Nonholonomic Reorientation of Free-flying Space Robots Using Parallelogram Actuation in Joint Space

Yuki Kubo *
The University of Tokyo, Tokyo 113-8656, Japan

Junichiro Kawaguchi†
Tohoku University, Sendai 980-8579, Japan

A robot with large-degree-of-freedom joints is a promising future space robot capable of adapting to various environments and can perform dexterous tasks with multiple manipulators. The attitude dynamics of a free-flying robot shows nonholonomy, which enables the robot to reorient its attitude by moving its body. However, previous studies were not generally able to handle the nonholonomy, especially for robots with large degrees of freedom of joints. In the present study, we analytically investigate a maneuver in which joints are actuated along a parallelogram trajectory in joint angle space and propose an analytical path modification method in joint angle space in order to achieve the target attitude. Parallelogram actuation guarantees that the body configuration is returned to the initial state after one set of maneuvers, and thus the attitude of the robot is independently reoriented to the target maintaining the body configuration. The analytical solution is provided by applying Magnus expansion to the kinematics equation of rotational matrices, and its Lie group structure contributes to concise mathematical expressions. In addition, the analytical solution does not cause numerical difficulties, such as combinatorial explosion, and so can fully make use of the reorientation ability of a robot with large degrees of freedom.

Nomenclature

C = Direction cosine matrix (rotational matrix)

d = Second-order term of a generalized velocity

$f_X^{\ddagger}$ = Inertial force exerted on the center of mass of body X, N

f_X = External force exerted on the center of mass of body X, N

h_c = Angular momentum around the center of mass of the entire spacecraft, $\mathrm{kg} \cdot \mathrm{m}^2/\mathrm{s}$

I_X = Moment of inertia of body X around the center of mass of body X, $\mathrm{kg} \cdot \mathrm{m}^2$

*Ph.D. Candidate, Department of Aeronautics and Astronautics, 7-3-1 Hongo, Bunkyo-ku, kubo.yuki@ac.jaxa.jp

†Specially Appointed Professor, Department of Aerospace Engineering, 6-6 Aramaki Aza Aoba, Aoba-ku, Sendai, Miyagi, junichiro.kawaguchi.d1@tohoku.ac.jp

M = Generalized mass matrix

m_X = Total mass of body X, kg

$n_X^{\ddagger}$ = Inertial torque exerted on the center of mass of body X, N $\cdot$ m

n_X = External torque exerted on the center of mass of body X, N $\cdot$ m

R_X = Absolute position of the center of mass of body X, m

r_X = Relative position of the center of mass of body X with respect to the center of mass of the first body, m

r_{XY} = Relative position of the center of mass of body X with respect to the center of mass of body Y, m

U = Identity matrix

u = Control input

v = Generalized velocity

w = Derivative of generalized velocity

θ = Joint angle vector (each component holds the angular displacement of a joint), rad

θ^k = Angular displacement of the k-th joint, rad

λ_k = Rotational axis of the k-th joint

τ = Generalized force

Φ = State transition matrix

ω = Angular velocity vector, rad/s

$(\cdot)_k$ = Subscript indicating the k-th body

$(\cdot)_{\mathrm{c}}$ = Subscript indicating the entire spacecraft

$(\cdot)_{\hat{k}}$ = Subscript indicating the k-th outer group

$(\cdot)_{\underline{k}}$ = Subscript indicating the inside neighbor of the k-th body

$(\cdot)_{\mathrm{h}_k}$ = Subscript indicating the k-th joint

$(\cdot)^{\times}$ = Skew symmetric matrix of a vector $(\cdot)$

$(\cdot)^{\vee}$ = Inverse operator of $(\cdot)^{\times}$

$(\cdot)_j^i$ = (i, j) component of a matrix $(\cdot)$

$(\cdot)^{\dagger}$ = Pseudo-inverse of a matrix $(\cdot)$

I. Introduction

In recent years, space activities have been expanding. Under such circumstances, future space robots (including spacecraft in a broad sense) will need to be, for example, more autonomous, more intelligent, more adaptive, more dexterous, and more economical. In fact, the OSAM-1 mission by NASA will attempt on-orbit refueling and assembly, and the Robotic Servicing of Geosynchronous Satellites program of DARPA is investigating on-orbit satellite repair

and orbit correction in the near future [1, 2]. Such missions indicate a growing demand for space robots that can perform more complicated tasks than conventional manipulation. In this context, a free-flying space robot that has large degrees of freedom of active joints is promising as a future space robot in the following aspects:

1) the robot can be adaptive by reconfiguring itself into multiple different morphologies,
2) the robot can be dexterous by manipulating multiple instruments leveraging its redundancy, and
3) the robot can be economical by saving fuel owing to its nonholonomic reorientation technique.

We refer to such a robot as a transformable free-flying space robot. A typical example of such a transformable free-flying space robot is the "Transformer" spacecraft under consideration mainly by JAXA since 2019. This spacecraft is designed to be able to flexibly adapt to multiple environments by reconfiguring itself into several operation modes. For example, the principal operation modes are a stowed mode in the launch phase, a solar sailing mode suitable for saving fuel consumption, and an observation mode suitable for scientific observation [3]. In addition, the spacecraft can economically maintain its nominal orbit around the sun-earth L2 using solar radiation pressure [4]. Moreover, by using redundant degrees of freedom of manipulators, the system is expected to achieve multi-tasking, such as pointing the telescope toward the target object and simultaneously carrying out solar sailing by pointing a reflective surface toward the sun.

The attitude motion of a transformable free-flying robot is governed by the angular momentum conservation law if no external force/torque is exerted. Due to this dynamic constraint, when a part of the body is actuated, the attitude of the remainder of the body reactively changes. The mechanics has been studied extensively for a long time, mainly regarding space manipulators [5, 6]. These studies have numerically solved a forward problem in which attitude motion is computed under the given joint actuation procedure. Such investigations have also revealed that this attitude motion is nonholonomic. Since the moment of inertia of a transformable robot is not constant when reconfiguring its body, the angular momentum conservation law becomes a non-integrable differential constraint, i.e., a nonholonomic constraint.

Due to this nonholonomy, the attitude of the free-flying robot depends on its joint actuation procedure, and thus can be reoriented by actuating its body [7, 8]. This attitude reorientation maneuver is induced only by internal torque. It consumes electric energy stored in a spacecraft but does not consume any expendable fuel. In addition, this maneuver does not cause momentum accumulation of reaction wheels. These characteristics are preferable to elongate the life-span of the spacecraft. However, in order to solve the inverse problem of designing the joint actuation procedure that achieves the target attitude, the following three issues must be resolved:

1) In a non-holonomic system, state variables cannot reach the target state by time-invariant continuous feedback control. This fact is mathematically proven by the negative aspect of the Brockett's theorem on necessary conditions of asymptotic stability in a nonlinear system [9].
2) The attitude equation of a transformable free-flying robot cannot generally be transformed into a canonical form, except for special types of systems such as a planar system [10]. Therefore it cannot be controlled by common

methods that are effective for such systems.

3) In the case of general three-dimensional attitude motion, an angular velocity vector becomes non-integrable, and therefore, additional nonholonomy needs to be taken into account as well as the nonholonomy derived from the angular momentum conservation law. In the present paper, the former nonholonomy is referred to as *kinematic nonholonomy*, whereas the latter nonholonomy is referred to as *momentum nonholonomy* [11].

In order to solve the above problems, we herein propose a method that we refer to as parallelogram actuation. In this method, joints are actuated along a parallelogram trajectory in joint angle space, which is a multi-dimensional space in which each point corresponds to one set of joint states, i.e., a body configuration. With the aid of rectilinear movement in joint angle space, analytical solution of its attitude motion becomes concise, which helps to solve the aforementioned inverse problem. In addition, the fact that a parallelogram is a closed loop guarantees that the body configuration of a robot returns to the initial state after a set of parallelogram actuations. Therefore, the robot can independently achieve the target attitude while maintaining its initial body configuration. In a previous study, Ohashi et al. also numerically investigated a similar maneuver, which he referred to as nonholonomic turn [12]. The proposed method is applicable to any type of transformable free-flying robot and is widely applicable because the proposed method is expressed in a form that is valid for general three-dimensional attitude motion and is not limited to planar motion. Furthermore, since the solution is expressed analytically, the proposed method can handle large degrees of freedom, which was difficult with conventional combinatorial optimization methods, such as that proposed by Ohashi et al.

Classic problem of the nonholonomic system is sliding or rolling motion of rigid bodies such as skidding knife edges [13] and ground vehicles with rolling wheels [14]. In particular, a system which can be transformed into a chained form has been thoroughly investigated [15, 16]. The chained form is one of the most well-structured set of equations of motion, which makes its control much easier. However, it is difficult to transform general nonholonomic systems into the chained form. A well-known phenomenon regarding the nonholonomic property of free-flying robots is the falling-cat phenomenon. This is a phenomenon in which a cat can always land in an upright attitude by twisting its body in the air, even when dropped in an upside-down attitude. There have been many studies to elucidate the dynamics of this phenomenon [17, 18] and to apply the attitude maneuver of this phenomenon to free-flying robots [19–22]. In the 1980s, since numerous space robots were proposed to construct space stations [23, 24], more studies focused on the dynamics of transformable free-flying robots. Some hardware experiments have demonstrated successful control of space manipulator, e.g., ETS-VII, the first successful demonstrator of on-orbit manipulation [25] and an experiment using a planar air-bearing microgravity simulator or a parabolic flight [26, 27].

