Acta Astronautica 163 (2019) 287–298

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

A new multi-satellite autonomous mission allocation and planning method

Bin Du[a,b], Shuang Li[a,b,*]

[a] College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
[b] Advanced Space Technology Laboratory, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, China

ARTICLE INFO

Keywords:
Multi-Agent System
Task clustering
Allocation and collaboration
Mission planning

ABSTRACT

To overcome the issues of inflexible interactive mode, low negotiation efficiency and poor dynamic response capability of satellite clusters, we propose a new multi-dimensional and multi-Agent clusters collaboration model (MDMA-CCM), including task pre-processing, allocation, scheduling and re-planning under failed conditions. First, the task clustering based pre-processing method is adopted to improve the observation efficiency by combining potential targets. Second, the Contract Net Protocol (CNP) based secondary allocation strategy is developed to effectively increase the observation benefit and reduce the impact of task conflicts through announcement, bidding and awarding. Third, a new Interactively Re-planning Method (IRM) is proposed to effectively reduce the loss by re-inserting tasks in case of failure. Simulation results show that the new planning model not only has obvious advantages in calculation time, but also outperforms the centralized multi-Agent system in terms of benefit value and completion rate.

1. Introduction

With the rapid development of remote-sensing technology, sensors can capture higher-resolution images. This has stimulated users' demand for spatial information, and the application of remote sensing has become more widespread. Due to the limited efficiency, traditional single-satellite is difficult to meet the increasing demand for observation data. The multi-satellite can make up for the deficiency of single-satellite. In the past centralized satellite cooperative mode, the management satellite has the planning capability, while the sub-satellites only have tasks execution capability. Compared with the traditional multi-satellite network mission planning methods that lack independent task classification and processing, the new generation of satellites with autonomous online mission planning capabilities can fully utilize the individual intelligence of Agents. Under this precondition, it is necessary to develop a new mission planning method based on mission-efficient allocation strategy with autonomous planning capabilities for remote-sensing satellites.

Satellites mission planning method can be divided into single-satellite scheduling and multi-satellite scheduling. Single-satellites observation scheduling usually includes rational assignment of task time-windows and satellite resources under complex constraints, which corresponds to a NP-complete problem [1,2]. Bensana et al. regarded the single-satellite scheduling problem as a value constraint satisfaction problem. Not only exact methods like a depth-first branching and bounding algorithm or Russian doll searches but also approximate methods such as the greedy search or the Tabu Search (TS) are used to solve the SPOT5 Satellite scheduling problem [3–5]. Their studies showed that the exact algorithm was only suitable for tackling small size rather than large-scale scheduling problems. Therefore, heuristic algorithms have been widely used to solve large-scale problems. Xhafa presented the formulations of the satellite scheduling, merged multiple criteria into one criterion and adopted Genetic Algorithm (GA) to solve various forms of optimization objectives [10–12]. A satellite's resource scheduling model under constraint conditions was established by Xu [31], and the corresponding solution was given based on ant colony algorithm.

Multi-satellite task scheduling issues involve task allocation among single-satellites. Information exchange between the main-satellite and the sub-satellite is required to properly assign tasks. The entire process of communication is called negotiation. There are mainly two negotiation architectures that can be used in multi-satellite collaboration. One of them is the hierarchical structure. Zhang designed a hierarchical hybrid Agent structure and proposed a mission planning and controlling framework for the distributed Multi-Agent System (MAS) [6]. The Hierarchical Task Network (HTN) planning method was used to solve the problem of satellite autonomous tasks assignment [7–9]. However, the HTN approach is unsuitable for large-scale problems and moving targets due to its limited scalability and low scheduling efficiency. The other one is Contract Net Protocol (CNP) introduced by Smith, which is

* Corresponding author. Department of Astronautics Engineering, China.
E-mail address: lishuang@nuaa.edu.cn (S. Li).

https://doi.org/10.1016/j.actaastro.2018.11.001
Received 26 July 2018; Received in revised form 17 September 2018; Accepted 2 November 2018
Available online 09 November 2018

a high-level protocol for achieving efficient cooperation based on a market-like agreement [15–18]. The CNP is widely applied to distributed satellite mission allocation because of its simple principle, easy implementation, high efficiency and superior effect [19,20]. Zhu et al. put forward a multi-Agent cooperation task scheduling framework based on the service center-worker model, which has a feedback correction mechanism when dealing with the complex mission required by CNP [21]. Peng et al. designed the computation framework according to multi-Agent system contract net protocol, and then proposed a load reduction method based on mean shift clustering and a bid evaluation method based on GA [22]. The method has good performance in the case of numerous satellites and large-scale problems. However, most of the existing research on the CNP has focused on the bidding processes based on past experience. In the actual observation mission, the satellite can evaluate the bid according to the visibility of the target and the conflict value between the scheduled tasks. Previous bidding experience does not have much reference value. In the past studies, the tasks of multiple satellites allocation system were usually assigned one by one. The reason behind this phenomenon lies in the fact that the previous sub-satellites only had mission execution capabilities and lacked of the independent mission planning function. In a new generation of multi-Agent system, each Agent can individually plan the assigned tasks sequence and autonomously adjust the planned tasks when a conflict occurs.

Based on the above analysis of the current research status, it is clear that more improvements can be made in this regard. We can assign a set of pre-processed tasks to a single-satellite and adjust the allocation strategy when tasks conflict. In this paper, a new multi-dimensional multi-Agent clusters collaboration model (MDMA-CCM) is designed for the target sets allocation problem, and a task Secondary Allocation Strategy (SAS) is proposed to reduce the impact of task conflicts. In addition, an Interactively Re-planning Method (IRM) is developed to deal with task redistribution in the event of Agents failure. Compared with the traditional centralized model, the model presented in this paper has three improvements. First, in order to maximize the use of independent scheduling ability and reduce the planning burden of the management satellite, we have innovatively designed a task clustering method that transforms targets into multiple target sets by pre-processing and then assigns the target sets to sub-satellites. The sub-satellite independently completes the mission planning with sets. The idea of task clustering originated from the computer field. Michael A et al. firstly merged fine grain tasks into single coarser ones to minimize overall execution time, which helps reduce high communication overhead in existing parallel machines [13]. Mathias Dewatripont et al. introduced task clustering into the planning field to address the problem of optimal task clustering in multitask agency theory. They fully considered the issues of effort substitution, tasks conflicts and missions implicit incentives [14]. However, the researches mentioned above do not concern the task scheduling process. Meanwhile, the current studies of satellite mission clustering do not involve the field of multi-satellite collaboration. In this paper, the task clustering method is introduced into the field of the satellite-to-Earth observation, and the observation target is pre-processed to shorten the allocation time. Second, a SAS is proposed to handle unbid tasks that conflict with sub-satellites' scheduled tasks. With the second bid for conflict tasks, the task sequence is adjusted according to the task benefit in order to increase the number of tasks. Third, for satellites that are unable to perform tasks due to failure [23–25], the IRM is presented to adjust the satellite's task sequence to avoid abandoning tasks.

The rest of this paper is organized as follows. Section 2 describes the detailed formulation of the Earth observation problem based on task clustering. In section 3, a MDMA-CCM and task allocation method is designed. Computer simulation and analysis are presented in Section 4. Finally, Section 5 contains a conclusion.

2. Earth observation problem

When Earth observation satellites fly over the target, the camera orients the target in order to collect image data. In the course of the flight, planned mission queue may not be able to meet the observation requirements, as observation is usually affected by the orbit, the maneuverability of satellite platform, load and energy [26]. Because targets can only be observed in a specific time window, the duty of management Agent is to select and arrange mission's time line in order to avoid task conflicts and ensure maximum mission revenue. As the number of tasks increases, the management agent will have a large planning burden under the limited operational capabilities of onboard satellite's computer. To effectively reduce the task distribution pressure of the system, it is necessary to present a task clustering method in the following section.

2.1. Task clustering method

In actual observation missions, target points are often concentrated in certain hot spots. By the traditional multi-Agent task allocation method, the management Agent assigns point tasks to worker Agents one by one, which is not conducive to large-scale observation targets. The new generation of Earth observation satellites has powerful artificial intelligence, and each Agent has autonomous capability, including mission planning, execution, and failed diagnosis. This paper designs a method for task clustering by grouping closer targets in one set and then assigning set to worker Agents (see Fig. 1). The worker Agent autonomously plans tasks in the set.

An example of the target queue is given in Table 1. The target list is defined by five parameters: target number, latitude, longitude, benefit, required observation duration. Benefit is a unit quantity that represents the importance of the task and is used to characterize the priority of a task in this paper. $Pot_1(\lambda_1, \phi_1)$ represents number 1 point target, λ_1 is longitude and ϕ_1 is latitude. Distance can be calculated for every pair of point target in the longitude and latitude coordinates.

