	0%	0.5%

TABLE 3. Comparing performance value of the optimal configuration.

Performance Value	[0,0.7)	[0.7,0.8)	[0.8,0.9)	[0.9,1.0)	[1.0,1.1)	[1.1,1.2)	[1.2,1.3)
Presented Method	3.5%	2%	17.5%	33.5%	0%	43.5%	0%
iBOSS Method	1%	1.5%	30.5%	53%	0%	14%	0%

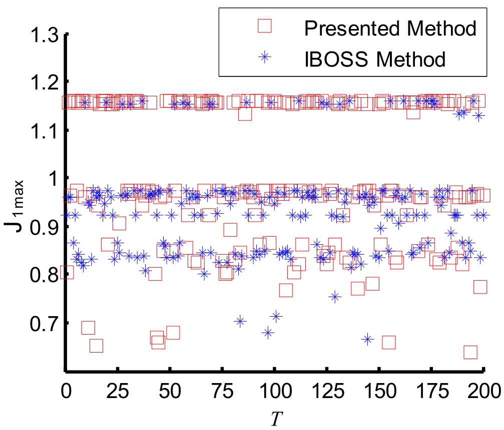

FIGURE 7. Performance value of the optimal configuration from two methods.

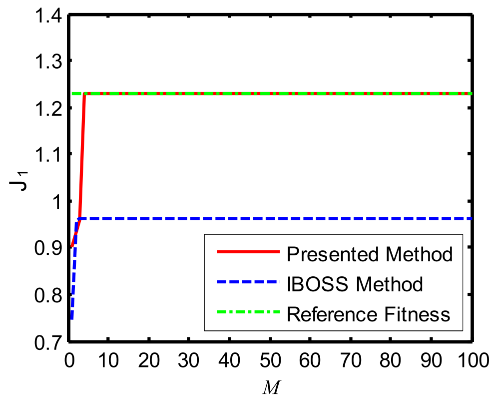

FIGURE 8. The evolution diagram of case 1 by two methods in a single experiment.

Evolution diagram. This is an effective tool to reflect the evolution process from the initial population to the optimal population in a single experiment. It can directly compare the differences between the two methods in initialization result, final result and also the evolution process.

To avoid contingency, 200 independent repeated experiments are used to show CER and optimal fitness, and a single experiment is used to draw the evolution diagram.

Firstly, the CER in 200 experiments is compared in Table 2. According to Table 2, our method achieves a higher CER for 82% in [1500, 2000) and 16% in [2000, 2500). That is to say, although the individuals with higher fitness can be found in our method, its absolute benefit in the initialization process is actually lower.

Secondly, the optimal fitness of two methods is compared as shown in Table 3 and Figure 7. From Table 3, we find that the presented method can obtain more configuration samples with higher performance value, typically, for the fitness in [1.1, 1.2). The presented method can achieve 43.5% while the iBOSS method is 14%. Figure 7 shows that the presented method can overall achieve better performance than the iBOSS method.

Figure 8 is the evolution diagram of the optimal individual fitness searched by two methods in a single experiment. A reference fitness ($J_1 = 1.2276$) of an manual planned configuration (3U: [0, 0, 0, 1, 3], 2U: [0, 1, 1, 1, 3], 1U: [0, 1, 0, 1, 3]) for case 1 is used to be a reference to show the global optimization performance of two methods. The result shows that CMGA can achieve a better global performance. And for a more clear display, the geometry of the optimal configuration in this experiment is compared visually. The geometry of two final configurations obtained by two methods respectively is shown in Figure 9. Obviously, the presented method can find the configuration with a higher EFR (100%) and better MI (MI = [0.0755, 0.0614, 0.0260]).

B. CASE2 FOR APPLICATION

In case 2, a 27U-class CRS with 20 modules as shown in Table 4 is used to design the optimal configuration of CRS for our task application by using the presented method in this paper. The constraints and parameters set for calculating optimal configuration are same with the case 1. Meanwhile, for the task needs, we considered the following 3 extra constraints:

- ➢ The envelope filling ratio: EFR > 90%;
- ➢ The direction of the 2U imaging module and the 1U video module are both required to the nadir facing;

TABLE 4. Module configuration of 27U-Class CRS Case2.

