Robotica (2025), **43**, pp. 2193–2211
doi:10.1017/S0263574725101756

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Configuration optimization of variable topological space robot for impulse minimization based on bilevel approach

Hongxu Wang[1,2,3], Jinsheng Guo[1,2], Carlo Canali[3], Chengfei Yue[2,4], Xibin Cao[1,2] and Darwin G. Caldwell[3]

[1]Research Center of Satellite Technology, Harbin Institute of Technology, Harbin, China
[2]State Key Laboratory of Micro-Spacecraft Rapid Design and Intelligent Cluster, Harbin Institute of Technology, Harbin, China
[3]Advanced Robotics, Italian Institute of Technology, Genova, Italy
[4]School of Aerospace Science, Harbin Institute of Technology Shenzhen, Shenzhen, China
Corresponding author: Chengfei Yue; Email: yuechengfei@hit.edu.cn

Received: 18 December 2024; **Revised:** 4 April 2025; **Accepted:** 4 May 2025; **First published online:** 30 May 2025

Keywords: impulse minimization; bilevel optimization; space robots with variable topology; topology transformation; heuristic algorithm

Abstract

Variable topological space robots are essential for providing adaptability and flexibility, enabling the robot to adjust its morphology to perform a range of tasks in the unstructured environment of space. However, impact is a common consequence of topology transformation in space robotics, which may lead to irreversible damage, such as the shedding of solid lubrication on joints. Nevertheless, determining the precise force-time relationships of such impacts poses significant challenges, especially when accounting for various connection mechanisms. In this work, a docking strategy that optimizes the manipulator's joint angle configuration to minimize the impulse when the topology changes is proposed. First, an estimation technique is developed to quantify the impulse generated by topology transformation, employing spatial operator algebra and generalized momentum balance equations. Based on this model, the impulse minimization is modelled as a bilevel optimization problem, which decomposes a complex multipolar problem into two simpler subproblems. Although this optimization model may compromise computational efficiency, it increases the probability of achieving an optimal solution. To address this, a bilevel solution strategy based on a heuristic algorithm is proposed. In this framework, the lower level uses particle swarm optimization to determine the global optimum, while the upper level adopts simulated annealing to enhance computational speed. Finally, simulations are conducted to validate the proposed approach. Results demonstrate that the proposed method substantially reduces impulse.

1. Introduction

With developments in space technology, the demand for extensive space facilities has significantly increased [1–6]. Space robots characterized by flexible and extensible topologies are poised to play essential roles in future applications [7], due to their adaptabilities to various morphologies to meet diverse operational requirements. However, the impact is a common consequence [8] during docking between modules in topological transformations, resulting in abrupt velocity changes. Potentially, these may cause significant problems, including base attitude destabilization, joint failure, if without carefully designed topology transformation method.

The variable topology approach primarily refers to the specific joint angle configuration in which the topology transformation occurs. The optimal joint angle configuration aimed at minimizing the base disturbance has been extensively studied [9, 10]. Nevertheless, there are fewer studies on minimizing joint impact, despite their considerable long-term influence on the operational performance of space robots. For example, because of the extremely high vacuum environments where the space robots operate, the contact surfaces between moving parts inside the space robots will be bonded together due to

© The Author(s), 2025. Published by Cambridge University Press.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

the cold welding phenomena, without sufficient lubrication. But the repeated cumulative impacts within the robotic system will cause irreversible damage to the solid lubricant layer, until delamination. This delamination will lead to the robot failure and operational termination, thereby shortening the operational lifespan of the robotic system. However, if designed with careful consideration, these negative effects may be mitigated or even entirely eliminated. To address this problem, two key challenges still remain: first, a universal model to describe their dynamic behaviour during transformations considering the diverse designs of connection mechanisms must be developed, and second, a specific topology transformation strategy, such as the optimal contact position and orientation, the optimal joint angles configurations, needs to be designed. These challenges place heightened demands on dynamic modelling and pre-dock joint angle optimization throughout the topology transformation process.

The issues above can be categorized into three primary aspects: dynamic modelling regardless of detailed connection mechanism designs, the definition of optimization indicators for topology transformation, and the modelling and solving of configuration optimization problems. For dynamic modelling of connections, this is fundamentally a constrained multibody dynamics problem. The most common approach is to establish a comprehensive dynamic model of the entire contact process. For instance, Yuki [11] developed a docking dynamic model for target and chaser satellites using a probe-cone docking mechanism. Similarly, Kamran [12] established a dynamic model for spacecraft equipped with a spring-damper buffering device during impact phenomena, while Zhao [13] derived the contact and impact dynamics of a docking system employing a two-androgynous peripheral docking mechanism. These models require detailed design of the specific connection mechanisms, yet the variety of such mechanisms makes uniform dynamic modelling nearly impossible. In addition, for space robots with variable topologies, the contact process is just a concise duration, especially compared with its working time. Only the pre-contact, post-contact and the difference between them, is the essential process that could influence the system's performance. Thus, for simplification, the contact process is considered to be instantaneous, leading to a pure impulse at the contact zone. Haug [14] was among the first to define topology changes as the addition or removal of constraints and to derive the resultant changes in motion states. Based on this, Khulief [15] regarded these changes as impulsive events and proposed a numerical modelling method. Chang [16] employed a piecewise model and Kane's method to investigate the impulsive motion of multibody systems. In contrast, Raina [17] offered a unified framework for modelling impact dynamics using the impulse-momentum formulation. However, this approach may have some problems, as the duration of the impact depends on various factors, such as the specific mechanism design and impact velocities. When these factors vary, the duration changes at the same time. Furthermore, the impact is calculated by dividing the impulse by the duration of impulse. It is also a little unreasonable because the change of impact during the process may not be linear.

Regarding the optimization of topology transformations, there are many ways to define the objective function, such as consumed energy minimization [18], grasping force optimization [19], manipulability measure [20], etc. Xu [21] defined the global manipulability of cooperative manipulation for dual-arm space robots and optimized the corresponding grasping poses. However, this research primarily addressed post-capture manipulation and did not consider the effects of pre-capture joint angle configurations. Toglia [22] devised optimization strategies to minimize the amplitude of any vibrations at the end effector of a flexible space manipulator during grasping. Lee [23] proposed a dual-arm manipulability measure for handling multiple objects and identified a series of optimal joint angle configurations. Notably, most of these objective functions pertain to object grasping and overlook another critical topology transformation scenario: the connection of one link to another.

In terms of modelling and solving joint angle configuration optimization problems, the redundancy inherent in manipulators typically results in multipolar point optimization challenges that can easily become stuck in local optimization. The common way is to model it directly and utilize a heuristic algorithm to solve the problem. Osgouie [24] utilized a genetic algorithm to identify optimal joint angle configurations for a dual-arm cam-lock robot, while Xue [25] employed particle swarm optimization (PSO) for optimizing joint angle configurations of a closed-chain dual-arm reconfigurable space robot.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

Since the objective function for optimization is often not unique, developing multi-objective optimization algorithms is essential. For instance, Yan [9] used multi-objective particle swarm optimization to maximize manipulability and minimize base disturbance, while Zhao [26] focused on maximizing manipulability while minimizing joint torque. However, as the complexity of robots increases, the number of polar points also increases significantly. This makes finding optimal solutions increasingly challenging using these algorithms. Bilevel optimization is a specialized form of optimization wherein one problem is nested within another [27], effectively decomposing a multi-point optimization problem into two simpler subproblems, making it possible to find the optimal solution.

In summary, extensive research has been conducted on dynamic analysis and joint angle configuration optimization. However, most dynamic models focus on specific mechanisms, such as probe-cone docking, which limits their applicability across various connection forms. Additionally, existing research on joint angle configuration optimization predominantly emphasizes object manipulation or grasping with single or dual-arm robots, largely neglecting the unique challenges of topology transformation involving link connections. The optimization algorithms employed are commonly heuristic, such as genetic algorithms and particle swarm optimization, which may be insufficient for addressing the complexities of higher-degree-of-freedom multi-arm robots.