1) Taking Pot_x point as center, the relative distance $Dist(x, i)$ between Pot_x point and Pot_i is calculated in the latitude and longitude coordinates. If $Dist(x, i)$ of Pot_i is less than extend distance $Mdist$, the Pot_i point is put into the set $Cluster_x$ in Eq. (1).

$$Cluster_x = \{Pot_i | 1 \le i \le N, \quad Dist(x, i) < Mdist\} \tag{1}$$

where the N denotes the total number of tasks.

2) As shown in Fig. 2, the relative distances between the targets in the $Cluster_x$ and the surrounding target Pot_j is computed in the latitude and longitude coordinates. Afterwards, if the distance is less than $Mdist$, then Pot_j is placed into the set $Cluster_{x\_\exp}$ in Eq. (2).

$$Cluster_{x\_\exp} = \{Pot_j | 1 \le j \le N, \quad Pot_k \in Cluster_x, Dist(k, j) < Mdist\} \tag{2}$$

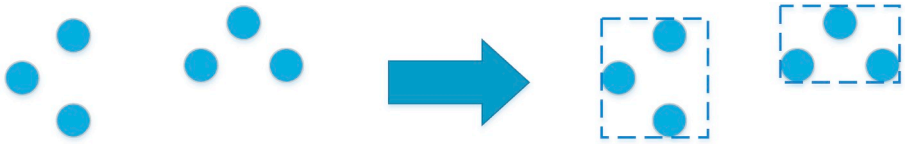

Fig. 1. Task clustering schematic.

Table 1
The latitude and longitude of the target points.

Number	Latitude (degree)	Longitude (degree)	Benefits	Duration(s)
1	109.48	22.22	1	10
2	105.94	50.57	1	10
3	114.00	41.20	3	10
4	112.09	20.96	2	10
5	116.15	9.41	1	10
6	117.46	8.33	3	10
7	105.40	30.44	1	10
8	107.45	50.80	2	10
9	114.89	54.62	1	10
10	111.30	41.20	3	10
11	106.00	31.50	1	10
12	108.00	52.10	1	10
13	104.00	29.00	2	10

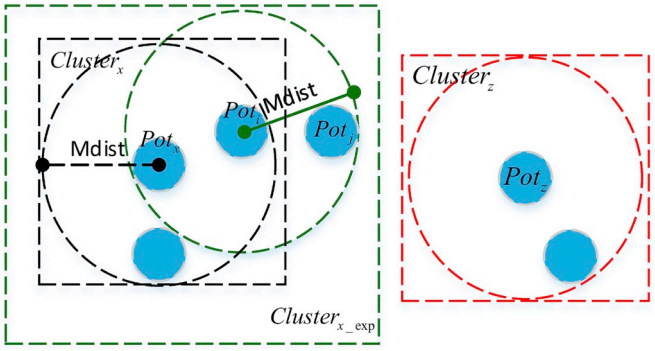

Fig. 2. Illustration of $Cluster_x$, $Cluster_{x\_\exp}$ $Cluster_z$.

Table 2
Result of task clustering.

number	1	2	3	4	5	6	7	8	9
Task sequence	2	3	9	10	4	1	[8,12]	[5,6]	[7,11,13]

After completing the above procedure, the target Pot_z that does not include in the set $Cluster_{x\_\exp}$ is selected as the center target, and the process of (1) (2) is repeated to obtain $Cluster_z$. And so on, the target set $Task_l = \{Pot_1, Pot_2, Pot_3, ..., Cluster_{x\_\exp}, Cluster_{y\_\exp}, ..., Cluster_{n\_\exp},\}$, which composed of targets and clustering sets, is finally obtained in Table 2. l represents the total number of tasks to be assigned.

2.2. Observation window

During the observation process, each target has a different visible window for different satellites. In order to obtain the visible window of the target as a constraint for subsequent schedule, the following coordinate systems are defined to properly describe the movement and the attitude of the satellite (see Fig. 3) [27].

According to the relative motion and attitude of satellites and targets, satellites and targets have the following relationships:

$$\mathbf{r}_t^d = \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = r_e \cdot \begin{bmatrix} \cos(\phi_i)\cdot\cos(\lambda_i) \\ \sin(\phi_i)\cdot\cos(\lambda_i) \\ \sin(\lambda_i) \end{bmatrix} \tag{3}$$

where, r_e denotes the radius of Earth in Eq. (3); $\mathbf{r}_t^d$ represents the position of a ground target $Pot_i(\lambda_i, \phi_i)$ in the Earth fixed coordinate.

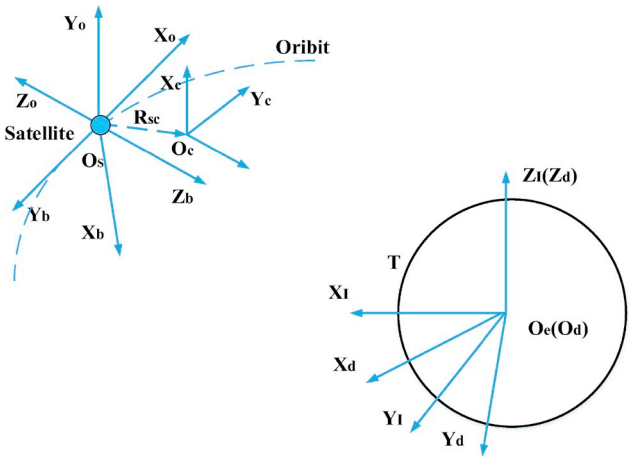

Fig. 3. Definition of coordinate systems.

1) The Earth-fixed coordinate system $O_e - x_d y_d z_d$: This coordinate system is to define the orientation of satellite and the positions of all the ground targets. The origin of this coordinate system is the geocentric O_e. The $O_e z_d$ axis points to the Earth's north pole, the $O_e x_d$ axis points to the intersection point between the meridian plane of Greenwich and the equator of the Earth, the $O_e y_d$ axis completes the right-hand rule.
2) The Earth inertial coordinate coordination system $O_e - x_I y_I z_I$: The coordination system can be used to calculate the orbit of satellite. From the orbit information, the observation window of all the targets can be determined.
3) The orbit coordinate system $O_s - x_o y_o z_o$: This coordinate system is used to describe the position of satellite in orbit. The $O_s z_o$ is collinear to the satellite position vector in the Earth inertial coordinate system. The $O_s x_o$ is perpendicular to $O_s z_o$ and points to the satellite's velocity direction and the $O_s y_o$ completes the right-hand rule.
4) The satellite body coordinate system $O_s - x_b y_b z_b$.: This coordinate system is defined to calculate the pointing direction of the instrument in the inertial space. The center of the coordinate locates at the center of mass of the satellite: The $O_s z_b$ direction is defined by the bore sight of the instrument. The $O_s y_b$ direction is defined by the rotational axis of spacecraft's solar panel and the $O_s x_b$ direction completes the right hand rule.

$$\mathbf{r}_s^I = \begin{bmatrix} x_s^I \\ y_s^I \\ z_s^I \end{bmatrix} \tag{4}$$

where, $\mathbf{r}_s^I$ stands for the position of satellite in the Earth inertial coordinate by Eq. (4);

$$\mathbf{r}_s^d = \begin{bmatrix} x_s^d \\ y_s^d \\ z_s^d \end{bmatrix} = \mathbf{A}_{Id}(t)\cdot \begin{bmatrix} x_s^I \\ y_s^I \\ z_s^I \end{bmatrix} \tag{5}$$

where, $\mathbf{r}_s^d$ denotes the position of satellite in Earth fixed coordinate in Eq. (5); $\mathbf{A}_{Id}$ stands for the transformation matrix from the Earth inertial coordinate to the Earth fixed coordinate.

$$\mathbf{r}_{st}^d = \mathbf{r}_t^d - \mathbf{r}_s^d \tag{6}$$

where, $\mathbf{r}_{st}^d$ represents relative position between the target and the satellite in the Earth fixed coordinate in Eq. (6).

$$\mathbf{r}_{st}^I = \mathbf{A}_{Id}{}^{-1}\cdot \begin{bmatrix} x_{st}^d \\ y_{st}^d \\ z_{st}^d \end{bmatrix} \tag{7}$$

where, $\mathbf{r}_{st}^I$ denotes the relative position between the target and the satellite in the inertial coordinate in Eq. (7);

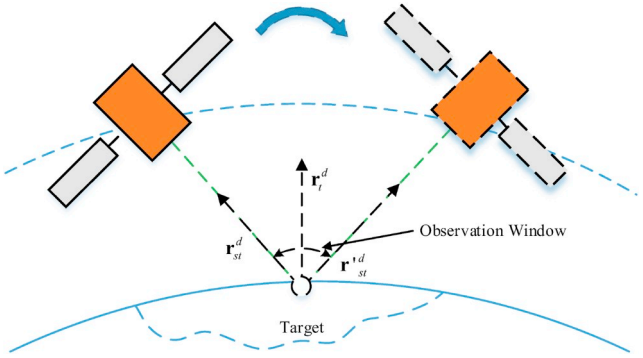

Fig. 4. Illustration of observation window.

$$\mathbf{r}_{st}^{b} = \mathbf{A}_{Ib}(\bar{\mathbf{q}})\cdot\mathbf{r}_{st}^{I} \tag{8}$$

where, $\mathbf{A}_{Ib}$ stands for the transformation matrix from the Earth inertial coordinate to the satellite body coordinate; $\bar{\mathbf{q}}$ represents the attitude of the satellite; $\mathbf{r}_{st}^{d}$ denotes the relative position between the target and the satellite in the satellite body coordinate in Eq. (8).

