

Enhanced Agility and Safety in Mobile Manipulators through Centroidal Momentum-Based Motion Planning

Dai, Min; Lu, Zehui; Li, Na; Wang, Yebin

TR2025-092 June 26, 2025

Abstract

This paper presents a framework for the planning and control of wheeled mobile manipulators, integrating a novel reduced-order dynamics model that can be used during motion planning to enhance safety and prevent tip-overs. Leveraging centroidal momentum dynamics, the model captures key forces at the center of mass, enabling efficient, dynamically feasible trajectory generation that considers both manipulator motion and ground interaction forces. Unlike traditional methods that rely on conservative planning or separate reactive tip-over prevention mechanisms, our approach incorporates stability considerations directly into the motion planning phase. By embedding the zero moment point criterion within the model, our framework ensures tip-over prevention even during fast and payload-intensive tasks. Simulations demonstrate the effectiveness of this approach, achieving stable and efficient task execution across various scenarios.

European Control Conference (ECC) 2025

© 2025 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

Enhanced Agility and Safety in Mobile Manipulators through Centroidal Momentum-Based Motion Planning

Min Dai, Zehui Lu, Na Li, and Yebin Wang

Abstract—This paper presents a framework for the planning and control of wheeled mobile manipulators, integrating a novel reduced-order dynamics model that can be used during motion planning to enhance safety and prevent tip-overs. Leveraging centroidal momentum dynamics, the model captures key forces at the center of mass, enabling efficient, dynamically feasible trajectory generation that considers both manipulator motion and ground interaction forces. Unlike traditional methods that rely on conservative planning or separate reactive tip-over prevention mechanisms, our approach incorporates stability considerations directly into the motion planning phase. By embedding the zero moment point criterion within the model, our framework ensures tip-over prevention even during fast and payload-intensive tasks. Simulations demonstrate the effectiveness of this approach, achieving stable and efficient task execution across various scenarios.

I. INTRODUCTION

Wheeled mobile manipulators combine the advantages of both mobile platforms and robotic manipulators, significantly extending their workspace and flexibility compared to conventional fixed-base manipulators [1], [2]. This capability enables their application across a wide range of industrial tasks, from factory automation to service robots. However, achieving fast and safe motion control remains a challenging problem, primarily due to the inherent instability caused by the unilateral contact forces between the wheels and the ground. Stability concerns, especially the risk of tipping, complicate agile motion planning for these systems.

Existing research has addressed the coordination of mobile bases and manipulators in various ways. Traditional approaches [3], [4] often decouple the planning of the base and manipulator, leading to sequential movements—first moving the base into position and then controlling the manipulator for precision tasks. These methods are slow due to the sequential movement, and tight velocity limits are imposed to reduce the impact of dynamic coupling between the base and the manipulator and to prevent tip-over of the manipulator.

Recent advances in computational power have enabled synchronized motion planning, where kinematic models are commonly used for model-based or sample-based approaches

M. Dai is with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125. This work was done while she was a research intern at Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139. (email: mdai@caltech.edu)

Z. Lu is with the School of Aeronautics and Astronautics, Purdue University, IN 47907. (email: lu846@purdue.edu)

N. Li is with Harvard School of Engineering and Applied Sciences, Allston, MA 02134, and MERL, Cambridge, MA 02139. This work was done while she was a visiting researcher at MERL. (email: nali@seas.harvard.edu)

Y. Wang is with MERL. (email: yebinwang@ieee.org)

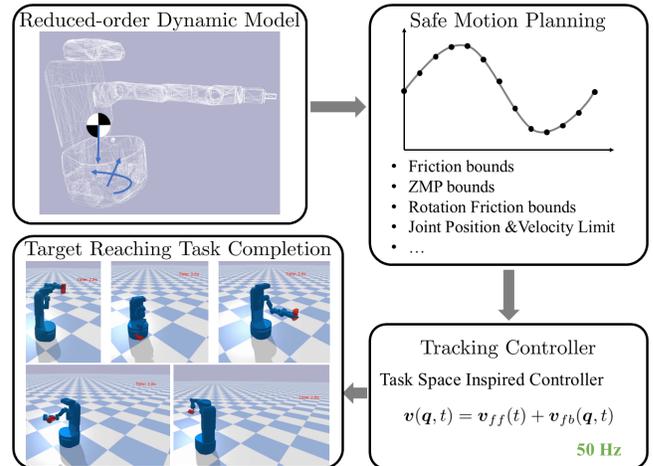


Fig. 1: Overview of the proposed reduced-order dynamics model and control framework designed for target end-effector positioning tasks, illustrating the integration of stability and maneuverability considerations.

to address end-effector tracking problems while avoiding obstacles [5]–[8]. Although synchronized kinematic planning has faster task execution time than sequential approaches, it still imposes conservative velocity limits to prevent tip-overs due to the absence of dynamic models that explicitly capture the dynamic coupling between the base and the manipulator.

However, efficiency is crucial for real-world industry applications. Traditional methods to counteract tip-over risk, such as adding counterweights or limiting operational speed as mentioned earlier, can enhance stability but often compromise the system’s overall efficiency and responsiveness [9]. As a result, these strategies are less suited for modern applications where speed and adaptability are crucial.

Various alternative solutions have been proposed to assess tip-over. Initial analyses, such as system energy level with respect to the tilt axes [10] and force-angle stability metrics [11], offered foundational insights into stability. However, these studies focused mainly on static scenarios and did not account for dynamic effects arising from the base’s acceleration or deceleration. The Moment Height Stability (MHS) measure [12] offered a refinement over purely static methods by considering the height of the overturning moment, thereby capturing the effects of rotational forces. The Zero Moment Point (ZMP) concept [13], originally developed for robotic bipedal walking, was adapted for mobile manipulators to address dynamic stability concerns [14], [15]. However, precise calculations of forces and moments at each wheel-ground contact remain challenging [16], which complicates

the development of effective motion planning algorithms using MHS and ZMP criteria.