The related formulations are roughly classified into two types: Euler-Lagrange formulation and Kane's formulation. The Euler-Lagrange formulation is energy-based and systematic but requires complex differentiation of a Lagrange function [28]. On the other hand, Kane's formulation is based on the Newton-Euler equation. The Newton-Euler formulation generally has difficulty in handling the constraint forces, but Kane solved this problem by performing a

mathematical operation that prevents the constraint forces from appearing in the equations. In general, Kane method is suitable for rapid numerical computation for large degree-of-freedom system [29]. The fundamental equations in the present study are also based on Kane's formulation. Other researchers used different formulations such as Hamiltonian-based method [30].

The principal task of a space manipulator is to control the position and attitude of the end-effector, and the reaction of attitude is usually canceled by thrusters or reaction wheels [31–33]. However, some researchers proposed actively using the attitude reaction when actuating its manipulator. In the case of planar motion, several attitude reorientation methods have been proposed that leverage the nonholonomy of the attitude motion [34–38]. However, since the angular velocity is integrable in planar motion (i.e., the angular velocity can be expressed as the time derivative of the azimuthal angle), the kinematic nonholonomy is not taken into account in these studies. Therefore, the abovementioned methods cannot be directly applied to general three-dimensional attitude control. As for the case of three-dimensional attitude motion, few studies actively use the nonholonomy of attitude motion. The hierarchical Lyapunov method proposed by Nakamura and Mukherjee[8] is one of the few examples by which to achieve the desired end-effector trajectory under some joint constraints using its nonholonomy. In this method, the first-priority Lyapunov function is constructed to obtain the target end-effector state and the second-priority Lyapunov function is constructed to satisfy the constraints on the state space. However, it is known that the constraints of the second Lyapunov function are not always satisfied, and convergence to the target is not always guaranteed in this method. Another example is the method of using sinusoidal input in joint actuation [39, 40]. This method can achieve arbitrary attitude reorientation by repeating sinusoidal joint actuation. However, the applicability of this method is limited because the input is restricted to sinusoids. In addition, this method is not agile because the attitude is gradually shifted to the target by repeating sinusoidal actuation. In addition, none of these methods cover large-scale body reconfiguration that dramatically changes the structure of a robot. Thus, although the dynamics of transformable free-flying robots has been studied for a long time, a method by which to achieve a target morphology and attitude simultaneously has not been sufficiently studied.

Research closely related to our research was conducted by Ohashi et al. [12]. Their method was motion-primitive based and took a comprehensive combination of joint actuation patterns. Their method stored all results of the total attitude changes of the corresponding joint actuation patterns. In this method, the number of combinations increases exponentially with the increase in the number of joints. Therefore, their control method cannot fully make use of the attitude reorientation ability of the robot. In addition, they solved the attitude motion of the robot by using numerical integration, which cannot extract any analytical information, to solve the inverse problem, which limited their method to a brute-force search. Moreover, Ohashi et al. used a heuristic or meta-heuristic algorithm, such as a genetic algorithm, to solve the inverse problem, but this requires a huge calculation for each attitude reorientation planning task.

Most of the abovementioned related studies assume total angular momentum of the system is always conserved and set to zero. In such systems, the angular momentum conservation law serves as a symmetrically affine (driftless)

system. On the other hand, a few studies handle a system with non-zero angular momentum [41, 42]. Such an affine (drifted) system is much more difficult to handle and still needs to be investigated.

In the present paper, we propose an attitude reorientation method by parallelogram actuation with angular momentum always zero. First, we derive an approximate analytical solution for rectilinear actuation in joint angle space. Next, by taking the Jacobian for the analytical solution, a joint path modification law to achieve the target attitude is analytically obtained, which enables us to directly solve the aforementioned inverse problem. Finally, we present an application of the attitude reorientation planning. The proposed method can handle large degrees of freedom of joints without causing combinatorial explosion, as is the case for Ohashi's method. The main findings and limitations of the proposed method are presented after numerical simulation.

II. Preliminary Formulations

In this section, attitude motion of a transformable free-flying robot is formulated. The nomenclature of body components and their positions are shown in Fig. 1. In this representation, an entire spacecraft is divided into a main body, assigned to index $k = 0$, and some other branches rooted from the main body. Here, for convenience of formulation, we assume that each branch has no loop structure. This assumption assures one-directional index assignment from the main body (innermost) to the tip of each branch (outermost). Body indices in each branch are allotted sequentially from an inner body to an outer body. The hinge joint component connecting the k-th body and its inside neighbor is labeled as the k-th joint, and all bodies outside from the k-th joint are labeled as $\hat{k}$, which is referred to as the k-th outer group. (From the above definitions, all indices in the outer group $\hat{k}$ are larger than k.) This $\hat{k}$ grouping contributes to simple formulations because all bodies in the k-th outer group move in the same manner with respect to the main body when the k-th joint is actuated. The rotational degrees of freedom of each hinge joint is set to be 1, and the rotational angle of the k-th joint is described as θ^k. Any joint rotation can be expressed by a combination of 1-D rotations, and hence this assumption does not lose generality. The body frame is fixed to the main body and its origin is at the center of mass (CoM) of the main body. In addition, the orientation of the main body is aligned with the arbitrary geometry of the main body. Note that the position of the CoM of an entire robot is not constant in the body-fixed coordinate as the robot changes its body configuration.

We use Kane's formulation in the present paper, which is, in general, suitable for handling three-dimensional attitude motion [5, 6]. A general form of the dynamics equation can be described as follows:

$$\begin{bmatrix} M_{vv} & M_{v\omega} & M_{v\theta} \\ M_{\omega v} & M_{\omega\omega} & M_{\omega\theta} \\ M_{\theta v} & M_{\theta\omega} & M_{\theta\theta} \end{bmatrix} \begin{bmatrix} w_v \\ w_\omega \\ w_\theta \end{bmatrix} + \begin{bmatrix} d_v \\ d_\omega \\ d_\theta \end{bmatrix} = \begin{bmatrix} \tau_v \\ \tau_\omega \\ \tau_\theta \end{bmatrix} \tag{1}$$

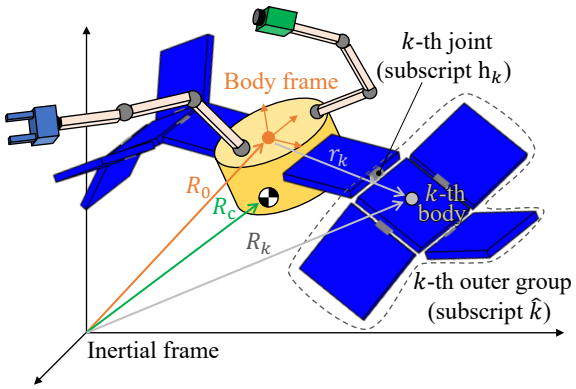

Fig. 1 Nomenclature of body components and their positions

or simply,

$$Mw + d = \tau \tag{2}$$

The corresponding generalized velocities are the translational velocity of the CoM of the entire spacecraft v_v, the angular velocity of the main body v_ω, and the joint actuation speed v_θ, which are all expressed in the body frame:

$$v_v = v_\mathrm{c} = \begin{bmatrix} v_{\mathrm{c},x} & v_{\mathrm{c},y} & v_{\mathrm{c},z} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^3 \qquad v_\omega = \omega_0 = \begin{bmatrix} \omega_{0,x} & \omega_{0,y} & \omega_{0,z} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^3 \qquad v_\theta = \dot{\theta} = \begin{bmatrix} \dot{\theta}_1 & \cdots & \dot{\theta}_m \end{bmatrix}^\mathsf{T} \in \mathbb{R}^m \tag{3}$$

An explicit description of the components in Equation (1) is given in Appendix A. In particular, $M_{\omega\omega}$ and $M_{\omega\theta}$ are important in the following discussion because these values provide the relationship between joint actuation and attitude motion:

$$\begin{aligned} M_{\omega\omega} &= I_\mathrm{c} \\ M_{\omega\theta,j} &= \left(I_{\hat{\jmath}} - m_{\hat{\jmath}} r^\times_{\hat{\jmath}\mathrm{c}} r^\times_{\hat{\jmath}\mathrm{h}_j} \right)^\mathsf{T} \lambda_j \end{aligned} \tag{4}$$

where $M_{\omega\theta,j}$ indicates the j-th column of $M_{\omega\theta}$. With these expressions, total angular momentum h_c around the centroid of the entire spacecraft can be represented as follows:

$$h_\mathrm{c} = M_{\omega\omega} v_\omega + M_{\omega\theta} v_\theta \tag{5}$$

In the present paper, we focus on attitude motion in which the total angular momentum is always zero. Applying this assumption to Equation (5) yields the following kinematic relationship:

$$v_\omega = \omega_0 = -M_{\omega\omega}^{-1} M_{\omega\theta} v_\theta = -M_{\omega\omega}^{-1} M_{\omega\theta} \dot{\theta} \tag{6}$$

Now, we use a rotational matrix (i.e., direction cosine matrix) C to express the attitude of the main body. This is because the rotational matrix is the most natural expression of the Lie group structure of the three-dimensional rotation, which helps to analytically solve the differential equation. Substituting Equation (6) into the equation of a rotational matrix, we obtain

$$\dot{C} = -\omega_0^{\times} C = \left(M_{\omega\omega}^{-1} M_{\omega\theta} \dot{\theta} \right)^{\times} C = (g(\theta)u)^{\times} C \tag{7}$$

where $g(\theta) = M_{\omega\omega}^{-1} M_{\omega\theta} \in \mathbb{R}^{3\times m}$, and control input $u = \dot{\theta} \in \mathbb{R}^m$. Similar kinematics formulation is found in coordinate-free motion planning method such as [43]. In the present research, the joint actuation speed $\dot{\theta}$ is regarded as control input, rather than joint actuation torque. This is because the angular momentum conservation law is expressed in integral form, and, therefore, the kinematics level equation can appropriately grasp the characteristics of the dynamics when there is no external force or torque. This equation expresses the relationship between attitude change and hinge joint actuation and has two important features:

1) Equation (7) is a nonholonomic constraint (or non-integrable differential constraint). Although the system is underactuated because the attitude (three DoF) + body configuration (m DoF) is controlled by m-dimensional control input $u = \dot{\theta} \in \mathbb{R}^m$, the nonholonomy practically provides extra degrees of freedom in control space and therefore enables the underactuated system to be controllable (locally, at least).
2) The function $g(\theta)$ is independent of attitude C, and therefore, in this equation, we can independently handle kinematic nonholonomy (induced by the non-integrability of a three-dimensional angular velocity vector) and momentum nonholonomy (induced by the change in the moment of inertia).