To optimize the pre-docking joint angle configuration of a space robot connecting two arms into a longer one, this work proposes a strategy that minimizes impulse. The main work and contribution of this paper are listed as follows:

1) A universal dynamic model for topology transformation in space robot is proposed, utilizing spatial operator algebra and impulse-momentum equations. This modelling approach is independent of specific connection mechanisms, making it applicable to a broad range of transformation scenarios. Given the challenges of calculating impact forces, impulse is used as a substitute and serves as the evaluation parameter in the optimization process.
2) The angular configuration optimization model is formulated as a bilevel optimization to address the challenges associated with multi-polar points. In the lower-level optimization task, the contact pose of two arms' ends is provided, aiming to determine the optimal angular configuration and minimize impulse for the specific pose. In the upper-level optimization task, the minimized impulse from the lower-level task serves as the objective function, enabling the determination of the optimal contact pose for the arm ends. Additionally, closed-loop constraints and self-avoidance constraints are incorporated into this optimization model.
3) A heuristic-based algorithm is proposed to address the optimization problem. The PSO [28, 29] is applied in the lower-level task to identify the global optimal solution, while the simulated annealing (SA) [30] algorithm is utilized in upper level to improve computational efficiency. The optimization results demonstrate that the proposed method effectively determines the optimal angular configuration for space robots with variable topologies.

This paper is organized as follows: Section 2 derives the dynamic model and introduces optimization metrics aiming at minimizing impulse. Section 3 presents the bilevel optimization model for space robots performing link connections. In Section 4, a solution methodology for the bilevel optimization model is developed based on heuristic algorithms, along with a presentation of the optimal results and a comparative analysis. Finally, Section 5 provides the conclusion.

2. Dynamic modelling and impulse estimation during transformation

2.1. Variable topological space robot description

The space robot and its topology transformation process studied in this paper are illustrated in Figure 1. It's designed as a universal on-orbit operational tool spacecraft, aiming to be applied in numerous scenarios, including the construction of new large-scale space facilities, maintenance of the International Space Station, and the refurbishment and repair of existing defunct satellites, etc. The robot features a

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

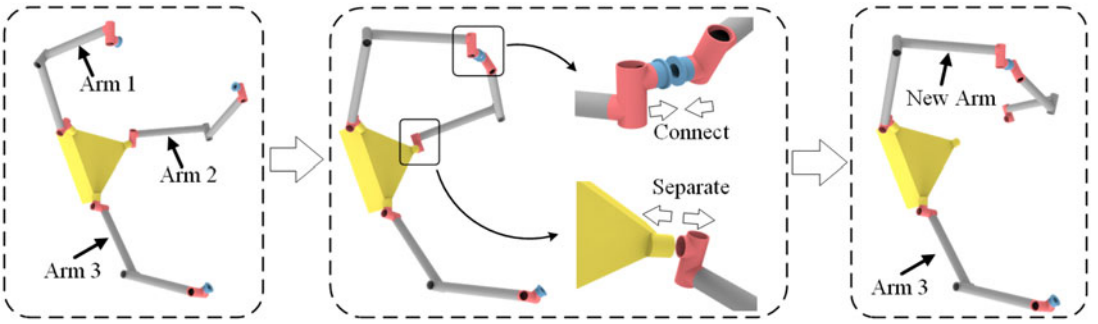


Figure 1. *Diagram of space robot and topology transformation.*

base that can translate and rotate in six degrees of freedom (DOFs), operating either under control or freely. It has three arms, each comprising seven links and revolute joints, resulting in 27 DOFs. Here, the triple-arm system is designed to simultaneously accommodate both grasping and manipulation tasks. Besides, it can perform dual-arm cooperative tasks even if one arm is failed. The 7 DOFs in every arm are to ensure that each arm can independently perform tasks within a 6-degree-of-freedom operation space. And the additional 1 degree of freedom ensures the arm can achieve additional functionalities such as obstacle avoidance, singularity avoidance and fault-tolerance control. All components of the robot are treated as rigid bodies. Furthermore, the end of each arm is equipped with a connection mechanism to facilitate topology transformation, enabling it to connect with various links [31].

To adapt to the variable construction tasks of future space facilities, space robots may need to be able to alter their topology configurations to meet specific task requirements autonomously. One potential scenario is illustrated in Figure 1. The left figure is the initial topology configuration, which is also the configuration unfolded after launch. The workspace or the degree of freedom for the single arm is not large enough for some missions, and there is a need for a more extended arm. Under this circumstance, arm 1 can be separated from the base and connected to arm 2, forming a more extended arm with increased degrees of freedom. To achieve this action, the uncoupled end links of the two arms approach each other and connect when sufficiently close. Subsequently, the root link of one arm detaches from its base, resulting in a "new" single robot with a long, hyper-redundant arm. These coupling/uncoupling operations rely on an advanced connection mechanism that provides mechanical and electrical interfaces, allowing for detachment and attachment on demand. Although extensive research has been conducted on such mechanisms, there is no standard approach or design and each exhibits unique characteristics, making it challenging to describe their dynamic behaviours uniformly. This lack of a standard design or solution underscores the need for a universal dynamic model.

2.2. Dynamic model for link connection

The dynamic model for link connection mainly includes two parts: the space robot dynamic model and the connection process model.

The coordinate system definitions for the robot discussed in this paper are shown in Figure 2. For the definitions of single arm, the links in the arm are numbered in increasing order from tip to base. The X_j^i denotes the X axis for the i-th link of the arm j. The origin of coordinate system for i-th link is located on the internal hinge. The X axis points to the motion axis, the Y axis points to the external hinge, and Z axis direction is determined according to the right-hand rule. For the coordinate system of the base, the origin is located on its centre of mass, with the axes coinciding with the inertial coordinate system at the beginning.

The dynamic model of the space robot can be described as follows based on spatial operator algebra [32]. The velocity, acceleration and force of every joint can be described as:

$$V(k) = \begin{pmatrix} \omega_k \\ v_k \end{pmatrix} \tag{1}$$

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

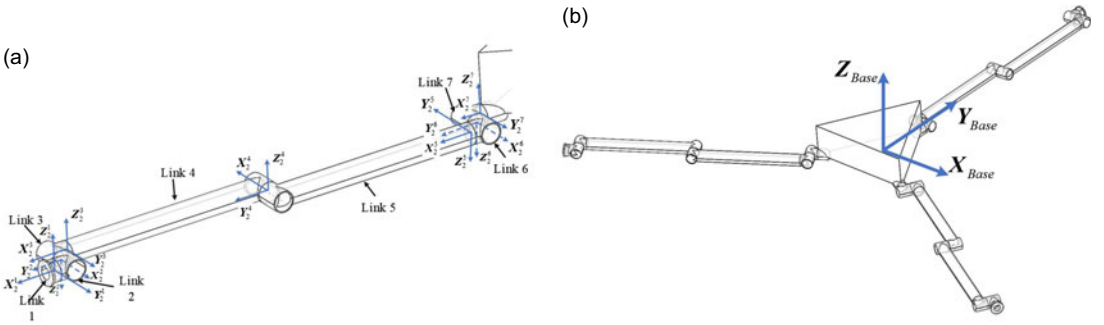


Figure 2. *Diagram of coordinate systems definition.*

$$\alpha\,(k) = \begin{pmatrix} \dot{\omega}_k \\ \dot{v}_k \end{pmatrix} \tag{2}$$

$$f\,(k) = \begin{pmatrix} N_k \\ F_k \end{pmatrix} \tag{3}$$

where the subscript k represents the k-th link; ω_k and v_k denote the angular and linear velocity vectors of joint k; N_k and F_k mean the joint moments and force vectors, respectively.

The velocity and acceleration recursive relationship between adjacent links is:

$$\begin{cases} V(k) = \phi^T(k+1,k)V(k+1) + H^T(k)\dot{\theta}(k), \\ \alpha(k) = \phi^T(k+1,k)\alpha(k+1) + H^T(k)\ddot{\theta}(k) + a(k) \end{cases} \tag{4}$$

where $a(k)$ is the Coriolis acceleration term; $\phi(k+1,k)$ denotes the interbody transformation operator; $H(k)$ denotes the joint map matrix; $\theta(k)$ is the vector of generalized variables for the k-th hinge. Specially, the base is connected with inertial space by a virtual hinge, so the $\theta(k)$ for the base denotes both the translation and rotation displacement of the base.

The force and torque recursive relationship between adjacent links is:

$$f(k) = \phi(k,k-1)f(k-1) + M(k)\alpha(k) + K(k)\theta(k) + D(k)\dot{\theta}(k) + b(k) \tag{5}$$

where $M(k)$ denotes the inertia matrix; $K(k)$ denotes the stiffness matrix; $D(k)$ denotes the damp matrix; $b(k)$ is the centrifugal force term for the k-th body, whose detailed expression can be found in ref. [33].