In order to ensure acceptable observation quality, the observable angle, defined as the angle between the sight of the bore in the instrument and the local normal direction of the ground target, should be limited to a certain range. The value of observable angle depends on the imaging capabilities and image quality requirements of the instrument. During the simulation of this paper, the maximum value of observable angle is set to be 50° [32]. Through the relative motion relationship between the satellite and the target, the angle between the targets is calculated by Eq. (9).

$$\alpha_{(\mathbf{r}_{st}^{d},\mathbf{r}'^{d}_{st})}(t) = angle(\mathbf{r}_{st}^{d}, \mathbf{r}'^{d}_{st}) \tag{9}$$

where, $angle(\cdot)$ indicates the angle between two vectors.

Fig. 4 depicts the definition of the observation window by illustrating the geometric relationship between the satellite and the ground target. When the observable angle between the target and satellite is smaller than the maximum observable angle, the target can be observed. Taking the point target Pot_i as an example, the st_i is defined as the start time of the time window when the target is observable. et_i represents the end time of the time window. The purpose of planning is to select appropriate start time s_i and end time e_i of target Pot_i within the observable time window TW_i.

$$s_i \geq st_i \;\&\&\; s_i + l_i \leq et_i \tag{10}$$

$$st_i,\ et_i \in TW_i \tag{11}$$

Eq. (10) (11) indicates that the observation start time s_i and observed time of the target l_i must be within the time window. The satellite needs to adjust the attitude when it switches targets. Therefore, in the planning process, the time for switching needs to be taken into consideration.

The transferring time $Shift(i, j)$ from Pot_i to Pot_j can be estimated according to Eq. (12)

$$Shift(i, j) = \alpha(\mathbf{r}_{st\_i}^{d}, \mathbf{r}_{st\_j}^{d})/\omega \tag{12}$$

where, $\alpha(\mathbf{r}_{st\_i}^{d}, \mathbf{r}_{st\_j}^{d})$ means the angle between two vectors in the satellite body coordinate. The satellite angular speed ω is related to the maneuverability of satellites and the value of ω is set to be 0.8°/s [34].

2.3. Evaluation

High-benefit tasks will be prioritized during the observation process. The planning evaluation includes three indicators: total benefit described in Eq. (13), tasks completion rate defined in Eq. (14) and computing time.

$$\sum_{i\in Task_{Completion}} p_i \tag{13}$$

$$\text{Completion rate} = \frac{\text{Number of completed tasks}}{\text{Total number of tasks}} \tag{14}$$

where, $Task_{Completion}$ represents set of tasks that can be completed.

3. Multi-agent autonomous mission allocation and planning

3.1. Multi-agent collaborative framework

A new generation of Earth observation satellites has a certain degree of onboard self-planning and communication capabilities among individuals. It can achieve the goals through a cooperative mode. Traditional collaborative frameworks lack of collaboration among subsatellites. Therefore, this paper designs a multi-dimensional and multi-Agent task cooperative structure model according to the theory of MAS [28,29], which combines centralized feedback and fully distributed control concepts.

In this model, Agent represents the Earth observation satellite, and several Agents form the MAS system. The entire system structure is divided into two layers: management layer and working layer. As shown in Fig. 5, the Agent receiving the command from the ground monitoring station is selected as the management Agent (the satellite "1″ in Fig. 5). The management Agent can interact with any other work-type Agent in two directions (shown by solid arrows in Fig. 5) in order to complete the negotiation, distribution, execution, and feedback of tasks. The work-type Agent is in the working layer ("Satellite 2″ in Fig. 5). The work-type Agent generally only interacts with the management Agent. Only under certain circumstances can it interact with other work-type Agents (dotted arrows in Fig. 5). The work-type Agent can be used as a backup of the management Agent and replace it while the management Agent fails.

On the one hand, this model realizes the interaction among all Agents, which reflects the flexibility and reliability of distributed cooperation. Meanwhile, setting constraints on the interaction avoids the complexity of the system and reduces the pressure of computation and communication. On the other hand, there is only one managed Agent in a single task, which reflects the high efficiency of centralized collaboration. At the same time, the work-type Agent can act as a backup of the managed Agent, which ensures the redundancy in the system. The multi-dimensional and multi-Agent task cooperative structure designed in this paper is a semi-open MAS framework. When it is applied, the interaction constraints must be set according to the negotiation mechanism adopted.

3.2. Task allocation rules and secondary allocation strategy

The CNP is a negotiation protocol proposed by Randall Davis and Reid G Smith to solve the task cooperation and distribution among the

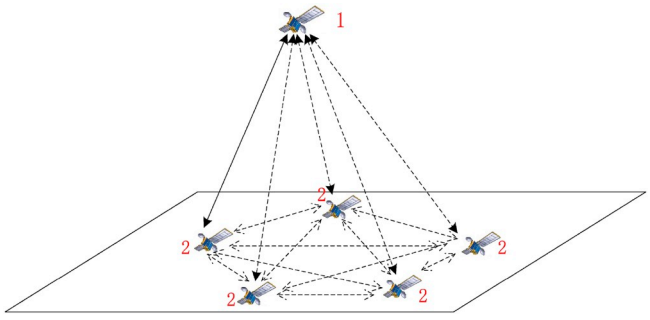

Fig. 5. Schematic of MDMA-CCM

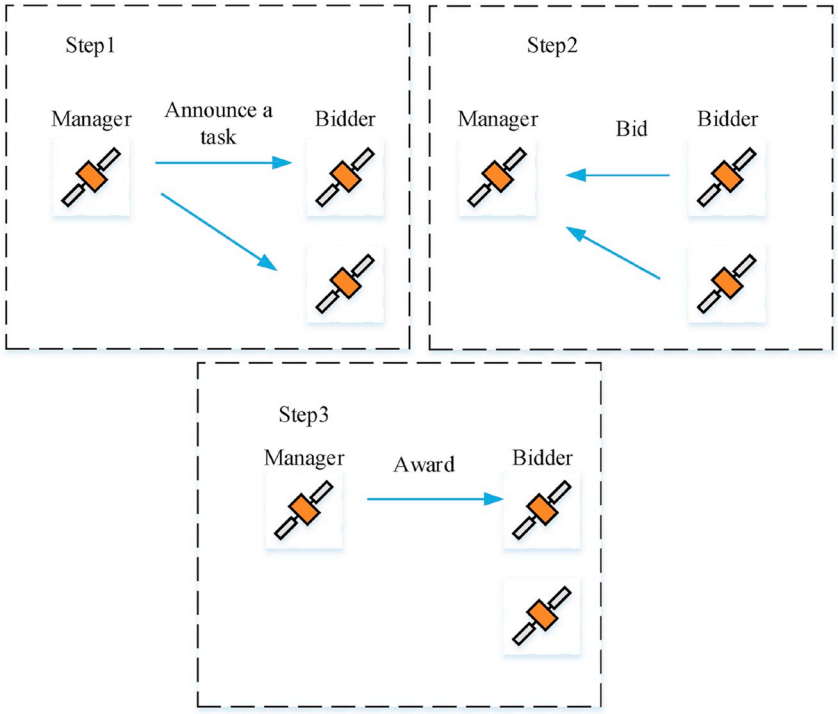

Fig. 6. Schematic of contract net protocol.

nodes of the distributed system [28–30]. It is widely used in the MAS system. There are three mainly steps as shown in Fig. 6.

Announcement: The management agent issues a notification to work-type Agents. The notification includes tasks description and tasks constraints.

Bidding: The work-type agent that receives the notification determines whether or not to select the task to be executed. Then, it returns a bid to the management Agent.

Awarding: After receiving the bid, the management Agents must choose a final task performer Agent. The result of choice is transmitted to the work-type agent.