In the next section, Equation (7) is further investigated in order to obtain an approximate analytical attitude solution.

III. Derivation of Approximate Analytical Attitude Solution

The following derivations are based on a previous study [11]. The fundamental equation given by Equation (7) has a form similar to that of a linear autonomous differential equation $\dot{x} = Ax$ with respect to C. If matrix A in $\dot{x} = Ax$ is constant, then the equation can be solved analytically with $x = e^{At} x_0$. However, if matrix A is a general time-variant function, this matrix can be solved by Magnus expansion in the form of an infinite series [44]. We obtain a general solution of Equation (7) by the Magnus expansion in the following form:

$$\begin{aligned} C &= \exp\left(\sum_{k=1}^{\infty} \Omega_k(t) \right) C_0 \\ \text{where} \quad \Omega_1(t) &= \int_0^t \left(g(t)\dot{\theta} \right)^{\times} \mathrm{d}t \\ \Omega_2(t) &= \frac{1}{2} \int_0^t \mathrm{d}t_1 \int_0^{t_1} \mathrm{d}t_2 \left[\left(g(t_1)\dot{\theta} \right)^{\times}, \left(g(t_2)\dot{\theta} \right)^{\times} \right] \\ &\vdots \end{aligned} \tag{8}$$

where $[X,Y] = XY - YX$ is a commutator or a Lie bracket operator, and $\exp(\cdot)$ is a matrix exponential. The higher-order Ω_k consists of higher-order Lie brackets. If the components in the Lie bracket are always commutative for any combination of t, then the Lie bracket term becomes zero. Therefore, the term Ω_k $(k \geq 2)$ indicates non-commutativity of the matrix $\left(g(t)\dot{\theta}\right)^{\times}$.

This solution is the most general form but is not suitable for practical use due to integrals in Ω_k. In order to derive a more practical solution, we first assume rectilinear actuation in joint angle space. The rectilinear actuation is a joint actuation law in which joint motion is expressed as a straight path in joint angle space. The simple rectilinear trajectory in joint angle space contributes to simple joint actuation input. Each point in the m-dimensional joint angle space corresponds to one joint angle vector $\theta \in \mathbb{R}^m$. The joint angle history θ from the initial body configuration θ_0 to θ_1 is expressed as follows:

$$\begin{aligned} \theta(s) &= \theta_0 + s\Delta\theta \quad (0 \leq s \leq 1) \\ \Delta\theta &= \theta_1 - \theta_0 \end{aligned} \tag{9}$$

Owing to this equation, $\dot{\theta}\mathrm{d}t = \frac{\mathrm{d}\theta}{\mathrm{d}s}\mathrm{d}s = \Delta\theta\mathrm{d}s$ can replace the time variable t with the arc length parameter s. In addition, the function $g(\theta) = g(s)$ is approximated by a polynomial function with a certain interpolation method, such as Newton interpolation and Hermite interpolation. In the present study, we use the Newton interpolation because it does not require derivative terms and hence is computationally inexpensive. As a general form of the Newton interpolation, the $g(s)\Delta\theta$ can be expressed as follows:

$$g(s)\Delta\theta = \sigma_0(s)^{\mathsf{T}} A_0 \gamma_0 \tag{10}$$

where $\sigma_0(s)$ is a vector that contains s^{i-1} in the i-th component, A_0 is a constant matrix determined by coefficients of Newton interpolation, and γ_0 consists of values of interpolation points $g(s_j)\Delta\theta$ $(j = 0, 1, \cdots, n)$, where n is the order of the polynomial. In the case of second-order Newton interpolation, for example, these terms are given as follows:

$$\begin{aligned} \sigma_0(s) &= \begin{bmatrix} 1 & s & s^2 \end{bmatrix}^{\mathsf{T}} \\ A_0 &= \begin{bmatrix} 1 & -s_0 & s_0 s_1 \\ 0 & 1 & -(s_0 + s_1) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & s_1 - s_0 & 0 \\ 1 & s_2 - s_0 & (s_2 - s_0)(s_2 - s_1) \end{bmatrix}^{-1} \\ \gamma_0 &= \begin{bmatrix} g_0\Delta\theta & g_1\Delta\theta & g_2\Delta\theta \end{bmatrix}^{\mathsf{T}} \end{aligned} \tag{11}$$

where g_j is defined as $g_j = g(s_j)$ $(j = 0, 1, 2)$. The interpolation interval s_j can be arbitrarily sampled, and we use Chebyshev nodes as the most promising solution. The Chebyshev nodes prevent the Runge phenomenon, in which the

approximation error diverges at the end of the interval. Here, s_j is determined by the following equation:

$$s_j = \cos\left(\frac{2j+1}{2(n+1)}\pi\right) \qquad (j = 0, 1, 2, \cdots, n). \tag{12}$$

Throughout the above operations, all terms in the integrals in Equation (8) can be expressed with polynomials of s, which can be analytically integrable. Finally, we can obtain the following tensor form solution:

$$\begin{aligned} C &= \exp\left(\sum_{k=1}^{\infty} \Omega_k(s)\right) C_0 \\ &\text{where} \\ \Omega_1(s) &= \sigma_1(s)_i A_1{}^i_j \gamma_1{}^j \\ \Omega_2(s) &= \sigma_2(s)_{ik} A_2{}^{ik}_{jl} \gamma_2{}^{jl} \\ &\vdots \end{aligned} \tag{13}$$

This equation uses Einstein's summation convention, where contraction is carried out between a pair of subscripts/superscripts. The σ_k is a k-th-order tensor containing s polynomials, and A_k is a $k \times k$-th-order constant tensor determined by the applied polynomial interpolation method. In addition, γ_k is a k-th order vectrix containing functions of $g(s_j)\Delta\theta$ $(j = 1, 2, \cdots, n)$, which indicates that the elements in the vectrix are treated as if they were scalars throughout this contraction. Generally, Ω_k is expressed by contraction of the k-th-order tensor σ_k, γ_k, and the $k \times k$-th-order tensor A_k. If $g(s)\Delta\theta$ is interpolated as Equation (11), then the specific expressions of each tensor in Ω_1 and Ω_2 are as follows:

$$\begin{aligned} \sigma_1(s) &= \left(\int_0^s \sigma_0(s)\mathrm{d}s\right) = \begin{bmatrix} s & s^2/2 & s^3/3 \end{bmatrix}^{\mathsf{T}} \\ A_1 &= A_0 \\ \gamma_1 &= \begin{bmatrix} (g_0\Delta\theta)^{\times} & (g_1\Delta\theta)^{\times} & (g_2\Delta\theta)^{\times} \end{bmatrix}^{\mathsf{T}} \end{aligned} \tag{14}$$

$$
\begin{aligned}
\sigma_2(s) &= \left(\int_0^s \sigma_1(s)\sigma_0(s)^\mathsf{T} \mathrm{d}s \right) = \begin{bmatrix} s^2/2 & s^3/3 & s^4/4 \\ s^3/6 & s^4/8 & s^5/10 \\ s^4/12 & s^5/15 & s^6/18 \end{bmatrix} \\
A_2{}^{ik}_{jl} &= \frac{1}{2} A_0{}^i_j A_0{}^k_l \\
\gamma_2 &= \begin{bmatrix} 0 & (g_0\Delta\theta) \wedge (g_1\Delta\theta) & (g_0\Delta\theta) \wedge (g_2\Delta\theta) \\ (g_1\Delta\theta) \wedge (g_0\Delta\theta) & 0 & (g_1\Delta\theta) \wedge (g_2\Delta\theta) \\ (g_2\Delta\theta) \wedge (g_0\Delta\theta) & (g_2\Delta\theta) \wedge (g_1\Delta\theta) & 0 \end{bmatrix}
\end{aligned} \tag{15}
$$

where the wedge product $\wedge$ is defined by $x \wedge y = [y^\times, x^\times]$ for $x, y \in \mathbb{R}^3$. In order to express its final attitude after the rectilinear actuation, we define a state transition matrix as $C_{\text{final}} = \Phi C_0$, where Φ can be obtained by substituting $s = 1$ into the above equation as $\Phi = \exp(\Omega(s))|_{s=1}$. Throughout the above equations, the edge nodes in joint angle space θ_0 and θ_1 are given as fixed points. However, the equation $C_{\text{final}} = \Phi C_0$ no longer depends on s and can be interpreted as a function of θ_0 and θ_1 as follows:

$$
C(\theta_1, \theta_0) = \exp\left(\Omega(\theta_1, \theta_0)\right) C_0 = \Phi(\theta_1, \theta_0) C_0 \tag{16}
$$

This equation is further investigated in the following section in order to derive the proposed parallelogram actuation.

