

Hierarchical Incremental MPC for Redundant Robots: a Robust and Singularity-Free Approach

Yongchao Wang¹, Yang Liu¹, Marion Leibold¹, Martin Buss¹, *Fellow, IEEE*,
 and Jinoh Lee^{2,†}, *Senior Member, IEEE*

Abstract—This paper presents a model predictive control (MPC) method for redundant robots controlling multiple hierarchical tasks formulated as multi-layer constrained optimal control problems (OCs). The proposed method, named hierarchical incremental MPC (HIMPC), is robust to dynamic uncertainties, untethered from kinematic/algorithmic singularities, and capable of handling input and state constraints such as joint torque and position limits. To this end, we first derive robust incremental systems that approximate uncertain system dynamics without computing complex nonlinear functions or identifying model parameters. Then the constrained OCs are cast as quadratic programming problems which result in linear MPC, where dynamically-consistent task priority is achieved by deploying equality constraints and optimal control is attained under input and state constraints. Moreover, hierarchical feasibility and recursive feasibility are theoretically proven. Since the computational complexity of HIMPC drastically decreases compared with nonlinear MPC-based methods, it is implemented under the sampling frequency of 1 kHz for physical experiments with redundant manipulator setups, where robustness (high tracking accuracy and enhanced dynamic consistency), admissibility of multiple constraints, and singularity-avoidance nature are demonstrated and compared with state-of-the-art task-prioritized controllers.

Index Terms—Model predictive control, incremental system, task prioritized control, time-delay estimation, redundant robots.

NOMENCLATURE

$\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{> 0}$	Real, non-negative, and positive sets.
$\mathbb{I}, \mathbb{I}_{> 0}$	Integer, and positive integer sets.
$\mathbb{I}_{[a,b]}$	$\mathbb{I}_{[a,b]} = \{x \in \mathbb{I} : a \leq x \leq b\}$.
$\mathbf{I}, \mathbf{O}, \mathbf{0}$	Identity matrix, null matrix, and null vector.
$\text{col}(\mathbf{x}_1, \mathbf{x}_2)$	Stack variable $\text{col}(\mathbf{x}_1, \mathbf{x}_2) := [\mathbf{x}_1^\top, \mathbf{x}_2^\top]^\top$.
$\mathbf{Q} \succ 0$	Positive definite matrix \mathbf{Q} .
$\ \mathbf{x}\ $	Euclidean norm of vector/matrix \mathbf{x} .
$\ \mathbf{x}\ _{\mathbf{Q}}$	$\ \mathbf{x}\ _{\mathbf{Q}} = \sqrt{\mathbf{x}^\top \mathbf{Q} \mathbf{x}}$ for vector \mathbf{x} and $\mathbf{Q} \succ 0$.
$\lambda(\mathbf{Q})$	Eigenvalue of \mathbf{Q} .
$\lambda_{\min}(\mathbf{Q})$	Minimal eigenvalue of \mathbf{Q} .
$\lambda_{\max}(\mathbf{Q})$	Maximal eigenvalue of \mathbf{Q} .
dom	Domain of one function.
$\text{id}(\bullet)$	Identify function, i.e., $\text{id}(x) = x$.
$\hat{\bullet}$	The estimated or approximated value of \bullet .

This work was partially supported by the National Natural Science Foundation of China (No. 62303484) and by ITECH R&D programs of MOTIE/KEIT under projects No. 20014398 and No. 20014485. ([†]Corresponding Author: Jinoh Lee)

¹Authors are with Chair of Automatic Control Engineering (LSR), Technical University of Munich, Munich, 80333, Germany.

²Author is with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany, and is also an adjunct professor with the Department of Mechanical Engineering, Korea Advanced Institute of Science & Technology (KAIST), Daejeon, Korea.

Copyright ©2024 IEEE

I. INTRODUCTION

KINEMATICALLY redundant robotic systems (e.g., high-degree-of-freedom robot manipulators and humanoid robots) are capable of executing several operational space tasks simultaneously. For example in robotic welding, the primary task is often to implement accurate trajectory tracking of its welding torch, i.e., end-effector (EE), and to obtain satisfactory welding seam additional tasks such as maintaining the orientation of EE will be performed by deploying degree-of-redundancy of the robot, e.g. the welding torch is expected to be perpendicular to the welding surface or maintain a specific angle. Although the multiple tasks can be controlled by being concatenated into a single task vector, however, if a control conflict occurs among tasks, control performance will be adversely affected.

To ensure conflict resolution, the operational space control was employed to bestow control capability of multiple tasks in a strictly prioritized manner for redundant robots, mostly with the null-space projection method [1]–[6]. These are called task-prioritized controllers grouped into three categories [6], where either the desired joint velocities [1], [2], or accelerations [3], or forces/torques [4], [5] are computed respectively. Since most of commercial robots do not allow for a torque-level control interface, the velocity-based scheme is preferred in robotic applications. Nevertheless, for tracking with the second-order dynamics of a rigid body robot system, the acceleration-based and force/torque-based schemes are more appealing owing to the explicit incorporation of accelerations.

In particular, the force/torque-based scheme brings significant advantages in such applications that manipulation in contact with environments is required, and constraints on torques are imposed to guarantee safety. In the last few decades, a variety of force/torque-based methods, such as operational space formulation (OSF) [4]–[12], hierarchical quadratic programming (HQP) [13]–[16], and more recently model predictive control (MPC) [17], [18], are developed. Nevertheless, the most practical performance criteria—*robustness*, *singularity-avoidance*, *optimality*, and *constraint admissibility*—have not been simultaneously conquered yet. Concurrently addressing these issues is challenging given the inevitable modeling errors, inherent inversion operation in algorithms and limited computing power of systems. Accordingly, this article aims at developing a task-prioritized control scheme taking into account all aforementioned criteria.

A. Related Works

1) *OSF Schemes*: The OSF [4] relying on null-space projections [19] has been widely used and became a popular

tool for task-prioritized control since it has been introduced to robotics in late 80's. It offers dynamically-decoupled control of multiple tasks based on a property known as *dynamic consistency*: forces/torques for lower priority tasks do not affect the execution of high-priority tasks [5], where model-based feedback linearization is applied on all hierarchical levels and decoupled control behavior of each task is obtained. Recently, the formal null-space stability of hierarchical control is theoretically analyzed for a regulation case in [7], [8] and for a tracking case in [9], respectively, although this is generally known to be difficult to be proven [6], [20].

Nevertheless, both the novel hierarchical controller and the OSF are vulnerable to modeling errors [6] since precisely identified parameters of the mathematical model are crucial to attain feedback linearization in the controller design. In practice, unfortunately, modeling errors are inevitable and moreover, dynamics terms undergo severe changes especially when the robotic system carries/releases unknown payloads, which indeed deteriorate desired control performance.

To deal with modeling errors, adaptive methods [21] and learning techniques [22] are propitiously employed in operational space control, where uncertain/unknown functions are identified or optimal control is found online that maximizes an immediate reward. However, the computational complexity is such high that it is difficult to apply in practice. Moreover, parameters have to be heuristically selected, which also increases the complexity of these methods. As an alternative, time-delay estimation (TDE) based OSF methods were developed in [10]–[12]. TDE [23]–[27] is a model-free controller design method, which uses time-delayed input and output signals to estimate system dynamics without a concrete mathematical model, laborious parameter identification, and linearization around equilibrium points of the system. It results in enhanced accuracy in terms of *dynamic consistency* and control performance along with enhanced computational efficiency.

Another hurdle in the OSF-based methods, typically using null-space projection [4]–[12], is the singularity problem, especially the algorithmic one which occurs when tasks conflict significantly; this finally results in unstable behavior. Although a damping factor [28] and a continuous null-space projection [29] are introduced to address the singularity problem, there is a laborious tuning process of parameters and unavoidable trade-off between accuracy and feasibility of the solution. In addition, physical limitations, such as input and state constraints, are also not considered in all approaches mentioned so far. Constraints are generally used to describe safety requirements and should be considered in safety-critical applications, e.g., close to or in cooperation with humans.

2) *HQP Schemes*: Considering input and/or state (motion) constraints, HQP is developed in the hierarchically-ordered framework of multiple quadratic programs (QPs) [13]–[16], where the task priority is strictly achieved using the hierarchical structure of the QPs [2], [30]. In [13], the robustness of HQP is addressed by sliding mode control (SMC). However, since this method is based on the null-space projection to design the sliding variable of hierarchical tasks, a potential risk of singularity exists.

Without using the null-space projection method, the authors

in [14] implement the task hierarchy by solving the cascaded QPs in lexicographic order, while the authors in [15], [16] propose that equality constraints are imposed to guarantee strict task hierarchy. Thus, in [14]–[16], algorithmic singularity is avoided. However, once the Jacobian matrix becomes singular, Hessian matrices of QPs in [14], [15] are no longer positive definite, and kinematic singularity occurs. This is because, for QP solvers, numerical weakness increases with the rise of the condition number of the Hessian matrix. When the Hessian matrix is positive semi-definite, undesired (large-value) control signals are computed and the system becomes unstable. Whereas, although the Hessian matrix is always positive definite in [16], the risk of kinematic singularity still remains because inverse calculation of the terms with respect to (w.r.t.) Jacobian matrices is exhibited in the algorithm.

Additionally, it is noteworthy that solutions of the HQP are optimal to the current robot configuration, but not w.r.t. global tasks. In other words, the locally optimal controllers might drive the controlled robotic systems to disadvantageous configurations in the context of global tasks [31]. Last, the HQP control structure may cause torque peaks and oscillations [3], attributed to its limited (a single step) prediction horizon.

3) *MPC Schemes*: Model predictive controllers have been proposed to address drawbacks of locally optimal controllers [31], [32]. Given that control performance is considered over a finite-time horizon [33]–[35], MPC provides a powerful option to fulfill control objectives of multiple tasks while constraints are not violated. Nevertheless, in [31], [32], the task hierarchy is acquired via a weight-prioritized optimization problem [36]. It results in a soft task-prioritized inverse kinematics control scheme. When tasks conflict, control performance of the highest-priority task, which is usually the primary or safety-related work, will be adversely compromised. The torque-based task-prioritized MPC schemes are developed independently in [17], [18]. A single convex constrained optimization problem is constructed in [17] where the task hierarchy is achieved by a quadratic inequality constraint that tracking errors of high-priority tasks must be less than or equal to that of low-priority tasks. It still results in a soft task hierarchy. Besides, the desired joint position and velocities are calculated using the null-space projection method, resulting again in risk of algorithmic singularity. In addition, predictions are generated based on the nominal model of robotic systems while uncertainties and/or disturbances are not considered.

To improve robustness of MPC, learning-based MPC methods [37], [38] are developed for a single-task control problem, where Gaussian processes (GP) and neural networks (NN) are employed to identify nonlinear models online. Unfortunately, GP and NN learning techniques further increase the computational complexity of MPC. In [39], the SMC scheme is combined with MPC to address modeling errors and external disturbances with better computational efficiency for a single task. However, when multiple tasks are considered, task hierarchy will be adversely affected by the compensating SMC method. To guarantee task hierarchy, the task-prioritized sliding variable for each layer is designed and then an operational space model predictive SMC is developed in [18]. Nevertheless, the task-prioritized sliding variable is

designed using the null-space projection method, which means it still suffers from the risk of algorithmic singularity. Besides, the aforementioned SMC-based methods request the nominal mathematical model to be fairly identified. In addition, when they are applied in task-space control schemes, due to the introduced pseudo control law, constraints imposed on joint positions, velocities and torques are nonlinear and non-convex and thus the considered optimal control problems (OCPs) are general nonlinear programming (NLP) problems whose convexity is not guaranteed. In the non-convex programming problem, the solution is easy to fall into local optimum and is computationally demanding in general.