Traditional observed satellite CNP system allocates only a single task, which is inefficient. Furthermore, due to the lack of negotiation and communication among work-type Agents, the task will be abandoned once the bid fails. This paper has made some improvements to the traditional CNP. Therefore, this paper proposes a new CNP based secondary negotiation strategy to overcome this issue.

3.2.1. Task announcement

The task announcement strategy is based on the benefit of the observation task. Management Agent publishes tender information to all work-type Agents $\{Sat_1, Sat_2, ..., Sat_M\}$. By clustering the target, the bidding round of the task can be effectively reduced during the bidding process, and the overall efficiency of the system is improved.

3.2.2. Bidding method

The task bid is completed by the work-type Agent. After receiving the released task information, the work-type Agent first judges whether it has the ability to bid according to the task and its own status. If it has the bidding capability, it further estimates the cost for executing the task and uses the relevant data as an indicator to bid. The bidding process is actually a process of capability judgment and task insertion. Before the new task was received, there had been already several tasks to be executed in the task execution set. These tasks occupy part of the time window of the Agent and have constraints on the related resources of the Agent. The Agent needs to determine whether the new task has an execution time, whether it meets constraints such as timing, resources, and whether it will cause conflicts with existing tasks. Agents have different strategies for evaluating different types of tasks.

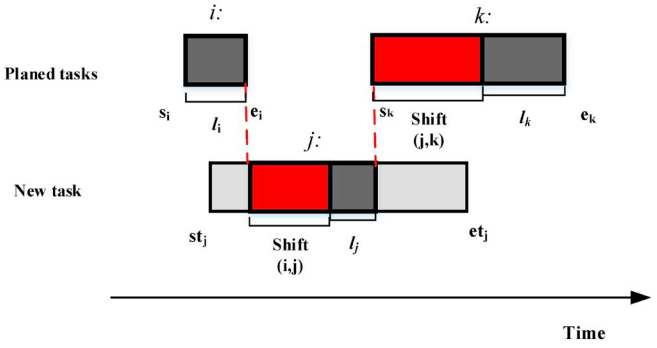

Fig. 7. Tasks insertion strategy schematic.

3.2.2.1. Point target bidding. As shown in Fig. 7, gray box represents the observable window of the point target. The newly inserted point target Pot_j with the existing scheduled task Pot_i and Pot_k must meet the relationship of Eq. (15) before being inserted into the observation sequence of the satellite.

$$e_i + Shift(i, j) + l_j + Shift(j, k) \le s_k \tag{15}$$

In the initial stage, the end time e_i of the task Pot_i is selected as the insertion time of the target Pot_j, and whether or not it can be inserted is judged by Eq. (15). If possible, place the target Pot_j in the observation sequence set Sat as in Eq. (16).

$$Sat = \{Pot_i | Pot_i \in Task_l, e_i + Shift(i, j) + l_j + Shift(j, k) \le s_k, \} \tag{16}$$

If the newly inserted target Pot_j collides with a planned task, the higher-benefit task is put into the observation sequence Sat by comparing the benefit value between them; then, the lower-benefit task is put into the conflict task set $Clash$. And the Agent's benefit increment and end time of the retention task are returned to the management Agent as bid evaluation information.

3.2.2.2. Task clustering set bidding

First, determine whether there are unobservable tasks, and put unobservable tasks into $Clash$ set. Second, judge if there are conflicting tasks, and if so, put the conflicting tasks into $Clash$ set. All remaining tasks make up the clusters set and all possible observation tasks sequences in the set are listed by an exhaustive method. $perms(Cluster_{x\_exp})$ represents the full arrangement of tasks in the $Cluster_{x\_exp} = \{Pot_i, Pot_j, Pot_k\}$ set. The result is shown by Eq. (17).

$$perms(Cluster_{x\_exp}) = \begin{Bmatrix} [Pot_i, Pot_j, Pot_k] \\ [Pot_i, Pot_k, Pot_j] \\ [Pot_j, Pot_i, Pot_k] \\ [Pot_j, Pot_k, Pot_i] \\ [Pot_k, Pot_i, Pot_j] \\ [Pot_k, Pot_j, Pot_i] \end{Bmatrix} \tag{17}$$

Third, calculate the time consuming of task sequences and select the sequence that takes the shortest time. The benefits of sequences and the time of the retention tasks are returned to the management Agent as bid evaluation information. The algorithm flow is shown in Fig. 8.

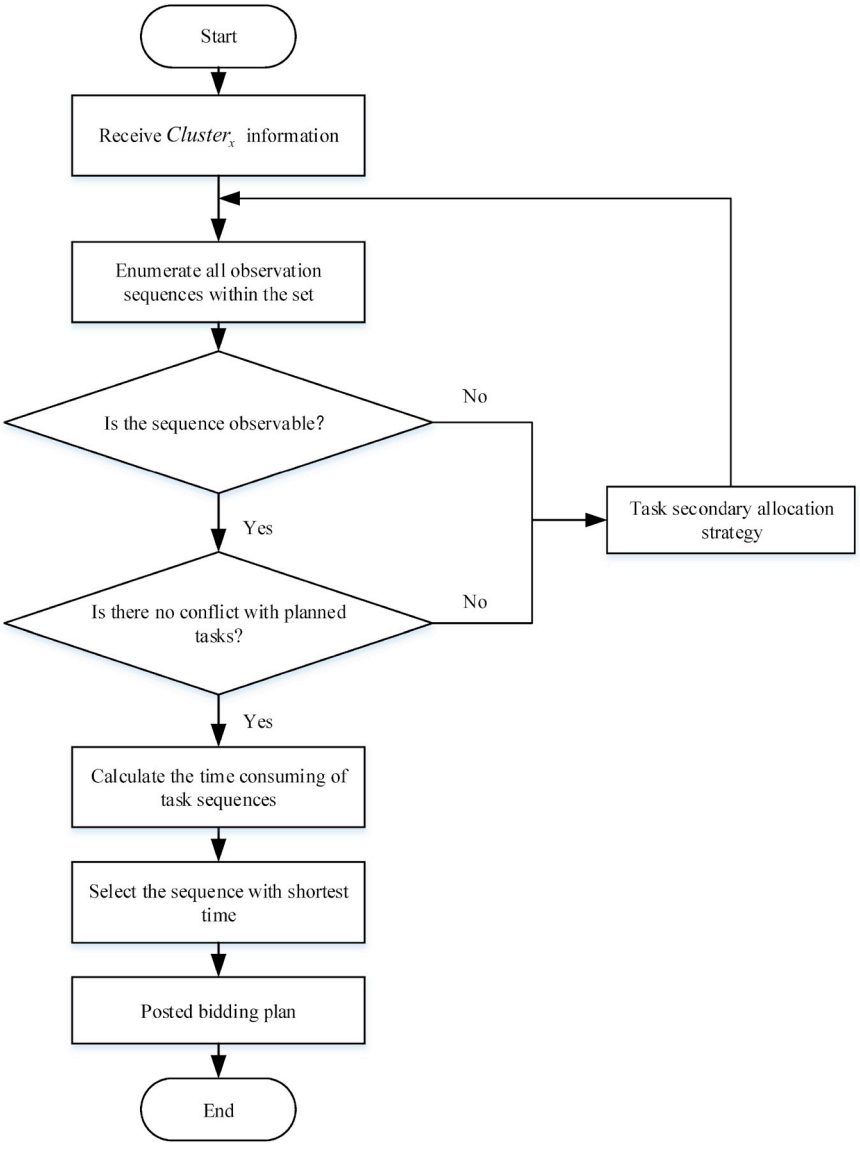

Fig. 8. Flow chart of cluster target planning.

3.2.3. Awarding and secondary allocation strategy (SAS)

If all work-type Agents are unable to complete the task, the bids for the second allocation will be evaluated. With the increase in benefit as the primary goal, the end time of the newly inserted task is the secondary goal, and the winner Agent is selected. The selected Agent acts as a manager Agent in the next round of bidding, and bids the targets in the conflicting task set *Clash* to other Agents in turn. Other work-type Agents decide whether to insert the task based on the revenue of the task to be assigned. If they can insert, they send bid information to management Agent; if they cannot, the bid fails. The management Agent will perform the second round allocation of the targets in the conflict set *Clash* according to the bid evaluation method of the point target. Meanwhile, the goals that cannot be allocated are discarded. After the target allocation in the *Clash* is completed, it returns to the previous round of $Task_l$ assignment.

