A Differentiable Dynamic Modeling Approach to Integrated Motion Planning and Actuator Physical Design for Mobile Manipulators

Lu, Zehui; Wang, Yebin

TR2024-095 July 16, 2024

Abstract

This paper investigates the differentiable dynamic modeling of mobile manipulators to facilitate efficient motion planning and physical design of actuators, where the actuator design is parameterized by physically meaningful motor geometry parameters. These parameters impact the manipulator's link mass, inertia, center-of-mass, torque constraints, and angular velocity constraints, influencing control authority in motion planning and trajectory tracking control. A motor's maximum torque/speed and how the design parameters affect the dynamics are modeled analytically, facilitating differentiable and analytical dynamic modeling. Additionally, an integrated locomotion and manipulation planning problem is formulated with direct collocation discretization, using the proposed differentiable dynamics and motor parameterization. Such dynamics are required to capture the dynamic coupling between the base and the manipulator. Numerical experiments demonstrate the effectiveness of differentiable dynamics in speeding up optimization and advantages in task completion time and energy consumption over established sequential motion planning approach. Finally, this paper introduces a simultaneous actuator design and motion planning framework, providing numerical results to validate the proposed differentiable modeling approach for co-design problems.

Journal of Field Robotics 2024

A Differentiable Dynamic Modeling Approach to Integrated Motion Planning and Actuator Physical Design for Mobile Manipulators

Zehui Lu, Yebin Wang

Abstract—This paper investigates the differentiable dynamic modeling of mobile manipulators to facilitate efficient motion planning and physical design of actuators, where the actuator design is parameterized by physically meaningful motor geometry parameters. These parameters impact the manipulator's link mass, inertia, center-of-mass, torque constraints, and angular velocity constraints, influencing control authority in motion planning and trajectory tracking control. A motor's maximum torque/speed and how the design parameters affect the dynamics are modeled analytically, facilitating differentiable and analytical dynamic modeling. Additionally, an integrated locomotion and manipulation planning problem is formulated with direct collocation discretization, using the proposed differentiable dynamics and motor parameterization. Such dynamics are required to capture the dynamic coupling between the base and the manipulator. Numerical experiments demonstrate the effectiveness of differentiable dynamics in speeding up optimization and advantages in task completion time and energy consumption over established sequential motion planning approach. Finally, this paper introduces a simultaneous actuator design and motion planning framework, providing numerical results to validate the proposed differentiable modeling approach for co-design problems.

I. Introduction

Mobile manipulators offer an evolution in robotic system architectures, enabling them to transition from automated systems to autonomous ones [1]. These systems combine mobility, provided by the mobile platform, with the manipulation capabilities of the mounted articulated arm, resulting in a versatile manipulation workspace [2]. Mobile manipulators have gained attention across various domains, including industrial settings like factories and warehouses [3]–[5], indoor environments such as healthcare [6], and outdoor field applications like environment exploration [7], [8], excavation [9], and satellite service [10]. Their flexibility makes them well-suited for undertaking and assisting with a wide range of tasks [2]. Nevertheless, mobile manipulators are expected to operate in complex, less structured, and dynamic environments, which presents unique challenges in system design, perception, motion (i.e. locomotion and manipulation) planning, and control. These challenges have motivated researchers to explore various approaches in planning for mobile manipulators.

Motion planning is crucial in bridging the gap between perception and control in autonomous mobile manipulators across different deployment stages. This continuum is characterized by the level of feedback between perception and control, spanning from motion planning methodologies primarily used in offline settings to various forms of online replanning or closed-loop control suitable for navigating complex, less structured, and dynamic environments [2]. Motion planning typically involves predicting how the ego robot behaves based on a certain range of actions, and assessing the impacts of these actions on both the environment and the robot's subsequent decision-making process. The prediction usually relies on either a kinematic or dynamic model of the ego robot. The choice between these models depends on the required accuracy of the modeling and the specific mission requirements.

Over the past two decades, researchers have been focusing on developing computationally efficient methods for generating safe and robust motion planning in a complex environment. The planning/control hierarchy of a robot typically consists of at least two levels [11]. A high-level planner, often a motion planner, generates a desired trajectory of velocity, acceleration, or torque for each actuator, which is then passed to a low-level controller. The low-level controller for each actuator is responsible for tracking the desired trajectory provided by the high-level planner. However, the actual behavior of the robot may not always align exactly with what the motion planning algorithm anticipates due to the robot's modeling fidelity and the inherent dynamics of its actuators. Based on existing literature, this discrepancy or uncertainty is somewhat manageable to some extent in specific domains. However, from the perspective of designers or manufacturers, this level of modeling and its subsequent motion planning is insufficient. In other words, manufacturers must understand how critical design parameters influence the system dynamics, actuator capabilities, subsequent motion planning, and eventually closed-loop robot control. Given some performance metrics, these factors can assist in iteratively updating the design parameters to achieve optimal robot performance. The entire process is known as robot co-design. Additionally, both the system dynamics and the impact of design parameters on the system dynamics are expected to be analytically modeled, which aims to facilitate solving some underlying optimization problems related to motion planning or robot design given

Z. Lu is with the School of Aeronautics and Astronautics, Purdue University, IN 47907, USA. Email: lu846@purdue.edu This work was done while Z. Lu was an intern at Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA.

Y. Wang is with MERL. Email: yebinwang@ieee.org

existing numerical gradient-based optimization solvers.

To summarize, is it necessary to understand how major design parameters analytically influence the system dynamics and actuator capabilities, and how these factors affect motion planning subsequently. This paper investigates the analytical and differentiable dynamic modeling of mobile manipulators with motor parameterization given physically meaningful motor geometry parameters, which enables integrated motion (locomotion and manipulation) planning and actuator design. This paper also studies how the design parameters affect the actuators' capabilities and subsequently the motion planning given the parameterized dynamics model. The differentiability in this paper means that the gradient of the system dynamics regarding the underlying optimization decision variables, such as the system's states and inputs, is analytical.

A. Literature Review

1) Motion Planning: The mobile base and manipulator have significantly different dynamic characteristics, as the mobile base generally has higher inertia than the manipulator. Despite this disparity, these two systems are strongly coupled, leading to complex dynamic behaviors. These characteristics compound the complexity of the planning problem.

Various approaches have been employed to solve the motion planning problem for mobile manipulators. The planning algorithms are generally categorized into two classes, i.e. combined or separate planning, based on whether they consider the behavioral differences between the mobile base and the robot manipulator or not [12].

For separate planning, a complex task is often divided into a sequence of sub-tasks, and planning is conducted separately for each sub-task. This approach essentially decouples the planning for the mobile base and the robot manipulator due to the lack of a dynamic model of the entire mobile manipulator. Nevertheless, it reduces the complexity of planning over a high-dimensional space. While various planning algorithms exist in the literature for locomotion (mobile base) [13] and manipulation planning [14]–[23], they have not been specifically implemented for mobile manipulators. When motion planning is separately performed for the mobile base and the manipulator, the existing algorithms for each component can be utilized once a goal configuration is determined for both the mobile base and the manipulator. The separation of locomotion and manipulation results in inefficient and suboptimal solutions even if optimal solutions are obtained for each sub-task [12]. Furthermore, the separation may lead to infeasibility since poor base placement could render the final goal state unreachable [12], [24].

As for the combined planning, existing methods can be categorized into several subsets based on utilizing different levels of model information. First, some methods rely on the kinematic model of a simple mobile manipulator to design position-tracking controllers [25] and perform trajectory planning [26]. Particularly, [27] perform position-tracking control based on a kinematic model of the base and a dynamic model of the manipulator. Also, [28] derive the dynamics for a two-wheel differential unicycle with a two-link manipulator, and proposes an optimization-based trajectory planning with constant joint velocity limits.

The majority of the literature relies on fast sampling/searching [5], [8], [17], [29]–[38] or optimizing [5], [9], [16], [39]–[42] over a generally high-dimensional configuration space or the task space (i.e. the Euclidean space) with kinematic constraints, where the collision avoidance is typically implemented by checking the intersection between the forward occupancy (volume) of the robot and the volume of obstacles. Since implementing the forward occupancy or enforcing kinematic constraints only requires a kinematic model, there is no essential difference in doing so for a mobile manipulator or a fixed 6-DOF (degree of freedom) manipulator. Consequently, a mobile manipulator is considered a single system despite the behavioral difference between the mobile base and the manipulator. Due to the absence of dynamics or actuator information, the motion trajectory may not always be dynamically feasible. As a result, one can only assume that, given empirical kinematic constraints on joint velocities and accelerations, the desired inputs for actuators are always feasible. Thus, at run-time, the actual trajectory may deviate from the expected one from motion planning. On the other hand, these empirical kinematic constraints could also significantly affect the operation speed of robots. This effect is especially noticeable for mobile manipulators, where limiting acceleration and velocity is necessary in the literature to minimize the jerk caused by sudden movements and the manipulator's sway while the base is in motion.

To reduce this discrepancy, [43] and [44] employ a high-fidelity dynamic model for offline modeling error computation and a low-fidelity kinematic/dynamic model for real-time planning. These methods incorporate pre-computed modeling errors between two models to reduce real-time computational burden and address safety concerns arising from inaccuracies in real-time planning. Additionally, some methods such as [22] and [23] utilize two models of a fixed manipulator with low and high fidelity for real-time motion planning and control. Therefore, a high-fidelity dynamic model for a mobile manipulator is essential for fast and high-precision motion planning in safety-critical scenarios, as such a model accurately characterizes the dynamic coupling between the base and the manipulator. Moreover, from the manufacturers' perspective, understanding how major design parameters analytically affect the dynamics and actuator capabilities is necessary. This understanding allows for the consideration of these factors in subsequent motion planning, ensuring that the planned trajectories are dynamically feasible given the physical limitations of the system and its actuators.

2) Dynamic Modeling: According to the above literature review on motion planning, dynamic modeling for mobile manipulators is essential for accurate and safe motion planning, as well as robot design problems. The literature on dynamic modeling of mobile manipulators

can be generally categorized into two classes, i.e. modeling for a particular type of mobile manipulators or general multiple rigid bodies.

Regarding the dynamic modeling for some particular mobile manipulators, [26] and [25] develop kinematic models for differential unicycles with a two-link/three-link manipulator. [27] combine the kinematics of a nonholonomic cart with the dynamics of a three-link manipulator. [28] discuss how to model the dynamics of a differential unicycle with a two-link manipulator. [45] develop a dynamic model for a nonholonomic cart with a two-flexible-link manipulator by recursive Gibbs–Appell (Lagrange) formulations. The above works model the kinematics or dynamics of the mobile base as a simple differential unicycle, where the contact between the wheels and the ground is not considered. In contrast, [46] investigate the high-fidelity modeling of a four-wheeled robot, including the modeling of contact forces between the wheels and the ground, as well as how the terrain affects the robot's dynamics.

As for the analytical modeling methodologies for general multibody dynamics in the literature, [47] proposes the Articulated Body Algorithm (ABA) to efficiently and analytically generate forward dynamics for a tree-like chain of articulated links. [48] investigate the analytical and symbolic computation of derivatives for the analytical forward dynamics given the ABA. [49] propose Pinocchio, a fast implementation of rigid body dynamics algorithms and their analytical derivatives. Pinocchio can take a robot's `urdf` configuration file as input to generate its analytical dynamics. While Pinocchio is widely used in the research community, it cannot analytically embed actuators into the system dynamics, as they are implicitly included in the mass, inertia, and center-of-mass (CoM) of each link in an `urdf` file. [50] propose a revised ABA algorithm for calculating the forward dynamics of a fixed 6-DOF manipulator, where the motor dynamics are coupled with the manipulator dynamics. Although the dynamics' fidelity is relatively high, combining the slow manipulator dynamics ($\sim$ 100 Hz) with the fast motor dynamics (over 2000 Hz) can significantly slow down any computation that involves dynamics forward propagation as it requires a smaller discretization. Hence, for the sake of computational efficiency, it is necessary to analytically understand the motors' capacity and how they affect the entire robot without including their faster dynamics. Therefore, this paper aims to investigate how to include this information analytically into the dynamics of a mobile manipulator.

B. Paper Organization and Contributions

The rest of this paper is organized as follows: Section II introduces necessary notations and preliminary definitions; Section III presents the definition of a class of mobile manipulators and all its necessary components; Section IV introduces the parameterization of servomotors given motor geometry parameters, as well as presents both the numerical and analytical modeling of motor torque capacity; Section V presents the forward and inverse dynamics modeling of a mobile manipulator and cross-validates the algorithms; Section VI formulates an integrated locomotion and manipulation planning optimization problem with discretization by direct collocation; Section VII presents some numerical experiments for the integrated planning method and a benchmark method; Section VIII showcases a simultaneous actuator design and motion planning framework with some numerical results; Section IX discusses the limitations of the proposed modeling approach and concludes this paper.

The contributions of this paper are summarized as follows:

1) An analytical modeling approach for a manipulator's actuators based on motor design parameters;
2) A differentiable and analytical modeling approach for mobile manipulators given actuators parameterized by motor design parameters;
3) An analytical modeling approach for a motor's maximum speed/torque as a function of its design parameters;
4) An integrated locomotion and manipulation planning approach with motor torque/speed constraints and direct collocation discretization;
5) A framework of simultaneous actuator design and integrated motion planning for mobile manipulator co-design.

II. Preliminaries

This section introduces necessary notations and preliminary definitions.

A. Notations

Denote $\mathbb{R}$ as the real number set and $\mathbb{R}_+$ as the positive real number set. Denote $\mathbb{Z}_+$ as the positive integer set. Denote $\mathbb{E}^m$ as the m-dimensional Euclidean vector space. Denote SO(3) as the Special Orthogonal Group associated with $\mathbb{E}^3$. Denote SE(3) as the Special Euclidean Group associated with $\mathbb{E}^3$. For $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, $\boldsymbol{x} \leq \boldsymbol{y}$ indicates element-wise inequality. Let $\text{col}\{\boldsymbol{v}_1, \cdots, \boldsymbol{v}_a\}$ denote a column stack of elements $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_a$, which may be scalars, vectors or matrices, i.e. $\text{col}\{\boldsymbol{v}_1, \cdots, \boldsymbol{v}_a\} \triangleq \begin{bmatrix} {\boldsymbol{v}_1}^\top & \cdots & {\boldsymbol{v}_a}^\top \end{bmatrix}^\top$. Let $\boldsymbol{0}_n, \boldsymbol{1}_n \in \mathbb{R}^n$ denote a zero and an one vector. Let $\boldsymbol{I}_n \in \mathbb{R}^{n \times n}$ denote an identity matrix. Let $[\![a, b]\!]$ denote a set of all integers between $a \in \mathbb{Z}$ and $b \in \mathbb{Z}$, with both ends included. Denote $\text{diag}(a_1, \cdots, a_n)$ as a diagonal matrix in $\mathbb{R}^{n \times n}$ with diagonal elements $a_1, \cdots, a_n$. Given a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times m}$, $\boldsymbol{A}[a : b, c] \in \mathbb{R}^{(b-a+1)}$ $(1 \leq a \leq b \leq n, 1 \leq c \leq m)$ represents a vector slice of $\boldsymbol{A}$ from c-th column, a-th row until b-th row; $\boldsymbol{A}[a : b, c : d] \in \mathbb{R}^{(b-a+1) \times (d-c+1)}$ $(1 \leq a \leq b \leq n, 1 \leq c \leq d \leq m)$ represents a matrix slice of $\boldsymbol{A}$ from c-th column until d-th column, a-th row until b-th row.

B. Homogeneous Transformation

Let $\boldsymbol{T}_{L_1,L_2} \in \mathrm{SE}(3)$ be a homogeneous transformation. Its subscript indicates that $\boldsymbol{T}_{L_1,L_2}$ is a transformation from an inertia frame $\{L_2\}$ to an inertia frame $\{L_1\}$. Let $\boldsymbol{p}_{L_2}$ be a homogeneous coordinate of a point in $\mathbb{R}^3$ in the frame $\{L_2\}$, then $\boldsymbol{T}_{L_1,L_2} \cdot \boldsymbol{p}_{L_2}$ is the homogeneous coordinate of the same point in the frame $\{L_1\}$. Let $\boldsymbol{T}_{L_2,L_3} \in \mathrm{SE}(3)$ be a homogeneous transformation from an inertia frame $\{L_3\}$ to the inertia frame $\{L_2\}$. Then $\boldsymbol{T}_{L_1,L_3} = \boldsymbol{T}_{L_1,L_2} \cdot \boldsymbol{T}_{L_2,L_3}$ is the homogeneous transformation from the frame $\{L_3\}$ to the frame $\{L_1\}$. Denote $\boldsymbol{M}_{L_1,L_2} \in \mathrm{SE}(3)$ be a homogeneous transformation with the same linear translation of $\boldsymbol{T}_{L_1,L_2}$ and zero rotation, i.e. $\boldsymbol{M}_{L_1,L_2}[1:3,1:3] = \boldsymbol{I}_3$ and $\boldsymbol{M}_{L_1,L_2}[1:3,4] = \boldsymbol{T}_{L_1,L_2}[1:3,4]$.

C. Euclidean Cross Operator

For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$, the cross product $\boldsymbol{a} \times \boldsymbol{b}$ is equivalent to a linear operator $[\boldsymbol{a}\times]$ that maps $\boldsymbol{b}$ to $\boldsymbol{a} \times \boldsymbol{b}$. The operator $\boldsymbol{a}\times$ is given by

$$\boldsymbol{a}\times = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \times = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}. \tag{1}$$

If $\lambda \in \mathbb{R}$, then $(\lambda \boldsymbol{a})\times = \lambda(\boldsymbol{a}\times)$. The matrix in (1) is a 3×3 skew-symmetric matrix representation of vector $\boldsymbol{a}$.

D. Spatial Cross Operator

Given a homogeneous transformation $\boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} \in \mathrm{SE}(3)$, its adjoint representation $[\mathrm{Ad}_{\boldsymbol{T}}]$ is

$$[\mathrm{Ad}_{\boldsymbol{T}}] = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{0}_{3\times 3} \\ [\boldsymbol{p}\times]\boldsymbol{R} & \boldsymbol{R} \end{bmatrix} \in \mathbb{R}^{6\times 6}, \tag{2}$$

where $\boldsymbol{R} \in \mathrm{SO}(3)$ and $\boldsymbol{p} \in \mathbb{R}^3$. $[\mathrm{Ad}_{\boldsymbol{T}}]$ is a linear operator and has the following properties:

$$[\mathrm{Ad}_{\boldsymbol{T}^{-1}}] = [\mathrm{Ad}_{\boldsymbol{T}}]^{-1}, \quad [\mathrm{Ad}_{\boldsymbol{T}_1\boldsymbol{T}_2}] = [\mathrm{Ad}_{\boldsymbol{T}_1}][\mathrm{Ad}_{\boldsymbol{T}_2}]. \tag{3}$$

Denote a spatial twist $\boldsymbol{\mathcal{V}} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{v} \end{bmatrix} \in \mathbb{R}^6$ with angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$ and linear velocity $\boldsymbol{v} \in \mathbb{R}^3$ in its inertia frame $\{L_2\}$. Given a homogeneous transformation $\boldsymbol{T}_{L_1,L_2}$, its adjoint operator applying on a spatial twist, i.e. $[\mathrm{Ad}_{\boldsymbol{T}_{L_1,L_2}}]\boldsymbol{\mathcal{V}}$, is a transformation of the spatial twist from the inertia frame $\{L_2\}$ to the inertia frame $\{L_1\}$.

A spatial cross operator $[\boldsymbol{\mathcal{V}}\times^*]$ is given by

$$[\boldsymbol{\mathcal{V}}\times^*] \triangleq [\mathrm{ad}_{\boldsymbol{\mathcal{V}}}] = \begin{bmatrix} \boldsymbol{\omega}\times & \boldsymbol{0}_{3\times 3} \\ \boldsymbol{v}\times & \boldsymbol{\omega}\times \end{bmatrix} \in \mathbb{R}^{6\times 6}. \tag{4}$$

The spatial cross operator $[\boldsymbol{\mathcal{V}}\times^*]$ can be viewed as a differentiation operator that maps from a spatial force to the derivative of the spatial force.

E. Screw Axis to Homogeneous Transformation

Given a screw axis $\boldsymbol{\mathcal{S}} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{v} \end{bmatrix} \in \mathbb{R}^6$ with $||\boldsymbol{\omega}|| = 1$, for any angular distance $\theta \in \mathbb{R}$ traveled around the axis $\boldsymbol{\mathcal{S}}$, the corresponding homogeneous transformation matrix $\boldsymbol{T}(\boldsymbol{\mathcal{S}}, \theta) \in \mathrm{SE}(3)$ is

$$\boldsymbol{T}(\boldsymbol{\mathcal{S}}, \theta) = e^{[\boldsymbol{\mathcal{S}}]\theta} = \begin{bmatrix} \boldsymbol{R}(\boldsymbol{\omega}, \theta) & \boldsymbol{p}(\boldsymbol{\mathcal{S}}, \theta) \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}, \tag{5}$$

where $\boldsymbol{R}(\boldsymbol{\omega}, \theta) = \boldsymbol{I}_3 + \sin\theta[\boldsymbol{\omega}\times] + (1-\cos\theta)[\boldsymbol{\omega}\times]^2$, and $\boldsymbol{p}(\boldsymbol{\mathcal{S}}, \theta) = (\boldsymbol{I}_3\theta + (1-\cos\theta)[\boldsymbol{\omega}\times] + (\theta - \sin\theta)[\boldsymbol{\omega}\times]^2)\boldsymbol{v}$. The matrix exponential $e^{[\boldsymbol{\mathcal{S}}]\theta}$ converts a rotation around the screw axis into a homogeneous transformation in SE(3).

III. Mobile Manipulator

This paper considers a class of mobile manipulators given in Fig. 1. Suppose that the degree of freedom (DOF) of the manipulator is $n \in \mathbb{Z}_+$. Link 2 is the first (fixed) link of the manipulator. Joint 3 is the first joint of the manipulator. Link 2 is installed rigidly on Link 1, a mobile base, and thus Joint 2 is a 0-DOF fixed joint. Joint 1 is defined as a 3-DOF planar joint, which includes the linear translation along the x-axis and the y-axis and the rotation around the z-axis. Joint 1 connects Link 1 with the ground, which is also denoted as Joint 0 and Link 0. From Joint 3 to Joint $2+n$, each joint is a 1-DOF revolute joint along a screw axis.

As illustrated in Fig. 2, for each Joint k, $k = 3, \cdots, n+2$, there is a motor installed, denoted by Motor $k (k \geq 3)$. In detail, for each $k = 3, \cdots, n+2$, the stator of Motor k is installed rigidly on Link $k-1$ and the rotor of Motor k connects with Link k through Joint k's gearbox.

Remark 1. This paper models the mobile base as one rigid body with one 3-DOF planar joint on SE(2) because this paper focuses on the modeling methodology and algorithm of the forward and inverse dynamics of the entire chain of rigid bodies given motor parameterization. To model the full 6-DOF motion of a mobile base, including pitch and roll movements, one has to define a particular mechanical configuration for the mobile base, including how motors are attached to the wheels, and how wheels are attached to the base and interact with the ground, which introduces extra modeling complexity and is out of the scope of this paper.

A. Joints

Before introducing the forward dynamics, the kinematics of each joint and link need to be defined first. If Joint 1 is just a naive 3-DOF joint allowing movement in SE(2), its motion subspace matrix in Joint 1's inertia frame $\{J_1\}$ is given by

$$\boldsymbol{S}_{J_1} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^\top \in \mathbb{R}^{6\times 3}. \tag{6}$$

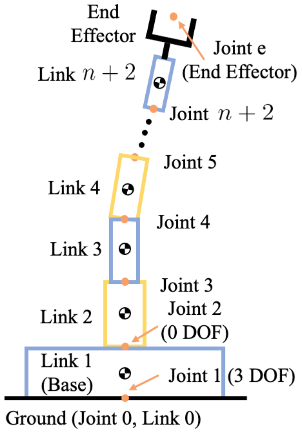


Figure 1. Definition of a mobile manipulator. Joint 3 to Joint $n+2$ are all 1-DOF joints; Joint e is only for notation and has 0-DOF.

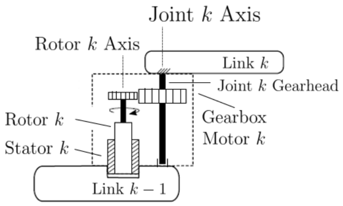


Figure 2. Scheme of a geared motor between Link $k-1$ and Link k ($k = 3, \cdots, n+2$).