The permitted sampling period of robotic systems is usually down to hundredths or even thousandths of a second, and thus with the general NLP solvers, such as IPOPT [40] and LOQO [41], it is difficult to guarantee real-time control, finally causing feedback delays. The most popular and commonly used method is sequential quadratic programming (SQP) [42]–[44], where a series of approximated QP problems is solved sequentially until the solution converges. Owing to the efficient SQP, the computation time is down to the order of milliseconds or tens of milliseconds [42]–[44]. However, due to demands for high-precision in robotics systems, the fast control frequency, e.g., 1 kHz, is generally suggested since the control accuracy is in line with the sampling rate [10]–[12], [25]–[27], [45]–[48]. To this end, the approximated algorithm which trades control performance for speed, such as real-time iteration (RTI) [49] and feasibility-driven differential dynamic programming (FDDP) [50], are developed to avoid solving NLP iteratively. In RTI, a single convex QP that locally approximates the original optimization problem is solved per feedback step [46]. The gap contraction of FDDP is equivalent to direct multiple-shooting formulations [50]. However, only equality constraints can be considered in FDDP. Besides, applications of RTI and FDDP rely on accurate models of controlled plants and solver stability problems arise in the presence of reference changes and/or large external disturbances [51]. In contrast to NLP solvers, QP solvers which are running at microseconds deliver solutions in a more reliable and efficient way [52], [53]. Therefore, it is appealing that the task-prioritized MPC scheme can be formulated into a linear form whose OCP can be cast as a QP.

B. Method and Contributions

In this article, a task-prioritized control scheme is proposed for robotic systems modeled by Euler-Lagrange equations and we call it hierarchical incremental MPC (HIMPC). To improve robustness against uncertainties and disturbances, we first develop a robust incremental model exploiting the TDE method. Then, the HIMPC is formulated with multi-level constrained OCPs sequentially ordered in accordance with task priorities. The state predictions are generated from the discretized incremental model. Task hierarchy is fulfilled through equality constraints on control signals for lower-priority tasks, based on the *dynamic consistency* principle. Moreover, each constrained OCP of the HIMPC is cast to a QP. The hierarchical feasibility and recursive feasibility of the HIMPC are verified theoretically.

The contributions of this article are summarized as follows:

- 1) *Robust.* Robustness of the controller in terms of tracking accuracy and also *dynamic consistency* is enhanced. Different from existing task-prioritized MPC [17], [18] and optimization-based methods [2], [13]–[16], the nominal mathematical model of the robotic system will not be required for the proposed controller. We will approximate both system dynamics and equations of motion of tasks using TDE. This model-free nature improves robustness against uncertainties and disturbances. Besides, TDE will allow that task priority constraints are set independent of the accurate inertia matrix.
- 2) *Computationally Efficient.* Relying on the TDE method will not only improve robustness of the proposed method but also simplify nonlinear equations to linear ones. The proposed HIMPC is essentially a linear MPC and each constrained OCP can be cast to a QP. Compared with the nonlinear task-prioritized MPC schemes developed in [17], [18], the proposed HIMPC decreases computational complexity dramatically, which makes it possible to allow for real-time control in milliseconds.
- 3) *Singularity-Free.* The null-space projection idea will not be employed in this approach, and task hierarchy is achieved by imposing equality constraints on input signals for lower-priority tasks. Since no inverse calculation is involved in these equality constraint equations, algorithmic singularity is avoided. Moreover, the inverse calculation of terms w.r.t. Jacobian matrices is not required to formulate the constrained OCPs, and the Hessian matrix of each constrained OCP is verified to be positive definite. Thus, the proposed HIMPC is also a kinematically singularity-free method.

C. Organisation

The rest of this article is organized as follows: in Sec. II, the robotic system dynamics and control objective are introduced. In Sec. III, equations of motion of tasks and system dynamics are approximated by incremental systems using TDE, which do not need dynamic model parameters. Further, the accuracy of this approximation method is analyzed. Sec. IV presents the HIMPC. The hierarchical and recursive feasibility of HIMPC is analyzed. In Secs. V and VI, the effectiveness of the proposed method is verified by simulation and experimental results. Finally, Sec. VII draws a conclusion and discussion on future research directions.

II. PRELIMINARIES

This section is devoted to giving the problem statement in terms of a control objective. Therefore, at first, the robotic system dynamics and then the tasks are introduced.

A. Robot Dynamics

The dynamics of an n -link robotic system is given by an Euler-Lagrange equation [45]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_d = \boldsymbol{\tau}, \quad (1)$$

where $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$ represent the vectors of position, velocity, and acceleration, respectively. $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ is the symmetric

inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ is the Coriolis/centrifugal matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$ contains the gravitational terms exerting on the robotic system, $\mathbf{F}(\dot{\mathbf{q}}) \in \mathbb{R}^n$ denotes viscous friction, $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of torque supplied by the actuators, and $\boldsymbol{\tau}_d \in \mathbb{R}^n$ denotes the unknown disturbance.

The uniform positive definiteness property of $\mathbf{M}(\mathbf{q})$ is introduced in the following. Later we will need this property to determine design parameters (see *Remark 4*).

Property 1 (Positive Definite Inertia Matrix [45], [47]): The inertia matrix $\mathbf{M}(\mathbf{q})$ is uniformly positive definite and there exist constants $\mu_1, \mu_2 \in \mathbb{R}_{>0}$ such that each eigenvalue of $\mathbf{M}(\mathbf{q})$, denoted by $\lambda_i(\mathbf{M}(\mathbf{q}))$, satisfies the following:

$$\mu_1 \leq \lambda_i(\mathbf{M}(\mathbf{q})) \leq \mu_2, \quad \forall i \in \mathbb{I}_{[1,n]}.$$

Note that due to inevitable modeling errors, the inertia matrix $\mathbf{M}(\mathbf{q})$, and other dynamics terms $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$, $\mathbf{G}(\mathbf{q})$, and $\mathbf{F}(\dot{\mathbf{q}})$ are considered to be unknown/uncertain in this article. Besides, to guarantee the system is controllable, the unknown disturbance $\boldsymbol{\tau}_d$ is assumed to be bounded [54].

B. Task Hierarchy, Tasks, and Control Objective

A control task hierarchy including $r \in \mathbb{I}_{>0}$ levels is introduced and it is assumed that the i -th task has a lower priority level than all previous ($i-1$) tasks [9]. Especially, the first and r -th tasks have the highest and lowest priority level, respectively. The task-space coordinate $\mathbf{x}_i \in \mathbb{R}^{m_i}$ of the i -th task is

$$\mathbf{x}_i = \mathbf{f}_i(\mathbf{q}), \quad i \in \mathbb{I}_{[1,r]}, \quad (2)$$

where $\mathbf{f}_i(\mathbf{q}) : \mathbb{R}^n \rightarrow \mathbb{R}^{m_i}$ is the task mapping, which is known exactly and continuously differentiable. With the continuous differentiable task mapping $\mathbf{f}_i(\mathbf{q})$, the Jacobian $\mathbf{J}_i(\mathbf{q}) \in \mathbb{R}^{m_i \times n}$ corresponding to \mathbf{x}_i is defined as

$$\mathbf{J}_i(\mathbf{q}) = \frac{\partial \mathbf{f}_i(\mathbf{q})}{\partial \mathbf{q}}. \quad (3)$$

Since $\mathbf{f}_i(\mathbf{q})$ is exactly known in general, all Jacobians are regarded as known functions without uncertainties.

In this article, task i is defined to make \mathbf{x}_i track a desired reference trajectory \mathbf{x}_{id} and achieve the following target motion dynamics in free space [10]:

$$\ddot{\tilde{\mathbf{x}}}_i + \mathbf{K}_{V_i} \dot{\tilde{\mathbf{x}}}_i + \mathbf{K}_{P_i} \tilde{\mathbf{x}}_i = \mathbf{0}, \quad (4)$$

where $\tilde{\mathbf{x}}_i := \mathbf{x}_i - \mathbf{x}_{id}$ is the tracking error, and $\mathbf{K}_{V_i} \succ 0$ and $\mathbf{K}_{P_i} \succ 0$ denote damping and stiffness matrices, respectively.

Non-smooth reference trajectories can cause damage to mechanical systems due to sharp actuator torque changes. Thus, \mathbf{x}_{id} is assumed to be bounded and smooth.

Assumption 1 (Bounded and Smooth Reference Trajectory, [55]): The reference trajectory \mathbf{x}_{id} is bounded and smooth, satisfying $0 \leq \underline{\mathbf{x}}_i \leq \|\mathbf{x}_{id}\| \leq \bar{\mathbf{x}}_i < \infty$, $0 \leq \underline{\dot{\mathbf{x}}}_i \leq \|\dot{\mathbf{x}}_{id}\| \leq \bar{\dot{\mathbf{x}}}_i < \infty$, and $0 \leq \underline{\ddot{\mathbf{x}}}_i \leq \|\ddot{\mathbf{x}}_{id}\| \leq \bar{\ddot{\mathbf{x}}}_i < \infty$.

Note that exact values of $\underline{\mathbf{x}}_i$, $\bar{\mathbf{x}}_i$, $\underline{\dot{\mathbf{x}}}_i$, $\bar{\dot{\mathbf{x}}}_i$, $\underline{\ddot{\mathbf{x}}}_i$ and $\bar{\ddot{\mathbf{x}}}_i$ in *Assumption 1* are not required for the controller design.

We aim to design a strict task-prioritized controller which is robust yet singularity-free while physical constraints of the

robot manipulator, such as constraints on joint positions, velocities, and torques, are satisfied. The following box constraints are considered:

$$\mathbf{q}_{\min} \leq \mathbf{q} \leq \mathbf{q}_{\max}, \quad \dot{\mathbf{q}}_{\min} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{\max}, \quad \boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{\max},$$

where $\bullet_{\min}, \bullet_{\max}$ denote specific lower/upper bounds of \bullet .

Remark 1 (Delete an Assumption of Non-Singular Task Space): In literature, it is common to avoid singularities by restricting allowed tasks, and one assumption is made that goal tasks are designed not to include singularities. However, this assumption delimits the feasible task space of the robotic system. In this paper, to ensure sufficient task space for the robotic system and to guarantee safety, we will strive for designing a singularity-free controller. Thus, the assumption on non-singular task space is removed.

III. ROBUST INCREMENTAL SYSTEMS

In this section and the following Sec. IV, a task-prioritized control scheme is developed. We aim at taking into account optimality, input and state constraints, strict task hierarchy, and robustness, simultaneously. To achieve optimality and address input and state constraints, the control scheme will be developed in the framework of MPC. Considering the unknown system dynamics and also mitigating the computational complexity of the algorithm, the TDE method is used to accurately approximate equations of motion of tasks and system dynamics to avoid online identification or other time-consuming operations. Besides, the accuracy of the resulting TDE-based incremental systems is analyzed.

A. Derivation of Two Incremental Systems

In this subsection, uncertain equations of motion of tasks and system dynamics will be approximated using TDE, and incremental systems are obtained, which are used to generate state predictions.