Utilizing the proposed criteria, existing tip-over avoidance solutions operate reactively [17], [18]. For instance, when tip-over risks are detected using real-time analysis tools (e.g., MHS or ZMP-based criterion), the system may respond by slowing down the base or adjusting the manipulator's position to a pre-defined safe state. Some methods utilize fuzzy logic control schemes to adjust the arm's configuration [19]. As these methods are not inherently part of the motion planning framework, they can produce behaviors that conflict with the initial plan. Consequently, task completion may become slower or less precise, as the robot repeatedly adjusts between following the planned path and responding to stability alarms. These limitations highlight the need for a more integrated solution.

In this work, we introduce a novel framework that incorporates tip-over prevention directly into the motion planning phase, enhancing both safety and maneuverability. This framework is built on a reduced-order dynamical model that leverages centroidal momentum dynamics, including the linear and angular centroidal momentum [20]. While full dynamic models consider every joint and link of the robot, centroidal momentum dynamics significantly reduces this complexity by focusing on the forces and moments acting on the center of mass (CoM) while maintaining the necessary components to ensure dynamic feasibility. Additional simplification of wheel-ground force modeling allows the embedding of the ZMP criterion into the planning phase. As shown in Fig. 1, together with a task-space-styled tracking controller, our framework realized dynamically feasible trajectories while avoiding tip-overs even during fast maneuvers or while carrying payloads, ensuring stable and efficient task execution across a variety of scenarios. The collection of the result videos is available in the accompanying video ¹.

The remainder of this paper is organized as follows. Section II reviews the kinematic and dynamic models commonly used in motion planning for mobile manipulators, highlighting their limitations. Building upon these limitations, section III introduces our proposed reduced-order dynamics model, along with the constraints necessary to ensure dynamic feasibility and prevent tip-over. Section IV formulates the motion planning problem and details the trajectory tracking controller implemented in this work. Section V presents simulation results that demonstrate the effectiveness of our approach. Finally, section VI concludes the paper and discusses directions for future work.

II. EXISTING MODELS

To contextualize our approach, we first review commonly used kinematic and dynamic models for wheeled mobile manipulators. In particular, we focus on a nonholonomic mobile base that operates under Pfaffian velocity constraints, which limit lateral movements. These constraints are characteristic of many practical wheeled mobile systems, including unicycles, differential drive vehicles, and Ackermann

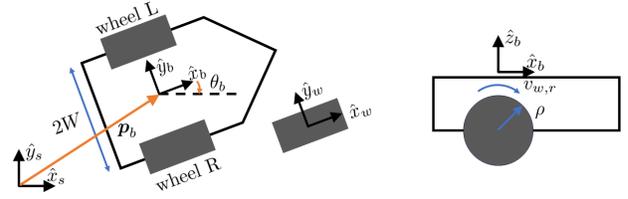


Fig. 2: Schematic representation of a differential drive mobile robot, showing both top-down and side views. Subscripts are used to define coordinate frame where $(\cdot)_s$ is world frame, $(\cdot)_w$ is wheel frame and $(\cdot)_b$ is the body frame.

steering systems. Unlike holonomic systems, which allow omnidirectional movement and are often equipped with specialized wheels, nonholonomic ones are more robust in real-world environments, such as warehouses and factories. While holonomic bases are prevalent in academic settings, where precise maneuverability is prioritized, they are susceptible to jamming and wear in industrial conditions due to complex wheel mechanisms. Although our framework can directly be applied to holonomic systems, we specifically target nonholonomic bases for broader applicability in the industry.

A. Kinematic Model

The kinematic model for a differential drive robot is based on a coordinate system centered at the robot's center of rotation. As shown in Fig. 2, the position and orientation of the mobile base on flat ground are given by generalized coordinates $\mathbf{q}_{b,3} = [\mathbf{p}_b^T \ \theta_b]^T$, where \mathbf{p}_b and θ_b represent the Cartesian coordinates and angle of the base frame relative to the world frame. This simplified 3D coordinate representation is often used because it provides an efficient and practical approach for modeling motion in planar environments, which are common in robotics applications like factory floors or warehouses. Let W be half of the car width and let ρ be the wheel radius. Under the assumption of no lateral slip and single ground contact points for each wheel, the robot's motion is constrained by a non-holonomic constraint:

$$\underbrace{\begin{bmatrix} 1 & 0 & W \cos(\theta_b) & -\rho \cos(\theta_b) & 0 \\ 0 & 1 & W \sin(\theta_b) & -\rho \sin(\theta_b) & 0 \\ 1 & 0 & -W \cos(\theta_b) & -\rho \cos(\theta_b) & 0 \end{bmatrix}}_{\Lambda_b} \begin{bmatrix} \mathbf{v}_{b,3} \\ v_{w,r} \\ v_{w,l} \end{bmatrix} = 0, \quad (1)$$

where $\mathbf{v}_{b,3}$, $v_{w,r}$, and $v_{w,l}$ represent the velocities of the base coordinate, the right wheel, and the left wheel, respectively. Rearranging this constraint, the equation of motion for the kinematic model of the mobile base is given by:

$$\frac{d}{dt} \mathbf{q}_{b,3} = \underbrace{\begin{bmatrix} \frac{\rho}{2} \cos(\theta_b) & \frac{\rho}{2} \cos(\theta_b) \\ \frac{\rho}{2} \sin(\theta_b) & \frac{\rho}{2} \sin(\theta_b) \\ \frac{\rho}{2W} & -\frac{\rho}{2W} \end{bmatrix}}_{J_{w,3}} \begin{bmatrix} v_{w,r} \\ v_{w,l} \end{bmatrix} \quad (2)$$

where $J_{w,3}$ represents the Jacobian that maps wheel velocity to the 3-dimensional base velocity. As a result, the kinematic model of a mobile manipulator with a non-holonomic base can be summarized as:

$$\frac{d}{dt} \begin{bmatrix} \mathbf{q}_{b,3} \\ \mathbf{q}_a \end{bmatrix} = \begin{bmatrix} J_{w,3}(\mathbf{q}_{b,3}) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{v}_w \\ \mathbf{v}_a \end{bmatrix}, \quad (3)$$

¹<https://vimeo.com/1027881425?share=copy>

where $\mathbf{q}_a \in Q^{n_{\text{arm}}}$ and $\mathbf{v}_a \in \mathbb{R}^{n_{\text{arm}}}$ represent the position and velocity of the articulated manipulator arm aside from base coordinates. Additionally, $\mathbf{v}_w = [v_{w,r} \ v_{w,l}]^T$ represents wheel velocities.

