Singularity-Robust Full-Pose Workspace Control of Space Manipulators with Non-Zero Momentum*

Patrick Rousso, Robin Chhabra*

Autonomous Space Robotics and Mechatronics Laboratory, Carleton University, Ottawa, ON K1S 5B6, Canada

ARTICLE INFO

Keywords:
Space Manipulator
Non-zero Momentum
Workspace Trajectory Tracking
Full-pose Control
Lie Groups

ABSTRACT

We address the end-effector full-pose tracking control problem in free-floating space manipulators, experiencing constant non-zero linear and angular momentum. The aim is to develop an output-tracking (workspace) control law free of singularities due to parameterizing the end-effector motion and being robust against singularities of the input-output decoupling matrix (generalized Jacobian matrix). Space manipulators are modelled as open-chain multi-body systems with single- and multi-degree-of-freedom joints, whose kinematics and dynamics are formulated on the Special Euclidean group SE(3). Such systems exhibit conserved (not necessarily zero) total momentum when operating in the free-floating regime, which we use to systematically reduce their dynamical equations by eliminating the base spacecraft's motion. To avoid parameterizing the end-effector motion, we consider its full pose as the system output and develop a novel feedback linearization technique on the matrix Lie group SE(3) in the reduced phase space of the space manipulator. We then propose an intrinsic feedforward, feedback proportional-integral-derivative workspace controller involving a coordinate-free pose error function on SE(3) and velocity error on its Lie algebra. Using a Lyapunov candidate, this controller is proven to stabilize the end-effector pose to a feasible desired trajectory. The input-output decoupling matrix in the proposed control law can lose rank at some regions of the configuration space; hence, we implement a singularity-robust inverse, derived from the damped least squares method, to avoid impractical joint torques in these regions. The developed controller is implemented on a 7-degree-of-freedom manipulator onboard a spacecraft and its efficacy and robustness are demonstrated trough series of simulations.

1. Introduction

As society's dependence on space infrastructures continues its increasing trend, extending satellites' lifespans, technological capabilities, and commercial accessibility become increasingly important. Space manipulators that are deployed in On-Orbit Servicing (OOS) missions play a vital role in addressing the key problems (e.g., premature satellite failure and orbital debris) faced by the space industry [20, 19, 10, 37]. To reduce fuel consumption and perform safe contact operations during servicing missions, space manipulators may operate in their free-floating regime, where the base spacecraft freely rotates and translates [27]. Further, to maximize precision in the robotic operations, as well as improve their versatility and adaptability, planning and tracking workspace trajectories are favourable. Therefore, effective workspace Guidance, Navigation, and Control (GNC) of free-floating space manipulators is crucial to enhancing their functionality during the pre-capture phase and safe and successful capture and post-capture manoeuvres.

Research on the control of space manipulator systems in free-floating regime has been a highly attractive and motivated topic in space robotics. Operating in this regime comes with the complexities associated with the GNC design of underactuated systems [42, 16]. In this case, the coupled dynamics between the spacecraft and manipulator motion and the mitigation of the uncontrolled base motion are the key considerations [12, 15, 27]. Common control strategies consider reducing the coupled base motion during manipulator operation [3, 12]. For example, Wang develops a stable generalized parameter adaptive controller based on a dynamics regressor to investigate the adaptive inverse dynamics of free-floating space manipulators with parameter uncertainties [49]. Ulrich and Sasiadek demonstrate the efficacy of introducing feedforward parameters in an adaptive control structure on space manipulators with model uncertainties [43]. There is a category of free-floating space manipulator control strategies, predicated on the Virtual Manipulator (VM) technique introduced by Vafa and Dubowsky [46]. The VM method interprets free-floating space manipulators by a dynamically equivalent fixed-based manipulator system, provided no external disturbances and zero momentum. Therefore, controllers can be developed that are associated with the more trivial dynamics of a fixed-base robot [14, 47]. Parlaktuna *et al.* use the notion of the VM to develop a dynamically equivalent model of a free-floating space manipulator to perform adaptive control in the system's joint space [35, 33]. Furthermore, Torres and Dubowsky study path planning and control structures based on the VM approach [40, 13, 41]. A separate foundational approach for controlling free-floating space manipulators is introduced by Yoshida and Umetani in [44], using the notion of the Generalized Jacobian Matrix (GJM) [45]. This notion allowed for Nenchev and Yoshida to introduce the bias

*This work was partially supported by the Natural Sciences and Engineering Research Council of Canada and the Canada Research Chair Program.

*Corresponding author at: Department of Mechanical and Aerospace Engineering, 3135 Mackenzie Building, Carleton University, Ottawa, ON K1S 5B6, Canada, Tel.: 613-520-2600, ext. 4251

Email: robin.chhabra@carleton.ca (R. Chhabra)
ORCID(s): 0000-0003-0023-1928 (R. Chhabra)

momentum approach in [51] which considers impedance control and the minimization of changes in the base's orientation during the pre-capture of a tumbling target. The Reaction Null-Space (RNS) is developed using the GJM method to identify the motions in the manipulator's joint space which impart the lowest disturbance to the base [50].

The GJM approach allows for the implementation of workspace control techniques in space manipulators. In [4, 2], the authors feedback linearize space manipulators on $\mathbb{R}^n$ to perform end-effector position and base attitude control during pre- and post-capture maneuvers. A tracking controller in the presence of angular momentum is formulated in [32]. In the resulting linearized input-output relation, various types of strategies, including traditional linear Proportional-Derivative (PD) and Proportional-Integral-Derivative (PID) control, can be applied to follow a desired trajectory. Chhabra *et. al* formally study space manipulators with constant non-zero momentum and reduce their phase space in a Hamiltonian formalism [8]. Accordingly, they develop a feedback transformation on an output space with a Lie group structure that linearizes the closed loop system in the presence of non-zero momentum [9]. They employ a widely versatile PD control law on Lie groups, proposed by Bullo and Murray, to follow a desired end-effector pose in a stable manner [5]. To guarantee better performance and robustness, intrinsic PID control laws that have been developed on Lie groups can be also used at the output level [25, 53]. On the Lie group SE(3), these intrinsic PID methods have been combined with the error function proposed in [17] to design robust controllers for aerospace vehicles [18, 1].

Prior to analyzing and developing appropriate GNC strategies for free-floating space manipulators, their kinematics and dynamics models must be established. This paper considers free-floating space manipulators as open-chain multi-body systems. For general rigid multi-body systems, [30] and [34] have taken a geometric approach by developing kinematics and dynamics formulations of multi-body systems on Lie groups. The relationship between Lie groups and screw theory is analyzed in [39], where screws are presented as elements of Lie algebras, allowing for a geometric interpretation of rigid body motions. Provided the connection between screw motions and Lie group theory, kinematic mappings from the joint space to the task space (a subset of SE(3)) of a serial chain manipulator can be formulated [36, 28, 7]. In establishing a space manipulator's equations of motion, Hamiltonian and Lagrangian approaches prove beneficial due to their accommodation to Lie group formulations [7, 8, 38]. Note that due to the low-gravity orbital environment, space manipulators are often considered to possess negligible potential energy. The presence of external disturbances are often additionally presumed to be insignificant during the small time period in which space manipulators perform servicing operations, thus implying an assumption of conserved momentum in the system [27].

In this paper, we study the kinematics and dynamics of a free-floating space manipulator with multi-Degree-Of-Freedom (DOF) joints on Lie groups. Taking advantage of their conservation of momentum, we develop a control law that can control the full pose of the system's end-effector. The controller is predicated on coordinate-free pose and velocity error functions to achieve Lyapunov stable trajectory following. The main contributions of the paper can be summarized in the following.

1. We systematically extend the product of exponentials formula for fixed-base manipulators with single-Degree-Of-Freedom (DOF) joints to capture the kinematics and dynamics of free-floating space manipulators with multi-DOF joints on Lie groups.
2. We rigorously reduce the equations of motion of free-floating space manipulators with non-zero momentum. Accordingly, we propose a novel feedback transformation on the Lie algebra $\mathfrak{se}(3)$ that input-output linearizes the system.
3. A coordinate-free feedforward, feedback PID control law on SE(3) is developed to achieve Lyapunov stable trajectory following of the end-effector pose of the robot.
4. The developed output-tracking control structure implements a singularity-robust inverse method to accomodate kinematic and dynamic singularities of the space manipulator system.

The rest of this paper is organized as follows: Section 3 formulates the kinematics of space manipulators with multi-DOF joints. In Section 4, we study the dynamics of space manipulators and subsequently present its reduction under the conservation of momentum assumption in Sections 5 and 6. Section 7 performs feedback linearization on the end-effector motion to then develop a full-pose workspace controller in Section 8 with a singularity-robust feature outlined in Section 9. Section 10 provides a series of numerical studies. Some concluding remarks are included in Section 11.

2. Preliminaries

In this section, we review the Lie group SE(3) as the configuration space of a single rigid body moving in the 3-dimensional Euclidean. Members of the SE(3) describe the pose of a rigid body in the form of 4×4 homogeneous transformation matrices,

$$\boldsymbol{g} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}. \tag{1}$$

Here, $\boldsymbol{R} \in \mathrm{SO}(3) \subset \mathbb{R}^{3\times 3}$ is a 3×3 rotation matrix defining the body's orientation, where SO(3) is the Special Orthogonal group, and $\boldsymbol{p} \in \mathbb{R}^3$ is the body's position. Using the exponential map of SO(3), any rotation $\boldsymbol{R}$ about an axis $\boldsymbol{w} \in \mathbb{S}^2 \subset \mathbb{R}^3$ on the unit 2-sphere with a rotation angle Φ can be described by $\boldsymbol{R} = e^{\tilde{\boldsymbol{w}}\Phi} \in \mathrm{SO}(3)$. The tilde

operator indicates the vector space isomorphism between $\mathfrak{so}(3)$, the Lie algebra of SO(3), and $\mathbb{R}^3$, mapping a vector $\boldsymbol{u} = [u_1\ u_2\ u_3]^T \in \mathbb{R}^3$ to the skew-symmetric matrix

$$\tilde{\boldsymbol{u}} = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \in \mathfrak{so}(3). \tag{2}$$

The exponential map of SO(3) can be calculated in closed form using the Rodgrigues formula:

$$e^{\tilde{\boldsymbol{w}}\Phi} = \mathbf{1}_{3\times3} + \tilde{\boldsymbol{w}}\sin\Phi + \tilde{\boldsymbol{w}}^2(1-\cos\Phi).\ \|\boldsymbol{w}\| = 1 \tag{3}$$

Analogous to the exponential map for rotations in SO(3), the exponential map of SE(3) can be defined for the members of its Lie algebra, denoted $\mathfrak{se}(3)$. Members of $\mathfrak{se}(3)$ are referred to as twists, and take the following form:

$$\hat{\boldsymbol{\xi}} = \begin{bmatrix} \tilde{\boldsymbol{w}} & \boldsymbol{v} \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} \in \mathfrak{se}(3), \tag{4}$$

where $\boldsymbol{\xi} = [\boldsymbol{v}^T\ \boldsymbol{w}^T]^T \in \mathbb{R}^6$, and the hat operator is defined to be the vector space isomorphism between $\mathbb{R}^6$ and $\mathfrak{se}(3)$. The exponential map of SE(3) for a twist $\hat{\boldsymbol{\xi}} \in \mathfrak{se}(3)$ with $\|\boldsymbol{w}\| = 1$ and twist angle Φ is then calculated by

$$e^{\hat{\boldsymbol{\xi}}\Phi} = \begin{bmatrix} e^{\tilde{\boldsymbol{w}}\Phi} & (\mathbf{1}_{3\times3} - e^{\tilde{\boldsymbol{w}}\Phi})(\tilde{\boldsymbol{w}}\boldsymbol{v}) + \boldsymbol{w}\boldsymbol{w}^T\boldsymbol{v}\Phi \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}. \tag{5}$$

The resulting homogeneous transformation is the transformation obtained after performing a screw motion about the screw axis $\boldsymbol{\xi}$ for the angle Φ. For a purely translational rigid body motion, i.e., $\boldsymbol{w} = 0$, the exponential is simply

$$e^{\hat{\boldsymbol{\xi}}\Phi} = \begin{bmatrix} \mathbf{1}_{3\times3} & \boldsymbol{v}\Phi \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{6}$$

where for this case Φ refers to the amount of translational motion associated with the twist. For more general rigid body motions let $\boldsymbol{g}_{sb}(t) \in$ SE(3) indicate a trajectory of a body coordinate frame's pose with respect to the spatial frame. The matrices $\dot{\boldsymbol{g}}_{sb}\boldsymbol{g}_{sb}^{-1}$ and $\boldsymbol{g}_{sb}^{-1}\dot{\boldsymbol{g}}_{sb}$ belong to the Lie algebra $\mathfrak{se}(3)$, and they respectively define the rigid body's instantaneous spatial velocity $\boldsymbol{V}_{sb}^s \in \mathbb{R}^6$ and body velocity $\boldsymbol{V}_{sb}^b \in \mathbb{R}^6$ vectors with respect to the spatial frame. The spatial velocity in twist coordinates reads

$$\hat{\boldsymbol{V}}_{sb}^s = \dot{\boldsymbol{g}}_{sb}\boldsymbol{g}_{sb}^{-1} = \begin{bmatrix} \boldsymbol{v}_{sb}^s \\ \boldsymbol{\omega}_{sb}^s \end{bmatrix}^\wedge \in \mathfrak{se}(3). \tag{7}$$

The first three components of the spatial velocity vector $\boldsymbol{v}_{sb}^s \in \mathbb{R}^3$ correspond to the linear velocity with respect to the spatial coordinate frame. Physically speaking, this linear velocity represents the velocity of a particle in the rigid body which travels through the origin of the spatial frame at the given moment in time. Furthermore, the last three components of the spatial velocity $\boldsymbol{\omega}_{sb}^s \in \mathbb{R}^3$ specify the spatial angular velocity of the body. This angular velocity simply represents the rigid body's rate of rotation from the perspective of the spatial frame. Moreover the body velocity in twist coordinates is defined by

$$\hat{\boldsymbol{V}}_{sb}^b = \boldsymbol{g}_{sb}^{-1}\dot{\boldsymbol{g}}_{sb} = \begin{bmatrix} \boldsymbol{v}_{sb}^b \\ \boldsymbol{\omega}_{sb}^b \end{bmatrix}^\wedge \in \mathfrak{se}(3). \tag{8}$$

Here, the linear component of the body velocity $\boldsymbol{v}_{sb}^b \in \mathbb{R}^3$ is physically interpreted as the relative velocity between the origins of the body and spatial coordinate frames viewed from the perspective of the body frame. The angular velocity $\boldsymbol{\omega}_{sb}^b \in \mathbb{R}^3$ again represents the rotational velocity of the rigid body as viewed in the body frame.

Relating the spatial and body velocity vectors requires some form of a transformation between the two coordinate frames. The Adjoint operator is a linear automorphism of the Lie algebra $\mathfrak{se}(3)$ which converts the expression of a twist from one coordinate frame to another. Provided a relative homogeneous transformation between two coordinate frames $\boldsymbol{g} \in$ SE(3) with a rotation $\boldsymbol{R} \in$ SO(3) and a position $\boldsymbol{p} \in \mathbb{R}^3$, the associated Adjoint mapping denoted $\mathbf{Ad}_{\boldsymbol{g}}$ is defined by the following 6×6 matrix,

$$\mathbf{Ad}_g = \begin{bmatrix} \boldsymbol{R} & \tilde{\boldsymbol{p}}\boldsymbol{R} \\ \mathbf{0}_{3\times3} & \boldsymbol{R} \end{bmatrix}. \tag{9}$$

Further, for all $\hat{\boldsymbol{\xi}} \in \mathfrak{se}(3)$, the Lie bracket on $\mathfrak{se}(3)$ introduces a linear mapping $\mathbf{ad}_{\hat{\boldsymbol{\xi}}}$, coined as the adjoint operator, between elements of $\mathfrak{se}(3)$:

$$\mathbf{ad}_{\hat{\boldsymbol{\xi}}} = \begin{bmatrix} \tilde{\boldsymbol{w}}_1 & \tilde{\boldsymbol{v}}_1 \\ \mathbf{0}_{3\times3} & \tilde{\boldsymbol{w}}_1 \end{bmatrix}. \tag{10}$$

3. Kinematics of Space Manipulators Systems

This paper considers free-floating space manipulator systems consisting of an N-link manipulator attached to a 6-DOF base spacecraft. All manipulator links are interconnected through either single or multi-DOF joints. The space manipulator is modelled as a serial-link open-chain rigid multi-body system with $N+1$ bodies. All body coordinate frames are attached to the bodies' centers of masses. We number the bodies by $k = 0, \cdots, N$, where body 0 refers to the base, and we label joints by the number of their succeeding body. By $N_D = N_{D_M} + 6$ we denote the number of degrees of freedom in the system, where $N_{D_M} = \sum_{k=1}^{N} l_k$ indicates those included in the manipulator. Here, l_k is the number of degrees of freedom of Joint k.

We use the exponential map to parameterize single-DOF prismatic and revolute joints [30], and we extend this parameterization to a product of exponentials for a class of multi-DOF joints. Six common types of lower pair joints are considered in this paper, as listed in Table 1. The configuration space of a joint in this list can be parameterized by a product of exponentials corresponding to the joint's axes of

Table 1
Lower pair joints

Dim.	Configuration space in SE(3)	
3	SE(2) planar	SO(3) ball (spherical)
2	SO(2) × ℝ cylindrical	SO(2) × SO(2) universal
1	SO(2) revolute	ℝ prismatic

motion. Lower pair joints include single-DOF revolute and prismatic joints, along with the multi-DOF universal, planar, cylindrical, and spherical joints.

