

Whole-Body Model Predictive Control for Mobile Manipulation with Task Priority Transition*

Yushi Wang¹, Ruoqu Chen¹, Mingguo Zhao¹

Abstract—Mobile manipulators enable a wide range of operations with mobility and advanced manipulation capabilities. Despite their potential, existing approaches typically treat the mobile base and the manipulator separately, thereby limiting the optimality of the system for composite whole-body behaviors. In this work, we present a Whole-Body Model Predictive Control framework for mobile manipulation involving tasks with varying timelines. We integrate task priorities across both task and time dimensions, bringing inherent transition ability with enhanced performance. Our approach improves the trajectory tracking performance by up to 36% in terms of manipulability and reduces the maximum velocity during task priority transitions by 53% compared to the existing approach while maintaining a low computational cost of 4.3 ms, allowing for high reactivity in real-world applications. We demonstrate its effectiveness through a door-opening and traversing behavior, showcasing the first successful implementation of a non-holonomic mobile manipulator in such a scenario. See <https://yushiw.github.io/wbmvc> for supplemental materials.

I. INTRODUCTION

Implementing robots in real-world environments to perform everyday work remains a critical area of research. Mobile manipulators, which combine manipulation capabilities with the mobility offered by a mobile base, show great potential across various applications. However, the increased degrees of freedom result in greater kinematic complexity for control. Consequently, most research on mobile manipulation employs separate controllers for the base and manipulator, typically focusing on simple scenarios like pick-and-place [1], [2]. These approaches sacrifice optimality and necessitate manual coordination between both systems. This limitation becomes especially problematic when executing behaviors requiring tightly coordinated whole-body motions.

Enhancing coordination between locomotion and manipulation could significantly expand the capabilities of mobile manipulators, enabling them to perform more intricate behaviors such as opening cabinets [3] or doors [4]. These real-world scenarios are typically long-horizon and involve multiple tasks with varying timelines. For example, in a door-opening scenario, after the end-effector opens the door, a base task is introduced to prevent it from closing. Whole-body control (WBC) has been extensively studied for solving multi-task control problems by prioritizing tasks, yet most existing methods handle task priority at a single moment [5]. However, due to the varying timelines for different tasks, the priorities of tasks are also transitional in the time dimension.



Fig. 1: Our mobile manipulator with sideways movement locked. An additional base task is introduced to block the door while the end-effector holds the handle. Our method facilitates task priority transitions for composite behaviors.

In this paper, we propose a Whole-Body Model Predictive Control (WB MPC) framework that extends the integration of task priorities across both task and time dimensions by incorporating predictive capabilities. This approach inherently addresses the task priority transition challenge in coordinated mobile manipulation. We introduce a unified weight matrix representation to manage the added complexity of simultaneously considering tasks and time, thus achieving optimized joint configurations. With the expansion in dimensions, our controller is capable of optimizing and executing composite behaviors that involve transitional tasks.

We validate our method through real-world experiments involving a door-opening and traversing behavior—a challenging behavior that requires alternating between maintaining the door with the end-effector and the base, as shown in Fig. 1. Our approach enables seamless transitions between tasks, allowing the mobile manipulator to grasp, open, block, and pass through a self-closing door.

The key contributions of this work are:

- 1) We propose a WB MPC approach for mobile manipulators addressing the challenges of task priority transitions in coordinated behaviors. Our method outperforms the existing inverse-kinematic-based WBC approach in terms of singularity avoidance and task transitions.
- 2) To the best of our knowledge, this work represents one of the first successful implementations of a non-holonomic mobile manipulator opening and traversing through self-closing doors in real-world conditions.

*This work was supported by STI 2030—Major Projects 2021ZD0201402

¹The authors are with Department of Automation, Tsinghua University, Beijing, China mgzhao@mail.tsinghua.edu.cn

II. RELATED WORK

A. Mobile Manipulator Control

Common control methods for mobile manipulators typically begin with determining a feasible base pose, followed by the formulation of the manipulator's trajectory based on the planned movement of the base [1], [2], [6]. However, these approaches require considerable expertise to manually coordinate both systems, limiting their applicability in composite scenarios.

Some works integrate degrees of freedom associated with both the base and the manipulator into a unified structure to generate whole-body joint-space motions. In [7], trajectory optimization (TO) methods are employed to plan global joint-space trajectories for mobile manipulators. However, these methods are too complex for real-time solutions, necessitating periodic replanning to maintain reactivity to environmental changes. In contrast, inverse-kinematics (IK)-based methods provide a direct and efficient approach for translating end-effector tasks into joint-space motions [8] while concentrating on immediate objectives without considering long-term goals. Model predictive control (MPC), which is adopted in this work, strikes a balance between reactivity and global optimality to generate graceful motions by iteratively solving finite-horizon optimal control problems for mobile manipulators [9], [10], [11].