IV. Parallelogram Actuation

Nonholonomy appears as an accumulation of nonlinear effects along a trajectory in state space. In particular, this effect is distinctly confirmed by a circumvented path in state space. For example, a vehicle cannot directly move in the traversal direction, but going back and forth with proper handling can induce displacement in the traversal direction due to nonholonomy. In the case of a transformable free-flying robot, such a control corresponds to designing a proper circumvented path in joint angle space that achieves the target final attitude. As one of the most basic circumvented paths, a parallelogram path is used in the present study. Since a parallelogram path is closed, the body configuration returns to its initial state after four joint actuation strokes, but the final attitude becomes different due to the nonholonomic effect. Since the path is composed of two sets of parallel line segments, the free-flying robot executes two joint actuation patterns alternately, with the latter half reversing sign. We refer to such a maneuver as a parallelogram actuation. A similar actuation scheme has been used in a previous study by Ohashi et al. [12].

An example of the parallelogram actuation of a five-panel free-flying robot is shown in Fig. 2. In this example,

the joint actuation pattern A is set to $\Delta\theta_{\mathrm{A}} = [-90, -90, 0, 0]$ (deg) and the joint actuation pattern B is set to $\Delta\theta_{\mathrm{B}} = [+45, +45, +90, +90]$ (deg). Actuating with the pattern +A, +B, -A, and -B is equivalent to drawing a parallelogram in four-dimensional joint angle space. When the joint is actuated in this pattern, the final morphology returns to its original state, but we can confirm that the final attitude is different from the initial attitude.

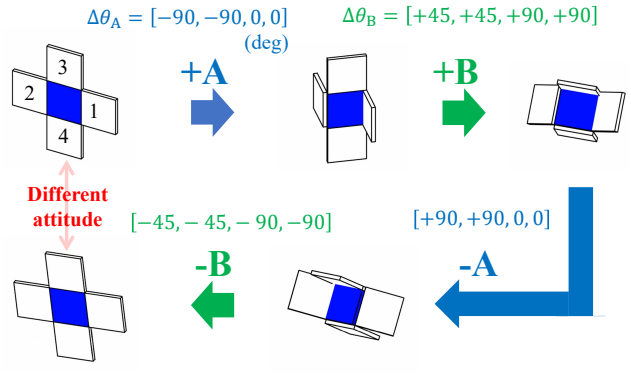

Fig. 2 Example of a parallelogram actuation

A. Analytical Attitude Solution of Parallelogram Actuation

Since the parallelogram actuation is represented as four consecutive rectilinear paths in joint angle space, the analytical solution presented in the previous section can be used in a piecewise manner. If the state transition matrix of the i-th stroke is represented as $\Phi_i(\theta_i, \theta_{i-1})$, then the final attitude after a parallelogram actuation can be expressed as consecutive matrix multiplications of the state transition matrices:

$$C_{\mathrm{final}} = \Phi_4(\theta_4, \theta_3)\Phi_3(\theta_3, \theta_2)\Phi_2(\theta_2, \theta_1)\Phi_1(\theta_1, \theta_0)C_0 \tag{17}$$

where $\theta_i \in \mathbb{R}^m$ is the joint angle state after the i-th stroke. From the definition of the parallelogram actuation, we can derive the following relationships for θ_0 through θ_4:

$$\begin{aligned}
\theta_1 &= \theta_0 + \Delta\theta_{\mathrm{A}} \\
\theta_2 &= \theta_1 + \Delta\theta_{\mathrm{B}} = \theta_0 + \Delta\theta_{\mathrm{A}} + \Delta\theta_{\mathrm{B}} \\
\theta_3 &= \theta_2 - \Delta\theta_{\mathrm{A}} = \theta_0 + \Delta\theta_{\mathrm{B}} \\
\theta_4 &= \theta_3 - \Delta\theta_{\mathrm{B}} = \theta_0
\end{aligned} \tag{18}$$

The final attitude C_{final} is dependent on a trajectory in joint angle space due to the nonholonomic property. However, the assumption of rectilinear actuation reduces this problem to the problem of allocating the waypoints θ_i in joint angle space. In the next section, we discuss how to modify these waypoints to achieve the target attitude.

B. Path Modification Method in Joint Space

Equation (17) is a forward solution of obtaining the attitude change given a transformation process of a space robot. On the other hand, in order to solve the inverse problem of obtaining a joint actuation process for a given target state, we need to modify the path in the joint angle space so that the final attitude C_{final} approaches the target attitude. In order to extract information for such path modification, we take a Jacobian of Equation (17) with respect to $\Delta\theta_{\text{A}}$ and $\Delta\theta_{\text{B}}$. Note that "Jacobian" here means the multi-dimensional matrix obtained by taking a gradient for each component in C_{final}. Thus, they constitute a third-order tensor $\frac{\partial C_{\text{final}}}{\partial \Delta\theta} \in \mathbb{R}^{3\times3\times m}$. A schematic diagram of modifying $\Delta\theta_{\text{A}}$ is shown in the Fig. 3. When modifying $\Delta\theta_{\text{A}}$, waypoints θ_1 and θ_2 are shifted by the same amount. Similarly, θ_2 and θ_3 are shifted by the same amount when modifying $\Delta\theta_{\text{B}}$. Thus, the Jacobian of Equation (17) can be described as follows:

$$
\begin{aligned}
\frac{\partial C_{\text{final}}}{\partial \Delta\theta_{\text{A}}} &= \frac{\partial C_{\text{final}}}{\partial \theta_1} + \frac{\partial C_{\text{final}}}{\partial \theta_2} \\
&= \Phi_4\Phi_3\left(\frac{\partial \Phi_2}{\partial \theta_1}\Phi_1 + \Phi_2\frac{\partial \Phi_1}{\partial \theta_1}\right)C_0 + \Phi_4\left(\frac{\partial \Phi_3}{\partial \theta_2}\Phi_2 + \Phi_3\frac{\partial \Phi_2}{\partial \theta_2}\right)\Phi_1 C_0 \\
\frac{\partial C_{\text{final}}}{\partial \Delta\theta_{\text{B}}} &= \frac{\partial C_{\text{final}}}{\partial \theta_2} + \frac{\partial C_{\text{final}}}{\partial \theta_3} \\
&= \Phi_4\left(\frac{\partial \Phi_3}{\partial \theta_2}\Phi_2 + \Phi_3\frac{\partial \Phi_2}{\partial \theta_2}\right)\Phi_1 C_0 + \left(\frac{\partial \Phi_4}{\partial \theta_3}\Phi_3 + \Phi_4\frac{\partial \Phi_3}{\partial \theta_3}\right)\Phi_2\Phi_1 C_0
\end{aligned}
\tag{19}
$$

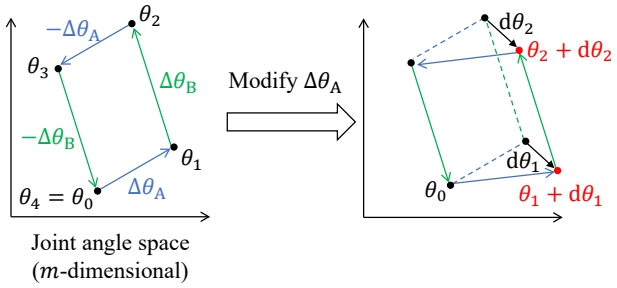

Fig. 3 Schematic diagram of modifying $\Delta\theta_{\text{A}}$ in joint angle space

The Jacobian of the state transition matrix $\frac{\partial \Phi}{\partial \theta}$ in Equation (19) can be calculated as follows by applying a derivative formula of an exponential matrix: [45]:

$$
\frac{\partial \Phi}{\partial \theta} = \frac{\partial \exp\left(\Omega(\theta)\right)}{\partial \theta} = \int_0^1 e^{\alpha\Omega(\theta)}\frac{\partial \Omega(\theta)}{\partial \theta}e^{(1-\alpha)\Omega(\theta)}\mathrm{d}\alpha \tag{20}
$$

where $\frac{\partial \Omega(\theta)}{\partial \theta}$ denotes $\frac{\partial \Omega(\theta_i,\theta_{i-1})}{\partial \theta_i}$ or $\frac{\partial \Omega(\theta_i,\theta_{i-1})}{\partial \theta_{i-1}}$ and is described as follows:

$$
\frac{\partial \Omega(\theta)}{\partial \theta} = \bar{\sigma}_{1i}A_1{}^i_j\frac{\partial \gamma_1(\theta)^j}{\partial \theta} + \bar{\sigma}_{2ik}A_2{}^{ik}_{jl}\frac{\partial \gamma_2(\theta)^{jl}}{\partial \theta} + \cdots \tag{21}
$$

in which $\bar{\sigma} = \sigma(s)|_{s=1}$ and Einstein's summation convention is applied for pairs of same indices. The expressions of $\frac{\partial \gamma_k}{\partial \theta}$ depend on its interpolation method. For example, in the case of the Newton interpolation, the j-th components of $\frac{\partial \gamma_1}{\partial \theta_i}$ and $\frac{\partial \gamma_1}{\partial \theta_{i-1}}$ are expressed as follows:

$$\begin{aligned} \frac{\partial {\gamma_1}^j}{\partial \theta_i} &= \left(s_j \frac{\partial g_j}{\partial \theta} \Delta\theta + g_j \right)^{\times} \\ \frac{\partial {\gamma_1}^j}{\partial \theta_{i-1}} &= \left((1 - s_j) \frac{\partial g_j}{\partial \theta} \Delta\theta - g_j \right)^{\times} \end{aligned} \tag{22}$$

where $g_j = g(s_j)$, and $\frac{\partial g_j}{\partial \theta} = \frac{\partial g}{\partial \theta}(s_j)$ $(j = 0, 1, \cdots, n)$. The detailed expression of $\frac{\partial g}{\partial \theta}$ is provided in Appendix B. Throughout these mathematical operations, all the components in Equation (19) can be explicitly computed.