For a revolute joint connecting Body $k-1$ and Body k, the joint twist is defined such that the unit vector $\boldsymbol{w} \in \mathbb{R}^3$ is the axis of rotation for the relative rigid body motion with respect to the preceding body, and $\boldsymbol{v} = -\tilde{\boldsymbol{w}}\boldsymbol{r}$, with $\boldsymbol{r} \in \mathbb{R}^3$ being a point along the axis of rotation. For a prismatic joint, the joint twist is defined such that $\boldsymbol{w} = 0$ and the unit vector $\boldsymbol{v} \in \mathbb{R}^3$ is the axis of translation relative to the preceding rigid body. Therefore given Joint k with the joint twist $\boldsymbol{\xi}'_k$ (observed from Body $k-1$ at its initial pose), the relative homogeneous transformation between the two bodies is parameterized using the joint parameter $\Phi_k \in \mathbb{R}$ as

$$\boldsymbol{g}_{(k-1)k}(\Phi_k) = e^{\hat{\boldsymbol{\xi}}'_k \Phi_k} \boldsymbol{g}_{(k-1)k}(0). \tag{11}$$

Here, $\boldsymbol{g}_{(k-1)k}(0)$ is the initial relative pose between the two bodies. Starting with the single-DOF joints located at the bottom row of Table 1, the revolute and prismatic joints refer to pure rotational and translational screw motions, respectively. Revolute joints are described by the one-parameter subgroups of SE(3), denoted by SO(2), which specifies rotation about a single axis, and prismatic joints are described by ℝ which indicates translation in a single direction. The helical joint contains a single independent motion, represented by the subgroup $\boldsymbol{H}_p$, to express a complete screw motion with a non-zero and finite pitch value. Cylindrical joints combine individual revolute and prismatic motions SO(2) × ℝ, to both rotate and translate an attached rigid body. The planar and spherical joints in the lower pair group both impart motions with three degrees of freedom. Planar joints posses full planar motion (i.e., translation and rotation in two-dimensional space), combining $\mathbb{R}^2$ and SO(2) to define the Special Euclidean space for two dimensional motion, SE(2). A spherical joint allows for complete rotational motion, described by rotation matrices belonging to SO(3). We may also include the free six-DOF joint to describe unconstrained motion in SE(3). This six-DOF joint proves necessary when describing the uncontrolled base motion of a space manipulator in its free-floating regime.

As mentioned, the motion of the more trivial single-DOF joints are represented in a geometric context by a screw motion (helical joint motion) with either infinite pitch (prismatic joint motion) or zero pitch (revolute joint motion). Subsequently, provided the twist $\boldsymbol{\xi}$ associated with the single-DOF joint motion, the relative motion between a rigid body's initial configuration and its configuration following the joint motion is obtained through the exponential mapping $\exp(\hat{\boldsymbol{\xi}}\Phi)$. With regards to the multi-DOF joints, this paper expresses their motion as an amalgamation of individual screw motions, each representing a single degree of freedom in the joint's motion. For example, the full rotational motion of a spherical joint is effectively defined by the combination of three revolute joints placed along each of the primary axes. For a body k with an initial configuration $\boldsymbol{g}_{sk}(0)$, and whose motion is defined by a spherical joint, its resulting configuration following the spherical motion $\boldsymbol{g}_{sk}(\Phi_z, \Phi_y, \Phi_x)$ is defined as follows,

$$\boldsymbol{g}_{sk}(\Phi_z, \Phi_y, \Phi_x) = e^{\hat{\boldsymbol{\xi}}_z \Phi_z} e^{\hat{\boldsymbol{\xi}}_y \Phi_y} e^{\hat{\boldsymbol{\xi}}_x \Phi_x} \boldsymbol{g}_{sk}(0). \tag{12}$$

The representation of the spherical joint above signifies an Euler angle representation of the joint's motion. The rotation angles Φ_z, Φ_y, and Φ_x refer to the yaw, pitch, and roll of the joint, and $\boldsymbol{\xi}_z, \boldsymbol{\xi}_y$, and $\boldsymbol{\xi}_x$ refer to twists along the z, y, and x axes. Note that for a planar joint (with motion in SE(2)), the same combination of exponential mappings in equation (12) is used, however with the last two exponentials referring to translational motions along the y and x axes. The principle of combining individual revolute and prismatic motions to represent higher DOF joints provides the following general representation of joint motions on Lie groups for a joint m with l_m degrees of freedom,

$$\boldsymbol{g}_m(\Phi_1, \ldots, \Phi_{l_m}) = e^{\hat{\boldsymbol{\xi}}_1 \Phi_1} \ldots e^{\hat{\boldsymbol{\xi}}_{l_m} \Phi_{l_m}} \boldsymbol{g}_m(0). \tag{13}$$

The exponential mapping $\exp(\hat{\boldsymbol{\xi}}_i \Phi_i)$ may represent either a pure rotational or translational motion. Based on the Lie group parameterization of the six lower pair joints in Table 1, and the defined motion for single and multi-DOF joints, the system's forward kinematics can be formed. The following subsection describes the method for combining consecutive joint motions to define the forward kinematics mapping from the system configuration to the end-effector pose in SE(3).

Note that the joint twists capturing the base motion are denoted by $\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_6$, while the twists of the manipulator joints are denoted by $\boldsymbol{\xi}_7, \cdots, \boldsymbol{\xi}_{N_D}$. The definition of joint twists throughout the system will be discussed in the next section. Let $\boldsymbol{\theta} \in \mathbb{R}^6$ be the collection of the parameters describing the base spacecraft's motion, and $\boldsymbol{q} \in \mathbb{R}^{N_{D_M}}$ be the collection of the joint parameters for the manipulator. The vector of generalized coordinates for a space manipulator system is thus the combination of $\boldsymbol{\theta}$ and $\boldsymbol{q}$, denoted by $\boldsymbol{\Phi} = [\boldsymbol{\theta}^T \ \boldsymbol{q}^T]^T \in \mathbb{R}^{N_D}$.

3.1. Forward Kinematics

As mentioned, the primary objective of the forward kinematics of a space manipulator system is to determine

the position and orientation of end-effector $\boldsymbol{g}_{sN}$ provided the joint configurations in the system. More generally, the forward kinematics model provides a function for computing the configuration of any body k in the system given the joint configurations preceding Body k. The region of allowable joint values is referred to as the system's joint space, and the region of achievable end-effector configurations is referred to as either the system's task space or workspace. Following a right-handed coordinate system, a counterclockwise rotation defines the positive direction of rotational motions if viewed along the axis of rotation. For translational motion, the displacement is measured positive in the direction of the corresponding screw axis.

Starting with the homogeneous transformation from the spatial frame to the coordinate frame following the first joint's motion $\boldsymbol{g}_{s0}$, the forward kinematics map sequentially merges individual joint transformations until reaching the end-effector. That is, for a general robotic manipulator with N bodies, and where Body k is attached to a joint with l_{m_k} degrees of freedom, the forward kinematics mapping to the end-effector is defined as,

$$\begin{aligned}\boldsymbol{g}_{sN}(\boldsymbol{\Phi}) = \boldsymbol{g}_{s0}(\Phi_1, \dots, \Phi_{l_0})\boldsymbol{g}_{01}(\Phi_{l_0+1}, \dots, \Phi_{l_1}) \\ \cdots \boldsymbol{g}_{(N-1)N}(\Phi_{l_{N-1}+1}, \dots, \Phi_{l_N})\boldsymbol{g}_{sN}(0).\end{aligned} \tag{14}$$

Here, $l_k = \sum_{i=0}^{k} l_{m_i}$ defines the number of degrees of freedom in the system preceding Body k. For example, the spacecraft's base has six degrees of freedom, thus defining $l_0 = 6$. The full forward kinematics mapping to the end-effector above can be easily generalized to any body k in the system by simply ending the product of joint transformations at Body k. This implies that the combination of joint transformations in equation (14) ends at $\boldsymbol{g}_{(k-1)k}$ as opposed to $\boldsymbol{g}_{(N-1)N}$. Note that the homogeneous transformation between two connected bodies is defined by equation (13) with $l_m = l_{m_k}$. Implementing equation (13) into equation (14) demonstrates the general forward kinematic mapping in terms of each individual prismatic or revolute motion included in the system. The notion of compounding individual motions (corresponding to either single or multi-DOF joints) using the exponential joint parameterization in the forward kinematics model is known as the product of exponentials formula. The forward kinematics mapping for Body k can thus be expressed by the product of exponentials formula as follows,

$$\boldsymbol{g}_{sk}(\boldsymbol{\Phi}) = e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \cdots e^{\hat{\boldsymbol{\xi}}_{l_k}\Phi_{l_k}}\boldsymbol{g}_{sk}(0). \quad k = 0, \cdots, N \tag{15}$$

The product of exponentials formula above signifies the transformation of the k^{th} Body's initial configuration into its updated pose based on the current configuration of the space manipulator system.

3.2. Differential Kinematics

Differentiating the forward kinematics map provides a linear mapping, known as the system's Jacobian $\boldsymbol{J}(\boldsymbol{\Phi}) \in \mathbb{R}^{6\times N_D}$, from the system's joint velocities to the velocity vector of a body k. This Jacobian matrix can be expressed in either the spatial frame $\boldsymbol{J}^s(\boldsymbol{\Phi})$ or the body frame $\boldsymbol{J}^b(\boldsymbol{\Phi})$ to map velocity vectors in the respective frames of reference. For example, the spatial velocity of Body k, denoted $\boldsymbol{V}_k^s$, requires Body k's spatial Jacobian $\boldsymbol{J}_k^s(\boldsymbol{\Phi})$ and the system's joint velocities $\dot{\boldsymbol{\Phi}}$,

$$\boldsymbol{V}_k^s = \boldsymbol{J}_k^s(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}} \in \mathfrak{se}(3). \tag{16}$$

As indicated by the nomenclature, the system Jacobian is a function of the generalized coordinates $\boldsymbol{\Phi}$ due to its dependency on the forward kinematics mapping. Recalling the definition of the spatial velocity in equation (7), we expand the time derivative of the homogeneous transformation $\boldsymbol{g}_{sk}$,

$$\dot{\boldsymbol{g}}_{sk}(\boldsymbol{\Phi}) = \sum_{i=1}^{N_D} \frac{\partial \boldsymbol{g}_{sk}(\boldsymbol{\Phi})}{\partial \Phi_i}\dot{\Phi}_i. \tag{17}$$

Based on the definition of the spatial velocity,

$$\hat{\boldsymbol{V}}_k^s = \sum_{i=1}^{N_D}\left(\frac{\partial \boldsymbol{g}_{sk}(\boldsymbol{\Phi})}{\partial \Phi_i}\dot{\Phi}_i\right)\boldsymbol{g}_{sk}^{-1}(\boldsymbol{\Phi}) = \sum_{i=1}^{N_D}\left(\frac{\partial \boldsymbol{g}_{sk}(\boldsymbol{\Phi})}{\partial \Phi_i}\boldsymbol{g}_{sk}^{-1}(\boldsymbol{\Phi})\right)\dot{\Phi}_i. \tag{18}$$

Appendix A provides the derivation of the partial derivative of a homogeneous transformation with respect to a generalized coordinate Φ. The result of this differentiation signifies that the argument of the summation above can be replaced by applying the Adjoint of the product of exponentials up to Φ_{i-1} to the i^{th} twist $\boldsymbol{\xi}_i$. That is,

$$\left(\frac{\partial \boldsymbol{g}_{sk}}{\partial \Phi_i}\boldsymbol{g}_{sk}^{-1}\right)^{\vee} = \mathbf{Ad}_{(e^{\hat{\boldsymbol{\xi}}_1\theta_1} \dots e^{\hat{\boldsymbol{\xi}}_{i-1}\theta_{i-1}})}\boldsymbol{\xi}_i. \tag{19}$$

Therefore, the spatial Jacobian matrix is defined by expressing the summation component of equation (18) in its matrix form, making use of the result in (19). This yields the following definition of $\boldsymbol{J}_k^s$,

$$\boldsymbol{J}_k^s = \begin{bmatrix}\boldsymbol{\eta}_1' & \cdots & \boldsymbol{\eta}_{l_k}' & \mathbf{0}_{6\times(N_D-l_k)}\end{bmatrix} \quad \in \mathbb{R}^{6\times N_D} \tag{20}$$

$$\boldsymbol{\eta}_i' = (\mathbf{Ad}_1^{i-1})'\boldsymbol{\xi}_i. \quad i = 1, \cdots, l_k, \tag{21}$$

where,

$$(\mathbf{Ad}_1^{i-1})' := \mathbf{Ad}_{(e^{\hat{\boldsymbol{\xi}}_1\theta_1} \dots e^{\hat{\boldsymbol{\xi}}_{i-1}\theta_{i-1}})}. \tag{22}$$

From this matrix definition, the i^{th} column of the spatial Jacobian represents the transformation from the i^{th} twist in the initial configuration to its updated location, given the multi-body system's current configuration, with respect to the spatial coordinate frame. This transformation is based on all preceding $i-1$ joint angles, and defined by the

Adjoint matrix in equation (22). Consequently, the spatial velocity of Body k only relies on the preceding joint angles and is independent of the succeeding joint parameters. This is expressed in the Jacobian matrix by all $l_k + 1, \cdots, N_D$ columns being zero.

Similarly, the body Jacobian associated with the k^{th} Body provides the linear mapping from the joint velocities to Body k's body velocity,

$$\boldsymbol{V}_k^b = \boldsymbol{J}_k^b(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}}. \tag{23}$$

Based on the definition of $\boldsymbol{V}_k^b$ in equation (8) and the time derivative of $\boldsymbol{g}_{sk}$, the body Jacobian can be formed provided the following general definition of its matrix columns,

$$\boldsymbol{J}_k^b = [\boldsymbol{\eta}_1 \ \cdots \ \boldsymbol{\eta}_{l_k} \quad \boldsymbol{0}_{6\times(N_D-l_k)}] \quad \in \mathbb{R}^{6\times N_D} \tag{24}$$

$$\boldsymbol{\eta}_i = \mathbf{Ad}_{\boldsymbol{g}_{sk}^{-1}(0)}\mathbf{Ad}_i^{l_k}\boldsymbol{\xi}_i, \quad i = 1, \cdots, l_k \tag{25}$$

where for $l > i$,

$$\mathbf{Ad}_i^l := \mathbf{Ad}_{(e^{-\hat{\xi}_l\Phi_l} \ \cdots \ e^{-\hat{\xi}_i\Phi_i})}. \tag{26}$$

In this body frame representation of the Jacobian matrix, the i^{th} column expresses the updated i^{th} twist with respect to the coordinate frame attached to Body k. Again, the transformation updating the twist to the current configuration is predicated on the joint configurations preceding Body k, as expressed by the Adjoint multiplication in equation (26). Analogous to the connection between the spatial and body velocities, the spatial and body manipulator Jacobians can also be related through the Adjoint operator associated with the relating homogeneous transformation $\boldsymbol{g}_{sk}$,

$$\boldsymbol{J}_k^s = \mathbf{Ad}_{\boldsymbol{g}_{sk}}\boldsymbol{J}_k^b. \tag{27}$$

The following section makes use of the differential kinematics mapping in forming the kinetic energy of a space manipulator system. Consequently, the system's Jacobian matrix forms the foundation for deriving the dynamical equations of space manipulators in the context of Lie groups.