B. Multi-task Whole-body Control

Whole-body control (WBC) is widely adopted for multi-task control problems, particularly in manipulators [12] and legged robots [13]. WBC assigns priorities to various tasks and generates whole-body motions that integrate these tasks based on their respective priorities. Several methods have been developed to ensure smooth transitions between task priorities in WBC, including algorithms based on weights [14], intermediate variables [15], [16], and projection matrices [17], [18]. A review of WBC is presented in [5].

Despite these advancements, existing WBC approaches handle multi-tasking and task transitions at a discrete moment without incorporating predictive capabilities. Concurrently, most MPC research excludes multi-tasking due to the complexity associated with simultaneously managing both task and time dimensions. One potential compromise is to focus solely on the task dimension while maintaining a consistent priority over the time dimension. In [11], a Hierarchical-Task MPC framework was proposed for efficiently executing sequential manipulation tasks by solving an individual optimization problem for each task. However, it fails to manage priority transitions for composite behaviors owing to its fixed hierarchy of the end-effector and base task throughout the process. Our approach provides a unified representation of both task and time dimensions by extending the weight-based approach in WBC to achieve task priority transitions.

C. Door Opening and Traversing Task

We demonstrate a door-opening and traversing action to evaluate the effectiveness of the proposed method. Door

opening is a widely addressed topic for mobile manipulators or legged manipulators, employing either model-based [19], [20] or learning-based approaches [4], [21]. However, most studies focus solely on the task of opening doors or consider door-opening and traversing as separate phases. Only a few studies demonstrate the capability to handle self-closing doors [22], [23], which is a daily yet challenging task that necessitates simultaneous execution of multiple tasks by both the end-effector and the base. In [22], a module integration method based on deep predictive learning was introduced to generate appropriate motions for mobile manipulators. In [23], an offline bilevel planner for multiple contact locomotion was proposed to produce high-fidelity joint-space plans, enabling a legged manipulator to utilize its leg for holding the door, while the MPC is perceived as a tracker of precomputed plans instead of generating reliable solutions. In contrast, our method employs MPC for online joint-space planning and is adaptable to challenging non-holonomic bases, where prior actions constrain feasible task-space motions [24].

III. PROBLEM FORMULATION

A. Control Architecture

Our proposed architecture is presented in Fig. 2. The task-space commander specifies either a single task or multiple tasks for the mobile manipulator. These tasks are represented as desired trajectories of various components, such as the end effector and mobile base, derived from user commands or planners. The WBMPC minimizes a cost function expressed as a weighted sum of task-tracking errors over a specified time horizon. It produces feasible joint-space trajectories that incorporate constraints based on a whole-body kinematics model. The joint-space controllers subsequently compute the final torque or velocity command—depending on the interface of the deployed robot—at a higher frequency for each joint, ensuring robust and accurate execution. Additionally, a state estimator provides real-time feedback of the robot's state, utilizing data from joint sensors and IMU. The subsequent sections detail how the WBMPC enables optimal performance for mobile manipulation.

B. Whole-body MPC Formulation

The WBMPC is based on continuously solving a nonlinear optimal control problem (OCP) with horizon $N \in \mathbb{N}$ given the current state \hat{x}_0 of the system and applying the first element of the optimized trajectory to the system. We consider the following discrete-time OCP structured nonlinear program (NLP) formulation

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \sum_{k=0}^{N-1} \ell_k(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}) + \ell_N(\boldsymbol{x}_N, \boldsymbol{\theta}) \quad (1a)$$

$$\text{subject to } \boldsymbol{x}_0 = \hat{\boldsymbol{x}}_0 \quad (1b)$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) \quad k \in \{0, \dots, N-1\} \quad (1c)$$

$$\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}) \leq \mathbf{0} \quad k \in \{0, \dots, N-1\} \quad (1d)$$

$$\boldsymbol{h}_N(\boldsymbol{x}_N, \boldsymbol{\theta}) \leq \mathbf{0}, \quad (1e)$$