As stated in the control objectives in Sec. II-B, we aim to impose constraints on torques. Thus, the torque is selected as the control variable in the proposed task-prioritized controller. First, the equations of motion of tasks are derived with the torques being the inputs. With (1) and (3), the equation of motion of the i -th task is obtained as follows:

$$\begin{aligned} \ddot{\mathbf{x}}_i &\stackrel{(3)}{=} \dot{\mathbf{J}}_i \dot{\mathbf{q}} + \mathbf{J}_i \ddot{\mathbf{q}} \\ &\stackrel{(1)}{=} \dot{\mathbf{J}}_i \dot{\mathbf{q}} + \mathbf{J}_i \mathbf{M}^{-1} (\boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} - \mathbf{G}(\mathbf{q}) - \mathbf{F}(\dot{\mathbf{q}}) - \boldsymbol{\tau}_d), \end{aligned} \quad (5)$$

which includes uncertain system dynamics terms and unknown disturbance. To deal with these uncertainties and reduce dependency on the concrete mathematical model, the TDE technique [23]–[27] is employed to approximate the equation of motion (5), which involves the following two steps:

Step 1: Separating Uncertain Terms from Known Terms. It is observed that uncertainties in the equations of motion (5) originate from the uncertain system dynamics (1). Thus, we will investigate (1) first. Introducing a diagonal positive definite matrix $\bar{\mathbf{M}}$, (1) is transformed into:

$$\bar{\mathbf{M}} \ddot{\mathbf{q}} + \underbrace{(\mathbf{M} - \bar{\mathbf{M}}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_d}_{\mathbf{H}_q} = \boldsymbol{\tau}. \quad (6)$$

Using (6), the equation of motion (5) is rewritten as:

$$\ddot{\mathbf{x}}_i = \underbrace{\dot{\mathbf{J}}_i \dot{\mathbf{q}} - \mathbf{J}_i \bar{\mathbf{M}}^{-1} \mathbf{H}_q}_{\mathbf{H}_i} + \mathbf{J}_i \bar{\mathbf{M}}^{-1} \boldsymbol{\tau}, \quad (7)$$

where \mathbf{H}_i is also uncertain/unknown because of the lumped uncertain nonlinear function \mathbf{H}_q in (6).

Step 2: Using the Time-Delayed Signals/Functions to Approximate the Unknown Terms. The value of \mathbf{H}_i at the time t is approximated by that of \mathbf{H}_i at $(t-L)$ for a sufficiently small delay time L [25]–[27]:

$$(\mathbf{H}_i)_t \cong (\mathbf{H}_i)_{(t-L)}. \quad (8)$$

We abbreviate $\ddot{\mathbf{x}}_{i,0} := (\ddot{\mathbf{x}}_i)_{(t-L)}$, $\boldsymbol{\tau}_0 := \boldsymbol{\tau}_{(t-L)}$, and $\mathbf{J}_{i,0} = (\mathbf{J}_i)_{(t-L)}$ for simplicity. From (7) and (8), one obtains the TDE of \mathbf{H}_i , i.e., $(\mathbf{H}_i)_{(t-L)}$, as follows:

$$(\mathbf{H}_i)_{(t-L)} = \ddot{\mathbf{x}}_{i,0} - \mathbf{J}_{i,0} \bar{\mathbf{M}}^{-1} \boldsymbol{\tau}_0. \quad (9)$$

Comparing (7) with (9), we see that the input matrices $\mathbf{J}_i \bar{\mathbf{M}}^{-1}$ and $\mathbf{J}_{i,0} \bar{\mathbf{M}}^{-1}$ are different. We will unify these input matrices using the continuity property. In practice, a digital system can be regarded as a continuous system when the sampling rate is faster than 30 times the system bandwidth [56]. Thus, according to the continuity property, the variations in the Jacobian \mathbf{J}_i during a short time period are negligible. That is, formally, for a sufficiently small L ,

$$\mathbf{J}_i \cong \mathbf{J}_{i,0}. \quad (10)$$

With (10), we then have an alternative TDE of \mathbf{H}_i , i.e., $\hat{\mathbf{H}}_i$, as follows:

$$\hat{\mathbf{H}}_i = \ddot{\mathbf{x}}_{i,0} - \mathbf{J}_i \bar{\mathbf{M}}^{-1} \boldsymbol{\tau}_0. \quad (11)$$

Finally, we obtain the incremental version of (5) with the combination of (7) and (11)

$$\ddot{\mathbf{x}}_i = \ddot{\mathbf{x}}_{i,0} + \mathbf{J}_i \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau} + \boldsymbol{\varepsilon}_x, \quad (12)$$

where $\Delta \boldsymbol{\tau} := \boldsymbol{\tau} - \boldsymbol{\tau}_0$ is the incremental control signal, and $\boldsymbol{\varepsilon}_x := \mathbf{H}_i - \hat{\mathbf{H}}_i$ is the TDE error in i -th task space which is considered as a disturbance to the incremental system (12). Note that only current values of $\ddot{\mathbf{x}}_i$ and $\boldsymbol{\tau}$ are used in (11) to get the approximation while complex and uncertain functions, such as $\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$, $\mathbf{G}(\mathbf{q})$, and $\mathbf{F}(\mathbf{q})$, are not required. To guarantee the approximation accuracy, we also assume state measurement errors are sufficiently small.

In accordance with the formal TDE implementation [10]–[12], [23]–[27], the sampling period T_s is selected as L , which guarantees the delay time is sufficiently small. Then according to the continuity property, $\boldsymbol{\varepsilon}_x$ can be sufficiently small when we select a sufficiently small delay time L . For $\boldsymbol{\varepsilon}_x = \mathbf{0}$, the nominal incremental system for i -th task is obtained as

$$\ddot{\mathbf{x}}_i = \ddot{\mathbf{x}}_{i,0} + \mathbf{J}_i \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau}. \quad (13)$$

In what follows, the Euler method is used to obtain a discrete-time version of (13), which is used to generate predictions of \mathbf{x}_i . Since the sampling period is sufficiently small, the discretization error is ignored. Thus, we have

$$\bar{\mathbf{x}}_i(k+1) = \mathbf{A}_{11} \bar{\mathbf{x}}_i(k) + \mathbf{A}_{12} \bar{\mathbf{x}}_i(k-1) + \mathbf{B}_{1,i} \Delta \boldsymbol{\tau}(k), \quad (14)$$

where

$$\bar{\mathbf{x}}_i := \text{col}(\mathbf{x}_i, \dot{\mathbf{x}}_i, \ddot{\mathbf{x}}_i), \quad \mathbf{A}_{11} := \begin{bmatrix} \mathbf{I} & T_s \mathbf{I} & \mathbf{O} \\ \mathbf{O} & 2\mathbf{I} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{I} \end{bmatrix},$$

$$\mathbf{A}_{12} := \begin{bmatrix} \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & -\mathbf{I} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} \end{bmatrix}, \quad \mathbf{B}_{1,i} := \begin{bmatrix} \mathbf{O} \\ \mathbf{J}_i \bar{\mathbf{M}}^{-1} T_s \\ \mathbf{J}_i \bar{\mathbf{M}}^{-1} \end{bmatrix}.$$

We define a stack variable $\bar{\mathbf{x}}_i(k+1) := \text{col}(\bar{\mathbf{x}}_i(k+1), \bar{\mathbf{x}}_i(k))$, and rewrite (14) as a canonical linear equation:

$$\bar{\mathbf{x}}_i(k+1) = \mathbf{A}_1 \bar{\mathbf{x}}_i(k) + \mathbf{B}_{1,i} \Delta \boldsymbol{\tau}(k) \quad (15)$$

with $\mathbf{A}_1 := \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{I} & \mathbf{O} \end{bmatrix}$, $\mathbf{B}_{1,i} := \begin{bmatrix} \mathbf{B}_{1,i} \\ \mathbf{O} \end{bmatrix}$.

It is noteworthy that the approximated equation of motion of the i -th task (13) is general and it is also suitable for posture tasks which are normally to regulate specific joint positions. Similarly, the approximated system dynamics can also be obtained, which will be used to predict joint space behavior (\mathbf{q} and $\dot{\mathbf{q}}$) to be able to evaluate constraints in joint space during predictions. In analogy to the procedures in *Step 1* and *Step 2* with the Jacobian matrix $\mathbf{J} := \mathbf{I}$, the following incremental system is obtained:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_0 + \bar{\mathbf{M}}^{-1} (\Delta \boldsymbol{\tau} + \boldsymbol{\varepsilon}_q), \quad (16)$$

where $\boldsymbol{\varepsilon}_q := (\mathbf{H}_q)_{(t-L)} - \mathbf{H}_q$ is the TDE error in joint space.

Let $\boldsymbol{\varepsilon}_q = \mathbf{0}$ from (16), one can obtain the nominal incremental system in the joint-space coordinate as follows:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_0 + \bar{\mathbf{M}} \Delta \boldsymbol{\tau}. \quad (17)$$

Then, similar to (15), let $\bar{\mathbf{q}}(k+1) := \text{col}(\bar{\mathbf{q}}(k+1), \bar{\mathbf{q}}(k))$, $\bar{\mathbf{q}} := \text{col}(\mathbf{q}, \dot{\mathbf{q}})$ and again a canonical linear equation is obtained:

$$\bar{\mathbf{q}}(k+1) = \mathbf{A}_2 \bar{\mathbf{q}}(k) + \mathbf{B}_2 \Delta \boldsymbol{\tau}(k) \quad (18)$$

with $\mathbf{A}_2 := \begin{bmatrix} \mathbf{A}_{21} & \mathbf{A}_{22} \\ \mathbf{I} & \mathbf{O} \end{bmatrix}$, $\mathbf{B}_2 := \begin{bmatrix} \mathbf{B}_{21} \\ \mathbf{O} \end{bmatrix}$, $\mathbf{A}_{21} := \begin{bmatrix} \mathbf{I} & T_s \mathbf{I} \\ \mathbf{O} & 2\mathbf{I} \end{bmatrix}$, $\mathbf{A}_{22} := \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{O} & -\mathbf{I} \end{bmatrix}$, $\mathbf{B}_{21} := \begin{bmatrix} \mathbf{O} \\ T_s \bar{\mathbf{M}}^{-1} \end{bmatrix}$.

In brief, using an auxiliary matrix $\bar{\mathbf{M}}$, we found two discrete-time incremental systems as approximations of equations of motion and system dynamics (cf. (15) and (18)). We will use (15) and (18) to generate state predictions over the prediction horizon.

Remark 2 (The Differences between the TDE Formulation in this Article and that in [10]): In [10], the TDE approximation was formulated by using conventional null-space projection, where task Jacobians are assumed to be full rank, i.e., the considered task space is non-singular. Whereas, in the proposed formulation as seen in (13), inversion of terms involving Jacobian matrices is not required and the assumption on non-singular task space is relaxed.

Remark 3 (Why Including $\dot{\mathbf{J}}_i \dot{\mathbf{q}}$ in the Lumped Function \mathbf{H}_i): It is noteworthy that the inclusion of $\dot{\mathbf{J}}_i \dot{\mathbf{q}}$ in the lumped function \mathbf{H}_i in (7) enables a canonical linear system approximation (cf. (15)) plugged into the linear MPC framework as introduced in Sec. IV. In fact, since time derivatives of the Jacobian ($\dot{\mathbf{J}}_i$) can be computed from the exact kinematic parameter values and mapping function $\mathbf{f}_i(\mathbf{q})$, it is not necessary to include the term $\dot{\mathbf{J}}_i \dot{\mathbf{q}}$ in the lumped function \mathbf{H}_i .