Equation (19) expresses the difference of C_{final} when $\Delta\theta_{\text{A}}$ and $\Delta\theta_{\text{B}}$ are varied. Therefore, a path in joint angle space can be modified by leveraging this equation. We use the Newton-Raphson method to modify the parallelogram path in the present study. First, $\frac{\partial C_{\text{final}}}{\partial \Delta\theta}$ is converted to $\frac{\partial e_{\text{final}}}{\partial \Delta\theta}$, where e_{final} is the corresponding final Euler angle. This is because $\frac{\partial C_{\text{final}}}{\partial \Delta\theta}$ exhibits singularity due to its constraints of orthonormality. The conversion equation is

$$\frac{\partial e_{\text{final}}}{\partial \Delta\theta} = E(e_{\text{final}}) \left(-\frac{\partial C_{\text{final}}}{\partial \Delta\theta} C_{\text{final}}^{\mathsf{T}} \right)^{\vee} \tag{23}$$

where $E(e_{\text{final}})$ is a matrix that converts angular velocity ω to Euler angle velocity $\dot{e}$. Finally, the path in joint angle space can be modified by the Newton-Raphson method using Equation (23) as follows:

$$\Delta\theta^{(k+1)} = \Delta\theta^{(k)} + \alpha_k \left(\frac{\partial e_{\text{final}}}{\partial \Delta\theta} \right)^{\dagger} \left(e^* - e_{\text{final}}^{(k)} \right) \tag{24}$$

where $(\cdot)^{\dagger}$ is the pseudo-inverse, $(\cdot)^{(k)}$ denotes the value in the k-th iteration, and e^* is the target Euler angle. In addition, α_k is a modification coefficient in the k-th iteration determined so that the modification value in one iteration step is kept sufficiently small. In the present study, we set α_k as

$$\alpha_k = \begin{cases} 1/\beta & (\text{if} \quad \beta > 1) \\ 1 & (\text{if} \quad 0 \leq \beta \leq 1) \end{cases} \tag{25}$$

where $\beta = \left| \left(\frac{\partial e_{\text{final}}}{\partial \Delta\theta} \right)^{\dagger} \left(e^* - e_{\text{final}}^{(k)} \right) \right|$. Here, $\Delta\theta_{\text{A}}$ and $\Delta\theta_{\text{B}}$ are repeatedly modified using Equation (24) until the final attitude e_{final} approaches the target attitude e^* within a certain tolerance.

The kinematic relationship (24) involves pseudo-inverse operation of the Jacobian. This operation can cause singularity when rank of the Jacobian is degenerated. Similar singularity problem is one of the main issues of robotic manipulators and a lot of studies are focused on it [46, 47]. When the path modification law encounters a singularity,

such singularity avoidance method developed for robotic manipulators can also be applied to this path modification law. This singularity problem is in the scope of our future studies.

V. Numerical Simulation

A. Model Definitions

Using the path modification method described in the previous section, we specifically design a parallelogram actuation to achieve an arbitrary target attitude. As mentioned in Section I, the proposed method is especially useful for robots with many degrees of freedom of joints. Here we use the 8-DoF free-flying robot shown in Fig. 4 a). This figure shows the fully deployed body configuration, in which all joint angles are set to zero. The local coordinates fixed on each body are defined such that they correspond to the x-y-z coordinates in this fully deployed body configuration. The mass and dimensions of each body are according to Table 1. All bodies are defined as cuboids and the corresponding side lengths are L_x, L_y, and L_z. The definitions of $p_{\underline{k}}$, p_k, and λ_k in Table 1 are shown in Fig. 4b). The subscript $\underline{k}$ indicates the inside neighbor of the k-th body. The values of $p_{\underline{k}}$ in Table 1 is expressed in the local body-fixed coordinates of the $\underline{k}$-th body, and p_k and λ_k are expressed in the local body-fixed coordinates of the k-th body. The angle limit of each joint is $-\pi/2 \leq \theta^k \leq \pi/2$. The maximum joint speed is set to be 10 deg/s, and the joint actuation speed in each stroke is normalized such that the fastest joint is actuated at 10 deg/s.

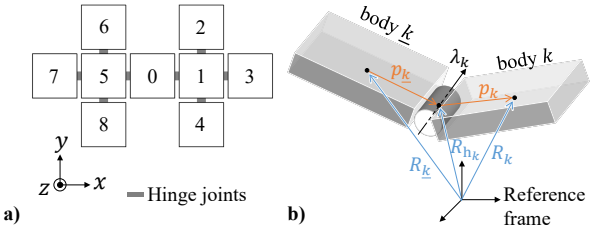

Fig. 4 Model definitions: a) Body connection map of the 9-panel free-flying robot, b) Illustration of $p_{\underline{k}}$, p_k, and λ_k

B. Initial Guess Search

In order to start the Newton-Raphson iteration described in Section IV.B, a proper initial guess of parallelogram actuation must be provided, although it is generally a difficult task. In the present study, we construct a motion primitive database beforehand, and one promising maneuver is selected from the database as an initial guess. We constructed the database of the parallelogram actuations such that only one joint is actuated in each stroke of parallelogram actuations. Joint actuation angles of the database are sampled with 15-degree interval between maximum and minimum joint angle limits: $[-90, -75, -60, -45, -30, -15, 15, 30, 45, 60, 75, 90]$ [deg]. Provided the total number of actuatable joints is m and the number of joint actuation pattern is n_a, the total number of datasets adds up to ${}_m\mathrm{C}_2 \times n_a^2 \times 2$, which becomes

Table 1 Mass and dimension properties of the 9-panel free-flying robot

Body #	Mass [kg]	$L_x \times L_y \times L_z$ [mm]	$p_{\underline{k}}$ [mm]	p_k [mm]	λ_k
0	10.0	$1000 \times 1000 \times 100$	-	-	-
1	10.0	$1000 \times 1000 \times 100$	[550, 0, 0]	[550, 0, 0]	[0, 1, 0]
2	10.0	$1000 \times 1000 \times 100$	[0, 550, 0]	[0, 550, 0]	[1, 0, 0]
3	10.0	$1000 \times 1000 \times 100$	[550, 0, 0]	[550, 0, 0]	[0, 1, 0]
4	10.0	$1000 \times 1000 \times 100$	[0, -550, 0]	[0, -550, 0]	[1, 0, 0]
5	10.0	$1000 \times 1000 \times 100$	[-550, 0, 0]	[-550, 0, 0]	[0, 1, 0]
6	10.0	$1000 \times 1000 \times 100$	[0, 550, 0]	[0, 550, 0]	[1, 0, 0]
7	10.0	$1000 \times 1000 \times 100$	[-550, 0, 0]	[-550, 0, 0]	[0, 1, 0]
8	10.0	$1000 \times 1000 \times 100$	[0, -550, 0]	[0, -550, 0]	[1, 0, 0]

very large for robots with a large number of joints. However, these attitude calculations are not difficult because the entire spacecraft can be divided into only two rigid bodies when only one joint is actuated while the other joints are rigidly fixed. Therefore, we can construct the motion primitive database with less computational effort.

Figure 5 shows one example of the motion primitive database for the fully deployed body configuration. (Note that the database distribution is dependent on the initial body configuration, and therefore the database must be reconstructed for different body configurations.) The attitude change of each parallelogram actuation is expressed by a dot in the y-x-z Euler angle space supposing initial attitude is $[e_1, e_2, e_3] = [0, 0, 0]$ (the rotation angles around each axis are e_1, e_2, and e_3, respectively, in this order). In addition, each point is colored by its rotational efficiency, which is defined as follows: (total rotational angle of attitude)/(total joint actuation angle). This model has 8 joints in total, and therefore the total number of points is ${}_8\mathrm{C}_2 \times 12^2 \times 2 = 8,064$. From this database, the point that is the closest to the target attitude in Euler angle space is selected as a promising initial guess.

In the proposed path modification law, constraints on joint angles are not included, and therefore the resulting path in joint angle space could violate $\theta_{\max}$ and $\theta_{\min}$ limits. In particular, actuated joints in the chosen initial guess are the most likely to violate the constraints because they are already set to be $\theta_{\max}$ or $\theta_{\min}$ at the start of the iteration. For this reason, we omitted the corresponding columns of $\frac{\partial e_{\text{final}}}{\partial \Delta\theta}$ and rows of $\Delta\theta^{(k)}$ in Equation (24) in order that angles of these joints were not updated.