The operator expressions for the generalized forces $f = col\,\{f(k)\}$ and the spatial velocities $V = col\,\{V(k)\}$ of all the bodies in the system are given by:

$$V = \phi^T H^T \dot{\theta} \tag{6}$$

$$\alpha = \phi^T H^T \ddot{\theta} + \phi^T a \tag{7}$$

$$f = \phi(M\alpha + K\theta + D\dot{\theta} + b) \tag{8}$$

$$T_a = Hf \tag{9}$$

Write the these items as the operator expressions, and the dynamic model for the whole system is as follows:

$$\begin{aligned} T_a = Hf &= H\phi M\phi^T H^T\ddot{\theta} + H\phi K\theta + H\phi D\dot{\theta} + H\phi(M\phi^T a + b) \\ &= \mathcal{M}\ddot{\theta} + \mathcal{K}\theta + \mathcal{D}\dot{\theta} + \mathcal{C} \end{aligned} \tag{10}$$

Here, $\mathcal{M}$ is the system mass matrix, $\mathcal{K}$ is the system stiffness matrix, $\mathcal{D}$ is the system damp matrix, and $\mathcal{C}$ is the vector of Coriolis and centrifugal forces for the system.

The docking process can be divided into three phases: pre-, during- and post-connection, just as shown in Figure 3. The docking discussed here is only the connection between different modules

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

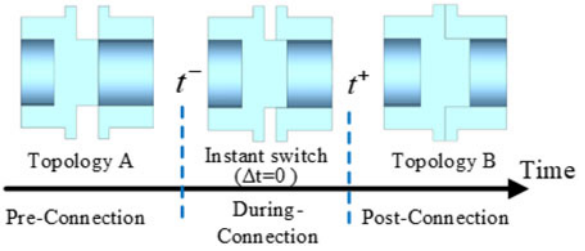


Figure 3. *Diagram of docking process.*

inside the space robot system. Here, we take a simple probe-cone mechanism as an example. The pre-connection phase indicates the stage just before the probe makes contact with the outer edge of the cone. This remains in the configuration of Topology A. The during-connection shows the stage of the probe's movement through the cone, where the topology transformation happens. The post-connection indicates the stage after the probe is in full contact with the cone. This becomes a new topology, for example, Topology B. The dynamic behaviours during the pre-connection phase can be described as in (10). The motion occurring during the connection phase is inherently complex, involving movements within the connection mechanism as it transitions from the connection initiation to a stable state. This phase represents a constrained dynamical process highly dependent on the specific design features of the connection mechanism. For analytical simplicity, the during-connection phase is considered to occur instantaneously. Then the dynamic process for this phase becomes a transient impact, which is also the one that needs to be minimized. The impact from reconfiguration just occurs at this moment. Before or after this instance, there is no more impact from this transformation.

Before proceeding with the derivation of the dynamic equations, several assumptions need to be introduced:

1) The reconfiguration is assumed to take place at an instant of time, in other words, the short interval Δt can be regarded as 0.
2) The generalized coordinates of the system remain unchanged during the reconfiguration. In other words, the joint angles, base position and base orientation do not change. But their velocities may have a sudden jump.
3) The generalized coordinates and the externally applied forces are continuous, and the velocities are bounded during the sudden event.

Given this assumption, the position coordinates remain unchanged, while the system's velocity changes suddenly, as follows:

$$\begin{gathered} \Delta t \to 0 \\ \theta^- = \bar{\theta} = \theta^+ = \text{const} \\ \dot{\theta}^- \neq \dot{\theta}^+ \end{gathered} \tag{11}$$

Here, $-$ and $+$ represent the start and end moments of an event, respectively.

Under these assumptions, integrate the robot dynamics over the short interval $\Delta t = [t^-, t^+]$, take limits as Δt tends to 0, and the following can be obtained:

$$\lim_{\Delta t \to 0} \int_{t^-}^{t^+} \left\{ \mathcal{M}\ddot{\theta} + \mathcal{K}\theta + \mathcal{D}\dot{\theta} + \mathcal{C} \right\} dt = \lim_{\Delta t \to 0} \int_{t^-}^{t^+} \left\{ T_a - C_c^T \lambda \right\} dt \tag{12}$$

where C_c represents the Jacobian of the constraint equations, and λ denotes the Lagrange multipliers. Incorporating the constraint term allows for a more comprehensive description of the system and is essential for deriving the jump in velocities.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

Here $\mathcal{M}$, $\mathcal{K}$, $\mathcal{D}$ and C_c are functions of only generalized coordinates, which means $\mathcal{M}(\theta)$, $\mathcal{K}(\theta)$, $\mathcal{D}(\theta)$ and $C_c(\theta)$. Considering that θ is constant during the integration, then (12) can be expressed as:

$$\lim_{\Delta t\to 0}\left(\mathcal{M}(\theta)\Delta\dot{\theta}+\mathcal{K}(\theta)\theta\Delta t+\mathcal{D}(\theta)\Delta\theta+\int_{t^-}^{t^+}\mathcal{C}(\theta,\dot{\theta})dt\right)=\lim_{\Delta t\to 0}(\bar{T}_a\Delta t-C_c(\theta)^T\mu) \tag{13}$$

In (13), $\mathcal{C}(\theta,\dot{\theta})$ is a continuous function on the small-scale time interval $[t^-,t^+]$. Using the mean value theorem for integrals, there is one point t_ξ such that:

$$\int_{t^-}^{t^+}\mathcal{C}(\theta,\dot{\theta})\,\mathrm{d}t=\mathcal{C}(\bar{\theta},\dot{\theta}_\xi)(t^+-t^-)=\mathcal{C}(\bar{\theta},\dot{\theta}_\xi)\Delta t \tag{14}$$

Then considering the generalized coordinates θ remain constant at this interval $\Delta\theta\to 0$ and $\Delta t\to 0$. Hence, (13) can be simplified to:

$$\mathcal{M}\Delta\dot{\theta}=-C_c^T\mu \tag{15}$$

where $\delta\dot{\theta}$ and μ are all unknown variables.

Then, we define C_c^- and C_c^+ as the constraint Jacobian matrices before and after the impact switch, respectively. Generally, the constraint equations at the position level must be satisfied at all times during the interval. Thus, the following equations are obtained:

$$\varphi^-(\theta^-)=0,\ \varphi^+(\theta^-)=0,\ \varphi^+(\theta^+)=0 \tag{16}$$

Then, the velocity level constraint equations can be derived by differentiating (16):

$$\dot{\varphi}^+(\theta^+)=C_c^+\dot{\theta}^+=0 \tag{17}$$

Considering $\Delta\dot{\theta}=\dot{\theta}^+-\dot{\theta}^-$, one obtains:

$$C_c^+\dot{\theta}^-+C_c^+\Delta\dot{\theta}=0 \tag{18}$$

Associating (15) with (18), the following is obtained:

$$\begin{bmatrix}\mathcal{M}(q) & \left[C_c^+\right]^T\\ C_c^+ & 0\end{bmatrix}\begin{bmatrix}\Delta\dot{\theta}\\ \mu\end{bmatrix}=\begin{bmatrix}0\\ -C_c^+\dot{\theta}^-\end{bmatrix} \tag{19}$$

where, $\Delta\dot{\theta}$ denotes the jump in velocities and μ denotes the total impulse during connection at the contact point.