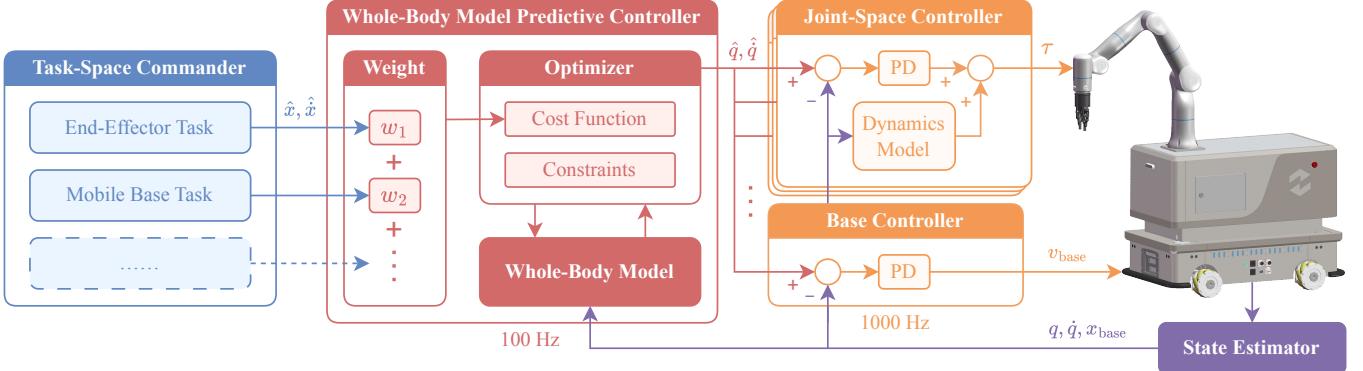


Fig. 2: The proposed mobile manipulation control architecture. The arrows indicate the data flow between modules.

where $\mathcal{X} = \{\mathbf{x}_0, \dots, \mathbf{x}_N, \mathbf{u}_0, \dots, \mathbf{u}_{N-1}\}$ is the set of decision variables composed of the state vector $\mathbf{x}_k \in \mathbb{R}^{n_x}$ and control vector $\mathbf{u}_k \in \mathbb{R}^{n_u}$ at stage k , $\hat{\mathbf{x}}_0$ is the current estimated state, θ is a parameter vector, $\ell_k(\cdot)$ and $\ell_N(\cdot)$ are respectively the path and terminal cost, $\mathbf{f}(\cdot)$ is the discrete-time dynamics, $\mathbf{h}(\cdot)$ and $\mathbf{h}_N(\cdot)$ are the path and terminal constraints.

C. System Model

We denote the state of the mobile manipulator as $\mathbf{x} = (\mathbf{x}_{\text{base}}, \mathbf{x}_{\text{arm}})$, where $\mathbf{x}_{\text{base}} = (x, y, \theta) \in \mathbb{R}^3$ is the position and orientation of the mobile base, and $\mathbf{x}_{\text{arm}} = (q_1, \dots, q_n) \in \mathbb{R}^n$ is the joint positions of the manipulator with n degrees of freedom (DoF).

The kinematic model of the mobile base can be described as

$$\dot{\mathbf{x}}_{\text{base}} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (2)$$

where v and ω are respectively the forward and turning rates of the base. The mobile base can be either omnidirectional or non-holonomic. In our system model, sideways movement is disabled to ensure compatibility with non-holonomic bases such as differential-drive or tracked robots. For omnidirectional bases, the model can simply be replaced by $\dot{\mathbf{x}}_{\text{base}} = (v_x, v_y, \omega)$. MPC incorporates future tasks when determining current actions, thereby addressing challenges posed by non-holonomic constraints, while IK-based methods operate with a myopic focus on immediate moments.

The state of the manipulator is directly determined by joint velocities through the kinematic model $\dot{\mathbf{x}}_{\text{arm}} = (\dot{q}_1, \dots, \dot{q}_n)$. In order to consistently take advantage of the whole-body DoF, we combine the joint velocities of both the base and manipulator into a unified control vector \mathbf{u}

$$\mathbf{u} = [\dot{q}_l \quad \dot{q}_r \quad \dot{q}_1 \quad \dots \quad \dot{q}_n]^\top, \quad (3)$$

where \dot{q}_l and \dot{q}_r are respectively the velocities of the left and right wheels of the base. They can be easily related to the rates of the base by

$$v = \frac{R}{L} (\dot{q}_r - \dot{q}_l), \quad \omega = \frac{R}{2} (\dot{q}_r + \dot{q}_l), \quad (4)$$

where R is the radius of the wheels, and L is the distance between them.

The system model is discretized using Runge-Kutta integration to obtain the equality constraints (1c) imposed at every stage k , ensuring that the state trajectory is consistent with the control inputs.

D. Cost Function

The cost function seeks to minimize the task-space tracking error and maximize the manipulability of the robot. In addition, a control input regularization term is included to penalize the control effort. The path cost in (1a) is written as

$$\ell_k(\mathbf{x}_k, \mathbf{u}_k, \theta) = \ell_{t,k}(\mathbf{x}_k, \theta) + \ell_m(\mathbf{x}_k) + \|\mathbf{u}_k\|_R^2, \quad (5)$$

where $\ell_{t,k}(\cdot)$ is task-space tracking error at stage k , $\ell_m(\cdot)$ is manipulability term, R is a positive definite matrix.