C. Simulation Conditions and Result

We carried out a numerical simulation to validate the proposed path modification method to achieve target attitude with parallelogram actuation. The initial body configuration is set to be the fully deployed state (in which all joint angles are zero), as shown in the Fig. 4 a), and the corresponding motion primitive database is as shown in Fig. 5. The initial and target attitudes were set to y-x-z Euler angles of $[0, 0, 0]$ deg and $[7.5, -7.5, 7.5]$ deg, respectively. Under these conditions, the chosen initial guess is the case of $[{\Delta\theta_{\mathrm{A}}}^8, {\Delta\theta_{\mathrm{B}}}^5] = [90, -90]$ deg. Throughout path modification,

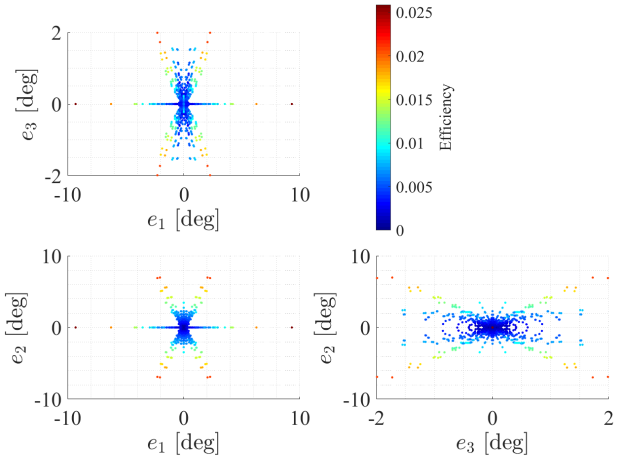

Fig. 5 Motion primitive database in y-x-z Euler angle space and its rotational efficiency (three-view display)

$\Delta\theta_A{}^8$ and $\Delta\theta_B{}^5$ are not updated, as explained at the end of the previous section. The infinite series of Ω in Equation (13) is truncated at $k = 2$, and the number of interpolation points in Chebyshev nodes is set to be eight by trading off computational cost and approximation accuracy. Newton-Raphson iteration is finished when the norm of the Euler angle difference with respect to the target Euler angle is less than 10^{-3} radians.

Figure 6 shows the iteration process in Euler angle space. In this example, the solver exits the iterations after three loops. The figure shows that the final attitude of the parallelogram actuation gradually approaches the target point as the iteration proceeds. Table 2 shows the values of $\Delta\theta_A$ and $\Delta\theta_B$. Figure 7 is an illustration of the body reconfiguration procedure. This figure clearly shows that the spacecraft successfully achieves the target attitude after the designed parallelogram actuation and that its body configuration returned to the original state in the end. The Euler angle of the final attitude of the designed parallelogram actuation was $[7.509, -7.507, 7.503]$ [deg].

This numerical simulation validated that the proposed path modification method can effectively design the parallelogram actuation to achieve the target attitude. This maneuver guarantees that the space robot maintains the initial body configuration after one set of maneuvers. Therefore, the robot can independently achieve the target attitude while maintaining its initial body configuration.

Table 2 Components in $\Delta\theta_A$ and $\Delta\theta_B$ [deg]

Joint 1–12			Joint 13–24		
i	$\Delta\theta_A{}^i$	$\Delta\theta_B{}^i$	i	$\Delta\theta_A{}^i$	$\Delta\theta_B{}^i$
1	-6.60	81.04	5	0.00	-90.00
2	-12.31	-21.87	6	0.32	18.96
3	0.07	17.72	7	2.07	10.11
4	-19.39	-2.18	8	90.00	0.00

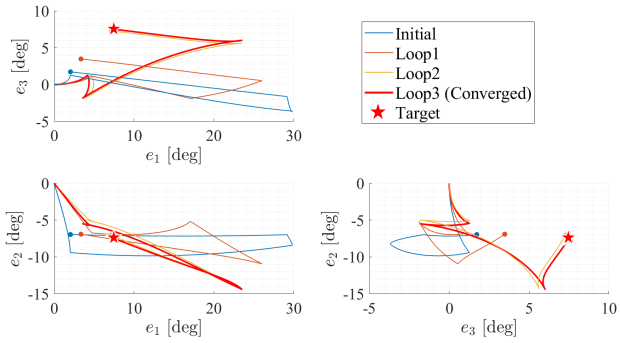

Fig. 6 Path modification process in Euler angle space (three-view display)

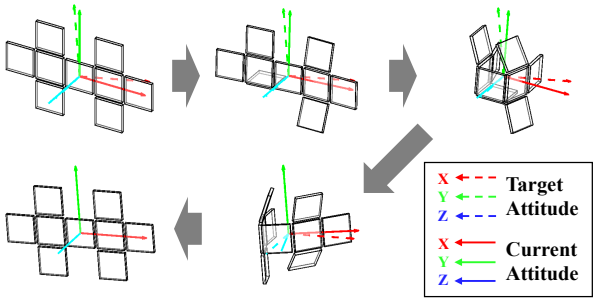

Fig. 7 Illustration of body reconfiguration procedure in the designed parallelogram actuation

VI. Discussions

A. Main Findings

The present research constructed a general analytical solution and approximate analytical solution for the rectilinear actuation, which is not usually adopted in the context of space manipulators. This solution is described in a widely applicable form and hence is a promising analytical tool for control of various space robots. In addition, the analytical solution can fully use large degrees of freedom of joints with much less computational effort than brute-force numerical search, such as Ohashi's motion primitive method [12]. Although the proposed method is also required to construct a database for the initial guess, it is computationally inexpensive if only one joint is actuated in each stroke.

In particular, the $\frac{\partial C_{\text{final}}}{\partial \theta_k}$ in Equation (19) can be an important indicator. This is because the $\frac{\partial C_{\text{final}}}{\partial \theta_k}$ provides information about the sensitivity of the final attitude with respect to the difference of the intermediate point θ_k in joint angle space. In other words, the $\frac{\partial C_{\text{final}}}{\partial \theta_k}$ can provide how the path should be circumvented in joint angle space, which is strongly related to the essence of nonholonomic motion planning. Therefore, this analytical formulation is also expected to be a powerful tool for other motion planning methods using free-flying robots.

B. Limitations

We formulated the equations for the kinematics level in the present study. This formulation is suitable to capture the characteristics of a nonholonomic system because nonholonomy is characterized by a dynamic constraint on velocity, rather than on acceleration. However, it does not directly provide a torque-based control law, and thus the designed path in joint angle space cannot be exactly traced in most cases. In addition, the control law in the proposed method is provided in feedforward form and does not compensate for the deviation from the designed trajectory. Further studies should provide control laws that can provide torque-based input and also compensate for control errors.

The proposed method cannot generate a path that satisfies joint angle constraints. In the proposed method, the constraint violation is avoided by not updating $\Delta\theta$ for joints close to their angle limits. However, other types of constraints, such as self-collision avoidance, are difficult to handle using the proposed method. One promising solution to overcome this problem is a numerical optimization, such as interior point methods, which can systematically handle various types of constraints [48]. For the initial guess of such an optimization method, the trajectory generated by the proposed method can be applicable.

The last important limitation of the proposed method is the difficulty of preparing the initial guesses. The successful convergence of the Newton-Raphson method depends on the quality of the initial guesses. Although the database construction can be completed relatively quickly, not all combinations can be explored for large-degree-of-freedom joints, which limits the possibility of providing a good initial guess. One possible solution is to develop a learning-based motion planner to provide an initial guesses. Some research focused on reinforcement learning in order to generate a nonholonomic reorientation procedure with a free-flying space robot [49, 50]. The problem of the learning based method is a huge learning time for large degrees of freedom, and therefore we can expect a low-resolution path by restricting the search space to explore.

VII. Conclusion

In the present paper, we developed parallelogram actuation, which achieves the target attitude with an analytical solution for the rectilinear actuation. The final part demonstrated a desired attitude reorientation using the developed parallelogram actuation. The proposed attitude reorientation does not consume any propellant and does not accumulate the momentum of wheels, which contributes to the longer life-span of the robot. The developed analytical solution is a widely applicable and powerful tool for motion planning for free-flying space robots. In particular, the proposed method can fully make use of the reorientation ability of robots with large degrees of freedom of joints without causing numerical difficulties. Such robots with large degrees of freedom of joints are promising future space robots that can be adaptive, dexterous, and economical, and the present study contributes to greatly enhancing the ability of such future space robots.

Appendix A: Components in a Generalized Equation of Motion

The explicit expressions in Equation (1) are provided as follows. These equations can be derived through Kane's method [6].

$$
\begin{aligned}
&M_{vv} = mU \\
&M_{\omega\omega} = I_{\mathrm{c}} \\
&M_{\omega\theta,j} = \left(I_{\hat{j}} - m_{\hat{j}} r^{\times}_{\hat{j}\mathrm{c}} r^{\times}_{\hat{j}\mathrm{h}_j}\right)^{\mathsf{T}} \lambda_j \\
&M_{\theta\omega,i} = \lambda_i^{\mathsf{T}} \left(I_{\hat{i}} - m_{\hat{i}} r^{\times}_{\hat{i}\mathrm{c}} r^{\times}_{\hat{i}\mathrm{h}_i}\right) \\
&M_{\theta\theta,ij} = M^{*}_{\theta\theta,ij} + \frac{m_{\hat{i}} m_{\hat{j}}}{m} \lambda_i^{\mathsf{T}} \left(r^{\times}_{\hat{i}\mathrm{h}_i} r^{\times}_{\hat{j}\mathrm{h}_j}\right)^{\mathsf{T}} \lambda_j \\
&M^{*}_{\theta\theta,ij} = \begin{cases} \lambda_i^{\mathsf{T}} \left(I_{\hat{j}} - m_{\hat{j}} r^{\times}_{\hat{j}\mathrm{h}_i} r^{\times}_{\hat{j}\mathrm{h}_j}\right)^{\mathsf{T}} \lambda_j & (j \in \hat{i}) \\ \lambda_i^{\mathsf{T}} \left(I_{\hat{i}} + m_{\hat{i}} r^{\times}_{\hat{i}\mathrm{h}_i} r^{\times}_{\hat{i}\mathrm{h}_j}\right)^{\mathsf{T}} \lambda_j & (i \in \hat{j}) \\ 0 & (i \notin \hat{j} \ \cap \ j \notin \hat{i}) \end{cases} \\
&M_{v\omega} = M_{v\theta} = M_{\omega v} = M_{\theta v} = O \qquad \text{(Zero matrix)}
\end{aligned}
\tag{26}
$$