2.3. Optimization metrics during transformation

From (19), it can be observed that the impulse is primarily induced by topology changes and the end velocities prior to transformation. However, the end velocities cannot perfectly reach 0. Thus, some impulse is unavoidable during topology changes; in other words, an impact is also unavoidable, although potentially small. These impacts occur at every joint and contact point within the system and can lead to irreversible damage, such as the shedding of solid lubrication. This characteristic directly influences space robots' lifespan; when damage accumulates to a critical threshold, joint failure occurs, rendering the robot incapable of continuing operations. Therefore, minimizing the impact associated with each topology change process is essential for efficiently functioning space robots. Given the difficulties in quantifying the impact force without detailed knowledge of the connection mechanism design, this paper utilizes impulse as the integral of the impact force over time as the optimization metric. To derive the impulse in every joint, integrate (8) over the short interval Δt and the following can be obtained:

$$L=M_1\Delta\dot{\theta}+\lim_{\Delta t\to 0}\int_{t^-}^{t^+}K_1\theta+D_1\dot{\theta}+C_1dt=M_1\Delta\dot{\theta} \tag{20}$$

where L denotes the total impulse in all the joints, C_1 represents all the nonlinear items of every link, including Coriolis forces, centrifugal forces and elastic forces. By substituting the $\Delta\dot{\theta}$ from (19) into the (20), the impulse experienced by all joints during the connection process can be calculated.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

For the space robot depicted in Figure 1, each joint is constrained to revolve only around the X-axis within its respective body-based coordinate frame, commonly called the driven direction. Since each joint possesses degrees of freedom to rotate in its driven direction, the impulse applied along this axis will not cause damage; instead, it will result in rotation. Conversely, impulses in non-driven directions are significant sources of potential damage and must be minimized. This requirement can be articulated as follows:

$$L(k) = \begin{bmatrix} L_k^{rx} \\ L_k^{ry} \\ L_k^{rz} \\ L_k^{tx} \\ L_k^{ty} \\ L_k^{tz} \end{bmatrix} \rightarrow L'(k) = \begin{bmatrix} L_k^{ry} \\ L_k^{rz} \\ L_k^{tx} \\ L_k^{ty} \\ L_k^{tz} \end{bmatrix} \tag{21}$$

So the optimization metrics can be defined as:

$$F(\theta) = \Sigma \left\| L'(k) \right\|_2 \tag{22}$$

3. Bi-level optimization model with impulse minimization

The optimization problem addressed in this work aims to determine the optimal joint angle configuration and contact poses to minimize impulse in non-driven directions. Since contact poses are functions of angles, solving for the optimal joint angle configuration alone is sufficient. These angles must satisfy close-loop constraints to ensure the ends of the two arms can connect. Traditionally, such configuration optimization is modelled as a constrained optimization problem and solved directly using various algorithms. However, due to the redundancy of the two arms, the problem presented here involves many polar points, making it susceptible to convergence on local optimal solutions. To address this challenge, a bilevel optimization model is proposed, in which one problem is embedded within another. The outer optimization task is the upper-level optimization task, while the inner optimization task is the lower-level optimization task.

3.1. Lower-level optimization model

For the lower-level optimization, the contact position and orientation with respect to the base x_0 are given. This optimization aims to determine the optimal joint angles to minimize the impulse. These joint angles should satisfy the constraints to ensure the ends reach the specified contact position and orientation. Additionally, they must adhere to self-avoidance constraints to prevent interference among individual links. The range of joint angles is limited to $[-\pi, \pi]$. The lower-level optimization problem can be described as:

$$\begin{aligned} \text{Min } & f(\theta) = F(\theta, x_0) \\ \text{s.t. } & \Phi_c^l(\theta, x_0) = 0 \\ & \Phi_o^l(\theta, x_0) = 0 \\ & \theta_i \in [-\pi, \pi] \end{aligned} \tag{23}$$

where superscript l indicates constraints in lower-level optimization; $\Phi_c^l(\theta)$ represents the constraints that ensure the ends of the two arms achieve the specified contact position and orientation; $\Phi_o^l(\theta)$ denotes the self-avoidance constraint. $\Phi_c^l(\theta)$ can be expressed as residuals resulting from the solution of inverse kinematics equations:

$$\Phi_c^l(\theta, x_0) = a_1 \cdot R_1 + a_2 \cdot R_2 \tag{24}$$

where R_1 and R_2 denote the residuals in solving inverse kinematics equations for the two arms, respectively, a_1 and a_2 represent close-loop coefficients.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

The self-avoidance constraint is formulated using the artificial potential field method. The repulsive force can be calculated as follows:

$$f_r(\theta) = \begin{cases} \sum (k_r \cdot (\frac{1}{d} - \frac{1}{d_0})^2) & d \le d_0 \\ 0 & d > d_0 \end{cases} \tag{25}$$

where k_r denotes the repulsion coefficient; d_0 represents the safety distance threshold. The repulsive force can be directly utilized as the self-avoidance constraint $\Phi_o^l(\theta)$.

Most existing optimization algorithms cannot handle constraints directly. A common method is to reformulate the constraint conditions as penalty terms and incorporate them into the objective function. For the inequality constraint $\theta_i \in [-\pi, \pi]$, and considering the periodic nature of the joints, when a joint angle exceeds these bounds, it is corrected to fall within the allowable range by adding or subtracting 2π to it. Then, the lower-level optimization problem can be expressed as:

$$\begin{aligned} &\text{Min } f(\theta, x_0) = F(\theta, x_0) + \Phi_c^l(\theta, x_0) + \Phi_o^l(\theta, x_0) \\ &\text{s.t. } \theta_i \in [-\pi, \pi] \end{aligned} \tag{26}$$

3.2. Upper-level optimization model

The upper-level optimization aims to determine the optimal contact position and orientation that minimize the impulse. The contact position and orientation are utilized in the lower-level to calculate the minimal impulse, with the optimal value serving as the objective metric in the upper-level. The constraints include ensuring that the contact pose lies within the dual-arm public workspace, thereby making the pose reachable. Additionally, the self-avoidance constraint is incorporated. Thus, the upper-level optimization problem can be formulated as follows:

$$\begin{aligned} &\text{Min } f(x) = f_{best}^l(x) \\ &\text{s.t. } \begin{matrix} \Phi_c^u(x) = 0 \\ \Phi_o^u(x) = 0 \end{matrix} \end{aligned} \tag{27}$$

where x denotes the vector of contact position and orientation, and the superscript u denotes constraints in the upper-level optimization. The term $f_{best}^l(x)$ is the optimal value of the lower-level optimization for a given contact pose x; $\Phi_c^u(x)$ denotes the constraints to ensure the pose is reachable; $\Phi_o^u(x)$ means the self-avoidance constraint.

These two constraints are all considered in the lower-level optimization. With well-chosen coefficients, it may not be necessary to consider these constraints explicitly at the upper level. For example, if the contact position and orientation are unreachable, the residuals yield a significant value. Given an appropriately chosen coefficient, the objective metrics become significantly larger than the optimal impulse, automatically excluding unreachable poses. The same applies to the self-avoidance constraint. Consequently, the upper-level optimization problem can be simplified as follows:

$$\text{Min } f(x) = f_{best}^l(x) \tag{28}$$

which is an unconstrained optimization problem that is much easier to solve than the original one.

In summary, the bilevel problem can be expressed as:

$$\begin{aligned} &\text{Min } f(x, \theta) = F(x, \theta) + \Phi_c^l(\theta, x) + \Phi_o^l(\theta, x) \\ &\text{s.t. } \theta_i \in [-\pi, \pi] \end{aligned} \tag{29}$$

4. Heuristic based algorithm to solve bilevel optimization

4.1. Bilevel optimization solving approach

The model established in Section 3 represents a typical bilevel optimization problem, wherein one optimization task is embedded with another. Consequently, each iteration of the upper-level optimization necessitates solving for the optimal solution of the lower-level optimization. This solution process is

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

computationally intensive and can be exceedingly time-consuming. Additionally, the problem is strongly coupled and highly nonlinear, complicating the search for an optimal solution. The most common approach to addressing a bilevel problem involves reforming it into an equivalent single-level problem. This is achieved based on the necessary and sufficient conditions of Karush Kuhn Tucker associated with the lower-level problem [34]. However, this reformulation is only feasible when the lower-level problem is convex, and this cannot be applied to the non-convex problem presented in this work. Moreover, it is also impossible to transfer the problem into a single-level one. Because this will lead to the solution being stuck in local optimality, as discussed in Section 3.

Classical methods make it hard to solve the problem proposed due to their high nonlinearity and non-convexity. However, heuristic algorithms are well-suited to solving bilevel problems owing to their advantages in handling nonlinearities and achieving computational efficiency. This work employs PSO [35] to solve the lower-level optimization problem and find the global optimal solution. At the same time, the SA method is utilized for the upper-level optimization to enhance solving speed. For lower-level optimization, the objective function is represented as (26), an unconstrained optimization problem where coefficients can adjust the penalty degree. The PSO algorithm randomly searches each particle based on swarm intelligence. In n-dimensional optimization space, the individual position and velocity of the i-th particle are denoted as $q(i) = \left(q_1(i), q_2(i), \ldots q_n(i)\right)$ and $v(i) = \left(v_1(i), v_2(i), \ldots v_n(i)\right)$, respectively. The individual optimal position for the i-th particle is $q_{best}(i)$, and the global optimal particle position is denoted as q_{global}. The update strategy for the particle is as follows:

$$\begin{cases} v_{k+1}(i) = \mu v_k(i) + c_1 r_1 (q_{best}(i) - q_k(i)) + c_2 r_2 (q_{global} - q_k(i)), \\ q_{k+1}(i) = q_k(i) + v_{k+1}(i) \end{cases} \tag{30}$$

where c_1 and c_2 are learning factors; r_1 and r_2 are random numbers between 0 and 1; μ is the inertia weight factor.