B. Dynamic Model

A dynamic model is required to account for dynamic mass inertia properties, which can be derived using rigid body dynamics. Most work in mobile manipulator literature [21]–[23] use the Lagrange dynamic approach or Newton-Euler approach. With added dimension due to the additional manipulator, other rigid-body dynamics algorithms including composite rigid-body algorithm [24] can be efficiently applied. The final equation of motion is given as:

$$D(\mathbf{q})\ddot{\mathbf{q}} + H(\mathbf{q}, \dot{\mathbf{q}}) = B\boldsymbol{\tau} + \Lambda^T \boldsymbol{\lambda}, \quad (4)$$

$$\Lambda \ddot{\mathbf{q}} + \dot{\Lambda} \dot{\mathbf{q}} = 0, \quad (5)$$

where $\mathbf{q} = [\mathbf{q}_{b,3}^T \ \mathbf{q}_w^T \ \mathbf{q}_a^T]^T$ and $\mathbf{q}_w \in \mathbb{R}^2$ is the generalized coordinate for wheels. D , H , and B are the inertia matrix, the collection of centrifugal, Coriolis and gravitational forces, and the actuation matrix. The input torque is denoted by $\boldsymbol{\tau}$ and $\boldsymbol{\lambda}$ represents the Lagrange multiplier that corresponds to the non-holonomic constraints where

$$\Lambda = [\Lambda_b \ 0_{3 \times n_{\text{arm}}}] .$$

The use of a 3-dimensional base coordinate in dynamic models for mobile manipulators, like in kinematic models, stems from its ability to capture the primary motion characteristics needed for most planar movement tasks while maintaining computational simplicity. Since a 3D base model does not account for forces and torques along the vertical axis or rotations around the base's pitch and roll axes, it cannot fully capture dynamics that are critical for tip-over prevention. In high-speed or high-load conditions, omitting vertical dynamics can lead to inaccuracies in predicting stability. As a result, while existing models handle basic motion requirements, they omit stability-critical factors and are thus insufficient for preventing the tip-over, which motivates our development of a new model tailored to these considerations.

III. REDUCED-ORDER DYNAMICS MODEL

Given the limitations of existing models discussed in the earlier section, we propose a novel reduced-order model that effectively captures the essential dynamics of mobile manipulators while maintaining computational efficiency. Our model operates on the following state variables:

$$\mathbf{x} = \begin{bmatrix} \mathbf{h}_{\text{com}} \\ \mathbf{q}_b \\ \mathbf{q}_w \\ \mathbf{q}_a \end{bmatrix} \in \mathbb{R}^{14+n_{\text{arm}}} \quad (6)$$

where $\mathbf{h}_{\text{com}} \in \mathbb{R}^6$ is the centroidal momentum, $\mathbf{q}_b \in \text{SE}(3)$ represents the 6-dimensional configuration of the base, $\mathbf{q}_w \in \mathbb{R}^2$ represents the configuration of the actuated wheels, and \mathbf{q}_a denotes the n_{arm} dimensional configuration of the manipulator. This setup allows our model to represent the

dynamics associated with roll, pitch, and vertical motion, essential for stability analysis. By abstracting individual joint and link forces into a consolidated representation of the system's stability-related dynamics, our model effectively filters out less impactful interactions compared to the model presented in section II-B while preserving accuracy in areas crucial for tip-over prevention and overall stability.

The centroidal momentum dynamics have the external wrench \mathbf{f} as its input, whereas \mathbf{q}_w and \mathbf{q}_a have their velocities as inputs, respectively. Thus, the control inputs of the reduced order model are given by:

$$\mathbf{u} = \begin{bmatrix} \mathbf{f} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{v}_w \\ \mathbf{v}_a \end{bmatrix} \in \mathbb{R}^{8+n_{\text{arm}}}, \quad (7)$$

where \mathbf{v} denotes the actuated joint velocity. It is important to note that these inputs cannot be controlled independently; further details will be provided in the dynamics section. Note that the wrench \mathbf{f} considered here is a lumped wrench vector at the base frame mapped to 2D ground:

$$\mathbf{f} = [F_x \ F_y \ F_z \ M_x \ M_y \ M_z]^T. \quad (8)$$

Ideally, a mobile manipulator experiences contact wrenches at all wheels that are in contact with the ground. However, the wheel-ground contact wrench, which encompasses normal forces, frictional forces and moments, and the resulting slip dynamics, is greatly influenced by factors such as slip angle, tire properties, surface material, and contact mechanics [25]. We simplify the wheel-ground contact by treating the mobile base as a rigid body in contact with the ground. While modeling individual contact wrenches at each wheel could theoretically offer added safety by capturing more granular wheel-ground interactions, it also introduces considerable computational complexity. Since our main safety consideration is tip-over prevention, we focus on ensuring that the ZMP remains within the support polygon on the ground, a critical criterion for maintaining stability. As a result, the reduction from full dynamics to centroidal dynamics and the treatment of a single lumped wrench provide a practical balance that preserves critical safety aspects while significantly reducing computational demands.