4. Space Manipulator System Dynamics

In this section, the equations of motion for a free-floating space manipulator system are derived based on an Euler-Lagrangian approach. This method is known to be computationally inefficient in comparison to the Newton-Euler process; however its use is necessary for developing model-based control schemes. The equations of motion dictate the acceleration of a rigid multi-body system under actuation forces and torques applied to the joints. Using a Lagrangian approach requires definitions of the system's kinetic and potential energies to define the system's Lagrangian $\boldsymbol{L}$. In the case of a space manipulator system in orbit, the system's potential energy can be neglected due to the low-gravity environment. Consequently, the Lagrangian of a space manipulator is simply equal to the system's kinetic energy $\boldsymbol{K}$. Differentiating the Lagrangian with respect to the generalized coordinates and their velocities establishes a relationship between the generalized accelerations $\ddot{\boldsymbol{\Phi}}$ and the generalized forces and moments $\boldsymbol{T}$. To start, we may express the equation of motion for a single generalized coordinate as follows,

$$\frac{d}{dt}\frac{\partial \boldsymbol{L}}{\partial \dot{\Phi}_i} - \frac{\partial \boldsymbol{L}}{\partial \Phi_i} = T_i. \tag{28}$$

Given the definition of the Lagrangian for space manipulator systems $\boldsymbol{L} = \boldsymbol{K}$ previously mentioned, we begin by defining the kinetic energy associated with the k^{th} Body,

$$\boldsymbol{K}_k(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \frac{1}{2}(\boldsymbol{V}_k^b)^T\boldsymbol{D}_k\boldsymbol{V}_k^b, \tag{29}$$

where $\boldsymbol{D}_k$ denotes Body k's constant 6×6 inertia matrix with respect to its body frame. Again, this paper considers body-fixed coordinate frames to be located at the body's center of mass and aligned in the direction of the body's principal axes, resulting in the mass matrix to take the following general form (assuming a uniform mass distribution),

$$\boldsymbol{D}_k = \begin{bmatrix} m_k & 0 & 0 & 0 & 0 & 0 \\ 0 & m_k & 0 & 0 & 0 & 0 \\ 0 & 0 & m_k & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xk} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yk} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zk} \end{bmatrix}. \tag{30}$$

Here, m_k refers to the mass of Body k, and I_{xk}, I_{yk}, and I_{zk} refer to Body k's principal moments of inertia about the x, y, and z axes respectively. The kinetic energy's dependency on $\boldsymbol{\Phi}$ and $\dot{\boldsymbol{\Phi}}$ becomes apparent when expressing the body velocity vectors in equation (29) in terms of their differential kinematics mappings,

$$\boldsymbol{K}_k(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \frac{1}{2}(\boldsymbol{J}_k^b(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}})^T\boldsymbol{D}_k\boldsymbol{J}_k^b(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}}, \tag{31}$$

which is rewritten as,

$$\boldsymbol{K}_k(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \frac{1}{2}\dot{\boldsymbol{\Phi}}^T(\boldsymbol{J}_k^b(\boldsymbol{\Phi}))^T\boldsymbol{D}_k\boldsymbol{J}_k^b(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}}. \tag{32}$$

The resulting expression above signifies a dependency on Body k's inertia matrix and body Jacobian in describing its kinetic energy. Combining the kinetic energies of each body yields the system's total kinetic energy,

$$\boldsymbol{K}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \sum_{k=0}^{N}\boldsymbol{K}_k(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}), \tag{33}$$

which thus depends on the matrix products of the body Jacobians and body inertia matrices. This summation can be simplified by introducing the system inertia matrix $\bar{\boldsymbol{M}}(\boldsymbol{\Phi}) \in \mathbb{R}^{N_D \times N_D}$ defined as

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi}) := \sum_{k=1}^{N} (\boldsymbol{J}_k^b(\boldsymbol{\Phi}))^T \boldsymbol{D}_k \boldsymbol{J}_k^b(\boldsymbol{\Phi}), \tag{34}$$

such that,

$$\boldsymbol{K}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \frac{1}{2} \dot{\boldsymbol{\Phi}}^T \bar{\boldsymbol{M}}(\boldsymbol{\Phi}) \dot{\boldsymbol{\Phi}}. \tag{35}$$

To efficiently formulate $\bar{\boldsymbol{M}}(\boldsymbol{\Phi})$, the summation in equation (34) can be expressed in terms of a single matrix multiplication. In doing so, we define a $6N_D \times N_D$ matrix $\boldsymbol{J}$ which contains the body Jacobian matrices for each body in the system,

$$\boldsymbol{J}(\boldsymbol{\Phi}) := \begin{bmatrix} \boldsymbol{J}_0^b(\boldsymbol{\Phi}) \\ \boldsymbol{J}_1^b(\boldsymbol{\Phi}) \\ \vdots \\ \boldsymbol{J}_N^b(\boldsymbol{\Phi}) \end{bmatrix}, \tag{36}$$

and a $6N_D \times 6N_D$ block diagonal matrix collecting the individual body inertia matrices,

$$\boldsymbol{D} := \begin{bmatrix} \boldsymbol{D}_0 & 0 & \dots & 0 \\ 0 & \boldsymbol{D}_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{D}_N \end{bmatrix}. \tag{37}$$

Based on the definition of the matrices $\boldsymbol{J}$ and $\boldsymbol{D}$, the system inertia matrix is expressed as follows,

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi}) = \boldsymbol{J}^T(\boldsymbol{\Phi}) \boldsymbol{D} \boldsymbol{J}(\boldsymbol{\Phi}). \tag{38}$$

The inertia matrix can be partitioned by rows and columns to define four sub-matrices. These sub-matrices separate the inertial contributions to the system dynamics from the spacecraft base and the manipulator,

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi}) = \begin{bmatrix} \boldsymbol{H}_o & \boldsymbol{H}_{om} \\ \boldsymbol{H}_{om}^T & \boldsymbol{H}_m \end{bmatrix}. \tag{39}$$

Here, $\boldsymbol{H}_o$ is the 6×6 matrix of the base contribution to the system's momentum represented in the base parameterization, $\boldsymbol{H}_{om}$ is the $6 \times N$ matrix of the manipulator contribution to the system's momentum (similarly expressed in the base parameterization), and $\boldsymbol{H}_m$ is the $N \times N$ matrix of the manipulator's inertia.

Given the total system kinetic energy expressed in equation (35), the resulting equations of motion of a space manipulator are found using the Euler-Lagrangian equation in (28). Since the system inertia matrix is only dependent on the generalized coordinates $\boldsymbol{\Phi}$, the partial derivative of the Lagrangian with respect to the generalized velocities is

$$\frac{\partial \boldsymbol{L}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}})}{\partial \dot{\boldsymbol{\Phi}}} = \bar{\boldsymbol{M}}(\boldsymbol{\Phi}) \dot{\boldsymbol{\Phi}}. \tag{40}$$

Taking the time derivative of this equation defines the first term in the Euler-Lagrange equation:

$$\frac{d}{dt} \frac{\partial \boldsymbol{L}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}})}{\partial \dot{\boldsymbol{\Phi}}} = \bar{\boldsymbol{M}}(\boldsymbol{\Phi}) \ddot{\boldsymbol{\Phi}} + \dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi}) \dot{\boldsymbol{\Phi}}, \tag{41}$$

where the time derivative of the system inertia matrix is calculated as follows,

$$\dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi}) = \sum_{i=1}^{N_D} \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_i} \dot{\Phi}_i. \tag{42}$$

As the inertia matrix's dependency on the generalized coordinates stems from the body Jacobian, the partial derivative of $\bar{\boldsymbol{M}}(\boldsymbol{\Phi})$ with respect to Φ_i can be expressed in terms of equivalent partial derivatives of the Jacobian matrix by way of the chain rule,

$$\frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_i} = \left(\frac{\partial \boldsymbol{J}(\boldsymbol{\Phi})}{\partial \Phi_i} \right)^T \boldsymbol{D} \boldsymbol{J}(\boldsymbol{\Phi}) + \boldsymbol{J}^T(\boldsymbol{\Phi}) \boldsymbol{D} \frac{\partial \boldsymbol{J}(\boldsymbol{\Phi})}{\partial \Phi_i}. \quad i = 1, \cdots, N_D \tag{43}$$

Given the form of the j^{th} column of Body k's body Jacobian in equation (25), only the Adjoint operator $\mathbf{Ad}_i^{l_k}$ is dependent on the system's configuration $\boldsymbol{\Phi}$. Note that the Adjoint of all twists, and the initial pose $\boldsymbol{g}_{sk}(0)$ are independent of the system's configuration. Consequently, the derivative of the j^{th} column of Body k's body Jacobian with respect to Φ_i becomes

$$\frac{\partial \boldsymbol{J}_{k,j}^b}{\partial \Phi_i} = \mathbf{Ad}_{\boldsymbol{g}_{sk}^{-1}(0)} \frac{\partial \mathbf{Ad}_j^{l_k}}{\partial \Phi_i} \boldsymbol{\xi}_j. \tag{44}$$

The expression for the partial derivative of the Adjoint operator is presented below, resulting from the complete derivation provided in Appendix B,

$$\frac{\partial \mathbf{Ad}_j^{l_k}}{\partial \Phi_i} = -\mathbf{Ad}_{i+1}^{l_k} \mathbf{ad}_{\boldsymbol{\xi}_i} \mathbf{Ad}_j^i. \tag{45}$$

Implementing the derivative of the Adjoint operator into equation (44), the partial derivative of the j^{th} column of the body Jacobian is written as,

$$\frac{\partial \boldsymbol{J}_{k,j}^b}{\partial \Phi_i} = -\mathbf{Ad}_{\boldsymbol{g}_{sk}^{-1}(0)} \mathbf{Ad}_{i+1}^{l_k} \mathbf{ad}_{\boldsymbol{\xi}_i} \mathbf{Ad}_j^i \boldsymbol{\xi}_j. \quad i \leq j \tag{46}$$

For $i > j$ this partial derivative is identically equal to zero. The equations of motion can now be demonstrated in matrix form provided the expression for the partial derivative of the inertia matrix,

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi})\ddot{\boldsymbol{\Phi}} + \dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}} - \frac{1}{2}\dot{\boldsymbol{\Phi}}^T \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \boldsymbol{\Phi}}\dot{\boldsymbol{\Phi}} = \boldsymbol{\tau}. \tag{47}$$

The vector $\boldsymbol{\tau} \in \mathbb{R}^{N_D}$ contains the forces and torques collocated with the generalized coordinates in a space manipulator system. Due to the free-floating nature of the spacecraft's base, the first 6 elements of $\boldsymbol{\tau}$ (corresponding to the base motion) are equal to zero. That is, $\boldsymbol{\tau} = [\boldsymbol{0}_{1\times 6} \;\; \boldsymbol{\tau}_m^T]^T$, where $\boldsymbol{\tau}_m \in \mathbb{R}^N$ is the vector of torques applied to the manipulator joints. The system's generalized velocities $\dot{\boldsymbol{\Phi}}$ can additionally be factored out from the second and third terms in equation (47) to yield

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi})\ddot{\boldsymbol{\Phi}} + \left(\sum_{i=1}^{N_D} \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_i}\dot{\Phi}_i - \frac{1}{2}\begin{bmatrix} \dot{\boldsymbol{\Phi}}^T \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_1} \\ \vdots \\ \dot{\boldsymbol{\Phi}}^T \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_{N_D}}. \end{bmatrix} \right)\dot{\boldsymbol{\Phi}} = \boldsymbol{\tau}. \tag{48}$$

The resulting term multiplying $\dot{\boldsymbol{\Phi}}$ represents the system Coriolis matrix, denoted by $\bar{\boldsymbol{C}}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) \in \mathbb{R}^{N_D \times N_D}$. This matrix contains the space manipulator's Coriolis and centrifugal effects, i.e.,

$$\bar{\boldsymbol{C}}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \sum_{i=1}^{N_D} \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_i}\dot{\Phi}_i - \frac{1}{2}\begin{bmatrix} \dot{\boldsymbol{\Phi}}^T \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_1} \\ \vdots \\ \dot{\boldsymbol{\Phi}}^T \frac{\partial \bar{\boldsymbol{M}}(\boldsymbol{\Phi})}{\partial \Phi_{N_D}}. \end{bmatrix}. \tag{49}$$

In a similar fashion to the inertia matrix partitioning, the system Coriolis matrix can likewise be broken down into its matrix elements. In this case, these sub-matrices demonstrate the Coriolis and centrifugal effects stemming from the motion of the base and that of the manipulator,

$$\bar{\boldsymbol{C}}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = \begin{bmatrix} \boldsymbol{C}_o & \boldsymbol{C}_{om} \\ \boldsymbol{C}_{mo} & \boldsymbol{C}_m \end{bmatrix}. \tag{50}$$

The 6×6 and $6 \times N$ matrices $\boldsymbol{C}_o$ and $\boldsymbol{C}_{om}$ represent the contribution of the Coriolis forces on the total system momentum expressed in the parameterization of the base, resulting from base and manipulator motions, respectively. Additionally, $\boldsymbol{C}_{mo}$ and $\boldsymbol{C}_m$ are $N \times 6$ and $N \times N$ matrices summarizing those contributions in the Coriolis forces on the manipulator. Making use of the system inertia and Coriolis matrices, the equations of motion for a free-floating space manipulator system are written in matrix form as follows,

$$\bar{\boldsymbol{M}}(\boldsymbol{\Phi})\ddot{\boldsymbol{\Phi}} \; + \; \bar{\boldsymbol{C}}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}})\dot{\boldsymbol{\Phi}} = \boldsymbol{\tau}, \tag{51}$$

$$\begin{bmatrix} \boldsymbol{H}_o & \boldsymbol{H}_{om} \\ \boldsymbol{H}_{om}^T & \boldsymbol{H}_m \end{bmatrix}\begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{C}_o & \boldsymbol{C}_{om} \\ \boldsymbol{C}_{mo} & \boldsymbol{C}_m \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0}_{6\times 1} \\ \boldsymbol{\tau}_m \end{bmatrix}. \tag{52}$$

Given these matrix partitionings, the dynamics of a free-floating space manipulator system can be divided into the equations of motion pertaining to the base (i.e., the first six rows of equation (51)) and and those concerning the manipulator (i.e., the last N rows of equation (51)) separately,

$$\boldsymbol{H}_o\ddot{\boldsymbol{\theta}} + \boldsymbol{H}_{om}\ddot{\boldsymbol{q}} + \boldsymbol{C}_o\dot{\boldsymbol{\theta}} + \boldsymbol{C}_{om}\dot{\boldsymbol{q}} = \boldsymbol{0}_{6\times 1} \tag{53}$$

$$\boldsymbol{H}_{om}^T\ddot{\boldsymbol{\theta}} + \boldsymbol{H}_m\ddot{\boldsymbol{q}} + \boldsymbol{C}_{mo}\dot{\boldsymbol{\theta}} + \boldsymbol{C}_m\dot{\boldsymbol{q}} = \boldsymbol{\tau}_m. \tag{54}$$

Separating the equations of motion demonstrates the coupling between the spacecraft base and the manipulator. Equation (54) emphasizes the dynamic influence that an actuation torque applied to the manipulator joints imposes on the base motion. Due to the application of $\boldsymbol{\tau}_m$, the spacecraft base's acceleration is proportional to the coupling inertia sub-matrix $\boldsymbol{H}_{om}^T$ and its velocity related by the coupling component of the Coriolis matrix $\boldsymbol{C}_{mo}$. Thus, having no control of the spacecraft base in the free-floating regime imposes difficulties in controlling the entire space manipulator system due to the coupled motion between the base and the manipulator.

5. Conservation of Momentum

The conservation of linear and angular momentum is considered as an affine nonholonomic constraint to free-floating space manipulator systems. Consequently, defining an expression for the system's conserved momentum allows for a relationship which demonstrates the coupled motion between the base and manipulator. We start by defining the generalized system momentum of space manipulators to be the matrix product of the systems' inertia and their associated generalized coordinate velocities,

$$\boldsymbol{P} := \bar{\boldsymbol{M}}(\boldsymbol{\Phi})\dot{\boldsymbol{\Phi}}. \tag{55}$$

which can be rewritten in terms of the matrices $\boldsymbol{D}$ and $\boldsymbol{J}$ as,

$$\boldsymbol{P} = \boldsymbol{J}^T\boldsymbol{D}\boldsymbol{J}\dot{\boldsymbol{\Phi}}. \tag{56}$$

Expanding the matrices in the generalized system momentum above allows for the momentum associated with each body to be explicitly shown (with reference to its associated body Jacobian and individual inertia matrix),

$$\boldsymbol{P} = [(\boldsymbol{J}_0^b)^T \;\; (\boldsymbol{J}_1^b)^T \; \dots \; (\boldsymbol{J}_N^b)^T]\begin{bmatrix} \boldsymbol{D}_0 & 0 & \dots & 0 \\ 0 & \boldsymbol{D}_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{D}_N \end{bmatrix}\begin{bmatrix} \boldsymbol{J}_0^b \\ \boldsymbol{J}_1^b \\ \vdots \\ \boldsymbol{J}_N^b \end{bmatrix}\dot{\boldsymbol{\Phi}}, \tag{57}$$

which is simplified to

$$\boldsymbol{P} = \begin{bmatrix}(\boldsymbol{J}_0^b)^T & \dots & (\boldsymbol{J}_N^b)^T\end{bmatrix}\begin{bmatrix}\boldsymbol{D}_0\boldsymbol{J}_0^b\dot{\boldsymbol{\Phi}}\\ \boldsymbol{D}_1\boldsymbol{J}_1^b\dot{\boldsymbol{\Phi}}\\ \vdots\\ \boldsymbol{D}_N\boldsymbol{J}_N^b\dot{\boldsymbol{\Phi}}\end{bmatrix}$$
$$= \begin{bmatrix}(\boldsymbol{J}_0^b)^T & \dots & (\boldsymbol{J}_N^b)^T\end{bmatrix}\begin{bmatrix}\boldsymbol{D}_0\boldsymbol{V}_0^b\\ \boldsymbol{D}_1\boldsymbol{V}_1^b\\ \vdots\\ \boldsymbol{D}_N\boldsymbol{V}_N^b\end{bmatrix}. \tag{58}$$

The vector $\boldsymbol{D}_k\boldsymbol{V}_k^b \in \mathbb{R}^6$ represents the body momentum of the k^{th} Body, denoted by $\boldsymbol{P}_k^b$, about Body k's center of mass, expressed in the body coordinate frame. The total momentum of the space manipulator system is simply the summation of the momenta. However, this summation is only valid provided that the individual momenta are expressed with respect to the spatial coordinate frame. That is,

$$\boldsymbol{P}^s = \sum_{k=0}^{N} \boldsymbol{P}_k^s \tag{59}$$

indicates the total spatial momentum of the system $\boldsymbol{P}^s$, when the body momenta expressed in equation (58) are transformed to the spatial coordinate frame. For Body k this spatial momentum is denoted by $\boldsymbol{P}_k^s$ and it is related to the body momentum by

$$\boldsymbol{P}_k^s = \mathbf{Ad}_{g_{sk}}^{-T}\boldsymbol{P}_k^b. \tag{60}$$

As previously discussed in Section 4, the first 6 rows of the inertia matrix correspond to the contributions of the base and manipulator motions on the system momentum expressed in the parameterization of the base. As this section aims to develop a relationship between the base and manipulator motions, only the first 6 rows of the inertia matrix are of concern in the calculation of the total momentum. Based on the inertia matrix definition, the first 6 rows of $\bar{\boldsymbol{M}}$ correspond to the first 6 rows of the $\boldsymbol{J}^T$ matrix, and thus the first 6 columns of $\boldsymbol{J}_0^b$ (i.e. the first 6 rows of $(\boldsymbol{J}_0^b)^T$). Note that the first 6 columns of $\boldsymbol{J}_0^b$ are the only nonzero elements in the base's Jacobian since the base's motion is defined by six degrees of freedom ($l_0 = 6$). We denote the collection of the nonzero columns of the base's body Jacobian by the matrix $\bar{\boldsymbol{J}}_0^b$. That is,

$$\bar{\boldsymbol{J}}_0^b = \mathbf{Ad}_{\boldsymbol{g}_{s0}^{-1}(0)}[\mathbf{Ad}_1^6\boldsymbol{\xi}_1 \;\; \mathbf{Ad}_2^6\boldsymbol{\xi}_2 \;\; \dots \;\; \mathbf{Ad}_6^6\boldsymbol{\xi}_6] \in \mathbb{R}^{6\times 6}. \tag{61}$$