The position and velocity variables of Joint 1 are given by

$$\boldsymbol{q}_1 \triangleq \begin{bmatrix} \theta_z & p_x & p_y \end{bmatrix}^\top, \quad \dot{\boldsymbol{q}}_1 \triangleq \begin{bmatrix} \omega_z & v_x & v_y \end{bmatrix}^\top. \tag{7}$$

Since Joint 2 is a fixed joint with 0-DOF, its motion subspace matrix in Joint 2's inertia frame $\{J_2\}$ is given by

$$\boldsymbol{S}_{J_2} = \boldsymbol{0}_6. \tag{8}$$

And its position and velocity variables $\boldsymbol{q}_2$ are given by

$$\boldsymbol{q}_2 \equiv 0 \equiv \dot{\boldsymbol{q}}_2. \tag{9}$$

For Joint $k = 3, \cdots, n+2$, its position and velocity variables are given by $\boldsymbol{q}_k \triangleq \theta_k \in \mathbb{R}$ and $\dot{\boldsymbol{q}}_k \triangleq \dot{\theta}_k \in \mathbb{R}$, where θ_k and $\dot{\theta}_k$ are the joint angular position and velocity around the screw axis of Joint k, respectively. For Joint $k = 3, \cdots, n+2$, its motion subspace matrix in Joint k's inertia frame $\{J_k\}$ is equivalent to its screw axis. For Joint k rotating around its y-axis and z-axis of the frame $\{J_k\}$, $\boldsymbol{S}_{J_k} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^\top$ and $\boldsymbol{S}_{J_k} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^\top$, respectively.

By [47, Chapter 3.5], the apparent derivative of Joint k's motion subspace matrix $\boldsymbol{S}_{J_k}$ is defined by

$$\mathring{\boldsymbol{S}}_{J_k} = \frac{\partial \boldsymbol{S}_{J_k}}{\partial t} + \sum_{j=1}^{n_k} \frac{\partial \boldsymbol{S}_{J_k}}{\partial \boldsymbol{q}_k[j]} \dot{\boldsymbol{q}}_k[j], \tag{10}$$

where n_k is the dimension of Joint k's position variable $\boldsymbol{q}_k$; $\boldsymbol{q}_k[j]$ denotes the j-th element of $\boldsymbol{q}_k$. Denote $\mathring{\boldsymbol{S}}_{J_k}$ as the apparent derivative of $\boldsymbol{S}_{J_k}$ in the frame $\{J_k\}$, and we have

$$\mathring{\boldsymbol{S}}_{J_1} = \boldsymbol{0}_{6\times 3}, \quad \mathring{\boldsymbol{S}}_{J_k} = \boldsymbol{0}_6, \ \forall k = 2, \cdots, n+2. \tag{11}$$

Given the class of mobile manipulators defined in Fig. 1, the linear transformation from $\{J_k\}$ to $\{J_{k-1}\}$ is given by

$$\boldsymbol{M}_{J_{k-1},J_k} = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{p}_{J_{k-1},J_k} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}, k = 3, \cdots, n+2, \tag{12}$$

where $\boldsymbol{p}_{J_{k-1},J_k} \in \mathbb{R}^3$ indicates the position coordinates of Joint k in the frame $\{J_{k-1}\}$.

B. Links

Given a specific kinematic chain, the homogeneous transformation from Link 1's inertia frame $\{L_1\}$ to Link 0's inertia frame $\{L_0\}$ (or the global frame), is given by

$$\boldsymbol{T}_{L_0,L_1}(\boldsymbol{q}_1) = \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 & p_x \\ \sin(\theta_z) & \cos(\theta_z) & 0 & p_y \\ 0 & 0 & 1 & h_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{13}$$

where $h_1 > 0$ is the height of CoM of the mobile base (Link 1) from the global frame. $\boldsymbol{T}_{L_1,L_0} = (\boldsymbol{T}_{L_0,L_1})^{-1}$.

The transformation from Link k's inertia frame $\{L_k\}$ to the frame $\{J_k\}$ is given by

$$\boldsymbol{M}_{J_k,L_k} = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{p}_{J_k,L_k} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}, k = 1, \cdots, n+2, \tag{14}$$

where $\boldsymbol{p}_{J_k,L_k} \in \mathbb{R}^3$ indicates the position coordinates of Link k in the frame $\{J_k\}$. Note that for $k = 1$, $\boldsymbol{p}_{J_1,L_1}[3] = h_1$.

The homogeneous transformation from $\{J_2\}$ to $\{L_1\}$ is given by

$$\boldsymbol{T}_{L_1,J_2} = \boldsymbol{M}_{L_1,J_2} = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{p}_{L_1,J_2} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}, \tag{15}$$

where $\boldsymbol{p}_{L_1,J_2} \in \mathbb{R}^3$ is the position coordinates of Joint 2 in the frame $\{L_1\}$.

The homogeneous transformation from the frame $\{L_2\}$ to the frame $\{L_1\}$ is given by

$$\boldsymbol{T}_{L_1,L_2} = \boldsymbol{T}_{L_1,J_2} \cdot \boldsymbol{T}_{J_2,L_2} = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{p}_{L_1,J_2} + \boldsymbol{p}_{J_2,L_2} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix}. \tag{16}$$

Thus, $\boldsymbol{T}_{L_2,L_1} = (\boldsymbol{T}_{L_1,L_2})^{-1} = \boldsymbol{M}_{L_2,L_1}$ since there is no rotation between Link 1 and Link 2.

For $k = 3, \cdots, n+2$, the transformation of the frame $\{L_{k-1}\}$ to the frame $\{L_k\}$ is given by

$$\begin{aligned} \boldsymbol{M}_{L_k,L_{k-1}} &= (\boldsymbol{M}_{J_k,L_k})^{-1} (\boldsymbol{M}_{J_{k-1},J_k})^{-1} \boldsymbol{M}_{J_{k-1},L_{k-1}} \\ &= (\boldsymbol{M}_{J_{k-1},J_k} \boldsymbol{M}_{J_k,L_k})^{-1} \boldsymbol{M}_{J_{k-1},L_{k-1}}. \end{aligned} \tag{17}$$

C. Inertia of Links

The spatial inertia matrix $\hat{\boldsymbol{G}}_{L_k} \in \mathbb{R}^{6\times 6}$ of Link k in the frame $\{L_k\}$ is given by

$$\hat{\boldsymbol{G}}_{L_k} = \begin{bmatrix} \boldsymbol{I}_{k,\mathrm{G}} & \boldsymbol{0}_{3\times 3} \\ \boldsymbol{0}_{3\times 3} & m_k \boldsymbol{I}_3 \end{bmatrix}, \tag{18}$$

where $\boldsymbol{I}_{k,\mathrm{G}} \in \mathbb{R}^{3\times 3}$ is the inertia tensor in matrix form for Link k; m_k is the mass of Link k.

The stator of Motor $k (k \geq 3)$ is mounted on Link $k-1$ and hence each link's inertia should add the effect from the stator. The mass and inertia of a manipulator's link are typically greater than the ones of a motor stator due to the requirements of the mechanical design of the manipulator. Therefore, to simplify the parameterization of a manipulator, this paper adopts the following assumption on the CoM of Link $k-1$.

Assumption 1 (Constant Link CoM). For $k \geq 3$, the CoM of Link $k-1$ with Motor k's stator installed on the end of Link $k-1$ stays the same as the CoM of Link $k-1$.

Suppose the inertia tensor matrix for the stator of Motor k $(k \geq 3)$ is $\boldsymbol{I}_{k,\mathrm{S}} \in \mathbb{R}^{3\times 3}$ and the stator mass is $m_{k,\mathrm{S}}$. With Assumption 1, the spatial inertia matrix $\boldsymbol{G}_{L_{k-1}}$ of Link $k-1$ in the frame $\{L_{k-1}\}$ is given by

$$\boldsymbol{G}_{L_{k-1}} = \hat{\boldsymbol{G}}_{L_{k-1}} + \begin{bmatrix} \hat{\boldsymbol{I}}_{k,\mathrm{S}} & \boldsymbol{0}_{3\times 3} \\ \boldsymbol{0}_{3\times 3} & m_{k,\mathrm{S}} \boldsymbol{I}_3 \end{bmatrix}, \ k = 3, \cdots, n+2, \tag{19}$$

where $\hat{\boldsymbol{I}}_{k,\mathrm{S}} = \boldsymbol{I}_{k,\mathrm{S}} + m_{k,\mathrm{S}} \cdot \mathrm{diag}(l_{\mathrm{x}}, l_{\mathrm{y}}, l_{\mathrm{z}})$; $l_{\mathrm{x}} = (\boldsymbol{d}_{k,\mathrm{c2t}}[2])^2 + (\boldsymbol{d}_{k,\mathrm{c2t}}[3])^2$, $l_{\mathrm{y}} = (\boldsymbol{d}_{k,\mathrm{c2t}}[1])^2 + (\boldsymbol{d}_{k,\mathrm{c2t}}[3])^2$, $l_{\mathrm{z}} = (\boldsymbol{d}_{k,\mathrm{c2t}}[1])^2 + (\boldsymbol{d}_{k,\mathrm{c2t}}[2])^2$; $\boldsymbol{d}_{k,\mathrm{c2t}} \in \mathbb{R}^3$ is the position coordinate of Link $k-1$'s link tip in the frame $\{L_k\}$, i.e. the location where Motor k's stator is mounted on. For Link 1 and Link $n+2$, $\boldsymbol{G}_{L_1} = \hat{\boldsymbol{G}}_{L_1}$, $\boldsymbol{G}_{L_{n+2}} = \hat{\boldsymbol{G}}_{L_{n+2}}$.

D. Homogeneous Transformation

To calculate rigid-body mechanics, Joint k's motion subspace matrix $\boldsymbol{S}_{J_k}$ in the frame $\{J_k\}$ needs to be transformed into the matrix $\boldsymbol{A}_{L_k}$ in the frame L_k, i.e.

$$\boldsymbol{A}_{L_k} = [\mathrm{Ad}_{(\boldsymbol{M}_{J_k,L_k})^{-1}}] \boldsymbol{S}_{J_k}, \ k = 1, \cdots, n+2. \tag{20}$$

Similarly,

$$\mathring{\boldsymbol{A}}_{L_k} = [\mathrm{Ad}_{(\boldsymbol{M}_{J_k,L_k})^{-1}}] \mathring{\boldsymbol{S}}_{J_k}, \ k = 1, \cdots, n+2. \tag{21}$$

Note that $\boldsymbol{A}_{L_2} = \boldsymbol{0}_6$, $\mathring{\boldsymbol{A}}_{L_k} = \boldsymbol{0}_6$ for all $k = 2, \cdots, n+2$.

Determining the transformation between the screw axis $\boldsymbol{S}_{R_k} \in \mathbb{R}^6$ of Motor k's rotor and the screw axis of Joint k $(k \geq 3)$ requires a specific mechanical configuration of a gearbox, which further introduces the modeling complexity. Due to the absence of detailed modeling of gearboxes, this paper adopts the following assumption. Since the gearbox of robotic manipulators is typically a compact harmonic drive, the transformation between rotor axes and joint axes can be neglected.

Assumption 2 (Coincident Axes). For $k \geq 3$, the screw axis of Joint k is the same as the screw axis of Motor k's rotor.

Assumption 2 assumes that $\boldsymbol{M}_{R_k,J_k} = \boldsymbol{I}_4$ and the frame $\{R_k\}$ is equivalent to the frame $\{J_k\}$, for $k \geq 3$. Nevertheless, one can readily define $\boldsymbol{M}_{R_k,J_k}$ differently according to the modeling of harmonic drives.

The inertia of rotors is modeled as follows. Denote the gear ratio at Joint k as $Z_k > 0$, for $k \geq 3$. The motion subspace matrix of Rotor k in the frame $\{R_k\}$ is given by

$$\boldsymbol{A}_{R_k} = Z_k [\mathrm{Ad}_{\boldsymbol{M}_{R_k,J_k}}] \boldsymbol{S}_{J_k} = Z_k \boldsymbol{S}_{J_k}. \tag{22}$$

According to the definition of apparent derivatives in (10),

$$\mathring{\boldsymbol{A}}_{R_k} = Z_k [\mathrm{Ad}_{\boldsymbol{M}_{R_k,J_k}}] \mathring{\boldsymbol{S}}_{R_k} = Z_k \mathring{\boldsymbol{S}}_{J_k} = \boldsymbol{0}_6, \ \forall k \geq 3. \tag{23}$$

The spatial inertia matrix $\boldsymbol{G}_{R_k} \in \mathbb{R}^{6\times 6}$ of Motor k's rotor in the frame $\{R_k\}$ is given by

$$\boldsymbol{G}_{R_k} = \begin{bmatrix} \boldsymbol{I}_{k,\mathrm{R}} & \boldsymbol{0}_{3\times 3} \\ \boldsymbol{0}_{3\times 3} & m_{k,\mathrm{R}} \boldsymbol{I}_3 \end{bmatrix}, \tag{24}$$

where $\boldsymbol{I}_{k,\mathrm{R}} \in \mathbb{R}^{3\times 3}$ is the rotor's inertia tensor in matrix form; $m_{k,\mathrm{R}}$ is the mass of Motor k's rotor. The transformation from the frame $\{L_{k-1}\}$ to the frame $\{R_k\}$ is

$$\begin{aligned} \boldsymbol{M}_{R_k,L_{k-1}} &= \boldsymbol{M}_{R_k,J_k} \boldsymbol{M}_{J_k,L_k} \boldsymbol{M}_{L_k,L_{k-1}} \\ &= \boldsymbol{M}_{R_k,L_k} \boldsymbol{M}_{L_k,L_{k-1}}, \ \forall k \geq 3. \end{aligned} \tag{25}$$

Finally, for $k \geq 3$,

$$\boldsymbol{T}_{L_k,L_{k-1}} = e^{[-\boldsymbol{A}_{L_k}]\theta_k} \cdot \boldsymbol{M}_{L_k,L_{k-1}}, \tag{26a}$$

$$\boldsymbol{T}_{R_k,L_{k-1}} = e^{[-\boldsymbol{A}_{R_k}]\theta_k} \cdot \boldsymbol{M}_{R_k,L_{k-1}}, \tag{26b}$$

where $e^{[\cdot]\theta_k}$ is given by (5).

The homogeneous transformations from the inertia frame $\{L_k\}$ to the global inertia frame $\{L_0\}$ (equivalent to $\{J_0\}$) are given by

$$\boldsymbol{T}_{L_1} \triangleq \boldsymbol{T}_{L_0,L_1}, \boldsymbol{T}_{L_k} \triangleq \boldsymbol{T}_{L_0,L_k} = \boldsymbol{T}_{L_{k-1}} (\boldsymbol{T}_{L_k,L_{k-1}})^{-1}. \tag{27}$$

The homogeneous transformations from the inertia frame $\{J_k\}$ to the global inertia frame $\{L_0\}$ are given by

$$\boldsymbol{T}_{J_k} \triangleq \boldsymbol{T}_{L_0,J_k} = \boldsymbol{T}_{L_k} (\boldsymbol{M}_{J_k,L_k})^{-1}. \tag{28}$$

Given the homogeneous transformation $\boldsymbol{T}_{J_{n+2},J_{\mathrm{e}}}$ from the end effector frame $\{J_{\mathrm{e}}\}$ to the frame $\{L_{n+2}\}$, the transformation from the frame $\{J_{\mathrm{e}}\}$ to the global inertia frame $\{L0\}$ is given by

$$\boldsymbol{T}_{J_{\mathrm{e}}} \triangleq \boldsymbol{T}_{L_0,J_{\mathrm{e}}} = \boldsymbol{T}_{J_{n+2}} \cdot \boldsymbol{T}_{J_{n+2},J_{\mathrm{e}}}. \tag{29}$$

For every Joint k and the end effector, its position coordinates in the global inertia frame are given by $\boldsymbol{T}_{J_k}[1:3,4]$ and $\boldsymbol{T}_{J_{\mathrm{e}}}[1:3,4]$, respectively.

IV. Motor Parameterization

Surface-mounted permanent magnet synchronous motors (SPMSM) are typically used as the actuators (servomotors) of robotic manipulators. This section introduces the modeling of a class of SPMSMs admitting the design parameterization summarized in Table I. Fig. 3 illustrates the physical meaning of the design variables, where the

Table I
SPMSM Design Parameters

Parameter	Description	Parameter	Description
l	Axial length of core	h_{sy}	Stator yoke
r_{ro}	Outer radius of rotor	w_{tooth}	Width of tooth
r_{so}	Outer radius of stator	b_0	Slot opening
h_{m}	Height of magnet		

Units of all design parameters are mm

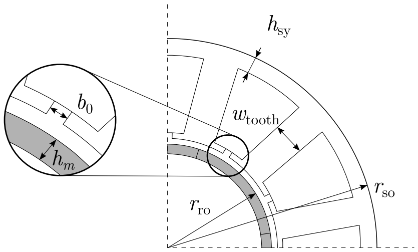


Figure 3. The cross-section of an SPMSM design. The core axial length l is not illustrated in this figure.

axial length l is not shown. Denote an arbitrary motor's design variables as

$$\boldsymbol{\beta} \triangleq \mathrm{col}\{l, r_{\mathrm{ro}}, r_{\mathrm{so}}, h_{\mathrm{m}}, h_{\mathrm{sy}}, w_{\mathrm{tooth}}, b_0\} \in \mathbb{R}^7.$$

This section also introduces how motor parameterization could affect the dynamic modeling and motion planning of the mobile manipulator.

A. Magnetic Equivalent Circuit Modeling

This subsection presents the magnetic equivalent circuit (MEC) modeling [51] for an SPMSM as depicted in Fig. 3. To avoid confusion on whether a parameter is applied universally or just on one particular motor, this subsection introduces an index j to indicate a particular motor. $\boldsymbol{\beta}_j \in \mathbb{R}^7$ denotes the design parameter associated with motor j. $\gcd(a, b)$ denotes the greatest common divisor of two positive integers a and b. The constant parameters for an SPMSM are:

- Number of slots $Q = 12$
- Number of slots per phase $q_1 = Q/3$
- Number of pole pairs $p = 4$
- Number of slots per pole per phase $q_{\mathrm{pm}} = \frac{q_1}{\gcd(q_1, 2p)}$
- Number of winding turns per tooth $n_{\mathrm{s}} = 50$
- Number of coils connected in parallel $C_{\mathrm{p}} = 1$
- Gear ratio at the j-th joint $Z_j = 50\ \forall j$
- Height of tooth tip $h_{\mathrm{tip}} = 2$ mm
- Width of air gap $\delta = 0.5$ mm
- Magnet width in electric angle $\alpha_{\mathrm{m}} = \pi$
- Remanent flux density of the magnet $B_{\mathrm{r}} = 1.38$ T
- Maximum limitation for flux density $B_{\max} = 1.5$ T
- Mass density of iron $\rho_{\mathrm{iron}} = 7.8 \cdot 10^{-6}$ kg/mm^3
- Mass density of copper $\rho_{\mathrm{cu}} = 8.93 \cdot 10^{-6}$ kg/mm^3
- Electric resistivity of copper winding $\rho_{\mathrm{e}} = 1.8 \cdot 10^{-5}$ Ω·mm
- Permeability of air $\mu_0 = 4\pi \cdot 10^{-7}$ N/A^2
- Relative recoil permeability of the magnet $\mu_{\mathrm{r}} = 1.05$
- Filling factor $f_{\mathrm{f}} = 0.55$

1) Geometric Parameters: According to Fig. 3, the expression for the slot height is

$$h_{\mathrm{ss},j} = r_{\mathrm{so},j} - h_{\mathrm{sy},j} - r_{\mathrm{ro},j} - \delta - h_{\mathrm{tip}}. \tag{30}$$

For a rectangular tooth cross-section, one can compute the slot width as $b_{\mathrm{ss},j} = A_{\mathrm{slot},j}/h_{\mathrm{ss},j}$, where $A_{\mathrm{slot},j}$ is the slot area, i.e.

$$A_{\mathrm{slot},j} = \frac{\pi((r_{\mathrm{so},j} - h_{\mathrm{sy},j})^2 - (r_{\mathrm{ro},j} + \delta + h_{\mathrm{tip},j})^2)}{Q} - w_{\mathrm{tooth},j} h_{\mathrm{ss},j}. \tag{31}$$

The cross-section area of the stator core is given by

$$A_{\mathrm{so},j} = \pi r_{\mathrm{so},j}^2 - \pi (r_{\mathrm{ro},j} + \delta)^2 - Q(A_{\mathrm{slot},j} + b_0 h_{\mathrm{tip}}). \tag{32}$$

Thus the volume of the stator core is given by

$$V_j = A_{\mathrm{so},j} l_j. \tag{33}$$

The copper area is given by $A_{\mathrm{cu},j} = A_{\mathrm{slot},j} f_f$. For concentrated windings and assuming one winding is a complete turn around a tooth, the area of a single coil is given by $A_{\mathrm{coil},j} = A_{\mathrm{cu},j}/(2n_{\mathrm{s}})$. The minimal wire diameter is

$$D_{\mathrm{wire},j} = \sqrt{4 A_{\mathrm{coil},j}/\pi}. \tag{34}$$

The arc span per slot can be determined by $\tau_{\mathrm{s},j} = 2\pi(r_{\mathrm{ro},j} + \delta)/Q$. The average length of the coil end-winding $l_{\mathrm{end,av},j}$ and the total coil length $l_{\mathrm{coil},j}$ are given by

$$l_{\mathrm{end,av},j} = (w_{\mathrm{tooth},j}(2 - \pi/2) + \pi\tau_{\mathrm{s},j}/2)/2,\ l_{\mathrm{coil},j} = 2l_j + 2l_{\mathrm{end,av},j}.$$

Then the weight of the stator and rotor are given by:

$$\begin{aligned} m_{\mathrm{rotor},j} &= \rho_{\mathrm{iron}} \pi r_{\mathrm{ro},j}^2 l_j, \\ m_{\mathrm{stator},j} &= \rho_{\mathrm{iron}} \pi r_{\mathrm{so},j}^2 l_j - \rho_{\mathrm{iron}} \pi (r_{\mathrm{ro},j} + \delta)^2 l_j \\ &\quad - \rho_{\mathrm{iron}} A_{\mathrm{slot},j} l_j Q + \rho_{\mathrm{cu}} A_{\mathrm{coil},j} l_{\mathrm{coil},j} n_{\mathrm{s}} Q. \end{aligned} \tag{35}$$

Without loss of generality, define the x-axis as the central axis of each rotor or stator, i.e. x-axis coincides with the axial length of core l_j; consequently, define the y-axis and z-axis by following the right-hand rule and the two axes indicate the central radius. All three axes originate at the centroid of the rotor or stator, i.e. the center of the axial length l_j. Since each rotor is a solid cylinder, the moment of inertia about three principal axes of each rotor is given by:

$$\begin{aligned} I_{\mathrm{xx,r},j} &= \tfrac{1}{2} \rho_{\mathrm{iron}} \pi r_{\mathrm{ro},j}^4 l_j = \tfrac{1}{2} m_{\mathrm{rotor},j} r_{\mathrm{ro},j}^2, \\ I_{\mathrm{yy,r},j} &= I_{\mathrm{zz,r},j} = \tfrac{1}{12} \rho_{\mathrm{iron}} \pi r_{\mathrm{ro},j}^2 l_j (3 r_{ro,j}^2 + l_j^2). \end{aligned} \tag{36}$$

To simplify the inertia calculation for stators, each stator is simplified as a hollow cylinder with outer radius $r_{\mathrm{so},j}$ and inner radius $r_{\mathrm{ro},j} + \delta$. Then the moment of inertia about three principal axes of each stator is given by:

$$\begin{aligned} I_{\mathrm{xx,s},j} &= \tfrac{1}{2} m_{\mathrm{stator},j} (r_{\mathrm{so},j}^2 + (r_{\mathrm{ro},j} + \delta)^2), \\ I_{\mathrm{yy,s},j} &= I_{\mathrm{zz,s},j} = \tfrac{1}{12} m_{\mathrm{stator},j} (3(r_{\mathrm{so},j}^2 + (r_{\mathrm{ro},j} + \delta)^2) + l_j^2). \end{aligned} \tag{37}$$

For a motor installed on a joint with a known screw axis, one can use $I_{\mathrm{xx,r},j}, I_{\mathrm{xx,s},j}$ defined in (36) and (37) as the inertia around the screw axis, and use the rest as the inertia around the other two axes.