After the management Agent publishes the tender information for a period of time, it will receive the bidding information from the work-type Agent. The following three results may occur: 1) none of Agent can complete the bid, indicating that no Agent satisfies the completion condition. After that, it will bid again through the task secondary allocation strategy; 2) only one Agent completes the bid and can only set it as a successful bidder, and there are no other candidates; 3) there are multiple Agents who complete the bid. At this time, multiple bidding need to be evaluated and screened. It is assumed that the number of agents that can complete the bid is m. Each bid proposal contains the following indicator $\{\Delta p_x, e_x, Wtime_x\}$, where Δp_x represents the increment of the Agent's revenue; e_x represents the end time of the new insertion target; $Wtime_x$ represents the total consuming time of the newly inserted target, which includes the camera pose transition adjustment time between the insertion target and the neighboring planned targets Pot_i and Pot_j, and the observation time l_x of the insertion target, as shown in Eq. (18).

$$Wtime_x = Shift(i, x) + l_x + Shift(x, j) \tag{18}$$

The detailed process of evaluation can be divided into the following four steps:

First, filtering out the maximum value of each indicator $\Delta p_{\max}$, $e_{\max}$, $Wtime_{\max}$;

Second, normalizing all indicators;

$$b\_submit_i = \begin{cases} norm\_p_i \\ norm\_e_i \\ norm\_Wtime_i \end{cases} = \begin{cases} p_i/p_{\max} \\ e_i/e_{\max} \\ Wtime_i/Wtime_{\max} \end{cases} \tag{19}$$

where, b_submit_i represents normalized indicators; $norm\_X$ denotes normalized item of X. Then, comprehensive indicators Bid_i can be computed according to Eq. (20):

$$Bid_i = \beta_p norm\_p_i - \beta_e norm\_e_i - \beta_t norm\_Wtime_i \tag{20}$$

where, β_p, β_e, β_t are the weights of the three indicators. In the bid evaluation strategy, different weight values can be selected according to different user needs;

Third, the cost $Bid_i (i = 1,2,...,m)$ of each task is sorted. The task bidding scheme with the lowest value (the lowest cost task execution scheme) can be used as the winning scheme Bid_win_j for the number j task.

Fourth, sending a successful bid notification to the winning Sat_i.

3.3. Interactively Re-planning method (IRM)

In the real-world space mission, satellites possibly get stuck with unforeseen situation such as failure of sensors, antennas or actuators, which leads to the stagnation of planned tasks. Therefore, it is essential to develop a re-planning method with uncertain emergency situations. This section proposes an Interactively Re-planning Method (IRM) consisting of tasks allocation and mission re-planning under the breakdown of sensors or actuators.

3.3.1. Principle

There are several advantages for using this method: Firstly, in the real-world scenarios, entire mission planning and allocation procedure cost plenty of time. However, it is inappropriate to overturn the whole scheduled scheme just for a provisional failed Agent. Using the IRM can simplify the processes as well as reduce the re-planning time. Secondly, instead of putting re-planning tasks into the end of task sequence with a traditional method, under the IRM tasks insert strategy, tasks can be added into vacancy time as far as possible. And if there are conflicts between tasks, high-value task will substitute for the low one in order to harvest greatest benefits of task replacement. All these advantages make the IRM as a rapid calculation and effective operation re-planning method.

3.3.2. Steps

During the re-planning produce, steps of using this method are as follows:

1) Once the emergency scenery has been determined, failed Agent builds connection with functional Agents. Then, failed Agents distribute assignment requests to health Agents one by one. If the faulty Agent is no longer capable of establishing a communication link with the functional agent, all tasks on the failed Agent will be lost.
2) After that a new bid round *b_bid* starts, when health Agents received assignment requests, they begin to evaluate executive capacity by $Judge(TW_i, FaultAgentTasks_i, HealthAgentTasks_j)$ judging whether a new task can be inserted into original task sequence. And then the task will be put into ahead of vacant time to make sure task carrying out forward. And work-type Agents will return *b_bid* with insertion time to assigning Agent. If there is a conflict between an original task and new one, accepting Agent will choose a task which is more important after abandoning less value one and then feed back a paradoxical index to assigning Agent.
3) As soon as failed Agent get feedbacks from health Agents, failed Agent allocates tasks by turn with *Evaluation*() that firstly considering the number of tasks completed then followed by the task insertion time. Finally, failed Agent distributes all its tasks to health Agents.

3.3.3. Variable definitions and pseudo-code

$$\begin{aligned} &HealthAgentTasks = \{Hpot_1, Hpot_2, ...\} \\ &FaultAgentTasks = \{Fpot_1, Fpot_2, ...\} \\ &Hpot_i = \{s_i, e_i\} \\ &Tpot_i = \{s_i, e_i\} \\ &TW = \{[st_1, et_1], [st_2, et_2], ...\} \\ &conf = \begin{cases} 1 & conflict \\ 0 & no\ \ conflict \end{cases} \end{aligned} \tag{21}$$

HealthAgentTasks means task set of *Hpot* to be execution on the normal Agent. *FaultAgentTasks* represents task set of *Fpot* on the failed Agent. $conf = 1$ means that there is a conflict between new task and existing tasks, and $conf = 0$ indicates there is no conflict.

The pseudo-code of IRM

Input: planned tasks number $AgentTasks$, planned tasks window $Tpot$, all tasks' observation window TW, number of failed Agent's tasks NE, number of health Agents NH.

Output: New plans NP

1: $Function\ X \leftarrow IRM(HealthAgentTasks, FaultAgentTasks, TW, NE, NH)$

2: $For\ i \leftarrow 1\ to\ NE$

3: $do\ \ index \leftarrow numel(FaultAgentTasks)$ /* $numel()$ returns number of elements in the $FaultAgentTasks$.*/

4: $For\ k \leftarrow 1\ to\ index$

5: $do\ \ numT \leftarrow FaultAgentTasks(k)$

6: $For\ \ j \leftarrow 1\ to\ NH$

7: $do\ \ \{conf_j, Tpot_j\} \leftarrow function(Judge\{TW, numT, HealthAgentTasks_j\})$ /*Judge if task $numT$ can be inserted in $HealthAgentTasks_j$. If it can, newly inserted time window is assigned to $Tpot$.*/

8: $End\ \ For$

9: $\{x_k, Top_k\} \leftarrow function(Evaluation\{[conf_1, conf_2, \mathrm{K}, conf_{NH}], [Tpot_1, Tpot_2, \mathrm{K}\ Tpot_{NH}]\})$ /*Select an optimal Agent x_k to perform task k.*/

10: $End\ For$

11: $X_i \leftarrow \{[x_1, Top_1], [x_2, Top_2], \ldots, [x_k, Top_k]\}$ /*Tasks on failed Agent i are reallocated.*/

12: $End\ For$

13: $NP \leftarrow [X_1, X_2, \ldots, X_{NE}]$

14: $\mathrm{Return}\ NP$

4. Simulation and analysis

In order to confirm the effectiveness of the planning algorithm developed in this paper, two simulation test cases are designed. In the first case, three satellites are cooperatively observed for 13 point targets. The performance of the algorithm is tested through this small size example, and the planning results are analyzed in detail. In the second case, the performance of the MDMA-CCM model, which designed in this paper, is compared with that of traditional centralized model talking. Therefore, the validity of the MDMA-CCM model can be discussed.

4.1. Algorithm performance test

This performance of the algorithm is verified through computer simulation. The simulation is realized in Matlab 2016a and computer parameters are Intel Core i5-6500 processor 3.20 GHz CPU and 8 GB RAM memory configuration. All the data of satellites and targets comes from Satellite Tool Kit 8.0. The Earth basic parameters of simulated experiment are given as follows: the radius of the Earth is 6378.13655 km, the Earth's radius curvature is 1/298.25722, and the Earth's gravitational constant is 398600.44 km^3/s^2. The point targets are distributed in an area in East Asia. The latitude and longitude coordinates of the points are shown in Table 1. The location of the point targets is shown in Fig. 9. Satellites orbit elements are shown in Table 3. The total time of scenario is set to be 800s.

The planning results are shown in Fig. 10 and Table 4. In the latitude and longitude coordinates, the blue circles represent the point targets; three dashed lines with different color represent the track of sub-satellite point of three satellites; the boxes of different colors represent the observation tasks assigned to different satellites, including the set of point tasks and cluster task points. The direction of movement of the satellite is from bottom to top in Fig. 10. It can be seen that the tasks assigned are mainly concentrated near the sub-satellite point. Because when the satellite observes tasks near the sub-satellite point,

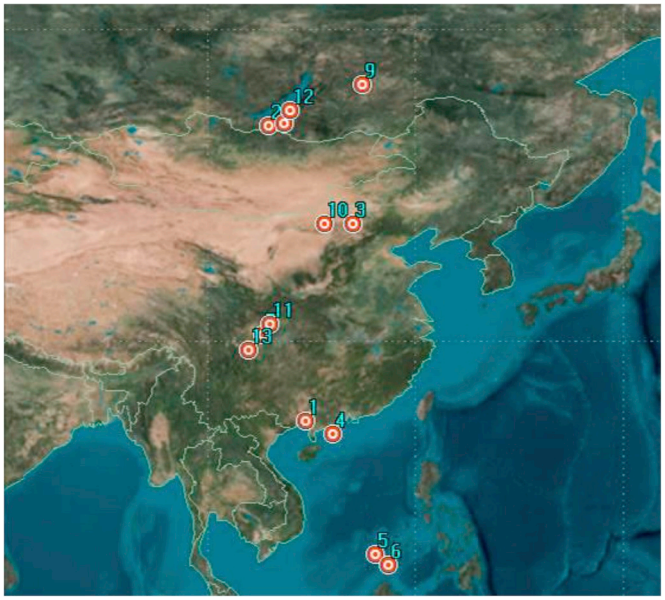

Fig. 9. Point targets location map.