Factoring out the Adjoint operator for the product of exponentials going from the first generalized coordinate to the sixth, i.e. $\mathbf{Ad}_1^6$, and recalling the definition of the spatial Jacobian in (20),

$$\bar{\boldsymbol{J}}_0^b = \mathbf{Ad}_{\boldsymbol{g}_{s0}^{-1}(0)}\mathbf{Ad}_1^6[\boldsymbol{\xi}_1 \;\; \mathbf{Ad}_{e^{\hat{\xi}_1\theta_l}}\boldsymbol{\xi}_2 \;\; \dots \;\; \mathbf{Ad}_{e^{\hat{\xi}_1\theta_l}\dots e^{\hat{\xi}_5\theta_5}}\boldsymbol{\xi}_6]$$
$$=: \mathbf{Ad}_{\boldsymbol{g}_{s0}^{-1}}\bar{\boldsymbol{J}}_0^s \in \mathbb{R}^{6\times 6}. \tag{62}$$

Here, $\bar{\boldsymbol{J}}_0^s$ is comprised of the first 6 columns of the spacecraft base's spatial Jacobian. Since the first 6 columns of every body's body Jacobian matrix corresponds to the same screw motions (however with respect to the coordinate frame attached to the respective Body k) the truncated Jacobian,

$$\bar{\boldsymbol{J}}_k^b := \mathbf{Ad}_{\boldsymbol{g}_{sk}^{-1}}\bar{\boldsymbol{J}}_0^s \in \mathbb{R}^{6\times 6}, \qquad k = 1, \cdots, N \tag{63}$$

forms the first 6 columns of $\boldsymbol{J}_k^b$. Hence, the first 6 rows of equation (58) read,

$$\bar{\boldsymbol{P}} := \begin{bmatrix}(\bar{\boldsymbol{J}}_0^s)^T\mathbf{Ad}_{\boldsymbol{g}_{s0}^{-1}}^T & (\bar{\boldsymbol{J}}_0^s)^T\mathbf{Ad}_{\boldsymbol{g}_{s1}^{-1}}^T & \dots & (\bar{\boldsymbol{J}}_0^s)^T\mathbf{Ad}_{\boldsymbol{g}_{sN}^{-1}}^T\end{bmatrix}\begin{bmatrix}\boldsymbol{P}_0^b\\ \boldsymbol{P}_1^b\\ \vdots\\ \boldsymbol{P}_N^b\end{bmatrix}$$
$$= (\bar{\boldsymbol{J}}_0^s)^T(\mathbf{Ad}_{\boldsymbol{g}_{s0}^{-1}}^T\boldsymbol{P}_0^b + \mathbf{Ad}_{\boldsymbol{g}_{s1}^{-1}}^T\boldsymbol{P}_1^b + \dots + \mathbf{Ad}_{\boldsymbol{g}_{sN}^{-1}}^T\boldsymbol{P}_N^b)$$
$$= (\bar{\boldsymbol{J}}_0^s)^T\sum_{k=0}^{N}\boldsymbol{P}_k^s$$
$$= (\bar{\boldsymbol{J}}_0^s)^T\boldsymbol{P}^s. \tag{64}$$

Referring back to the partitioning of the inertia matrix $\bar{\boldsymbol{M}}(\boldsymbol{\Phi})$ in equation (39), the total spatial momentum of the system is obtained by,

$$\boldsymbol{P}^s(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}) = (\bar{\boldsymbol{J}}_0^s)^{-T}\bar{\boldsymbol{P}} = (\bar{\boldsymbol{J}}_0^s)^{-T}\begin{bmatrix}\boldsymbol{H}_o & \boldsymbol{H}_{om}\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{q}}\end{bmatrix}. \tag{65}$$

Under the aforementioned assumption that the space manipulator system is in an undisturbed environment, the total momentum of the system in the spatial coordinate frame is assumed constant and equal to $\boldsymbol{\mu} \in \mathbb{R}^6$. Therefore, the conservation of momentum equation yields

$$\boldsymbol{H}_o\dot{\boldsymbol{\theta}} + \boldsymbol{H}_{om}\dot{\boldsymbol{q}} = (\bar{\boldsymbol{J}}_0^s)^T\boldsymbol{\mu}. \tag{66}$$

This equation may be considered as an affine nonholonomic constraint on the space manipulator motion. We denote the submanifold of the tangent bundle of the configuration manifold that is defined via this equation by

$$\mathcal{M} := (\boldsymbol{P}^s)^{-1}(\boldsymbol{\mu}). \tag{67}$$

This defines all possible states of the system in which the system can evolve in when the total momentum of the system is conserved and equal to $\boldsymbol{\mu}$. Note that the case of $\boldsymbol{\mu} = 0$ has been extensively studied in the literature [45]. However, the system may not be in rest (either deliberately or non-deliberately) when beginning the capture process of a target object; thus signifying the importance of studying control scenarios involving systems with conserved nonzero momentum.

6. Dynamic Reduction

Free-floating space manipulators are inherently under-actuated with available control only in the joint space of the manipulator. Taking advantage of the conserved quantities (the total system momentum), a systematic procedure is developed to reduce the system dynamics at a non-zero momentum by restricting to the space of allowable velocities. This implies developing dynamical equations which are independent of the uncontrolled base motion, and exclusively consider the manipulator torque and the motion of the manipulator joints (i.e., the space of control actions and controlled states). The following dynamic reduction is predicated on the assumption of an undisturbed environment, signifying that the system momentum is constant, however not necessarily zero. We begin the reduction process by using the conserved linear and angular momentum presented in Section 5 to restrict the system velocities. From this conserved quantity, a non-linear relationship between the base and manipulator motions arises. Recalling the conservation of total momentum in equation (66), this relation is obtained by isolating for the base velocity as follows,

$$\begin{aligned}\dot{\boldsymbol{\theta}} &= \boldsymbol{H}_o^{-1}(\bar{\boldsymbol{J}}_0^s)^T\boldsymbol{\mu} - \boldsymbol{H}_o^{-1}\boldsymbol{H}_{om}\dot{\boldsymbol{q}} \\ &= \boldsymbol{B} - \boldsymbol{S}\dot{\boldsymbol{q}},\end{aligned} \tag{68}$$

where we define:

$$\boldsymbol{S}(\boldsymbol{\Phi}) := \boldsymbol{H}_o^{-1}\boldsymbol{H}_{om}, \tag{69}$$

$$\boldsymbol{B}(\boldsymbol{\Phi};\boldsymbol{\mu}) := \boldsymbol{H}_o^{-1}(\bar{\boldsymbol{J}}_0^s)^T\boldsymbol{\mu}. \tag{70}$$

With the objective of the dynamic reduction being to remove all dependency on the base motion in the dynamic equations, the base acceleration (appearing in equation (54)) must also be expressed in terms of the manipulator joint accelerations. This implies taking the time derivative of the relationship between the base and manipulator joint velocities in equation (68) to yield the following relationship,

$$\ddot{\boldsymbol{\theta}} = \dot{\boldsymbol{B}} - \dot{\boldsymbol{S}}\dot{\boldsymbol{q}} - \boldsymbol{S}\ddot{\boldsymbol{q}}, \tag{71}$$

where the time derivatives $\dot{\boldsymbol{S}}$ and $\dot{\boldsymbol{B}}$ are defined as,

$$\dot{\boldsymbol{S}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}}) = \boldsymbol{H}_o^{-1}\dot{\boldsymbol{H}}_{om} - \boldsymbol{H}_o^{-1}\dot{\boldsymbol{H}}_o\boldsymbol{H}_o^{-1}\boldsymbol{H}_{om}, \tag{72}$$

$$\dot{\boldsymbol{B}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}};\boldsymbol{\mu}) = -\boldsymbol{H}_o^{-1}\dot{\boldsymbol{H}}_o\boldsymbol{H}_o^{-1}(\bar{\boldsymbol{J}}_0^s)^T\boldsymbol{\mu} + \boldsymbol{H}_o^{-1}(\dot{\bar{\boldsymbol{J}}}_0^s)^T\boldsymbol{\mu}. \tag{73}$$

Similar to the partitioning introduced in equation (39), the time derivative of the inertia matrix components $\dot{\boldsymbol{H}}_{om}$ and $\dot{\boldsymbol{H}}_o$ are defined by partitioning the time derivative of the system inertia matrix $\dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}})$,

$$\dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}}) = \begin{bmatrix}\dot{\boldsymbol{H}}_o & \dot{\boldsymbol{H}}_{om}\\ \dot{\boldsymbol{H}}_{om}^T & \dot{\boldsymbol{H}}_m\end{bmatrix}. \tag{74}$$

The matrix $\dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}})$ is calculated based on equation (42), as detailed in Section 4. Referring to the matrix $\dot{\boldsymbol{B}}$, the time derivative of the truncated spatial Jacobian for the base must be determined. A similar computation presented in equation (46) is required to calculate $\dot{\bar{\boldsymbol{J}}}_0^s$, however for a Jacobian matrix in the spatial frame. Based on the definition of the spatial Jacobian in equation (20), the partial derivative of the j^{th} column of $\bar{\boldsymbol{J}}_0^s$ with respect to the i^{th} generalized coordinate Φ_i is calculated by the following,

$$\frac{\partial \bar{\boldsymbol{J}}_{0,j}^s}{\partial \Phi_i} = \frac{\partial(\mathbf{Ad}_1^{j-1})'}{\partial \Phi_i}\boldsymbol{\xi}_j, \tag{75}$$

where,

$$\frac{\partial(\mathbf{Ad}_1^{j-1})'}{\partial \Phi_i} = (\mathbf{Ad}_1^{i})'\mathbf{ad}_{\boldsymbol{\xi}_i}(\mathbf{Ad}_{i+1}^{j-1})'. \quad 1 \le i \le (j-1) \tag{76}$$

Appendix B provides the full derivation for the partial derivative of the Adjoint operator $(\mathbf{Ad}_1^{j-1})'$ in equation (76) above. For any $i > (j-1)$ this derivative is identically equal to zero. Consequently, the expression for the partial derivative of the column $\bar{\boldsymbol{J}}_{0,j}^s$ with respect to Φ_i becomes,

$$\frac{\partial \bar{\boldsymbol{J}}_{0,j}^s}{\partial \Phi_i} = (\mathbf{Ad}_1^{i})'\mathbf{ad}_{\boldsymbol{\xi}_i}((\mathbf{Ad}_{i+1}^{j-1})'\boldsymbol{\xi}_j). \tag{77}$$

Since the first column of the spatial Jacobian expresses the unchanged twist $\boldsymbol{\xi}_1$, and the spacecraft base contains 6 degrees of freedom, equation (77) applies to all columns $2 \le j \le 6$. Consequently, the time derivative $\dot{\bar{\boldsymbol{J}}}_0^s$ is computed as follows,

$$\dot{\bar{\boldsymbol{J}}}_0^s(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}}) = \sum_{i=2}^{6}\frac{\partial \bar{\boldsymbol{J}}_0^s}{\partial \Phi_i}\dot{\Phi}_i. \tag{78}$$

Based on the matrices $\dot{\bar{\boldsymbol{M}}}(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}})$ and $\dot{\bar{\boldsymbol{J}}}_0^s(\boldsymbol{\Phi},\dot{\boldsymbol{\Phi}})$, a dependency on the base velocities is introduced in the relationship between the base and manipulator accelerations. Such dependency on $\dot{\boldsymbol{\theta}}$ is removed through restricting to $\mathcal{M}$ by implementing equation (68) to express the base velocity with that of the manipulator in $\dot{\boldsymbol{\Phi}}$. The same dependency on $\dot{\boldsymbol{\theta}}$ in the system Coriolis matrix can analogously be removed by restricting to $\mathcal{M}$. That is, we define the following matrices to be independent of the base velocity,

$$\mathbb{S}(\boldsymbol{\Phi},\dot{\boldsymbol{q}};\mu) := \dot{\boldsymbol{S}}(\boldsymbol{\Phi},[(\boldsymbol{B}-\boldsymbol{S}\dot{\boldsymbol{q}})^T\ \dot{\boldsymbol{q}}^T]^T), \tag{79}$$

$$\mathbb{B}(\boldsymbol{\Phi},\dot{\boldsymbol{q}};\mu) := \dot{\boldsymbol{B}}(\boldsymbol{\Phi},[(\boldsymbol{B}-\boldsymbol{S}\dot{\boldsymbol{q}})^T\ \dot{\boldsymbol{q}}^T]^T), \tag{80}$$

$$\mathbb{C}(\boldsymbol{\Phi},\dot{\boldsymbol{q}};\mu) := \bar{\boldsymbol{C}}(\boldsymbol{\Phi},[(\boldsymbol{B}-\boldsymbol{S}\dot{\boldsymbol{q}})^T\ \dot{\boldsymbol{q}}^T]^T), \tag{81}$$

where the matrix $\mathbb{C}(\boldsymbol{\Phi},\dot{\boldsymbol{q}};\mu)$ is partitioned as follows,

$$\mathbb{C}(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu) = \begin{bmatrix} \mathbb{C}_o & \mathbb{C}_{om} \\ \mathbb{C}_{mo} & \mathbb{C}_m \end{bmatrix}. \tag{82}$$

Note that the equations of motion pertaining to the base in equation (53) are equivalent to the derived relationship between the base and manipulator accelerations stemming from the conservation of momentum. Hence by restricting the dynamics to $\mathcal{M}$, we can omit these equations from the set of dynamical equations of the space manipulator. The equations of motion governing the motion of the manipulator can then be obtained by restricting equation (54) to $\mathcal{M}$. Substituting the relationships from the conservation of momentum in equations (68) and (71) into equation (54) yields the following reduced set of dynamical equations independent of the base motion:

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{E} = \boldsymbol{\tau}_m, \tag{83}$$

where,

$$\boldsymbol{M}(\boldsymbol{\Phi}) := \boldsymbol{H}_m - \boldsymbol{H}_{om}^T \boldsymbol{S}, \tag{84}$$

$$\boldsymbol{C}(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu) := \mathbb{C}_m - \boldsymbol{H}_{om}^T \mathbb{S} - \mathbb{C}_{mo} \boldsymbol{S}, \tag{85}$$

$$\boldsymbol{E}(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu) := \boldsymbol{H}_{om}^T \mathbb{B} + \mathbb{C}_{mo} \boldsymbol{B}. \tag{86}$$

The reduced equations of motion in equation (83) express the relationship between internal torques applied to the manipulator's joints and the resulting joint motions considering the contributions of the base motion. As mentioned, the resulting equations are independent of the base motion (i.e., $\dot{\boldsymbol{\theta}}$ and $\ddot{\boldsymbol{\theta}}$), however still depend on the base configuration $\boldsymbol{\theta}$. The base configuration is a measurable quantity of a space manipulator system and therefore does not need to be expressed in terms of the position of the manipulator joints. The following section develops a input-output feedback linearization law for free-floating manipulators, using the reduced dynamical equations to derive an appropriate control action (in the manipulator joint space) which linearizes the end-effector's motion in SE(3).