2) Resistance: The resistance per tooth is given by $R_{1,j} = (n_{\mathrm{s}}^2 \rho_{\mathrm{e}} l_{\mathrm{coil},j})/(A_{\mathrm{slot},j} f_f)$. Using this, the phase resistance can be calculated as

$$R_j = q_1 R_{1,j}/C_{\mathrm{p}}^2. \tag{38}$$

3) Permeance: The permeance of the magnetic path across the air gap and the slot opening, denoted by $p_{\mathrm{g},j}$ and $p_{\mathrm{so},j}$, are given by:

$$p_{\mathrm{g},j} = \frac{2\pi r_{\mathrm{ro},j}\mu_0 l/Q}{\delta + h_{\mathrm{m},j}/\mu_r}, \quad p_{\mathrm{so},j} = \frac{\mu_0 h_{\mathrm{tip}} l_j}{b_{0,j}}.$$

The permeance of the magnetic path that curves from tip to tip is given by $p_{\mathrm{tt},j} = \frac{\mu_0(\delta + h_{\mathrm{m},j}) l_j}{\pi(\delta + h_{\mathrm{m},j})/2 + b_{0,j}}$.

4) Inductance: For an SPMSM, its d-axis and q-axis inductance are equivalent to each other and given by

$$L_{\mathrm{d},j} = L_{\mathrm{q},j} = q_1 n_{\mathrm{s}}^2 L_{1,j}/C_{\mathrm{p}}^2, \tag{39}$$

where $L_{1,j}$ is the inductance per turn and per tooth, given by $L_{1,j} = p_{\mathrm{g},j} + 3p_{\mathrm{so},j} + 3p_{\mathrm{tt},j}$.

5) Flux: To proceed with the calculation, it is necessary to determine Carter's coefficient denoted by $k_{\mathrm{C},j}$, given by

$$k_{\mathrm{C},j} = \frac{t_{\mathrm{pitch},j}}{t_{\mathrm{pitch},j} - \gamma_j \delta}, \ \gamma_j = \frac{(b_{0,j}/\delta)^2}{5 + b_{0,j}/\delta}, \ t_{\mathrm{pitch},j} = \frac{2\pi r_{\mathrm{ro},j}}{Q}. \tag{40}$$

Then, the magnetic flux density across the gap is given by $B_{\mathrm{g},j} = B_{\mathrm{r}} \frac{h_{\mathrm{m},j}/\mu_r}{h_{\mathrm{m},j}/\mu_r + \delta k_{\mathrm{C},j}}$. The flux density corresponding to the first harmonics can be calculated as $B_{\mathrm{g},1,j} = 4B_{\mathrm{g},j}/\pi$. Consequently, the flux per tooth per single turn is given by

$$\Phi_{1,j} = B_{\mathrm{g},1,j} l_j 2\pi r_{\mathrm{ro},j}/Q. \tag{41}$$

In the absence of skewness, the permanent flux linkage is given by

$$\Phi_{\mathrm{pm},j} = k_{\mathrm{w}} n_{\mathrm{s}} \Phi_{1,j} q_1 / C_{\mathrm{p}}, \tag{42}$$

where $k_{\mathrm{w}} = k_{\mathrm{p}} k_{\mathrm{d}}$ denotes the winding factor, and

$$k_{\mathrm{p}} = \sin(\pi p/Q), \quad k_{\mathrm{d}} = \frac{\sin(\pi/6)}{q_{\mathrm{pm}} \sin(\pi/(6q_{\mathrm{pm}}))}. \tag{43}$$

B. Torque Capacity Modeling

This subsection introduces the modeling of SPMSM torque capacity, which is crucial to motion planning and control of the mobile manipulator. Particularly, the maximum and minimum torques are modeled as an analytical function of motor speed and design parameters.

The dynamics of a permanent magnet synchronous motor (PMSM) are governed by ordinary differential equations (ODEs) as follows:

$$\begin{aligned} \dot{i}_{\mathrm{d},j} &= (-R_j i_{\mathrm{d},j} + p\omega_j L_{\mathrm{q},j} i_{\mathrm{q},j} + u_{\mathrm{d},j})/L_{\mathrm{d},j}, \\ \dot{i}_{\mathrm{q},j} &= (-R_j i_{\mathrm{q},j} - (L_{\mathrm{d},j} i_{\mathrm{d},j} + \Phi_{\mathrm{pm},j}) p\omega_j + u_{\mathrm{q},j})/L_{\mathrm{q},j}, \\ \tau_j &= 1.5p(\Phi_{\mathrm{pm},j} i_{\mathrm{q},j} + (L_{\mathrm{d},j} - L_{\mathrm{q},j}) i_{\mathrm{q},j} i_{\mathrm{d},j}), \end{aligned} \tag{44}$$

where $i_{\mathrm{d},j}, i_{\mathrm{q},j}$ are the current in d- and q-axis; ω_j is the rotor speed of the motor (motor speed in short); $u_{\mathrm{d},j}, u_{\mathrm{q},j}$ are the voltage in d- and q-axis; τ_j denotes the electric torque produced by the motor.

Remark 2. For an SPMSM, $L_{\mathrm{d},j} = L_{\mathrm{q},j}$ and thus the electric torque τ_j in (44) can be simplified as $\tau_j = 1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q},j}$.

For illustration purposes, this section presents the motor operation when $\tau_j > 0$ and $\omega_j > 0$, i.e. the first quadrant. The torque capacity for the other three quadrants is equivalent to the one in the first quadrant. A motor control strategy must be determined first to model the motor torque capacity. The following assumption defines a common motor control strategy.

Assumption 3. For an arbitrary SPMSM, the current signals i_{d} and i_{q} and the DC bus voltage are measured. The motor controller first follows the maximum torque per ampere (MPTA) strategy before hitting voltage constraints. After the motor hits either the current or voltage constraint, it follows the maximum torque per voltage (MPTV) strategy.

For an arbitrary SPMSM, given the max DC bus voltage $V_{\max,j}$ and the max current $I_{\max,j}$, one can estimate the feasible torque region in the speed-torque plane. The max voltage dropping to overcome back-EMF (counter-electromotive force) can be calculated as

$$V_{\mathrm{dq},\max,j} = V_{\max,j}/\sqrt{3} - R_j I_{\max,j} \geq 0. \tag{45}$$

Without flux weakening, i.e. $i_{\mathrm{d},j} = 0$, the corner electric speed before hitting the voltage constraint while keeping the max torque ($i_{\mathrm{q},j} = I_{\max,j}$) is given by

$$\omega_{\mathrm{ce},j} = \frac{V_{\mathrm{dq},\max,j}}{\sqrt{(L_{\mathrm{q},j} I_{\max,j})^2 + \Phi_{\mathrm{pm},j}^2}}. \tag{46}$$

Given $i_{\mathrm{d},j}$, $i_{\mathrm{q},j}$ and ω_j, the back-EMF is given by

$$V_{\mathrm{bemf},j} = p\omega_j \sqrt{(L_{\mathrm{q},j} i_{\mathrm{q},j})^2 + (\Phi_{\mathrm{pm},j} + L_{\mathrm{d},j} i_{\mathrm{d},j})^2}. \tag{47}$$

Remark 3. Determining the feasibility of an operating point (ω_j, τ_j) is equivalent to find a feasible pair $(i_{\mathrm{d},j}, i_{\mathrm{q},j})$ such that the following voltage and current constraints hold:

$$\begin{aligned} p\omega_j \sqrt{(L_{\mathrm{q},j} i_{\mathrm{q},j})^2 + (\Phi_{\mathrm{pm},j} + L_{\mathrm{d},j} i_{\mathrm{d},j})^2} &\leq V_{\mathrm{dq},\max,j}, \\ i_{\mathrm{d},j}^2 + i_{\mathrm{q},j}^2 &\leq I_{\max,j}^2, \end{aligned} \tag{48}$$

where $i_{\mathrm{q},j} = \tau_j/(1.5p\Phi_{\mathrm{pm},j}) \leq I_{\max,j}$.

Denote $\omega_{\mathrm{e},j} \triangleq p\omega_j$ as the electric speed. If $\omega_{\mathrm{e},j} \leq \omega_{\mathrm{ce},j}$ and $\tau_j \leq 1.5p\Phi_{\mathrm{pm},j} I_{\max,j}$, then the voltage constraint is always satisfied and the feasible solution with max torque per ampere is $i_{\mathrm{d},j} = 0$ and $i_{\mathrm{q},j} = \tau_j/(1.5p\Phi_{\mathrm{pm},j})$; if $\omega_{\mathrm{e},j} \leq \omega_{\mathrm{ce},j}$ and $\tau_j > 1.5p\Phi_{\mathrm{pm},j} I_{\max,j}$, the operating point is infeasible. Denote the corner speed when the maximum constant torque $1.5p\Phi_{\mathrm{pm},j} I_{\max,j}$ is about to decrease as

$$\omega_{\mathrm{r},j} \triangleq \omega_{\mathrm{ce},j}/p, \tag{49}$$

where $\omega_{\mathrm{ce},j}$ is given by (46).

With $\omega_{\mathrm{e},j} > \omega_{\mathrm{ce},j}$, one needs to infer the max allowable q-axis current $i_{\mathrm{q},\mathrm{lim},j}$ which is restricted by either the max voltage or max current constraint. For the operating point (ω_j, τ_j), it requires $i_{\mathrm{q},j} = \tau_j/(1.5p\Phi_{\mathrm{pm},j})$. We first consider the special case when both current and voltage constraints are active, the solution of which is obtained by solving two unknown $i_{\mathrm{d},j}, i_{\mathrm{q},j}$ from two equations

$$p\omega_j\sqrt{(L_{\mathrm{q},j}i_{\mathrm{q},j})^2+(\Phi_{\mathrm{pm},j}+L_{\mathrm{d},j}i_{\mathrm{d},j})^2}=V_{\mathrm{dq,max},j}, \\ i_{\mathrm{d},j}^2+i_{\mathrm{q},j}^2=I_{\mathrm{max},j}^2. \tag{50}$$

Its solution is given by

$$i_{\mathrm{d,lim},j}=\frac{(V_{\mathrm{dq,max},j}/\omega_{\mathrm{e},j})^2-(L_{\mathrm{d},j}I_{\mathrm{max},j})^2-\Phi_{\mathrm{pm},j}^2}{2\Phi_{\mathrm{pm},j}L_{\mathrm{d},j}}, \\ i_{\mathrm{q,lim},j}=\sqrt{I_{\mathrm{max},j}^2-i_{\mathrm{d,lim},j}^2}. \tag{51}$$

When $\omega_j=\omega_{\mathrm{r},j}$, $i_{\mathrm{d,lim},j}=0$; when $\omega_j>\omega_{\mathrm{r},j}$, $i_{\mathrm{d,lim},j}$ is always negative to weaken the permanent flux. Since the current is bounded by $i_{\mathrm{d},j}^2+i_{\mathrm{q},j}^2\leq I_{\mathrm{max},j}^2$, the magnitude of $i_{\mathrm{d},j}$ cannot exceed $I_{\mathrm{max},j}$. According to (51) with $\omega_{\mathrm{e},j}:=p\omega_j$, $i_{\mathrm{d,lim},j}(\omega_j)$ is a monotonically decreasing function to the motor speed ω_j. Therefore, one can analytically find the crossover point $\omega_{\mathrm{max},j}$ for $i_{\mathrm{d,lim},j}(\omega_{\mathrm{max},j})=-I_{\mathrm{max},j}$, which yields

$$\omega_{\mathrm{max},j}\triangleq\frac{V_{\mathrm{dq,max},j}}{p|\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}|}. \tag{52}$$

$\omega_{\mathrm{max},j}$ is the maximum motor speed given both the current and voltage constraints only when $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}>0$. From (52), it is obvious that when $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}=0$, $\omega_{\mathrm{max},j}=\infty$. In other words, the voltage and current constraints will not limit the motor speed. When $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}<0$, $\omega_{\mathrm{max},j}=\infty$. The following two figures illustrate the reason as (52) cannot.

First, a concrete example of $i_{\mathrm{d,lim},j}$ with three different signs of $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}$ is shown in Fig. 4, where only $I_{\mathrm{max},j}$ changes as 5 A, 21.67 A, and 30 A for Fig. 4a - Fig. 4c, respectively. When $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}>0$, there exists a finite $\omega_{\mathrm{max},j}$. When $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}=0$, $i_{\mathrm{d,lim},j}(\omega_j)$ asymptotically converges to $I_{\mathrm{max},j}$ as ω_j goes to infinity. In Fig. 4c, ω_j exceeds the intersection (its value can be calculated by (52)) when $i_{\mathrm{d,lim},j}$ reaches $I_{\mathrm{max},j}=30$ A. When ω_j keeps increasing after this intersection, by Assumption 3, the motor cannot operate at both the boundary of the current and voltage constraint. Thus, one cannot use (50) to jointly determine $i_{\mathrm{d},j}$ and $i_{\mathrm{q},j}$, and its further derivation (51) and (52) cannot be used as well. Fig. 5c illustrates the reason in this particular case by showing the current constraint and the contours of the voltage constraint. Before explaining that, it is necessary to introduce the voltage constraint in the $i_{\mathrm{d},j}$-$i_{\mathrm{q},j}$ plane.

The max torque for a given $\omega_j\geq\omega_{\mathrm{r},j}$ always achieves at the boundary of the maximum voltage constraint. This is because if the max voltage constraint is inactive while the current constraint is hit, then from (47), one can always reduce $i_{\mathrm{d},j}$ and increase $i_{\mathrm{q},j}$ to produce a larger torque while ensuring the current constraints. On the other hand, the max current constraint is not necessarily active.

Remark 4. The voltage constraint can be rearranged as follows

$$i_{\mathrm{q},j}^2+(\frac{\Phi_{\mathrm{pm},j}}{L_{\mathrm{d},j}}+i_{\mathrm{d},j})^2=\frac{V_{\mathrm{dq,max},j}^2}{\omega_{\mathrm{e},j}^2L_{\mathrm{d},j}^2}, \tag{53}$$

which is a circle centered at $(-\frac{\Phi_{\mathrm{pm},j}}{L_{\mathrm{d},j}},0)$ in the $i_{\mathrm{d},j}$-$i_{\mathrm{q},j}$ plane with a radius of $\frac{V_{\mathrm{dq,max},j}}{\omega_{\mathrm{e},j}L_{\mathrm{d},j}}$. On the other hand, the current constraint is a circle centered at the origin with a radius $I_{\mathrm{max},j}$.

The contours of voltage constraints (53) are shown in Fig. 5, where the red crosses represent the center $(-\frac{\Phi_{\mathrm{pm},j}}{L_{\mathrm{d},j}},0)$; the blue circles represent the current constraints; green solid/dashed lines represent the max torque loci while increasing the motor speed. One can see that if $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}-I_{\mathrm{max},j}\geq0$, the maximum torque always achieves at the boundary of the current and voltage constraints because there are always intersections between the current constraint (blue circle) and the voltage constraint (contours). When $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}-I_{\mathrm{max},j}<0$, the max torque does not achieve at the maximum current, but inside along the green line inside the blue circle. At some points, there is no intersection between the blue circle and the contours. This indicates that one cannot calculate $i_{\mathrm{d},j}$ and $i_{\mathrm{q},j}$ by (50), i.e. on both the voltage and current boundary, but only on the voltage boundary (every contour).

To summarize, one can determine the feasible operation region, i.e. the torque capacity, of a motor numerically using the following algorithms. Algorithm 1 converts every arbitrary operating point (ω_j,τ_j) into $i_{\mathrm{d},j}$ and $i_{\mathrm{q},j}$ when feasible, and returns a false flag when infeasible. Note that given motor design parameters $\boldsymbol{\beta}_j$, one can calculates R_j, $L_{\mathrm{d},j}$, $L_{\mathrm{q},j}$, $\Phi_{\mathrm{pm},j}$ from Section IV-A. Algorithm 2 presents how to generate a motor feasible operation map given an arbitrary range of motor speed $[0,\hat{\omega}_{\mathrm{max},j}]$ and torque $[0,\hat{\tau}_{\mathrm{max},j}]$. Keeping all the parameters the same but given different $I_{\mathrm{max},j}$, Fig. 6 presents the motor operation map and torque capacity given three different signs of $\Phi_{\mathrm{pm},j}-L_{\mathrm{d},j}I_{\mathrm{max},j}$. The red region represents the feasible operation range, whereas the white region represents the infeasible range. The boundary between the white and the red region is the maximum motor torque at each motor speed obtained numerically by Algorithm 2. The dashed lines/curves are some analytical functions for torque bounds; the vertical dash-dot lines indicate some critical speeds; the details of the analytical torque bound functions are given in Section IV-C.

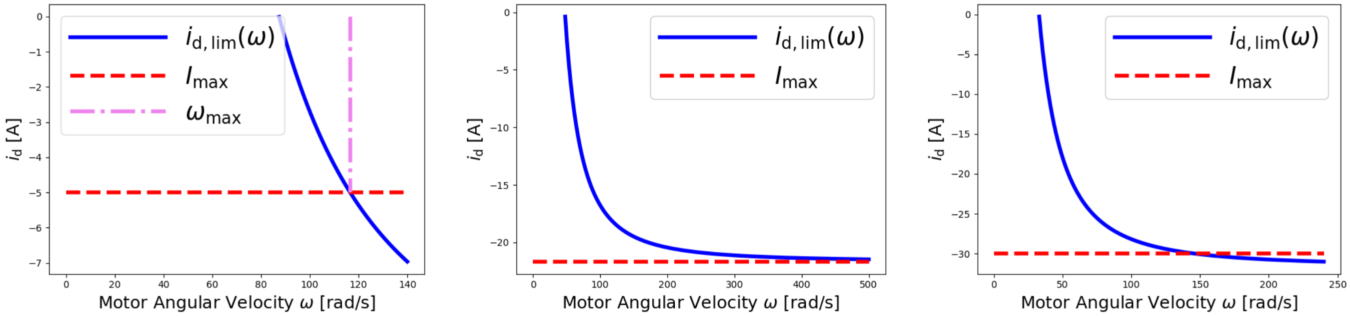
C. Analytical Torque Bound Modeling

Given the motor control strategy defined in Assumption 3 and the derivation in Section IV-B, the motor torque bound can be modeled analytically, which can be categorized into three cases, depending on the sign of $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}-I_{\mathrm{max},j}$.

When $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}-I_{\mathrm{max},j}>0$, the maximum motor torque $\tau_{\mathrm{max},j}(\omega_j,\boldsymbol{\beta}_j)$, as a function of motor speed and motor design parameters, is given by

$$\tau_{\mathrm{max},j}=\begin{cases}1.5p\Phi_{\mathrm{pm},j}I_{\mathrm{max},j}, & \text{if } |\omega_j|\leq\omega_{\mathrm{r},j}\\ 1.5p\Phi_{\mathrm{pm},j}i_{\mathrm{q,lim},j}, & \text{if } |\omega_j|\in[\omega_{\mathrm{r},j},\omega_{\mathrm{max},j}]\end{cases} \tag{54}$$

where $\omega_{\mathrm{r},j}$, $i_{\mathrm{q,lim},j}$, and $\omega_{\mathrm{max},j}$ are given by (49), (51) and (52), respectively. This piecewise-defined function

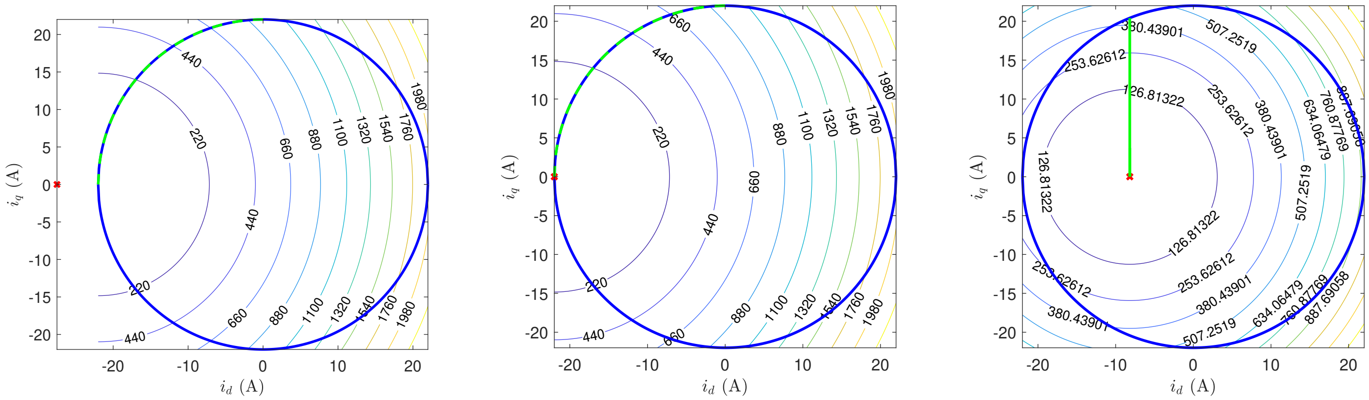


(a) $i_{\mathrm{d,lim},j}$ for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} > I_{\max,j}$.

(b) $i_{\mathrm{d,lim},j}$ for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} = I_{\max,j}$.

(c) $i_{\mathrm{d,lim},j}$ for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$.

Figure 4. $i_{\mathrm{d,lim},j}$ as a function of ω_j in all cases. The blue lines represent the function $i_{\mathrm{d,lim},j}(\omega_j)$; the red dashed lines represent $I_{\max,j}$; the pink dash-dot lines represent $\omega_{\max,j}$ from (52). The intersection in (a) indicates the maximum motor speed constrained by the maximum current $I_{\max,j}$.

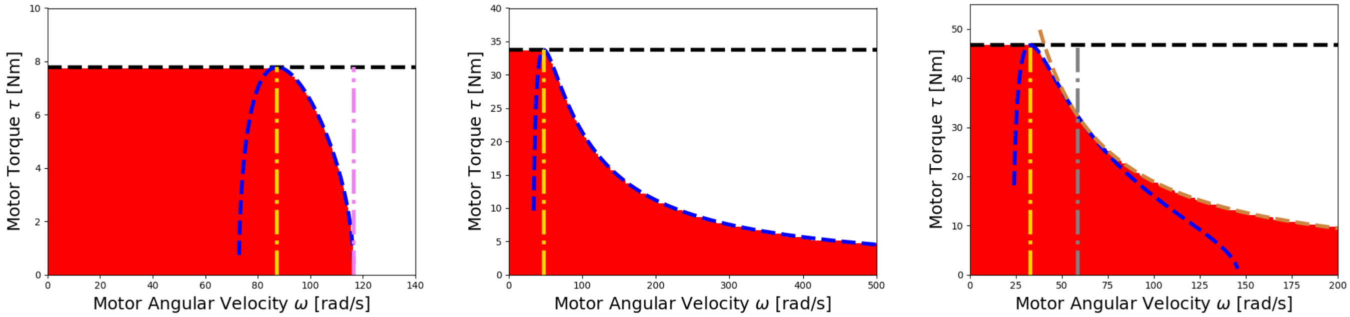


(a) Visualization of current and voltage constraints for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} > I_{\max,j}$.

(b) Visualization of current and voltage constraints for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} = I_{\max,j}$.

(c) Visualization of current and voltage constraints for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$.

Figure 5. Visualization of current and voltage constraints in all cases given the same motor speed. Blue circles represent current constraints; contours represent voltage constraints; red cross marks represent centers of the voltage constraint contours; green (dashed) lines represent the max torque loci while increasing the motor speed.



(a) Motor operation map for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} > I_{\max,j}$.

(b) Motor operation map for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} = I_{\max,j}$.

(c) Motor operation map for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$.

Figure 6. Motor operation map in all cases. The red region represents the feasible operation range; the white region represents the infeasible range. The meaning of the dashed and dash-dot lines/curves is given in Section IV-C.

