

Hierarchical path planning for multi-arm spacecraft with general translational and rotational locomotion mode

YUE ChengFei¹, LIN Tao², ZHANG Xiao¹, CHEN XueQin² & CAO XiBin^{2*}

¹*Institute of Space Science and Applied Technology, Harbin Institute of Technology Shenzhen, Shenzhen 518055, China;*

²*Research Center of the Satellite Technology, Harbin Institute of Technology, Harbin 150001, China*

Received July 26, 2022; accepted December 7, 2022; published online March 24, 2023

On-orbit construction and maintenance technology will play a significant role in future space exploration. The dexterous multi-functional spacecraft equipped with multi-arm, for instance, SpiderFab Bot, has attracted a great deal of focus due to its versatility in completing these missions. In such engineering practice, point-to-point moving in a complex environment is the fundamental issue. This paper investigates the three-dimensional point-to-point path planning problem, and a hierarchical path planning architecture is developed to give the trajectory of the multi-arm spacecraft effectively and efficiently. In the proposed 3-level architecture, the high-level planner generates the global constrained centric trajectory of the spacecraft with a rigid envelop assumption; the middle-level planner contributes the action sequence, a combination of the newly developed general translational and rotational locomotion mode, to cope with the relative position and attitude of the arms about the centroid of the spacecraft; the low-level planner maps the position/attitude of the end-effector of each arm from the operational space to the joint space optimally. Finally, the simulation experiment is carried out, and the results verify the effectiveness of the proposed three-layer architecture path planning strategy.

multi-arm spacecraft, path planning, hierarchical architecture, locomotion mode, translational locomotion, rotational locomotion

Citation: Yue C F, Lin T, Zhang X, et al. Hierarchical path planning for multi-arm spacecraft with general translational and rotational locomotion mode. Sci China Tech Sci, 2023, 66: 1180–1191, <https://doi.org/10.1007/s11431-022-2275-2>

1 Introduction

On-orbit service technology extends the lifetime, scale, and function of the space infrastructure through providing necessary maintenance and on-orbit assembly [1, 2], and it will play a significant role in future on-orbit construction of solar power stations, space telescopes and kilometre-scale apertures [3, 4]. Among these missions, the dexterous multi-functional spacecraft equipped with multi-arms shows their advantages in accommodating complex tasks with strict reliability, efficiency, and safety requirements [5, 6]. For

instance, SpiderFab Bot demonstrates its superiority with the capability of fabrication, assembly, and integration, which makes the complete in-space manufacturing process possible [7]. Yue et al. [8] have also proposed similar versatile spacecraft in completing various aerospace missions. In such applications, multi-level path planning to govern the position and attitude of the spacecraft, arms, and joints are the prerequisite.

Path planning has been investigated in the application of ground mobile robots extensively [9–13], and the planning method is governed by the mechanism of the robot itself [14] and the environment it suffers from. For a specific robot, the planning strategy is further divided into body-waist motion planning, leg motion planning, and toe trajectory optimiza-

*Corresponding author (email: xbcao@hit.edu.cn)

tion [15]. The body-waist motion describes the global position of the robot, and it generates redundant freedom to adapt to the rugged ground. The leg motion planning generates the gait of the robot, and it is characterized by the number of legs, the permutation of swing-stance legs, and the duty cycle in the gait executing period [16]. For different numbers of legs, flexibility and map adaptability of biped robots, quadrupedal robots [17], hexapod robots, and octopus robots [18] are different. The capability to adapt to various terrains can be enhanced by specific permutation of swing-stance legs and gait switching [19]. In addition to the mechanism of robot, another important character that influences the design of the planning is the environment, especially gravity. For the ground mobile robot, the center of the mass should be regulated to lie in the support polygon to keep static stability [20], and it restricts the locomotivity of the spacecraft in conducting on-orbit missions [21–23].

To maximize the moving ability, various space chain robots with multi-branches are proposed. For chain robots, the path planning is separated into two-end trajectory planning and solving the forward-inverse kinematics. In ref. [24], a 7-axis walking space manipulator, ReCoBot, is designed, and it constantly moves through a connection of its symmetric ends with the compatible structures. Multi-link robots and hyper-redundance robots, for instance, BogieBot [25] and MCDRs (multi-link cable-driven robots) [26], are proposed to enhance manoeuvrability in tight spaces or around obstacles. The redundant kinematic mechanism of such robots offers extra freedom in solving the forward-inverse optimization problem, and self-motion is introduced. For more complex motions such as periodical movement in certain terrain, middle-level planning, summarized as behaviour, is introduced between the end-to-end planning and solving the kinematics. You et al. [27, 28] Investigated the on-orbit truss climbing behaviour of cellular space robot. In ref. [27], three types of climbing behaviour are introduced, including horizontal climbing, vertical climbing, and rotating oblique climbing. Different climbing modes are realized through the elementary junctional cell and rotational cell. Other climbing behaviour of a modular biomimetic biped robot is summarized into inchworm climbing, twisting climbing, and flip climbing [29].

For multi-branch robots, the pose and position of the body should also be considered. Ramón et al. [30] present a humanoid biped robot with a body climbing on the handrails outside the international space station (ISS), and both the body arm motion and the arm motion are considered and optimized in completing a specific task. The complexity of behaviour planning increases with the number of branches. Two types of tumbling and inchworm climbing motion are

summarized for a three-branch climbing robot [31]. For motion planning of robots with more branches, Katherine [32] investigates the quadrupedal crawling robot, Tang et al. [33] analyze a hexapod truss-crawling robot, and Sun et al. [34] design a bioinspired controllable wing mechanism. These papers focus on the movement sequence of each leg, and the whole process is divided into lifting, dropping, and adhesion. To sum up, the motion character of the individual arm/leg and the relative relationship between each other has been investigated, while the posture of the body, the end-effector, and the overall relationship between the robot and the environment are neglected.

Motivated by the observations above, a systematic point-to-point path planning problem of a multi-armed spacecraft on a complex truss is investigated. A hierarchical path planning architecture is proposed, which factors the path planning problem into three levels. The high-level global position planning governs the position of the centroid of the spacecraft. The middle-level behaviour planning determines the locomotion type and regulates the relative attitude of the arms to the body center. The low-level planner maps the trajectory of the end-effector into the joint space considering potential collision avoidance.

(1) A systematic hierarchical path planning architecture containing global trajectory planning, local behaviour planning, and arm motion planning is proposed for the multi-arm spacecraft system. Through this architecture, all the motions of the body, arms, and joints are planned and synthesized.

(2) General translational and rotational locomotion modes are newly defined to cope with the high-efficient movement problem in a weightless environment and on discontinuous surfaces. Most coplanar motion can be achieved using the translational mode, while the motion between different surfaces is completed with a rotational maneuver.

(3) An all-process simulation is conducted and visualized, including the ant colony optimization based on neighborhood continuation search in global trajectory planning, gait sequence generation and optimization with minimum energy consumption index, and solving the optimized trajectory mapping with obstacle avoidance, to show the effectiveness of the proposed method in solving path planning problem.

This paper is organized as follows. Sect. 2 introduces the basic concepts of the research, which illustrates the model of the multi-arm spacecraft, map construction of the truss environment, and the representation of the investigated locomotion problem. Hierarchical path planning is studied in Sect. 3, including global path planning, behaviour planning and arm motion planning. Simulation results and analysis are demonstrated in Sect. 4. Sect. 5 presents conclusions and an outlook for future research work.

2 Preliminary and problem statement

This paper images an on-orbit manipulation scenario where a point-to-point movement of a multi-arm spacecraft is required on a structured truss of ISS. To solve this problem, preliminary concepts and representation of the spacecraft, environment, and the description of the investigated problem will be presented in this section.