Table 3
Satellites orbit elements.

Number	Semimajor axis (km)	Eccentricity	Inclination (degree)	Right ascension of ascending node (degree)	Argument of perigee (degree)	True anomaly (degree)
Satellite1	748	0.02	90	50.01	0	10
Satellite2	748	0.02	90	55.01	0	10
Satellite3	748	0.02	90	45.01	0	10

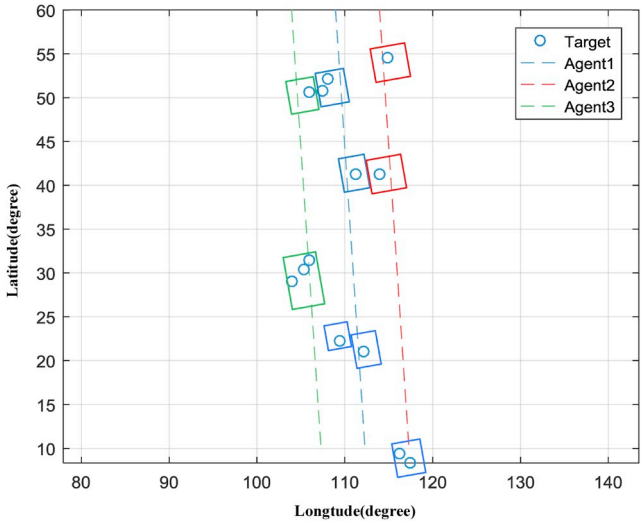

Fig. 10. Planning results.

Table 4
Agent assignment and observation time.

Number	Task sequence	Observation interval (s)	Benefit
Satellite 1	(6,5)→4→1→10 → (8,12)	[1,31] → [56,66] → [82,92] → [376,386] → [535,569]	22
Satellite 2	3→9	[378,388]→[595,605]	
Satellite 3	(13,7,11)→2	[188,242] → [528,538]	

the attitude adjustment time between different tasks is the shortest, which helps to reduce the overall observation time. Although the task set $\{Pot_6, Pot_5\}$ is close to the sub-satellite point trajectory of satellite 2, due to the observation angle, satellite 1 can complete the observation earlier than satellite 2. Therefore $\{Pot_6, Pot_5\}$ was allocated to satellite 1. It can be seen from Table 4 that the observation time of the clustering task is obviously longer than that of a single task. This is because the clustering task must not only consider the observation time of a single task, but also record the attitude adjustment time between different tasks to the total task time. Simulation experiments verify the effectiveness of MDMA-CCM model in solving MAS task allocation problems.

4.2. Re-planning simulation result

In this case, we employ the IRM to plan different simulation cases that share the same basic setting, including the coordinate system, satellites' orbit elements and targets' position. The detailed task information is given in Table 5. The different setting lies in the failed Agent's number. First, if all seven Agents work normally, we can get an initial assignment that depicts in Tables 6 and 7. Second, we assume that the Agents fail in turn and then assign the tasks of the failed Agents to the health Agents. Finally, do assess re-planning task completion rate. The relative position of satellite ground tracks and targets is shown in Fig. 11.

Table 5
Targets information.

number	longitude	latitude	priority	number	longitude	latitude	priority
1	110.81	14.61	1.00	51	120.90	17.78	1.00
2	110.29	17.14	3.00	52	121.02	18.21	2.00
3	110.98	51.06	3.00	53	128.89	28.21	3.00
4	109.31	51.61	2.00	54	128.46	29.15	3.00
5	120.79	32.63	2.00	55	124.78	56.55	3.00
6	119.42	34.33	2.00	56	127.32	55.63	3.00
7	134.63	47.57	1.00	57	93.01	19.11	1.00
8	134.35	47.14	3.00	58	95.99	22.90	3.00
9	105.75	15.05	1.00	59	119.73	55.69	1.00
10	107.25	15.90	3.00	60	119.57	56.82	1.00
11	110.33	29.69	1.00	61	99.47	30.11	3.00
12	110.81	28.74	2.00	62	100.96	30.88	2.00
13	122.16	20.57	2.00	63	87.61	44.83	2.00
14	120.51	22.49	1.00	64	87.10	44.14	2.00
15	124.33	22.18	2.00	65	112.73	55.27	1.00
16	124.70	22.71	1.00	66	113.67	56.85	1.00
17	113.91	20.78	1.00	67	124.59	18.02	1.00
18	114.31	23.02	1.00	68	122.50	17.22	3.00
19	99.75	54.73	1.00	69	108.73	13.72	1.00
20	98.66	55.88	3.00	70	109.59	13.40	1.00
21	91.34	37.41	2.00	71	105.07	49.48	2.00
22	91.69	36.78	3.00	72	104.37	50.30	2.00
23	96.24	54.33	3.00	73	130.31	27.51	1.00
24	94.42	50.71	1.00	74	129.54	27.20	2.00
25	94.98	32.38	1.00	75	116.37	37.63	3.00
26	95.68	31.32	2.00	76	116.50	37.61	3.00
27	94.33	20.51	2.00	77	94.79	50.29	1.00
28	96.83	18.41	3.00	78	95.07	48.16	2.00
29	124.66	22.27	1.00	79	89.07	20.03	2.00
30	123.11	21.05	3.00	80	88.46	15.03	2.00
31	127.49	40.10	2.00	81	105.12	41.19	3.00
32	126.85	36.40	3.00	82	108.07	42.15	3.00
33	117.51	55.38	2.00	83	118.85	14.09	1.00
34	118.44	56.31	1.00	84	118.37	12.73	1.00
35	102.51	52.93	1.00	85	121.04	37.49	2.00
36	103.60	52.91	2.00	86	121.78	38.24	3.00
37	115.12	56.65	1.00	87	102.93	30.45	2.00
38	115.98	57.40	2.00	88	104.81	29.83	3.00
39	113.25	52.99	1.00	89	130.83	58.63	1.00
40	110.88	52.97	3.00	90	132.61	57.17	1.00
41	109.74	46.52	3.00	91	132.44	17.03	3.00
42	112.39	46.07	1.00	92	131.57	17.90	3.00
43	133.35	40.09	2.00	93	105.55	51.71	2.00
44	130.28	38.07	2.00	94	106.73	52.62	2.00
45	122.58	48.76	1.00	95	105.02	23.89	2.00
46	122.07	49.11	1.00	96	103.75	26.57	3.00
47	112.62	35.36	1.00	97	99.27	50.76	3.00
48	112.70	33.77	3.00	98	97.38	53.32	1.00
49	89.64	19.22	3.00	99	102.42	25.95	1.00
50	91.38	20.04	2.00	100	102.17	25.22	1.00

Table 6
Original tasks planning results.

Agent Number	Tasks sequence
1	(1,2) → (17,18) → (12,11) → (48,47) →41→3→40
2	(76,75) →39→ (37,38)
3	10→95→88→81→82→ (71,72,93,94,36,35)
4	(51,52) →30→14→5→6→ (85,86) → (45,46) → (33,59,60,34)
5	(100,99) →96→61→ (97,19,20)
6	(91,92) → (15,29,16) → (74,73,53,54) →32→44→31→43→ (8,7) →56→55→90→89
7	57→ (27,28,58,26,25) → (64,63) →24→23

Table 7
Original tasks time sequence.