7. Input-Output Feedback Linearization

Let us define the output of the system to be the pose of the end-effector $\boldsymbol{g}_{sN}$ and consider free-floating space manipulator systems with constant non-zero momentum. The objective of feedback linearization in this instance is to first obtain an input-output relationship (restricted to $\mathcal{M}$) between the manipulator torque $\boldsymbol{\tau}_m$ and the motion of the output in SE(3). A linearization law is subsequently derived such that all non-linearities are removed in the end-effector's motion. This process leads to the following control problem addressed in this paper:

Problem 1. *Consider the free-floating space manipulator dynamics in* (53) *and* (54)*, and the position and orientation of the end-effector, i.e.,* $\boldsymbol{g}_{sN}$ *defined in* (14)*, as the output of the system. Given a twice differentiable desired trajectory* $\bar{\boldsymbol{g}}_{sN}(t) \in SE(3)$ *for the output, find* $\boldsymbol{\tau}_m$ *to make the desired output trajectory exponentially stable.*

We begin deriving the relationship between the manipulator torques and end-effector motion by establishing the body velocity of the end-effector based on the system's body Jacobian matrix, as shown in equation (23). The Jacobian matrix $\boldsymbol{J}_N^b$ may be partitioned by columns to explicitly distinguish the contributions on the end-effector's linear and angular velocity from the spacecraft base and manipulator configurations. Consequently, the body Jacobian matrix is expanded to incorporate a 6×6 sub-matrix including all columns pertaining to the base denoted $\boldsymbol{J}_o^b$, and a $6 \times N_{D_M}$ sub-matrix including the remaining columns corresponding to the manipulator denoted $\boldsymbol{J}_m^b$,

$$\boldsymbol{V}_N^b = [\boldsymbol{J}_o^b \quad \boldsymbol{J}_m^b] \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{q}} \end{bmatrix} \in \mathbb{R}^6. \tag{87}$$

To express the end-effector's linear velocity in terms of only the controlled states, we restrict the system velocity vector to $\mathcal{M}$ and remove the dependency on the base motion,

$$\boldsymbol{V}_N^b = [\boldsymbol{J}_o^b \quad \boldsymbol{J}_m^b] \begin{bmatrix} \boldsymbol{B} - \boldsymbol{S}\dot{\boldsymbol{q}} \\ \dot{\boldsymbol{q}} \end{bmatrix}. \tag{88}$$

Performing the matrix multiplication above explicitly expresses the $6 \times N_{D_M}$ Generalized Jacobian Matrix (GJM) denoted $\boldsymbol{J}^*$,

$$\boldsymbol{V}_N^b = \boldsymbol{J}^* \dot{\boldsymbol{q}} + \boldsymbol{J}_o^b \boldsymbol{B}, \tag{89}$$

$$\boldsymbol{J}^* = \boldsymbol{J}_m^b - \boldsymbol{J}_o^b \boldsymbol{S} \quad \in \mathbb{R}^{6 \times N_{D_M}}, \tag{90}$$

which provides a mapping from the manipulator's joint space in $\mathbb{R}^{N_{D_M}}$ to the system's workspace in SE(3), without parameterizing the output at any point. The GJM expressed in equation (90) is an extension of the original GJM definition in [45] which includes a general Jacobian matrix mapping from $\mathbb{R}^n \to \mathbb{R}^n$. Note that the end-effector velocity is additionally influenced by the total system momentum through the parameter $\boldsymbol{B}$. Taking the time derivative of the output's body velocity in equation (89) yields the desired relationship between the end-effector's motion and the joint accelerations,

$$\dot{\boldsymbol{V}}_N^b = \boldsymbol{J}^* \ddot{\boldsymbol{q}} + \mathbb{J}^* \dot{\boldsymbol{q}} + \mathbb{J}_o \boldsymbol{B} + \boldsymbol{J}_o \mathbb{B}, \tag{91}$$

where $\mathbb{J}^*(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu)$ and $\mathbb{J}_o(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu)$ denote restrictions of the time derivatives $\dot{\boldsymbol{J}}^*$ and $\dot{\boldsymbol{J}}_o$ to $\mathcal{M}$, defined in the same manner as (79)-(81):

$$\mathbb{J}^*(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu) := \dot{\boldsymbol{J}}^*(\boldsymbol{\Phi}, [(\boldsymbol{B} - \boldsymbol{S}\dot{\boldsymbol{q}})^T \ \dot{\boldsymbol{q}}^T]^T) \in \mathbb{R}^{6 \times N_{D_M}}, \tag{92}$$

$$\mathbb{J}_o(\boldsymbol{\Phi}, \dot{\boldsymbol{q}}; \mu) := \dot{\boldsymbol{J}}_o(\boldsymbol{\Phi}, [(\boldsymbol{B} - \boldsymbol{S}\dot{\boldsymbol{q}})^T \ \dot{\boldsymbol{q}}^T]^T) \in \mathbb{R}^{6 \times 6}. \tag{93}$$

Substituting $\ddot{\boldsymbol{q}}$ from the system's reduced dynamics in equation (83) yields the following relationship between the end-effector's motion and the joint torques,

$$\dot{\boldsymbol{V}}_N^b = \boldsymbol{J}^* \boldsymbol{M}^{-1}(\boldsymbol{\tau}_m - \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{E}) + \mathbb{J}^*\dot{\boldsymbol{q}} + \mathbb{J}_o \boldsymbol{B} + \boldsymbol{J}_o \mathbb{B}. \tag{94}$$

Feedback linearization can now be implemented to linearize the system's output by defining an appropriate input torque $\boldsymbol{\tau}_m$ which cancels the non-linearities in equation (94). Note that there is assumed to be sufficient dexterity in the space manipulator to fully actuate the system's output for control in SE(3). This implies an ability to control six independent motions of the end-effector, leading to the following assumption:

Assumption 1. *The manipulator system contains actuators imparting at least six independent motions.*

Given the relationship between $\boldsymbol{\tau}_m$ and $\dot{\boldsymbol{V}}_N^b$ in equation (94), and under Assumption 1, the following feedback control law:

$$\boldsymbol{\tau}_m = \boldsymbol{M}\boldsymbol{J}^{*\diamond}(\boldsymbol{U} - \mathbb{J}^*\dot{\boldsymbol{q}} - \mathbb{J}_o\boldsymbol{B} - \boldsymbol{J}_o\mathbb{B}) + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{E}, \tag{95}$$

linearizes the input-output relationship between the linear and angular end-effector motion and the output tracking controller $\boldsymbol{U}$ for control in SE(3),

$$\dot{\boldsymbol{V}}_N^b = \boldsymbol{U}. \tag{96}$$

Here, $\boldsymbol{U} \in \mathbb{R}^6$ is the vector of control input in the resulting closed-loop control system, outlined in the following section. Consequently, the motion of the end-effector is fully described by the control forces and torques imparted by $\boldsymbol{U}$. Moreover the $\diamond$ operator signifies the Singularity-Robust (SR) inverse of a matrix, defined for non-square matrices. The SR-inverse is implemented as a general case of Assumption 1, pertaining to redundant manipulator designs. For a manipulator designed with actuation in exactly six independent directions, $\boldsymbol{J}^*$ becomes a 6×6 square matrix and the natural matrix inverse can be performed (i.e., equation (95) includes $\boldsymbol{J}^{*-1}$ in place of $\boldsymbol{J}^{*\diamond}$). Details regarding the formulation of the SR-inverse are included in Section 9.

8. Full Pose Control Structure

In this section, a control law on the SE(3) Lie group is proposed, stabilizing any twice differentiable feasible trajectory $\bar{\boldsymbol{g}}_{sN}(t) \in \text{SE(3)}$ of a space manipulator's end-effector. This output tracking controller builds upon a PID control design by implementing an additional feedforward component to achieve Lyapunov stability. This section additionally defines a configuration error function and a velocity error associated with the system's output, necessary for developing the modified PID control structure. The proposed control law is proven to stabilize the full end-effector pose toward a feasible trajectory using a Lyapunov function based on the total energy of the error dynamics.

8.1. Error Function

A positive definite error function is used to establish the configuration difference between the desired and actual end-effector trajectories. In this case, the error function is thus referred to as a configuration error function. This paper considers a quadratic error function going from $\text{SE(3)} \to \mathbb{R}_{\geq 0}$, a positive scalar. Using group operation on the desired and actual end-effector configurations in SE(3), the output pose error $\boldsymbol{g}_e$ is defined as follows,

$$\boldsymbol{g}_e = \bar{\boldsymbol{g}}_{sN}^{-1}\boldsymbol{g}_{sN} := \begin{bmatrix} \boldsymbol{R}_e & \boldsymbol{p}_e \\ \boldsymbol{0} & 1 \end{bmatrix} \in \text{SE(3)}. \tag{97}$$

This representation of the error expresses the relative configuration of the actual end-effector with respect to the desired trajectory at an instance in time. Note that in practice, the end-effector's actual pose is estimated using the forward kinematics mapping presented in Section 3 provided sensor measurements of the system's generalized coordinates. As a member of SE(3), the pose error contains components $(\boldsymbol{R}_e, \boldsymbol{p}_e)$ indicating the orientation and position errors respectively. The corresponding orientation error is defined from group operation on the desired and actual end-effector orientations ($\bar{\boldsymbol{R}}_{sN}$ and $\boldsymbol{R}_{sN}$) and the position error is simply the Euclidean distance between the desired and actual end-effector positions ($\bar{\boldsymbol{p}}_{sN}$ and $\boldsymbol{p}_{sN}$),

$$\boldsymbol{R}_e = \bar{\boldsymbol{R}}_{sN}^T \boldsymbol{R}_{sN} \in \text{SO(3)}, \tag{98}$$

$$\boldsymbol{p}_e = \bar{\boldsymbol{R}}_{sN}^T(\boldsymbol{p}_{sN} - \bar{\boldsymbol{p}}_{sN}) \in \mathbb{R}^3. \tag{99}$$

As mentioned, the configuration error function maps the group error $\boldsymbol{g}_e$ to a positive scalar. Let $\psi : \text{SE(3)} \to \mathbb{R}_{\geq 0}$ be the error function which considers both orientation and position errors. We define the function $\psi(\boldsymbol{g}_e)$ as

$$\psi(\boldsymbol{g}_e) = \psi_1(\boldsymbol{p}_e) + \psi_2(\boldsymbol{R}_e), \tag{100}$$

based on the two positive definite error functions $\psi_1 : \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ and $\psi_2 : \text{SO(3)} \to \mathbb{R}_{\geq 0}$ that are specified in the following. The positive definite error function associated with the position error is trivially defined as the quadratic norm of $\boldsymbol{p}_e$,

$$\psi_1(\boldsymbol{p}_e) = \frac{1}{2}||\boldsymbol{p}_e||^2 = \frac{1}{2}||\boldsymbol{p}_{sN} - \bar{\boldsymbol{p}}_{sN}||^2, \tag{101}$$

and the positive definite error function on the orientation is defined by Koditschek in [23] as follows,

$$\psi_2(\boldsymbol{R}_e) = \frac{1}{2}\text{tr}(\boldsymbol{1}_{3\times 3} - \boldsymbol{R}_e), \tag{102}$$

where $\text{tr}(\boldsymbol{A})$ refers to the trace of a square matrix $\boldsymbol{A}$. Combining the position and orientation error functions in equations (101) and (102) yields the following positive

definite error function associated with the group element $\boldsymbol{g}_e$ considered here,

$$\psi(\boldsymbol{g}_e) = \frac{1}{2}\text{tr}(\mathbf{1}_{3\times3} - \boldsymbol{R}_e) + \frac{1}{2}||\boldsymbol{p}_e||^2. \tag{103}$$

This configuration error becomes crucial in designing the proportional and integral control actions in the output tracking control structure. In the following subsection, a compatible output velocity error is defined for the subsequent formation of the controller's derivative action. Note that for simplicity in notation, we refer to the error function on $\boldsymbol{g}_e$ as simply ψ.

8.2. Velocity Error

We begin by defining the body velocity associated with the desired trajectory $\bar{\boldsymbol{V}}^b_{sN}$. From the kinematics of the end-effector, we define the reference body velocity of the output as $\hat{\bar{\boldsymbol{V}}}^b_{sN} = \bar{\boldsymbol{g}}^{-1}_{sN}\dot{\bar{\boldsymbol{g}}}_{sN} \in \mathfrak{se}(3)$, which expresses the desired body velocity in the coordinate frame of the desired trajectory. In comparing this desired end-effector velocity with the true motion of the end-effector, the quantities $\bar{\boldsymbol{V}}^b_{sN}$ and $\boldsymbol{V}^b_{sN}$ must be expressed in a consistent frame of reference to quantify the error. This introduces a transport map in the definition of the velocity error to project the desired body velocity into the coordinate frame attached to the end-effector.

Recall that the Adjoint operator transforms elements of $\mathfrak{se}(3)$ between two coordinate frames provided their relative homogeneous transformation. Thus, the Adjoint operator provides the necessary transport map to convert $\bar{\boldsymbol{V}}^b_{sN}$ into the frame of reference of the end-effector. Since the group error $\boldsymbol{g}_e$ represents the relative motion from the end-effector's frame to the desired trajectory frame, taking its inverse defines the required transformation from the desired trajectory to the output's body frame. Taking the Adjoint of $\boldsymbol{g}^{-1}_e = \boldsymbol{g}^{-1}_{sN}\bar{\boldsymbol{g}}_{sN}$ and pre-multiplying with $\bar{\boldsymbol{V}}^b_{sN}$ now expresses the actual and desired body velocities in a shared reference frame. The velocity error $\boldsymbol{V}_e$ compatible with the group error $\boldsymbol{g}_e$ can now be defined as follows,

$$\boldsymbol{V}^b_e = \boldsymbol{V}^b_{sN} - \mathbf{Ad}_{\boldsymbol{g}^{-1}_e}\bar{\boldsymbol{V}}^b_{sN} \in \mathbb{R}^6. \tag{104}$$

Evidently, the body velocity error is similarly expressed with respect to the end-effector's body coordinate frame. With the definitions of the configuration error function in (103) and the velocity error associated with the output above, the following subsection introduces the output tracking control structure to drive $\boldsymbol{g}_e$ to the identity and $\boldsymbol{V}^b_e$ to zero.

8.3. Output Tracking Control Design

The developed control law is predicated on the combination of feedback terms and a feedforward term to satisfy the stability in the output tracking control task outlined in Problem 1. The feedback terms involve proportional, integral, and derivative control actions which make use of the coordinate-free error function and velocity error definitions in the previous subsections. The modified feedforward, feedback PID control law thus contains the following actions to control the end-effector motion in SE(3),

$$\boldsymbol{U} = \boldsymbol{U}_{pi} + \boldsymbol{U}_d + \boldsymbol{U}_{ff}. \tag{105}$$

Given a 6×6 symmetric positive definite matrix $\boldsymbol{K}_d$ expressing the derivative gain, the control action to dissipate the velocity error to zero is defined as follows,

$$\boldsymbol{U}_d = -\boldsymbol{K}_d\boldsymbol{V}^b_e = -\boldsymbol{K}_d(\boldsymbol{V}^b_{sN} - \mathbf{Ad}_{\boldsymbol{g}^{-1}_e}\bar{\boldsymbol{V}}^b_{sN}). \tag{106}$$

The 6×6 symmetric positive definite proportional and integral gains $\boldsymbol{K}_p$ and $\boldsymbol{K}_i$ are applied to drive the output's configuration error function ψ to zero, and accordingly $\boldsymbol{g}_e$ to the identity. In defining the proportional and integral control actions, the gradient of the error function in (103) must first be defined. This gradient $\nabla\psi$ becomes apparent from the time derivative of the error function which can be computed as follows,

$$\dot{\psi} = (\nabla\psi)^T\boldsymbol{V}^b_e := \begin{bmatrix}\boldsymbol{p}_e \\ \text{skew}(\boldsymbol{R}_e)^\vee\end{bmatrix}^T\boldsymbol{V}^b_e, \tag{107}$$

where $\text{skew}(\boldsymbol{A}) = \frac{1}{2}(\boldsymbol{A} - \boldsymbol{A}^T)$ for all 3×3 matrices $\boldsymbol{A}$, and the operator $\vee$ converts skew symmetric matrices to their vector representation (i.e., $\vee : \mathfrak{so}(3) \to \mathbb{R}^3$). The definition of the error function's gradient is demonstrated in the expression for $\dot{\psi}$ above, using the gradient definition on Riemannian manifolds [21]. The proportional and integral control actions are predicated on the gradient of the error function as follows,

$$\boldsymbol{U}_{pi} = -\boldsymbol{K}_p\nabla\psi - \boldsymbol{K}_i\boldsymbol{F}_i, \tag{108}$$

$$\dot{\boldsymbol{F}}_i = \boldsymbol{K}_p\nabla\psi + \boldsymbol{K}_d\boldsymbol{V}^b_e. \tag{109}$$

As indicated by equation (109), $\boldsymbol{F}_i$ refers to the integral of the proportional and derivative actions. The feedforward component of the output tracking control law is designed in a manner which aids in the stability of the closed loop system. The impact of the feedforward term will become apparent in the proof of Theorem 1. Based on the configuration and velocity error definitions previously defined, an appropriate structure for the output tracking feedforward action is as follows,

$$\boldsymbol{U}_{ff} = \mathbf{ad}_{\boldsymbol{V}^b_{sN}}\mathbf{Ad}_{\boldsymbol{g}^{-1}_e}\bar{\boldsymbol{V}}^b_{sN} + \mathbf{Ad}_{\boldsymbol{g}^{-1}_e}\dot{\bar{\boldsymbol{V}}}^b_{sN}. \tag{110}$$

The feedforward control action is necessary for performing trajectory following tasks, and defines the desired acceleration associated with the reference trajectory, expressed with respect to the body coordinate frame attached to the end-effector. Note that a stability analysis on the internal

dynamics of the considered free-floating space manipulators is outside the scope of this paper, leading to the following assumption:

Assumption 2. *The internal dynamics of the free-floating space manipulator system remains bounded if the output of the system $\boldsymbol{g}_{sN}$ is stable.*

Theorem 1. *Consider the dynamics of a free-floating space manipulator's end-effector in (96) and the position and orientation of the end-effector, i.e., $\boldsymbol{g}_{sN} \in SE(3)$, in (15) as the output of the system. Assume that the system is externally unperturbed and its total linear and angular momentum is conserved and equal to $\boldsymbol{\mu} \in \mathbb{R}^6$. Let $\bar{\boldsymbol{g}}_{sN}(t) \in SE(3)$ be a twice differentiable feasible trajectory of the end-effector of the space manipulator.*

Provided the modified feedforward, feedback PID control law defined in equations (106)-(110) in addition to the feedback linearization law in (95), and under Assumptions 1 and 2, the desired trajectory $\bar{\boldsymbol{g}}_{sN}(t)$ is Lyapunov stable provided the following Lyapunov function $V_L : \mathbb{R} \to \mathbb{R}_{\geq 0}$ based on the total energy of the error dynamics:

$$V_L(t) = \psi + \frac{1}{2}||\boldsymbol{V}_e^b||^2_{\boldsymbol{K}_p^{-1}} + \frac{1}{2}||\boldsymbol{F}_i + \boldsymbol{V}_e^b||^2_{\boldsymbol{K}_p^{-1}}, \tag{111}$$

where the quadratic norm of a vector $\boldsymbol{V} \in \mathbb{R}^n$ subject to the metric $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ is defined as,

$$||\boldsymbol{V}||^2_{\boldsymbol{K}} = \boldsymbol{V}^T\boldsymbol{K}\boldsymbol{V}. \tag{112}$$