Algorithm 1: Conversion of motor speed and torque to motor currents

```
Input: $\boldsymbol{\beta}_j$, $V_{\max,j}$, $I_{\max,j}$, number of pole pairs $p$
def omgTau2currents($\omega_j$, $\tau_j$):
    $V_{\mathrm{dq,max},j} \leftarrow V_{\max,j}/\sqrt{3} - R_j I_{\max,j}$, flag $\leftarrow$ True
    $\omega_{\mathrm{r},j} \leftarrow \frac{V_{\mathrm{dq,max},j}}{p\sqrt{(L_{\mathrm{q},j} I_{\max,j})^2 + \Phi^2_{\mathrm{pm},j}}}$, $\omega_{\mathrm{e},j} \leftarrow p\omega_j$
    if $\omega_j \leq \omega_{\mathrm{r},j}$ then
        $i_{\mathrm{q},j} \leftarrow \tau_j/(1.5p\Phi_{\mathrm{pm},j})$, $i_{\mathrm{d},j} \leftarrow 0$
        if $i_{\mathrm{q},j} > I_{\max,j}$ then  flag $\leftarrow$ False,
          $i_{\mathrm{d},j}, i_{\mathrm{q},j} \leftarrow$ None
    else
        $i_{\mathrm{d,lim},j} \leftarrow \frac{(V_{\mathrm{dq,max},j}/\omega_{\mathrm{e},j})^2 - (L_{\mathrm{d},j} I_{\max,j})^2 - \Phi^2_{\mathrm{pm},j}}{2\Phi_{\mathrm{pm},j} L_{\mathrm{d},j}}$
        $i_{\mathrm{q,lim,V},j} \leftarrow V_{\mathrm{dq,max},j}/(\omega_{\mathrm{e},j} L_{\mathrm{q},j})$,
          $i_{\mathrm{q,t},j} \leftarrow \tau_j/(1.5p\Phi_{\mathrm{pm},j})$
        if $i_{\mathrm{d,lim},j} \leq -I_{\max,j}$ then
            if $i_{\mathrm{q,t},j} > i_{\mathrm{q,lim,V},j}$ or
              $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} \geq 0$ then
                flag $\leftarrow$ False, $i_{\mathrm{d},j}, i_{\mathrm{q},j} \leftarrow$ None
            else  $i_{\mathrm{q},j} \leftarrow i_{\mathrm{q,t},j}$, $i_{\mathrm{d},j} \leftarrow -\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}$
        else
            $i_{\mathrm{d,lim},j} \leftarrow$
              $\mathrm{sign}(i_{\mathrm{d,lim},j})\min(|i_{\mathrm{d,lim},j}|, I_{\max,j})$
            if $i_{\mathrm{d,lim},j} L_{\mathrm{d},j} + \Phi_{\mathrm{pm},j} < 0$ then
                $i_{\mathrm{d,lim},j} \leftarrow -\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}$
                $i_{\mathrm{q,lim,C},j} \leftarrow \sqrt{I^2_{\max,j} - i^2_{\mathrm{d,lim},j}}$
                $i_{\mathrm{q,lim},j} \leftarrow \min(i_{\mathrm{q,lim,V},j}, i_{\mathrm{q,lim,C},j})$
            else  $i_{\mathrm{q,lim},j} \leftarrow \sqrt{I^2_{\max,j} - i^2_{\mathrm{d,lim},j}}$
            if $i_{\mathrm{q,t},j} > i_{\mathrm{q,lim},j}$ then
                flag $\leftarrow$ False, $i_{\mathrm{d},j}, i_{\mathrm{q},j} \leftarrow$ None
            else  $i_{\mathrm{q},j} \leftarrow i_{\mathrm{q,t},j}$, $i_{\mathrm{d},j} \leftarrow i_{\mathrm{d,lim},j}$
    return $i_{\mathrm{d},j}$, $i_{\mathrm{q},j}$, flag
```

Algorithm 2: Generation of motor operation map and torque capacity

```
Input: $\boldsymbol{\beta}_j$, $V_{\max,j}$, $I_{\max,j}$, number of pole pairs $p$,
       $N_\omega \in \mathbb{Z}_+$, $N_\tau \in \mathbb{Z}_+$, $\hat{\omega}_{\max,j} > 0$, $\hat{\tau}_{\max,j} > 0$
$\boldsymbol{M} \leftarrow$ zeros$(N_\tau, N_\omega)$, $\boldsymbol{v}_\omega \leftarrow$ linspace$(0, \hat{\omega}_{\max,j}, N_\omega)$,
  $\boldsymbol{v}_\tau \leftarrow$ linspace$(0, \hat{\tau}_{\max,j}, N_\tau)$
for $i_\tau$ in range$(N_\tau)$ do
    for $i_\omega$ in range$(N_\omega)$ do
        $\omega_j \leftarrow \boldsymbol{v}_\omega[i_\omega]$, $\tau_j \leftarrow \boldsymbol{v}_\tau[i_\tau]$
        $i_{\mathrm{d},j}, i_{\mathrm{q},j}$, flag $\leftarrow$ omgTau2currents($\omega_j$, $\tau_j$)
        if flag is True then  $\boldsymbol{M}[i_\tau, i_\omega] = 1$
return $\boldsymbol{M}$
```

is visualized in Fig. 6a. The yellow dash-dot vertical line represents $\omega_{\mathrm{r},j}$, which is the motor corner speed to start decreasing the constant maximum torque. The purple dash-dot vertical line represents $\omega_{\max,j}$, which is the maximum motor speed given current and voltage constraints. The black dashed horizontal line represents the constant maximum torque given by the first row of (54). The blue dashed curve represents the maximum torque, which decreases as the motor speed increases, according to the second row of (54). Fig. 6a verifies the analytical torque bound given the numerical result from Algorithm 2. Even though the particular shape of the maximum torque in $[\omega_{\mathrm{r},j}, \omega_{\max,j}]$ is concave in Fig. 6a, with $I_{\max,j}$ increasing but still less than $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}$, the shape could become convex.

When $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} = 0$, the maximum motor torque is given by

$$\tau_{\max,j} = \begin{cases} 1.5p\Phi_{\mathrm{pm},j} I_{\max,j}, & \text{if } |\omega_j| \leq \omega_{\mathrm{r},j} \\ 1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q,lim},j}, & \text{if } |\omega_j| \in [\omega_{\mathrm{r},j}, \infty) \end{cases} \tag{55}$$

where $\omega_{\mathrm{r},j}$ and $i_{\mathrm{q,lim},j}$ are given by (49) and (51), respectively. Note that the only difference between (54) and (55) is whether there exists a maximum motor speed $\omega_{\max,j}$ or not. This piecewise-defined function is visualized in Fig. 6b. The yellow dash-dot vertical line represents $\omega_{\mathrm{r},j}$. According to (52), there is no $\omega_{\max,j}$ or $\omega_{\max,j} = \infty$. The black dashed horizontal line represents the constant maximum torque given by the first row of (55). The blue dashed curve represents the maximum torque based on the second row of (55). Fig. 6b verifies the analytical torque bound given the numerical result from Algorithm 2.

When $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$, one needs to find the critical motor speed $\omega_{\mathrm{s},j}$ that switches from the boundary of the current constraint to the boundary of the voltage constraint. This point is equivalent to the intersection point between the blue circle (current constraint) and the green line (voltage constraint) in Fig. 5c.

There are two ways to find $\omega_{\mathrm{s},j}$ analytically. Regarding the first method, when $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} \geq 0$, there always holds $\Phi_{\mathrm{pm},j} - i_{\mathrm{d,lim},j} L_{\mathrm{d},j} \geq 0$ due to $-I_{\max,j} \leq i_{\mathrm{d,lim},j} \leq 0$. This means that $i_{\mathrm{d},j} = i_{\mathrm{d,lim},j}$ when the torque is maximum. However, when $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$, $i_{\mathrm{d,lim},j} L_{\mathrm{d},j} + \Phi_{\mathrm{pm},j}$ could vary from positive to negative, depending on the motor speed. Note that this $i_{\mathrm{d,lim},j}$ is obtained by both the activated current and voltage constraint, according to (51). When $\Phi_{\mathrm{pm},j} + i_{\mathrm{d,lim},j} L_{\mathrm{d},j} < 0$, to avoid the flux go to negative, the current limit on the d-axis is no longer given by (51), but given by $\Phi_{\mathrm{pm},j} + i_{\mathrm{d},j} L_{\mathrm{d},j} = 0$ to compensate the flux, thus

$$i_{\mathrm{d,lim,V},j} = -\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j}. \tag{56}$$

In this case, since the voltage constraint is activated, the maximum current on the q-axis, i.e. the maximum torque, is derived from the voltage constraint, which reads

$$i_{\mathrm{q,lim,V},j} = V_{\mathrm{dq,max},j}/(\omega_{\mathrm{e},j} L_{\mathrm{q},j}). \tag{57}$$

Therefore, the critical condition that determines whether the maximum torque is constrained by the voltage constraint only or by both the voltage and current constraints is $i_{\mathrm{d,lim},j} L_{\mathrm{d},j} + \Phi_{\mathrm{pm},j} = 0$ together with $i_{\mathrm{d,lim},j} = \frac{(V_{\mathrm{dq,max},j}/\omega_{\mathrm{e},j})^2 - (L_{\mathrm{d},j} I_{\mathrm{max},j})^2 - \Phi_{\mathrm{pm},j}^2}{2\Phi_{\mathrm{pm},j} L_{\mathrm{d},j}}$, i.e.

$$\frac{(V_{\mathrm{dq,max},j}/\omega_{\mathrm{e},j})^2 - (L_{\mathrm{d},j} I_{\mathrm{max},j})^2 - \Phi_{\mathrm{pm},j}^2}{2\Phi_{\mathrm{pm},j} L_{\mathrm{d},j}} L_{\mathrm{d},j} + \Phi_{\mathrm{pm},j} = 0, \tag{58}$$

which gives the critical speed $\omega_{\mathrm{s},j}$ as

$$\omega_{\mathrm{s},j} \triangleq \frac{V_{\mathrm{dq,max},j}}{p\sqrt{(L_{\mathrm{d},j} I_{\mathrm{max},j})^2 - \Phi_{\mathrm{pm},j}^2}}. \tag{59}$$

When $\omega_j \leq \omega_{\mathrm{r},j}$, the maximum torque is a constant as in the other two cases above; when $\omega_j \in [\omega_{\mathrm{r},j}, \infty)$, both the voltage and current constraints are activated, thus the maximum torque occurs when the currents are given by (51); when $\omega_j \in [\omega_{\mathrm{s},j}, \omega_{\mathrm{max},j}]$, only the voltage constraint is activated, thus the maximum torque occurs when the currents are given by (56) and (57).

As for the second method to determine $\omega_{\mathrm{s},j}$, inspired by Fig. 5c, the critical speed occurs when the motor torque induced by both the current and voltage constraints is equal to the torque induced by only the voltage constraint. The former is equal to $1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q,lim},j}$, where $i_{\mathrm{q,lim},j}$ is given by (51). The latter is equal to $1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q,lim,V},j}$, where $i_{\mathrm{q,lim,V},j}$ is given by (56). Thus one has

$$\begin{aligned} \frac{V_{\mathrm{dq,max},j}}{\omega_{\mathrm{e},j} L_{\mathrm{q},j}} &= \sqrt{I_{\mathrm{max},j}^2 - i_{\mathrm{d,lim},j}^2}, \\ i_{\mathrm{d,lim},j} &= \frac{(V_{\mathrm{dq,max},j}/\omega_{\mathrm{e},j})^2 - (L_{\mathrm{d},j} I_{\mathrm{max},j})^2 - \Phi_{\mathrm{pm},j}^2}{2\Phi_{\mathrm{pm},j} L_{\mathrm{d},j}}, \end{aligned} \tag{60}$$

which can be simplified as a quadratic-like equation, i.e.

$$\begin{aligned} &(L_{\mathrm{d},j}^2 I_{\mathrm{max},j}^2 - \Phi_{\mathrm{pm},j}^2)^2 \omega_{\mathrm{e},j}^4 - 2V_{\mathrm{dq,max},j}^2 (L_{\mathrm{d},j}^2 I_{\mathrm{max},j}^2 - \\ &\Phi_{\mathrm{pm},j}^2)\omega_{\mathrm{e},j}^2 + V_{\mathrm{dq,max},j}^4 = 0. \end{aligned} \tag{61}$$

Since $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\mathrm{max},j} < 0$, one has $L_{\mathrm{d},j} I_{\mathrm{max},j} > \Phi_{\mathrm{pm},j} > 0$ and thus $L_{\mathrm{d},j}^2 I_{\mathrm{max},j}^2 - \Phi_{\mathrm{pm},j}^2 > 0$. Then (61) can be written as

$$(\omega_{\mathrm{e},j}^2 - \frac{V_{\mathrm{dq,max},j}^2}{L_{\mathrm{d},j}^2 I_{\mathrm{max},j}^2 - \Phi_{\mathrm{pm},j}^2})^2 = 0. \tag{62}$$

Together with $\omega_{\mathrm{e},j} = p\omega_j > 0$, solving (62) gives

$$\omega_{\mathrm{s},j} = \frac{V_{\mathrm{dq,max},j}}{p\sqrt{(L_{\mathrm{d},j} I_{\mathrm{max},j})^2 - \Phi_{\mathrm{pm},j}^2}},$$

which verifies (59) obtained by the first method.

Thus, when $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\mathrm{max},j} < 0$, the maximum motor torque $\tau_{\mathrm{max},j}(\omega_j, \boldsymbol{\beta}_j)$, as a function of motor speed and motor design parameters, is given by

$$\tau_{\mathrm{max},j} = \begin{cases} 1.5p\Phi_{\mathrm{pm},j} I_{\mathrm{max},j}, & \text{if } |\omega_j| \leq \omega_{\mathrm{r},j} \\ 1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q,lim},j}, & \text{if } |\omega_j| \in [\omega_{\mathrm{r},j}, \omega_{\mathrm{s},j}] \\ 1.5p\Phi_{\mathrm{pm},j} i_{\mathrm{q,lim,V},j}, & \text{if } |\omega_j| \in [\omega_{\mathrm{s},j}, \infty) \end{cases} \tag{63}$$

where $\omega_{\mathrm{r},j}$, $i_{\mathrm{q,lim},j}$, $\omega_{\mathrm{s},j}$, and $i_{\mathrm{q,lim,V},j}$ are given by (49), (51), (59), and (57), respectively. This piecewise-defined function is visualized in Fig. 6c. The yellow dash-dot vertical line represents $\omega_{\mathrm{r},j}$. The grey dash-dot vertical line represents $\omega_{\mathrm{s},j}$, which is the critical motor speed switching from activating both the current and voltage constraints to only the voltage constraint. The black dashed horizontal line represents the constant maximum torque given by the first row of (63). The blue dashed curve represents the maximum torque given by the second row of (63). The brown dashed curve represents the maximum torque given by the third row of (63). Fig. 6c verifies the analytical torque bound given the numerical result from Algorithm 2. Regardless of the sign of $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\mathrm{max},j}$, the minimum motor torque is given by

$$\tau_{\mathrm{min},j}(\omega_j, \boldsymbol{\beta}_j) \triangleq -\tau_{\mathrm{max},j}(\omega_j, \boldsymbol{\beta}_j). \tag{64}$$

Therefore, for all allowable motor speeds, the lower and upper bound of motor torque is illustrated in Fig. 7, based on the analytical piecewise functions (54), (55), and (63). The motor operation maps from Fig. 6 verify the correctness of the analytical functions for the torque bounds.

V. Dynamics Modeling

This section introduces the modeling methodology for a mobile manipulator's forward and inverse dynamics.

A. Inverse Dynamics

This subsection presents the recursive Newton-Euler algorithm (RNEA) for the inverse dynamics of a mobile manipulator with motor parameterization. The details are summarized in Algorithm 3. Its derivation follows the methodology from [52, Chapter 8.9.3] but is revised accordingly for mobile manipulators with motor parameterization. First, this paper adopts the following assumption on each motor's gearbox.

Assumption 4 (Massless Gearbox). The mass of the gearbox for Motor k ($k \geq 3$) is negligible.

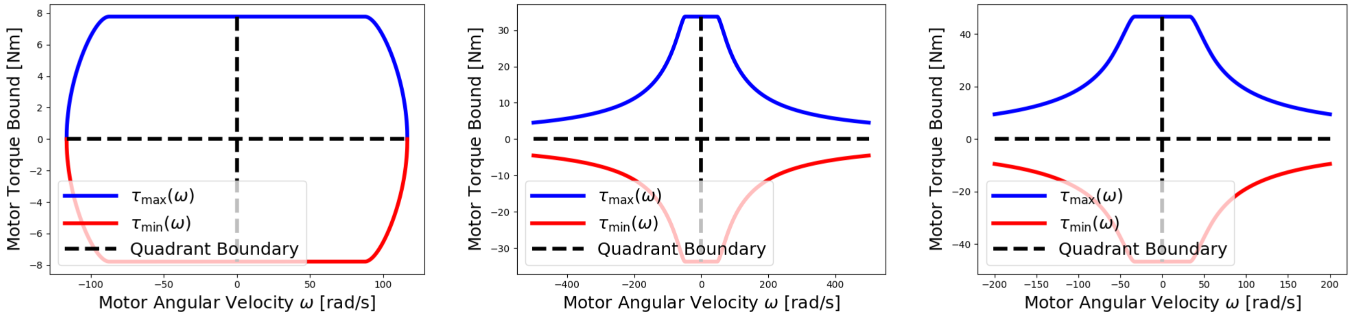
Assumption 4 is adopted due to the absence of the detailed inertia modeling of harmonic drives. Otherwise, one can lump the inertia of the primary gear, respectively the secondary gear, into its corresponding rotor, respectively its corresponding link. Remark 1 discusses the reason why no motors are considered for the base.

To recursively model the inverse dynamics, one first needs to compute all the necessary quantities recursively, including the twist, its time derivative, and the velocity-product accelerations of each rigid body. This procedure is reflected in Line 3 - Line 10 of Algorithm 3. The twist $\boldsymbol{\mathcal{V}}_{L_k}$ of Link k is the sum of the twist of Link $k-1$ expressed in frame $\{L_k\}$ and the twist due to the joint velocity $\dot{\boldsymbol{q}}_k$ or $\dot{\boldsymbol{\theta}}_k$, i.e.

$$\boldsymbol{\mathcal{V}}_{L_k} = \mathrm{Ad}_{\boldsymbol{T}_{L_k, L_{k-1}}} \boldsymbol{\mathcal{V}}_{L_{k-1}} + \boldsymbol{A}_{L_k} \dot{\boldsymbol{q}}_k. \tag{65}$$

The velocity-product acceleration $\boldsymbol{\zeta}_{L_k}$ of Link k is related to the acceleration of Link k by the equation of motion, i.e.

$$\boldsymbol{\zeta}_{L_k} = \mathring{\boldsymbol{A}}_{L_k} \dot{\boldsymbol{q}}_k + \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{L_k}} \boldsymbol{A}_{L_k} \dot{\boldsymbol{q}}_k. \tag{66}$$



(a) Motor torque bound for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} > I_{\max,j}$.

(b) Motor torque bound for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} = I_{\max,j}$.

(c) Motor torque bound for case $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j} < 0$.

Figure 7. Motor torque bound in all three cases.

The acceleration $\dot{\boldsymbol{\mathcal{V}}}_{L_k}$ of Link k, is given by the time derivative of $\boldsymbol{\mathcal{V}}_{L_k}$, i.e.

$$\dot{\boldsymbol{\mathcal{V}}}_{L_k} = \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} \dot{\boldsymbol{\mathcal{V}}}_{L_{k-1}} + \boldsymbol{\zeta}_{L_k} + \boldsymbol{A}_{L_k} \ddot{\boldsymbol{q}}_k. \tag{67}$$

Similarly, Line 5 - Line 7 hold for Rotor k, $k \geq 3$. Denote $\boldsymbol{\mathcal{V}}_{L_0} = \boldsymbol{0}_6$ as the twist of the base frame $\{L_0\}$ expressed in frame $\{L_0\}$. Gravity is treated as an acceleration of the base in the opposite direction, and thus $\dot{\boldsymbol{\mathcal{V}}}_{L_0} = -\boldsymbol{a}_{\mathrm{g}} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 9.81\end{bmatrix}^\top$, where $\boldsymbol{a}_{\mathrm{g}}$ denotes the gravity expressed in frame $\{L_0\}$.

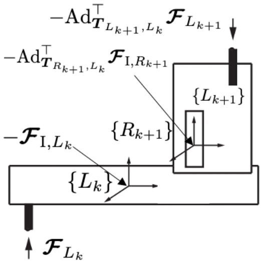

$-\mathrm{Ad}^\top_{\boldsymbol{T}_{L_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{L_{k+1}}$
$-\mathrm{Ad}^\top_{\boldsymbol{T}_{R_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{\mathrm{I},R_{k+1}}$
$\{L_{k+1}\}$
$-\boldsymbol{\mathcal{F}}_{\mathrm{I},L_k}$
$\{R_{k+1}\}$
$\{L_k\}$
$\boldsymbol{\mathcal{F}}_{L_k}$

Figure 8. Free-body diagram for Link k and Link $k+1$. Revised based on Fig. 8.12 of [52].

Secondly, one needs to identify all the wrenches applied on Link k and Link $k+1$ ($k \geq 2$) with the existence of Rotor $k+1$. This procedure is reflected in Line 11 - Line 18 of Algorithm 3. The free-body diagram for Link k and Link $k+1$ is illustrated in Fig. 8. Link k receives a wrench $\boldsymbol{\mathcal{F}}_{L_k}$ from Motor k's gearhead (expressed in frame $\{L_k\}$) and a fictitious force (or inertia force) due to the inertia of Link k (expressed in frame $\{L_k\}$). Link $k+1$ receives a wrench $\boldsymbol{\mathcal{F}}_{L_{k+1}}$ from Motor $k+1$'s gearhead (expressed in frame $\{L_{k+1}\}$) and a fictitious force due to the inertia of Rotor $k+1$ (expressed in frame $\{R_{k+1}\}$). To represent all the wrenches in the same frame $\{L_k\}$, the adjoint representation of $\boldsymbol{T}_{L_k,L_{k+1}}$ and $\boldsymbol{T}_{L_k,R_{k+1}}$ are used to convert the wrenches from frame $\{L_{k+1}\}$ to frame $\{L_k\}$, and from $\{R_{k+1}\}$ to frame $\{L_k\}$, respectively. Thus, the following equation, equivalent to Line 16, holds for each pair of Link k and Link $k+1$, $k \geq 2$.

$$\boldsymbol{\mathcal{F}}_{L_k} - \boldsymbol{\mathcal{F}}_{\mathrm{I},L_k} - \mathrm{Ad}^\top_{\boldsymbol{T}_{L_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{L_{k+1}} - \mathrm{Ad}^\top_{\boldsymbol{T}_{R_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{\mathrm{I},R_{k+1}} = \boldsymbol{0}. \tag{68}$$

Algorithm 3: Recursive Newton-Euler algorithm for the inverse dynamics of a mobile manipulator

```
1  Inputs:
     $\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1, \ddot{\boldsymbol{q}}_1; \boldsymbol{A}_{L_k}, \boldsymbol{G}_{L_k}, \mathring{\boldsymbol{A}}_{L_k}, \boldsymbol{T}_{L_k,L_{k-1}},\ k \in [\![1, n+2]\!]$;
     $\theta_k, \dot{\theta}_k, \ddot{\theta}_k, \boldsymbol{A}_{R_k}, \mathring{\boldsymbol{A}}_{R_k}, \boldsymbol{G}_{R_k}, \boldsymbol{T}_{R_k,L_{k-1}}, Z_k,\ k \in [\![3, n+2]\!]$
2  $\boldsymbol{\mathcal{V}}_{L_0} = \boldsymbol{0}_6, \dot{\boldsymbol{\mathcal{V}}}_{L_0} = \begin{bmatrix}\boldsymbol{0}_{1\times5} & 9.81\end{bmatrix}^\top, \boldsymbol{q}_2 = \dot{\boldsymbol{q}}_2 = \ddot{\boldsymbol{q}}_2 = 0$
3  for $k = 1$ to $n+2$ do
4      if $k \geq 3$ then
5          $\boldsymbol{\mathcal{V}}_{R_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{R_k,L_{k-1}}} \boldsymbol{\mathcal{V}}_{L_{k-1}} + \boldsymbol{A}_{R_k} \dot{\theta}_k$
6          $\boldsymbol{\zeta}_{R_k} \leftarrow \mathring{\boldsymbol{A}}_{R_k} \dot{\theta}_k + \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{R_k}} \boldsymbol{A}_{R_k} \dot{\theta}_k$
7          $\dot{\boldsymbol{\mathcal{V}}}_{R_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{R_k,L_{k-1}}} \dot{\boldsymbol{\mathcal{V}}}_{L_{k-1}} + \boldsymbol{\zeta}_{R_k} + \boldsymbol{A}_{R_k} \ddot{\theta}_k$
8      $\boldsymbol{\mathcal{V}}_{L_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{\mathcal{V}}_{L_{k-1}} + \boldsymbol{A}_{L_k} \dot{\boldsymbol{q}}_k$
9      $\boldsymbol{\zeta}_{L_k} \leftarrow \mathring{\boldsymbol{A}}_{L_k} \dot{\boldsymbol{q}}_k + \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{L_k}} \boldsymbol{A}_{L_k} \dot{\boldsymbol{q}}_k$
10     $\dot{\boldsymbol{\mathcal{V}}}_{L_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} \dot{\boldsymbol{\mathcal{V}}}_{L_{k-1}} + \boldsymbol{\zeta}_{L_k} + \boldsymbol{A}_{L_k} \ddot{\boldsymbol{q}}_k$
11 for $k = n+2$ to 1 do
12     $\boldsymbol{\mathcal{F}}_{\mathrm{I},L_k} \leftarrow \boldsymbol{G}_{L_k} \dot{\boldsymbol{\mathcal{V}}}_{L_k} - \mathrm{ad}^\top_{\boldsymbol{\mathcal{V}}_{L_k}} \boldsymbol{G}_{L_k} \boldsymbol{\mathcal{V}}_{L_k}$
13     if $k \geq 3$ then
14         $\boldsymbol{\mathcal{F}}_{\mathrm{I},R_k} \leftarrow \boldsymbol{G}_{R_k} \dot{\boldsymbol{\mathcal{V}}}_{R_k} - \mathrm{ad}^\top_{\boldsymbol{\mathcal{V}}_{R_k}} \boldsymbol{G}_{R_k} \boldsymbol{\mathcal{V}}_{R_k}$
15     if $k \leq n+1$ then
16         if $k \geq 2$ then $\boldsymbol{\mathcal{F}}_{L_k} \leftarrow \boldsymbol{\mathcal{F}}_{\mathrm{I},L_k} +$
             $\mathrm{Ad}^\top_{\boldsymbol{T}_{L_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{L_{k+1}} + \mathrm{Ad}^\top_{\boldsymbol{T}_{R_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{\mathrm{I},R_{k+1}}$
17         else $\boldsymbol{\mathcal{F}}_{L_k} \leftarrow \boldsymbol{\mathcal{F}}_{\mathrm{I},L_k} + Ad^\top_{\boldsymbol{T}_{L_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{L_{k+1}}$
18     else $\boldsymbol{\mathcal{F}}_{L_k} \leftarrow \boldsymbol{\mathcal{F}}_{\mathrm{I},L_k}$
19     if $k \geq 3$ then
20         $\tau_{k,\mathrm{gear}} \leftarrow \boldsymbol{A}^\top_{L_k} \boldsymbol{\mathcal{F}}_{L_k}$
21         $\tau_k \leftarrow \tau_{k,\mathrm{gear}}/Z_k + \boldsymbol{A}^\top_{R_k} \boldsymbol{\mathcal{F}}_{\mathrm{I},R_k}/Z_k$
22     else $\boldsymbol{f}_k \leftarrow \boldsymbol{A}^\top_{L_k} \boldsymbol{\mathcal{F}}_{L_k}$
23 return $\boldsymbol{f}_1, \tau_k, k = 3, \cdots, n+2$
```

Note that when $k = 1$, since there is no rotor, (68) is reduced to

$$\boldsymbol{\mathcal{F}}_{L_k} - \boldsymbol{\mathcal{F}}_{\mathrm{I},L_k} - \mathrm{Ad}^\top_{\boldsymbol{T}_{L_{k+1},L_k}} \boldsymbol{\mathcal{F}}_{L_{k+1}} = \boldsymbol{0}, \tag{69}$$

which is equivalent to Line 17. The fictitious forces of Link k and Rotor k are given by Line 12 and Line 14, respectively.