1) *Task-space Tracking:* The tracking cost $\ell_{t,k}$ is defined as a weighted sum of multiple task-space tracking errors

$$\ell_{t,k} = \sum_i w_{i,k} \|e_{i,k}\|_{Q_i}^2, \quad (6)$$

where $w_{i,k}$ weights the i^{th} tracking task at stage k , $e_{i,k}$ defines the deviation between the current and reference trajectory, Q_i is a positive semi-definite matrix.

The deviation of the end-effector is given by a combined position and orientation error

$$e_{ee} = \begin{bmatrix} \mathbf{p}_{ee} - \hat{\mathbf{p}}_{ee}^\top \\ \frac{1}{2} \text{tr}(\mathbf{I} - \hat{\mathbf{R}}_{ee} \mathbf{R}_{ee}) \end{bmatrix} \in \mathbb{R}^4 \quad (7)$$

where $\mathbf{p}_{ee} \in \mathbb{R}^3$, $\mathbf{R}_{ee} \in SO(3)$ are respectively the position and rotation of the end-effector with respect to the world reference frame, which can be computed from the state vector \mathbf{x}_k using forward kinematics, $\hat{\mathbf{p}}_{ee}$, $\hat{\mathbf{R}}_{ee}$ are the reference trajectories provided in parameter θ for each stage. We employ the real-valued attitude error function over $SO(3)$ to measure orientation deviation in (7).

The weight $w_{i,k}$ is used to coordinate and manage transitions between different tasks based on specific scenarios. A larger weight indicates higher priority, while a weight of zero signifies that the task is disabled. We represent task

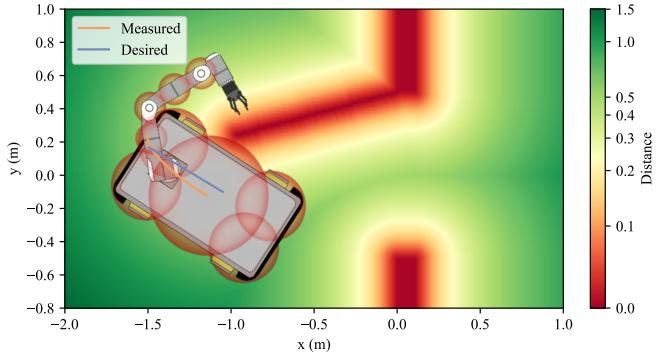


Fig. 3: The ESDF map when the door is open. The collision model includes 5 circles for the base and 4 circles for manipulator links. The end-effector is not covered by the collision circle to allow for contact with the door.

priorities consistently across both task and time dimensions, facilitating smooth transitions in priority. The total tracking cost in MPC is weighted by the corresponding elements in a weight matrix

$$\mathbf{W} = \{\{w_{i,k}\}_{k=0}^{N-1}\}_{i \in \text{Task}}, \quad (8)$$

where a sequence of weights evolving over the MPC horizon is employed for each task to achieve transitional priority. The columns of the matrix represent the relative priority of each task at any given moment.

If the priority of the i^{th} task changes at the switching time t_s , the weight $w_{i,k}$ is updated to the new value for $k > (t_s - t)/\Delta t$. As time t progresses, the proportion of the updated weight gradually increases within the MPC horizon, enabling a smooth transition over time.

2) *Manipulability*: Due to the additional DoF provided by the mobile base, the mobile manipulator exhibits redundancy configurations. To avoid singularities while fully leveraging the redundancy to enhance optimal configurations of the mobile manipulator, we incorporate the maximization of manipulability into our optimization objectives

$$\ell_m = \frac{1}{m(\mathbf{q})}. \quad (9)$$

The manipulability measure $m(\mathbf{q})$ reflects how close the robot is to being singular

$$m(\mathbf{q}) = \sqrt{\det(\mathbf{J}_t(\mathbf{q})\mathbf{J}_t(\mathbf{q})^\top)}, \quad (10)$$

where $\mathbf{J}_t(\mathbf{q})$ is the rows associated with translational movements of the manipulator Jacobian matrix, similar to [25].

Given that the mobile base primarily provides an infinite operational space with mobility, the importance of translational manipulability outweighs rotational manipulability in terms of avoiding singularities. To prevent issues related to dimensional non-homogeneity, only the translational part is taken into consideration.