Proof. We prove the Lyapunov stability of the proposed output tracking control law by demonstrating that the time derivative of the Lyapunov candidate in (111) along the trajectories of the system is negative semidefinite. In this process, the time derivatives of the three terms in (111) are analyzed separately. Recall that the time derivative of the error function has been previously established in equation (107) to produce an expression for $\dot{\psi}$. Thus, we begin the derivation process by taking the time derivative of the second term in (111) which expresses the kinetic energy of the error dynamics. Based on the vector norm definition in (112) and symmetry,

$$\frac{d}{dt}\left(\frac{1}{2}||\boldsymbol{V}_e^b||^2_{\boldsymbol{K}_p^{-1}}\right) = (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{V}}_e^b. \tag{113}$$

The time derivative of the velocity error makes use of the linearized end-effector dynamics in (96), and demonstrates the significance of the feedforward control action's structure. We perform the derivative of the velocity error in (104) as follows,

$$\begin{aligned}
\dot{\boldsymbol{V}}_e^b &= \frac{d}{dt}\left(\boldsymbol{V}_{sN}^b - \mathbf{Ad}_{\boldsymbol{g}_e^{-1}}\bar{\boldsymbol{V}}_{sN}^b\right)\\
&= \dot{\boldsymbol{V}}_{sN}^b - \frac{d}{dt}\left(\mathbf{Ad}_{\boldsymbol{g}_e^{-1}}\right)\bar{\boldsymbol{V}}_{sN}^b - \mathbf{Ad}_{\boldsymbol{g}_e^{-1}}\dot{\bar{\boldsymbol{V}}}_{sN}^b\\
&= (\boldsymbol{U}_{ff} + \boldsymbol{U}_{pi} + \boldsymbol{U}_d) - \mathbf{ad}_{\boldsymbol{V}_{sN}^b}\mathbf{Ad}_{\boldsymbol{g}_e^{-1}}\bar{\boldsymbol{V}}_{sN}^b - \mathbf{Ad}_{\boldsymbol{g}_e^{-1}}\dot{\bar{\boldsymbol{V}}}_{sN}^b\\
&= \boldsymbol{U}_{pi} + \boldsymbol{U}_d.
\end{aligned} \tag{114}$$

Moreover, the time derivative of the third term in (111) is computed by virtue of the chain rule,

$$\begin{aligned}
\frac{d}{dt}\left(\frac{1}{2}||\boldsymbol{F}_i + \boldsymbol{V}_e^b||^2_{\boldsymbol{K}_p^{-1}}\right) &= (\boldsymbol{F}_i + \boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}(\dot{\boldsymbol{F}}_i + \dot{\boldsymbol{V}}_e^b)\\
&= \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{F}}_i + \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{V}}_e^b +\\
&\quad (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{F}}_i + (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{V}}_e^b,
\end{aligned} \tag{115}$$

which can be simplified by including the expressions for $\dot{\boldsymbol{F}}_i$ and $\dot{\boldsymbol{V}}_e^b$ from equations (109) and (114). We analyze each of the four terms above separately in the following:

$$\begin{aligned}
\boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{F}}_i &= \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}(\boldsymbol{K}_p\nabla\psi + \boldsymbol{K}_d\boldsymbol{V}_e^b)\\
&= \boldsymbol{F}_i\nabla\psi + \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b,
\end{aligned} \tag{116}$$

$$\begin{aligned}
\boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{V}}_e^b &= \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}(\boldsymbol{U}_{pi} + \boldsymbol{U}_d)\\
&= \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}(-\boldsymbol{K}_p\nabla\psi - \boldsymbol{K}_i\boldsymbol{F}_i - \boldsymbol{K}_d\boldsymbol{V}_e^b)\\
&= -\boldsymbol{F}_i^T\nabla\psi - \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i -\\
&\quad \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b,
\end{aligned} \tag{117}$$

$$(\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{F}}_i = (\boldsymbol{V}_e^b)^T\nabla\psi + (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b, \tag{118}$$

$$\begin{aligned}
(\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\dot{\boldsymbol{V}}_e^b &= -(\boldsymbol{V}_e^b)^T\nabla\psi - (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i -\\
&\quad (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b.
\end{aligned} \tag{119}$$

Implementing the expanded interpretations above for each term in (115) yields,

$$\frac{d}{dt}\left(\frac{1}{2}||\boldsymbol{F}_i + \boldsymbol{V}_e^b||^2_{\boldsymbol{K}_p^{-1}}\right) = -\boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i - (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i \tag{120}$$

Combining the derivatives in equations (107), (113) and (120), we derive the expression for $\dot{V}_L$ as follows,

$$\begin{aligned}
\dot{V}_L &= (\nabla\psi)^T\boldsymbol{V}_e^b + (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}(\boldsymbol{U}_{pi} + \boldsymbol{U}_b) + (-\boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i -\\
&\quad (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i)\\
&= (\nabla\psi)^T\boldsymbol{V}_e^b + (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}(-\boldsymbol{K}_p\nabla\psi - \boldsymbol{K}_i\boldsymbol{F}_i - \boldsymbol{K}_d\boldsymbol{V}_e^b) -\\
&\quad \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i - (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i\\
&= -(\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i - (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b - \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i\\
&\quad - (\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i\\
&= -(\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_d\boldsymbol{V}_e^b - 2(\boldsymbol{V}_e^b)^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i - \boldsymbol{F}_i^T\boldsymbol{K}_p^{-1}\boldsymbol{K}_i\boldsymbol{F}_i,
\end{aligned} \tag{121}$$

which can be written in matrix form as:

$$\dot{V}_L = [\boldsymbol{F}_i^T \quad \boldsymbol{V}_e^T] \begin{bmatrix} \boldsymbol{K}_p^{-1}\boldsymbol{K}_i & \boldsymbol{K}_p^{-1}\boldsymbol{K}_i \\ \boldsymbol{K}_p^{-1}\boldsymbol{K}_i & \boldsymbol{K}_p^{-1}\boldsymbol{K}_d \end{bmatrix} \begin{bmatrix} \boldsymbol{F}_i \\ \boldsymbol{V}_e \end{bmatrix}. \tag{122}$$

Provided that $\boldsymbol{K}_d > \boldsymbol{K}_i$, and due to the fact that control gain matrices are positive definite, the time derivative of the Lyapunov function is negative semidefinite. □

9. Singularity Accommodation

Within a trajectory following task, it is possible for the desired trajectory to force the free-floating space manipulator system into a singular configuration at some point along the requested path. This singular configuration stems from the internal dynamics of the system and causes the system's Jacobian matrix to contain linear dependency and lose rank. Referring back to the definition of the GJM in (90), the system Jacobian losing rank in turn causes $\boldsymbol{J}^*$ to become ill conditioned. Simply taking the Moore-Penrose pseudo-inverse of the GJM (in the case of a redundant manipulator), denoted by $\boldsymbol{J}^{*\dagger}$, when in the neighbourhood of a singular configuration would result in unattainable actuation torques commanded by a controller, even for minimal movement in the output (in the singular directions). Managing an output tracking controller to operate within a safe region in the manipulator's joint space (i.e., a region which prevents excessive and potentially damaging joint velocities) requires the introduction of some method in the control design to push the system away from singular configurations.

This paper implements the damped least-squares method separately proposed by [31] and [48] to address the singularity avoidance problem in the output tracking control task. The damped-least squares method adds a damping factor to the inverse differential kinematics which reduces the trajectory following capabilities, however maintains the joints within a tolerable motion threshold. Consequently, singularity accommodation through the damped least-squares method is a constant exchange, quantified by the damping factor, between performance and practical control actions at the joints. First, let us explicitly define the formulation of the pseudo-inverse in a redundant manipulator's inverse kinematics with some Jacobian matrix $\boldsymbol{J}$,

$$\dot{\boldsymbol{\Phi}} = \boldsymbol{J}^{\dagger}\boldsymbol{V}_N = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1}\boldsymbol{V}_N. \tag{123}$$

The pseudo-inverse shown in equation (123) satisfies a solution for the inverse kinematics problem which minimizes the least-squares norm. That is, equation (123) satisfies:

$$\min(||\boldsymbol{V}_N - \boldsymbol{J}\dot{\boldsymbol{\Phi}}||). \tag{124}$$

Evidently, the solution associated with the minimum norm provides a set of joint velocities (where infinitely many solutions exist for redundant manipulators) which are most accurate in generating the end-effector velocity vector $\boldsymbol{V}_N$. As previously mentioned, strictly choosing the solution yielding the minimum norm is troublesome when the system is in the region of singular configurations. This solution no longer becomes feasible in its practical implementation due to the large joint velocities required to achieve the corresponding end-effector motion in the singular direction(s).

The damped least-squares method alters the problem for which the inverse kinematics is solved in order to consider the practical feasibility of the solution as well. In addition to considering the accuracy of the solution through the least-squares minimum norm problem in equation (124), the damped least-squares method additionally considers the norm of the joint velocities. Now, the solution to the inverse kinematics problem aims to minimize the weighted sum of the joint velocities' feasibility and the differential kinematic mapping provided positive definite weighting factors $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$. That is, the damped least squares solution satisfies the following problem [31],

$$\min(||\boldsymbol{V}_N - \boldsymbol{J}\dot{\boldsymbol{\Phi}}||^2_{\boldsymbol{W}_1} + ||\dot{\boldsymbol{\Phi}}||^2_{\boldsymbol{W}_2}), \tag{125}$$

where the vector norms above are defined equivalently to equation (112). As indicated by (125), the weight $\boldsymbol{W}_1$ signifies the emphasis placed on the accuracy of the mapping between joint and end-effector velocities, whereas the weight $\boldsymbol{W}_2$ denotes the importance of feasible joint velocities in the inverse kinematics solution. In [31], the authors present the solution to the damped least squares problem in (125) known as the Singularity Robust inverse (SR-inverse). As mentioned in [11], a considerable amount of the literature studies the case involving no task priority (i.e., $\boldsymbol{W}_1 = \boldsymbol{1}$) and the damping of joint velocities through setting $\boldsymbol{W}_2 = \lambda\boldsymbol{1}$. Since the weights are defined as positive definite matrices, the damping factor $\lambda \geq 0$. Using these weights, the SR-inverse of $\boldsymbol{J}$, denoted $\boldsymbol{J}^{\diamond}$, is defined as follows,

$$\boldsymbol{J}^{\diamond} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T - \lambda\boldsymbol{1})^{-1}. \tag{126}$$

Note that the SR-inverse solution above reduces to the Moores-Penrose pseudo-inverse in the case of no damping on the joint velocities (i.e. $\lambda = 0$). In the context of a trajectory following task, increasing the damping factor reduces the performance of the output's tracking capabilities. Consequently, it is most beneficial during a trajectory following scenario to only implement a joint velocity damping effect once the system's Jacobian matrix begins to lose rank close to a singular configuration. Outside of this region, when the Jacobian is well defined, the Moores-Penrose pseudo-inverse solution is preferred to achieve the greatest performance of the output tracking control law. Taking the determinant of the matrix $\boldsymbol{J}\boldsymbol{J}^T$ (i.e., the matrix being inverted in the pseudo-inverse) provides a metric for identifying ill conditioned systems. When $\det(\boldsymbol{J}\boldsymbol{J}^T) = 0$, the system reaches a singular configuration and the matrix $\boldsymbol{J}\boldsymbol{J}^T$ loses rank. Using the definition of a system's manipulability w presented by Yoshikawa in [52] as,

$$w := \sqrt{\det(\boldsymbol{J}\boldsymbol{J}^T)}, \tag{127}$$

to quantify the proximity to a singularity, the authors in [31] scale the damping factor based on w and a manipulability threshold value w_t. Above this threshold, the system is deemed sufficiently well defined to strictly consider accuracy in the inverse kinematics solution using (123). Once below w_t, the system is in the neighbourhood of a singular configuration, thus requiring a compromise in accuracy to provide damping to the joint velocities. Provided a maximum damping value λ_0 for when the system reaches singularity and $w = 0$, the damping factor scales as follows,

$$\lambda = \begin{cases} \lambda_0 \left(1 - \frac{w}{w_t}\right)^2, & w < w_t, \\ 0, & w \geq w_t. \end{cases} \tag{128}$$

Note that additional methods for defining the damping factor have been studied in the literature, involving the rate of change of the manipulatbility factor, the tracking error, the minimum singular value of $\boldsymbol{J}$, and the condition number of $\boldsymbol{J}\boldsymbol{J}^T + \lambda\mathbf{1}$, for example [22, 6, 24, 26, 11]. Based on an empirical analysis, the manipulability threshold w_t is set to $\sqrt{10}$ and the maximum damping λ_0 is set to 200 in this paper.

10. Simulation Analysis

The performance of the proposed output tracking controller is analyzed using a free-floating space manipulator simulation platform developed by the authors in Python for simulating the behaviour of free-floating multi-body systems. The simulation implements the dynamic and kinematic models on Lie groups presented in Sections 3 and 4 to model a space manipulator's response to control actions in the manipulator's joint space. The developed controller is implemented in this testing platform to observe its output tracking capabilities in a simulation setting. The algorithms associated with the dynamic and kinematic modellings have been carefully developed for optimizing computational efficiency to achieve a near real-time simulation of a controlled free-floating space manipulator in orbit. Note that a simulation timestep of 0.01 seconds is used here.

10.1. System Description

The simulation models a 2-link, 7-DOF shoulder-elbow-wrist space manipulator system illustrated in its zero configuration in Fig. 1. The full space manipulator system contains 13 twists: 3 translational and 3 rotational twists for the base motion, and 7 rotational twists to define the motion of the manipulator. The variables Φ_i appearing in Fig. 1 indicate the direction of positive rotation or translation associated with the i^{th} degree of freedom in the system. For simulation purposes, the two spherical joints (located at the manipulator's shoulder and wrist) are modeled as three individual revolute joints, separated by small masses. Note that such design of spherical joints in the simulation are still considered multi-DOF joints as they define motion in SO(3) between two rigid bodies possessing comparatively large masses. Provided this modelling of the spherical joints, the simulation replicates a space manipulator system with 10 bodies ($N = 9$) and 13 degrees of freedom ($N_D = 7$).

Table 2
Physical Properties of Simulated System

Body	**Index** (k)	**Dimension** [m] (l, w, h)	**Mass** [Kg]
Base	0	(1, 1, 1)	200
Shoulder bodies	1, 2, 3	(0.05, 0.1, 0.1)	0.5
Link 1	4	(2, 0.1, 0.1)	2
Link 2	5	(2, 0.1, 0.1)	2
Wrist bodies	6, 7, 8	(0.04, 0.04, 0.04)	0.1
End-effector	9	(0.05, 0.05, 0.05)	0.5

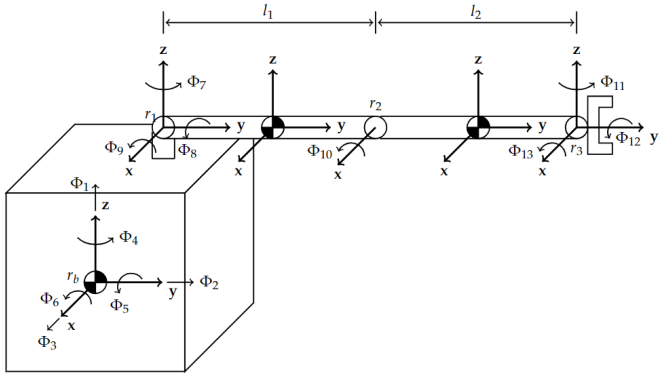


Figure 1: Simulated space manipulator system

Table 2 contains the physical properties of each body in the simulated system. Note that l, w, and h refer to the body's length, width, and height dimensions measured along the y, x, and z axes of the body coordinate frames, respectively. The moments of inertia for each body are computed based on the geometries and masses provided in Table 2 with the assumption of constant densities. Note that the densities of different components in the simulated system vary. For Body k, the corresponding inertia tensor $\boldsymbol{I}_k$ is calculated as follows,

$$\boldsymbol{I}_k = \begin{bmatrix} \frac{m_k}{12}\left(l^2 + h^2\right) & 0 & 0 \\ 0 & \frac{m_k}{12}\left(w^2 + h^2\right) & 0 \\ 0 & 0 & \frac{m_k}{12}\left(l^2 + w^2\right) \end{bmatrix} \tag{129}$$

where m_k is the mass of Body k. Additionally, the center of masses of each body are assumed to be located at their geometric centers, as specified by the centroids illustrated in Fig. 1. The spatial coordinate frame is initially located at

the center of mass of the base body, and the initial spatial positions of the joints and end-effector in metres are as follows:

$$\begin{aligned} \boldsymbol{p}_{shoulder} &= [0.1 \;\; 0.5 \;\; 0.6]^T \\ \boldsymbol{p}_{elbow} &= [0.1 \;\; 2.5 \;\; 0.6]^T \\ \boldsymbol{p}_{wrist} &= [0.14 \;\; 4.5 \;\; 0.64]^T \\ \boldsymbol{p}_{end-effector} &= [0.14 \;\; 4.55 \;\; 0.64]^T. \end{aligned}$$

Note that $\boldsymbol{p}_{shoulder}$ and $\boldsymbol{p}_{wrist}$ refer to the locations of the first members in the shoulder and wrist clusters, corresponding to the motions about the z-axes (i.e., Φ_7 and Φ_{11}, respectively). The centers of the subsequent joint bodies are located at a width's distance away in the x-direction (defining the axes of rotation for Φ_8 and Φ_{12}), followed by an additional offset distance equal to the respective joint bodies' height in the z-direction (defining the axes of rotation for Φ_9 and Φ_{13}) to complete the spherical joint clusters. Link 1 and the end-effector are attached to the last joint members (i.e., the revolute joints with motion about the x-axes) in the shoulder and wrist joints, respectively.