Module Name	Assembly Number	Mass (kg)	Size (U)	MI (kg·m²)	PC
Basic Module	1	3.3780	6	0.03660,0.02815,0.01408	(0.5,1,1.5)
Aerosphere Payload	1	0.2850	2	0.00120,0.00120,0.00020	(0.5,0.5,1)
Imaging Payload	1	1.5560	2	0.00650,0.00650,0.00260	(0.5,0.5,1)
Video Payload	1	0.6172	1	0.00103,0.00103,0.00103	(0.5,0.5,0.5)
Payload Communication	1	0.3070	1	0.00051,0.00051,0.00051	(0.5,0.5,0.5)
GNSS Enhance	1	0.3070	1	0.00051,0.00051,0.00051	(0.5,0.5,0.5)
Infrared Spectrometer	2	0.3672	1	0.00060,0.00060,0.00060	(0.5,0.5,0.5)
Battery Enhance	4	0.8072	1	0.00060,0.00060,0.00060	(0.5,0.5,0.5)
Reaction Wheel	4	0.6870	1	0.00130,0.00130,0.00130	(0.5,0.5,0.5)
Magnetorquer	2	0.3872	1	0.00115,0.00115,0.00115	(0.5,0.5,0.5)
Storage Space Enhance	1	1.1872	1	0.00065,0.00065,0.00065	(0.5,0.5,0.5)
Earth sensor	1	0.2202	1	0.00198,0.00198,0.00198	(0.5,0.5,0.5)

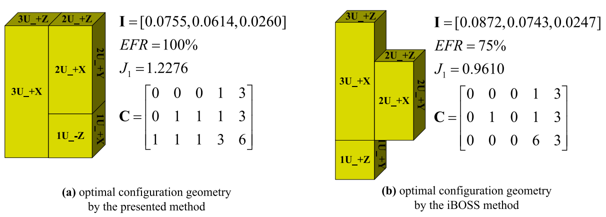

FIGURE 9. The visual geometry effect of the optimal configuration scheme from two methods.

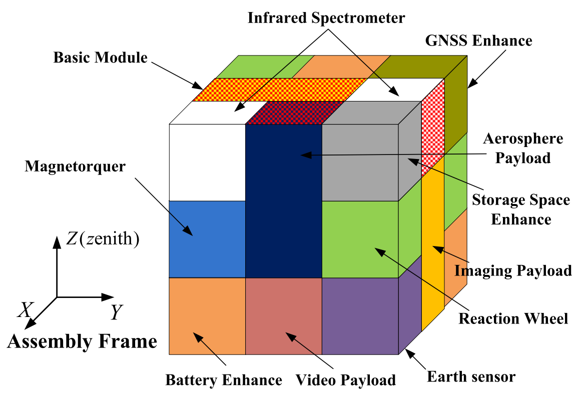

FIGURE 10. The geometry structure of the final optimal configuration with 27 modules.

➢ The payload data downlink services in S-band and the direction of 1U communication enhance module is not required.

The final optimal configuration scheme ($J_1 = 1.285$) given by the presented method has an EFR at 100% and a $\mathbf{MI} = [0.2959, 0.2880, 0.2928]$. All the extra constraints are satisfied. The geometry of this designed optimal configuration is shown in Figure 10 and the evolution diagram is shown in Figure 11. It indicates that the presented method is also efficient for an actual CRS application task.

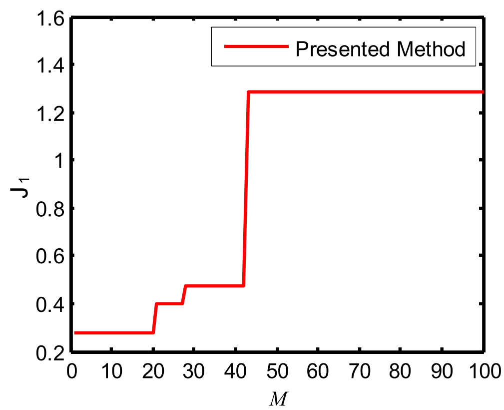

FIGURE 11. The evolution diagram of case 2 by presented method in a single experiment.

From the above, the results of comparing experiment show that the presented method in this paper has better performance for the optimal configuration design of CRS.