Agent Number	Time line
1	[1,33] → [57,108] → [178,207] → [259,295] → [462,472] → [536,546] → [568,578]
2	[319,342] → [569,579] →[628,655]
3	[1,11] → [105,115] → [197,207] → [376,386] → [395,405] → [510,625]
4	[8,30] → [58,68] → [82,92] → [239,249] → [269,279] → [317,344] → [502,525] → [609,669]
5	[123,146] → [148,158] → [203,213] → [531,630]
6	[29,56] → [82,119] → [162,220] → [300,310] → [341,351] → [360,370] → [410,420] → [533,557] → [614,624] → [626,636] → [663,673] → [675,685]
7	[45,55] → [56,251] → [474,497] → [530,540] → [590,600]

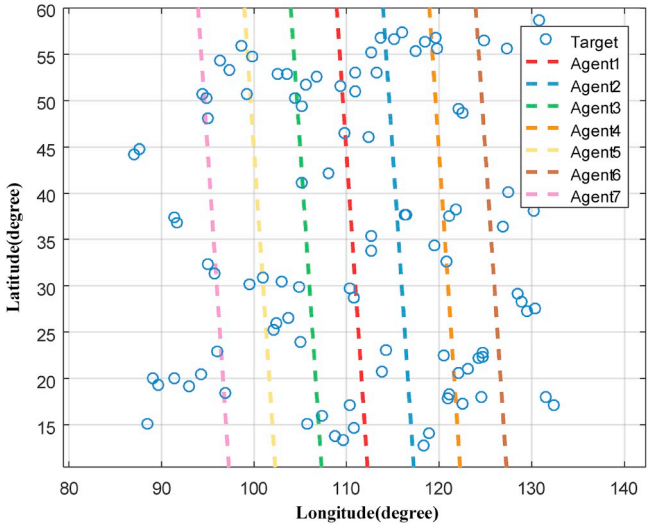

Fig. 11. Relative position of satellite ground tracks and targets.

Table 8
Re-planning result with Agent 1 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
2	(1,2), (12,11), (48,47),41,3	[15,76], [210,244], [268,303], [481,491], [544,554]	100%
4	(17,18), 40	[170,203], [607,617]	

Table 9
Re-planning result with Agent 2 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
1	(76,75), (37,38)	[345,369], [640,670]	100%
4	39	[558,598]	

When one of seven agents (numbered 1 to 7) fails, the re-planning results are shown from Table 8 through 14. The completion rate with different failed Agent is illustrated in Fig. 12. It can be seen from Fig. 12 that the completion rates for different Agents combinations are significantly different, and the IRM has a high re-planning completion rate when any one of the Agents 1 through 5 fails. The reason lies in the fact that they are in the middle of team and the adjacent Agents can help them complete tasks re-allocation, and the tasks in the marginal failed Agents 6 and 7 are difficult to re-plan. In general, the IRM can successfully reduce the losses caused by failures, although the effect of Agents re-planning depends on the location.

Table 10
Re-planning result with Agent 3 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
1	95,88, (71,72,93,94,36,35)	[155,165], [240,250], [396,406]	83.33%
5	10,81	[7,17], [390,400]	

Table 11
Re-planning result with Agent 4 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
1	(33,59,60,34)	[634,706]	85.71%
2	(51,52), 30, 5, 6	[21,43], [107,117], [258,268], [275,285]	
6	14, (45,46)	[128,138], [502,525]	

Table 12
Re-planning result with Agent 5 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
1	(97,19,20)	[600,685]	100%
3	(100,99), 96, 61	[144,167], [152,162], [245,259]	

Table 13
Re-planning result with Agent 6 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
2	56, 55	[701,711], [673,683]	33.33%
4	31, 90	[397,407], [723,733]	

Table 14
Re-planning result with Agent 7 failure.

Agent number	Re-planning Tasks number	Tasks re-planning time	Completion rate
1	23	[696,706]	25%

4.3. Comparison of the MDMA-CCM model with centralized MAS model

To demonstrate the time efficiency of the MDMA-CCM model developed in this paper, the comparison analysis is conducted between the new model and the original centralized MAS model with CNP, which assigns tasks one by one without task clustering and secondary

Fig. 12. The completion rate with different failed Agent.

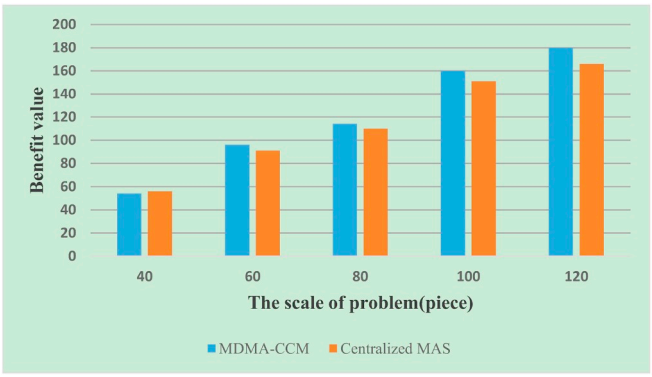

Fig. 13. Comparison of benefit values with different problem scales.

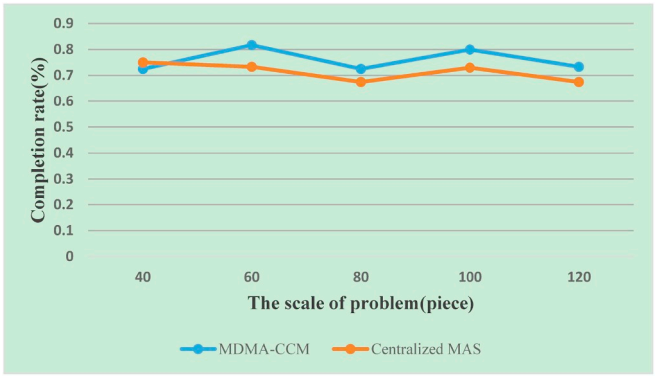

Fig. 14. Comparison of completion rates with different problem scales.

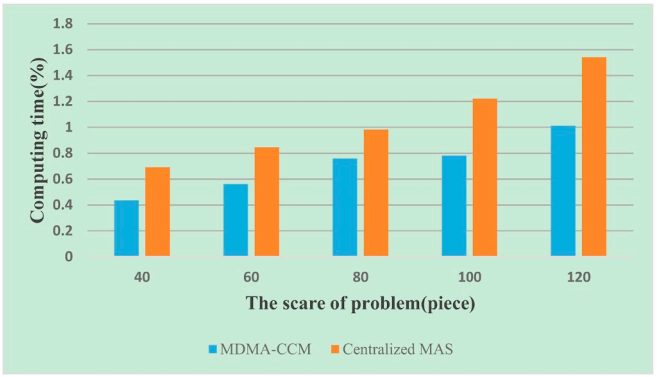

Fig. 15. Comparison of computing time with different problem scales.

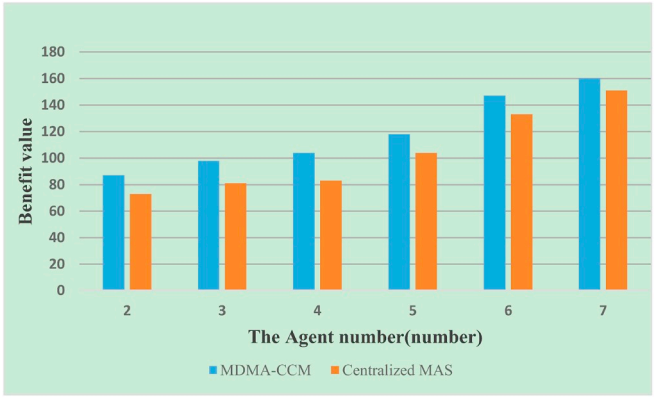

Fig. 16. Comparison of benefit values with different number of Agent.

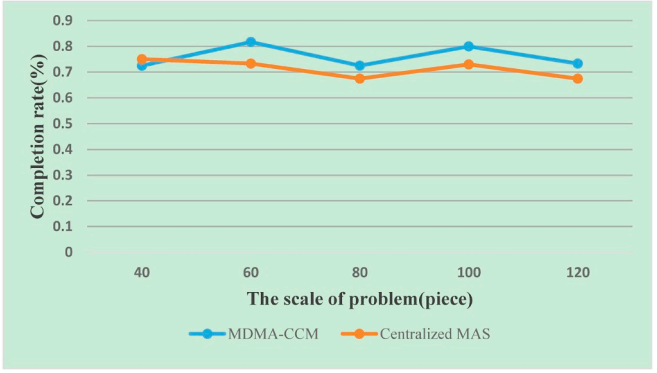

Fig. 17. Comparison of completion rates with different number of Agent.

allocation strategy. Simulation results show that both models can solve the MAS task allocation problem, but there is a certain gap in the solution performance.

Fig. 13 through 14 show the performance comparison between the MDMA-CCM model and the centralized MAS model under different task scales. As shown in Fig. 13, in the aspect of task benefit value, there is not much difference between the two methods in the small-scale example. However, when the scale of the task increases, MDMA-CCM's benefit value is higher than that of the centralized model, and the gap between the two models has an increasing trend. This is because it is possible to ensure the best possible planning result for each Agent by the exhaustive method in the MDMA-CCM model. As a result, the overall benefit value is increased when the task size increases. As shown in Fig. 14, the task completion rate of MDMA-CCM is higher than that of the centralized MAS. The completion rate of the two models has small fluctuations with the increase in the scale of the task. Fig. 15 shows the difference in calculation time between different models. Because of task clustering, the number of biddings of the management Agent can be reduced. When the task scale increases, the calculation time is decreased. Therefore, there is big difference between MDMA-CCM and centralized MAS in computing time.