For each independent motion in the system the associated twist elements $\boldsymbol{v}$ and $\boldsymbol{w}$ are defined based on the axes of motion portrayed in Fig. 1. For the three translational motions of the spacecraft base Φ_1, Φ_2, and Φ_3, the respective components $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ are defined as unit vectors in the z, y, and x directions respectively (with rotational components $\boldsymbol{w}_1 = \boldsymbol{w}_2 = \boldsymbol{w}_3 = \boldsymbol{0}_{3\times1}$ due to the pure translational motion). The rotational components of the remaining revolute motions are defined along the following axes,

$$\begin{aligned} \boldsymbol{w}_4 = \boldsymbol{w}_7 = \boldsymbol{w}_{11} &= \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T, \\ \boldsymbol{w}_5 = \boldsymbol{w}_8 = \boldsymbol{w}_{12} &= \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T, \\ \boldsymbol{w}_6 = \boldsymbol{w}_9 = \boldsymbol{w}_{10} = \boldsymbol{w}_{13} &= \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T. \end{aligned}$$

The corresponding linear velocity component of a purely rotational screw motion is defined as $\boldsymbol{v} = -\boldsymbol{w}\times r$ with $r \in \mathbb{R}^3$ being a point along the associated axis of motion. The locations of the points along each twist axis with respect to the spatial coordinate frame are as follows,

$$\begin{aligned} \boldsymbol{r}_0 &= \begin{bmatrix}0\\0\\0\end{bmatrix}, \\ \boldsymbol{r}_7 = \begin{bmatrix}0\\0.5\\0.5\end{bmatrix} \quad \boldsymbol{r}_8 &= \begin{bmatrix}0.1\\0.5\\0.5\end{bmatrix} \quad \boldsymbol{r}_9 = \begin{bmatrix}0.1\\0.5\\0.6\end{bmatrix}, \\ \boldsymbol{r}_{10} &= \begin{bmatrix}0.1\\2.5\\0.6\end{bmatrix}, \\ \boldsymbol{r}_{11} = \begin{bmatrix}0.1\\4.5\\0.6\end{bmatrix} \quad \boldsymbol{r}_{12} &= \begin{bmatrix}0.14\\4.5\\0.6\end{bmatrix} \quad \boldsymbol{r}_{13} = \begin{bmatrix}0.14\\4.5\\0.64\end{bmatrix}, \end{aligned}$$

where the coordinates are a consequence of the dimensions provided in Table 2. The initial homogeneous transformation matrices between the spatial and body coordinate frames $\boldsymbol{g}_{sk}(0)$ make use of the spatial position vectors above, and the assumption that all body coordinate frames are aligned with the spatial frame. This leads to the following general definition for $\boldsymbol{g}_{sk}(0)$:

$$\boldsymbol{g}_{sk}(0) = \begin{bmatrix}\boldsymbol{1}_{3\times3} & \boldsymbol{r}_k \\ \boldsymbol{0} & 1\end{bmatrix}. \tag{130}$$

10.2. Simulation Results

The efficacy of the proposed output tracking controller presented in Sections 7 and 8, with the singularity accommodation outlined in Section 9, are evaluated under two trajectory following scenarios. The end-effector's trajectory following capabilities are tested for trivial trapezoidal trajectories in both the linear and angular motions. This desired trajectory is applied to the space manipulator system described in Fig. 1 containing some base motion (i.e., $\boldsymbol{\mu} \neq 0$) to demonstrate the controller's ability to dissipate an initial motion in the output and achieve path following. In addition, the robustness of the developed controller is also tested by introducing uncertainties to the mass values in the system. The second trajectory following task evaluates the singularity accommodation component of the controller by choosing a desired trajectory which forces the space manipulator to pass in the proximity of a singular configuration. The progression of the system's manipulability and the damping factor in the SR-inverse method are observed to ensure safe joint actuation while operating in the singular region.

The following controller gains are used in the simulation of the full pose controller to satisfy the Lyapunov stability condition discussed in Theorem 1: $\boldsymbol{K}_p = \text{diag}\{60, 60, 60, 60, 60, 60\}$, $\boldsymbol{K}_d = \text{diag}\{15, 15, 15, 15, 15, 15\}$, $\boldsymbol{K}_i = \text{diag}\{10, 10, 10, 10, 10, 10\}$. Note that the diagonal values of the PID gains are empirically established by observing the error response resulting from the modified feedforward, feedback PID control law in equations (106), (108), and (110). The output tracking testing scenario presented in the following subsection starts with an initial configuration $\boldsymbol{\Phi}_{ini}$ of $\Phi_{9,ini} = 60°$ and $\Phi_{10,ini} = -90°$ (all other generalized coordinates are initialized to zero). This initial configuration corresponds to an initial end-effector pose of:

$$\boldsymbol{g}_{sN}(0) = \begin{bmatrix}1 & 0 & 0 & 0.14\\ 0 & 0.866 & 0.5 & 3.25\\ 0 & -0.5 & 0.866 & 1.37\\ 0 & 0 & 0 & 1\end{bmatrix}, \tag{131}$$

where the end-effector's position is measured in meters. This choice of $\boldsymbol{\Phi}_{ini}$ ensures that the space manipulator is starting from a well defined system, away from the region of singular configurations.

Table 3
Mass Values used in the True Space Manipulator Model

Body Index	True Mass (kg)	Mass Variation (kg)	Percent Variation
0	228.91	28.91	14.45 %
1	0.442	-0.0579	11.59 %
2	0.408	-0.0921	18.41 %
3	0.443	-0.0572	11.44 %
4	2.215	0.215	10.76 %
5	2.367	0.367	18.34 %
6	0.0881	-0.0119	11.89 %
7	0.119	0.0193	19.31 %
8	0.0891	-0.0109	10.92 %
9	0.585	0.0852	17.04 %

10.2.1. Trapezoidal Trajectory Following Task

The desired linear and angular trapezoidal trajectories initially command constant accelerations of 0.2 m/s^2 and 0.09 rad/s^2 respectively for 0.65 seconds. These accelerations are equally applied along/about the primary coordinate axes. The trajectory then contains smooth decelerations for 0.1 seconds to reach constant velocities. This deceleration period avoids sharp changes in the desired velocities, in turn requiring smoother control actions during the switch from constant acceleration to constant velocity in the output. The end-effector moves at constant velocity for 0.75 seconds followed by a constant deceleration of the same magnitudes for 0.65 seconds to reach the target position with no linear velocity in the output. The same smoothing period of gradual deceleration from constant velocity to constant deceleration is again implemented. The consequent time to reach the target point approximately 36 cm away is 2.25 seconds, resulting in angular rotations of approximately 0.1 rad about all axes.

An initial angular motion of 0.2 rad/s is now added about the z-axis of the spacecraft's base (i.e., $\dot{\Phi}_{4,ini} = 0.2$ rad/s) to implement a conserved non-zero momentum to the system. The initial base rotation introduces an equivalent angular rotation to the end-effector, as viewed from the body coordinate frame attached to the base. Consequently, the system's output starts the trajectory following task with a non-zero velocity along the x-axis of the base coordinate frame (initially coincident with spatial coordinate frame). Note that the simulation is run for 1.2 seconds longer than the time required for target capture to permit the tracking and velocity errors to settle.

Uncertainties are added to the masses of each body in the true space manipulator system (i.e., the model associated with the plant dynamics). Consequently, the inertia and Coriolis matrices for the actual system use masses which have been varied by a random amount. In the simulation, this random variation is determined by a random number between 20 and 10 percent of the associated mass's estimated value. Such variations are randomly chosen to be either added or subtracted to the estimated masses values.

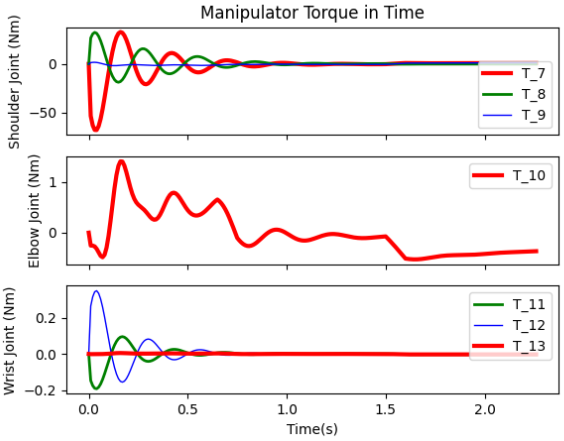
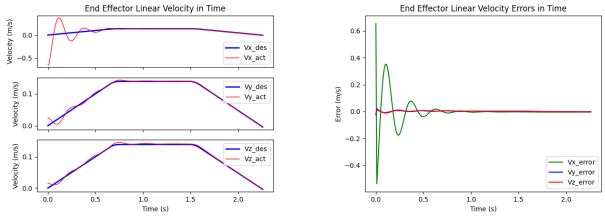


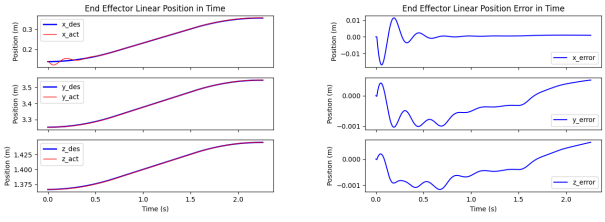
Figure 2: Control torques for the trapezoidal trajectory following task applied to an uncertain system with non-zero momentum.

The physical properties included in Table 2 are consequently treated as the estimated values, and are used in the dynamics reduction and feedback linearization processes. Note that the column titled "Mass Variation" refers to the mass quantity added or subtracted to the body's corresponding value in Table 2, and the column "Percent Variation" refers to the percentage of the estimated mass value associated with the variation.



(a) Desired and actual end-effector linear velocity

(b) Linear velocity error

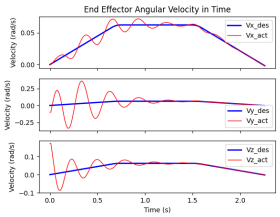


(c) Desired and actual end-effector linear position

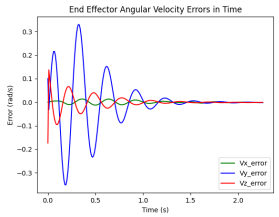
(d) Linear tracking error

Figure 3: End-effector linear response to the full pose trapezoidal trajectory applied to an uncertain non-zero momentum.

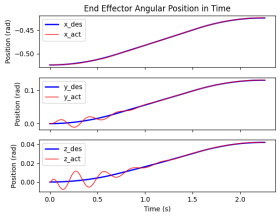
Fig. 2 shows the control torques performed in the full pose trapezoidal trajectory following task implemented on the space manipulator with non-zero momentum and model uncertainties. As expected, to dissipate the initial linear motion in the x-axis of the output, the greatest amount of



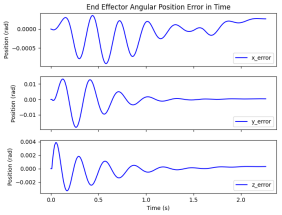
(a) Desired and actual end-effector angular velocity



(b) Angular velocity error

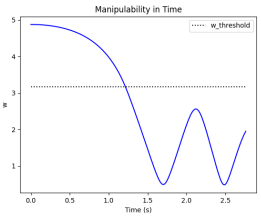


(c) Desired and actual end-effector angular position

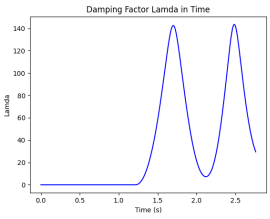


(d) Angular tracking error

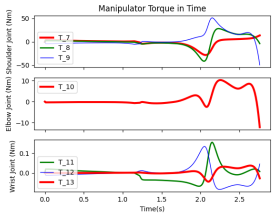
Figure 4: End-effector angular response to the full pose trapezoidal trajectory applied to an uncertain system with non-zero momentum.



(a) Manipulability



(b) Damping factor



(c) Control torque

Figure 5: Damped least squares parameters for SR-inverse in the linear controller with associated control input

torque is initially required about the z-axis of the shoulder joint (i.e., $\boldsymbol{T}_7$). The actuation about the elbow joint remains within 1.5 Nm in magnitude as this joint induces no motion about the x-axis of the output's spatial position. Fig. 3a and Fig. 4a respectively depict the resulting linear and angular velocity responses of the end-effector in the trajectory following task. Recall that the velocities shown in Fig. 3a are with respect to the end-effector's body coordinate frame, and the linear positions in Fig. 3c express the end-effector's position with respect to the spatial frame. The developed output tracking controller is shown to effectively diminish the initial velocity in the x-axis due to the base's rotational velocity. Based on the linear velocity error displayed in Fig. 3b, a maximum overshoot of magnitude of 0.54 m/s along the x-axis occurs, eventually settling after around 0.56 s to within 2 cm/s. The initial oscillation about the desired trajectory in the linear position along the x-axis, shown in Fig. 3c and Fig. 3d, reaches a maximum undershoot of 1.66 cm which reduces to within 0.5 cm after approximately 0.24 s.

Initial motions of -0.1 rad/s and 0.17 rad/s about the y and z axes are induced by the initial motion of the base. In dissipating such rotational motions, maximum errors of -0.35 rad/s and -0.092 rad/s about the y and z axes are shown in Fig. 4b. Such angular velocity error about the z-direction is decreased to within 0.02 rad/s after approximately 0.63 s and to within 0.02 rad/s after 1.11 s about the y-axis. Note that the base's initial angular rotation about the z-axis in turn induces an equivalent angular rotation of 0.2 rad/s in the end-effector's angular velocity with respect to the spatial frame. This angular motion in the output is apparent from the oscillations in the end-effector's angular position about the z-axis, as shown in Fig. 4c. The associated angular tracking error, depicted in Fig. 4d, about the z-axis demonstrates a maximum error of 0.0038 rad which settles to within 0.0005 rad after 0.9 s. The controlled system's trajectory following capabilities exhibited in this testing scenario highlight the developed controller's ability to accommodate a system with conserved non-zero momentum while remaining robust to uncertainties in the system's inertia matrix.

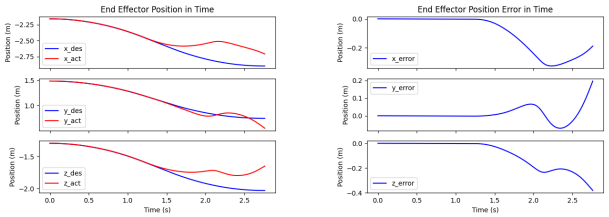
10.2.2. Singularity Accommodation Task

This trajectory following task implements a desired trajectory which requires the space manipulator to pass near a known singular configuration to exhibit the proficiency of the singularity accommodation feature in the developed control structure. Starting from the following initial configuration (in degrees):

$$\boldsymbol{\Phi}_{ini} = [0,\ 0,\ 0,\ -28,\ 20,\ 15,\ 90,\ 20,\ -25,\ -60,\ 0,\ 0,\ 0]^T$$

a trapezoidal trajectory (exclusively applied to the output's linear motion) which decelerates by a magnitude of 0.4 m/s^2 for 1.25 s, travels at constant velocity for 0.25 s, and accelerates by 0.4 m/s^2 for 1.25 s along all axes is provided. Figures 5a and 5b display the progression of the space manipulator's manipulability and damping factor throughout this trajectory following task. As indicated by the plot of manipulability, the system starts in a well-defined configuration with sufficient manipulability, and as the simulation progresses, the space manipulator becomes increasingly ill-conditioned.

Figure 5a shows the system's manipulability to fall below the threshold value of $\sqrt{10}$ after about 1.2 s, consequently inducing a shift from the pseudo-inverse solution to the SR-inverse in the feedback linearization law.



(a) Desired and actual end-effector position

(b) Tracking error

Figure 6: Output tracking performance in the presence of a singularity

This shift is demonstrated by the introduction of a non-zero damping factor once the system enters the region of insufficient manipulability, as shown in Figure 5b. The damping factor is shown to quickly increase as the system progresses towards the singularity. At the system's closest proximity to the singular configuration, occurring at 1.7 s, a manipulability of 0.5 and a maximum damping factor of 142 (where $\lambda_0 = 200$, as mentioned in section 9) are reached. The increase in the damping factor is shown to effectively limit the actuation torques to within 10 Nm at the shoulder and elbow joints at the time of lowest manipulability, as per Figure 5c (demonstrating the control input in time). Figures 6a and 6b show the end-effector's linear position and the associated tracking error in time, respectively, to exhibit the resulting trajectory following capabilities of the space manipulator system with insufficient manipulability. Highly accurate trajectory following is achieved (within 2 mm) until around 1.4 s, afterwhich the end-effector is shown to deviate from the desired path by about 1 cm in all axes. This tracking error continues to increase by an order of magnitude as the damping factor similarly increases, and less emphasis is placed on the tracking accuracy. After 1.7 s, the space manipulator is shown in Figure 5a to become more well-conditioned, in turn causing the damping factor to decrease. In this case, the controller attempts to regain trajectory following in the output, consequently allowing for larger control torques to be performed, as shown by Figure 5c. Note that the torque levels at this point (still below the manipulability threshold) are within an acceptable range of 50 Nm at the shoulder and 10 Nm at the elbow. In trying to re-follow the desired path, the system again falls towards the singularity, causing the damping to increase and the manipulator joint torques to again be effectively maintained within a safe limit.