Finally, the actuator at each Joint k only has to provide force or torque in its motion subspace, which results in Line 20 and Line 22 of Algorithm 3. The torque for Motor k ($k \geq 3$) is the torque at Rotor k, which is given by the sum of the torque transmitted from the gearhead and the fictitious force of Rotor k projected on Rotor k's motion subspace, i.e. Line 21. $\boldsymbol{f}_1 \triangleq \mathrm{col}\{\tau_{1,\mathrm{z}}, f_{1,\mathrm{x}}, f_{1,\mathrm{y}}\} \in \mathbb{R}^3$ denotes the input wrench applied on Joint 1, whose vector element represents the torque applied around Joint 1's z-axis and the forces applied along Joint 1's x- and y-axis. τ_k ($k = 3, \cdots, n+2$) denotes the input motor torque applied on Joint $3, \cdots, n+2$. The input of the entire mobile manipulator is denoted by $\boldsymbol{u} \triangleq \mathrm{col}\{\boldsymbol{f}_1, \tau_3, \cdots, \tau_{n+2}\} \in \mathbb{R}^{n+3}$.

B. Forward Dynamics

This subsection presents the articulated-body algorithm (ABA) for the forward dynamics of a mobile manipulator with motor parameterization. The details are summarized in Algorithm 4. The derivation of Algorithm 4 follows the methodology from [47, Chapter 7.3], but is revised accordingly for mobile manipulators with motor parameterization.

First, one needs to define how the rigid bodies are articulated together since it determines how bias forces and spatial inertia matrices are updated recursively when using ABA to propagate forward dynamics. As illustrated in Fig. 2, for $k \geq 3$, Rotor k is articulated with Link k via its gearbox, and Link $k-1$ is articulated with Link k via its gearbox. Link 1 is articulated (or rigidly connected) with Link 2 via Joint 2, a 0-DOF joint.

Then, Line 3 - Line 12 of Algorithm 4 recursively calculates and initializes the twists, velocity-product accelerations, spatial inertia matrices, and bias forces of each link and rotor from the base to the last link. Specifically, Line 11 and Line 12 initialize the spatial inertia matrices $\boldsymbol{\mathcal{I}}^A_{L_k}$ and the bias forces $\boldsymbol{p}^A_{L_k}$ for each link, respectively. Line 7 and Line 8 initialize the same quantities for each rotor, i.e. $\boldsymbol{\mathcal{I}}^A_{R_k}$ and $\boldsymbol{p}^A_{R_k}$, respectively.

Thirdly, Line 13 - Line 34 recursively updates the spatial inertia matrices and bias forces from the last link to the first link. Specifically, Line 14 - Line 22 defines some intermediate quantities that will be used later. Note that this part skips Link 2 because there is no input applying on Link 2 and Link 2 is rigidly connected with Link 1. The reason for not combining Link 1 and Link 2 as one rigid body is for the ease of modularization. Line 24 - Line 27 updates Link 1's spatial inertia matrix $\boldsymbol{\mathcal{I}}^A_{L_1}$ and bias force $\boldsymbol{p}^A_{L_1}$ from Link 2. Line 31 - Line 32 updates Rotor k's spatial inertia matrix and bias force given Link k's effect. Line 33 - Line 34 updates Link $k-1$'s spatial inertia matrix and bias force given Link k's effect.

Finally, Line 35 - Line 40 recursively calculates the joint accelerations from the base to the last link. Note that the joint acceleration for Joint 2 is constantly zero since Link 2 is rigidly connected with Link 1.

Algorithm 4: Articulated-body algorithm for the forward dynamics of a mobile manipulator

```
1  Inputs:
     $\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1, \boldsymbol{f}_1; \boldsymbol{A}_{L_k}, \boldsymbol{G}_{L_k}, \mathring{\boldsymbol{A}}_{L_k}, \boldsymbol{T}_{L_k,L_{k-1}},\ k \in [\![1, n+2]\!];$
     $\theta_k, \dot{\theta}_k, \tau_k, \boldsymbol{A}_{R_k}, \mathring{\boldsymbol{A}}_{R_k}, \boldsymbol{G}_{R_k}, \boldsymbol{T}_{R_k,L_{k-1}}, Z_k,\ k \in [\![3, n+2]\!];$
2  $\boldsymbol{\mathcal{V}}_{L_0} = \boldsymbol{0}_6, \dot{\boldsymbol{\mathcal{V}}}_{L_0} = \begin{bmatrix}\boldsymbol{0}_{1\times5} & 9.81\end{bmatrix}^\top, \boldsymbol{q}_2 = \dot{\boldsymbol{q}}_2 = 0$ ;
3  for $k = 1$ to $n+2$ do
4      if $k \geq 3$ then
5          $\boldsymbol{\mathcal{V}}_{R_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{R_k,L_{k-1}}} \boldsymbol{\mathcal{V}}_{L_{k-1}} + \boldsymbol{A}_{R_k}\dot{\theta}_k;$
6          $\boldsymbol{\zeta}_{R_k} \leftarrow \mathring{\boldsymbol{A}}_{R_k}\dot{\theta}_k + \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{R_k}} \boldsymbol{A}_{R_k}\dot{\theta}_k;$
7          $\boldsymbol{\mathcal{I}}^A_{R_k} \leftarrow \boldsymbol{G}_{R_k};$
8          $\boldsymbol{p}^A_{R_k} \leftarrow \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{R_k}} \boldsymbol{\mathcal{I}}^A_{R_k} \boldsymbol{\mathcal{V}}_{R_k};$
9      $\boldsymbol{\mathcal{V}}_{L_k} \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{\mathcal{V}}_{L_{k-1}} + \boldsymbol{A}_{L_k}\dot{\boldsymbol{q}}_k;$
10     $\boldsymbol{\zeta}_{L_k} \leftarrow \mathring{\boldsymbol{A}}_{L_k}\dot{\boldsymbol{q}}_k + \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{L_k}} \boldsymbol{A}_{L_k}\dot{\boldsymbol{q}}_k;$
11     $\boldsymbol{\mathcal{I}}^A_{L_k} \leftarrow \boldsymbol{G}_{L_k};$
12     $\boldsymbol{p}^A_{L_k} \leftarrow \mathrm{ad}_{\boldsymbol{\mathcal{V}}_{L_k}} \boldsymbol{\mathcal{I}}^A_{L_k} \boldsymbol{\mathcal{V}}_{L_k};$
13 for $k = n+2$ to 1 do
14     $\boldsymbol{U}_{L_k} \leftarrow \boldsymbol{\mathcal{I}}^A_{L_k} \boldsymbol{A}_{L_k};$
15     if $k = 1$ then
16         $\boldsymbol{D}_k \leftarrow (\boldsymbol{A}^\top_{L_k} \boldsymbol{U}_{L_k})^{-1};$
17         $\boldsymbol{\mu}_{L_k} \leftarrow \boldsymbol{f}_k - \boldsymbol{A}^\top_{L_k} \boldsymbol{p}^A_{L_k};$
18     else if $k \geq 3$ then
19         $\boldsymbol{\mu}_{L_k} \leftarrow Z_k\tau_k - \boldsymbol{A}^\top_{L_k} \boldsymbol{p}^A_{L_k};$
20         $\boldsymbol{U}_{R_k} \leftarrow \boldsymbol{\mathcal{I}}^A_{R_k} \boldsymbol{A}_{R_k};$
21         $\boldsymbol{\mu}_{R_k} \leftarrow Z_k\tau_k - \boldsymbol{A}^\top_{R_k} \boldsymbol{p}^A_{R_k};$
22         $\boldsymbol{D}_k \leftarrow (\boldsymbol{A}^\top_{L_k} \boldsymbol{U}_{L_k} + \boldsymbol{A}^\top_{R_k} \boldsymbol{U}_{R_k})^{-1};$
23     if $k = 2$ then
24         $\boldsymbol{I}^a_{L_k} = \boldsymbol{\mathcal{I}}^A_{L_k};$
25         $\boldsymbol{p}^a_{L_k} = \boldsymbol{p}^A_{L_k} + \boldsymbol{I}^a_{L_k} \boldsymbol{\zeta}_{L_k};$
26         $\boldsymbol{\mathcal{I}}^A_{L_{k-1}} \leftarrow \boldsymbol{\mathcal{I}}^A_{L_{k-1}} + \mathrm{Ad}^\top_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{I}^a_{L_k} \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}};$
27         $\boldsymbol{p}^A_{L_{k-1}} \leftarrow \boldsymbol{p}^A_{L_{k-1}} + \mathrm{Ad}^\top_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{p}^a_{L_k};$
28     else if $k \geq 3$ then
29         $\boldsymbol{I}^a_{L_k} \leftarrow \boldsymbol{\mathcal{I}}^A_{L_k} - \boldsymbol{U}_{L_k} \boldsymbol{D}_k \boldsymbol{U}^\top_{L_k};$
30         $\boldsymbol{p}^a_{L_k} \leftarrow \boldsymbol{p}^A_{L_k} + \boldsymbol{I}^a_{L_k} \boldsymbol{\zeta}_{L_k} + \boldsymbol{U}_{L_k} \boldsymbol{D}_k \boldsymbol{\mu}_{L_k};$
31         $\boldsymbol{I}^a_{R_k} \leftarrow \boldsymbol{\mathcal{I}}^A_{R_k} - \boldsymbol{U}_{R_k} \boldsymbol{D}_k \boldsymbol{U}^\top_{R_k};$
32         $\boldsymbol{p}^a_{R_k} \leftarrow \boldsymbol{p}^A_{R_k} + \boldsymbol{I}^a_{R_k} \boldsymbol{\zeta}_{R_k} + \boldsymbol{U}_{R_k} \boldsymbol{D}_k \boldsymbol{\mu}_{R_k};$
33         $\boldsymbol{\mathcal{I}}^A_{L_{k-1}} \leftarrow$
             $\boldsymbol{\mathcal{I}}^A_{L_{k-1}} + \mathrm{Ad}^\top_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{I}^a_{L_k} \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} +$
             $\mathrm{Ad}^\top_{\boldsymbol{T}_{R_k,L_{k-1}}} \boldsymbol{I}^a_{R_k} \mathrm{Ad}_{\boldsymbol{T}_{R_k,L_{k-1}}};$
34         $\boldsymbol{p}^A_{L_{k-1}} \leftarrow$
             $\boldsymbol{p}^A_{L_{k-1}} + \mathrm{Ad}^\top_{\boldsymbol{T}_{L_k,L_{k-1}}} \boldsymbol{p}^a_{L_k} + \mathrm{Ad}^\top_{\boldsymbol{T}_{R_k,L_{k-1}}} \boldsymbol{p}^a_{R_k};$
35 for $k = 1$ to $n+2$ do
36     $\hat{\boldsymbol{a}}_k \leftarrow \mathrm{Ad}_{\boldsymbol{T}_{L_k,L_{k-1}}} \dot{\boldsymbol{\mathcal{V}}}_{L_{k-1}};$
37     if $k = 1$ then $\ddot{\boldsymbol{q}}_k \leftarrow \boldsymbol{D}_k(\boldsymbol{\mu}_{L_k} - \boldsymbol{U}^\top_{L_k}(\hat{\boldsymbol{a}}_k + \boldsymbol{\zeta}_{L_k}));$
38     else if $k = 2$ then $\ddot{\boldsymbol{q}}_k \leftarrow 0;$
39     else $\ddot{\theta}_k \leftarrow \boldsymbol{D}_k(\boldsymbol{\mu}_{L_k} - \boldsymbol{U}^\top_{L_k}(\hat{\boldsymbol{a}}_k + \boldsymbol{\zeta}_{L_k}) -$
         $\boldsymbol{U}^\top_{R_k}(\hat{\boldsymbol{a}}_k + \boldsymbol{\zeta}_{R_k}) - \boldsymbol{A}^\top_{R_k} \boldsymbol{p}^A_{R_k});$
40     $\dot{\boldsymbol{\mathcal{V}}}_{L_k} \leftarrow \hat{\boldsymbol{a}}_k + \boldsymbol{\zeta}_{L_k} + \boldsymbol{A}_{L_k}\ddot{\boldsymbol{q}}_k;$
41 Return: $\ddot{\boldsymbol{q}}_1, \ddot{\theta}_k, k = 3, \cdots, n+2;$
```

C. Modeling Validation

This subsection presents the modeling validation by comparing the forward dynamics and the inverse dynamics obtained from Algorithm 4 and Algorithm 3, respectively. Since both algorithms are derived by different methodologies, it is reasonable to infer that the modeling is correct if there is no difference or only numerical error between the numerical results from the two algorithms. The definition of the so-called numerical results is given below.

First, the continuous-time forward dynamics of the mobile manipulator can be obtained symbolically by Algorithm 4. One could obtain a discrete-time trajectory of states and inputs by solving a motion planning problem, which will be introduced in Section VI. Then at each time instance, one could compute the joint accelerations $\ddot{\boldsymbol{q}}_1, \ddot{\boldsymbol{\theta}}_k, k \geq 3$ given the current state, i.e. the joint positions $\boldsymbol{q}_1, \boldsymbol{\theta}_k$ and the velocities $\dot{\boldsymbol{q}}_1, \dot{\boldsymbol{\theta}}_k$, and the current input $\boldsymbol{f}_1, \tau_k, k \geq 3$ with the continuous-time forward dynamics. Finally, at each time instance, with Algorithm 3, given $\boldsymbol{q}_1, \boldsymbol{\theta}_k$, $\dot{\boldsymbol{q}}_1, \dot{\boldsymbol{\theta}}_k$, and $\ddot{\boldsymbol{q}}_1, \ddot{\boldsymbol{\theta}}_k$, one could compute the desired input and compare the inputs from the forward dynamics and the inverse dynamics. Since this process only evaluates the state and input by continuous-time forward dynamics at each time instance, how the continuous-time forward dynamics are discretized will not affect the comparison accuracy. In other words, the comparison does not involve forward propagating the continuous-time dynamics and thus there is no accumulated error regarding how to integrate the continuous-time dynamics. The only reason why this process involves motion planning is to generate a reasonable trajectory of states and inputs for point-wise evaluation.

The statistics about the error between the forward and inverse dynamics are summarized in Table II, where the percentage error is defined by the error percentage regarding the maximum absolute value of this variable's trajectory. Based on Table II, the error might only be a relatively large round-off error. According to [47, Chapter 10], large round-off errors can arise during dynamics calculations for the following reasons: (1) using a far-away coordinate system; (2) large velocities; (3) large inertia ratios. For reason (1), a coordinate system is far away from a rigid body if the distance between the origin and the center of mass is many times the body's radius of gyration. Ideally, this distance should be no more than about one or two radii of gyration [47, Chapter 10]. Thus this paper conjectures that using a global coordinate system for mobile manipulators can cause large round-off errors. For reason (3), one generally requires the base to have a larger inertia than the manipulator for stability, and thus a large inertia ratio is also a possible reason for large round-off errors. How to compute mobile manipulator dynamics in a more numerically stable coordinate system requires further investigation.

Table II
Error between Forward and Inverse Dynamics

Name	mean ± std	quartiles[†]	max	mean
$\tau_{1,\mathrm{z}}$	0.154 ± 0.113	0.051, 0.149, 0.229	0.417	0.41 %
$f_{1,\mathrm{x}}$	0.188 ± 0.114	0.102, 0.193, 0.235	0.492	0.23 %
$f_{1,\mathrm{y}}$	0.226 ± 0.173	0.104, 0.196, 0.281	0.706	0.24 %
τ_3	0.003 ± 0.002	0.001, 0.003, 0.004	0.008	1.26 %
τ_4	0.002 ± 0.002	0.001, 0.001, 0.003	0.013	0.07 %
τ_5	0.002 ± 0.003	0.001, 0.001, 0.003	0.013	0.21 %
τ_6	0.003 ± 0.003	0.001, 0.002, 0.005	0.013	1.97 %
τ_7	0.002 ± 0.002	0.000, 0.001, 0.002	0.011	0.37 %
τ_8	0.003 ± 0.002	0.001, 0.002, 0.004	0.008	1.57 %

[†] 25th, 50th, 75th percentile
[‡] If no indication, unit for torque is Nm; unit for force is N

VI. Integrated Locomotion and Manipulation Planning

This section introduces an optimization-based integrated locomotion and manipulation planning given the differentiable dynamics of a mobile manipulator with motor parameterization. Consider a mobile manipulator with $n = 6$. The motor design variables for manipulator's j-th motor, $j \in [\![3, n+2]\!]$, are denoted by l_j, $r_{\mathrm{ro},j}$, $r_{\mathrm{so},j}$, $h_{\mathrm{m},j}$, $h_{\mathrm{sy},j}$, $w_{\mathrm{tooth},j}$, $b_{0,j}$. And further denote

$$\begin{aligned} \boldsymbol{\beta}_j &\triangleq \mathrm{col}\{l_j, r_{\mathrm{ro},j}, r_{\mathrm{so},j}, h_{\mathrm{m},j}, h_{\mathrm{sy},j}, w_{\mathrm{tooth},j}, b_{0,j}\}, \\ \boldsymbol{\beta} &\triangleq \mathrm{col}\{\boldsymbol{\beta}_3, \cdots, \boldsymbol{\beta}_{n+2}\} \in \mathbb{R}^{7n}, \end{aligned}$$

where $\boldsymbol{\beta}$ indicates all the motor design parameters of the mobile manipulator. Denote $\boldsymbol{\theta} \triangleq \mathrm{col}\{\theta_3, \cdots, \theta_{n+2}\} \in \mathbb{R}^n$ as the joint angular positions of the arm and denote $\dot{\boldsymbol{\theta}} \in \mathbb{R}^n$ as the joint velocity similarly. The complete forward dynamics for the mobile manipulator, detailed in Algorithm 4, are abstracted as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_{\mathrm{c}}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\beta}), \tag{70}$$

where $\boldsymbol{x} \triangleq \mathrm{col}\{\boldsymbol{q}_1, \boldsymbol{\theta}, \dot{\boldsymbol{q}}_1, \dot{\boldsymbol{\theta}}\} \in \mathbb{R}^{2(n+3)}$ and $\boldsymbol{u} \triangleq \mathrm{col}\{\boldsymbol{f}_1, \tau_3, \cdots, \tau_{n+2}\} \in \mathbb{R}^{n+3}$.

Given a task, a trajectory planning optimization can return its optimal trajectories of states and controls that align with the task requirements:

$$\begin{aligned} \min_{\boldsymbol{x}(t), \boldsymbol{u}(t), t_{\mathrm{f}}} \quad & J(\boldsymbol{x}(t), \boldsymbol{u}(t), t_{\mathrm{f}}) && \text{(71a)} \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}_{\mathrm{c}}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\beta}), && \text{(71b)} \\ & \forall t \in [0, t_{\mathrm{f}}] \text{ with given } \boldsymbol{x}(0), && \text{(71c)} \\ & \text{other constraints on } \boldsymbol{x}(t), \boldsymbol{u}(t), && \text{(71d)} \end{aligned}$$

where $t_{\mathrm{f}} > 0$ denotes the final time for trajectory planning, which could either be a decision variable or a prescribed parameter. Note that (71) adopts the most general form, which represents time-optimal trajectory planning, trajectory tracking, or energy-optimal trajectory planning. For the latter two cases, t_{f} is a fixed prescribed parameter. Given a particular value of $\boldsymbol{\beta}$, the optimal states $\boldsymbol{x}^*(t)$, controls $\boldsymbol{u}^*(t)$, and final time t_{f}^* (if applicable) optimize the cost function J.

Remark 5. One can combine the motor dynamics (44) with the manipulator dynamics (70) together as an entire system with its state $\boldsymbol{q}_1, \boldsymbol{\theta}, \dot{\boldsymbol{q}}_1, \dot{\boldsymbol{\theta}}, i_{\mathrm{d},j}, i_{\mathrm{q},j}$ and its input $\boldsymbol{f}_1, \tau_3, \cdots, \tau_{n+2}, u_{\mathrm{d},j}, u_{\mathrm{q},j}$. However, the motor dynamics typically operate at frequencies in the thousands of Hertz, while the arm dynamics often run at frequencies in the hundreds of Hertz. The frequency discrepancy necessitates discretizing the entire system with a time step much smaller than 1 millisecond, which can lead to an unnecessary computational burden. In practice, it is reasonable to assume that between every two consecutive time steps of the manipulator dynamics, the motor dynamics have enough time to adjust their speed and torque to match the desired values, considering control constraints.

A. Discretization by Direct Collocation

Discretization is typically used to efficiently solve the continuous-time problem (71) with numerical solvers. Among different discretization methods, the direct collocation method minimizes the error between state derivatives from continuous-time dynamics and state derivatives from approximated polynomial differentiation on every control interval. For systems with fast and nonlinear dynamics with running frequencies higher than hundreds of Hz, direct collocation generally outperforms the Euler method (ODE1) and the Runge–Kutta methods such as ODE4. To discretize the optimal control problem (71) with direct collocation, the continuous-time dynamics $\dot{\boldsymbol{x}} = \boldsymbol{f}_{\mathrm{c}}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\beta})$ is replaced by a set of discrete-time collocation equations.

First, discretize the entire time horizon $[0, t_{\mathrm{f}}]$ into $N \in \mathbb{Z}_+$ uniform intervals, then the discretized time grids are given by $t_k = k \cdot t_{\mathrm{f}}/N,\ k = 0, \cdots, N$. Denote the uniform time step as $\Delta \triangleq t_{\mathrm{f}}/N$ and the state at time t_k as $\boldsymbol{x}_k \triangleq \boldsymbol{x}(t_k)$. Let the control over the time interval $[t_k, t_{k+1})$ as constant, i.e. $\boldsymbol{u}(t_k) = \boldsymbol{u}_k,\ \forall t \in [t_k, t_{k+1}),\ k = 0, \cdots, N$.