The update process in the PSO algorithm incorporates both the individual optimal values and the global optimal value, thereby endowing it with robust global search capabilities. Furthermore, as particles share each other's best experiences – namely, their individual best positions and the global best position, PSO can rapidly converge on improved solutions. The interactions among particles influence their positions, enhancing the search process's overall efficiency. Consequently, PSO is employed as the solving algorithm for lower-level optimization to identify the global optimal solution effectively.

In the upper-level optimization, the objective function, defined in (28), is structured as an unconstrained optimization problem. The penalty level can be fine-tuned by adjusting coefficients within the penalty terms. Given the computationally intensive nature of bilevel optimization, the upper-level algorithm should operate efficiently with minimal steps to accelerate the overall solving process. Unlike PSO, which requires multiple evaluations per step, simulated annealing (SA) calculates the objective function only once per iteration, making it more suitable for the upper-level optimization. SA finds the global optimal solution using the Metropolis principle [30], which probabilistically accepts new states rather than relying solely on deterministic rules. This acceptance probability is represented as:

$$P = \begin{cases} 1, & f(k+1) < f(k) \\ e^{-\frac{f(k+1)-f(k)}{T}}, & f(k+1) \geq f(k) \end{cases} \tag{31}$$

In general, simulated annealing algorithms work as follows. The temperature progressively decreases from an initial positive value to zero. At each time step, the algorithm randomly selects a solution close to the current one, measures its quality, and moves to it according to the temperature-dependent probabilities of selecting better or worse solutions, which during the search respectively remain at 1 (or positive) and decrease toward zero.

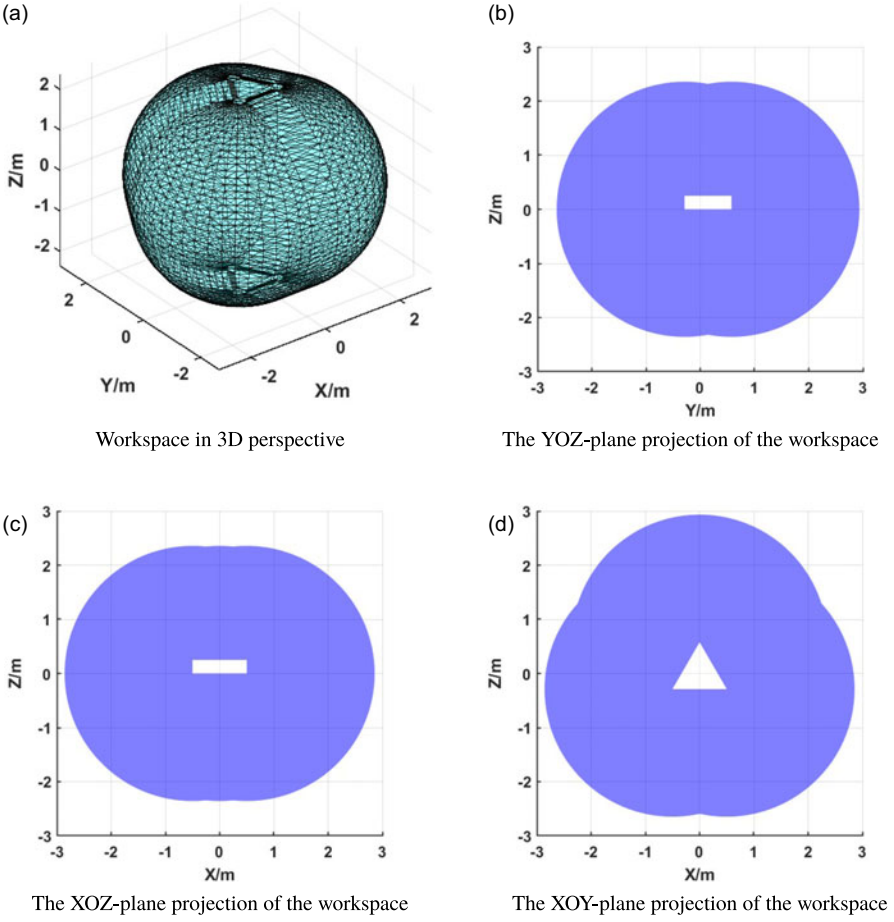
4.2. Optimization results

In this subsection, the optimization problem in Section 3 is solved with the algorithm in Section 4.1. The detailed parameters of the space robot employed are shown in Table I. Here, the No means the link

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

Table I. *Parameter for space robot.*

No	Mass/kg	Length/m
1	0.7719	(0.05,0,0)
2	0.5216	(0.06,0.07,0)
3	0.5216	(0.07,0.06,0)
4	4.87	(0.06,1,0)
5	4.87	(1,0.06,0)
6	0.5216	(0.06,0.07,0)
7	0.5216	(0.07,0.06,0)
Base	5	L:1 H:0.3



Workspace in 3D perspective

The YOZ-plane projection of the workspace

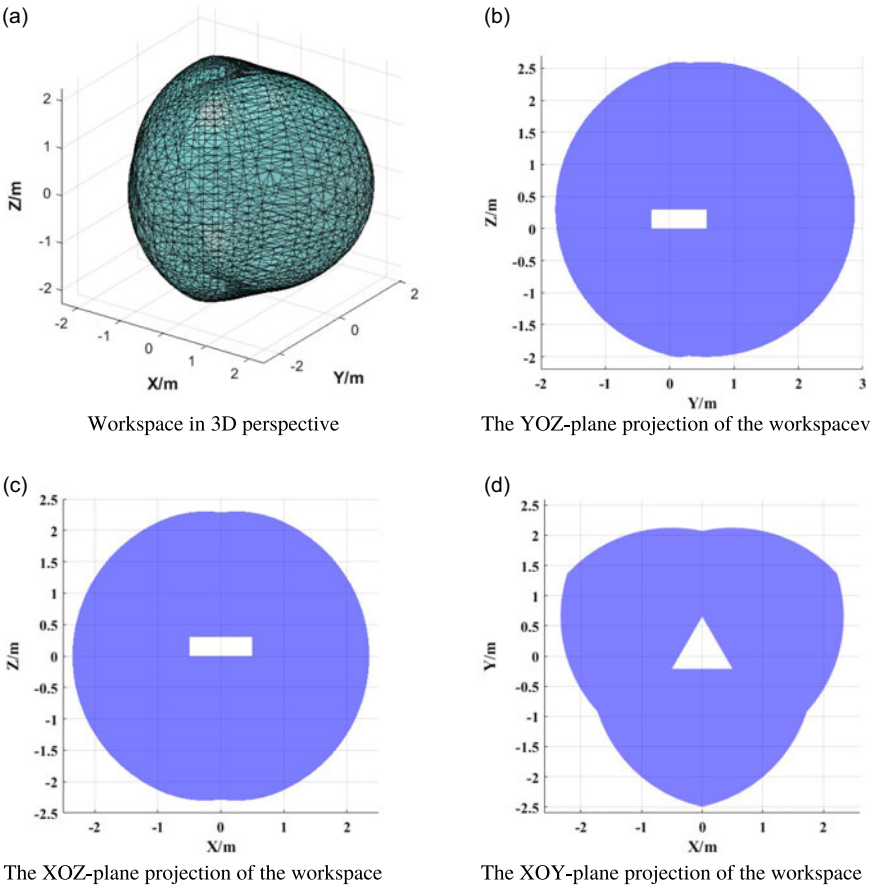
The XOZ-plane projection of the workspace

The XOY-plane projection of the workspace

Figure 4. *Diagram of workspace.*

number, as shown in Figure 2. The length for the base means the base refers to a triangular prism with sides 1 m long and 0.3 m high. The link parameters listed are those of links in arm 1; the parameters for arm 2 and arm 3 are the same. If the base is fixed, the workspace for this space robot is shown in Figure 4. Considering there are no more constraints on the robot, the workspace for each arm is a sphere centred at the installation point of the arm on the base, with a radius equal to the maximum reach of the arm. Additionally, the volume occupied by the base must be subtracted from this sphere. And the whole workspace for the system is the union of the workspaces of all individual arms.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press



Workspace in 3D perspective

The YOZ-plane projection of the workspacev

The XOZ-plane projection of the workspace

The XOY-plane projection of the workspace

Figure 5. *Diagram of collaborative manipulation workspace.*

And the collaborative manipulation workspace for every two arms is shown in Figure 5.