However, this requests a nonlinear MPC framework which is computationally so demanding that real-time execution is impossible. In addition, the proposed linear formulation will allow a more in-depth and accessible theoretical analysis of the controller performance. Moreover, owing to its continuity property, the lumped \mathbf{H}_i including $\dot{\mathbf{J}}_i\dot{\mathbf{q}}$ term can be accurately approximated by time-delayed signals with a sufficiently high sampling rate; this also saves additional computation of $\dot{\mathbf{J}}_i\dot{\mathbf{q}}$.

Remark 4 (How to Determine $\bar{\mathbf{M}}$): In existing TDE-based control schemes [10]–[12], [25]–[27], the stability condition $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| \leq 1$ needs to be fulfilled. To avoid performance deterioration by inappropriate $\bar{\mathbf{M}}$, this stability condition is also required to be satisfied in this article. For a specific robotic system, the principle to determine the diagonal matrix $\bar{\mathbf{M}}$ such that $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| \leq 1$ is presented as follows. According to *Property 1*, $\lambda_i(\mathbf{M})$ is bounded. Suppose \bar{M}_i is the i -th diagonal element of $\bar{\mathbf{M}}$, then, $\forall i \in \mathbb{I}_{[1,n]}$, $\left(1 - \frac{\bar{M}_i}{\lambda_i}\right)$ is an eigenvalue of $(\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}})$. If $|1 - \frac{\bar{M}_i}{\lambda_i}| < 1$, i.e., \bar{M}_i satisfies $0 < \bar{M}_i < 2\lambda_i$, then $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| < 1$ holds. This also implies that the sufficient condition $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| < 1$ can be achieved by a small positive \bar{M}_i although exact expressions and eigenvalues of \mathbf{M} are unknown [27]. From the nature of TDE, too small $\bar{\mathbf{M}}$ results in large TDE errors while too large $\bar{\mathbf{M}}$ causes noisy responses, and vice versa. Accordingly, the constant $\bar{\mathbf{M}} := \text{diag}(\bar{M}_1, \dots, \bar{M}_n)$ is selected in a manual tuning process [10]: (i) begin with a sufficiently small positive \bar{M}_i to guarantee stability; and (ii) increase \bar{M}_i until tracking performance is satisfactory or the closed-loop system almost shows a noisy response. In practice, a wide range of $\bar{\mathbf{M}}$ can be selected [23]–[27], since the error $(\mathbf{M} - \bar{\mathbf{M}})$ will be compensated by time-delayed signals (cf. Sec. III-B).

B. Analysis of Approximation Accuracy

Although the exact mathematical model of the robotic system is not required, there is a discrepancy between equations, (1) and (5), and the nominal incremental systems, (13) and (17), because of inevitable TDE errors. In this subsection, we will show that the incremental system exhibits a higher approximation accuracy than the nominal nonlinear mathematical model (abbreviated as the nominal model in Sec. III-B).

First, approximation errors for the nominal model cases are derived. Denote the nominal inertial matrix, Coriolis/centrifugal matrix, gravitational matrix, and viscous matrix as \mathbf{M}_n , \mathbf{C}_n , \mathbf{G}_n , and \mathbf{F}_n , respectively. In accordance with (1), the following nominal system dynamics (19) is used to estimate \mathbf{q} and $\dot{\mathbf{q}}$, where $\hat{\mathbf{q}}_{\text{nom}}$ is the approximation for the nominal model case:

$$\ddot{\hat{\mathbf{q}}}_{\text{nom}} = -\mathbf{M}_n^{-1}\mathbf{N}_n + \mathbf{M}_n^{-1}\boldsymbol{\tau}, \quad (19)$$

where $\mathbf{N}_n := \mathbf{C}_n\dot{\mathbf{q}} + \mathbf{G}_n + \mathbf{F}_n$ is the nominal value of the lumped nonlinear dynamics term \mathbf{N} ($\mathbf{N} := \mathbf{C}\dot{\mathbf{q}} + \mathbf{G} + \mathbf{F}$). To obtain the approximation error, we recall the real system dynamics (1) and transform it into the following form using \mathbf{M}_n [10]:

$$\mathbf{M}_n\ddot{\mathbf{q}} + \tilde{\mathbf{M}}\dot{\mathbf{q}} + \mathbf{N} + \boldsymbol{\tau}_d = \boldsymbol{\tau}, \quad (20)$$

where $\tilde{\mathbf{M}} := \mathbf{M} - \mathbf{M}_n$ is the modeling error of \mathbf{M} .

From (19) and (20), the system dynamics approximation error $\boldsymbol{\gamma} := \ddot{\hat{\mathbf{q}}}_{\text{nom}} - \ddot{\mathbf{q}}$ for the nominal model case is obtained:

$$\boldsymbol{\gamma} = \mathbf{M}_n^{-1} \left(\tilde{\mathbf{M}}\dot{\mathbf{q}} + \tilde{\mathbf{N}} + \boldsymbol{\tau}_d \right), \quad (21)$$

where $\tilde{\mathbf{N}} := \mathbf{N} - \mathbf{N}_n$ is the modeling error of \mathbf{N} .

If the nominal model is available, the approximated equation of motion of the i -th task is obtained according to (5):

$$\ddot{\hat{\mathbf{x}}}_{i,\text{nom}} = \dot{\mathbf{J}}_i\dot{\mathbf{q}} - \mathbf{J}_i\mathbf{M}_n^{-1}\mathbf{N}_n + \mathbf{J}_i\mathbf{M}_n^{-1}\boldsymbol{\tau}, \quad (22)$$

where $\hat{\mathbf{x}}_{i,\text{nom}}$ is the approximation generated by the nominal model. Correspondingly, according to (5) and (20), the equation of motion of the i -th task is rewritten as

$$\ddot{\mathbf{x}}_i = \dot{\mathbf{J}}_i\dot{\mathbf{q}} - \mathbf{J}_i\mathbf{M}_n^{-1} \left(\tilde{\mathbf{M}}\dot{\mathbf{q}} + \mathbf{N} + \boldsymbol{\tau}_d \right) + \mathbf{J}_i\mathbf{M}_n^{-1}\boldsymbol{\tau}. \quad (23)$$

From (22) with (23), the equation of motion approximation error $\boldsymbol{\beta}_i := \ddot{\hat{\mathbf{x}}}_{i,\text{nom}} - \ddot{\mathbf{x}}_i$ is obtained as

$$\boldsymbol{\beta}_i = \mathbf{J}_i\hat{\mathbf{M}}^{-1} \left(\tilde{\mathbf{M}}\dot{\mathbf{q}} + \tilde{\mathbf{N}} + \boldsymbol{\tau}_d \right). \quad (24)$$

Next, approximation errors for incremental system cases, (13) and (17)), are derived. Comparing (16) with (17), one observes that the system dynamics approximation error for the incremental system case is $\boldsymbol{\delta}$ defined as $\boldsymbol{\delta} := \bar{\mathbf{M}}^{-1}\boldsymbol{\varepsilon}_q$:

$$\begin{aligned} \boldsymbol{\delta} = \bar{\mathbf{M}}^{-1} \left[(\mathbf{M} - \bar{\mathbf{M}})\ddot{\mathbf{q}} - (\mathbf{M}_{(t-L)} - \bar{\mathbf{M}})\ddot{\mathbf{q}}_{(t-L)} \right. \\ \left. + \boldsymbol{\varepsilon}_N + \boldsymbol{\tau}_d - (\boldsymbol{\tau}_d)_{(t-L)} \right], \end{aligned} \quad (25)$$

where $\boldsymbol{\varepsilon}_N := \mathbf{N} - \mathbf{N}_{(t-L)}$ is the TDE error of \mathbf{N} .

Then, comparing (12) with (13), one observes that the approximation error for the incremental system case is the TDE error $\boldsymbol{\varepsilon}_x$. Besides, for a sufficiently small sampling period, it holds that

$$\dot{\mathbf{J}}_i\dot{\mathbf{q}} \approx \left(\dot{\mathbf{J}}_i\dot{\mathbf{q}} \right)_{(t-L)}. \quad (26)$$

Thus, combining (7), (9), (10), (11), (25), and (26) results in

$$\begin{aligned} \boldsymbol{\varepsilon}_x = \mathbf{H}_i - \hat{\mathbf{H}}_i \stackrel{(9),(10),(11)}{\approx} \mathbf{H}_i - (\mathbf{H}_i)_{(t-L)} \stackrel{(7),(26)}{\approx} \mathbf{J}_i\boldsymbol{\delta} \\ \stackrel{(25)}{\approx} \mathbf{J}_i\bar{\mathbf{M}}^{-1} \left[(\mathbf{M} - \bar{\mathbf{M}})\ddot{\mathbf{q}} - (\mathbf{M}_{(t-L)} - \bar{\mathbf{M}})\ddot{\mathbf{q}}_{(t-L)} \right. \\ \left. + \boldsymbol{\varepsilon}_N + \boldsymbol{\tau}_d - (\boldsymbol{\tau}_d)_{(t-L)} \right], \end{aligned} \quad (27)$$

For the TDE approximation (cf. (25) and (27)), we learn that, if the delay time L (sampling period T_s) is sufficiently small, the inertia modeling error $(\mathbf{M} - \bar{\mathbf{M}})\ddot{\mathbf{q}}$ and disturbance $\boldsymbol{\tau}_d$ are compensated by their time-delayed values $(\mathbf{M}_{(t-L)} - \bar{\mathbf{M}})\ddot{\mathbf{q}}_{(t-L)}$ and $(\boldsymbol{\tau}_d)_{(t-L)}$ effectively, and the smaller the sampling period, the smaller is the TDE error. For the nominal model, there are no terms to compensate for modeling errors and disturbances. Besides, according to the continuity property [56], $\boldsymbol{\varepsilon}_N$ is small enough, and then its value will be less than that of $\tilde{\mathbf{N}}$, if the sampling rate is sufficiently high. Therefore, for a sufficiently small sampling period, the incremental systems derived using TDE exhibit high approximation accuracy and show strong robustness against modeling errors and disturbances.

So far, we have derived incremental systems which will be used to determine state predictions. Further, we verified their

high approximation accuracy if a sufficiently small sampling period is employed. Next, the task-prioritized MPC scheme will be developed with robust incremental systems.

Remark 5 (Effects of the Accuracy of Actuators and Sensors): In practice, there is a difference between the control law τ and the real supplied torque τ_r due to motor dynamics and accuracy. In Sec. II-A, this difference is considered as one part of τ_d . From (25) and (27), we learn that if L is sufficiently small, τ_d is compensated by its time-delayed value $(\tau_d)_{(t-L)}$ effectively. Hence, the effects of actuator dynamics and accuracy on control performance are limited. Besides, since incremental systems are constructed using time derivatives of states, the effect of the measurement error from the sensor resolution is amplified by numerical derivatives, thus, noisy responses can be triggered if the resolution is coarse. In Sec. VI, simulations are implemented to quantitatively investigate the allowable measurement resolution for HIMPC.

Remark 6 (Allowable Sampling Periods): The control accuracy is in line with the sampling rate, which is generally up to 1 kHz or even higher for high-precision robotics systems. Moreover, to guarantee the required approximation accuracy of the incremental systems, the sampling period has to be chosen sufficiently small. While there are, unfortunately, no systematic methods to determine the maximum allowable sampling period, one can find in [10]–[12], [25]–[27], TDE methods have been successfully demonstrated high-accuracy approximation capability with sampling periods of 1 ms and 2 ms for robot manipulator control. Accordingly, 1 ms sampling period is considered throughout the paper.