Secondly, one needs to compute the coefficients of the polynomials on each interval $[t_k, t_{k+1})$ to ensure that the continuous-time dynamics (i.e. some ODEs) are exactly satisfied at the collocation points $t_{k,i} = t_k + c_i\Delta,\ i = 1, \cdots, n_{\mathrm{p}}$, where $n_{\mathrm{p}} \in \mathbb{Z}_+$ denotes the polynomial degree. The collocation points $c_1 < \cdots < c_{n_{\mathrm{p}}}$ are defined in an open interval $(0, 1)$. For completeness, the left endpoint $c_0 = 0$ is also included and thus $0 = c_0 < c_1 < \cdots < c_{n_{\mathrm{p}}} < 1$. The choice of the collocation points determines the accuracy and numerical stability of the discretization and the optimal control solution. This paper chooses the Gauss collocation points, obtained as the roots of a shifted Gauss-Jacobi polynomial, thanks to the good numerical stability and the attribute of living in the open interval $(0, 1)$ [53].

Thirdly, to use the polynomials to approximate the continuous-time dynamics, one needs to define the basis of polynomials. This paper utilizes the Lagrange interpolation polynomial basis. Define the basis for polynomials as $\ell_i : \mathbb{R} \mapsto \mathbb{R},\ i = 0, \cdots, n_{\mathrm{p}}$, and

$$\ell_i(t) = \textstyle\prod_{j=0, i\neq j}^{n_{\mathrm{p}}} \frac{t - c_j}{c_i - c_j},\ \ i \in [\![0, n_{\mathrm{p}}]\!]. \tag{72}$$

Then the approximated state trajectory $\boldsymbol{q}_k(t)$ on each time interval $[t_k, t_{k+1})$ is given by a linear combination of the basis functions:

$$\boldsymbol{q}_k(t) = \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \ell_j(\frac{t - t_k}{\Delta})\boldsymbol{x}_{k,j},$$

where $\boldsymbol{x}_{k,j} \in \mathbb{R}^{2(n+3)},\ \forall k \in [\![0, N]\!], j = [\![1, n_{\mathrm{p}}]\!]$ is the new decision variable for the discretized optimal control problem. By differentiation, the approximated time derivative of the state at each collocation point over the interval $[t_k, t_{k+1})$ is given by

$$\dot{\boldsymbol{q}}_k(t_{k,i}) = \frac{1}{\Delta}\sum_{j=0}^{n_{\mathrm{p}}} \dot{\ell}_j(c_i)\boldsymbol{x}_{k,j} := \frac{1}{\Delta}\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{C}_{j,i}\boldsymbol{x}_{k,j}, i \in [\![0, n_{\mathrm{p}}]\!], \tag{73}$$

where $\dot{\ell}_j(c_i) = \frac{d\ell_j(t)}{dt}|_{t=c_i}$; $\boldsymbol{C}_{j,i} \in \mathbb{R}$ is a constant given a particular n_{p}, polynomial basis and collocation points; denote $\boldsymbol{C} \in \mathbb{R}^{(n_{\mathrm{p}}+1)\times(n_{\mathrm{p}}+1)}$ as the differentiation matrix, where its element on the j-th row and i-th column is $\boldsymbol{C}_{j,i}$. Note that even though $\boldsymbol{q}_k(t)$ is dependent on $\Delta := t_{\mathrm{f}}/N$, according to (73), the constant matrix $\boldsymbol{C}$ is independent on t_{f} and N. The state at the end of each time interval $[t_k, t_{k+1})$, i.e. at time t_{k+1}, is given by

$$\boldsymbol{x}_{k+1} = \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \ell_j(1)\boldsymbol{x}_{k,j} := \sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{D}_j\boldsymbol{x}_{k,j}, \tag{74}$$

where $\boldsymbol{D}_j \in \mathbb{R}$ is a constant; denote $\boldsymbol{D} \in \mathbb{R}^{n_{\mathrm{p}}+1}$ as the continuity vector, where its element on the j-th row is $\boldsymbol{D}_j$.

In many applications, the objective function (71a) typically includes an integration of a stage cost and a terminal cost, i.e.

$$J = \boldsymbol{h}(\boldsymbol{x}(t_{\mathrm{f}}), t_{\mathrm{f}}) + \int_0^{t_{\mathrm{f}}} \boldsymbol{c}(\boldsymbol{x}(t), \boldsymbol{u}(t))dt,$$

where $\boldsymbol{h} : \mathbb{R}^{2(n+3)} \times \mathbb{R}_+ \mapsto \mathbb{R}$ is the terminal cost function.; $\boldsymbol{c} : \mathbb{R}^{2(n+3)} \times \mathbb{R}^{n+3} \mapsto \mathbb{R}$ is the stage cost function. Then, using the approximation $\boldsymbol{q}_k(t)$, one can integrate the stage cost over each time interval $[t_k, t_{k+1})$ given constant control $\boldsymbol{u}_k$ and obtain the following:

$$\begin{aligned}&\int_{t_k}^{t_{k+1}} \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \ell_j(\frac{t-t_k}{\Delta})\boldsymbol{c}(\boldsymbol{x}_{k,j}, \boldsymbol{u}_k)dt = \\ &\Delta\sum_{j=0}^{n_{\mathrm{p}}}\int_0^1 \ell_j(t)dt\ \boldsymbol{c}(\boldsymbol{x}_{k,j}, \boldsymbol{u}_k) := \Delta\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{B}_j\ \boldsymbol{c}(\boldsymbol{x}_{k,j}, \boldsymbol{u}_k),\end{aligned} \tag{75}$$

where $\boldsymbol{B}_j \in \mathbb{R}$ is a constant; denote $\boldsymbol{B} \in \mathbb{R}^{n_{\mathrm{p}}+1}$ as the quadrature vector, where its element on the j-th row is $\boldsymbol{B}_j$.

Therefore, with the direction collocation, the continuous-time optimal control problem can be rewritten in discrete-time:

$$\min_{(*)} \quad \boldsymbol{h}(\boldsymbol{x}_{N,0}, t_{\mathrm{f}}) + \frac{t_{\mathrm{f}}}{N}\textstyle\sum_{k=0}^{N-1}\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{B}_j\ \boldsymbol{c}(\boldsymbol{x}_{k,j}, \boldsymbol{u}_k) \tag{76a}$$

$$\text{s.t.} \quad \frac{t_{\mathrm{f}}}{N}\boldsymbol{f}_{\mathrm{c}}(\boldsymbol{x}_{k,i}, \boldsymbol{u}_k, \boldsymbol{\beta}) - \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{C}_{j,i}\boldsymbol{x}_{k,j} = \boldsymbol{0}, \tag{76b}$$

$$\boldsymbol{x}_{k+1,0} - \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{D}_j\boldsymbol{x}_{k,j} = \boldsymbol{0}, \tag{76c}$$

$$\forall k \in [\![0, N-1]\!], i \in [\![1, n_{\mathrm{p}}]\!],\ \text{given } \boldsymbol{x}_0, \tag{76d}$$

$$\text{other constraints on decision variables,} \tag{76e}$$

where $(*)$ denotes all the decision variables, i.e. $t_{\mathrm{f}} \in \mathbb{R}_+$, $\boldsymbol{x}_{k,i} \in \mathbb{R}^{2(n+3)}, \forall k \in [\![0, N-1]\!], i \in [\![1, n_{\mathrm{p}}]\!]$, $\boldsymbol{u}_k \in \mathbb{R}^{n+3}, \forall k \in [\![0, N-1]\!]$, and $\boldsymbol{x}_{N,0}$.

VII. Numerical Experiments

This section presents some numerical experiments on a mobile manipulator, including an example of integrated locomotion and manipulation planning and a comparison with a benchmark planning method. This section aims to emphasize that the dynamic modeling of a mobile manipulator enables the integrated planning of both the base and the manipulator. This integrated approach leads to faster movement compared to a benchmark planning method, where locomotion and manipulation planning are performed sequentially.

A. Proposed: Integrated Planning

This subsection introduces a concrete example of the integrated locomotion and manipulation planning problem (76) with direct collocation:

$$\begin{aligned}
\min_{(*)} \quad & \frac{t_\mathrm{f}}{N}\textstyle\sum_{k=0}^{N-1}\sum_{j=0}^{n_\mathrm{p}} \boldsymbol{B}_j ||\boldsymbol{u}_k||_2^2 && \text{(77a)}\\
\text{s.t.} \quad & \frac{t_\mathrm{f}}{N}\boldsymbol{f}_\mathrm{c}(\boldsymbol{x}_{k,i},\boldsymbol{u}_k,\boldsymbol{\beta}) - \textstyle\sum_{j=0}^{n_\mathrm{p}} \boldsymbol{C}_{j,i}\boldsymbol{x}_{k,j} = \boldsymbol{0}, && \text{(77b)}\\
& \boldsymbol{x}_{k+1,0} - \textstyle\sum_{j=0}^{n_\mathrm{p}} \boldsymbol{D}_j \boldsymbol{x}_{k,j} = \boldsymbol{0}, && \text{(77c)}\\
& \underline{\boldsymbol{x}} \le \boldsymbol{x}_{k,i} \le \overline{\boldsymbol{x}}, && \text{(77d)}\\
& \forall k \in [\![0, N-1]\!], i \in [\![1, n_\mathrm{p}]\!], \text{ given } \boldsymbol{x}_0, && \text{(77e)}\\
& \tau_{k,r} - \tau_{\max,r}(Z_r\theta_{k,i,r},\boldsymbol{\beta}_r) \le 0, && \text{(77f)}\\
& \tau_{\min,r}(Z_r\theta_{k,i,r},\boldsymbol{\beta}_r) - \tau_{k,r} \le 0, && \text{(77g)}\\
& \underline{\boldsymbol{f}_1} \le \boldsymbol{f}_{k,1} \le \overline{\boldsymbol{f}_1}, && \text{(77h)}\\
& \forall k \in [\![0, N-1]\!], i \in [\![1, n_\mathrm{p}]\!], r \in [\![3, n+2]\!], && \text{(77i)}\\
& ||\boldsymbol{p}_{J_\mathrm{e}}(\boldsymbol{x}_{N,0}) - \boldsymbol{p}_{J_\mathrm{e},\mathrm{des}}||_2 = 0, && \text{(77j)}
\end{aligned}$$

where $(*)$ denotes all the decision variables, i.e. $\boldsymbol{x}_{k,i} \in \mathbb{R}^{2(n+3)}, \forall k \in [\![0, N-1]\!], i \in [\![1, n_\mathrm{p}]\!]$, $\boldsymbol{u}_k \in \mathbb{R}^{n+3}, \forall k \in [\![0, N-1]\!]$, and $\boldsymbol{x}_{N,0}$; $n_\mathrm{p} = 1$. $\underline{\boldsymbol{x}}$ and $\overline{\boldsymbol{x}}$ represent the lower and upper bound of states, respectively. $\tau_{k,r}$ represents Motor r's torque from the input $\boldsymbol{u}_k$. $\theta_{k,i,r}$ represents Joint r's position from the state $\boldsymbol{x}_{k,i}$; $Z_r = 50$ represents the gear ratio of Joint r. $\tau_{\min,r}$ and $\tau_{\max,r}$ are given by (54), (55), or (63), depending on the sign of $\Phi_{\mathrm{pm},r}/L_{\mathrm{d},r} - I_{\max,r}$. $\underline{\boldsymbol{f}_1}$ and $\overline{\boldsymbol{f}_1}$ represent the lower and upper bound of the base's control input, respectively. $\boldsymbol{f}_{k,1}$ represents the base's control from the input $\boldsymbol{u}_k$. Eq. (77j) constrains the terminal state $\boldsymbol{x}_{N,0}$, where $\boldsymbol{p}_{J_\mathrm{e}}(\cdot)$ is a mapping from the state (joint positions) to the end effector position in Euclidean space and $\boldsymbol{p}_{J_\mathrm{e},\mathrm{des}} \in \mathbb{R}^3$ represents the desired end effector position. The details on $\boldsymbol{p}_{J_\mathrm{e}}(\cdot)$ are given by Section III-D.

The parameters related to the mobile manipulator are given as follows. $n = 6$. The base mass is 90 kg, the manipulator mass (including the motors) is 47.03 kg, and the end effector is carrying a 5 kg solid iron ball. $\overline{\boldsymbol{f}_1} = -\underline{\boldsymbol{f}_1} = \begin{bmatrix}150 \text{ Nm} & 150 \text{ N} & 150 \text{ N}\end{bmatrix}^\top$, which can generate an acceleration of approximately $1 \text{ m/s}^2 \approx 0.1$ g in the x- and y-axis. Given some empirical motor design parameters, the constant maximum torque is 13 Nm and starts decreasing at 168 rad/s until reaching the maximum speed at 301 rad/s. The base's initial yaw angle is 0 rad; the base's initial positions in the x- and y-axis are 0 m; the initial and the desired terminal velocities for every joint of the mobile manipulator are all 0.

During the simulation, after an optimal trajectory is obtained by solving (77), a PID (Proportional–Integral–Derivative) controller is adopted to track the desired torques from the optimal trajectory, i.e. for $i \ge 3$,

$$\begin{aligned}
\tau_i(t) &:= \tau_{i,\mathrm{des}}(t) + K_{\mathrm{P},i}\Big(\theta_{i,\mathrm{des}}(t) - \theta_i(t)\Big) + K_{\mathrm{I},i} \\
&\int_0^t \Big(\theta_{i,\mathrm{des}}(t) - \theta_i(t)\Big)dt + K_{\mathrm{D},i}\Big(\dot{\theta}_{i,\mathrm{des}}(t) - \dot{\theta}_i(t)\Big),
\end{aligned} \tag{78}$$

where $K_{\mathrm{P},i}, K_{\mathrm{I},i}, K_{\mathrm{D},i} > 0$ are the proportional, integral, and derivative gains, respectively; $\theta_{i,\mathrm{des}}(t)$ and $\dot{\theta}_{i,\mathrm{des}}(t)$ are the desired joint position and velocity from the optimal trajectory. Similarly, for the base, a similar PID controller is used, i.e.

$$\begin{aligned}
\boldsymbol{f}_1(t) &:= \boldsymbol{f}_{1,\mathrm{des}}(t) + \boldsymbol{K}_{\mathrm{P},1}\Big(\boldsymbol{q}_{1,\mathrm{des}}(t) - \boldsymbol{q}_1(t)\Big) + \boldsymbol{K}_{\mathrm{I},1} \\
&\int_0^t \Big(\boldsymbol{q}_{1,\mathrm{des}}(t) - \boldsymbol{q}_1(t)\Big)dt + \boldsymbol{K}_{\mathrm{D},1}\Big(\dot{\boldsymbol{q}}_{1,\mathrm{des}}(t) - \dot{\boldsymbol{q}}_1(t)\Big),
\end{aligned} \tag{79}$$

where $\boldsymbol{K}_{\mathrm{P},1}, \boldsymbol{K}_{\mathrm{I},1}, \boldsymbol{K}_{\mathrm{D},1} \in \mathbb{R}^3$ are the PID gain diagonal matrices with positive diagonal elements. The closed-loop system result from the integrated motion planning approach is illustrated in Fig. 9, where the PID feedback is given in (78) and (79). Note that for the integrated motion planning, $\tau_{i,\mathrm{des}}(t)$ and $\boldsymbol{f}_{1,\mathrm{des}}(t)$ in (78) and (79) are the feed-forward terms and given by the optimal solution of (77). The integrated motion planning problems with different parameters are denoted as: (1)

1) P1: the proposed integrated planning (77) with final time $t_\mathrm{f} = 5.0$ s and number of intervals $N = 500$;
2) P2: the same as P1, but $t_\mathrm{f} = 3.5$ s, $N = 350$;
3) P3: the same as P1, but $t_\mathrm{f} = 2.0$ s, $N = 220$.

Note that $t_\mathrm{f} = 2.0$ s is the minimal dynamically feasible time for this particular example given the full dynamics of the mobile manipulator. This is obtained by solving a time optimal integrated motion planning problem, which is similar to (77), but adding t_f as one decision variable and revising the objective function (77a) as just t_f.

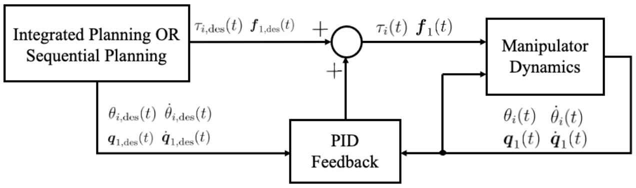

Integrated Planning OR Sequential Planning
$\tau_{i,\mathrm{des}}(t)$ $\boldsymbol{f}_{1,\mathrm{des}}(t)$
+
+
$\tau_i(t)$ $\boldsymbol{f}_1(t)$
Manipulator Dynamics
$\theta_{i,\mathrm{des}}(t)$ $\dot{\theta}_{i,\mathrm{des}}(t)$
$\boldsymbol{q}_{1,\mathrm{des}}(t)$ $\dot{\boldsymbol{q}}_{1,\mathrm{des}}(t)$
PID Feedback
$\theta_i(t)$ $\dot{\theta}_i(t)$
$\boldsymbol{q}_1(t)$ $\dot{\boldsymbol{q}}_1(t)$

Figure 9. The closed-loop system with PID feedback for both the integrated motion planning approach and the sequential planning approach.

B. Comparison of Integrated Motion Planning with Analytical and Numerical Derivatives

To demonstrate the importance of using differentiable dynamical modeling, this subsection provides a comparison between solving the integrated motion planning optimization using the analytical gradients from the differentiable dynamic model and using the numerical gradients from the finite difference method. Specifically, this paper uses CasADi [54] to construct the analytical dynamic model with Algorithm 4, and employs the interface between CasADi and the nonlinear optimization solver IPOPT [55] to automatically provide the analytical gradients of the objective function and the equality/inequality constraints (including the analytical gradients of the forward dynamics), as well as the analytical Hessian matrix of the Lagrangian, to the IPOPT solver. As for the numerical gradient approach, the numerical derivatives are provided by the central finite difference method in CasADi, and the Hessian matrix approximation is given by the limited-memory quasi-Newton approximation provided by IPOPT. The maximum iteration for both approaches is 1500 and the optimization initial guess for both approaches is a zero vector.

The desired end effector position is randomly generated given the constant distance of 2.25 m and the constant height of 0.5 m to the initial base position. Each approach is performed for the same random end effector desired position for 20 different trials and the result is summarized in Table III, where Iteration Count indicates the total IPOPT iterations; n_call_nlp_f indicates the number of function calls for the objective function; similarly, n_call_nlp_g for the constraints, n_call_nlp_grad_f for the objective function gradient, n_call_nlp_jac_g for the constraints' gradients, n_call_nlp_hess_l for the Hessian matrix of the Lagrangian; $||\boldsymbol{u}||_2$ indicates the total summation of control magnitude over time; Error indicates the error between the final actual end effector position and the desired one on in millimeters given the closed-loop PID tracking controllers; Success indicates the success rate of solving optimization successfully.

Table III demonstrates the significance of using a differentiable dynamics model for solving motion planning optimization problems, regarding computational efficiency. The differentiable dynamics model could provide the exact Hessian matrix and exact gradients for the optimization, which results in less total amount of function calls and thereby reduces the total computing time. Moreover, for interior-point methods, providing the exact Hessian matrix makes the algorithm converge faster than using the Quasi-Newton method with the limited memory BFGS (Broyden-Fletcher-Goldfarb-Shanno) update [55]. As for the optimality, according to the same result from $||\boldsymbol{u}||_2$ and Error, the optimization with analytical and numerical gradients in this comparison yields the same optimality, i.e. the same control effort.

Table III
Motion Planning with Analytical and Numerical Gradients

Item	P2	P2[†]
IPOPT Time [s]	**28.13 ± 1.07**	267.21±24.42
Iteration Count	**57.6 ± 2.4**	679.6±50.6
n_call_nlp_f	**758 ± 375**	16795±8315
n_call_nlp_g	**758 ± 375**	16795±8315
n_call_nlp_grad_f	**630 ± 340**	7267±3863
n_call_nlp_jac_g	**630 ± 340**	7267±3863
n_call_nlp_hess_l	610±329	**0**
$\|\|\boldsymbol{u}\|\|_2$	14716±453	14716±453
Error [mm]	1.09±0.02	1.09±0.02
Success [%]	**100**	**100**

[†] with numerical gradients

C. Benchmark: Sequential Planning

This subsection presents a benchmark approach, i.e., sequentially doing locomotion and manipulation planning. The sequential planning approach has two phases. In the first phase, the base moves to a location within a circle centering at the projection of $\boldsymbol{p}_{J_\mathrm{e},\mathrm{des}}$ on the ground (XOY plane). The base's desired position, velocity, and acceleration trajectory at the x- and y-axis are parameterized by a prescribed maximum acceleration magnitude $a_\mathrm{m} > 0$ and a desired final time t_f1. The dynamic coupling between the base and the manipulator is essentially a wrench (or a spatial force) applied on both parts. To reduce the effect of this wrench, as discussed inSection I-A1, the motion must minimize the jerk caused by sudden movements. The trajectory parameterization is given by (80) - (82), and its acceleration parameterization is illustrated in Fig. 10. The acceleration increases from 0 initially and decreases to 0 finally for smooth movement.

Without loss of generality (WLOG), the desired base movement at each axis is given by

$$a(t) = \begin{cases} \frac{4a_\mathrm{m}}{t_\mathrm{f1}}t, & \text{if } t \in [0,\ \frac{t_\mathrm{f1}}{4}] \\ \frac{-4a_\mathrm{m}}{t_\mathrm{f1}}t + 2a_\mathrm{m}, & \text{if } t \in [\frac{t_\mathrm{f1}}{4},\ \frac{3t_\mathrm{f1}}{4}] \\ \frac{4a_\mathrm{m}}{t_\mathrm{f1}}t - 4a_\mathrm{m}, & \text{if } t \in [\frac{3t_\mathrm{f1}}{4},\ t_\mathrm{f1}] \end{cases} \tag{80}$$

$$v(t) = \begin{cases} \frac{2a_\mathrm{m}}{t_\mathrm{f1}}t^2, & \text{if } t \in [0,\ \frac{t_\mathrm{f1}}{4}] \\ \frac{-2a_\mathrm{m}}{t_\mathrm{f1}}(t - \frac{1}{2}t_\mathrm{f1})^2 + \frac{a_\mathrm{m}t_\mathrm{f1}}{4}, & \text{if } t \in [\frac{t_\mathrm{f1}}{4},\ \frac{3t_\mathrm{f1}}{4}] \\ \frac{2a_\mathrm{m}}{t_\mathrm{f1}}(t - t_\mathrm{f1})^2, & \text{if } t \in [\frac{3t_\mathrm{f1}}{4},\ t_\mathrm{f1}] \end{cases} \tag{81}$$

$$p(t) = \begin{cases} \frac{2a_\mathrm{m}}{3t_\mathrm{f1}}t^3, & \text{if } t \in [0,\ \frac{t_\mathrm{f1}}{4}] \\ \frac{-2a_\mathrm{m}}{3t_\mathrm{f1}}(t - \frac{1}{2}t_\mathrm{f1})^3 + \frac{4a_\mathrm{m}t_\mathrm{f1}t - a_\mathrm{m}t_\mathrm{f1}^2}{16}, & \text{if } t \in [\frac{t_\mathrm{f1}}{4},\ \frac{3t_\mathrm{f1}}{4}] \\ \frac{2a_\mathrm{m}}{3t_\mathrm{f1}}(t - t_\mathrm{f1})^3 + \frac{a_\mathrm{m}t_\mathrm{f1}^2}{8}, & \text{if } t \in [\frac{3t_\mathrm{f1}}{4},\ t_\mathrm{f1}] \end{cases} \tag{82}$$

Note that $a(0) = a(t_\mathrm{f1}) = 0$; $v(0) = v(t_\mathrm{f1}) = 0$; WLOG, $v(t) > 0\ \forall t \in (0, t_\mathrm{f1})$, $p(t) = 0$ and $p(t)$ is monotonically increasing in $[0, t_\mathrm{f1}]$. Given a particular final position p_f, the parameterized trajectory in (82) yields $t_\mathrm{f1} = \sqrt{8p_\mathrm{f}/a_\mathrm{m}}$. The desired yaw angle in the first phase is always the initial yaw angle, so as the desired yaw velocity, to reduce the influence of the base movement on the manipulator.