VI. CONCLUSION

Today, the miniaturization and modularization of spacecraft is becoming an important developing trend. Due to some excellent advantages such as flexible, faster and cheaper for space tasks, cellular reconfigurable spacecraft (CRS) has been concerned widely. Considering that CubeSat has mature foundation of service and industrial system, we hope to develop our CRS by using CubeSat components. In this paper, a novel configuration optimization method for our CubeSat CRS is reported. By designing the three-level module management architecture and the suitable performance index, the configuration design is modeled as an optimal problem.

And then, inspired by the population concentrating behavior of wildebeests in nature, a concentrating & migration genetic algorithm (CMGA) is developed, which can overcome the limitation of existing methods. Finally, a configuration optimization method based on CMGA is presented. Simulation results indicate the presented method can obtain the configuration scheme with better overall performance for CRS. In the future, some intelligent application based on this presented method will be continually explored. Meanwhile, as a next study, this proposed optimization problem is considered thereby separating the objective functions instead of collecting a single objective function in order to exhibit and visualize the conflicted relations among them.

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their careful review and helpful suggestions that greatly improved the manuscript. They also thank Jianguo Guo and Qirui Zhang for suggesting improvements after reading early versions of this manuscript.

REFERENCES

[1] L. Hill, D. Barnhart, E. Fowler, R. Hunter, L. M. Hoag, B. Sullivan, and P. Will, "The market for Satellite cellularization: A historical view of the impact of the Satlet morphology on the space industry," in *Proc. AIAA SPACE Conf. Expo.*, Sep. 2013, p. 5486, doi: 10.2514/6.2013-5486.

[2] A. Siddiqi and O. de Weck, "Self-similar modular architectures for reconfigurable space systems," in *Proc. 57th Int. Astron. Congr.*, Oct. 2006, pp. 1–13, doi: 10.2514/6.IAC-06-D1.4.03.

[3] H. Tanaka, N. Yamamoto, T. Yairi, and K. Machida, "Cellular satellites and precise assembly by robots for reconfigurable space system," *J. Jpn. Soc. Aeronaut. Space Sci.*, vol. 56, no. 649, pp. 47–56, 2008, doi: 10.2322/jjsass.56.47.

[4] H. Tanaka, N. Yamamoto, T. Yairi, and K. Machida, "Precise assembly by autonomous space robot using skill acquisition learning," in *Proc. 8th Int. Symp. Artif. Intell., Robot. Automat. Space (iSAIRAS)*, Sep. 2005.

[5] L. Gunn, J. Palmer, J. Leung, A. Ogilvie, A. Tadros, L. Chappell, K. Weldy, E. Fowler, J. Eagen, and B. Sullivan, "DARPA phoenix payload orbital delivery system: Progress towards small satellite access to GEO," presented at the 29th Annu. AIAA/USU Conf. Small Satell. (SSC15-II-1), Logan, UT, USA, Jul. 2015.

[6] D. Barnhart, B. Sullivan, R. Hunter, J. Bruhn, E. Fowler, L. M. Hoag, S. Chappie, G. Henshaw, B. E. Kelm, T. Kennedy, M. Mook, and K. Vincent, "Phoenix program Status–2013," in *Proc. AIAA SPACE Conf. Exposit.*, Sep. 2013, doi: 10.2514/6.2013-5341.

[7] P. Melroy, L. Hill, E. E. Fowler, R. Hunter, J. Eagen, B. R. Sullivan, P. Will, and J. Palmer, "DARPA Phoenix Satlets: Progress towards satellite cellularization," in *Proc. AIAA SPACE Conf. Expo.*, Aug. 2015, p. 4487, doi: 10.2514/6.2015-4487.

[8] J. Talbot and M. Walter, "Phoenix and the new Satellite paradigm created by HISat," presented at the 28th Annu.AIAA/USU Conf. Small Satell., Los Alamitos, CA, USA, 2014.

[9] C. Underwood, S. Pellegrino, V. J. Lappas, C. P. Bridges, and J. Baker, "Using CubeSat/micro-satellite technology to demonstrate the autonomous assembly of a reconfigurable space telescope (AAReST)," *Acta Astronaut.*, vol. 114, pp. 112–122, Sep./Oct. 2015, doi: 10.1016/j.actaastro.2015.04.008.

[10] K. Martin, R. Steffen, and W. Jana, "Building block based iBoss approach: Fully modular systems with standard interface to enhance future satellites," presented at the 66rd Int. Astron. Congr., Jerusalem, Israel, 2015.