2.1 Modeling of the multi-arm spacecraft

A multi-arm spacecraft consisting of a spacecraft base and four arms is considered as in Figure 1(a). Each arm has 7 degrees of freedom for dexterous manipulation. Assuming that a proper mapping between the operational space and the joint space has been built, the end-effector's pose and position can be guaranteed via optimization in the joint space with singularity and obstacle avoidance. In the path planning problem investigated in this paper, the joint space optimization can be separated from the planning phase, and the multi-arm spacecraft can be simplified into Figure 1(b). As shown in Figure 1(b), the simplified model consists of a cubic base, four variable-length connecting rods, and four end-effectors. The front side and back side of the base are distinguished according to the installed equipment and the corresponding function. For instance, lighting equipment and a camera may be employed on the front side for sensing, and the antenna may be installed in the back front for communication. This distinct feature form constraints in path planning. The connecting rods are symmetrically mounted around the base through spherical hinges, and the workspace of the original arm determines their length interval. The gripper is equipped as the end-effector, and it is regarded as a mass point. A base coordinate is defined at the base's geometric center to describe the base's position and attitude. Arm coordinates are also defined at each manipulator base position to describe the position and attitude of the end of the arm with respect to the arm base for joint angle solving.

In path planning, various configurations can be adopted. To simplify the planning process and accommodate the major vast of the mission, the default configuration of the spacecraft is designed as that the ends of arms are evenly distributed around the base, grabbing the four corners of the plane of the cell grid, simultaneously the front of the spacecraft faces the space station with a certain distance.

2.2 Representation of the 3-dimensional environment

To complete the path planning problem, the environment should be first built. By comparing the different features of scale map, topology map, and semantic map [35], grid map,

which is one kind of scale map, is adopted. It possesses the advantages of simple description and convenient adaptation of node-based search. Taking the ISS as an example, as shown in Figure 2(a), it contains a large number of truss structures, and the truss can be divided into cubic grid cells, whose length is set as 1 m according to the workspace of the multi-arm spacecraft studied in this paper. Furthermore, obstacles or infeasible cell is marked as occupied, and others are marked as free. In addition, a global coordinate fixed in the ISS is built to describe the position and attitude of the spacecraft as in Figure 2(b).

Focusing on path planning, the interaction between the multi-arm spacecraft and the environment is ignored, and we assume that the end gripper can grasp any node of the grid.

2.3 Description of point-to-point locomotion problem

The problem investigated in this paper can be described as a path planning problem of the simplified multi-arm spacecraft as in Figure 1(b) from an initial grid to a target grid in a 3D raster map shown in Figure 2(b), i.e., the point-to-point locomotion problem. Mathematically, the problem can be modeled as finding a feasible path $P_{\text{global}}(q)$ from the initial position P_{initial} to the target position P_{target} , with q being the arm joint angle.

In this problem, some constraints are considered: (1) The spacecraft cannot be completely floating, and there must exist at least one point adhering to the truss. In other words, the duty cycle D of N legs should satisfy equation: $D(t) = \frac{\sum_i^N P_{s,i}(t)}{N \cdot T} > 0, \forall t \in T$, where D represents the duty cycle and

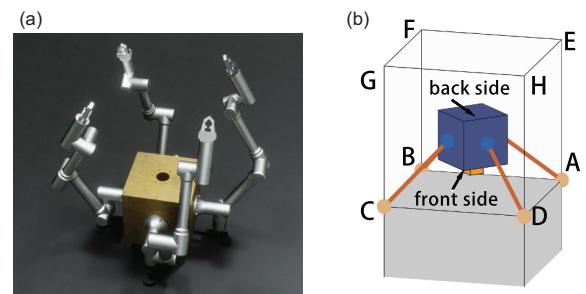


Figure 1 (Color online) (a) Conceptual model and (b) simplified model of multi-arm spacecraft.

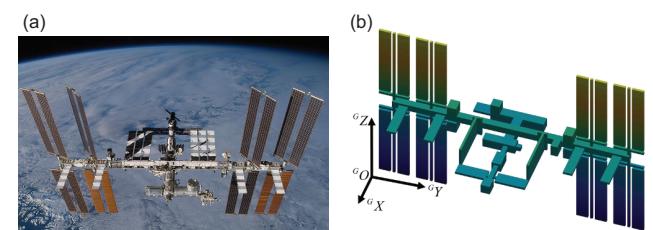


Figure 2 (Color online) (a) International space station and (b) the grid map of ISS.

$P_{s,i}$ is the period of leg i in support phase. (2) The front side of the spacecraft must face to the target plane, which the rotation change number η satisfies: $\sum \eta_i \in \{x \mid x = 2k, k \in N\}$, where η represents the concavity and convexity of the rotation. This constraint means that the spacecraft will not be upside-down at the end of each rotation step. (3) The arms of the spacecraft cannot collide with the environment and each other, which includes the global path cannot cross obstacle (refers to eqs. (1) and (2)), motion sequences through obstacle should be omitted (refers to Sect. 3.2.2), no geometrical overlap occurs in the trajectories of the arms (refers to eq. (17)). In addition, some optimization indexes are also considered, such as consuming as little energy as possible during the locomotion ($\min(Q)$ in eq. (12)).

3 Hierarchical path planning

The essence of multi-arm spacecraft locomotion planning in a complex frame is to generate a series of end trajectories so that the spacecraft can reach the destination according to the desired attitude and consume as little energy as possible. Due to the complexity of the task, we propose a hierarchical planning architecture as shown in Figure 3, which includes three levels.

- **High-level planning: global position planning**, which is responsible for searching for a global path in a 3D grid map from the initial location to the target location. In this phase, the base and the arms of the spacecraft are considered

as a whole, and the envelope of the system is adapted to check whether the spacecraft can pass through while adhering to the truss. Then a globally constrained centric reference trajectory of the whole system is generated.

- **Mid-level planning: local behaviour planning**, the role is to plan the local locomotion mode between the adjacent point of global paths. The locomotion mode, namely action, describes the relative attitude of the arms with respect to the body center, and the successive action sequence forms the behaviour of the spacecraft. In this phase, the major vast of the coplanar locomotion is unified by a translational mode, and the “jump” between different planes is realized by rotational mode. For the minority of the coplanar movement, rotational locomotion is adopted to straddle the gap. Then a series of actions is generated, and hence, the trajectory of the end-effector of each arm is determined.

- **Low-level planning: arm motion planning**, which is used to generate the joint trajectory when both the base and the end are fixed. This part maps the state from the operational space into the joint space by solving the forward-inverse kinematics. It has been investigated extensively in the chain robot and serial manipulator. In the paper, we will describe the method we adopted, the constraints we considered, and the results we obtained.

3.1 High-level: global position planning

On this level, the initial position and the target position are

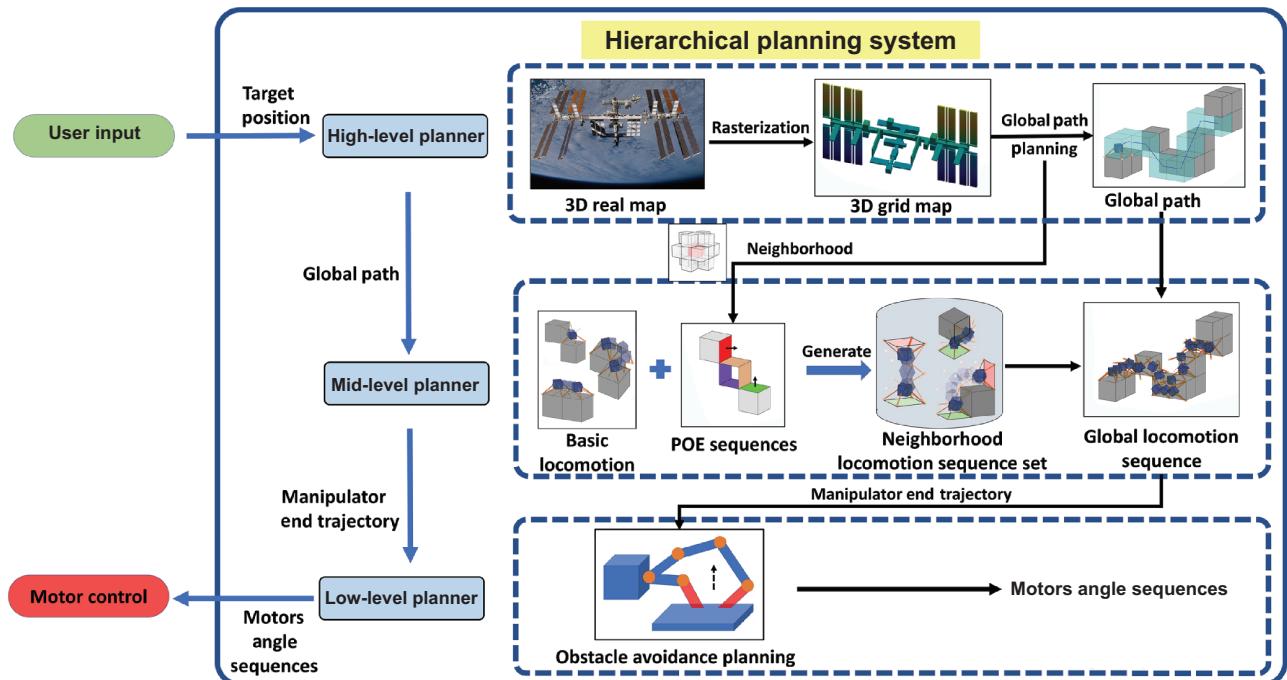


Figure 3 (Color online) The frame of hierarchical path planning.