A. Dynamics

The equation of motion for our reduced-order model, incorporating non-holonomic constraints, is formulated as:

$$\frac{d}{dt} \mathbf{x} = \frac{d}{dt} \begin{bmatrix} \mathbf{h}_{\text{com}} \\ \mathbf{q}_b \\ \mathbf{q}_w \\ \mathbf{q}_a \end{bmatrix} = \underbrace{\begin{bmatrix} J_f(\mathbf{q}) & 0 & 0 \\ 0 & J_w(\mathbf{q}_b) & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}}_{A(\mathbf{x})} \mathbf{u} + \underbrace{\begin{bmatrix} \mathbf{g} \\ 0 \\ 0 \\ 0 \end{bmatrix}}_b \quad (9)$$

where $J_f(\mathbf{q}) \in \mathbb{R}^{6 \times 6}$ is the Jacobian relating the effect of the wrench to the derivative of the centroidal momentum, $J_w(\mathbf{q}_b) \in \mathbb{R}^{6 \times 2}$ represents the non-holonomic constraints imposed by the wheel kinematics for 6-dimensional base definition, and $\mathbf{g} = [0 \ 0 \ -g \ 0 \ 0 \ 0]^T$ is the gravitational force vector. Note that $J_w(\mathbf{q}_b)$ includes zero entries

for the roll, pitch, and vertical (z-axis) degrees of freedom, extending the 3-dimensional wheel Jacobian $J_{w,3}$ to a 6-dimensional configuration space. This extended formulation also assumes no wheel slippage and that tip-over is avoided as models in section II. However, these assumptions are enforced through safety constraints, ensuring that the model's dynamics remain valid under these operating conditions.

The Jacobian $J_f(\mathbf{q})$ for centroidal dynamics is defined as:

$$J_f(\mathbf{q}) = \begin{bmatrix} I & 0 \\ [\mathbf{p}_f(\mathbf{q}) - \mathbf{p}_{\text{com}}(\mathbf{q})]_{\times} & I \end{bmatrix}, \quad (10)$$

where $\mathbf{p}_f(\mathbf{q})$ and $\mathbf{p}_{\text{com}}(\mathbf{q})$ represent the location of contact force and CoM, respectively. The notation $[\cdot]_{\times}$ denotes the transformation to a skew-symmetric matrix that models the cross product between the position and force vector.

An equality constraint is required to relate the robot's joint motions to its centroidal momentum [26], expressed as $\mathbf{h}_{\text{com}} = M(\mathbf{q})\dot{\mathbf{q}}$. The $M(\mathbf{q})$ matrix can be computed using centroidal composite rigid-body algorithm [24] given the robot configuration. Since the velocity of the robot's base is underactuated and determined by the motion of its wheels under non-holonomic constraints, the centroidal momentum constraint can be formulated as follows:

$$\mathbf{h}_{\text{com}} = M(\mathbf{q})\dot{\mathbf{q}} \quad (11)$$

$$= M_b(\mathbf{q})\dot{\mathbf{q}}_b + M_w(\mathbf{q})\dot{\mathbf{q}}_w + M_a(\mathbf{q})\dot{\mathbf{q}}_a \quad (12)$$

$$= M_b(\mathbf{q})J_w(\mathbf{q}_b)\mathbf{v}_w + M_w(\mathbf{q})\mathbf{v}_w + M_a(\mathbf{q})\mathbf{v}_a \quad (13)$$

$$=: M_{h\text{com}}(\mathbf{q}) \begin{bmatrix} \mathbf{v}_w \\ \mathbf{v}_a \end{bmatrix}, \quad (14)$$

where $M_{h\text{com}}(\mathbf{q})$ provides a compact representation of the centroidal momentum in terms of the reduced-order states \mathbf{x} and inputs \mathbf{u} . The centroidal momentum dynamics are subject to external forces and wrenches acting on the system, with the momentum itself derived from the robot's joint positions and velocities. This relationship ensures that the model captures the impact of forces and moments at the center of mass, allowing for dynamically feasible trajectory planning. The complete equation of motion, incorporating the dynamics and non-holonomic constraints, is given by:

$$\begin{cases} \frac{d}{dt} \begin{bmatrix} \mathbf{h}_{\text{com}} \\ \mathbf{q} \end{bmatrix} = A(\mathbf{x})\mathbf{u} + \mathbf{b} \\ \mathbf{h}_{\text{com}} - M_{h\text{com}}(\mathbf{q})\mathbf{v} = 0 \end{cases} \quad (\text{Dynamics})$$

This coupled representation for dynamics guarantees that the forces and moments applied through the control inputs are feasible through the robot's physical motion.

B. Safety Constraints

To ensure the safe and stable operation of the mobile manipulator, we impose the following safety constraints: ZMP constraints to prevent tip-over, friction constraints to avoid slipping, and joint limits to ensure physical feasibility.

1) *ZMP Constraints*: The ZMP, denoted by $\mathbf{p}_{\text{zmp}} = [r_x \ r_y \ 0]^T$, represents the point on the ground plane where the resultant moment about horizontal axes is zero. To prevent tip-over, the ZMP must remain within the support polygon defined by the wheel positions [13]. For this work, we assume a rectangular support polygon with ZMP constraints expressed as:

$$-\alpha l_a \leq r_x \leq \alpha l_b \quad (15)$$

$$-\alpha w_a \leq r_y \leq \alpha w_b \quad (16)$$

where l_a , l_b , w_a , w_b represent the distances to the edges of the support polygon from base frame location, and α is a tuning factor that adjusts the conservativeness of the constraints, allowing for compensation for dynamic uncertainty. The location of the ZMP is determined by identifying the point at which applying force results in zero moments about the horizontal axes. This requirement is defined by the following condition:

$$M_{\text{zmp}}^h = \begin{bmatrix} M_x \\ M_y \end{bmatrix} + \left(\begin{bmatrix} r_x \\ r_y \\ 0 \end{bmatrix} \times \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \right)_{x,y} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (17)$$