11. Conclusion

This paper developed an output tracking controller on the SE(3) Lie group to control the end-effector pose of free-floating space manipulators with conserved non-zero momentum. Using geometric mechanics, the kinematics and dynamics models of single-arm free-floating space manipulators were derived. The product of exponentials formula was formulated for space manipulators considered as rigid multi-body systems with multi-DOF joints to form the forward and differential kinematics mappings. For implementation in the product of exponentials, multi-DOF joints were newly modelled as the amalgamated exponential mappings of individual screw motions along a single axis. The dynamics of free-floating space manipulators was derived using an Euler-Lagrange approach, and subsequently decoupled into the base and manipulator motions. Using the conserved linear and angular momentum as affine nonholonomic constraints, the decoupled equations of motion were reduced and restricted to the manipulator's joint space. Input-output feedback linearization was employed on $\mathfrak{se}(3)$ to remove all non-linearities in the end-effector's motion. The proposed modified feedforward, feedback PID controller was structured to stabilized the end-effector's pose along a feasible desired trajectory, in the sense of Lyapunov. The full pose output tracking control law was shown to act on the gradient of the coordinate free error function applied to the pose group error, and the associated velocity error in the body frame. Singularity accommodation using the SR-inverse technique was implemented in the feedback linearization laws to mitigate the risk of commanding unachievable actuations at the manipulator joints. The proposed output tracking control law was tested in a Python simulation, demonstrating a robustness to model uncertainties in the inertia matrix, the ability to successfully dissipate unwanted motion in the output, due to an initial rotation of the spacecraft base (i.e., $\boldsymbol{\mu} \neq 0$), and achieve output tracking for a trapezoidal trajectory. Moreover when the space manipulator system was forced into a region of low manipulability (i.e., below the manipulability threshold), the damping factor in the SR-inverse was shown to increase, resulting in the control actions to be held within an acceptable boundary, at the expense of tracking accuracy.

12. Acknowledgements

The authors thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada Research Chair Program for funding this project.

Appendix A Homogeneous Transformation Partial Derivative

The following provides the derivation of the partial derivative of a homogeneous transformation between the spatial frame and a coordinate frame attached to Body k, $\boldsymbol{g}_{sk} \in \mathrm{SE}(3)$, with respect to the i^{th} generalized coordinate Φ_i [29]:

$$\frac{\partial \boldsymbol{g}_{sk}}{\partial \Phi_i} = e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_{i-1}\Phi_{i-1}} \frac{\partial}{\partial \Phi_i}\left(e^{\hat{\boldsymbol{\xi}}_i\Phi_i}\right) e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \dots e^{\hat{\boldsymbol{\xi}}_k\Phi_k} \boldsymbol{g}_{sk}(0) \tag{132}$$

$$= e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_{i-1}\Phi_{i-1}} \left(\hat{\boldsymbol{\xi}}_i\right) e^{\hat{\boldsymbol{\xi}}_i\Phi_i} \dots e^{\hat{\boldsymbol{\xi}}_k\Phi_k} \boldsymbol{g}_{sk}(0), \tag{133}$$

which when multiplied with $\boldsymbol{g}_{sk}^{-1}$ yields the following,

$$\left(\frac{\partial \boldsymbol{g}_{sk}}{\partial \Phi_i}\right) \boldsymbol{g}_{sk}^{-1} = e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_{i-1}\Phi_{i-1}} \left(\hat{\boldsymbol{\xi}}_i\right) e^{-\hat{\boldsymbol{\xi}}_{i-1}\Phi_{i-1}} \dots e^{-\hat{\boldsymbol{\xi}}_1\Phi_1}. \tag{134}$$

Appendix B Adjoint Operator Partial Derivatives

In the following, the derivation of the partial derivative of the Adjoint operator $\mathbf{Ad}_j^l$ being applied to an element $\hat{\boldsymbol{A}} \in \mathfrak{se}(3)$, with respect to a generalized coordinate Φ_i is provided:

$$\frac{\partial \mathbf{Ad}_j^l \boldsymbol{A}}{\partial \Phi_i} = e^{-\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{-\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} e^{-\hat{\boldsymbol{\xi}}_i\Phi_i} \left(-\hat{\boldsymbol{\xi}}_i\right) \dots e^{-\hat{\boldsymbol{\xi}}_k\Phi_k} \left(\hat{\boldsymbol{A}}\right) e^{\hat{\boldsymbol{\xi}}_k\Phi_k} \dots e^{\hat{\boldsymbol{\xi}}_l\Phi_l}$$

$$+ e^{-\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{-\hat{\boldsymbol{\xi}}_k\Phi_k} \left(\hat{\boldsymbol{A}}\right) e^{\hat{\boldsymbol{\xi}}_k\Phi_k} \dots e^{\hat{\boldsymbol{\xi}}_i\Phi_i} \left(\hat{\boldsymbol{\xi}}_i\right) e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \dots e^{\hat{\boldsymbol{\xi}}_l\Phi_l} \tag{135}$$

$$= -e^{-\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{-\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \left(\hat{\boldsymbol{\xi}}_i\right) \left(\mathbf{Ad}_j^i \boldsymbol{A}\right)^\wedge e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \dots e^{\hat{\boldsymbol{\xi}}_l\Phi_l}$$

$$+ e^{-\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{-\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \left(\mathbf{Ad}_j^i \boldsymbol{A}\right)^\wedge \left(\hat{\boldsymbol{\xi}}_i\right) e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \dots e^{\hat{\boldsymbol{\xi}}_l\Phi_l} \tag{136}$$

$$= -e^{-\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{-\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \left[\hat{\boldsymbol{\xi}}_i \left(\mathbf{Ad}_j^i \boldsymbol{A}\right)^\wedge + \left(\mathbf{Ad}_j^i \boldsymbol{A}\right)^\wedge \hat{\boldsymbol{\xi}}_i\right] e^{\hat{\boldsymbol{\xi}}_l\Phi_l} \dots e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \tag{137}$$

$$= -\mathbf{Ad}_{i+1}^l \mathbf{ad}_{\hat{\boldsymbol{\xi}}_i} \mathbf{Ad}_j^i(\boldsymbol{A}) \tag{138}$$

where between (135) and (136) we note that $e^{\hat{\boldsymbol{\xi}}_i\Phi_i}\hat{\boldsymbol{\xi}}_i = \hat{\boldsymbol{\xi}}_i e^{\hat{\boldsymbol{\xi}}_i\Phi_i}$.

We now present the derivation of the partial derivative $\frac{\partial(\mathbf{Ad}_1^{j-1})'}{\partial \Phi_i}$ required for equation (75). We again demonstrate the derivation for a general twist $\hat{\boldsymbol{A}} \in \mathfrak{se}(3)$:

$$\frac{\partial(\mathbf{Ad}_1^{j-1})'\boldsymbol{A}}{\partial \Phi_i} = e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_i\Phi_i} \left(\hat{\boldsymbol{\xi}}_i\right) e^{\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} \dots e^{\hat{\boldsymbol{\xi}}_{j-1}\Phi_{j-1}} \left(\hat{\boldsymbol{A}}\right) e^{-\hat{\boldsymbol{\xi}}_{j-1}\Phi_{j-1}} \dots e^{-\hat{\boldsymbol{\xi}}_1\Phi_1}$$

$$- e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_{j-1}\Phi_{j-1}} \left(\hat{\boldsymbol{A}}\right) e^{-\hat{\boldsymbol{\xi}}_{j-1}\Phi_{j-1}} \dots e^{-\hat{\boldsymbol{\xi}}_{i+1}\Phi_{i+1}} e^{-\hat{\boldsymbol{\xi}}_i\Phi_i} \left(\hat{\boldsymbol{\xi}}_i\right) \dots e^{-\hat{\boldsymbol{\xi}}_1\Phi_1} \tag{139}$$

$$= e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \dots e^{\hat{\boldsymbol{\xi}}_i\Phi_i} \left[\hat{\boldsymbol{\xi}}_i \left((\mathbf{Ad}_{i+1}^{j-1})'\boldsymbol{A}\right)^\wedge - \left((\mathbf{Ad}_{i+1}^{j-1})'\boldsymbol{A}\right)^\wedge \hat{\boldsymbol{\xi}}_i\right] e^{-\hat{\boldsymbol{\xi}}_i\Phi_i} \dots e^{\hat{\boldsymbol{\xi}}_1\Phi_1} \tag{140}$$

$$= (\mathbf{Ad}_1^i)'\mathbf{ad}_{\boldsymbol{\xi}_i}(\mathbf{Ad}_{i+1}^{j-1})'(\boldsymbol{A}). \tag{141}$$

References

[1] Abdollahi, F., Chhabra, R., 2022. Actuator fault recovery in formation control of uncertain multi-agent systems on the lie group se(3), in: American Control Conference.

[2] Aghili, F., 2009. Coordination control of a free-flying manipulator and its base attitude to capture and detumble a noncooperative satellite, in: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE. pp. 2365–2372.

[3] Al-Isawi, M., Sasiadek, J.Z., 2019. Guidance and control of a robot capturing an uncooperative space target. Journal of Intelligent & Robotic Systems 93, 713–721.

[4] Barcinski, T., Lisowski, J., Rybus, T., Seweryn, K., 2013. Controlled zero dynamics feedback linearization with application to free-floating redundant orbital manipulator, in: 2013 American Control Conference, IEEE. pp. 1834–1839.

[5] Bullo, F., Murray, R.M., 1999. Tracking for fully actuated mechanical systems: a geometric framework. Automatica 35, 17–34.

[6] Chan, S.K., Lawrence, P.D., 1988. General inverse kinematics with the error damped pseudoinverse, in: Proceedings. 1988 IEEE international conference on robotics and automation, IEEE. pp. 834–839.

[7] Chhabra, R., Emami, M.R., 2014. A generalized exponential formula for forward and differential kinematics of open-chain multi-body systems. Mechanism and Machine Theory 73, 61–75.

[8] Chhabra, R., Emami, M.R., 2015. Symplectic reduction of holonomic open-chain multi-body systems with constant momentum. Journal of Geometry and Physics 89, 82–110.

[9] Chhabra, R., Emami, M.R., 2016. A unified approach to input-output linearization and concurrent control of underactuated open-chain multi-body systems with holonomic and nonholonomic constraints. Journal of dynamical and control systems 22, 129–168.

[10] Daneshjou, K., Mohammadi-Dehabadi, A., Bakhtiari, M., 2017. Mission planning for on-orbit servicing through multiple servicing satellites: A new approach. Advances in Space Research 60, 1148–1162.

[11] Deo, A.S., Walker, I.D., 1995. Overview of damped least-squares methods for inverse kinematics of robot manipulators. Journal of Intelligent and Robotic Systems 14, 43–68.

[12] Dimitrov, D.N., 2006. Dynamics and control of space manipulators during a satellite capturing operation. Ph.D. thesis.

[13] Dubowsky, S., Torres, M.A., 1991. Path planning for space manipulators to minimize spacecraft attitude disturbances, in: Proceedings of IEEE International Conference on Robotics and Automation, Citeseer. pp. 2522–2528.

[14] Dubowsky, S., Vafa, Z., 1987. A virtual manipulator model for space robotic systems, in: Jet Propulsion Lab., California Inst. of Tech., Proceedings of the Workshop on Space Telerobotics, Volume 3.

[15] Ellery, A., 2004. An engineering approach to the dynamic control of space robotic on-orbit servicers. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 218, 79–98.

[16] Evangelos, P., Steven, D., 1991. On the nature of control algorithms for free-floating space manipulators. Robotics and Automation, IEEE Transactions on 7, 750–758.

[17] F. Bullo and A.D. Lewis, 2004. Geometric Control of Mechanical Systems: Modeling, Analysis, and Design for Simple Mechanical Control Systems. Springer, New York, NY.

[18] Ghasemi, K., Ghaisari, J., Abdollahi, F., 2019. Robust formation control of multiagent systems on the Lie group SE(3). International Journal of Robust and Nonlinear Control 30, No. 3, 966–998.

[19] Graham, A.R., Kingston, J., 2015. Assessment of the commercial viability of selected options for on-orbit servicing (oos). Acta Astronautica 117, 38–48.

[20] Grover, D., Jacobs, S., Abbasi, V., Cree, D., Daae, M., Hay, J., He, W., Huang, X., Jun, Z., Kearney, S., et al., 2008. Development of on-orbit servicing concepts, technology option and roadmap (part i)-commercial aspects. Journal of the British Interplanetary Society 61, 203–212.

[21] Jost, J., Jost, J., 2008. Riemannian geometry and geometric analysis. volume 42005. Springer.

[22] Kelmar, L., Khosla, P.K., 1988. Automatic generation of kinematics for a reconfigurable modular manipulator system, in: Proceedings. 1988 IEEE international conference on robotics and automation, IEEE. pp. 663–668.

[23] Koditschek, D.E., 1989. The application of total energy as a lyapunov function for mechanical control systems. Contemporary mathematics 97, 131.

[24] Maciejewski, A.A., Klein, C.A., 1988. Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. Journal of Robotic systems 5, 527–552.

[25] Maithripala, D., Berg, J.M., 2015. An intrinsic pid controller for mechanical systems on lie groups. Automatica 54, 189–200.

[26] Mayorga, R.V., Wong, A.K., Milano, N., 1992. A fast damped least-squares solution to manipulator inverse kinematics and singularities prevention, in: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, IEEE. pp. 1177–1184.

[27] Moghaddam, B.M., Chhabra, R., 2021. On the guidance, navigation and control of in-orbit space robotic missions: A survey and prospective vision. Acta Astronautica 184, 70–100.

[28] Muller, A., Maisser, P., 2003. A lie-group formulation of kinematics and dynamics of constrained mbs and its application to analytical mechanics. Multibody system dynamics 9, 311–352.

[29] Murray, R.M., Li, Z., Sastry, S.S., 2017. A mathematical introduction to robotic manipulation. CRC press.

[30] Murray, R.M., Li, Z., Sastry, S.S., Sastry, S.S., 1994. A mathematical introduction to robotic manipulation. CRC press.

[31] Nakamura, Y., Hanafusa, H., 1986. Inverse kinematic solutions with singularity robustness for robot manipulator control .

[32] Nanos, K., Papadopoulos, E.G., 2017. On the dynamics and control of free-floating space manipulator systems in the presence of angular momentum. Frontiers in Robotics and AI 4, 26.

[33] Papadopoulos, E.G., 1990. On the dynamics and control of space manipulators. Ph.D. thesis. Massachusetts Institute of Technology.

[34] Park, F.C., Bobrow, J.E., Ploen, S.R., 1995. A lie group formulation of robot dynamics. The International journal of robotics research 14, 609–618.

[35] Parlaktuna, O., Ozkan, M., 2004. Adaptive control of free-floating space manipulators using dynamically equivalent manipulator model. Robotics and Autonomous Systems 46, 185–193.

[36] Perez-Gracia, A., McCarthy, J.M., 2006. Kinematic synthesis of spatial serial chains using clifford algebra exponentials. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 220, 953–968.

[37] Sternberg, D., Chodas, M., Jewison, C., Jones, M., De Weck, O., 2015. Multidisciplinary system design optimization of on orbit satellite assembly architectures, in: 2015 IEEE Aerospace Conference, IEEE. pp. 1–14.

[38] Stramigioli, S., Maschke, B., Bidard, C., 2000. A hamiltonian formulation of the dynamics of spatial mechanisms using lie groups and screw theory, in: Proc. Symposium Commemorating the Legacy, Work and Life of Sir RS Ball, J. Duffy and H. Lipkin organizers.

[39] Stramigioli, S., Maschke, B., Bidard, C., 2002. On the geometry of rigid-body motions: The relation between lie groups and screws. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 216, 13–23.

[40] Torres, M.A., Dubowsky, S., 1992. Minimizing spacecraft attitude disturbances in space manipulator systems. Journal of guidance, control, and dynamics 15, 1010–1017.

[41] Torres, M.A., Dubowsky, S., 1993. Path-planning for elastically constrained space manipulator systems, in: [1993] Proceedings IEEE International Conference on Robotics and Automation, IEEE. pp. 812–817.

[42] Tortopidis, I., Papadopoulos, E., 2006. Point-to-point planning: methodologies for underactuated space robots, in: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., IEEE. pp. 3861–3866.

[43] Ulrich, S., Sasiadek, J.Z., 2015. On the simple adaptive control of flexible-joint space manipulators with uncertainties, in: Aerospace Robotics II. Springer, pp. 13–23.
[44] Umetani, Y., Yoshida, K., 1987. Continuous path control of space manipulators mounted on omv. Acta Astronautica 15, 981–986.
[45] Umetani, Y., Yoshida, K., et al., 1989. Resolved motion rate control of space manipulators with generalized jacobian matrix. IEEE Transactions on robotics and automation 5, 303–314.
[46] Vafa, Z., Dubowsky, S., 1987. On the dynamics of manipulators in space using the virtual manipulator approach, in: Proceedings. 1987 IEEE International Conference on Robotics and Automation, IEEE. pp. 579–585.
[47] Vafa, Z., Dubowsky, S., 1990. The kinematics and dynamics of space manipulators: The virtual manipulator approach. The International Journal of Robotics Research 9, 3–21.
[48] Wampler, C.W., 1986. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. IEEE Transactions on Systems, Man, and Cybernetics 16, 93–101.
[49] Wang, H., 2011. On adaptive inverse dynamics for free-floating space manipulators. Robotics and Autonomous Systems 59, 782–788.
[50] Yoshida, K., 2003. Engineering test satellite vii flight experiments for space robot dynamics and control: theories on laboratory test beds ten years ago, now in orbit. The International Journal of Robotics Research 22, 321–335.
[51] Yoshida, K., Dimitrov, D., Nakanishi, H., 2006. On the capture of tumbling satellite by a space robot, in: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE. pp. 4127–4132.
[52] Yoshikawa, T., 1985. Manipulability of robotic mechanisms. The international journal of Robotics Research 4, 3–9.
[53] Zhang, Z., Sarlette, A., Ling, Z., 2015. Integral control on lie groups. Systems & Control Letters 80, 9–15.