given as input, and the global centric trajectory of the spacecraft is generated as output. For high-efficient planning and search, a modified ant colony optimization method based on neighborhood continuation search (ACOBNCS) is proposed [36]. Three key steps of the feasible node search strategy, feasible node probability transfer rule, and pheromone update law are presented.

3.1.1 Pre-processing of environment

In this part, we assumed that the arms and the base are enclosed by a virtual spherical ball [37] as shown in Figure 4. Assume that the minimum rigid sphere model radius for the folded case of the spacecraft is r_{\min} and the radius of the free maneuvering space is r_{\max} . Then the distance between the center of the rigid sphere model and obstacle r should satisfy:

$$r_{\min} \leq r \leq r_{\max}, \quad (1)$$

$$\mathbf{r} = \mathbf{p}_o - \mathbf{p}_{rbc}, \quad (2)$$

where \mathbf{p}_o is the position of obstacle, \mathbf{p}_{rbc} is the position of base center.

According to eqs. (1) and (2):

$$\mathbf{p}_o - r_{\max} \leq \mathbf{p}_{rbc} \leq \mathbf{p}_o - r_{rb}, \quad (3)$$

which means the base center is constrained between the obstacle and the radius for the folded case and free maneuvering space. Then we close the infeasible space by marking the grid as occupied, and consequently, the feasible space is reduced. Instead of judging the distance during the search, this pre-processing of the environment can improve search efficiency.

3.1.2 Feasible node search strategy

To further enhance the search efficiency, additional feasible nodes are added to the search set. Imaging the current node is P_i , then potential node P_j is added into the search set S_{pm} :

- P_i and P_j are adjacent to each other or colinear;
- P_j is not an obstacle and there is no obstacle between P_i and P_j ;

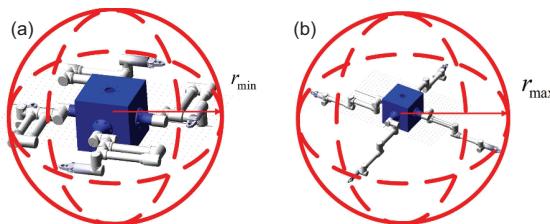


Figure 4 (Color online) (a) Minimum and (b) maximum virtual envelop of the multi-armed spacecraft.

- P_j is colinear with P_i and adjacent to a obstacle for adhering to when P_i and P_j are not a neighbor of each other.

By the above modification, the potential node is enlarged, and the computing time is shortened with the sacrifice of computing space. Thus the number of the iteration step and computational efficiency are improved. The detail of this strategy and the whole algorithm can be found in ref. [36].

3.1.3 Feasible node probability transfer rule

This paper adopts the ACOBNCs method to generate the feasible node. Using this method, any ant, for instance, the m th ant, moves from current node P_i to the next node P_j follows a probability distribution, and the transfer probabilities p_{ij}^m can be calculated as

$$p_{ij}^m = \begin{cases} \frac{\tau_{ij}^\alpha(t) \cdot \eta_{ij}^\beta(t)}{\sum_{j=1}^m \tau_{ij}^\alpha(t)^\alpha \cdot \eta_{ij}^\beta(t)^\beta}, & P_j \in S_{pm}, \\ 0, & P_j \notin S_{pm}, \end{cases} \quad (4)$$

with

$$\eta_{ij} = \frac{1}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}},$$

where $\tau_{ij}(t)$ represents the pheromone concentration from P_i to P_j , $\eta_{ij}(t)$ represents the heuristic information between P_i to P_j , taken as the inverse of the distance between P_i and P_j . α is the information heuristic factor and β is the heuristic function factor.

3.1.4 Pheromone update law

After the m th ant moves from P_i to P_j , the pheromone concentration $\tau_{ij}(t)$ is updated. By reducing the pheromone concentration on the traveled path, solutions falling into a local optimum are avoided. The updating law of τ_{ij} is formulated as

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k, \quad (5)$$

$$\Delta\tau_{ij}^k = \begin{cases} 0, & P_j \notin Route, \\ \frac{Q}{L_k}, & P_j \in Route, \end{cases} \quad (6)$$

where ρ ($0 < \rho < 1$) is the pheromone volatility coefficient, $\Delta\tau_{ij}^k$ is the pheromone concentration increment from P_i to P_j of the k th generation of ants, Q is the pheromone intensity, L_k is the length of the global optimal path searched by the k th generation of ants, and, $Route$ is the node set generated by the previous $k-1$ generations, and it will be the output of the global position planning when the ACOBNCs converges.

3.2 Middle-level: local behavior planning

3.2.1 Basic locomotion type design

A multi-arm spacecraft can move in various ways, such as jet maneuvering like a conventional satellite or crawling like a ground quadrobot. Among these forms of motion, changing the position and adjusting its posture to suit different working environments are fundamental. Further considering the reliability and economic efficiency, at least one end of the spacecraft should grasp the environmental surface. Furthermore, the moving process should consume as little energy as possible to avoid frequent replenishment.

Based on the above reasons, we propose two basic locomotion types, translational mode, and rotational mode. Translational locomotion allows the spacecraft to move in its current plane, like ground mobile robots, without changing its attitude. As a complement, rotational locomotion changes its position and posture simultaneously about a specific axis composed of grabbing points. Then the spacecraft can move between different planes and on discontinuous surfaces.

(1) Translational locomotion

Translational locomotion is the motion that moves the spacecraft a specified distance in one direction in the plane of the arm ends, POE for short, as the X - Y plane in Figure 5. The attitude of the spacecraft does not change during this motion. The whole process can be described by a function $\text{Translation}(\text{direction}, l)$. The **direction** means the direction that the robot moves forward to, and it can be chosen as $\pm X$ and $\pm Y$ as shown in Figure 5; l represents the spacecraft translational distance.

Translational locomotion can be divided into two phases: the stretching phase and the shrinking phase. In the stretching phase, the two front arms at the side of motion direction, arm 1 and arm 4 in Figure 5(a), release their end-effector, move l along the POE and tighten end-effectors again. During this phase, the base of the spacecraft moves $\frac{l}{2}$. In the shrinking phase, the other two arm ends at the opposite side and act like the previous two, and the base moves another $\frac{l}{2}$, see in Figure 5(b).

For control purposes, the motion trajectories of the arm ends and the base are discretized at N sampling points.

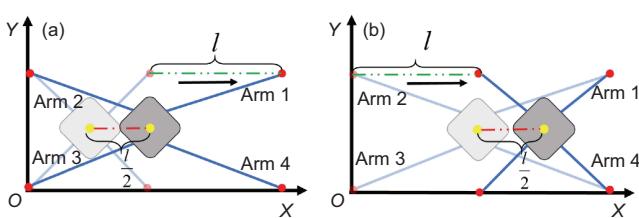


Figure 5 (Color online) (a) Stretching phase and (b) shrinking phase of spacecraft translational locomotion.