IV. HIERARCHICAL INCREMENTAL MPC

In this section, the task-prioritized control scheme, hierarchical incremental MPC (HIMPC), is developed as a series of constrained OCPs, where appropriate constraints are used to enforce the task hierarchy. Finally, the hierarchical and recursive feasibility of the proposed HIMPC is shown.

A. Method

In this subsection, at first, the stage cost for one task is introduced as the basis for developing the series of constrained OCPs. The stage cost is designed to take into account the target motion dynamics (4) as well as controller oscillations, energy efficiency, and actuator protection. Thus, the predicted motion dynamics error and the control signal are considered in the stage cost.

In accordance with (4), the predicted motion dynamics error $\mathbf{e}(\bar{\mathbf{x}}_{i,k+j+1|k})$ at time k is defined as

$$\begin{aligned} \mathbf{e}(\bar{\mathbf{x}}_{i,k+j+1|k}) &:= \ddot{\bar{\mathbf{x}}}_{i,k+j+1|k} + \mathbf{K}_V \dot{\bar{\mathbf{x}}}_{i,k+j+1|k} + \mathbf{K}_P \bar{\mathbf{x}}_{i,k+j+1|k} \\ &= \bar{\mathbf{K}}_i \bar{\mathbf{x}}_{i,k+j+1|k} - \mathbf{K}_i \bar{\mathbf{x}}_{id}(k+j+1). \end{aligned} \quad (28)$$

Here $\bar{\mathbf{x}}_{i,k+j+1|k} := \text{col}(\bar{\mathbf{x}}_{i,k+j+1|k}, \dot{\bar{\mathbf{x}}}_{i,k+j+1|k})$, $\bar{\mathbf{x}}_{i,k+j|k} := \mathbf{x}_{i,k+j+1|k} - \mathbf{x}_{id}(k+j+1)$, $\bar{\mathbf{K}}_i := [\mathbf{K}_i, \mathbf{O}]$, $\mathbf{K}_i := [\mathbf{K}_P, \mathbf{K}_V, \mathbf{I}]$, $\bar{\mathbf{x}}_{id} := \text{col}(\mathbf{x}_{id}, \dot{\mathbf{x}}_{id}, \ddot{\mathbf{x}}_{id})$, $j \in \mathbb{I}_{[0, N-1]}$ is an intermediate variable for time instance (the same hereinafter), N is the length of the prediction horizon, and $\bullet_{k+j|k}$ stands

for the predicted variables, in particular, $\bullet_{k|k} := \bullet(k)$, and $\mathbf{x}_{i,k+j+1|k}$ is calculated using the discrete-time incremental system (15). That is, for all $j \in \mathbb{I}_{[0, N-1]}$,

$$\bar{\mathbf{x}}_{i,k+j+1|k} = \mathbf{A}_1 \bar{\mathbf{x}}_{i,k+j|k} + \hat{\mathbf{B}}_{1i}^j \Delta \tau_{k+j|k} \quad (29)$$

with $\bar{\mathbf{x}}_{i,k|k} := \text{col}(\bar{\mathbf{x}}_{i,k|k}, \dot{\bar{\mathbf{x}}}_{i,k-1|k})$ and $\bar{\mathbf{x}}_{i,k-1|k} := \bar{\mathbf{x}}_i(k-1)$. In (14), the matrix \mathbf{B}_{1i} is derived using the Jacobian \mathbf{J}_i which is a function of the joint position. Similar to the approach in [31], we simplify calculation by approximating \mathbf{J}_i and \mathbf{B}_{1i} , and approximations over the prediction horizon, $\hat{\mathbf{J}}_{i,k}^j$ and $\hat{\mathbf{B}}_{1i}^j$, are obtained using the optimal joint state predictions at time $k-1$. The optimal joint state prediction $\mathbf{q}_{k+j|k-1}^*$ is calculated by the incremental system (18) with the optimal control input sequence $\Delta \bar{\tau}_{k-1}^* := [\Delta \tau_{k-1|k-1}^*, \dots, \Delta \tau_{k+N|k-1}^*]$, i.e.,

$$\bar{\mathbf{q}}_{k+j|k-1}^* = \mathbf{A}_2 \bar{\mathbf{q}}_{k-1+j|k-1}^* + \mathbf{B}_2 \Delta \tau_{k-1+j|k-1}^*. \quad (30)$$

Here, the incremental control signal $\Delta \tau$ is considered in the stage cost, since the equations of motion of tasks and system dynamics are all approximated by incremental systems. Thus, at time k , the following stage cost for task i is defined:

$$\ell(\bar{\mathbf{x}}_{i,k+j+1|k}, \Delta \tau_{k+j|k}) = \|\mathbf{e}(\bar{\mathbf{x}}_{i,k+j+1|k})\|_{\mathbf{Q}_i}^2 + \|\Delta \tau_{k+j|k}\|_{\mathbf{R}_i}^2 \quad (31)$$

where $\mathbf{Q}_i \succ 0$ and $\mathbf{R}_i \succ 0$ are weighting matrices.

Now the stage costs of all tasks are combined and deliver the proposed HIMPC framework where a constrained OCP for each task is introduced, starting with the task 1 as the highest-priority task. The task hierarchy will be achieved by constraints to the OCPs of lower-priority tasks.

Defining an admissible control input sequence $\Delta \bar{\tau}_{1,k} := [\Delta \tau_{1,k|k}, \dots, \Delta \tau_{1,k+N-1|k}]$, the constrained OCP (**Problem 1**) for task 1 is introduced.

Problem 1:

$$\Delta \bar{\tau}_{1,k}^* = \arg \min_{\Delta \bar{\tau}_1} \sum_{j=0}^{N-1} \ell(\bar{\mathbf{x}}_{1,k+j+1|k}, \Delta \tau_{1,k+j|k}) \quad (32a)$$

s.t.

$$\bar{\mathbf{x}}_{1,k+j+1|k} = \mathbf{A}_1 \bar{\mathbf{x}}_{1,k+j|k} + \hat{\mathbf{B}}_{11}^j \Delta \tau_{1,k+j|k} \quad (32b)$$

$$\bar{\mathbf{q}}_{1,k+j+1|k} = \mathbf{A}_2 \bar{\mathbf{q}}_{1,k+j|k} + \mathbf{B}_2 \Delta \tau_{1,k+j|k} \quad (32c)$$

$$\mathbf{q}_{\min} \leq \mathbf{q}_{1,k+j+1|k} \leq \mathbf{q}_{\max} \quad (32d)$$

$$\dot{\mathbf{q}}_{\min} \leq \dot{\mathbf{q}}_{1,k+j+1|k} \leq \dot{\mathbf{q}}_{\max} \quad (32e)$$

$$\tau_{\min} \leq \tau_0 + \sum_{s=0}^j \Delta \tau_{1,k+s|k} \leq \tau_{\max} \quad (32f)$$

where $\Delta \bar{\tau}_{1,k}^*$ is the optimal control input sequence, $\bar{\mathbf{q}}_{1,k+j+1|k}$ is the joint state prediction with $\Delta \tau_{1,k+j|k}$. Equality constraints (32b) and (32c) are employed to generate predictions $\bar{\mathbf{x}}_{1,k+j+1|k}$ and $\bar{\mathbf{q}}_{1,k+j+1|k}$ while constraints (32d)–(32f) are imposed on joint position, velocity, and torques, respectively. In case there is only one task, the first column of $\Delta \bar{\tau}_{1,k}^*$ is applied to the system directly. For multiple tasks, $\Delta \bar{\tau}_{1,k}^*$ is just an auxiliary signal to the constrained OCPs of lower-priority tasks while only task r , the lowest priority task, will finally deliver a control input that is applied to the system.

To achieve task hierarchy, relying on the *dynamic consistency* principle [5] (i.e., high-priority tasks will not be affected

by the torques designed for low-priority tasks), the following equality constraints will be imposed on control signals when **Problem** i ($i \in \mathbb{I}_{[2,r]}$) is formulated for task i : $\forall p \in \mathbb{I}_{[1,i-1]}$,

$$\hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau}_{p,k+j|k}^* = \hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau}_{i,k+j|k}. \quad (33)$$

Here, $\Delta \boldsymbol{\tau}_{p,k+j|k}^*$ and $\Delta \boldsymbol{\tau}_{i,k+j|k}$ are the optimal input and one admissible counterpart for the p -th and i -th layer constrained OCPs respectively, and $\hat{\mathbf{J}}_{p,k}^j$ is the approximated Jacobian for task p at time k and is calculated using $\mathbf{q}_{k+j|k-1}^*$. From (29) and (33), it is observed that the optimal motion of task p will not be disturbed by optimal inputs determined for lower-priority tasks. Here no null-space projections are used and thus matrix inversion to determine the prioritized Jacobian is not necessary. Thus, the algorithmic singularity is avoided.

With the task priority constraint (33), the following **Problem** i for task i ($i \in \mathbb{I}_{[2,r]}$) is introduced:

Problem i ($i \in \mathbb{I}_{[2,r]}$):

$$\Delta \bar{\boldsymbol{\tau}}_{i,k}^* = \arg \min_{\Delta \bar{\boldsymbol{\tau}}_i} \sum_{j=0}^{N-1} \ell(\bar{\mathbf{x}}_{i,k+j+1|k}, \Delta \boldsymbol{\tau}_{i,k+j|k}) \quad (34a)$$

s.t.

$$\bar{\mathbf{x}}_{i,k+j+1|k} = \mathbf{A}_1 \bar{\mathbf{x}}_{i,k+j|k} + \hat{\mathbf{B}}_{1i}^j \Delta \boldsymbol{\tau}_{i,k+j|k} \quad (34b)$$

$$\bar{\mathbf{q}}_{i,k+j+1|k} = \mathbf{A}_2 \bar{\mathbf{q}}_{i,k+j|k} + \mathbf{B}_2 \Delta \boldsymbol{\tau}_{i,k+j|k} \quad (34c)$$

$$\mathbf{q}_{\min} \leq \mathbf{q}_{i,k+j+1|k} \leq \mathbf{q}_{\max} \quad (34d)$$

$$\dot{\mathbf{q}}_{\min} \leq \dot{\mathbf{q}}_{i,k+j+1|k} \leq \dot{\mathbf{q}}_{\max} \quad (34e)$$

$$\boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau}_0 + \sum_{s=0}^j \Delta \boldsymbol{\tau}_{i,k+s|k} \leq \boldsymbol{\tau}_{\max} \quad (34f)$$

$$\hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau}_{p,k+j|k}^* = \hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta \boldsymbol{\tau}_{i,k+j|k}, \forall p \in \mathbb{I}_{[1,i-1]} \quad (34g)$$

where $\bar{\mathbf{q}}_{i,k+j+1|k}$ is the joint state prediction under the action of $\Delta \boldsymbol{\tau}_{i,k+j|k}$. The difference between **Problem** 1 and **Problem** i ($i \in \mathbb{I}_{[2,r]}$) is the series of equality constraints (34g), which allows to enforce the task hierarchy. Because of the approximated Jacobian $\hat{\mathbf{J}}_{p,k}^j$, the equality constraint (34g) is an affine linear system, which is also the basis for analyzing the existence and uniqueness of the solution to **Problem** i .