E. Inequality Constraints

We directly incorporate inequality constraints in the MPC. Simple box constraints are imposed on \mathbf{x}, \mathbf{u} to respect joint

position and velocity limits. To avoid collisions with the environment, especially in narrow doorways, we use the Euclidean Signed Distance Field (ESDF) generated with the mapping system FIESTA [26] for collision checking. We approximate the robot with a series of collision circles as illustrated in Fig. 3. The distance and gradient information are queried from the ESDF at the circle centers, which are determined based on the kinematic model and then projected onto the horizontal plane. Each circle contributes an obstacle avoidance constraint to MPC, defined by

$$h_j(\mathbf{x}) = r_j - d_j(\mathbf{x}) \leq 0, \quad (11)$$

where r_j is the radius of collision circle j and $d_j(\mathbf{x})$ is the closest distance from the circle center j to obstacles.

F. Solving MPC

We select a discretization timestep of $\Delta t = 0.1$ s, implicitly accounting for the latency in the response of the mobile base induced by the relatively large mass. With a time horizon of $T = 2$ s, MPC predicts and optimizes joint-space trajectories for a timestep length of $N = 20$, properly accounting for the effect of current control actions on future states.

We solve the NLP-formulated MPC problem through the sequential quadratic programming (SQP) approach. The iterative solving of SQP can be time-consuming, especially when inequality constraints are present. We employ the Real-Time Iteration (RTI) scheme [27] to perform only one SQP iteration at each control instant and continuously update the problem with the latest feedback states after every iteration. The average computation time of MPC on hardware is 4.3 ms. Thus, its update frequency is set at 100 Hz, achieving high reactivity for real-world implementation. The MPC solution is fed in the following PD control law to compute the joint-space commands at 1000 Hz.

IV. EXPERIMENTS

A. Platform Description

We evaluate our method through several experiments performed on a mobile manipulator consisting of a mobile base equipped with Mecanum wheels and a Flexiv Rizon 4s 7-DoF manipulator (Fig. 1). The sideways movement of the base is locked in all experiments. An Intel Realsense D435 is mounted on the front of the base for initial localization. Torque and joint velocity commands are sent to the manipulator and mobile base respectively with a rate of 1000 Hz. All computations run on an onboard computer with an Intel Core i7-1370PE CPU. The RTI-based SQP is implemented in C++ using *acados* [28]. The model of the mobile manipulator is constructed in symbolic framework *CasADi* [29] by code generation tools [30].

B. Experiment 1: Single Task Tracking

We first evaluate the effectiveness of our proposed method in comparison to the IK-based WBC baseline on trajectory tracking tasks. The IK-based WBC is formulated as a QP

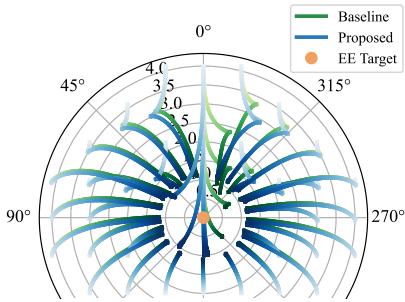


Fig. 4: The mobile manipulator achieves the end-effector target at the origin from various initial positions. The initial orientation of the base is consistently set at 0° . The base trajectories gradually darken from beginning to end.

TABLE I: Examples of manipulability for trajectory from different start points

Start Point	Method	Manipulability		
		Minimum	Average	Final
(-2, 0)	Baseline	1.805	2.015	2.120
	Proposed	1.827	2.108	2.146
(1, -2)	Baseline	1.616	1.804	2.047
	Proposed	1.703	1.942	2.155
(4, 0)	Baseline	1.020	1.364	1.748
	Proposed	1.390	1.659	1.916

problem, similar to the method described in [8]. Both methods share the same cost function composition, except that velocity dampers are added in IK-based WBC to avoid joint position limits.

In Fig. 4, we show how both controllers automatically generate the whole-body motion to match the end-effector task without manually providing any reference trajectory to the base. Table I presents the minimum, average, and final manipulability over the trajectory from three representative start points. Our approach consistently demonstrates higher manipulability across all tested cases compared to the baseline, with improvements particularly evident in complex cases such as turning backward, where an increase of 36% is observed in the worst case. This highlights our method’s enhanced ability to generate optimal trajectories that avoid singularities. This experiment is conducted in simulation to test a larger amount of situations, while all other experiments are performed on real robots.