Taking $\text{Translation}(+X, 1)$ (translate one meter in the $+X$ direction) as an example, the trajectory of the foot end of the first arm P_1 , the second arm P_2 and the trajectory of the base P_{base} at the i th sampling point can be expressed as follows:

$$\mathbf{p}_{1,i} = \begin{cases} \left[\frac{1}{2} + \frac{2il}{N}, \frac{1}{2}, -\frac{1}{2} \right]^T, & i \in [0, \frac{2}{N}], \\ \left[\frac{1}{2} + l, \frac{1}{2}, -\frac{1}{2} \right]^T, & i \in [\frac{2}{N} + 1, N], \end{cases} \quad (7)$$

$$\mathbf{p}_{2,i} = \begin{cases} \left[-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2} \right]^T, & i \in [0, \frac{2}{N}], \\ \left[-\frac{1}{2} + \frac{2l}{N} \left(i - \frac{N}{2} \right), \frac{1}{2}, -\frac{1}{2} \right]^T, & i \in [\frac{2}{N} + 1, N], \end{cases} \quad (8)$$

$$\mathbf{p}_{\text{base},i} = \left[\frac{i}{N} L, 0, 0 \right]^T, \quad i \in [0, N]. \quad (9)$$

(2) Rotational locomotion

Rotational locomotion is the motion that changes the attitude of spacecraft. According to the relationship of POE before and after the rotation, two kinds of rotation can be defined, i.e., POE rotation and rotation on the POE, see Figure 6. For the POE rotation, the spacecraft rotates about one axis through two arm ends, and the POE changes after rotation. For the rotation on POE, the spacecraft rotate about a perpendicular axis of the POE from one of the arm ends, and the POE keeps unchanged after the rotation. All of these rotational locomotions can be summarized as a function $\text{Rotation}(\text{axis}, \theta, \eta)$, where axis is the symmetric axis rotating about, θ is the rotational angle, and η represents the concavity and convexity. When $\eta = 0$, the spacecraft configuration is convex and stays at the same side of the POE. When $\eta = 1$, the configuration is concave, and the spacecraft should conduct a mirror flip about the POE as in Figure 6. During the whole process, the basic configuration of the spacecraft keeps fixed. For example, the spanning L of adjacent arms and the dihedral angle ϕ of POE and the side surface of two arms remain constant.

For the POE rotation, the rotational axis can be an arbitrary edge shown in Figure 1(b), for instance, (A, \vec{AB}) and its counterpart (B, \vec{BA}) . One of the sample rotational locomotion $\text{Rotation}((B, \vec{BA}), \theta, 0)$ of Figure 6(a) can be expressed as

$$\begin{aligned} \mathbf{p}_{1,i} &= \left[-\frac{1}{2} + L \cos\left(\frac{\theta i}{N}\right), \frac{1}{2}, -\frac{1}{2} + L \sin\left(\frac{\theta i}{N}\right) \right]^T, & i \in [0, N], \\ \mathbf{p}_{2,i} &= \left[-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2} \right]^T, & i \in [0, N], \\ \mathbf{p}_{\text{base},i} &= \left[-\frac{1}{2} + L_{\text{base}} \cos(\phi_i); 0; -\frac{1}{2} + L_{\text{base}} \sin(\phi_i) \right], \\ &\quad i \in [0, N], \end{aligned} \quad (10)$$

$$\text{with } \phi_i = \phi + \frac{\theta_{\text{base}} i}{N}.$$

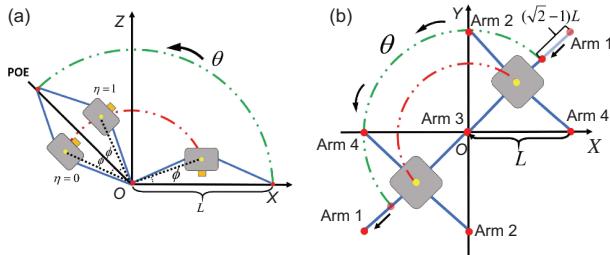


Figure 6 (Color online) POE of spacecraft rotational locomotion on (a) (B, \overrightarrow{BC}) and (b) (C, \overrightarrow{CG}) .

For the rotation on POE, the rotational axis can be $(A, \overrightarrow{AE}), (B, \overrightarrow{BF}), (C, \overrightarrow{CG}), (D, \overrightarrow{DH})$ and their corresponding counterpart as in Figure 1(b). In this type of rotational locomotion, the diagonal distance is $\sqrt{2}$ times the adjacent edge. Thus the far-end arm should conduct a shrinking movement of $-(\sqrt{2}-1)L$ before the rotation and a stretching movement of $(\sqrt{2}-1)L$ to avoid a potential collision. Also, in the rotation type, η keeps zero since the spacecraft always lie on the same side of the POE. Taking $\text{Rotation}((C, \overrightarrow{CG}), \theta, 0)$ shown in Figure 6(b) as an example, the end position and base position can be obtained as

$$\begin{aligned} P_{1,i} &= \begin{cases} [aL, aL, 0]^T, & i \in \left[0, \frac{N}{4}\right], \\ [\sqrt{2} \cos(b\pi)L, \sqrt{2} \sin(b\pi)L, 0]^T, & i \in \left[\frac{N}{4}, \frac{3N}{4}\right], \\ [cL, cL, 0]^T, & i \in \left[\frac{3N}{4}, N\right], \end{cases} \\ P_{2,i} &= \left[-\sin\left(\frac{i}{N}\theta\right)L, \cos\left(\frac{i}{N}\theta\right)L, 0\right]^T, \quad i \in [0, N], \\ P_{3,i} &= [0, 0, 0]^T, \quad i \in [0, N], \\ P_{4,i} &= \left[\cos\left(\frac{i}{N}\theta\right)L, \sin\left(\frac{i}{N}\theta\right)L, 0\right]^T, \quad i \in [0, N], \\ P_{\text{base }, i} &= \left[\frac{\sqrt{2}}{2} \cos\left(\frac{\pi}{4} + \frac{i}{N}\theta\right)L, \frac{\sqrt{2}}{2} \sin\left(\frac{\pi}{4} + \frac{i}{N}\theta\right)L, \frac{1}{2}\right]^T, \quad i \in [0, N], \end{aligned} \quad (11)$$

with $a = 1 - \frac{4(\sqrt{2}-1)i}{N}$, $b = \frac{2i}{N} - \frac{1}{4}$ and $c = -1 - (\sqrt{2}-1)\frac{4i-3N}{N}$.

3.2.2 Local locomotion sequence generation

(1) POE sequence generation

Tasks moving along a global path can be further broken down into successive moving between adjacent cells. According to the previously planned global grid path, this section generates the POE, one of the directional faces of the

adjacent grids. Considering the current position of the spacecraft in the middle as in Figure 7, A-star algorithm is adopted to generate the POE sequence, which is reachable by one-step rotational locomotion or translational locomotion.

As an example, take the plane furthest from the spacecraft in the upper left cell as the target plane. The search results are shown in Figure 8. The upper two planes shown as POE sequence 1 and the lower two planes shown as POE sequence 2 are the obtained two different POE sequences, respectively. Similarly, as many paths as possible are generated in this step to guarantee that the optimized one is obtained in these paths.

(2) Action sequence set generation

Many locomotion sequences are equivalent in POE transfer. As shown in the Figure 9, these four rotations can reach the same POE. However, the trajectories and final states are different. All feasible sequences are needed for comparison to obtain the optimal action sequence. In generating an action sequence, some constraints should be considered:

Position constraint: The target plane corresponds to two adjacent grids. Those sequences that lead to a non-target grid will be omitted. In Figure 10(a), the spacecraft reaches the target plane but is located in a non-target grid, so this sequence should be omitted.

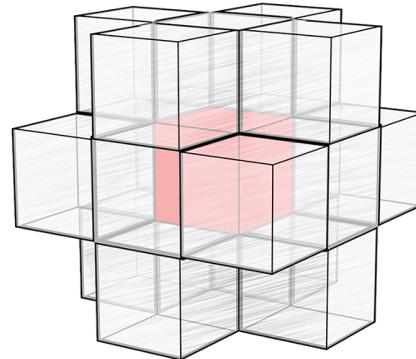


Figure 7 (Color online) Neighborhood used in the POE searching.