The increase in the number of Agents will exacerbate the complexity of the system coordination. Fig. 16 shows the benefit value for the different number of Agents. Generally, MDMA-CCM is better than centralized MAS. When the Agents 3, 4, and 5 cooperate, MDMA-CCM has obvious advantages. As the number of Agents increases, the gap between the two models gradually decreases. Because increasing the number of Agents can alleviate the conflict between tasks and thus improve the task completion rate, the completion rate of the two models aggrandizes with the increase of the number of Agents. In addition, it can be noticed that the MDMA-CCM model maintains a stable advantage in Fig. 17.

By comparing the results of the two groups, the MDMA-CCM model not only has obvious advantages in calculation time, but also is superior to the centralized MAS in terms of benefit value and completion rate. Therefore, the MDMA-CCM model is good at multi-Agent collaboration

and large-scale problems, and it has a strong adaptability to the problem.

5. Conclusion

A new model is presented in this paper to handle multi-satellite collaboration and task allocation problem. The new developed model promotes the distribution efficiency of large-scale tasks through the task of clustering and redistribution. Meanwhile, the secondary distribution strategy based on the contract protocol network is designed to improve the completion rate of tasks. For solving the failed Agent re-planning problem, we proposed IRM to maximize task completion rates in the event of a failure. The simulation results show that compared with centralized MAS model, the new model proposed in this paper has an obvious advantage in computing time and the IRM could solve the emergency scenario re-planning problem successfully. Therefore, conclusions can be drawn that the new model has a good performance in Multi-satellite with massive scale targets. The availability of this onboard task allocation and re-planning system is not limited to the multi-satellite system. Other systems with autonomous multi-agents and similar requirements are also applicable.

Acknowledgments

The work described in this paper was supported by the National Natural Science Foundation of China (Grant No. 11672126), the National Key Research and Development Program of China (Grant No. 2016YFB0500801), the Opening Grant from the Key Laboratory of Space Utilization, Chinese Academy of Sciences (LSU-2016-04-01), and the Foundation of Graduate Innovation Center in NUAA (Grant No. kfjj20171509). The authors appreciate their financial supports.

References

[1] Valeriy V. Darnopykh, V.V. Malyshev, Operative planning of functional sessions for multi-satellite observation and communication systems, Acta Astronaut. 73 (2012) 193–205.
[2] Steven P. Neeck, T.J. Magner, G.E. Paules, NASA's small satellite missions for Earth observation, Acta Astronaut. 56 (1–2) (2005) 187–192.
[3] J. Baltié, E. Bensana, P. Fabiani, et al., Multi-vehicle missions:Architecture and algorithms for distributed online planning, Artificial Intelligence for Advanced Problem Solving Techniques, 2008.
[4] E. Bensana, Eric, Michel Lemaitre, Gerard Verfaillie, Earth observation satellite management, Constraints 4 (3) (1999) 293–299.
[5] E. Bensana, G. Verfaillie, Exact &inexact methods for daily management of earth observation satellite, Space Mission Operations and Ground Data Systems-SpaceOps 96 (1996) 394.
[6] Zhengqiang Zhang, Yuejin Tan, Junmin Wang, Research on MAS-based mission planning for distributed satellite system, J. Syst. Simul. 12 (2007) 058.
[7] Kutluhan Erol, J. Hendler, D.S. Nau, UMCP: a Sound and Complete Procedure for Hierarchical *Task-network Planning*, Aips, 1994, pp. 249–254.
[8] Ian Warfield, et al., Adaptation of hierarchical task network plans, Twentieth International Florida Artificial Intelligence Research Society Conference, Key West, Florida, USA DBLP, 2007, pp. 429–434.
[9] Chengpo Mu, Y. Li, An intrusion response decision-making model based on hierarchical task network planning, Expert Syst. Appl. 37 (3) (2010) 2465–2472.
[10] F. Xhafa, J. Sun, A. Barolli, et al., Genetic algorithms for satellite scheduling problems, Mobile Inf. Syst. 8 (4) (2012) 351–377.
[11] Fatos Xhafa, et al., A simulated annealing algorithm for router nodes placement problem in Wireless Mesh Networks, Simulat. Model. Pract. Theor. 19 (10) (2011) 2276–2284.
[12] Alan Zemel, F. Xhafa, G. Stahl, Analyzing the organization of collaborative math problem-solving in online chats using statistics and conversation analysis, International Conference on Collaboration and Technology Springer Berlin Heidelberg, 2005, pp. 271–283.
[13] A. Palis Michael, Jing Chiou Liou, David S.L. Wei, Task clustering and scheduling for distributed memory parallel architectures, IEEE Trans. Parallel Distr. Syst. 7 (1) (1996) 46–55.
[14] Mathias Dewatripont, Ian Jewitt, Tirole Jean, Multitask agency problems: focus and task clustering, Eur. Econ. Rev. 44 (4–6) (2000) 869–877.
[15] R. Smith, The contract net protocol: high level communication and control in distributed problem solver, IEEE Trans. Comput. 29 (1980) 1104–1113.
[16] Reid G. Smith, Randall Davis, Frameworks for cooperation in distributed problem solving, IEEE Trans. Syst. Man, Cybernetics 11 (1) (1981) 61–70.
[17] Reid G. Smith, Randall Davis, Distributed Problem Solving: the Contract Net Approach, Stanford Univ Ca Dept of Computer Science, 1978 No. Stan-Cs-78-667.
[18] Smith, The contract net protocol: high-level communication and control in a distributed problem solver, IEEE Trans. Comput. C-29 (12) (2006) 1104–1113.
[19] Li Gao, J.C. Sha, Research on task optimal allocation for distributed satellites system based on contract net protocol, J. Astronaut. 28 (2) (2009) 815–820.
[20] Junichi Kodama, et al., Multi-Agent-based autonomous power distribution network restoration using contract net protocol, Electr. Eng. Jpn. 166 (4) (2009) 56–63.
[21] Zhoupeng Zhu, Hao Chen, L.I. Jun, et al., Earth observing satellite cooperation mission scheduling based on feedback amendatory strategy, *Satellite* Payload Technology Academic Annual Meeting, XiAn, 2013, 2013, pp. 71–77.
[22] Peng Feng, et al., A method of distributed multi-satellite mission scheduling based on improved contract net protocol, Natural Computation (ICNC), 2015 11th International Conference on, IEEE, 2015.
[23] Zixuan Zheng, Jian Guo, Eberhard Gill, Swarm satellite mission scheduling & planning using hybrid dynamic mutation genetic algorithm, Acta Astronaut. 137 (2017) 243–253.
[24] Zixuan Zheng, Jian Guo, Eberhard Gilla, Multi-satellite onboard behaviour planning using adaptive genetic algorithm, 67th International Astronautical Congress (IAC), Guadalajara, Mexico, 2016, pp. 26–30 September.
[25] Zixuan Zheng, Jian Guo, Eberhard Gill, Onboard autonomous mission re-planning for multi-satellite system, Acta Astronaut. 145 (2018) 28–43.
[26] M. Lemaître, G. Verfaillie, F. Jouhaud, et al., Selecting and scheduling observations of agile satellites, Aero. Sci. Technol. 6 (5) (2002) 367–381.
[27] Yuchen She, Shuang Li, Yanbin Zhao, Onboard mission planning for agile satellite using modified mixed-integer linear programming, Aero. Sci. Technol. 72 (2018) 204–216.
[28] Thomas Schetter, Mark Campbell, Derek Surka, Multiple Agent-based autonomy for satellite constellations, Artif. Intell. 145 (2003) 147–180 1(2).
[29] Mark Campbell, Thomas Schetter, Comparison of multiple Agent-based organizations for satellite constellations, J. Spacecraft Rockets 39 (2) (2002) 274–283.
[30] Randall Davis, Reid G. Smith, Erman Lee, Negotiation as a metaphor for distributed problem solving, Readings in Distributed Artificial Intelligence, 1988, pp. 333–356.
[31] R. Xu, H. Chen, X. Liang, Priority-based constructive algorithms for scheduling agile earth observation satellites with total priority maximization, Expert Syst. Appl. 51 (C) (2016) 195–206.
[32] Yinan, Wu, Yu Zhang, Shuang Han, Image Motion compensation and MTF analyse of long array TDICCD space camera, Electron. Measure. Technol. 10 (2014) 018.