During the transformation process, arm 1 and arm 2 move together until their ends are entirely in contact. Due to the control errors, the velocities of the two ends cannot reach 0 perfectly. Therefore, assume that there is a residual end velocity of 0.001 m/s along the normal to both ends upon contact. Meanwhile, arm 3 and the base remain stationary before reconfiguration.

Before simulation, some key information about the optimization must be introduced:

- Optimization objective: Minimization of impulse in non-actuated directions, as shown in (22).
- Optimization constraint: Self obstacle-avoidance and close-loop constraint.
- Design variables: Joint angle configuration(lower-level optimization)/contact pose(upper-level optimization).
- Initial guesses: θ_0(lower-level optimization)/x_0(upper-level. optimization)
- Simulation platform: Matlab R2023b with Intel i7 8086K at 4.00 GHz, without GPU acceleration or parallel computing.

where the θ_0 and x_0 denote the initial guess joint angle configuration and the contact pose, their detailed values are as follows. Based on the initial value above, the impulse before optimization is 0.0031 N · s. The joint configuration under initial condition is shown in Figure 6.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

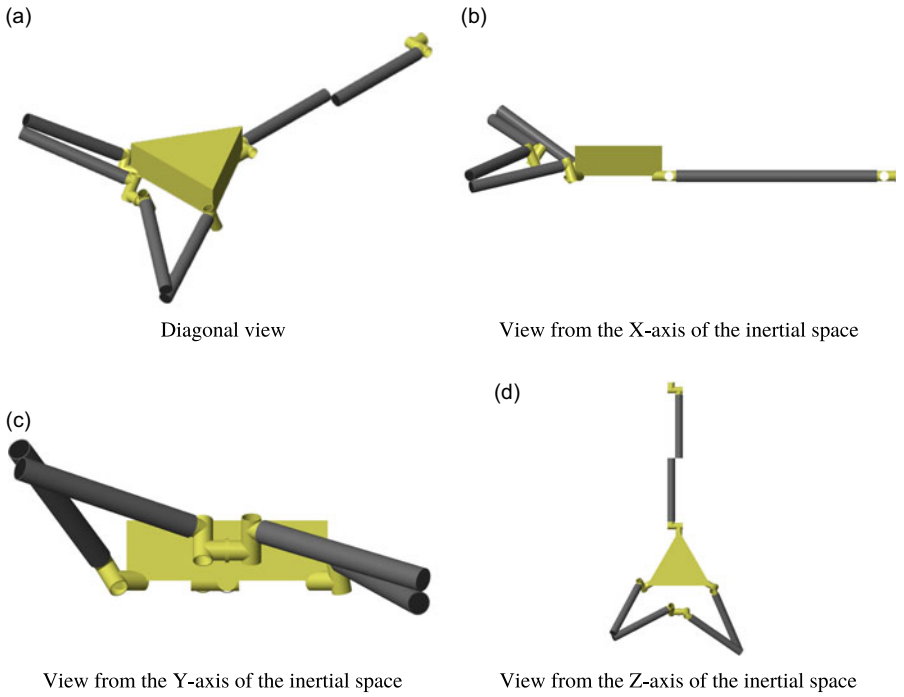


Figure 6. *Diagram of space robot at initial contact pose.*

$$\theta_0 = \begin{bmatrix} -0.9945 \\ 0.5565 \\ -3.0549 \\ -2.5366 \\ -0.3858 \\ 0.7727 \\ 2.4558 \\ -1.1606 \\ -0.6452 \\ -0.3217 \\ -2.5937 \\ 2.7493 \\ -0.6720 \\ 1.9168 \end{bmatrix} \text{rad} \tag{32}$$

$$x_0 = \left[0\,\text{m},\ -0.7\,\text{m},\ 0\,\text{m},\ 0\,\text{rad},\ 0\,\text{rad},\ 0\,\text{rad}\right]^T$$

First, a lower-level optimization is conducted to verify the effectiveness of lower-level optimization strategy. The solving processes are all run based on MATLAB Optimization Toolbox. The parameters for upper-level optimization with PSO methods are as follows:

- Particle swarm optimization: Swarm size: 10
- Max iterations: 100
- Function Tolerance: 10,000
- Learning factors c_1,c_2: 1.49
- Close-loop coefficient a_1,a_2: 1.49

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

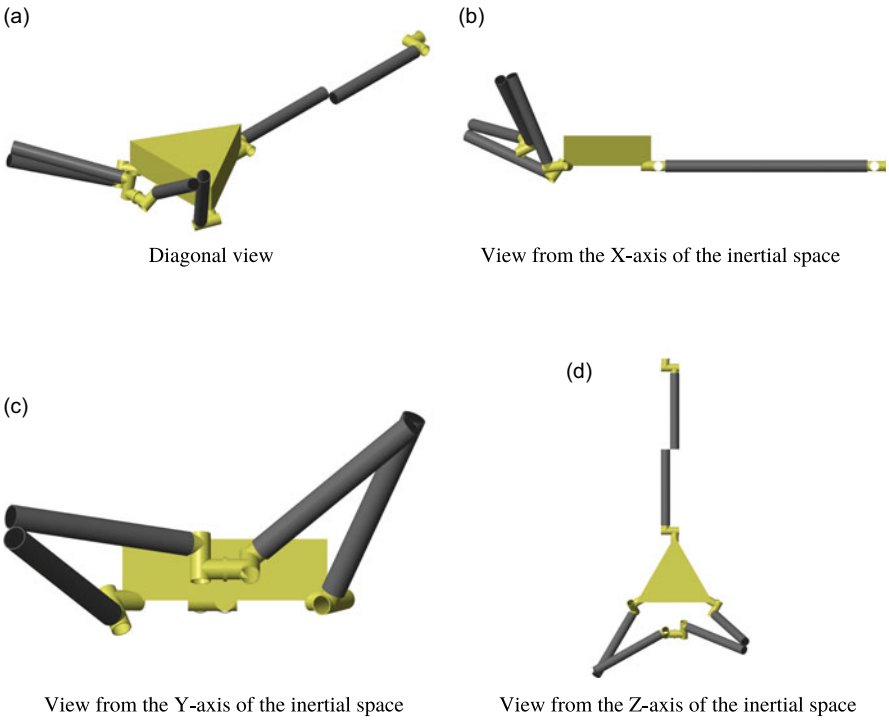


Figure 7. *Diagram of space robot after lower-level optimization.*

- Repulsion coefficient k_r: 5

The configuration after optimization is as shown in Figure 7, and the angles is as (33).

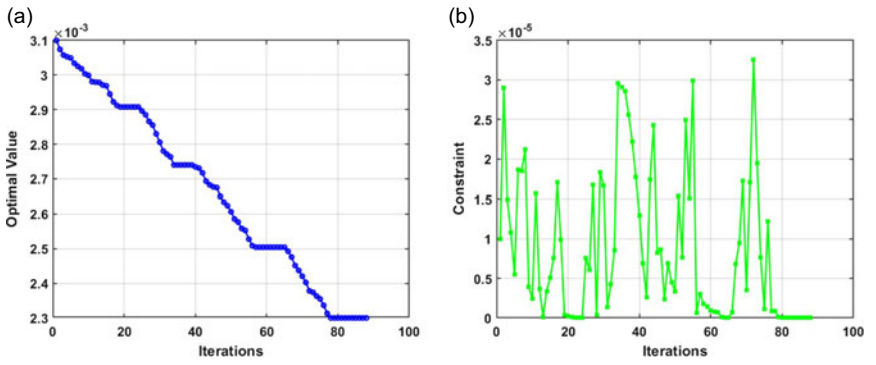
$$\theta_1 = \begin{bmatrix} -1.2994 \\ 0.5307 \\ 0.2738 \\ 2.5763 \\ -0.2071 \\ -0.6296 \\ -0.8879 \\ -2.6847 \\ -0.7455 \\ 2.9956 \\ 2.7149 \\ 3.5991 \\ 1.0266 \\ -3.0143 \end{bmatrix} \text{rad} \tag{33}$$

$$F_1 = 0.0022\ \text{N} \cdot \text{s}$$

where the θ_1 and F_1 denote the configuration angles after the lower-level optimization and the impulse under this condition. The evolutions of the optimality criteria and constraints are shown in Figure 8.