Solving this equation yields $r_x = \frac{M_y}{F_z}$ and $r_y = -\frac{M_x}{F_z}$. Substituting these expressions into the ZMP bounds, we can reformulate the constraints in terms of the model inputs \mathbf{u} :

$$\begin{cases} -\alpha F_z l_a \leq M_y \leq \alpha F_z l_b, \\ -\alpha F_z w_a \leq M_x \leq \alpha F_z w_b. \end{cases} \quad (\text{C-ZMP})$$

2) *Friction Constraints*: To ensure stable contact between the wheels and the ground, we enforce a friction cone constraint that limits the allowable lateral forces:

$$F_x^2 + F_y^2 \leq \mu^2 F_z^2, \quad (18)$$

where μ is the friction coefficient. This constraint ensures that the lateral forces do not exceed what the ground friction can support, thereby preventing slipping. For computational simplicity, the friction cone can be approximated by a friction pyramid with linear constraints:

$$\begin{cases} |F_x| \leq \frac{\mu}{\sqrt{2}} F_z, \\ |F_y| \leq \frac{\mu}{\sqrt{2}} F_z. \end{cases} \quad (\text{C-friction})$$

This approximation provides a conservative but computationally efficient alternative to the friction cone, allowing us to represent the friction limits using linear constraints.

In addition, we impose a constraint on the yaw moment to prevent excessive turning forces that could cause slippage [27]. Let ν denote the rotational friction coefficient, the yaw moment constraint is:

$$|M_z| \leq \nu F_z. \quad (\text{C-rotation})$$

Together, these constraints ensure that the applied forces and moments remain within the frictional stability limits, preventing lateral and rotational slipping during maneuvers.

3) *Joint Limits and Joint Velocity Limits*: To ensure that the planned trajectory respects the physical limitations of the robot's joints, we impose inequality constraints on both joint positions and velocities:

$$\begin{cases} \underline{\mathbf{q}}_a \leq \mathbf{q}_a \leq \overline{\mathbf{q}}_a, \\ \underline{\mathbf{v}}_a \leq \mathbf{v}_a \leq \overline{\mathbf{v}}_a. \end{cases} \quad (\text{C-limit})$$

Here, $\underline{\mathbf{q}}_a$ and $\overline{\mathbf{q}}_a$ represent the lower and upper bounds on the joint positions, while $\underline{\mathbf{v}}_a$ and $\overline{\mathbf{v}}_a$ define the velocity limits. These constraints ensure that joint movements remain within the robot's mechanical limits, preventing damage and ensuring safe operation.

IV. MOTION PLANNING AND TRACKING

This section introduces the dynamically feasible motion planning framework, leveraging the proposed reduced-order model and safety constraints. The planning focuses on target-reaching tasks, formulated as an offline nonlinear program (NLP) to ensure safe and effective movement towards predefined goals. Additionally, a feedback controller is presented for accurate trajectory tracking by the mobile manipulator.

A. Planning

To achieve dynamically feasible motion planning, we formulate a NLP that enables the generation of stable trajectories by solving for the state and control inputs across a preview horizon. We divide this planning horizon of t seconds into N intervals with $dt = \frac{t}{N}$. At each time instance, we denote the state and input by $\mathbf{x}_i \forall i \in [0, \dots, N]$ and $\mathbf{u}_i \forall i \in [0, \dots, N-1]$.

The objective function for the NLP formulation consists of two main terms: (1) an end-effector tracking term J_{ee} , which drives the mobile manipulator towards the target pose, and (2) a regularization term J_u that minimizes control efforts and promote smooth and efficient movements.

The end-effector tracking cost $J_{ee}(\mathbf{x}_i)$ is formulated to minimize the difference between the current and target poses of the end-effector in the $SE(3)$ space:

$$J_{ee}(\mathbf{x}_i) = \|\mathbf{e}(\mathbf{q}_i)\|_{W_{ee}}, \quad (19)$$

where \mathbf{q}_i is extracted from \mathbf{x}_i at each collocation node, and the notation $\|X\|_W^2 \triangleq X^T W X$ defines a quadratic cost such that W is a positive-definite diagonal weight matrix. A terminal cost $J_{ee,N}(\mathbf{x}_N)$ is defined with weight $W_{ee,N}$ to ensure accurate tracking at the final time step. The pose error $\mathbf{e}(\mathbf{q}_i)$ is calculated using twists from Lie algebra [28], which enables an efficient representation of the difference between the current and target poses in the $SE(3)$ space:

$$\mathbf{e}(\mathbf{q}) := \mathbf{T}_{s^*} \ominus \mathbf{T}_{sb}(\mathbf{q}) = \log_6(\mathbf{T}_{bs}(\mathbf{q})\mathbf{T}_{s^*}) = \log_6(\mathbf{T}_{b^*}(\mathbf{q}))$$

where \mathbf{T}_{s^*} represents the target end-effector pose in the world frame, $\mathbf{T}_{sb}(\mathbf{q})$ represents the pose of end-effector derived from forward dynamics of the robot in the world frame, and $\log_6 : SE(3) \rightarrow se(3)$ maps poses to twists. This twist-based error representation avoids issues like angle wraparound and ensures rotational errors are computed continuously without abrupt changes.