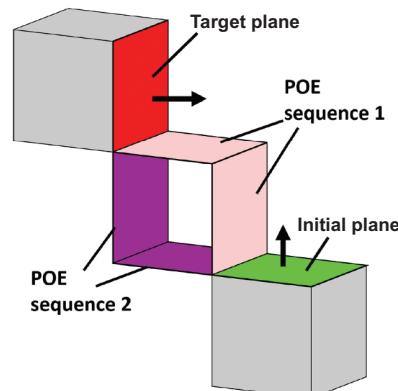


Figure 8 (Color online) Results of POE sequence.

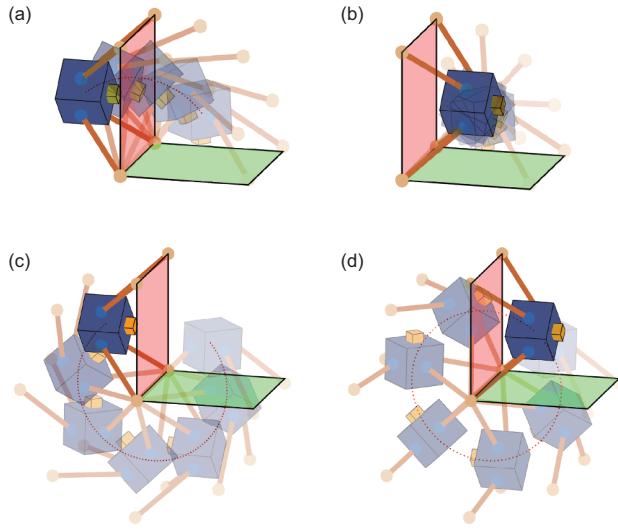


Figure 9 (Color online) Four different rotations between two POEs. (a) Anticlockwise convex rotation; (b) anticlockwise concave rotation; (c) clockwise convex rotation; (d) clockwise concave rotation.

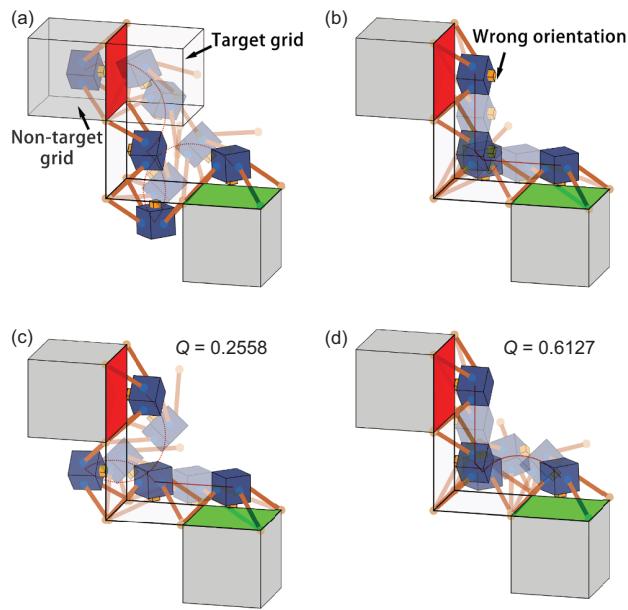


Figure 10 (Color online) Different locomotion sequences between same POE. (a), (b), (c), (d) sequence A, B, C, D.

Front side orientation constraint: In the practical working scenario, the side of the spacecraft equipped with a camera needs to face the working surface. Therefore, after the spacecraft transfer, the front side of the spacecraft needs to face the space station's surface. As shown in Figure 10(b), the spacecraft reaches the target plane, but the front side faces away from the target plane, so this sequence should also be omitted. According to the definition of the **Rotation**, the change of η corresponds to the exchange of the back side and the front side. Thus the summation of η for the whole se-

quence should be even to fulfill the requirement of the front-side orientation constraint.

Collision constraint: Because the action sequence is generated according to the POE sequence, different action sequences may cause the spacecraft to pass through different cells. The action sequences that make the spacecraft pass through the obstacle grid will be omitted.

For the same POE shown in the Figure 10, there are multiple action sequences that meet the above two constraints. These locomotion sequences have different requirements of the surrounding environment. Thus all these sequences are listed in the potential sequence set.

(3) Global action sequence optimization

Connecting the action sequence of adjacent POE along the global POE path, a whole sequence of the global path from the initial position to the target is obtained. However, different selections of the action sequence result in different consumption. There are many selection strategies, such as picking a random move sequence from the set. This strategy guarantees the reachability of the spacecraft, but few optimization indexes are considered. For on-orbit spacecraft, energy efficiency must be considered, and a simple index is adopted to evaluate the energy consumption of each basic locomotion roughly:

$$Q = \sum_{j=1}^M \sum_{i=0}^N \int_0^t \tau_{ji}^2(t) dt, \quad (12)$$

where N represents the actual number of manipulator joints (2 for the simplified model), M represents the number of arms (4 in this study), $\tau_{ji}(t)$ represents the torque of i th joint in j th arm, and t is the locomotion duration. To numerically evaluate the energy consumption, we assume that for a multi-arm spacecraft with size $0.31 \text{ m} \times 0.31 \text{ m} \times 0.31 \text{ m}$, mass $m=120 \text{ kg}$, inertia $[I_{xx} \ I_{yy} \ I_{zz}] = [12 \ 12 \ 12] \text{ kg m}^2$. Then we simplify the moving arm as a thin stick with length 1 m, mass $m=27.24 \text{ kg}$, inertia $[I_{xx} \ I_{yy} \ I_{zz}] = [0.0872 \ 0.5675 \ 0.5675] \text{ kg m}^2$. For each $\frac{\pi}{2}$ of rotation or 0.5 m translation, 10 s is required. Then the energy consumption with respect to the distance and angle by basic locomotion is illustrated as in Figure 11. Both

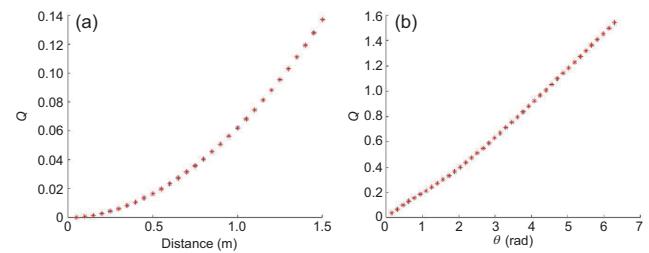


Figure 11 (Color online) Energy consumption with respect to (a) translational distance and (b) rotational angle of the basis motion.

the locomotion and translation can be approximated as a linear function of the distance and angle they travel, and the rotation consumes more energy than translation qualitatively. Comparing energy consumption of the sequence of Figure 10, sequence C in Figure 10(c) is optimal.

3.3 Low-level: arm motion planning

In the high-level and mid-level planning, we have used the simplified spacecraft model for path planning and obtained the global locomotion sequence and the end trajectories. Once the end trajectories are obtained, the realization of the trajectory should be converted to the actual model. This section describes the mapping method from the operational space of the arm to its joint space when the end position at the initial and final time instant of the action is given.

3.3.1 Inverse kinematics

Based on the solution of the base and POEs of the spacecraft, we focus on the inverse kinematic of updating the joint angle by solving the differential motion of the end and the Jacobian matrix of the manipulator. A simple illustration figure is listed as Figure 12.

As shown in Figure 12, firstly, we ignore the configuration change of the arms and move the base from the i th time instant, $\mathbf{p}_{\text{base},i}$ to the $(i+1)$ th position $\mathbf{p}_{\text{base},i+1}$, correspondingly the j th arm ends move from $\mathbf{p}_{j,i}$ to $\mathbf{p}_{j,i+1}^{i+1}$; secondly, we adjusting the spacecraft configuration, namely the arm ends from $\mathbf{p}_{j,i+1}^{i+1}$ to $\mathbf{p}_{j,i+1}$, accordingly. Specifically, $\mathbf{p}_{j,i+1}^{i+1}$ can be calculated as

$$\mathbf{p}_{j,i+1}^{i+1} = \text{Rot}(d\theta) \cdot (\mathbf{p}_{j,i} - \mathbf{p}_{\text{base},i}) + \mathbf{p}_{\text{base},i+1}, \quad (13)$$

where $\text{Rot}(d\theta)$ represents the elementary rotation matrix between the two adjacent sampling times. In Figure 12, $\text{Rot}([0, \frac{\theta}{N}, 0])$ indicates the base rotates $\frac{\theta}{N}$ about the Y axis.