Since the dynamic coupling between two bodies is essentially an unmodeled wrench, this wrench can be viewed as a disturbance and one can design a feedback controller to reject this disturbance. Hence, a PID controller, similar to (79), is used to track the desired trajectories for the base, where the feed-forward force on the x- and y-axis, i.e. the second and third element of $\boldsymbol{f}_{1,\mathrm{des}}(t)$, are given by the total mass of the robot multiplied by the desired acceleration defined in (80); the feed-forward torque around the z-axis, i.e. the first element of $\boldsymbol{f}_{1,\mathrm{des}}(t)$, is zero. To stabilize the manipulator while the base is moving, the desired motor torques are computed by the inverse dynamics of a fixed manipulator under the same actuator parameterization given desired zero joint velocities and accelerations. A PID controller, similar to (78), is adopted to track the desired torques for the manipulator, where $\theta_{i,\mathrm{des}}(t) = \theta_i(0)$ and $\dot{\theta}_{i,\mathrm{des}}(t) = 0$.

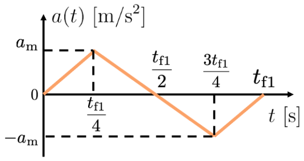


Figure 10. An illustration on the acceleration parameterization.

In the second phase, once the base arrives at the desired location, a motion planning problem, similar to (77), is solved given the fixed manipulator dynamics and the prescribed time duration t_f2. The PID controller for the manipulator in the first phase is used to track the desired torques returned by the optimization, where the feed-forward torques are the optimal torques returned by the fixed manipulator motion planning optimization. The PID controller for the base in the first phase is used to stabilize the base position at its desired one, with the feed-forward term $\boldsymbol{f}_{1,\mathrm{des}}(t)$ being zero. The closed-loop system of the sequential planning approach is also illustrated in Fig. 9, where the difference between two approaches is how the feed-forward terms, i.e. $\tau_{i,\mathrm{des}}(t)$ and $\boldsymbol{f}_{1,\mathrm{des}}(t)$, are calculated. The sequential planning problems with different parameters are denoted by: (1)

1) B1: the sequential planning with $\boldsymbol{a}_\mathrm{m} = \begin{bmatrix} 0.25 & 0.25 \end{bmatrix}^\top$ m/s^2;
2) B2: the same as B1, but $\boldsymbol{a}_\mathrm{m} = \begin{bmatrix} 0.50 & 0.50 \end{bmatrix}^\top$ m/s^2;
3) B3: the same as B1, but $\boldsymbol{a}_\mathrm{m} = \begin{bmatrix} 0.75 & 0.75 \end{bmatrix}^\top$ m/s^2;
4) B4: the same as B1, but $\boldsymbol{a}_\mathrm{m} = \begin{bmatrix} 1.00 & 1.00 \end{bmatrix}^\top$ m/s^2.

D. Comparison of Integrated and Sequential Planning Methods

This subsection compares the motion planning results from the proposed integrated planning approach and the sequential planning approach given different parameters. Each approach is tested for 20 trials, where the desired end effector position is randomly generated in the same way as Section VII-B does. The results are summarized in Table IV, where the column Time indicates the completion time in seconds. For the integrated motion planning method, only one set of PID gains is used for all t_f. For the sequential planning method, only one set of PID gains is used throughout two phases, for all $\boldsymbol{a}_\mathrm{m}$; $t_\mathrm{f2} = 2.5$ s is obtained by solving the time optimal motion planning problem given the fixed arm dynamics. Each set of gains is manually tuned to minimize the trajectory tracking error and the end effector position error.

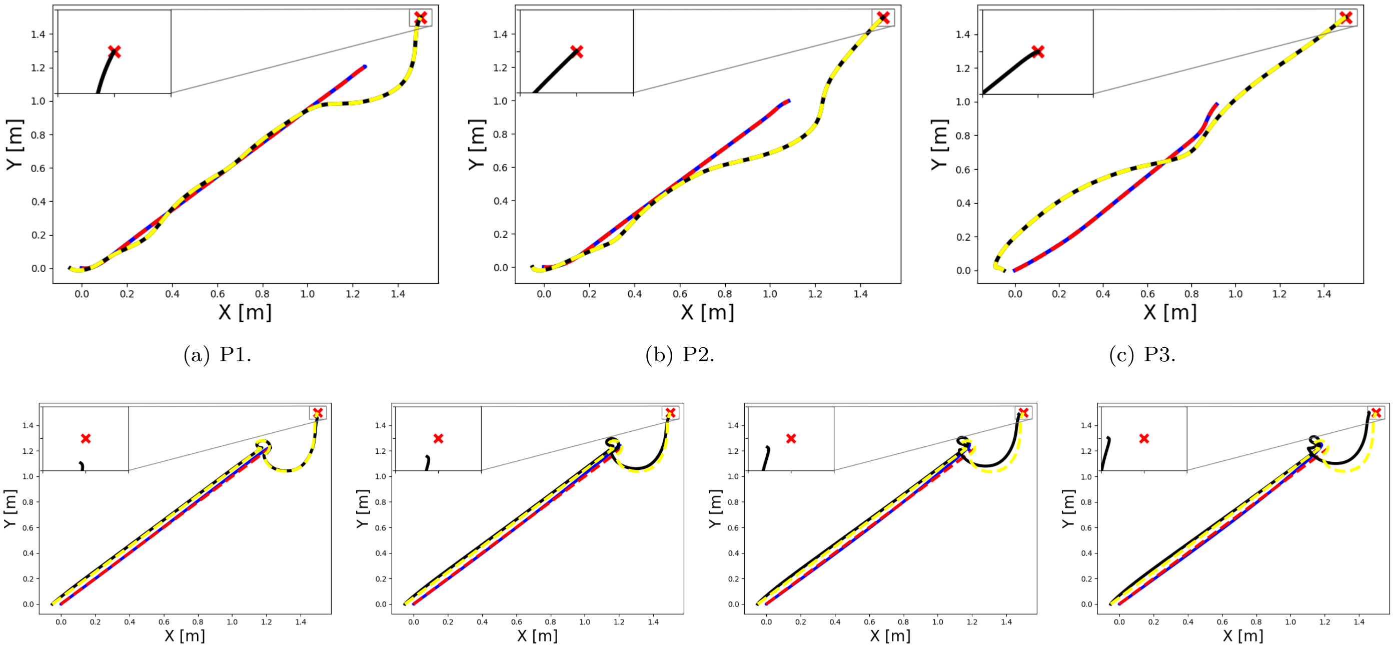
Table IV
Results From Multiple Planning Problems

Name	Time [s]	$\|\|\boldsymbol{u}\|\|_2$	Error [mm]	Success [%]
P1	5.0	6691±171	1.17±1.10	**100**
P2	3.5	13993±439	**0.27 ± 0.15**	**100**
P3	**2.0**	85705±6254	3.00±1.66	**100**
B1	9.35±0.22	**5453 ± 65**	38.68±9.83	90
B2	7.34±0.16	15289±157	36.26±11.19	80
B3	6.45±0.13	28017±264	34.64±11.93	85
B4	5.93±0.11	43090±423	34.35±6.98	90

According to Table IV, the integrated planning method achieves time-optimal motion with millimeter-level end effector positioning errors, compared to the sequential planning. When compared with B1, the sequential planning approach with the smallest acceleration, P1 reduces the completion time by almost half while consuming similar control effort. In fact, the integrated planning method with $t_\mathrm{f} = 6$ s completes the task with a control effort of approximately 4166, consuming less energy than B1 while still being more time-efficient. The optimal time of $t_\mathrm{f} = 2$ s is obtained by solving the time-optimal integrated motion planning given the full dynamics of the mobile manipulator, which guarantees dynamical feasibility. Thus, the motion planning with a final time greater than 2 s yields a 100% success rate in solving the motion planning optimization. By tuning the final time within the feasible range, one can customize motion planning to balance time and energy efficiency.

Fig. 11 illustrates how the mobile manipulator moves given different approaches and parameters. From Fig. 11a to Fig. 11c, the integrated planning with different final times all have precise trajectory tracking, and the zoom-in areas in all three subfigures show that the end effector reaches the desired goal position, even with the most aggressive but feasible task time of $t_\mathrm{f} = 2$ s. From Fig. 11d to Fig. 11g, where the sequential planning method is used, the manipulator fails to reach the desired position with the same precision. This is because the computation of the parameterized motion does not account for the dynamic coupling between the base and the manipulator. Thus, the dynamical coupling is assumed small enough so that it, being treated as a disturbance, can be compensated by a tracking controller. However, this assumption holds only if the desired motion has a small acceleration. Consequently, it is challenging for the sequential planning method to adjust the manipulator's configuration while the base is moving. Therefore, when a more time-efficient movement is required, the sequential planning method cannot perform as well as the integrated planning method.

Fig. 12 demonstrates the state and control trajectories of some joints given different approaches. When comparing the base movement in Fig. 12a - Fig. 12l, the sequential planning method struggles to precisely track the desired state trajectory in the second phase. This is due to the significant dynamical influence from the manipulator motion to the base, which is not considered and compensated for in the base motion. Similarly, in



(a) P1.

(b) P2.

(c) P3.

(d) B1.

(e) B2.

(f) B3.

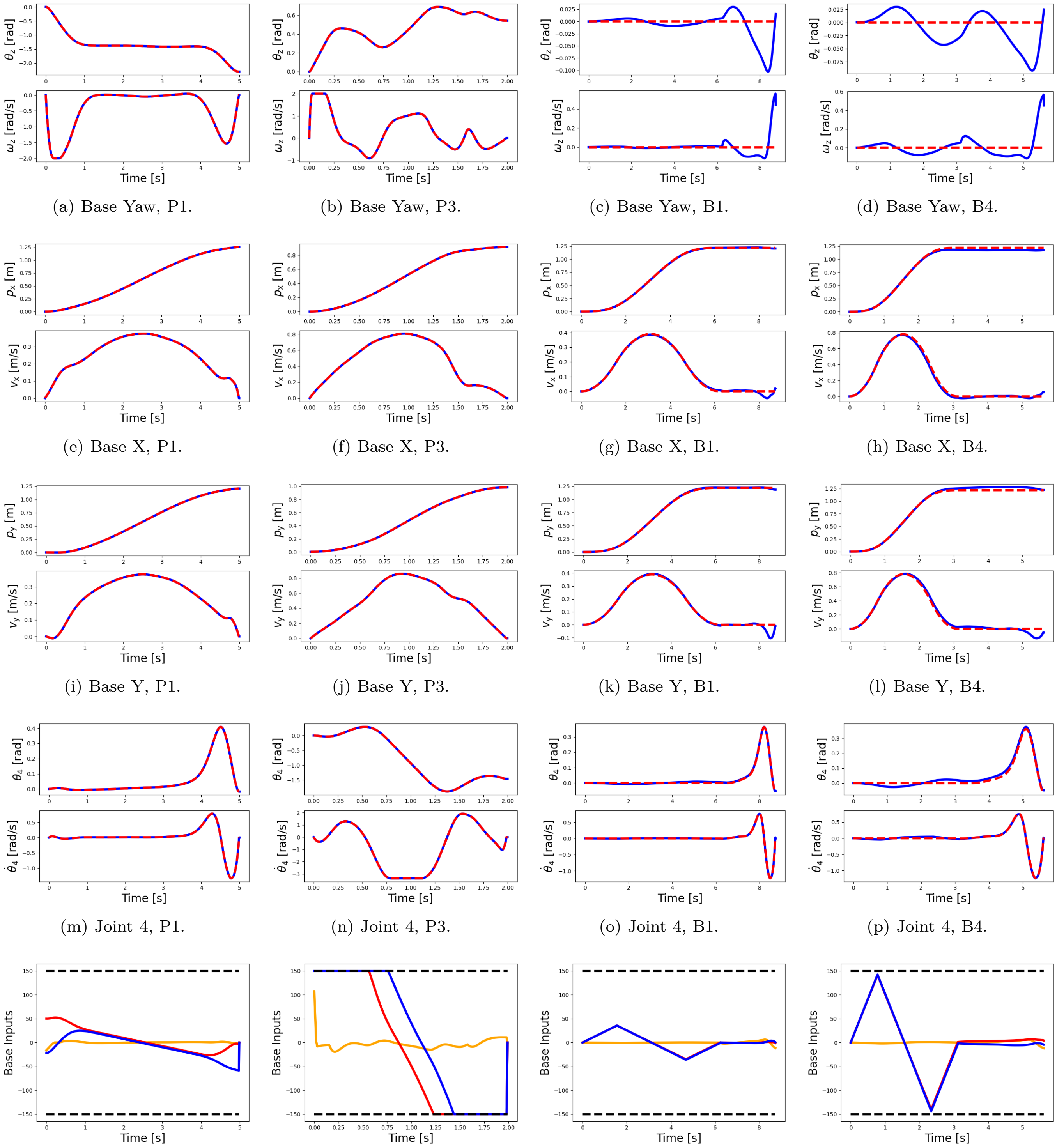
(g) B4.

Figure 11. The desired and actual position trajectories of the base and the end effector on the XOY plane for different approaches.

Fig. 12m - Fig. 12p, the sequential planning method fails to precisely track Joint 4's desired trajectory due to the dynamical influence from the base motion to the manipulator. Note that Joint 4 is the first joint of the manipulator that needs to generate torque to compensate for the gravity of the entire manipulator. Thus, any small tracking error or dynamics mismatch in Joint 4 can be amplified for the rest of the joints. Fig. 12q - Fig. 12t illustrates the control trajectory for different methods. Specifically, Fig. 12q - Fig. 12r demonstrates that the base yaw torque is constantly compensating for the dynamical effect between the base and the manipulator, especially for the most aggressive P3. This shows that for the sequential planning method, the desired yaw torque, i.e. the feed-forward term in the PID tracking controller, in both phases cannot constantly equal zero. Without modeling the dynamics of the entire mobile manipulator, one cannot calculate how much yaw torque is required to compensate for the motion. In other words, the base yaw torque in Fig. 12r compensates for the dynamic coupling wrench, whereas the sequential planning cannot compensate it well because it assumes that the desired wrench, i.e. the feed-forward term, is zero. Part 1 of the supplementary video demonstrates how the mobile manipulator behaves given different approaches.

E. Comparison with Measurement Noise

This subsection compares the motion planning results given measurement noise. The state observed at each time instance equals the true state plus a multivariate Gaussian noise with zero mean and a diagonal covariance matrix. In other words, each entry of the state follows an independent Gaussian distribution. Particularly, the localization and pose estimation of the mobile base are less accurate than the measurement of the manipulator joints' angular positions and velocities. The variance for joint angular positions and velocities is 1×10^{-5} rad due to accurate measurement. The variances for the base yaw angular position and velocity are 5.24×10^{-3} rad (0.3 degrees) and 1.75×10^{-3} rad/s (0.1 deg/s), respectively. The variances for the base position (along the x- and y-axis) and velocity are 0.02 m and 0.006 m/s, respectively. Note that the measurement for the base is quite noisy, as 99.7% of the position measurements in each axis lie within an error range of ±60 millimeters. The measured state is directly fed into the tracking controller of each approach, without any filtering or estimator. Each approach is performed for 20 different trials which are randomly generated in the same way as Section VII-D does. The result is summarized in Table V, where P2 and P3 perform similarly despite measurement noise. Given that 99.7% of the position measurements lie within an error range of ±60 millimeters, the end effector error of P2 and P3 is acceptable. Fig. 13 illustrates how the mobile manipulator moves to reach the desired position given measurement noise. Part 2 of the supplementary video demonstrates how the mobile manipulator tracks the desired trajectories under measurement noise. Adopting a noise filter or a state estimator such as an Extended Kalman Filter (EKF) for trajectory tracking might further reduce the tracking error.



(a) Base Yaw, P1.
(b) Base Yaw, P3.
(c) Base Yaw, B1.
(d) Base Yaw, B4.

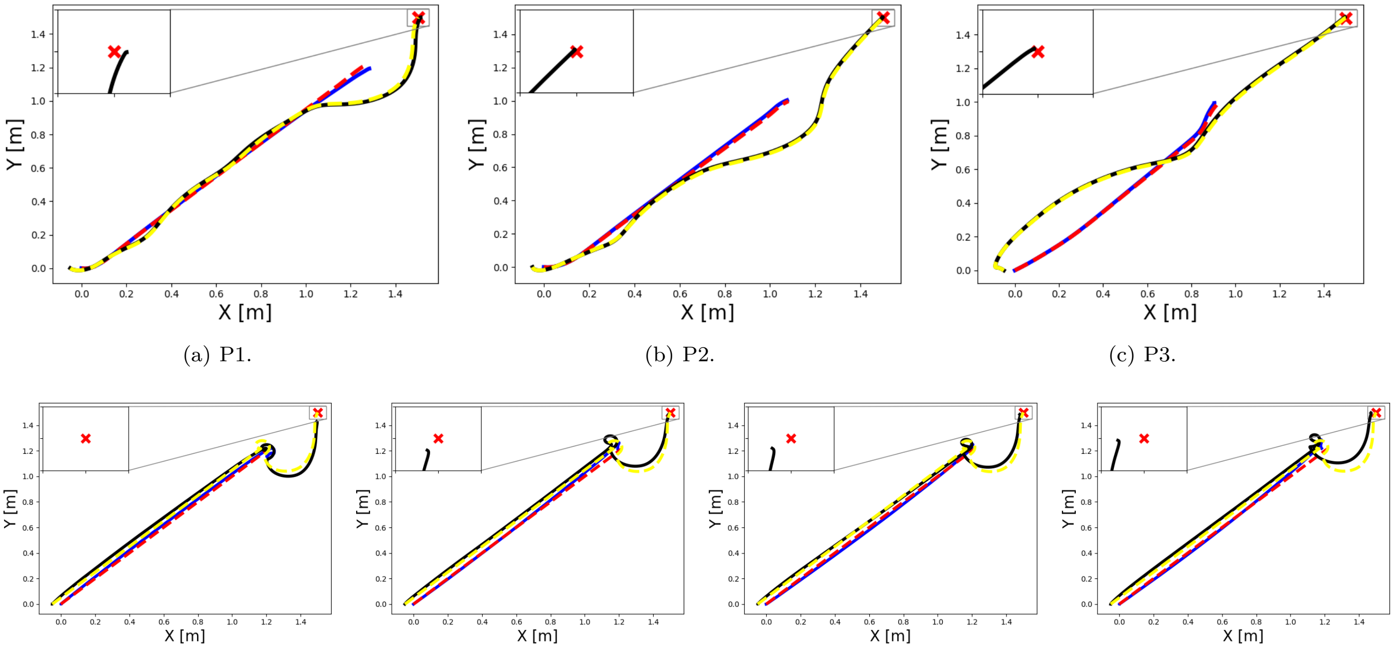
(e) Base X, P1.
(f) Base X, P3.
(g) Base X, B1.
(h) Base X, B4.

(i) Base Y, P1.
(j) Base Y, P3.
(k) Base Y, B1.
(l) Base Y, B4.

(m) Joint 4, P1.
(n) Joint 4, P3.
(o) Joint 4, B1.
(p) Joint 4, B4.

(q) Base Inputs, P1.
(r) Base Inputs, P3.
(s) Base Inputs, B1.
(t) Base Inputs, B4.

Figure 12. The state and control trajectories of some joints by different approaches.



(a) P1.

(b) P2.

(c) P3.

(d) B1.

(e) B2.

(f) B3.

(g) B4.

Figure 13. The desired and actual position trajectories of the base and the end effector on the XOY plane for different approaches with measurement noise.

Table V
Results With Measurement Noise

Name	Time [s]	$\|\|\boldsymbol{u}\|\|_2$	Error [mm]	Success [%]
P1	5.0	6697±173	45.48±23.82	**100**
P2	3.5	13994±438	11.10±9.03	**100**
P3	**2.0**	85551±6245	**6.23 ± 5.30**	**100**
B1	9.35±0.22	**5990 ± 329**	57.96±26.04	90
B2	7.34±0.16	15569±440	39.63±16.76	80
B3	6.45±0.13	28261±363	35.68±12.85	85
B4	5.93±0.11	43344±528	33.65±12.13	90

VIII. Simultaneous Actuator Design and Integrated Motion Planning

This section illustrates the effectiveness of the proposed modeling approach for mobile manipulator co-design, particularly by showcasing a simultaneous actuator design and motion planning framework. Specifically, simultaneous design involves solving a motion planning problem similar to (77), while also treating the motor design parameters $\boldsymbol{\beta}$ as decision variables, along with certain motor design constraints. The following subsections present the simultaneous design formulation and the corresponding numerical results.

A. Formulation

This subsection introduces the simultaneous design formulation. First, Motor j's motor design parameter $\boldsymbol{\beta}_j$ is subject to the following design constraints:

$$l_j \in [20, 100], r_{\mathrm{ro},j} \in [10, 100], r_{\mathrm{so},j} \in [10, 100], h_{\mathrm{m},j} \in [1, 5], \tag{83a}$$

$$h_{\mathrm{sy},j} \in [5, 10], w_{\mathrm{tooth},j} \in [5, 20], b_{0,j} \in [1, 10], \tag{83b}$$

$$h_{\mathrm{ss},j} > 0 \text{ mm}, \quad D_{\mathrm{wire},j} \geq 0.6 \text{ mm}, \quad k_{\mathrm{C},j} > 0, \tag{83c}$$

$$0 < \arcsin(\tfrac{w_{\mathrm{tooth},j}}{2(r_{\mathrm{ro},j}+\delta)}) + \arcsin(\tfrac{b_{0,j}}{2(r_{\mathrm{ro},j}+\delta)}) \leq \tfrac{\pi}{Q}, \tag{83d}$$

$$0 < m_{\mathrm{stator},j} + m_{\mathrm{rotor},j} \leq 3 \text{ kg}, \quad m_{\mathrm{stator},j} > 0 \text{ kg}, \tag{83e}$$

$$0 \text{ T} < \frac{k_{\mathrm{p}}\Phi_{1,j}}{w_{\mathrm{tooth},j} l_j} \leq 1.5 \text{ T}, \quad 0 \text{ T} < \frac{k_{\mathrm{p}}\Phi_{1,j}}{\sqrt{3} h_{\mathrm{sy},j} l_j} \leq 1.5 \text{ T}, \tag{83f}$$

$$A_{\mathrm{slot},j} > 0 \text{ mm}^2, \quad A_{\mathrm{so},j} > 0 \text{ mm}^2, \tag{83g}$$

where (83c) describe the minimal slot height, motor's minimal wire diameter, and minimal Carter's coefficient; (83d) and (83e) describe the tooth width bound and the motor weight bound, respectively; (83f) describes the magnetic flux bounds in the tooth and the stator yoke; (83g) describes the minimal slot area and the minimal cross-section area of the stator core. δ is a constant defined in Section IV-A; $h_{\mathrm{ss},j}$, $D_{\mathrm{wire},j}$, $m_{\mathrm{stator},j}$, $m_{\mathrm{rotor},j}$, $\Phi_{1,j}$, k_{p}, $k_{\mathrm{C},j}$ can be calculated by (30) - (35) and (40) - (43) from Section IV-A. The design constraints (83a) - (83b) can be summarized as a closed convex set, i.e. $\boldsymbol{\beta}_j \in \boldsymbol{\mathcal{B}} \subset \mathbb{R}^7$; the design constraints (83c) - (83g) can be written as an inequality constraint, i.e. $\boldsymbol{g}_{\mathrm{d}}(\boldsymbol{\beta}_j) \leq \mathbf{0}$.

Thus, the simultaneous actuator design and motion planning problem can be written as follows:

$$\min_{(*)} \quad \frac{t_{\mathrm{f}}}{N} \textstyle\sum_{k=0}^{N-1} \sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{B}_j ||\boldsymbol{u}_k||_2^2 \tag{84a}$$

$$\text{s.t.} \quad \boldsymbol{\beta}_r \in \boldsymbol{\mathcal{B}}, \ \boldsymbol{g}_{\mathrm{d}}(\boldsymbol{\beta}_r) \leq \mathbf{0}, \tag{84b}$$

$$\frac{t_{\mathrm{f}}}{N} \boldsymbol{f}_{\mathrm{c}}(\boldsymbol{x}_{k,i}, \boldsymbol{u}_k, \boldsymbol{\beta}) - \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{C}_{j,i} \boldsymbol{x}_{k,j} = \mathbf{0}, \tag{84c}$$

$$\boldsymbol{x}_{k+1,0} - \textstyle\sum_{j=0}^{n_{\mathrm{p}}} \boldsymbol{D}_j \boldsymbol{x}_{k,j} = \mathbf{0}, \tag{84d}$$

$$\underline{\boldsymbol{x}} \leq \boldsymbol{x}_{k,i} \leq \overline{\boldsymbol{x}}, \tag{84e}$$

$$\forall k \in \llbracket 0, N-1 \rrbracket, i \in \llbracket 1, n_{\mathrm{p}} \rrbracket, \text{ given } \boldsymbol{x}_0, \tag{84f}$$

$$\tau_{k,r} - \tau_{\max,r}(Z_r \theta_{k,i,r}, \boldsymbol{\beta}_r) \leq 0, \tag{84g}$$

$$\tau_{\min,r}(Z_r \theta_{k,i,r}, \boldsymbol{\beta}_r) - \tau_{k,r} \leq 0, \tag{84h}$$

$$\underline{\boldsymbol{f}_1} \leq \boldsymbol{f}_{k,1} \leq \overline{\boldsymbol{f}_1}, \tag{84i}$$

$$\forall k \in \llbracket 0, N-1 \rrbracket, i \in \llbracket 1, n_{\mathrm{p}} \rrbracket, r \in \llbracket 3, n+2 \rrbracket, \tag{84j}$$

$$||\boldsymbol{p}_{J_{\mathrm{e}}}(\boldsymbol{x}_{N,0}) - \boldsymbol{p}_{J_{\mathrm{e}},\mathrm{des}}||_2 = 0, \tag{84k}$$

where $(*)$ denotes all the decision variables, i.e. $\boldsymbol{\beta} \in \mathbb{R}^{7n}$, $\boldsymbol{x}_{k,i}, \forall k \in \llbracket 0, N-1 \rrbracket, i \in \llbracket 1, n_{\mathrm{p}} \rrbracket$, $\boldsymbol{u}_k, \forall k \in \llbracket 0, N-1 \rrbracket$, and $\boldsymbol{x}_{N,0}$.