$$
\begin{aligned}
&d_v = m\omega_0^{\times} \dot{R}_{\mathrm{c}} \\
&d_{\omega} = \sum_k \left\{ (R_k - R_{\mathrm{c}})^{\times} f_k^{\ddagger} + n_k^{\ddagger} \right\} \\
&d_{\theta,i} = \lambda_i^{\mathsf{T}} \left[\sum_{k \in \hat{i}} \left\{ (R_k - R_{\mathrm{h}_i})^{\times} f_k^{\ddagger} + n_k^{\ddagger} \right\} - m_{\hat{i}} r^{\times}_{\hat{i}\mathrm{h}_i} \omega_0^{\times} \dot{R}_{\mathrm{c}} \right]
\end{aligned}
\tag{27}
$$

$$
\begin{aligned}
&\tau_v = f_{\mathrm{c}} \\
&\tau_{\omega} = \sum_k \left\{ (R_k - R_{\mathrm{c}})^{\times} f_k + n_k \right\} \\
&\tau_{\theta,i} = \lambda_i^{\mathsf{T}} \left[\sum_{k \in \hat{i}} \left\{ (R_k - R_{\mathrm{h}_i})^{\times} f_k + n_k \right\} - \sum_k \frac{m_{\hat{i}}}{m} \left(r^{\times}_{\hat{i}\mathrm{h}_i} f_k \right) \right] + n_{\theta i}
\end{aligned}
\tag{28}
$$

where $M_{\omega\theta,j}$ is the j-th column of $M_{\omega\theta}$, $M_{\theta\omega,i}$ is the i-th row of $M_{\theta\omega}$, $d_{\theta,i}$ is the i-th component of d_{θ}, and $\tau_{\theta,i}$ is the i-th component of τ_{θ}.

Appendix B: Analytical Expression of $\frac{\partial g}{\partial \theta}$

For the sake of brevity, in this section, $M_{\omega\omega}$ is replaced by M and $M_{\omega\theta}$ is replaced by N. From Equation (7), $g(\theta)$ is expressed as follows:

$$
g(\theta) = M^{-1} N \tag{29}
$$

In this equation, M and N depend only on the body configuration $\theta \in \mathbb{R}^m$, and therefore $\frac{\partial g}{\partial \theta} \in \mathbb{R}^{3\times m\times m}$ is obtained as follows:

$$\frac{\partial g_j^i}{\partial \theta^k} = \frac{\partial M^{-1}N}{\partial \theta} = -{M^{-1}}_p^i \frac{\partial M_n^p}{\partial \theta^k} {M^{-1}}_r^n N_j^r + {M^{-1}}_p^i \frac{\partial N_j^p}{\partial \theta^k} \tag{30}$$

where X_j^i is the (i,j) component of matrix X, and the subscript/superscript expresses contraction with respect to the dimension (Einstein's summation convention). Moreover, all three-dimensional matrices, $\frac{\partial M}{\partial \theta}$ and $\frac{\partial N}{\partial \theta}$, can be described explicitly as follows:

$$\begin{aligned}\frac{\partial M}{\partial \theta^k} = \sum_{l\notin\hat{k}} \frac{m_l m_{\hat{k}}}{m}\left\{r_{l\mathrm{c}}^\times, \left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times\right\} + \left[\lambda_k^\times, I_{\hat{k}}\right] \\ - m_{\hat{k}}\left(1 - \frac{m_{\hat{k}}}{m}\right)\left\{r_{\hat{k}\mathrm{c}}^\times, \left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times\right\}\end{aligned} \tag{31}$$

$$\frac{\partial N_j}{\partial \theta^k} \begin{cases} = \lambda_k^\times I_{\hat{j}} \lambda_j - m_{\hat{j}} r_{\hat{j}\mathrm{c}}^\times \lambda_k^\times r_{\hat{j}\mathrm{h}_j}^\times \lambda_j \\ \quad -m_{\hat{j}}\left(\lambda_k^\times r_{\hat{j}\mathrm{h}_k} - \frac{m_{\hat{k}}}{m}\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times r_{\hat{j}\mathrm{h}_j}^\times \lambda_j \quad \left(j \in \hat{k}\right) \\ = -\sum_{\substack{p\in\hat{j}\\ p\notin\hat{k}}} \frac{m_{\hat{k}} m_p}{m_{\hat{j}}}\left\{r_{\hat{j}p}^\times, \left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times\right\}\lambda_j \\ \quad -m_{\hat{k}}\left(\frac{m_{\hat{k}}}{m_{\hat{j}}} - 1\right)\left\{r_{\hat{j}\hat{k}}^\times, \left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times\right\}\lambda_j \\ \quad -m_{\hat{j}}\left(\frac{m_{\hat{k}}}{m_{\hat{j}}} - \frac{m_{\hat{k}}}{m}\right)\left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times r_{\hat{j}\mathrm{h}_j}^\times \lambda_j \\ \quad +\left[\lambda_k^\times, I_{\hat{k}}\right]\lambda_j - m_{\hat{k}} r_{\hat{j}\mathrm{c}}^\times \left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times \lambda_j \quad \left(k \in \hat{j}\right) \\ = \frac{m_{\hat{j}} m_{\hat{k}}}{m}\left(\lambda_k^\times r_{\hat{k}\mathrm{h}_k}\right)^\times r_{\hat{j}\mathrm{h}_j}^\times \lambda_j \qquad \left(j \notin \hat{k} \ \cap \ k \notin \hat{j}\right) \end{cases} \tag{32}$$

where $[X,Y] = XY - YX$ is a commutator, and $\{X,Y\} = XY + YX$ is an anticommutator. Here, N_j is the j-th column of matrix $N \in \mathbb{R}^{3\times m}$ and represents the generalized moment of inertia of the j-th outer group. Note that there are three branches depending on the geometrical relationship between the j-th body and the k-th body when calculating $\frac{\mathrm{d}N_j}{\mathrm{d}\theta}$.

Funding Sources

The present study was supported by a Grant-in-Aid for Scientific Research (JP19J22108) from the Japan Society for the Promotion of Science.

Acknowledgments

The authors are grateful to the members of the Transformer working group in JAXA, who greatly enhanced the value of the present study through thorough and patient discussions.

References

[1] Shoemaker, M. A., Vavrina, M., Gaylor, D. E., Mcintosh, R., Volle, M., and Jacobsohn, J., "OSAM-1 DECOMMISSIONING ORBIT DESIGN," *Proceedings of AAS/AIAA Astrodynamics Specialist Conference*, Lake Tahoe, USA, Aug. 2020. AAS 20-460.

[2] Hudson, J. S., and Kolosa, D., "Versatile on-orbit servicing mission design in geosynchronous earth orbit," *Journal of Spacecraft and Rockets*, Vol. 57, No. 4, 2020, pp. 844–850. https://doi.org/10.2514/1.A34701.

[3] Sugawara, Y., Chujo, T., Kubo, Y., Sato, Y., Otsuki, M., Ikeda, R., Ikeda, K., Fujita, M., Sawada, K., Tsumura, K., Matsuura, S., Kotani, T., Sugihara, A., Torisaka, A., Mori, O., Kawasaki, S., and Kawaguchi, J., "Transformable spacecraft: Feasibility study and conceptual design," *Proceedings of the International Astronautical Congress*, Virtual, Oct. 2020. IAC-20-D1.2.8.x59887.

[4] Kubo, Y., and Kawaguchi, J., "Propellant Free Station Keeping around Sun Earth L2 Using Solar Radiation Pressure for a Transformable Spacecraft," *Proceedings of International Symposium on Space Technology and Science*, Fukui, Japan, June 2019. 2019-d-116s.

[5] Kane, T. R., and Levinson, D. A., "The use of Kane's dynamical equations in robotics," *The International Journal of Robotics Research*, Vol. 2, No. 3, 1983, pp. 3–21. https://doi.org/10.1177/027836498300200301.

[6] Kane, T. R., and Levinson, D. A., *Dynamics, theory and applications*, McGraw Hill, 1985, Chap. 6.

[7] Papadopoulos, E. G., "Nonholonomic behavior in free-floating space manipulators and its utilization," *Nonholonomic Motion Planning*, Springer, 1993, pp. 423–445. https://doi.org/10.1007/978-1-4615-3176-0_11.

[8] Nakamura, Y., and Mukherjee, R., "Exploiting nonholonomic redundancy of free-flying space robots," *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 4, 1993, pp. 499–506. https://doi.org/10.1109/70.246062.

[9] Brockett, R. W., et al., "Asymptotic stability and feedback stabilization," *Differential geometric control theory*, Vol. 27, No. 1, 1983, pp. 181–191.

[10] Ratajczak, J., and Tchoń, K., "Normal forms and singularities of non-holonomic robotic systems: A study of free-floating space robots," *Systems & Control Letters*, Vol. 138, 2020, p. 104661. https://doi.org/10.1016/j.sysconle.2020.104661.

[11] Kubo, Y., "Simultaneous Body Reconfiguration and Nonholonomic Attitude Reorientation of Free-flying Space Robots," Ph.D. thesis, The University of Tokyo, 2022.

[12] Ohashi, K., Chujo, T., and Kawaguchi, J., "Optimal Motion Planning in Attitude Maneuver Using Non Holonomic Turns for a Transformable Spacecraft," *Proceedings of AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, USA, Aug. 2018. AAS-18-359.