[11] J. Oberländer, K. Uhl, L. Pfotzer, M. Göller, A. Rönnau, and R. Dillmann, "Management and manipulation of modular and reconfigurable satellites," in *Proc. 7th German Conf. Robot.*, May 2012, pp. 1–6.

[12] M. Goeller, J. Oberlaender, K. Uhl, A. Roennau, and R. Dillmann, "Modular robots for on-orbit satellite servicing," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2012, pp. 2018–2023, doi: 10.1109/ROBIO.2012.6491265.

[13] P. F. Huang, H. T. Chang, Z. Y. Lu, and M. Wang, "Key techniques of on-orbit service-oriented reconfigurable cellularized satellite and its prospects," *J. Astronaut.*, vol. 31, pp. 1–10, Jan. 2016, doi: 10.3873/j.issn.1000-1328.2016.01.001.

[14] S. Xiaofeng, Z. Lingbin, J. Yongqiang, and Z. Qingzhan, "Status and prospect of on-orbit assembly technology," *Manned Spaceflight*, vol. 23, no. 2, pp. 228–235, Apr. 2017, doi: 10.3969/j.issn.1674-5825.2017.02.016.

[15] X. Cao, "Flexible platform based micro-satellite design method," *Aerosp. Sci. Technol.*, vol. 53, pp. 162–168, Jun. 2016, doi: 10.1016/j.ast.2016.03.012.

[16] Q.-X. Wu, G.-Y. Cao, and Y.-Q. Fei, "Described model of a modular self-reconfigurable robot," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Aug. 2005, pp. 188–193, doi: 10.1109/ICMLC.2005.1526942.

[17] T. Talaslioglu, "Optimal dome design considering member-related design constraints," *Frontiers Structural Civil Eng.*, vol. 13, no. 5, pp. 1150–1170, Oct. 2019, doi: 10.1007/s11709-019-0543-5.

[18] T. Talaslioglu, "Design Optimization of Tubular Lattice Girders," *Adv. Steel Construct.*, vol. 15, no. 3, pp. 274–287, 2019, doi: 10.18057/IJASC.2019.15.3.8.

[19] T. Talaslioglu, "Optimal design of steel skeletal structures using the enhanced genetic algorithm methodology," *Frontiers Struct. Civil Eng.*, vol. 13, pp. 863–889, Jul. 2019, doi: 10.1007/s11709-019-0523-9.

[20] T. Talaslioglu, "A unified optimal design approach for geometrically nonlinear skeletal dome structures," *Periodica Polytech. Civil Eng.*, vol. 63, no. 2, pp. 518–540, 2019, doi: 10.3311/PPci.13329.

JUN ZHOU received the Ph.D. degree in aircraft control, guidance and simulation from Northwestern Polytechnical University, Xi'an, China, in 1993. From 2000 to 2001, he was a Senior Visiting Scholar with the University of Maryland, USA, and The University of Sydney, in 2001, Australia. He is currently a Professor, a Ph.D. Supervisor, and the Director of the Shaanxi Microsatellite Engineering Laboratory (SMEL), Shaanxi Electric Servo System Engineering Research Center, Institute of Precision Guidance and Control, Northwestern Polytechnical University. He is the Coordinator of the QB50 Project in Asia. He has over 267 research articles and awards in spacecraft dynamics, control desire simulation, precision guidance and control of spacecraft, and advanced control theory and application.

BOSONG WEI received the B.S. degree from the School of Astronautics, Northwestern Polytechnical University, Xi'an, China, in 2017. He is currently pursuing the M.S. degree with the School of Astronautics, Institute of Precision Guidance and Control, Northwestern Polytechnical University. His research interests include attitude control system of CubeSats and the system design of modular reconfigurable CubeSats.

YINGYING LIU received the Ph.D. degree in navigation, guidance and control from Northwestern Polytechnical University, Xi'an, China, in 2008. She is currently a Professor and a master's supervisor with the School of Astronautics, Institute of Precision Guidance and Control, Northwestern Polytechnical University. She is also the Deputy Director of the Shaanxi Microsatellite Engineering Laboratory (SMEL). She has over 30 research articles and awards in spacecraft dynamics modeling and attitude control, spacecraft physical simulation technology, new method of spacecraft autonomous navigation, satellite formation and intersection control, fault diagnosis, and reconstruction of spacecraft control systems. She is a member of the Chinese Society of Astronautics and the Chinese Society of Theoretical and Applied Mechanics.