Since the simultaneous design optimization (84) includes motor design parameters $\boldsymbol{\beta}$ as decision variables, the objective function and the constraints of (84) need to be differentiable to $\boldsymbol{\beta}$. According to Section IV-C, the piece-wise analytical motor max/min torque functions from (54), (55) and (63) are not differentiable to $\boldsymbol{\beta}$. However, as discussed in Section IV-B, when the motor speed exceeds $\omega_{\mathrm{r},j} = \omega_{\mathrm{ce},j}(\boldsymbol{\beta}_j)/p$, the maximum torque starts to decrease from its constant value. This decrease occurs because either the voltage or the current increases, activating the voltage and/or current constraint. From the perspective of minimizing the motor input power (i.e. the product of current and voltage), it is reasonable to constrain the motor speed to be less than $\omega_{\mathrm{ce},j}(\boldsymbol{\beta}_j)/p$. Consequently, the motor's maximum torque remains constant, determined by a function of $\boldsymbol{\beta}$, regardless of the sign of $\Phi_{\mathrm{pm},j}/L_{\mathrm{d},j} - I_{\max,j}$. Therefore, for all Motor r, $r \in \llbracket 3, n+2 \rrbracket$, the motor torque constraints (84g) and (84h) are reduced to

$$\tau_{k,r} - \tau_{\max,r}(\boldsymbol{\beta}_r) \leq 0, \quad -\tau_{\max,r}(\boldsymbol{\beta}_r) - \tau_{k,r} \leq 0, \tag{85}$$

where $\tau_{\max,r} := 1.5 p \Phi_{\mathrm{pm},r} I_{\max,r}$. For every time instance and collocation point, i.e. $\forall k \in \llbracket 0, N-1 \rrbracket, i \in \llbracket 1, n_{\mathrm{p}} \rrbracket$, the joint speed is constrained by

$$\theta_{k,i,r} - \frac{\omega_{\mathrm{ce},r}(\boldsymbol{\beta}_r)}{Z_r p} \leq 0, \quad -\frac{\omega_{\mathrm{ce},r}(\boldsymbol{\beta}_r)}{Z_r p} - \theta_{k,i,r} \leq 0, \tag{86}$$

where $\omega_{\mathrm{ce},r}(\boldsymbol{\beta}_r)$ is given by (46).

B. Numerical Results

This subsection presents the numerical results of the simultaneous actuator design and motion planning. The mobile manipulator is desired to carry a 5 kg solid iron ball to reach a terminal position. The final time is 2 seconds. The initial guess of the motor design parameters is the empirical design used in Section VII. The empirical and optimized motor designs are visualized in Fig. 14, where the magnet height and tooth width of all motors are decreased to reduce the unnecessary permanent flux Φ_{m} and thereby motor torque capacity; the axial length, rotor radius, and stator radius of all motors are decreased to reduce the unnecessary motor mass and inertia. Since Motor 4 is the heaviest motor in the arm to compensate for the entire arm's gravity, the rotor radius, magnet height, and tooth width of Motor 4 are greater than the others.

Table VI
Motion Planning Results with Empirical and Optimized Motor Design

Item	Empirical	Optimized	Reduced by
Motor Mass [kg]	18.00	2.44	86.44%
$\|\|\boldsymbol{u}\|\|_2$	68667	49039	28.58%

The motion planning results based on empirical and optimized motor designs are summarized in Table VI. The total motor mass of the arm and the required control effort are significantly reduced by 86.44% and 28.58%, respectively, with the optimized design. Comparing both the integrated motion planning and sequential planning summarized in Table IV given the empirical design, the integrated motion planning with the optimized design consumes a similar amount of control effort as B4. But it achieves a more time-efficient motion, with its final time being almost one-third of B4's. These results verify the effectiveness of the simultaneous design. Part 3 of the supplementary video demonstrates how the mobile manipulator reaches the desired position given the empirical and optimized motor design.

While this specific simultaneous design scheme requires differentiability on design parameters, other co-design schemes, like the sequential co-design scheme in [50], do not. For instance, one can formulate two optimization problems and solve them sequentially. The first is the motion planning optimization with specific motor design parameters, while the second is a motor design optimization to minimize operational efficiency based on the optimal trajectory obtained from the motion planning optimization. In this case, the differentiability of the motor torque and speed constraints to the design parameters is relaxed for the motion planning optimization. This section highlights the capability of the proposed modeling approach for robot co-design by showcasing the simultaneous actuator design and motion planning framework.

IX. Conclusions and Future Work

This paper investigates differentiable dynamic modeling for mobile manipulators, where actuators are parameterized by physically meaningful motor geometry parameters. These parameters affect various aspects of the manipulator, such as link mass, inertia, center-of-mass, control constraints (motor torque capacity), and state constraints (motor maximum speed), all of which influence the performance of the motion planning and closed-loop control system. This paper presents an analytical model for the motor's maximum torque and speed and describes how design parameters impact the dynamics, enabling differentiable and analytical dynamic modeling. Furthermore, using the proposed differentiable dynamics and motor parameterization, this paper formulates an integrated locomotion and manipulation planning problem, discretized by direct collocation. The effectiveness of the differentiable dynamics is demonstrated through numerical experiments, showing improvements in computation

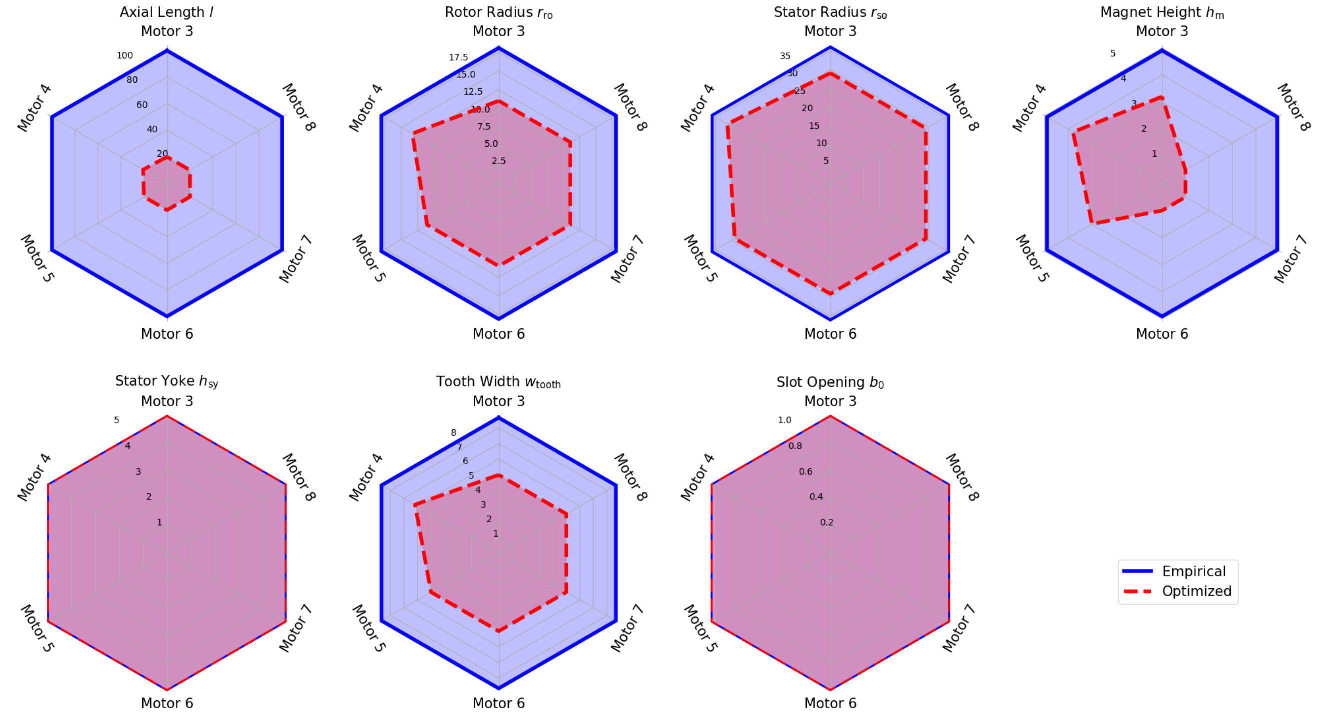


Figure 14. Empirical and optimized motor design parameters.

efficiency, task completion time, and energy consumption, compared to the prevailing sequential motion planning approach. Lastly, this paper introduces a framework for simultaneous actuator design and motion planning, offering numerical results to validate the effectiveness of the proposed differentiable modeling approach for co-design problems.

The limitations of the proposed modeling approach can be summarized in three main aspects. Firstly, this paper simplifies the mobile base as one rigid body with one 3-DOF planar joint on SE(2). This choice is made to focus on the methodology and algorithm for modeling the forward and inverse dynamics of the entire chain of rigid bodies based on motor parameterization. However, to model the full 6-degree-of-freedom motion of a mobile base, including pitch and roll behavior, a specific mechanical configuration for the mobile base, including wheel-ground contact, must be defined. This would introduce additional modeling complexity beyond the scope of this paper. For example, [46] investigate the high-fidelity modeling of a four-wheeled car-like mobile robot, including the modeling of contact forces between the wheels and the ground, as well as how the terrain affects the robot's dynamics. A future improvement of the proposed modeling approach could include similar techniques used in [46].

Secondly, this paper neglects the dynamics of the gearboxes, typically harmonic drives, in the manipulator. While harmonic drives are generally compact, and their dynamics are often negligible compared to the manipulator's dynamics, modeling a harmonic drive with major design parameters could increase the modeling fidelity for the entire robot. This would further enable the integrated design of motors and gearboxes.

Thirdly, the proposed modeling approach does not account for friction in each joint. Actual friction occurs at the contact surface between each motor's rotor and its corresponding gearhead. While some literature [56]–[58] models a manipulator's joint friction as a function of joint velocity, joint friction is also affected by the temperature of the contact surface, and any friction-related parameters could vary depending on the actual lubrication condition [57], [58]. Therefore, from the perspective of motion planning, it would be beneficial to consider unmodeled joint friction as an unknown disturbance. Designing planning and control techniques to compensate for this unknown disturbance is necessary for high-precision motion. From the perspective of robot co-design, reasonably modeling joint friction could make the co-design result more realistic. For instance, friction, as a function of joint velocity, may restrict fast joint movement to save energy or improve energy efficiency.

Regarding the proposed integrated locomotion and manipulation planning, this paper emphasizes that considering the entire dynamics of mobile manipulators could yield better motion performance compared to the most common sequential planning approach. Modeling a motor's maximum torque/speed as analytical functions ensures that the corresponding motion does not violate the control authority of the motors. Even though the proposed integrated motion planning is not a real-time algorithm, it provides insights into how the proposed modeling approach could benefit both offline and online motion planning. A differentiable dynamic model is necessary for many state-of-the-art online motion planning algorithms, according to this paper's literature review.

References

[1] H. Christensen, N. Amato, H. Yanco, M. Mataric, H. Choset, A. Drobnis, K. Goldberg, J. Grizzle, G. Hager, J. Hollerbach et al., "A roadmap for us robotics–from internet to robotics 2020 edition," Foundations and Trends® in Robotics, vol. 8, no. 4, pp. 307–424, 2021.

[2] S. Thakar, S. Srinivasan, S. Al-Hussaini, P. M. Bhatt, P. Rajendran, Y. Jung Yoon, N. Dhanaraj, R. K. Malhan, M. Schmid, V. N. Krovi et al., "A survey of wheeled mobile manipulation: A decision-making perspective," Journal of Mechanisms and Robotics, vol. 15, no. 2, p. 020801, 2023.

[3] A. Yamashita, T. Arai, J. Ota, and H. Asama, "Motion planning of multiple mobile robots for cooperative manipulation and transportation," IEEE Transactions on Robotics and Automation, vol. 19, no. 2, pp. 223–237, 2003.

[4] F. Chen, M. Selvaggio, and D. G. Caldwell, "Dexterous grasping by manipulability selection for mobile manipulator with visual guidance," IEEE Transactions on Industrial Informatics, vol. 15, no. 2, pp. 1202–1210, 2018.

[5] S. Thakar, P. Rajendran, A. M. Kabir, and S. K. Gupta, "Manipulator motion planning for part pickup and transport operations from a moving base," IEEE Transactions on Automation Science and Engineering, vol. 19, no. 1, pp. 191–206, 2020.

[6] Z. Li, P. Moran, Q. Dong, R. J. Shaw, and K. Hauser, "Development of a tele-nursing mobile manipulator for remote care-giving in quarantine areas," in 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017, pp. 3581–3586.

[7] J. Stückler, M. Schwarz, M. Schadler, A. Topalidou-Kyniazopoulou, and S. Behnke, "Nimbro explorer: Semiautonomous exploration and mobile manipulation in rough terrain," Journal of Field Robotics, vol. 33, no. 4, pp. 411–430, 2016.

[8] M. Naazare, F. G. Rosas, and D. Schulz, "Online next-best-view planner for 3d-exploration and inspection with a mobile manipulator robot," IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 3779–3786, 2022.

[9] D. Jud, P. Leemann, S. Kerscher, and M. Hutter, "Autonomous free-form trenching using a walking excavator," IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 3208–3215, 2019.

[10] F. Aghili, "A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics," IEEE Transactions on Robotics, vol. 28, no. 3, pp. 634–649, 2012.

[11] Y. Wang, E. Hansen, and H. Ahn, "Hierarchical planning for autonomous parking in dynamic environments," IEEE Transactions on Control Systems Technology, 2024.

[12] T. Sandakalum and M. H. Ang Jr, "Motion planning for mobile manipulators—a systematic review," Machines, vol. 10, no. 2, p. 97, 2022.

[13] S. G. Tzafestas, "Mobile robot control and navigation: A global overview," Journal of Intelligent & Robotic Systems, vol. 91, pp. 35–58, 2018.

[14] D. Berenson, S. S. Srinivasa, D. Ferguson, and J. J. Kuffner, "Manipulation planning on constraint manifolds," in 2009 IEEE International Conference on Robotics and Automation. IEEE, 2009, pp. 625–632.

[15] D. Berenson, S. Srinivasa, and J. Kuffner, "Task space regions: A framework for pose-constrained manipulation planning," The International Journal of Robotics Research, vol. 30, no. 12, pp. 1435–1460, 2011.

[16] M. Zucker, N. Ratliff, A. D. Dragan, M. Pivtoraiko, M. Klingensmith, C. M. Dellin, J. A. Bagnell, and S. S. Srinivasa, "Chomp: Covariant hamiltonian optimization for motion planning," The International Journal of Robotics Research, vol. 32, no. 9-10, pp. 1164–1193, 2013.

[17] M. Rickert, A. Sieverling, and O. Brock, "Balancing exploration and exploitation in sampling-based motion planning," IEEE Transactions on Robotics, vol. 30, no. 6, pp. 1305–1317, 2014.

[18] T. Gold, A. Völz, and K. Graichen, "Model predictive interaction control for robotic manipulation tasks," IEEE Transactions on Robotics, vol. 39, no. 1, pp. 76–89, 2022.

[19] P. Holmes, S. Kousik, B. Zhang, D. Raz, C. Barbalata, M. J. Roberson, and R. Vasudevan, " Reachable Sets for Safe, Real-Time Manipulator Trajectory Design," in Proceedings of Robotics: Science and Systems, Corvalis, Oregon, USA, July 2020.

[20] J. B. Michaux, Y. S. Kwon, Q. Chen, and R. Vasudevan, "Reachability-based Trajectory Design with Neural Implicit Safety Constraints," in Proceedings of Robotics: Science and Systems, Daegu, Republic of Korea, July 2023.

[21] T. Marcucci, M. Petersen, D. von Wrangel, and R. Tedrake, "Motion planning around obstacles with convex optimization," Science Robotics, vol. 8, no. 84, p. eadf7843, 2023.

[22] J. Michaux, P. Holmes, B. Zhang, C. Chen, B. Wang, S. Sahgal, T. Zhang, S. Dey, S. Kousik, and R. Vasudevan, "Can't touch this: Real-time, safe motion planning and control for manipulators under uncertainty," arXiv preprint arXiv:2301.13308, 2023.

[23] Z. Brei, J. Michaux, B. Zhang, P. Holmes, and R. Vasudevan, "Serving time: Real-time, safe motion planning and control for manipulation of unsecured objects," arXiv preprint arXiv:2309.03111, 2023.

[24] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, "Analysis and observations from the first amazon picking challenge," IEEE Transactions on Automation Science and Engineering, vol. 15, no. 1, pp. 172–188, 2016.

[25] A. De Luca, G. Oriolo, and P. R. Giordano, "Kinematic modeling and redundancy resolution for nonholonomic mobile manipulators," in Proceedings 2006 IEEE International Conference on Robotics and Automation. IEEE, 2006, pp. 1867–1873.

[26] C. P. Tang, P. T. Miller, V. N. Krovi, J.-C. Ryu, and S. K. Agrawal, "Differential-flatness-based planning and control of a wheeled mobile manipulator—theory and experiment," IEEE/ASME Transactions on Mechatronics, vol. 16, no. 4, pp. 768–773, 2010.

[27] J. Tan, N. Xi, and Y. Wang, "Integrated task planning and control for mobile manipulators," The International Journal of Robotics Research, vol. 22, no. 5, pp. 337–354, 2003.

[28] S. Furuno, M. Yamamoto, and A. Mohri, "Trajectory planning of mobile manipulator with stability considerations," in 2003 IEEE International Conference on Robotics and Automation, vol. 3. IEEE, 2003, pp. 3403–3408.

[29] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," IEEE Transactions on Robotics and Automation, vol. 12, no. 4, pp. 566–580, 1996.

[30] S. M. LaValle and J. J. Kuffner Jr, "Randomized kinodynamic planning," The International Journal of Robotics Research, vol. 20, no. 5, pp. 378–400, 2001.

[31] J. Vannoy and J. Xiao, "Real-time adaptive motion planning (ramp) of mobile manipulators in dynamic environments with unforeseen changes," IEEE Transactions on Robotics, vol. 24, no. 5, pp. 1199–1212, 2008.

[32] M. Stilman, "Global manipulation planning in robot joint space with task constraints," IEEE Transactions on Robotics, vol. 26, no. 3, pp. 576–584, 2010.

[33] B. J. Cohen, S. Chitta, and M. Likhachev, "Search-based planning for manipulation with motion primitives," in 2010 IEEE International Conference on Robotics and Automation. IEEE, 2010, pp. 2902–2908.

[34] S. Chitta, B. Cohen, and M. Likhachev, "Planning for autonomous door opening with a mobile manipulator," in 2010 IEEE International Conference on Robotics and Automation. IEEE, 2010, pp. 1799–1806.

[35] A. Hornung, M. Phillips, E. G. Jones, M. Bennewitz, M. Likhachev, and S. Chitta, "Navigation in three-dimensional cluttered environments for mobile manipulation," in 2012 IEEE International Conference on Robotics and Automation. IEEE, 2012, pp. 423–429.

[36] Y. Wang, D. K. Jha, and Y. Akemi, "A two-stage RRT path planner for automated parking," in 2017 13th IEEE Conference on Automation Science and Engineering (CASE), 2017, pp. 496–502.

[37] V. Pilania and K. Gupta, "Mobile manipulator planning under uncertainty in unknown environments," The International Journal of Robotics Research, vol. 37, no. 2-3, pp. 316–339, 2018.

[38] D. Honerkamp, T. Welschehold, and A. Valada, "N^2m^2: Learning navigation for arbitrary mobile manipulation motions in unseen and dynamic environments," IEEE Transactions on Robotics, 2023.

[39] D. Berenson, J. Kuffner, and H. Choset, "An optimization approach to planning for mobile manipulation," in 2008 IEEE International Conference on Robotics and Automation. IEEE, 2008, pp. 1187–1192.

[40] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel, "Motion planning with sequential convex optimization and convex collision checking," The International Journal of Robotics Research, vol. 33, no. 9, pp. 1251–1270, 2014.

[41] F. Marić, O. Limoyo, L. Petrović, T. Ablett, I. Petrović, and J. Kelly, "Fast manipulability maximization using continuous-time trajectory optimization," in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2019, pp. 8258–8264.

[42] Y. Zhao, Y. Wang, M. Zhou, and J. Wu, "Energy-optimal collision-free motion planning for multiaxis motion systems: An alternating quadratic programming approach," IEEE Transactions on Automation Science and Engineering, vol. 16, no. 1, pp. 327–338, 2019.

[43] S. Kousik, S. Vaskov, M. Johnson-Roberson, and R. Vasudevan, "Safe trajectory synthesis for autonomous driving in unforeseen environments," in Dynamic Systems and Control Conference, vol. 58271. American Society of Mechanical Engineers, 2017, p. V001T44A005.

[44] M. Chen, S. L. Herbert, H. Hu, Y. Pu, J. F. Fisac, S. Bansal, S. Han, and C. J. Tomlin, "Fastrack: a modular framework for real-time motion planning and guaranteed safe tracking," IEEE Transactions on Automatic Control, vol. 66, no. 12, pp. 5861–5876, 2021.

[45] M. Korayem and S. Dehkordi, "Derivation of motion equation for mobile manipulator with viscoelastic links and revolute–prismatic flexible joints via recursive gibbs–appell formulations," Robotics and Autonomous Systems, vol. 103, pp. 175–198, 2018.

[46] N. Seegmiller and A. Kelly, "High-fidelity yet fast dynamic models of wheeled mobile robots," IEEE Transactions on Robotics, vol. 32, no. 3, pp. 614–625, 2016.

[47] R. Featherstone, Rigid Body Dynamics Algorithms. Springer, 2014.

[48] J. Carpentier and N. Mansard, "Analytical derivatives of rigid body dynamics algorithms," in Proceedings of Robotics: Science and Systems, Pittsburgh, Pennsylvania, June 2018.

[49] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The pinocchio c++ library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in 2019 IEEE/SICE International Symposium on System Integration (SII). IEEE, 2019, pp. 614–619.

[50] A. Stein, Y. Wang, Y. Sakamoto, B. Wang, and H. Fang, "Application-oriented co-design of motors and motions for a 6dof robot manipulator," arXiv preprint arXiv:2310.03132, 2023.

[51] T. Higuchi, T. Abe, and Y. Yokoi, PRINCIPLE AND DESIGN OF AC MOTORS (in Japanese). Kagakujohoshuppan, 2017.

[52] K. M. Lynch and F. C. Park, Modern Robotics. Cambridge University Press, 2017.

[53] F. Magnusson, "Collocation methods in jmodelica.org," ISSN 0280-5316, 2012.

[54] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "Casadi: a software framework for nonlinear optimization and optimal control," Mathematical Programming Computation, vol. 11, pp. 1–36, 2019.

[55] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," Mathematical Programming, vol. 106, pp. 25–57, 2006.

[56] M. R. Kermani, R. V. Patel, and M. Moallem, "Friction identification and compensation in robotic manipulators," IEEE Transactions on Instrumentation and Measurement, vol. 56, no. 6, pp. 2346–2353, 2007.

[57] L. Simoni, M. Beschi, G. Legnani, and A. Visioli, "Modelling the temperature in joint friction of industrial manipulators," Robotica, vol. 37, no. 5, pp. 906–927, 2019.

[58] R. Pagani, G. Legnani, G. Incerti, and M. Gheza, "Evaluation and modeling of the friction in robotic joints considering thermal effects," Journal of Mechanisms and Robotics, vol. 12, no. 2, p. 021108, 2020.