The optimal control input sequence $\Delta \bar{\boldsymbol{\tau}}_{r,k}^*$ is obtained after a series of constrained OCPs (from **Problem** 1 to **Problem** r) is solved. $\Delta \bar{\boldsymbol{\tau}}_{r,k}^*$ is not only the optimal solution to **Problem** r , but it also guarantees that optimal motion of tasks 1 to $(r-1)$ are not affected owing to the equality constraint (34g). Thus, $\Delta \bar{\boldsymbol{\tau}}_{r,k}^*$ is considered as the optimal control input sequence to the proposed HIMPC. Combined with the recent control input $\boldsymbol{\tau}_0$, the first column of $\Delta \bar{\boldsymbol{\tau}}_{r,k}^*$ is applied to the system, i.e., $\boldsymbol{\tau}(k+1) = \boldsymbol{\tau}_0 + \Delta \bar{\boldsymbol{\tau}}_{r,k}^*$.

Note that the developed controller is also suitable for other fixed-base robotic systems. Further, it also can be extended to mobile robotic systems when kinematics constraints [44] are considered in the constrained OCP.

As a summary, Fig. 1 visualizes the structure and the control signal flow of the proposed HIMPC.

Remark 7 (Supplementary Notes to the Approximation in (29) and (33)): With (18) (or (34c)), over the prediction horizon \mathbf{J}_i can also be updated using the current joint state

predictions. However in this case, (29) and (33) will be nonlinear equations, and consequently a nonlinear MPC framework is obtained, resulting in higher computational complexity. For simplicity, in this article over the prediction horizon \mathbf{J}_i and \mathbf{B}_{1i} are approximated, and the approximations $\hat{\mathbf{J}}_{i,k}^j$ and $\hat{\mathbf{B}}_{1i}^j$ are determined using previous joint state predictions $\mathbf{q}_{k+j|k-1}^*$ before solving the constrained OCPs. In this way, (29) and (33) used to generate predictions of $\bar{\mathbf{x}}_i$ and guarantee task hierarchy, which are originally nonlinear equations, are approximated by canonical linear equations. Besides, $\Delta \bar{\boldsymbol{\tau}}_r^*$ is the optimal control input sequence to the proposed hierarchical control scheme, and thus $\Delta \bar{\boldsymbol{\tau}}_{r,k-1}^*$ will be used in (30) to generate $\mathbf{q}_{k+j|k-1}^*$.

Remark 8 (Serial Connection of Constrained OCPs): The computational complexity of HIMPC increases as the number of tasks increases. For a task hierarchy with n tasks, there will be n constrained OCPs to be solved. In this article, the constrained OCPs are solved by a QP solver which is possible efficiently. Nevertheless, the computing time increases with an increasing number of tasks because of the serial connection of constrained OCPs. Fortunately, this increase is linear. In [2], [57], dedicated solvers were developed to speed up the calculation of task-prioritized programming problems. However, the null-space projection idea is adopted and it has the risk of algorithmic singularity. To expedite calculations, one possible alternative is to select shorter prediction horizons for low-priority tasks on the basis of stability. The task hierarchy will be still guaranteed although the optimality of low-priority tasks will deteriorate.

B. Analysis

The developed HIMPC framework is only logical if each constrained OCP is feasible for every time, which means the set \mathcal{D}_i ($1 \leq i \leq r$) of admissible input trajectories $\Delta \bar{\boldsymbol{\tau}}_{i,k}$ for **Problem** i is non-empty. In this subsection, this hierarchical recursive feasibility will be analyzed first.

1) *Hierarchical Feasibility:* Due to the hierarchical structure and the equality constraint (34g), admissible sets of low-priority constrained OCPs are affected by those of high-priority ones. To analyze the feasibility of this hierarchical MPC, the definition of hierarchical feasibility is introduced.

Definition 1 (Hierarchical Feasibility, [58]): The hierarchical MPC (a series of constrained OCPs ordered hierarchically) admits hierarchical feasibility at time k if the feasibility of the i -th constrained OCP implies the feasibility of the $(i+1)$ -th constrained OCP for all $i \in \mathbb{I}_{[1,r-1]}$.

To prove the hierarchical feasibility of HIMPC, we first show the existence and uniqueness of solutions to **Problem** i using convex optimization theory. Notice that **Problem** i ($1 \leq i \leq r$) is a standard quadratic program (QP):

$$\min_{\mathbf{U}_i} \mathbf{U}_i^\top \mathbf{Q}_i \mathbf{U}_i + \mathbf{U}_i^\top \mathcal{L}_i \quad (35a)$$

s.t.

$$\mathcal{G}_{i1} = \mathbf{C}_{i1} \mathbf{U}_i + \mathbf{D}_{i1} \leq \mathbf{0} \quad (35b)$$

$$\mathcal{G}_{i2} = \mathbf{C}_{i2} \mathbf{U}_i + \mathbf{D}_{i2} \leq \mathbf{0} \quad (35c)$$

$$\mathcal{G}_{i3} = \mathbf{C}_{i3} \mathbf{U}_i + \mathbf{D}_{i3} \leq \mathbf{0} \quad (35d)$$

$$\mathcal{H}_{ip}(\mathbf{U}_i) = \mathbf{a}_{ip}^\top \mathbf{U}_i + \mathbf{b}_{ip} = \mathbf{0}, \forall p \in \mathbb{I}_{[1,i-1]} \quad (35e)$$

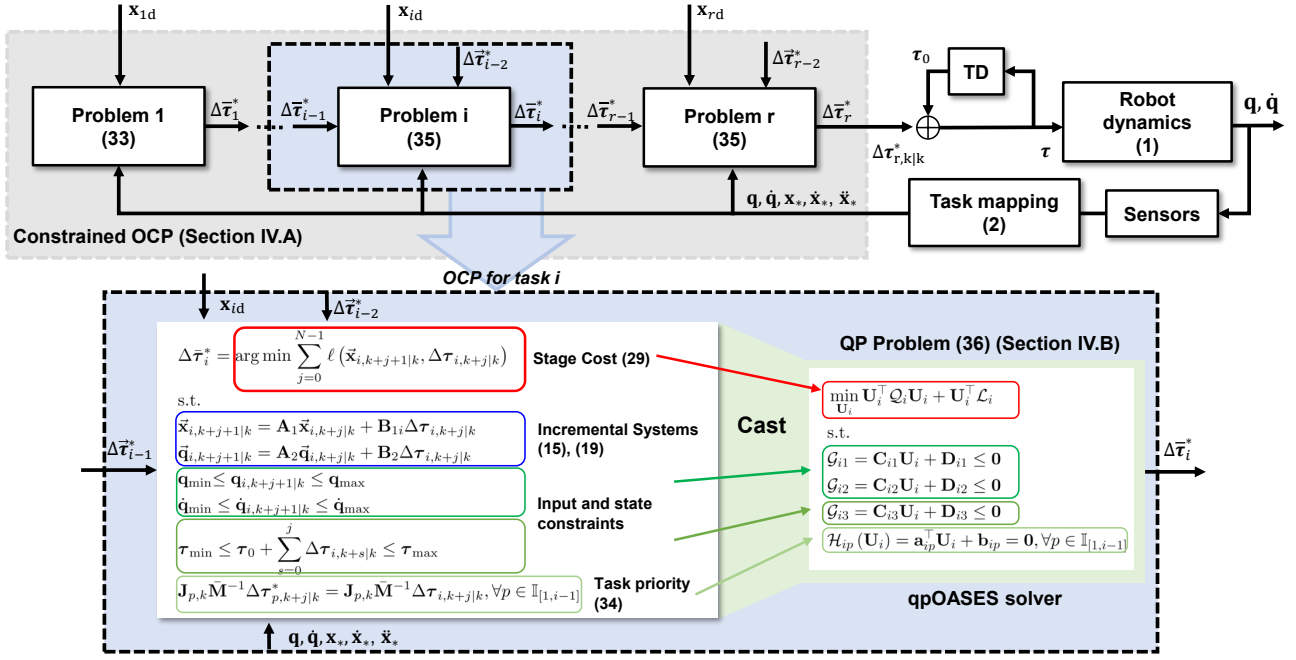


Fig. 1. The control structure of the HIMPC, where “TD” denotes time delay of one sampling period, and $\Delta\bar{\tau}_{i,k}^* := [\Delta\bar{\tau}_{1,k}^*, \dots, \Delta\bar{\tau}_{i,k}^*]$. Problems are solved sequentially following the task priority order, and the optimal control law can only be obtained after all problems are solved.

where $\mathbf{U}_i := \text{col}(\Delta\tau_{i,k|k}, \dots, \Delta\tau_{i,k+N-1|k})$ and (35b)-(35e) are joint position, velocity, input, and task priority constraints in (34), respectively. The equality constraints generating predictions (34b) and (34c) are omitted here since the predictions of task coordinates \bar{x}_i and joint angles \bar{q} can be expressed by functions of the input \mathbf{U}_i and recent states $\bar{x}_i(k)$ and $\bar{q}(k)$. For detailed expressions of Hessian matrix \mathcal{Q}_i , gradient vector \mathcal{L}_i , and other parameters in (35), refer to Appendix A. Note that for $i = 1$, the task priority constraint (35e) does not exist.

To examine the existence and uniqueness of solutions, the following lemma about the Hessian matrix \mathcal{Q}_i is given.

Lemma 1: The Hessian matrix \mathcal{Q}_i is positive definite.

Proof: As shown in Appendix A, $\mathcal{Q}_i := \bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i \bar{\mathbf{B}}_{1i} + \bar{\mathbf{R}}_i$, where $\bar{\mathbf{Q}}_i := \text{diag}(\bar{\mathbf{K}}_i^T \mathbf{Q}_i \bar{\mathbf{K}}_i, \dots, \bar{\mathbf{K}}_i^T \mathbf{Q}_i \bar{\mathbf{K}}_i)$ is a positive semi-definite matrix since $\mathbf{Q}_i \succ 0$. Then for any vector $\mathbf{X} \in \mathbb{R}^{nN \times 1}$, one obtains that

$$\mathbf{X}^T \bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i \bar{\mathbf{B}}_{1i} \mathbf{X} = (\bar{\mathbf{B}}_{1i} \mathbf{X})^T \bar{\mathbf{Q}}_i (\bar{\mathbf{B}}_{1i} \mathbf{X}) \geq 0.$$

Thus, $\bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i \bar{\mathbf{B}}_{1i}$ is also positive semi-definite. Besides, $\bar{\mathbf{R}}_i := \text{diag}(\bar{\mathbf{R}}_i, \dots, \bar{\mathbf{R}}_i) \succ 0$ since $\mathbf{R}_i \succ 0$. Therefore, $\mathcal{Q}_i := \bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i \bar{\mathbf{B}}_{1i} + \bar{\mathbf{R}}_i$ is positive definite, i.e., $\mathcal{Q}_i \succ 0$. ■

Then, assuming \mathcal{D}_i is nonempty, we present a theorem about the properties of the solution to **Problem i** with the positive definite Hessian matrix \mathcal{Q}_i .

Theorem 1 (Existence and Uniqueness of the Solution, [36], [59]): Given that \mathcal{D}_i is non-empty, the solution to **Problem i** exists and is unique.

Proof: Since \mathcal{D}_i , the set for the admissible input trajectory $\Delta\bar{\tau}_i$, is non-empty, the admissible set \mathcal{U}_i for the argument \mathbf{U}_i of the QP (35) is also non-empty. In the following, we will use the convexity of the QP (35) to verify the existence and uniqueness of the solution to **Problem i** .