We further test the reactivity of the proposed controller in the presence of unknown external disturbances. A fixed pose task was set for the end-effector, with no reference provided for the base. We apply external forces to the mobile manipulator and evaluate the end-effector tracking performance. The mobile manipulator resists interference by the motion of its whole body, thus keeping the pose of the end-effector almost unaffected. As shown in Fig. 5, the base spontaneously moves to adapt to the offset of the manipulator, and the maximum error of the end-effector is

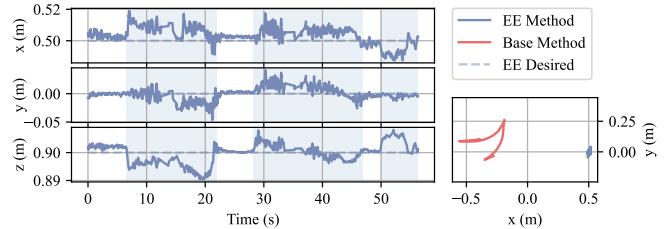


Fig. 5: The trajectory under external force disturbances. *Left:* The trajectory tracking of the end-effector position. *Right:* The end-effector holds at a fixed position while the base moves to resist disturbances.

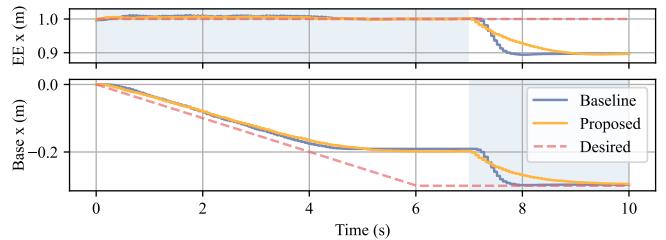


Fig. 6: Trajectory tracking of the end-effector and base along the x -axis. The desired pose for the end-effector is fixed, while the base is commanded to move backward. Shaded areas represent periods of higher task priority.

within 0.05 m.

C. Experiment 2: Task Priority Transition

In the second set of experiments, we evaluate the effectiveness of our approach in coordinating multiple tasks and assess its performance during task priority transitions. References are sent to both the base and the end-effector simultaneously. We construct conflicting tasks that bring the manipulator to a singularity configuration, in order to demonstrate the task priority switching. In this experiment, the manipulability maximization is deliberately excluded from the cost function to avoid instability when encountering singularities.

As shown in Fig. 6, the end-effector task is prioritized before 7 s. During this period, the base gradually moves backward and comes to a stop at a maximum displacement of approximately -0.2 m to maintain the end-effector’s position. A slightly dynamic tracking error in the base is observed due to response delays. At 7 s, the priority task shifts to base tracking, resulting in a compromise of the end-effector task as the base follows the desired trajectory. The maximum velocity of the base during transition is 0.108 m/s, representing a 53% reduction compared to 0.229 m/s in the IK-based WBC. Our approach enables smoother transitions and earlier responses owing to its predictive capabilities.

D. Experiment 3: Door Opening and Traversing

We conduct real-world experiments on a comprehensive scenario involving door opening and traversing, a process that requires tight coordination between the end-effector and the base. The tracking tasks for the mobile manipulator vary

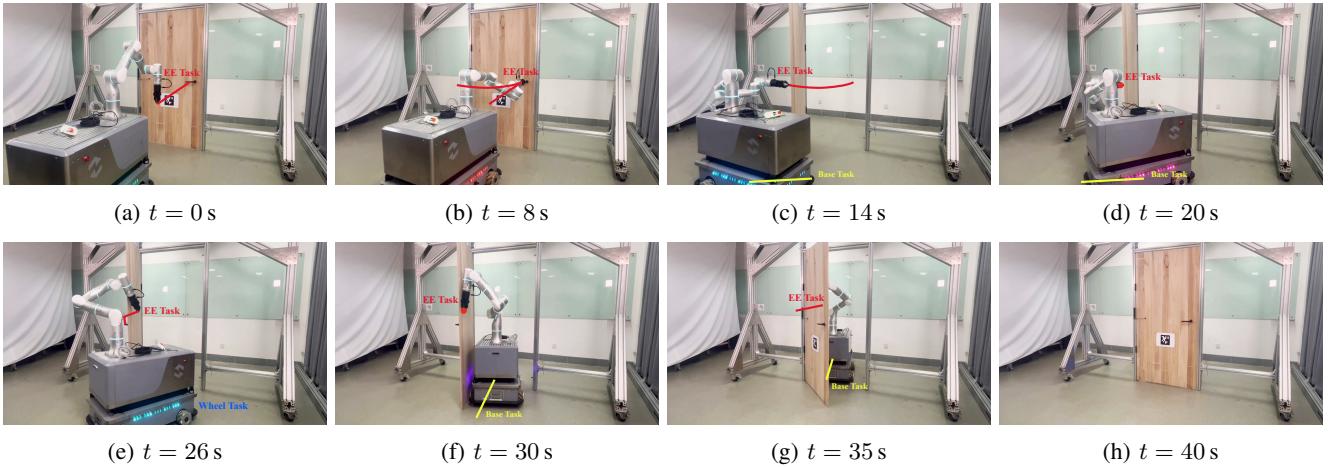


Fig. 7: The mobile manipulator opens and traverses through a self-closing door. The assigned tracking tasks for each stage are highlighted with lines and dots.

over time, alternating between holding the door open and moving forward, as illustrated in Fig. 7. Table II details the time and the enabled tracking tasks associated with each stage.