The input regularization cost on velocities and wrench is given as:

$$J_u(\mathbf{u}_i) = \|\mathbf{f}_i\|_{W_f} + \|\mathbf{v}_{w,i}\|_{W_w} + \|\mathbf{v}_{a,i}\|_{W_a} \quad (20)$$

The complete optimization problem is formulated as:

$$\begin{aligned} \min_{\mathbf{x}_0 \dots \mathbf{x}_N, \mathbf{u}_0 \dots \mathbf{u}_{N-1}} \quad & \sum_{i=1}^{N-1} (J_{ee}(\mathbf{x}_i) + J_u(\mathbf{u}_i)) + J_{ee,N}(\mathbf{x}_N) \\ \text{s.t.} \quad & (\text{Dynamics Constraints}), \\ & (\text{Safety Constraints}). \end{aligned} \quad (\text{NLP})$$

Here, the Dynamics Constraints are enforced as equality constraints where the forward Euler scheme is used for collocation with EoM (Dynamics). The Safety Constraints including (C-ZMP), (C-friction), (C-rotation) and (C-limit) are collectively represented as inequality constraints.

B. Tracking

To address modeling errors and integration inaccuracies from the collocation framework, we implement a real-time tracking controller that adjusts wheel and arm joint velocities. This tracking controller can be adapted for torque control if such functionality is supported by the hardware. The tracking controller is composed of two components, i.e.

$$\mathbf{v}(\mathbf{q}, t) = \mathbf{v}_{ff}(t) + \mathbf{v}_{fb}(\mathbf{q}, t). \quad (21)$$

The feedforward component is obtained directly from the solution of (??):

$$\mathbf{v}_{ff}(t) = \mathbf{v}_{\text{NLP}}^*(t) \quad (22)$$

The feedback component is designed to correct deviations from the planned trajectory. The tracking error $\mathbf{e}(\mathbf{q}, t)$ with respect to the reference trajectory is defined as:

$$\mathbf{e}(\mathbf{q}, t) := \mathbf{T}_{s^*}(t) \ominus \mathbf{T}_{ob}(\mathbf{q}),$$

where $\mathbf{T}_{s^*}(t)$ represents the target end-effector frame from the NLP solution over time. The feedback control law, inspired by the task-space Proportional Integral (PI) controller, is given as:

$$\mathbf{v}_{fb} = \tilde{\mathcal{J}}^\dagger \left(-K_p \mathbf{e}(\mathbf{q}, t) - K_i \int_0^t \mathbf{e}(\mathbf{q}, \tau) d\tau \right), \quad (23)$$

where $\tilde{\mathcal{J}}$ is the modified task Jacobian. Since the feedforward component already compensates for planned trajectory dynamics, $\dot{\mathbf{e}}(\mathbf{q}, t)$ is not included in the feedback controller. The derivation of the nominal task Jacobian is given by:

$$J_{\text{task}} := \frac{\partial \mathbf{e}}{\partial \mathbf{q}}(\mathbf{q}) = -J \log_6(\mathbf{T}_{s^*}) {}_b J_{ob}(\mathbf{q}) \quad (24)$$

J_{task} maps joint velocity to task space velocity; however, since not all joints are actuated, we further map the Jacobian to the actuated joints to obtain the modified task Jacobian:

$$\begin{aligned} \tilde{\mathcal{J}} &= J_{\text{task}} J_{\text{map}} \\ &= \begin{bmatrix} J_{\text{task}}^b & J_{\text{task}}^w & J_{\text{task}}^a \end{bmatrix} \begin{bmatrix} J_{bw}(\mathbf{q}_b) & 0 \\ I & 0 \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} J_{\text{task}}^b J_w(\mathbf{q}_b) + J_{\text{task}}^w & J_{\text{task}}^a \end{bmatrix} \in \mathbb{R}^{6 \times (n_{\text{wheel}} + n_{\text{arm}})} \end{aligned}$$

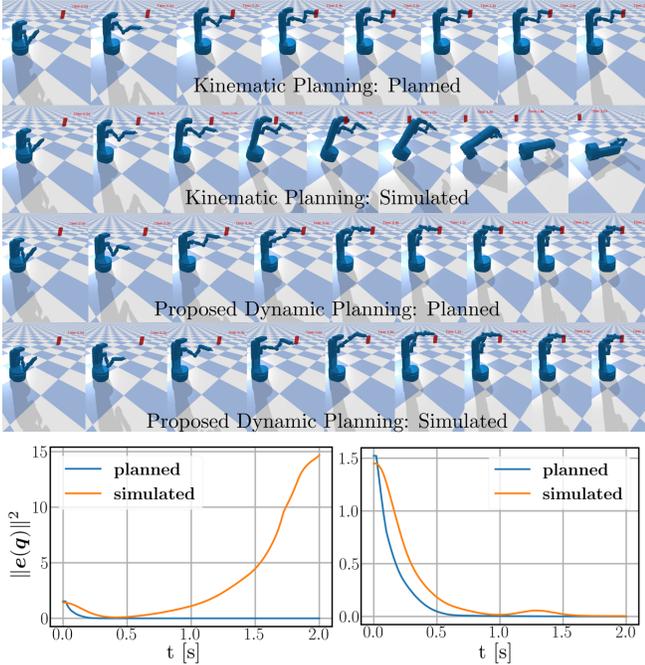


Fig. 3: Comparison of kinematic vs. proposed dynamic planning trajectories in simulation. Kinematic planning, lacking stability constraints, leads to instability and falls, while the dynamic model with stability constraints achieves the target successfully. The bottom panels display squared pose errors for kinematic (left) and dynamic (right) planning, highlighting reduced error and improved stability in the proposed dynamic planning.

The mapping to actuated joints restricts the task Jacobian to the DoF that can be directly controlled, ensuring compatibility with the physical limitations of the mobile manipulator. Together, the planning and tracking components ensure that the mobile manipulator can reach its target stably and efficiently, compensating for modeling errors and ensuring robust execution of the planned trajectory.

V. RESULTS

The proposed framework, as illustrated in Fig. 1, was tested using a Fetch robot [29], a differential drive mobile manipulator. The Fetch robot, weighing approximately 121 kg, has two drive wheels and four supporting caster wheels. The arm of the robot provides eight DoF, excluding the two DoF of the gripper’s claw. For the study, a 12.1 kg payload was attached to the end-effector to simulate a load-carrying task, which increases the system’s dynamic complexity.