According to (35), the admissible set \mathcal{U}_i for \mathbf{U}_i is

$$\mathcal{U}_i = \left(\bigcap_{p=1}^{i-1} \text{dom } \mathcal{H}_{ip} \right) \cap \left(\bigcap_{l=0}^3 \text{dom } \mathcal{G}_{il} \right), \quad (36)$$

i.e., the admissible set \mathcal{U}_i is composed of $i - 1$ hyperplanes $\{\mathbf{U}_i : \mathbf{a}_{ip}^T \mathbf{U}_i + \mathbf{b}_{ip} = 0\}$ ($p \in \mathbb{I}_{[0,i-1]}$) and 3 sublevel sets $\{\mathbf{U}_i : \mathcal{G}_{il} \leq 0\}$ ($l=1, 2, 3$). The equality constraint functions $\mathcal{H}_{ip}(\mathbf{U}_i)$ ($p \in \mathbb{I}_{[1,i-1]}$) are linear and affine, and inequality constraint functions \mathcal{G}_{i1} , \mathcal{G}_{i2} , and \mathcal{G}_{i3} are all linear and convex. Thus, the admissible set \mathcal{U}_i is convex. In addition, the cost function $\mathbf{U}_i^T \mathcal{Q}_i \mathbf{U}_i + \mathbf{U}_i^T \mathcal{L}_i$ is quadratic and convex. Thus, (35) is a convex optimization problem and solutions exist.

Moreover, since the Hessian matrix \mathcal{Q}_i is positive definite as shown in *Lemma 1*, the cost function $\mathbf{U}_i^T \mathcal{Q}_i \mathbf{U}_i + \mathbf{U}_i^T \mathcal{L}_i$ is thus strictly convex. Therefore, the solution to **Problem i** is unique given that the admissible set \mathcal{D}_i is non-empty. ■

Finally, the hierarchical feasibility of HIMPC is proven.

Theorem 2 (Hierarchical Feasibility of HIMPC): The proposed HIMPC admits hierarchical feasibility at each time k .

Proof: Assume **Problem i** is feasible. Then the admissible set \mathcal{D}_i is non-empty, and, in accordance with *Theorem 1*, there exists an optimal control input sequence $\Delta\bar{\tau}_{i,k}^*$ such that the state, input, and task priority constraints (task priority constraints are not required when $i = 1$)

$$\hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta\tau_{p,k+j|k}^* = \hat{\mathbf{J}}_{p,k}^j \bar{\mathbf{M}}^{-1} \Delta\tau_{i,k+j|k}^*$$

are not violated for $p \in \mathbb{I}_{[1,i-1]}$ and $j \in \mathbb{I}_{[0,N-1]}$.

Next, the feasibility of **Problem $(i+1)$** will be investigated by inspecting whether $\Delta\bar{\tau}_{i,k}^*$ is an admissible control input sequence also for **Problem $(i+1)$** . Substituting $\Delta\bar{\tau}_{i,k}^*$ into **Problem $(i+1)$** , obviously the state and input constraints, as well as priority constraints for $p \in \mathbb{I}_{[1,i-1]}$ are not violated

because these constraints are identical to that in **Problem** i . Besides, the priority constraint when $p = i$ is also fulfilled, i.e., if $\Delta\bar{\tau}_{i+1,k} = \Delta\bar{\tau}_{i,k}^*$, then

$$\hat{\mathbf{J}}_{i,k}^j \bar{\mathbf{M}}^{-1} \Delta\tau_{i,k+j|k}^* \equiv \hat{\mathbf{J}}_{i,k}^j \bar{\mathbf{M}}^{-1} \Delta\tau_{i+1,k+j|k}, \forall j \in \mathbb{I}_{[0,N-1]}.$$

Hence, no constraints are violated and $\Delta\bar{\tau}_{i,k}^*$ is a feasible control input sequence for **Problem** $(i+1)$. We conclude that **Problem** $(i+1)$ is feasible if **Problem** i is feasible, i.e., the proposed HIMPC is hierarchically feasible. ■

Assume **Problem** 1 is feasible at time k . By induction, we see that **Problems** 2 to r are also feasible at time k . Therefore, from *Theorem* 2, it is concluded that the admissible set \mathcal{D}_i ($1 \leq i \leq r$) is non-empty if **Problem** 1 is feasible.

In accordance with *Theorems* 1 and 2, we conclude that the solution to HIMPC exists and is unique if **Problem** 1 is feasible. In the sequel, recursive feasibility [34], [35] of **Problem** 1 is demonstrated.

2) *Recursive Feasibility*: In this part, recursive feasibility of **Problem** 1 in a local region around the reachable reference trajectory is demonstrated. First, similar to [34], [35], [45], the definition of the reachable reference trajectory is introduced.

Definition 2 (Reachable Reference Trajectory, [34], [35], [45]): The reference trajectory \mathbf{x}_{1d} is reachable if it implies the possible joint variable $\bar{\mathbf{q}}^P$ (satisfying $\mathbf{x}_{1d} = \mathbf{f}(\bar{\mathbf{q}}^P)$ and $\dot{\mathbf{x}}_{1d} = \mathbf{J}_1(\bar{\mathbf{q}}^P)\dot{\bar{\mathbf{q}}^P}$) and the corresponding controller \mathbf{u}^P lie in tightened constraint sets $\bar{\mathcal{Q}}$ and $\bar{\mathcal{U}}$ respectively, where $\bar{\mathcal{Q}} \oplus \mathbb{C}_s^{4n} \subseteq \mathcal{Q}$, $\bar{\mathcal{U}} \oplus \mathbb{C}_r^{4n} \subseteq \mathcal{U}$, $\mathcal{Q} := \{\bar{\mathbf{q}} : \mathbf{q}_{\min} \leq \bar{\mathbf{q}} \leq \mathbf{q}_{\max}, \dot{\mathbf{q}}_{\min} \leq \dot{\bar{\mathbf{q}}} \leq \dot{\mathbf{q}}_{\max}\}$, $\mathcal{U} := \{\mathbf{u} : \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max}\}$, and \oplus is the Minikowski sum.

In accordance with the definition of the reachable reference trajectory (*Definition* 2) and *Lemma* 4 in [45], we obtain that there exists a constant $V_{\max} \in \mathbb{R}_{>0}$ such that optimal solutions from $\bar{\mathbf{x}}_1(k)$ ($\bar{\mathbf{x}}_1(k) \in \mathbb{D}$, $\mathbb{D} := \{\bar{\mathbf{x}}_1, V_N(\bar{\mathbf{x}}_1(k), k) \leq V_{\max}\}$) also satisfy $\bar{\mathbf{q}}_{k+j|k}^* \in \bar{\mathcal{Q}}'$ and $\mathbf{u}_{k+j-1|k}^* \in \bar{\mathcal{U}}'$ for all $j \in \mathbb{I}_{[1,N]}$, where $V_N(\bar{\mathbf{x}}_1(k), k) := \min_{\Delta\bar{\tau}_1} \sum_{j=0}^{N-1} \ell(\bar{\mathbf{x}}_{1,k+j+1|k}, \Delta\tau_{1,k+j|k})$ (s.t. constraints (32b)-(32f)) is the value function, both $\bar{\mathcal{Q}}'$ and $\bar{\mathcal{U}}'$ are tightened constraint sets, i.e., $\bar{\mathcal{Q}}' \oplus \mathbb{C}_{s'}^{4n} \subseteq \mathcal{Q}$, $\bar{\mathcal{U}}' \oplus \mathbb{C}_{r'}^{4n} \subseteq \mathcal{U}$, s' and r' are positive scalars. In other words, the reachable reference trajectory can be tracked and lies strictly in tightened constraint sets.

Then, the recursive feasibility of MPC in a local region \mathbb{D} around the reachable reference trajectory will be demonstrated in *Theorem* 3, and the main work is to prove the regionally decreasing property of $V_N(\bar{\mathbf{x}}(k), k)$.

Theorem 3 (Recursive Feasibility): **Problem** 1 admits recursive feasibility in a local region around the reachable reference trajectory.

Proof: Because of the task-priority constraint (29), the optimal state prediction in task 1, $\bar{\mathbf{x}}_{1,k+1|k}^{1*}$, will not be disturbed by low-priority tasks, i.e., $\bar{\mathbf{x}}_{1,k+1|k}^{1*} = \bar{\mathbf{x}}_{1,k+1|k}^*$. Selecting $V_N(\bar{\mathbf{x}}_{1,k+1|k}^*, k+1)$ as the auxiliary value function, and then according to feasibility analysis for MPC without terminal ingredients in [34], [35], [45], we obtain that for a sufficient long prediction horizon N ,

$$V_N(\bar{\mathbf{x}}_{1,k+1|k}^*, k+1) - V_N(\bar{\mathbf{x}}(k), k) \leq -\alpha_V(V_N(\bar{\mathbf{x}}(k), k)), \quad (37)$$

where $\alpha_V(\cdot) \in \mathcal{K}_\infty$, $\alpha_V(\cdot) \leq \text{id}(\cdot)$. Besides, according to (12), we have $\|\bar{\mathbf{x}}_{1,k+1|k}^* - \bar{\mathbf{x}}_1(k+1)\| \leq \bar{\epsilon}$ with $\bar{\epsilon}$ being a bounded parameter w.r.t. the TDE error.

Next, according to the Lipschitz continuity property of $V_N(\bar{\mathbf{x}}(k), k)$ [45], there exists a constant $K_\ell \in \mathbb{R}_{>0}$ such that

$$V_N(\bar{\mathbf{x}}_1(k+1), k+1) - V_N(\bar{\mathbf{x}}_{1,k+1|k}^*, k+1) \leq K_\ell \bar{\epsilon}. \quad (38)$$

In combination of (37) and (38), we obtain the regionally decreasing property of $V_N(\bar{\mathbf{x}}_1(k), k)$, i.e.,

$$V_N(\bar{\mathbf{x}}_1(k+1), k+1) - V_N(\bar{\mathbf{x}}_1(k), k) \leq -\alpha_V(V_N(\bar{\mathbf{x}}_1(k), k)) + \epsilon_N, \quad (39)$$

where $\epsilon_N := K_\ell \bar{\epsilon}$ is the *cumulative error bound*.

Finally, based on the regionally decreasing property of $V_N(\bar{\mathbf{x}}(k), k)$, the recursive feasibility of **Problem** 1 will be verified. Assuming the TDE error is sufficiently small ($\bar{\epsilon} \leq \alpha_V(V_{\max})/K_\ell$ and $\epsilon_N \leq \alpha_V(V_{\max})$), $\bar{\mathbf{x}}_1(k) \in \mathbb{D}$, one has

$$\begin{aligned} V_N(\bar{\mathbf{x}}_1(k+1), k+1) &\leq (\text{id} - \alpha_V)(V_N(\bar{\mathbf{x}}_1(k), k)) + \epsilon_N \\ &\leq (\text{id} - \alpha_V)(V_{\max}) + \alpha_V(V_{\max}) \leq V_{\max}. \end{aligned} \quad (40)$$

Thus, if $\bar{\mathbf{x}}_1(k) \in \mathbb{D}$, then $\bar{\mathbf{x}}_1(k+1) \in \mathbb{D}$. Using induction, it is shown that $V_N(\bar{\mathbf{x}}_1(k+j), k+j) \leq V_{\max}$ for all $j \in \mathbb{I}_{>0}$. In other words, \mathbb{D} is a positive invariant set. Therefore, $V_N(\bar{\mathbf{x}}_1(k), k) \leq V_{\max}$ holds recursively. According to the definition of the reachable reference trajectory, when $\bar{\mathbf{x}}_1 \in \mathbb{D}$ the input and state constraints are satisfied. In other words, the recursive feasibility of **Problem** 1 in a local region around the reachable reference trajectory is demonstrated. ■

Remark 9 (How to Determine the Prediction Horizon): For MPC without terminal ingredients, a sufficiently large prediction horizon is required to guarantee feasibility [34], [35], [45]. However, in the context of the robotic system, the sampling period is 1 ms. It is challenging to solve OCPs with large prediction horizons in one sampling period. In [34], [35], [45], the minimal prediction horizon is theoretically derived. Nevertheless, the prediction horizon is normally overestimated. In [35], [45], it is discussed that in practice a short prediction horizon is sufficient for local stability. Through deriving the required prediction horizon theoretically in [35], [45], it is concluded that control performance improves with larger prediction horizons. Thus, it provides a guideline to tune the prediction horizon, i.e., considering available computing resources, the prediction horizon is selected as large as possible.