During the first two stages, the end-effector reference is determined by the positions of the handle and door shaft. Once the door opens to a certain angle, the end-effector maintains a fixed position while an additional task is introduced for the base to reach a given position blocking the door. The gripper then releases the door and moves to the opposite side to hold the door open. In this stage, the base tracking task transitions to a zero-velocity task for wheels, locking both rotation and translation of the base. Finally, the base passes through the door as the end-effector continues to hold the door before retracting, allowing the door to close and completing the process. The entire whole-body motion is generated online by MPC, with references set by a few key points, making the method generalized.

We further evaluate the performance of the mobile manipulator transitioning and coordinating multiple tasks in this experiment. As shown in Fig. 8, the base tracking task is incorporated into the cost function at the switching time $t_s = 14$ s. The desired trajectory is designed as a straight line from the first predicted switching time position to the target position. The trajectory is gradually introduced into the horizon of MPC, resulting in a smooth transition to the blocking stage. Fig. 3 illustrates the trajectory of the base during the blocking stage. The controller generates a base trajectory that conforms to kinematics and avoids collisions, guided by the desired trajectory, while simultaneously ensuring that the end-effector maintains a fixed position to hold the door.

V. CONCLUSION

We present a WBMPC framework to realize optimal task priority transition for mobile manipulation. Our work introduces a unified weight matrix to represent task priorities over both task and time dimensions. This expanded dimensionality

TABLE II: Stages of opening and traversing through the door

Stage	Time	Tracking Tasks		
		End-effector	Base Pos	Wheel Vel
Grasp	0–8 s	Yes	No	No
Open	8–14 s	Yes	No	No
Block	14–20 s	Yes	Yes	No
Release	20–26 s	Yes	No	Yes
Exit	26–40 s	Yes	Yes	No

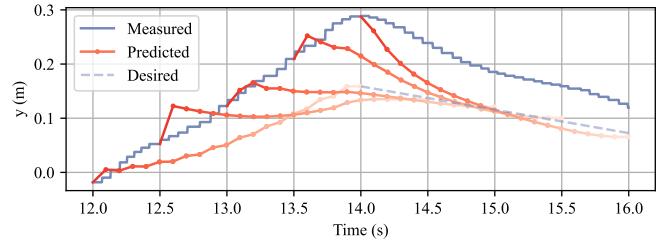


Fig. 8: The trajectories predicted by MPC enable smooth transitions when an additional base task is introduced. The MPC time horizon progressively incorporates the new task.

inherently supports task transitions in composite behaviors, addressing the challenge of simultaneously managing tasks and time within a single controller.

Through experiments 1 and 2, we demonstrate that our method generates well-conditioned joint configurations and achieves smooth task priority transitions, outperforming the IK-based WBC while maintaining high reactivity to external disturbances. Furthermore, we validate our method on a challenging door-opening and traversing scenario, showcasing its effectiveness in handling composite behaviors.

The unification of task and time dimensions in the proposed framework facilitates optimal whole-body motion, opening up new possibilities for mobile manipulators and enabling their deployment in more diverse and demanding real-world scenarios.