Then the adjusting amount of the arm ends can be further calculated by

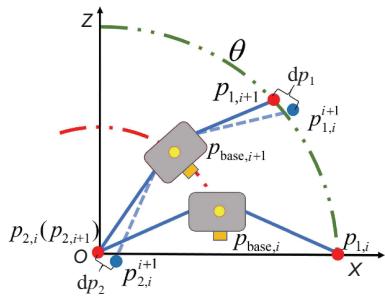


Figure 12 (Color online) Differentiation of the position of the end with respect to the base.

$$\begin{aligned} \text{global } d\mathbf{p}_j &= \mathbf{p}_{j,i+1} - \mathbf{p}_{j,i}^{i+1}, \\ \text{arm}_j \, d\mathbf{p}_j &= \underset{\text{robot}}{\overset{\text{arm}_j}{R}} \mathbf{R}_{\text{global}}^{\text{global}} d\mathbf{p}_j, \end{aligned} \quad (14)$$

with

$$\begin{aligned} \mathbf{J}_{j,i} &= \left(\frac{\partial \mathbf{p}_j}{\partial \mathbf{q}_j} \right)_{6 \times 7} \Big|_{\theta_j = \theta_{j,i}}, \\ d\mathbf{q}_j &= \mathbf{J}_{j,i}^{-1} * \underset{\text{arm}_j}{\overset{\text{robot}}{R}} \mathbf{R}_{\text{global}}^{\text{global}} d\mathbf{p}_j + (\mathbf{I} - \mathbf{J}_{j,i} \mathbf{J}_{j,i}^T (\mathbf{J}_{j,i} \mathbf{J}_{j,i}^T + \lambda^2 \mathbf{I})^{-1} \mathbf{J}_{j,i}) d\mathbf{q}_0, \end{aligned} \quad (15)$$

where $\text{global } d\mathbf{p}_j$ is the position discrepancy expressed in the global coordinate fixed on the truss, $\underset{\text{arm}_j}{\overset{\text{robot}}{R}} d\mathbf{p}_j$ is the position discrepancy expressed in the j th arm frame. $\underset{A}{R} \mathbf{B}$ describes the attitude transformation matrix from frame B to frame A . λ is the singularity damping parameter, which solves the practical singularity problem in the simulation process mainly based on singularity robust (SR) inverse [38] with

$$\lambda^2 = \begin{cases} \lambda_0^2 \left(1 - \frac{\sigma_m}{\sigma_0}\right), & \text{if } \sigma_m < \sigma_0, \\ 0, & \text{if } \sigma_m \geq \sigma_0, \end{cases} \quad (16)$$

where λ_0 is maximum damping parameter, σ_m is singularity value extracted from $\mathbf{J}_{j,i} \mathbf{J}_{j,i}^T$ and σ_0 is minimum singularity value boundary.

3.3.2 Obstacle avoidance planning

Substituting the obtained joint angle in previous subsection into the forward kinematics, the actual position of the arm ends can be obtained and it can be used for collision detection. If a collision occurs, some modification based on the gradient in the joint space is taken until the collision is avoided. The formula for modification is stated as

$$\begin{aligned} d\mathbf{q}_j &= \mathbf{J}_{j,i}^{-1} \cdot \underset{\text{arm}_j}{\overset{\text{robot}}{R}} d\mathbf{p}_j + \Delta \mathbf{p}_{OA}, \\ \mathbf{q} &= \mathbf{q}_0 + d\mathbf{q}_j. \end{aligned} \quad (17)$$

It should be noted that the collision avoidance method is a kind of “trial and error” method. The method uses an ideal model to detect collisions before control, iteratively modifies joint angles until no collision occurs, and then takes these joint angles as the desired position for control. In practical engineering [39] collisions may occur due to position estimation errors, environment modeling errors, or dynamic obstacles. Future real-time collision avoidance methods should result in low computational [40–42] via numerical-analytical approach [43] or equivalence of efficient inverse kinematics calculation [44, 45]. Besides, the surface property may also influence the planning and collision avoidance method. They may need to be considered using surface detection technology, such as the learning based approach [46].

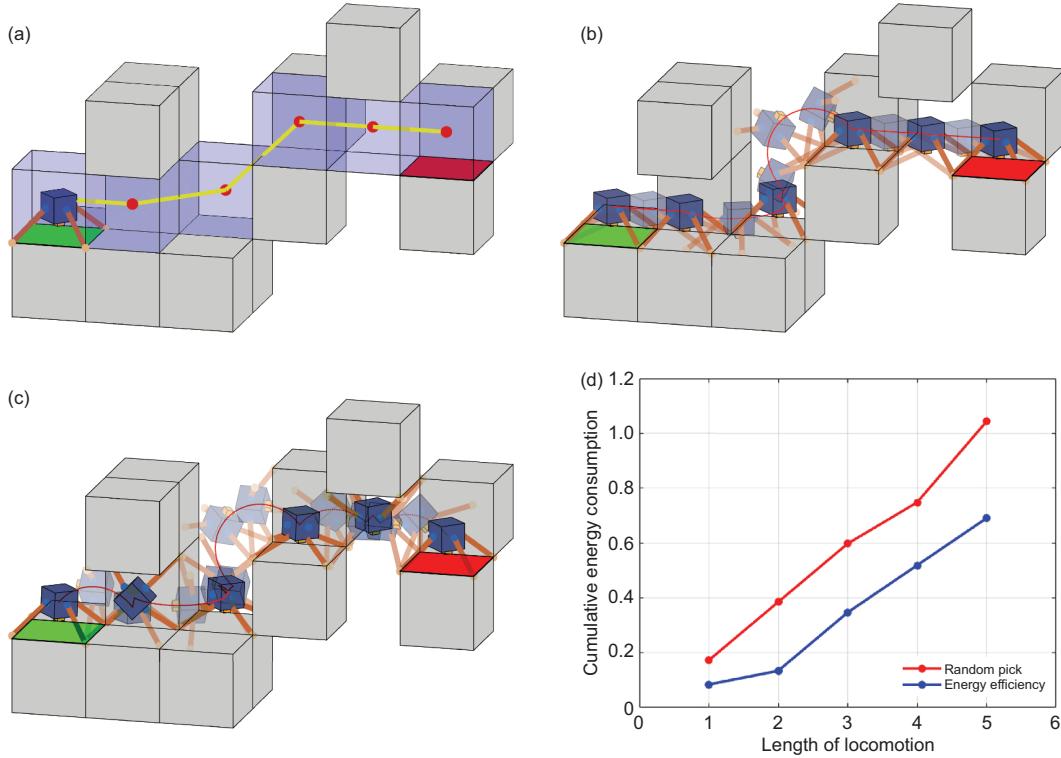


Figure 13 (Color online) Simulation results in the simple map. (a) High-level planning of the small map; (b) locomotion sequence chosen according to energy efficiency strategy; (c) locomotion sequence chosen according to randomly pick strategy; (d) comparison of energy consumption of the two generated sequences.

Table 1 Locomotion sequence obtained by different strategy

Step S/N	Global node	Energy efficiency strategy		Randomly pick strategy	
		Translation	Rotation	Translation	Rotation
1	(1, 0, 0)	(+X, L)	Null	Null	((D, \overrightarrow{DA}), π , 1), ((A, \overrightarrow{AB}), $\pi/2$, 1)
2	(2, 1, 0)	Null	((A, \overrightarrow{AE}), π , 0)	Null	((D, \overrightarrow{DA}), $3\pi/2$, 1), ((D, \overrightarrow{DA}), $\pi/2$, 1)
3	(3, 1, 1)	Null	((D, \overrightarrow{DA}), $\pi/2$, 1)	Null	((D, \overrightarrow{DA}), $\pi/2$, 1)
4	(4, 1, 1)	(+X, L)	Null	Null	((D, \overrightarrow{DA}), π , 1), ((A, \overrightarrow{AB}), $\pi/2$, 1)
5	(5, 1, 1)	(+X, L)	Null	Null	((C, \overrightarrow{CD}), $\pi/2$, 1), ((D, \overrightarrow{DA}), π , 1)

4 Simulation experiment

In this section, we constructed two maps of different lengths, as shown in Figures 13(a) and 14(a), to show the effectiveness of the proposed path planning method. The former is a small map with a path length of 5, and the latter is a large and complex map with a path length of 50. For comparison, different move sequences are selected by different strategies, and the energy consumption index is compared.