From the results, the configuration angles are well optimized and the impulse decreases from 0.0031 to 0.0022 N · s. From Figures 6 and 7, the obstacle avoidance constraints and close-loop constraints are well promised. It means that the lower-level optimization strategy is effective for impulse minimization. Thus the bilevel optimization can be further conducted. The computation time for the lower-level optimization is 2142 s.

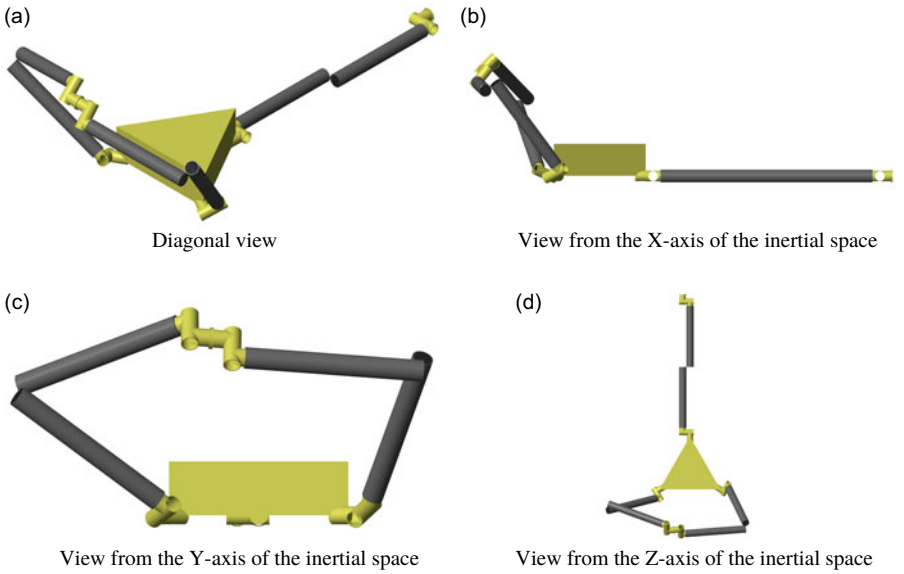
https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press



The evolution of the optimality criteria

The evolution of the constraint

Figure 8. *The evolutions of lower-level optimization.*



Diagonal view

View from the X-axis of the inertial space

View from the Y-axis of the inertial space

View from the Z-axis of the inertial space

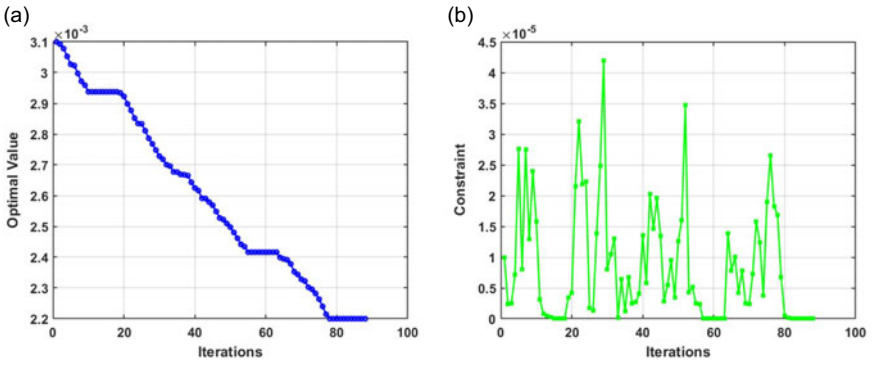
Figure 9. *Diagram of space robot after bilevel optimization.*

For the bi-level optimization, the model and algorithm are as described in Sections 3 and 4. The solving processes are all run based on MATLAB Optimization Toolbox. The parameters for upper-level optimization with SA methods are as follows:

- Simulating annealing:
- Max iterations: 100
- Function Tolerance: 1e-3
- Initial temperature: 1.0
- Cooling function: temperature fast

The configuration after bilevel optimization is as shown in Figure 9, and the angles is as (34).

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press



The evolution of the optimality criteria The evolution of the constraint

Figure 10. *The evolutions of bi-level optimization.*

$$\theta_2 = \begin{bmatrix} 0.0712 \\ -0.4193 \\ -1.1062 \\ 2.1076 \\ -0.1311 \\ -0.4253 \\ -0.4155 \\ 2.9746 \\ 0.0113 \\ 0.5801 \\ -1.8031 \\ -2.8318 \\ -0.9404 \\ 0.6106 \end{bmatrix} \text{rad} \tag{34}$$

$$F_2 = 0.0013\ \text{N} \cdot \text{s}$$

where the θ_2 and F_2 denote the configuration angles after the bilevel optimization and the impulse under this condition. The evolutions of the optimality criteria and constraints are shown in Figure 10.

After the optimization, the minimum impulse is: 0.0013 N · s, the contact pose is $[-0.265\ \text{m}, -0.952\ \text{m}, 0.835\ \text{m}, 0.360\ \text{rad}, 0.114\ \text{rad}, 0.055\ \text{rad}]^T$, the computation time for the bilevel optimization is 33,784 s.

The optimization results demonstrate a reduction in impulse from 0.0031 to 0.0013 N · s. This suggests that the proposed optimization methods can effectively be employed to design the optimal configuration for this type of space robot. If the total impact that the solid lubricant layer can stand is assumed to be a constant $L_{\max}$, the maximum number of reconfigurations allowed for the original configuration is $L_{ma}/0.0013$, and that after optimization is $L_{ma}/0.0031$. It means that the optimal transformation method can increase the allowable number of reconfigurations for the robotic system by approximately 238%, in other words, extending its operational lifespan by 238%. In other words, a single robot employing the optimized reconfiguration strategy can accomplish tasks that previously required at least two robots, which significantly reduces the mission costs. The whole optimization results can be summarized in Table II and Figure 11.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

Table II. *Optimization results.*

	Initial	After lower-level optimization	After bilevel optimization
Impulse (N · s)	0.0031	0.0022	0.0013
Configuration angles (rad)	θ_0	θ_1	θ_2
Obstacle constraints	✓	✓	✓
Close-loop constraints	✓	✓	✓

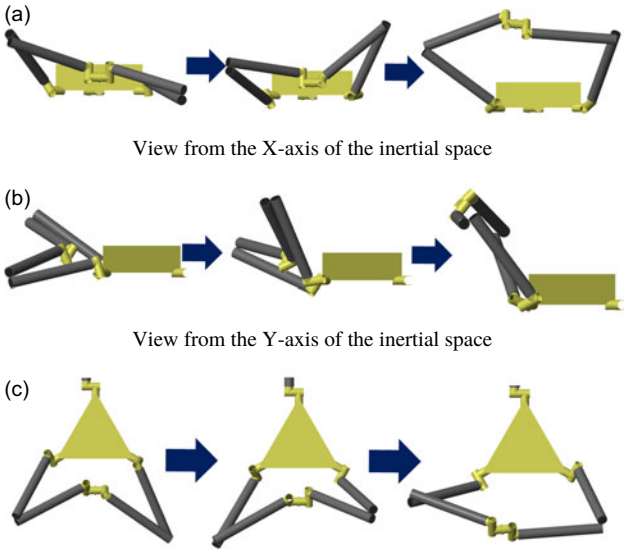


Figure 11. *Diagram of configuration changes.*

5. Conclusions

This paper presents a configuration optimization method for variable topological space robots to minimize impulse through a bilevel optimization approach. A universal dynamic model for space robot topology transformation is established using spatial operator algebra and impulse-momentum equations. It eliminates the need to consider specific connection mechanism designs. Given the challenges associated with calculating impact, the impulse of the entire system is derived as the primary metric for optimization. The configuration optimization model is structured as a bilevel optimization problem, where the lower-level task focuses on determining the optimal angle configuration with a given contact pose. And the upper-level task aims to identify the optimal contact pose. Several constraints are incorporated into this optimization framework. A heuristic-based algorithm is proposed for problem resolution, employing particle swarm optimization for the lower-level task and the simulated annealing algorithm for the upper-level task. The optimization results indicate that the proposed method effectively determines the optimal configuration for space robots with variable topologies. Future research will primarily concentrate on developing control strategies and assessing performance during the configuration transformation process [36].

Author contributions. HW, JG, CY and XC conceived and designed the study. HW and CC conducted data gathering. HW performed statistical analyses. HW, CC and DC wrote the article.