REFERENCES

- [1] F. Reister, M. Grotz, and T. Asfour, “Combining navigation and manipulation costs for time-efficient robot placement in mobile manipulation tasks,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9913–9920, 2022.
- [2] B. Burgess-Limerick, C. Lehnert, J. Leitner, and P. Corke, “An architecture for reactive mobile manipulation on-the-move,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1623–1629.
- [3] A. Gupta, M. Zhang, R. Sathua, and S. Gupta, “Opening cabinets and drawers in the real world using a commodity mobile manipulator,” *arXiv preprint arXiv:2402.17767*, 2024.
- [4] H. Xiong, R. Mendonca, K. Shaw, and D. Pathak, “Adaptive mobile manipulation for articulated objects in the open world,” *arXiv preprint arXiv:2401.14403*, 2024.
- [5] Q. Li, Y. Pang, W. Cai, Y. Wang, Q. Li, and M. Zhao, “An overview of multi-task control for redundant robot based on quadratic programming,” in *Chinese Intelligent Automation Conference*. Springer, 2023, pp. 641–666.
- [6] Q.-N. Nguyen and Q.-C. Pham, “Planning optimal trajectories for mobile manipulators under end-effector trajectory continuity constraint,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 14 356–14 362.
- [7] Z. Jiao, Z. Zhang, X. Jiang, D. Han, S.-C. Zhu, Y. Zhu, and H. Liu, “Consolidating kinematic models to promote coordinated mobile manipulations,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 979–985.
- [8] J. Haviland, N. Sünderhauf, and P. Corke, “A holistic approach to reactive mobile manipulation,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3122–3129, 2022.
- [9] 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, 2020.
- [10] M. Mittal, D. Hoeller, F. Farshidian, M. Hutter, and A. Garg, “Articulated object interaction in unknown scenes with whole-body mobile manipulation,” in *2022 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2022, pp. 1647–1654.
- [11] X. Du, S. Zhou, and A. P. Schoellig, “Hierarchical task model predictive control for sequential mobile manipulation tasks,” *IEEE Robotics and Automation Letters*, 2023.
- [12] Y. Wang, Y. Pang, Q. Li, W. Cai, and M. Zhao, “Trajectory tracking control for robot manipulator under dynamic environment,” in *International Conference on Intelligent Robotics and Applications*. Springer, 2023, pp. 513–524.
- [13] D. Kim, J. Di Carlo, B. Katz, G. Bledt, and S. Kim, “Highly dynamic quadruped locomotion via whole-body impulse control and model predictive control,” *arXiv preprint arXiv:1909.06586*, 2019.
- [14] J. Salini, V. Padois, and P. Bidaud, “Synthesis of complex humanoid whole-body behavior: A focus on sequencing and tasks transitions,” in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 1283–1290.
- [15] J. Lee, N. Mansard, and J. Park, “Intermediate desired value approach for task transition of robots in kinematic control,” *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1260–1277, 2012.
- [16] S. Kim, K. Jang, S. Park, Y. Lee, S. Y. Lee, and J. Park, “Continuous task transition approach for robot controller based on hierarchical quadratic programming,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1603–1610, 2019.
- [17] M. Liu, Y. Tan, and V. Padois, “Generalized hierarchical control,” *Autonomous Robots*, vol. 40, pp. 17–31, 2016.
- [18] G. Han, J. Wang, X. Ju, and M. Zhao, “Recursive hierarchical projection for whole-body control with task priority transition,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 312–11 319.
- [19] J.-P. Sleiman, F. Farshidian, M. V. Minniti, and M. Hutter, “A unified mpc framework for whole-body dynamic locomotion and manipulation,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4688–4695, 2021.
- [20] K. Jang, S. Kim, and J. Park, “Motion planning of mobile manipulator for navigation including door traversal,” *IEEE Robotics and Automation Letters*, vol. 8, no. 7, pp. 4147–4154, 2023.
- [21] Z. Wang, Y. Jia, L. Shi, H. Wang, H. Zhao, X. Li, J. Zhou, J. Ma, and G. Zhou, “Arm-constrained curriculum learning for loco-manipulation of the wheel-legged robot,” *arXiv preprint arXiv:2403.16535*, 2024.
- [22] H. Ito, K. Yamamoto, H. Mori, and T. Ogata, “Efficient multitask learning with an embodied predictive model for door opening and entry with whole-body control,” *Science Robotics*, vol. 7, no. 65, p. eaax8177, 2022.
- [23] J.-P. Sleiman, F. Farshidian, and M. Hutter, “Versatile multicontact planning and control for legged loco-manipulation,” *Science Robotics*, vol. 8, no. 81, p. eadg5014, 2023.
- [24] A. M. Bloch, *Nonholonomic Mechanics*. New York, NY: Springer New York, 2015, pp. 235–313. [Online]. Available: https://doi.org/10.1007/978-1-4939-3017-3_5
- [25] J. Haviland and P. Corke, “Neo: A novel expeditious optimisation algorithm for reactive motion control of manipulators,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1043–1050, 2021.
- [26] L. Han, F. Gao, B. Zhou, and S. Shen, “Fiesta: Fast incremental euclidean distance fields for online motion planning of aerial robots,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4423–4430.
- [27] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, “From linear to nonlinear mpc: bridging the gap via the real-time iteration,” *International Journal of Control*, vol. 93, no. 1, pp. 62–80, 2020.
- [28] R. Verschueren, G. Frison, D. Kouzoupias, J. Frey, N. van Duijkeren, A. Zanelli, B. Novoselnik, T. Albin, R. Quirynen, and M. Diehl, “acados – a modular open-source framework for fast embedded optimal control,” *Mathematical Programming Computation*, 2021.
- [29] 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.
- [30] A. Astudillo, J. Carpentier, J. Gillis, G. Pipeleers, and J. Swevers, “Mixed use of analytical derivatives and algorithmic differentiation for NMPC of robot manipulators,” in *Modeling, Estimation and Control Conference MECC 2021*. Elsevier BV, Oct. 2021.