In the small map, the global path, which is the blue grid sequence in Figure 13(a), is firstly obtained by high-level planning. Then the middle-level planning is carried out to generate the global locomotion sequence from the locomotion sequence set according to the energy optimal strategy and the random pick strategy, respectively. Finally, the lo-

comotion sequence generated by the middle-level is passed to the lower-level planner for joint angle solving, and the results are shown in Figure 13(b) and (c). These locomotion sequences are listed in Table 1. It can be seen from the Table 1 and Figure 13(c) that the move sequence generated by the randomly pick strategy has more unnecessary rotations compared with the energy optimal one, and it consumes more energy. Further considering the fact shown in Figure 11, the sequence generated by randomly pick consumes more energy compared with the one selected according to the energy efficiency index. This is further illustrated in Figure 13(d).

The complex experimental map is shown in Figure 14(a). The global path generated by the high-level planner is demonstrated in Figure 14(b). Figure 14(c) shows the move sequence of the spacecraft chosen by the energy optimal strat-

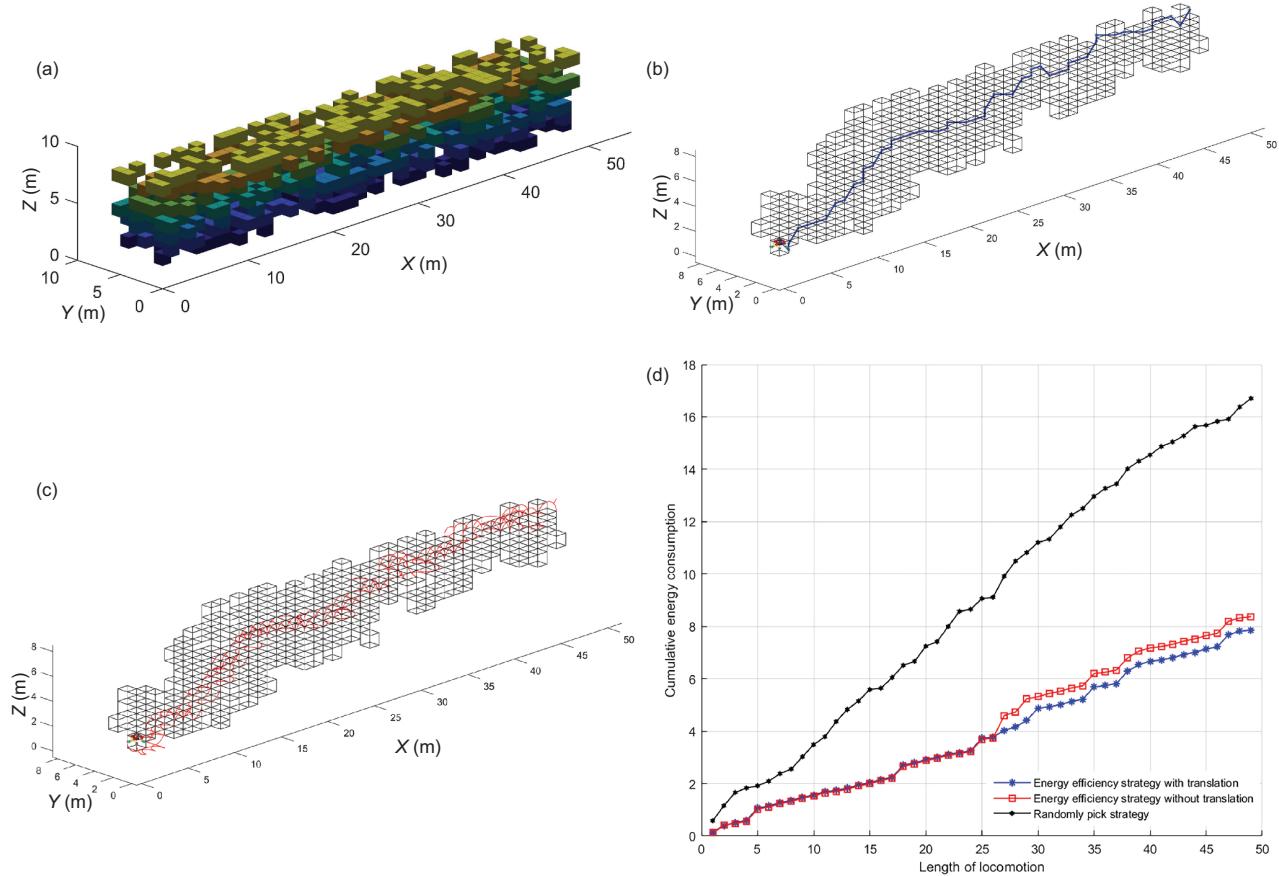


Figure 14 (Color online) Simulation results in the complex map. (a) Simulation map; (b) global path by high-level planning; (c) energy efficiency locomotion trajectory; (d) comparison of energy consumption of the three different strategies.

egy. To compare the energy efficiency, three different results are listed. As shown in Figure 14(d), the optimized one chosen according to the energy index consumes half less than that chosen by the stochastic strategy, which shows the superiority of the proposed energy optimization strategy.

5 Conclusion

The point-to-point path planning problem of spacecraft equipped with multi-arms on the structured truss is investigated in this paper. Two general kinds of locomotion modes, i.e., translational locomotion and rotational locomotion, for coplanar movement and moving between different planes are designed and introduced. Based on these two basic locomotion, a general hierarchical path planning method is proposed. In the hierarchical architecture, the high-level planner generates the global position information using the ACOBNCS method, the middle-level planner generates the energy efficiency move sequence, and the low-level planner gives the joint trajectory during the whole process. An all-process simulating experiment is conducted and visualized, and the effectiveness of the proposed method has been verified. For future work, unstructured environments and real-time colli-

sion monitoring methods will be considered. To further verify the effectiveness of hierarchical planning through physical experiments, we have assembled a multi-arm robot and built an air-bearing platform [47, 48]. Furthermore, collision detection based on proprioceptive sensors and emergency maneuver strategies after a collision will be embedded in future low-level planning.

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62003115 and 11972130), and the Shenzhen Natural Science Fund (the Stable Support Plan Program GXWD20201230155427003-20200821170719001).

- Li W J, Cheng D Y, Liu X G, et al. On-orbit service (OOS) of spacecraft: A review of engineering developments. *Prog Aerospace Sci*, 2019, 108: 32–120
- Ding X L, Wang Y C, Wang Y B, et al. A review of structures, verification, and calibration technologies of space robotic systems for on-orbit servicing. *Sci China Tech Sci*, 2021, 64: 462–480
- Boning P, Dubowsky S. Coordinated control of space robot teams for the on-orbit construction of large flexible space structures. *Adv Robotics*, 2010, 24: 303–323
- Hoyt R P, Cushing J, Jimmerson G, et al. Spiderfab: Process for on orbit construction of kilometer-scale apertures. Technical report. NASA Headquarters, Washington D C, 2018. HQ-E-DAA-TN62833, 2018
- Deremetz M, Gerhard G, Cavenago F, et al. Concept of operations and preliminary design of a modular multi-arm robot using standard in-