[13] Chaplygin, S. A., "On the theory of motion of nonholonomic systems. The reducing-multiplier theorem," *Regular and Chaotic Dynamics*, Vol. 13, No. 4, 2008, pp. 369–376.

[14] Reeds, J., and Shepp, L., "Optimal paths for a car that goes both forwards and backwards," *Pacific journal of mathematics*, Vol. 145, No. 2, 1990, pp. 367–393.

[15] Khennouf, H., and De Wit, C. C., "On the construction of stabilizing discontinuous controllers for nonholonomic systems," *IFAC Proceedings Volumes*, Vol. 28, No. 14, 1995, pp. 667–672.

[16] Murray, R. M., and Sastry, S. S., "Nonholonomic motion planning: Steering using sinusoids," *IEEE transactions on Automatic Control*, Vol. 38, No. 5, 1993, pp. 700–716.

[17] Muller, H. R., and Weed, L. H., "Notes on the falling reflex of cats," *American Journal of Physiology-Legacy Content*, Vol. 40, No. 3, 1916, pp. 373–379. https://doi.org/10.1152/ajplegacy.1916.40.3.373.

[18] Kane, T., and Scher, M., "A dynamical explanation of the falling cat phenomenon," *International journal of solids and structures*, Vol. 5, No. 7, 1969, pp. 663–670. https://doi.org/10.1016/0020-7683(69)90086-9.

[19] Yamafuji, K., Kobayashi, T., and Kawamura, T., "Elucidation of twisting motion of a falling cat and its realization by a robot," *Journal of the Robotics Society of Japan*, Vol. 10, No. 5, 1992, pp. 648–654. https://doi.org/10.7210/jrsj.10.648.

[20] Kawaguchi, J., Mori, O., and Minamikawa, K., "Non-holonomic turn and its application to spacecraft attitude maneuvers," *Proceedings of 56th International Astronautical Congress*, Fukuoka, Japan, Oct. 2005. IAC-05-C1.2.08.

[21] Li, L., Zhao, J., and Xia, Y., "Landing posture adjustment and buffer performance analysis of a cat robot," *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, IEEE, 2018, pp. 357–363. https://doi.org/10.1109/IMCEC.2018.8469620.

[22] Garant, X., and Gosselin, C., "Design and Experimental Validation of Reorientation Manoeuvres for a Free Falling Robot Inspired From the Cat Righting Reflex," *IEEE Transactions on Robotics*, 2020. https://doi.org/10.1109/TRO.2020.3031241.

[23] Akin, D. L., Minsky, M., Thiel, E., and Kurtzman, C., *Space applications of automation, robotics and machine intelligence systems (ARAMIS) phase II*, 1983, NASA Contract Report, Vol. 3734, Chap. 2.

[24] Bronez, M., Clarke, M., and Quinn, A., "Requirements development for a free-flying robot–The "Robin"," *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, Vol. 3, IEEE, Feb. 1987, pp. 667–672. https://doi.org/10.1109/ROBOT.1986.1087628.

[25] Mitsushige, O., "Summary of NASDA's ETS-VII robot satellite mission," *Journal of Robotics and Mechanics*, Vol. 12, No. 4, 2000, pp. 417–424. https://doi.org/10.20965/jrm.2000.p0417.

[26] Menon, C., Aboudan, A., Cocuzza, S., Bulgarelli, A., and Angrilli, F., "Free-flying robot tested on parabolic flights: kinematic control," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 4, 2005, pp. 623–630. https://doi.org/10.2514/1.8498.

[27] Rybus, T., Seweryn, K., Oleś, J., Basmadji, F. L., Tarenko, K., Moczydłowski, R., Barciński, T., Kindracki, J., Mężyk, Ł., Paszkiewicz, P., et al., "Application of a planar air-bearing microgravity simulator for demonstration of operations required for an orbital capture with a manipulator," *Acta Astronautica*, Vol. 155, 2019, pp. 211–229. https://doi.org/10.1016/j.actaastro.2018.12.004.

[28] Junkins, J. L., and Schaub, H., "An instantaneous eigenstructure quasivelocity formulation for nonlinear multibody dynamics," *The Journal of the Astronautical Sciences*, Vol. 45, No. 3, 1997, pp. 279–295. https://doi.org/10.1007/BF03546405.

[29] Li, K., Zhang, Y., and Hu, Q., "Dynamic modelling and control of a tendon-actuated lightweight space manipulator," *Aerospace Science and Technology*, Vol. 84, 2019, pp. 1150–1163. https://doi.org/10.1016/j.ast.2018.11.018.

[30] Malczyk, P., Chadaj, K., and Fraczek, J., "Parallel Hamiltonian Formulation for Forward Dynamics of Free-Flying Manipulators," *Aerospace Robotics III*, Springer, 2019, pp. 1–15. https://doi.org/10.1007/978-3-319-94517-0_1.

[31] Torres, M. A., and Dubowsky, S., "Minimizing spacecraft attitude disturbances in space manipulator systems," *Journal of guidance, control, and dynamics*, Vol. 15, No. 4, 1992, pp. 1010–1017. https://doi.org/10.2514/3.20936.

[32] Antonello, A., Valverde, A., and Tsiotras, P., "Dynamics and control of spacecraft manipulators with thrusters and momentum exchange devices," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 1, 2019, pp. 15–29. https://doi.org/10.2514/1.G003601.

[33] Shi, L., Katupitiya, J., and Kinkaid, N., "A robust attitude controller for a spacecraft equipped with a robotic manipulator," *2016 American Control Conference (ACC)*, IEEE, July 2016, pp. 4966–4971. https://doi.org/10.1109/ACC.2016.7526140.

[34] Mukherjee, R., and Kamon, M., "Almost smooth time-invariant control of planar space multibody systems," *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 2, 1999, pp. 268–280. https://doi.org/10.1109/70.760348.

[35] Hokamoto, S., and Funasako, T., "Feedback control of a planar space robot using a moving manifold," *Journal of the Robotics Society of Japan*, Vol. 25, No. 5, 2007, pp. 745–751. https://doi.org/10.7210/jrsj.25.745.

[36] Kojima, H., and Kasahara, S., "An adaptive invariant manifold-based switching control for a planar two-link space robot," *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol. 54, No. 184, 2011, pp. 144–152. https://doi.org/10.2322/jjsass.58.233.

[37] Lee, K. W., Kojima, H., and Trivailo, P. M., "Applications of Optimal Trajectory Planning and Invariant Manifold Based Control for Robotic Systems in Space," *Advances in Spacecraft Technologies*, 2011, p. 497. https://doi.org/10.5772/13455.

[38] Trovarelli, F., McRobb, M., Hu, Z., and McInnes, C., "Attitude Control of an Underactuated Planar Multibody System Using Momentum Preserving Internal Torques," *AIAA Scitech 2020 Forum*, Jan. 2021, p. 1686. https://doi.org/10.2514/6.2020-1686.

[39] Yamada, K., and Yoshikawa, S., "Feedback control of space robot attitude by cyclic arm motion," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 4, 1997, pp. 715–720. https://doi.org/10.2514/2.4102.

[40] Cerven, W. T., and Coverstone, V. L., "Optimal reorientation of a multibody spacecraft through joint motion using averaging theory," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, 2001, pp. 788–795. https://doi.org/10.2514/2.4779.

[41] Nanos, K., and Papadopoulos, E., "On the use of free-floating space robots in the presence of angular momentum," *Intelligent Service Robotics*, Vol. 4, No. 1, 2011, pp. 3–15. https://doi.org/10.1007/s11370-010-0083-2.

[42] Seweryn, K., Basmadji, F. L., and Rybus, T., "Tangent Capture of an Uncontrolled Target Satellite by Space Robot: Simulation Studies," *AIAA Scitech 2020 Forum*, 2020, p. 1602. https://doi.org/10.2514/6.2020-1602.

[43] Ratajczak, J., and Tchoń, K., "Coordinate-Free Jacobian Motion Planning: A 3-D Space Robot," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021. https://doi.org/10.1109/TSMC.2021.3125276.

[44] Magnus, W., "On the exponential solution of differential equations for a linear operator," *Communications on pure and applied mathematics*, Vol. 7, No. 4, 1954, pp. 649–673. https://doi.org/10.1002/cpa.3160070404.

[45] Wilcox, R. M., "Exponential operators and parameter differentiation in quantum physics," *Journal of Mathematical Physics*, Vol. 8, No. 4, 1967, pp. 962–982. https://doi.org/10.1063/1.1705306.

[46] Bedrossian, N. S., "Classification of singular configurations for redundant manipulators," *Proceedings., IEEE International Conference on Robotics and Automation*, IEEE, 1990, pp. 818–823. https://doi.org/10.1109/ROBOT.1990.126089.

[47] Chiaverini, S., "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 3, 1997, pp. 398–410. https://doi.org/10.1109/70.585902.

[48] Wächter, A., and Biegler, L. T., "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, Vol. 106, No. 1, 2006, pp. 25–57. https://doi.org/10.1007/s10107-004-0559-y.

[49] Rudin, N., Kolvenbach, H., Tsounis, V., and Hutter, M., "Cat-Like Jumping and Landing of Legged Robots in Low Gravity Using Deep Reinforcement Learning," *IEEE Transactions on Robotics*, 2021. https://doi.org/10.1109/TRO.2021.3084374.

[50] Bertran, R., "Deep Reinforcement Learning for the Transformer Mission," Master's thesis, The University of Tokyo, Tokyo, Japan, 2021.