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

Financial support. This research receives grants from "National Natural Science Foundation of China" and the grant number is 12372045.

Competing interest. The authors declare no conflicts of interest exist.

Ethical approval. Not applicable.

References

[1] W. K. Belvin, W. R. Doggett, J. J. Watson, J. T. Dorsey, J. E. Warren, T. C. Jones, E. E. Komendera, T. Mann and L. M. Bowman, "In-Space Structural Assembly: Applications and Technology," *3rd AIAA Spacecraft Structures Conference* (2016) p. 2163.
[2] Y. Zhou, J. Luo and M. Wang, "Dynamic manipulability analysis of multi-arm space robot," *Robotica* **39**(1), 23–41 (2021).
[3] X. Fu, H. Ai and L. Chen, "Integrated sliding mode control with input restriction, output feedback and repetitive learning for space robot with flexible-base, flexible-link and flexible-joint," *Robotica* **41**(1), 370–391 (2023).
[4] Y.-H. Wu, Z.-C. Yu, C.-Y. Li, M.-J. He, B. Hua and Z.-M. Chen, "Reinforcement learning in dual-arm trajectory planning for a free-floating space robot," *Aerospace Sci. Technol.* **98**, 105657 (2020).
[5] S. Kalaycioglu and A. de Ruiter, "Dual arm coordination of redundant space manipulators mounted on a spacecraft," *Robotica* **41**(8), 2489–2518 (2023).
[6] C. Yue, T. Lin, X. Zhang, X. Chen and X. Cao, "Hierarchical path planning for multi-arm spacecraft with general translational and rotational locomotion mode," *Sci. China Technol. Sci.* **66**(4), 1180–1191 (2023).
[7] D. Shang, X. Li, M. Yin, J. Liu and S. Zhou, "Dynamic modeling and tracking control strategy for flexible telescopic space manipulator based on neural network compensation," *Adv. Space Res.* **72**(9), 3645–3665 (2023).
[8] P.-C. Cong and X. Zhang, "Preimpact configuration analysis of a dual-arm space manipulator with a prismatic joint for capturing an object," *Robotica* **31**(6), 853–860 (2013).
[9] L. Yan, W. Xu, Z. Hu and B. Liang, "Multi-objective configuration optimization for coordinated capture of dual-arm space robot," *Acta Astronaut.* **167**, 189–200 (2020).
[10] X. Lu and Y. Jia, "Trajectory planning of free-floating space manipulators with spacecraft attitude stabilization and manipulability optimization," *IEEE Trans. Syst. Man Cybern. Syst.* **51**(12), 7346–7362 (2020).
[11] Y. Takao, O. Mori and J. Kawaguchi, "Analysis and design of a spacecraft docking system using a deployable boom," *Acta Astronaut.* **179**, 172–185 (2021).
[12] K. Daneshjou and R. Alibakhshi, "Multibody dynamical modeling for spacecraft docking process with spring-damper buffering device: A new validation approach," *Adv. Space Res.* **61**(1), 497–512 (2018).
[13] Z. Zhao, C. Liu and T. Chen, "Docking dynamics between two spacecrafts with apdses," *Multibody Syst. Dyn.* **37**(3), 245–270 (2016).
[14] E. J. Haug, S. C. Wu and S. M. Yang, "Dynamics of mechanical systems with coulomb friction, stiction, impact and constraint addition-deletion–i theory," *Mech. Mach. Theory* **21**(5), 401–406 (1986).
[15] Y. Khulief, "Numerical modelling of impulsive events in mechanical systems," *Int. J. Model. Simul.* **30**(1), 80–86 (2010).
[16] C.-C. Chang and S.-T. Peng, "Impulsive motion of multibody systems," *Multibody Syst. Dyn.* **17**(1), 47–70 (2007).
[17] D. Raina, S. Gora, D. Maheshwari and S. V. Shah, "Impact modeling and reactionless control for post-capturing and maneuvering of orbiting objects using a multi-arm space robot," *Acta Astronaut.* **182**, 21–36 (2021).
[18] B. R. Jouybari, K. G. Osgouie and A. Meghdari, "Optimization of kinematic redundancy and workspace analysis of a dual-arm cam-lock robot," *Robotica* **34**(1), 23–42 (2016).
[19] Y. Zhou, J. Luo and M. Wang, "Grasping force optimization for dual-arm space robot after capturing target based on task compatibility," *Adv. Space Res.* **70**(5), 1496–1511 (2022).
[20] B. Zhang, B. Liang, X. Wang, G. Li, Z. Chen and X. Zhu, "Manipulability measure of dual-arm space robot and its application to design an optimal configuration," *Acta Astronaut.* **128**, 322–329 (2016).
[21] R. Xu, J. Luo and M. Wang, "Optimal grasping pose for dual-arm space robot cooperative manipulation based on global manipulability," *Acta Astronaut.* **183**, 300–309 (2021).
[22] C. Toglia, M. Sabatini, P. Gasbarri and G. B. Palmerini, "Optimal target grasping of a flexible space manipulator for a class of objectives," *Acta Astronaut.* **68**(7-8), 1031–1041 (2011).
[23] H. Lee, J. Sohn and J. Lee, "Optimization of Dual-Arm Configurations for Multiple Object Handling," *2012 IEEE International Symposium on Industrial Electronics* (IEEE, 2012) pp. 1291–1296.
[24] K. G. Osgouie, A. Meghdari, S. Sohrabpour and M. S. Jelodar, "Genetic Algorithm Based Optimization for Dual-Arm Cam-Lock Robot Configuration," *2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (IEEE, 2007) pp. 1–6.
[25] Z. Xue, J. Liu, Y. Li and J. Liu, "Configuration optimization of a dual-arm reconfigurable space robot based on closed-chain inertia matching," *IEEE Trans. Autom. Sci. Eng.* **22**, 6421–6438 (2024).
[26] J. Zhao, X. Yang, Z. Zhao, G. Yang and L. Zhao, "Inverse kinematics and multi-objective configuration optimization of the SSRMS manipulator," *Adv. Space Res.* **72**(9), 3580–3594 (2023).
[27] B. Colson, P. Marcotte and G. Savard, "Bilevel programming: A survey," *4OR* **3**, 87–107 (2005).

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press

[28] S. Zhang, A. Li, J. Ren and R. Ren, "Kinematics inverse solution of assembly robot based on improved particle swarm optimization," *Robotica* **42**(3), 833–845 (2024).
[29] J. Fusic and R. Sitharthan, "Self-adaptive learning particle swarm optimization-based path planning of mobile robot using 2D Lidar environment," *Robotica* **42**(4), 977–1000 (2024).
[30] D. Delahaye, S. Chaimatanan and M. Mongeau, *Simulated Annealing: From Basics to Applications* (Springer International Publishing, Cham, 2019) pp. 1–35.
[31] J. Malzahn, N. Kashiri, E. Romiti, L. Baccelliere, S. Cordasco, A. Laurenzi, A. Margan, E. M. Hoffman, L. Muratore, and N. Tsagarakis, "Modular configurable robot, corresponding method and computer program product. US Patent App. 17/787,819.
[32] J. Hu and T. Wang, "An efficient high-precision recursive dynamic algorithm for closed-loop multibody systems," *Int. J. Numer. Methods Eng.* **118**(4), 181–208 (2019).
[33] A. Jain and G. Rodriguez, "Recursive flexible multibody system dynamics using spatial operators," *J. Guid. Control Dyn.* **15**(6), 1453–1466 (1992).
[34] J.-F. Camacho-Vallejo, C. Corpus and J. G. Villegas, "Metaheuristics for bilevel optimization: A comprehensive review," *Comput. Oper. Res.* **161**, 106410 (2024).
[35] J. Liang, H. Guo, K. Chen, K. Yu, C. Yue and X. Li, "An improved kalman particle swarm optimization for modeling and optimizing of boiler combustion characteristics," *Robotica* **41**(4), 1087–1097 (2023).
[36] T. Lin, C. Yue, Z. Liu and X. Cao, "Modular multi-level replanning tamp framework for dynamic environment," *IEEE Robot. Autom. Lett.* **9**(5), 4234–4241 (2024).

Cite this article: H. Wang, J. Guo, C. Canali, C. Yue, X. Cao and D. G. Caldwell (2025). "Configuration optimization of variable topological space robot for impulse minimization based on bilevel approach", Robotica **43**, 2193–2211. https://doi.org/10.1017/S0263574725101756

https://doi.org/10.1017/S0263574725101756 Published online by Cambridge University Press