The simulations were conducted in a custom-built environment using the PyBullet physics engine [30]. The velocity control loop operates at a frequency of 50 Hz to capture the responsiveness of the robot’s motion. We leverage the Pinocchio library [24] interfaced with CasADi [31] with symbolic computation to model and solve complex dynamics for our robotic system. The final optimal control problem (??) is solved using IPOPT [32] embedded in CasADi.

Control parameters for all end-effector targets were set as follows: $W_a = 0.01I$, $W_w = 0.001I$, $W_f = 0.001I$, $W_{ee} = 300I$, $W_{ee,N} = 6000I$. The safety constraints were configured with $\mu = 0.8$, $\nu = 0.05$, $l_a = l_b = 0.18\text{m}$, and

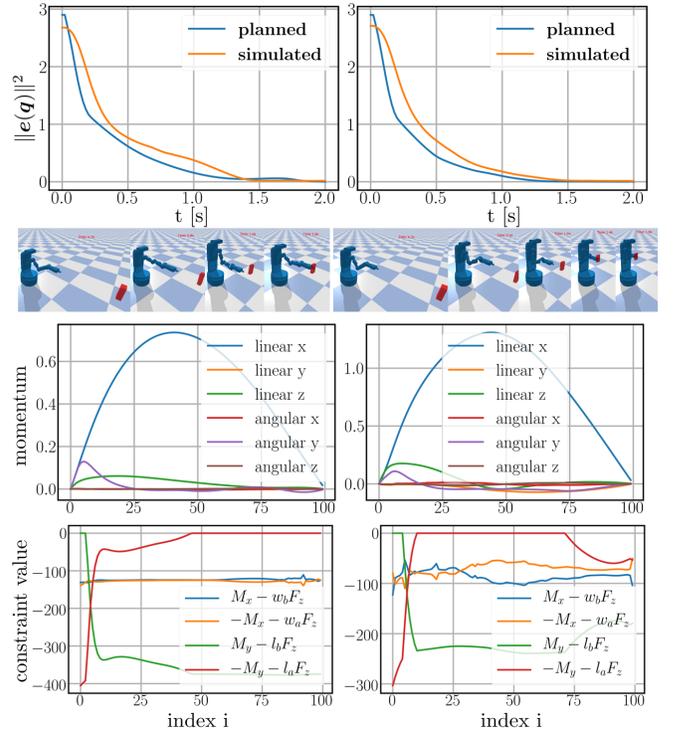


Fig. 4: Impact of ZMP bound parameter α on the trajectory. Left: A relaxed bound ($\alpha = 0.8$) allows extended arm positioning with lower base velocity, maintaining stability. Right: A conservative bound ($\alpha = 0.5$) necessitates a more compact arm posture but allows for higher base velocity to achieve a similar target reach, demonstrating the trade-off between arm extension and base adjustments under varying stability constraints.

$w_a = w_b = 0.12\text{m}$. The planning horizon was 2 seconds, divided into $N = 100$ intervals. The ZMP parameter α is set to 0.8 unless specified otherwise. Tracking controller is implemented with $K_p = 1I$ and $K_i = 0.05I$.

In comparing traditional kinematic planning with the proposed dynamic planning approach, we structured the kinematic planning model similarly to (??), replacing the dynamics equation with the kinematic model from Section II-A and using only the joint position and velocity limit (C-limit) for safety constraints. Parameters for W_a , W_w , W_{ee} , and $W_{ee,N}$ remained consistent across both approaches.

As shown in Figure 3, the kinematic planning approach, lacking safety constraints, caused aggressive movements that led to a forward fall during the simulation. In contrast, our dynamic model incorporated ZMP-based safety constraints and accounted for payload effects, resulting in a stable motion plan that successfully reached the target. The plot of pose error reveals that the kinematic simulation suffered high error due to instability, whereas the dynamic planning approach minimized this error effectively. Additional comparisons for different target poses are available in the supplementary video linked earlier.

Figure 4 illustrates the impact of different ZMP bound parameters on trajectory characteristics. On the left, the trajectory is generated with a relatively relaxed ZMP bound of $\alpha = 0.8$, enabling the arm joints to extend forward as the

manipulator reaches the target. This configuration remains stable, as the low base acceleration keeps the ZMP within safe limits. In contrast, a more conservative ZMP bound parameter $\alpha = 0.5$ constrains the arm joints to a more inward position. Although this configuration restricts the arm's range, the base compensates with a higher velocity, allowing the manipulator to maintain a similar overall target-reaching time. This results in generally higher linear and angular momentum on the right, highlighting the trade-off between arm extension and base velocity to meet stability requirements under various constraints. In general, if the user is confident in the dynamics model of the mobile manipulator, it is preferable to set α closer to 1.

VI. CONCLUSION AND FUTURE WORK

In this work, we proposed a novel reduced-order model for dynamically feasible motion planning. The problem is posed as an offline trajectory optimization problem and a task-space-styled tracking controller is implemented during execution. Since the reduced order model simplifies the full dynamics significantly, the planning problem can likely be framed as an online MPC-styled recursive planner when implemented properly. Additionally, this model is well suited for robust control, especially considering the uncertain load on the end effector. These will be addressed in future work.