- terconnects for on-orbit large assembly. In: Proceedings of the International Astronautical Congress (IAC). Dubai, 2021
- 6 Henshaw C G. The darpa phoenix spacecraft servicing program: Overview and plans for risk reduction. In: Proceedings of International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS). Montreal, 2014
 - 7 Hoyt R P. Spiderfab: An architecture for self-fabricating space systems. In: Proceedings of the AIAA SPACE 2013 Conference and Exposition. IAA, San Diego, 2013
 - 8 Yue C F, Wei C, Cao X B, et al. A kind of implemental space-craft system facing on-orbit manipulation missions. China Patent, CN114162353A, 2021
 - 9 Zhang Y, Zhang J X, Ren L Q. The terrestrial locomotion of a mole cricket with foreleg amputation. *Sci China Tech Sci*, 2015, 58: 999–1006
 - 10 Xie Z, Chen X, Chen D, et al. Initial design and implementation of a space quadruped crawling robot prototype. *Adv Astronaut Sci Technol*, 2021, 4: 173–181
 - 11 Miao C W, Chen G Z, Yan C L, et al. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput Indust Eng*, 2021, 156: 107–230
 - 12 Xiang D, Lin H, Ouyang J, et al. Combined improved A and greedy algorithm for path planning of multi-objective mobile robot. *Sci Rep*, 2022, 12: 13273
 - 13 Orok J A, Nyakoe G N. Obstacle avoidance and path planning schemes for autonomous navigation of a mobile robot: A review. In: Proceedings of the 2012 Sustainable Research & Innovation (SRI) Conference. Pretoria, 2022. 314–318
 - 14 Mo Y, Jiang Z H, Li H, et al. A kind of biomimetic control method to anthropomorphize a redundant manipulator for complex tasks. *Sci China Tech Sci*, 2020, 63: 14–24
 - 15 Tian Y, Gao F, Liu J M, et al. Step rolling planning of a six-legged robot with 1-DOF waist for slope climbing. *Sci China Tech Sci*, 2019, 62: 597–607
 - 16 Mao L H, Tian Y, Gao F, et al. Novel method of gait switching in six-legged robot walking on continuous-nondifferentiable terrain by utilizing stability and interference criteria. *Sci China Tech Sci*, 2020, 63: 2527–2540
 - 17 Spröwitz A, Tuleu A, Vespignani M, et al. Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot. *Int J Robotics Res*, 2013, 32: 932–950
 - 18 Kang R J, Guglielmino E, Branson D T, et al. Bio-inspired crawling-glocomotion of a multi-arm octopus-like continuum system. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. Vilamoura-Algarve, 2012
 - 19 Li Y, Zi B, Yang Z M, et al. Combined kinematic and static analysis of an articulated lower limb traction device for a rehabilitation robotic system. *Sci China Tech Sci*, 2021, 64: 1189–1202
 - 20 Ding X L, Wang Z Y, Rovetta A, et al. Locomotion analysis of hexapod robot. *Climb Walk Rob*, 2010, 30: 291–310
 - 21 Zhang Y, Li P, Quan J, et al. Progress, challenges, and prospects of soft robotics for space applications. *Adv Intell Syst*, 2022, 2200071
 - 22 Martínez-Moritz J, Rodríguez I, Nottensteiner K, et al. Hybrid planning system for in-space robotic assembly of telescopes using segment-edmirrors tiles. In: Proceedings of the 2021 IEEE Aerospace Conference (50100). Big Sky, 2021. 1–16
 - 23 Doyle R, Kubota T, Picard M, et al. Recent research and development activities on space robotics and AI. *Adv Robotics*, 2021, 35: 1244–1264
 - 24 Scherzinger S, Weinland J, Wilbrandt R, et al. A walking spacerobot for on-orbit satellite servicing: The recobot. arXiv: 2203.10217
 - 25 Adinehvand M, Asadi E, Lai C Y, et al. Bogiebot: A climbing robot in cluttered space of bogies with ferrous metal surfaces. In: Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Prague, 2021. 2459–2466
 - 26 Sheng X J, Tang L, Huang X J, et al. Operational-space wrench and acceleration capability analysis for multi-link cable-driven robots. *Sci China Tech Sci*, 2020, 63: 2063–2072
 - 27 You B D, Wen X L, Liu Y Q, et al. Gait analysis of cellular space robot for on-orbit climbing truss. *J Astronaut*, 2020, 41: 521–530
 - 28 Feng J K, Liu J G. Configuration analysis of a chain-type reconfigurable modular robot inspired by normal alkane. *Sci China Tech Sci*, 2021, 64: 1167–1176
 - 29 Jiang L. Gait analysis of a novel biomimetic climbing robot. *J Mech Eng*, 2010, 46: 17–22
 - 30 Ramón J L, Calvo R, Trujillo A, et al. Trajectory optimization and control of a free-floating two-arm humanoid robot. *J Guidance Control Dyn*, 2022, 45: 1661–1675
 - 31 Zhao J D, Han J G, Ni F L. Gait planning and study of climbing robot with three branches for space station truss. *Man Space*, 2018, 24: 14–19
 - 32 McBryan K. Comparison between stationary and crawling multi-arm robotics for in-space assembly. In: Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, 2020. 1849–1856
 - 33 Tang T, Hou X, Xiao Y, et al. Research on motion characteristics of space truss-crawling robot. *Int J Adv Robotic Syst*, 2019, 16: 172988141882157
 - 34 Sun T S, Wang Y H, Yang S Q, et al. Design, hydrodynamic analysis, and testing of a bioinspired controllable wing mechanism with multi-locomotion modes for hybrid-driven underwater gliders. *Sci China Tech Sci*, 2021, 64: 2688–2708
 - 35 Zhang W. Research on autonomous navigation method for indoor robots based on multisensor fusion. Dissertation for Doctoral Degree. Hefei: University of Science and Technology of China, 2017
 - 36 Yue C F, Zhang X, Wang H X, et al. Three-dimensional path planning of on-orbit manipulation robot based on neighborhood continuation search. *J Astronaut*, 2022, 43: 206–213
 - 37 Chen Y X, Dong W, Ding Y. An efficient method for collision-free and jerk-constrained trajectory generation with sparse desired way-points for a flying robot. *Sci China Tech Sci*, 2021, 64: 1719–1731
 - 38 Leve F A. Evaluation of steering algorithm optimality for singlegimbal control moment gyroscopes. *IEEE Trans on Contr Sys Tech*, 2013, 22: 1130–1134
 - 39 Li C Y, Li Z Q, Jiang Z N, et al. Autonomous planning and control strategy for space manipulators with dynamics uncertainty based on learning from demonstrations. *Sci China Tech Sci*, 2021, 64: 2662–2675
 - 40 Kenwright B. Inverse kinematics—Cyclic Coordinate Descent (CCD). *J Graphics Tools*, 2012, 16: 177–217
 - 41 Muller-Cajar R, Mukundan R. Triangulation—A new algorithm for inverse kinematics. In: Proceedings of Image and Vision Computing New Zealand 2007. New Zealand, 2007
 - 42 Sun H W, Yang J X, Li D W, et al. An on-line tool path smoothing algorithm for 6R robot manipulator with geometric and dynamic constraints. *Sci China Tech Sci*, 2021, 64: 1907–1919
 - 43 Anantharayanan H, Ordóñez R. Real-time inverse kinematics of $(2n+1)$ DOF hyper-redundant manipulator arm via a combined numerical and analytical approach. *Mech Mach Theor*, 2015, 91: 209–226
 - 44 Hu Z H, Yuan H, Xu W F, et al. A geometric method based on space arc for pose-configuration simultaneous planning of segmented hyper-redundant manipulators. *Sci China Tech Sci*, 2021, 64: 2389–2407
 - 45 Maric F, Giambou M, Hall A W, et al. Riemannian optimization for distance-geometric inverse kinematics. *IEEE Trans Robot*, 2022, 38: 1703–1722
 - 46 Shi Y, Li L, Yang J, et al. Center-based transfer feature learning with classifier adaptation for surface defect recognition. *Mech Syst Signal Pr*, 2023, 188: 110001
 - 47 Yue C F, Wei C, Cao X, et al. A virtual scene simulation system for space on-orbit manipulation. China Patent, CN114218702A, 2021
 - 48 Yue C F, Wei C, Cao X, et al. A ground test platform, system, and method for multi-arm spacecraft system. China Patent, CN114261543A, 2021