REFERENCES

- [1] N. Ghodsian, K. Benfriha, A. Olabi, V. Gopinath, and A. Arnou, "Mobile Manipulators in Industry 4.0: A Review of Developments for Industrial Applications," *Sensors*, vol. 23, no. 19, p. 8026, Jan. 2023.
- [2] R. Automation, "RB-KAIROS+: The new Mobile Manipulator," <https://www.youtube.com/watch?v=bqj...XyuG5M>.
- [3] J. Xu, Y. Domae, T. Ueshiba, W. Wan, and K. Harada, "Planning a sequence of base positions for a mobile manipulator to perform multiple pick-and-place tasks," *arXiv preprint arXiv:2010.00779*, 2020.
- [4] K. M. Lynch and F. C. Park, *Modern robotics: mechanics, planning, and control*. Cambridge, UK: Cambridge University Press, 2017, oCLC: ocn983881868.
- [5] J. Pankert and M. Hutter, "Perceptive Model Predictive Control for Continuous Mobile Manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6177–6184, Oct. 2020.
- [6] M. Spahn, B. Brito, and J. Alonso-Mora, "Coupled Mobile Manipulation via Trajectory Optimization with Free Space Decomposition," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China: IEEE, May 2021, pp. 12 759–12 765.
- [7] A. Heins and A. P. Schoellig, "Keep it Upright: Model Predictive Control for Nonprehensile Object Transportation with Obstacle Avoidance on a Mobile Manipulator," *IEEE Robotics and Automation Letters*, vol. 8, no. 12, pp. 7986–7993, Dec. 2023, arXiv:2305.17484 [cs].
- [8] S. Thakar, P. Rajendran, A. Kabir, and S. Gupta, "Manipulator Motion Planning for Part Pickup and Transport Operations From a Moving Base," *IEEE Transactions on Automation Science and Engineering*, vol. PP, pp. 1–16, Sep. 2020.
- [9] T. Song, F. J. Xi, S. Guo, and Y. Lin, "Optimization of a Mobile Platform for a Wheeled Manipulator," *Journal of Mechanisms and Robotics*, vol. 8, no. 6, p. 061007, Dec. 2016.
- [10] A. Ghasempoor and N. Sepelri, "A measure of machine stability for moving base manipulators," in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3, May 1995, pp. 2249–2254 vol.3, iSSN: 1050-4729.
- [11] E. Papadopoulos and D. Rey, "A new measure of tipover stability margin for mobile manipulators," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 4, Apr. 1996, pp. 3111–3116 vol.4, iSSN: 1050-4729.
- [12] S. Moosavian and K. Alipour, "On The Dynamic Tip-over Stability of Wheeled Mobile Manipulators," *International Journal of Robotics and Automation*, vol. 22, no. 4, 2007.
- [13] M. Vukobratović and B. Borovac, "Zero-moment point — thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 01, no. 01, pp. 157–173, Mar. 2004.
- [14] S. Sugano, Q. Huang, and I. Kato, "Stability criteria in controlling mobile robotic systems," in *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '93)*, vol. 2, Jul. 1993, pp. 832–838 vol.2.
- [15] Q. Huang, S. Sugano, and I. Kato, "Stability control for a mobile manipulator using a potential method," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '94)*, vol. 2, Sep. 1994, pp. 839–846 vol.2.
- [16] K. Iagnemma and S. Dubowsky, *Mobile robots in rough terrain: Estimation, motion planning, and control with application to planetary rovers*. Springer Science & Business Media, 2004, vol. 12.
- [17] A. Toledo Fuentes, F. Kempf, M. Kipfmüller, T. Bergmann, and M. J. Prieto, "Tip-Over Detection and Avoidance Algorithms as Stabilization Strategy for Small-Footprint and Lightweight Mobile Manipulators," *Machines*, vol. 11, no. 1, p. 44, Dec. 2022.
- [18] X. Ding, Y. Liu, J. Hou, and Q. Ma, "Online Dynamic Tip-Over Avoidance for a Wheeled Mobile Manipulator With an Improved Tip-Over Moment Stability Criterion," *IEEE Access*, vol. 7, pp. 67 632–67 645, 2019, conference Name: IEEE Access.
- [19] Y. Li and Y. Liu, "Real-Time Tip-Over Prevention and Path Following Control for Redundant Nonholonomic Mobile Modular Manipulators via Fuzzy and Neural-Fuzzy Approaches," *Journal of Dynamic Systems, Measurement, and Control*, vol. 128, no. 4, pp. 753–764, Dec. 2006.
- [20] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, vol. 35, no. 2-3, pp. 161–176, Oct. 2013.
- [21] R. D. Ahmad Abu Hatab, "Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework," *Advances in Robotics & Automation*, vol. 02, no. 02, 2013.
- [22] C.-C. Tsai, M.-B. Cheng, and S.-C. Lin, "Dynamic Modeling and Tracking Control of a Nonholonomic Wheeled Mobile Manipulator with Dual Arms," *Journal of Intelligent and Robotic Systems*, vol. 47, no. 4, pp. 317–340, Nov. 2006.
- [23] Z. Zhou, X. Yang, H. Wang, and X. Zhang, "Coupled dynamic modeling and experimental validation of a collaborative industrial mobile manipulator with human-robot interaction," *Mechanism and Machine Theory*, vol. 176, p. 105025, 2022.
- [24] 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)*, Jan. 2019, pp. 614–619, iSSN: 2474-2325.
- [25] J. Y. Wong, *Theory of ground vehicles*. John Wiley & Sons, 2022.
- [26] D. Orin and A. Goswami, "Centroidal Momentum Matrix of a humanoid robot: Structure and properties," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Nice: IEEE, Sep. 2008, pp. 653–659.
- [27] E. Rimon and J. Burdick, *The Mechanics of Robot Grasping*. Cambridge University Press, 2019.
- [28] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. CRC Press, Dec. 2017.
- [29] M. Wise, M. Ferguson, D. King, E. Diehr, and D. Dymesich, "Fetch and freight: Standard platforms for service robot applications," in *Workshop on autonomous mobile service robots*, 2016, pp. 1–6.
- [30] B. Ellenberger, "Pybullet gymperium," <https://github.com/benelot/pybullet-gym>, 2018–2019.
- [31] J. A. E. 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, no. 1, pp. 1–36, 2019.
- [32] 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.