C. Discussion

In this subsection, the advantages of the proposed HIMPC over state-of-the-art task-prioritized controllers are discussed. Besides, limitations of the proposed method are also analyzed.

1) Strengths of HIMPC:

Low Requirements for Modeling. In existing MPC based task-prioritized control schemes [17], [18], state predictions are generated using nominal models. This requires performing time-consuming identification of the mathematical model parameters of the plant. The proposed HIMPC employs incremental systems to obtain state predictions and thus it is not necessary to identify accurate models. For incremental systems

(cf. (13) and (17)), only $\bar{\mathbf{M}}$ is associated with the dynamics model because $\bar{\mathbf{M}}$ is selected such that $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| < 1$. As discussed in *Remark 4*, $\|\mathbf{I} - \mathbf{M}^{-1}\bar{\mathbf{M}}\| < 1$ holds for a sufficiently small positive \bar{M}_i even though the exact expression of the inertia matrix \mathbf{M} is unknown. Therefore, the proposed HIMPC has low requirements for modeling.

Robustness. As shown in Sec. III-B, the incremental systems show high approximation accuracy and strong robustness against inertia modeling errors and external disturbances. This is because the inertia modeling error and external disturbances are compensated by time-delayed signals. The robustness of this proposed method is also reflected in terms of *dynamic consistency*. In [16], a nominal \mathbf{M} is used to construct equality constraints (similar to (34g)), and it is obvious that *dynamic consistency* deteriorates when there are inertia modeling errors. In HIMPC, \mathbf{M} is replaced by a pre-determined $\bar{\mathbf{M}}$ in the task priority constraint (34g). Thus, enhanced *dynamic consistency* is received. This advantage of HIMPC is also visible in the simulations and experiments presented in Secs. V and VI.

Computational Efficiency. The TDE method not only improves robustness of the control scheme, but also simplifies equations of motion of tasks and system dynamics. As shown in (15) and (18), both, equations of motion and system dynamics, are approximated by linear systems, where online calculation of the complex nonlinear system dynamics terms is not required. As a result, each constrained OCP is a QP problem, and computational complexity will decrease dramatically, allowing for real-time control in milliseconds. In comparison, the MPC frameworks in [17], [18] are nonlinear since nonlinear equations are employed to generate state predictions. For a general NMPC scheme, heavy computational complexity restricts its real-time application [49].

Singularity Handling. Different from null-space projection based methods [4]–[12], we employ equality constraint (34g) to guarantee task hierarchy. No matrix inversion is required and algorithmic singularity is avoided. Besides, also kinematic singularity is avoided. In our approach, the cost function (31) is designed relying on target motion dynamics (4), where inverse calculation of the terms w.r.t. Jacobian matrices is not necessary. In contrast, e.g. in [16], Cartesian forces calculated using the equivalent Cartesian mass matrix $(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T)^{-1}$ are involved in stage costs, and kinematic singularity occurs when the Jacobian matrix loses its rank. Also, different from the approaches in [14], [15], Hessian matrices of the constrained OCPs are always positive definite (see *Lemma 1*). This allows to relax the common assumption that singularity-free tasks are defined and gives more flexibility in defining tasks. The simulations and experiments in Secs. V and VI will further discuss singularity handling. In [60], kinematic singularity is avoided by adding a manipulability constraint to the constrained OCP. Nevertheless, the constrained OCP in [60] can only be cast to a general NLP problem which is in addition non-convex because of the introduced nonlinear manipulability constraint. In this article no constraints are imposed in the considered OCP to avoid kinematic singularity and the constrained OCP is convex and is cast to a QP.

Weighting Matrices are Determined without Considering Reference Trajectories. The weighted control signal in (31)

not only allows for energy efficiency and attenuates controller oscillations, but also renders the Hessian matrix positive-definite. The result is similar to regularization [61], which is used to avoid kinematic singularity. We will verify that the incremental control structure exhibits superior control performance in terms of parameter adjustment. If, as an alternative, the torque $\boldsymbol{\tau}$ is considered, and the stage cost is designed: $\ell(\bar{\mathbf{x}}_{i,k+j+1|k}, \boldsymbol{\tau}_{k+j|k}) = \|\mathbf{e}(\bar{\mathbf{x}}_{i,k+j+1|k})\|_{\mathbf{Q}_i}^2 + \|\boldsymbol{\tau}_{k+j|k}\|_{\mathbf{R}_i}^2$. The term $\|\boldsymbol{\tau}_{k+j|k}\|_{\mathbf{R}_i}$ is used to achieve the Hessian matrix regularization. However, $\boldsymbol{\tau}$ is not always equal to zero even the system is in equilibrium, since non-zero $\boldsymbol{\tau}$ is required to maintain the configuration and compensate for gravity. To guarantee tracking accuracy, we need to scale the term $\|\boldsymbol{\tau}_{k+j|k}\|_{\mathbf{R}_i}$ to be roughly the same size to the tracking error term $\|\mathbf{e}(\bar{\mathbf{x}}_{i,k+j+1|k})\|_{\mathbf{Q}_i}$. Thus, \mathbf{R}_i should be tuned manually according to the target trajectory [61], and \mathbf{R}_i should be readjusted when the reference trajectory changes. This is the main limitation of the regularization method. In comparison, the standard cost function for tracking MPC (see (7) in [35]) involves the difference between the control signal and the reference counterpart, and the difference lies inside a small neighborhood around the origin for any reference signal. Thus, \mathbf{R}_i can be determined without considering reference signals. Unfortunately, for a redundant robot manipulator, the reference control signal is not determined priorly. This is the main reason for the limitation of the regularization method when it is applied to a redundant robotic system. In this article, the cost function (31) is similar to the standard stage cost for the tracking control scheme where the most recent control signal $\boldsymbol{\tau}_0$ is considered as the reference counterpart. Therefore, \mathbf{R}_i can be determined without considering reference trajectories, owing to the incremental control structure.

2) *Limitations of HIMPC—TDE Error:* In the constrained OCPs from Sec. IV-A, equality constraints (cf. (34g)) are imposed on control signals to achieve task hierarchy. However, the TDE error is ignored in the nominal incremental system (13). Consequently, task hierarchy will inevitably deteriorate, which is also visible in simulation and experimental results, see Secs. V and VI. In addition, given that the TDE error and measurement noise are ignored in this article, there are state prediction errors. As a result, the system suffers from the risk of state constraint violation. After all, huge deterioration of task hierarchy and strongly violating state constraints are avoided. This is because incremental systems exhibit high approximation accuracy as analyzed in Sec. III-B. A widely used method to deal with uncertainties and to guarantee strict state constraint satisfaction is tube-based MPC [62]. However, in this article, we control task-space variables and impose constraints on joint-space variables. In addition, the cost function does not involve terms w.r.t. joint-space variables since the considered robot manipulator is redundant and the reference signals for joint-space variables can not be determined priorly. As a result, the optimal initial joint-space variable can not be obtained and then tube-based MPC can not be implemented. A practical alternative is to use the data-driven method to estimate the uncertainty distribution and then employ stochastic MPC [63] to enhance the robustness of MPC in terms of strict state constraint satisfaction.

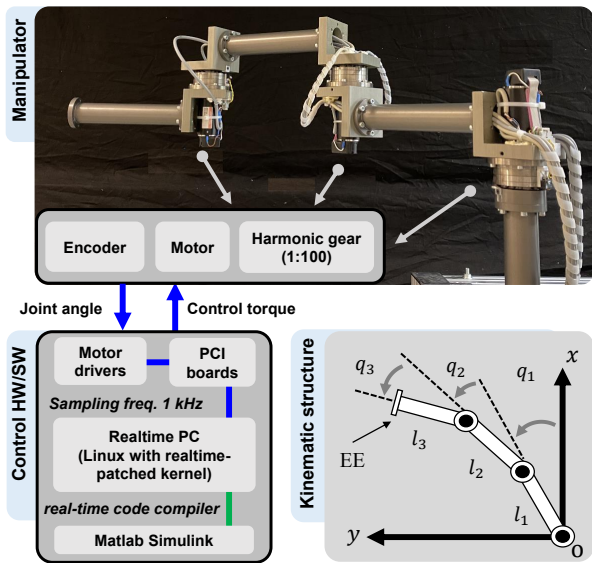


Fig. 2. Experimental setup of the robot manipulator with hardware and software architectures and its kinematic structure ($l_1 = l_2 = l_3 = 0.3$ m).

V. EXPERIMENT ON A 3-DOF ROBOT MANIPULATOR

The effectiveness of HIMPC is investigated in experiments. It is therefore compared to other state-of-the-art controllers, such as operational space formulation (OSF) [9], TDE enhanced OSF (TDEOSF) [10], and hierarchical QP (HQP) [16].

A. Experimental Setup

A custom-built 3-DoF planar robot manipulator is used for the experimental verification as shown in Fig. 2. See [48] for the system dynamics of the robot manipulator. Note that the identified system dynamics is not used for TDEOSF and HIMPC because they allow for model-free controller design. The manipulator is actuated by three torque-controlled (Maxon) motors with a turn ratio of 1:100. Mounted on each motor, the incremental encoders offer a joint position measurement with a resolution of 1.25×10^{-3} deg. Using a peripheral component interconnect (PCI) communication card, the sensors and actuators are connected with an Intel® Core™ (i7 8086K @4.0 GHz) CPU computer. The executable algorithm is created by Matlab 2017a in Ubuntu 14.04 LTS, using a first-order solver with a sampling rate of 1 kHz.

A task hierarchy with 2 priority levels is implemented. The task definitions and control gains are shown in Table I. For HQP and HIMPC, qpOASES [52] is used to solve the constrained OCPs. For HIMPC, weighting matrices are $\mathbf{Q}_i = 200\mathbf{I}$ and $\mathbf{R}_i = 10\mathbf{I}$ for $i = 1, 2$. The lower (upper) bounds of joint position, velocity, and torques are -150 ($+150$) deg, -100 ($+100$) deg/s, and -4 ($+4$) N·m. For TDEOSF and HIMPC, $\mathbf{M} = \text{diag}(0.033, 0.033, 0.033)$ is tuned, as addressed in Remark 4. On the premise of ensuring real-time control, we increase the prediction horizon as much as possible to receive satisfactory control performance. In the context of the hardware setup and requirements on safety, $N = 5$ is selected. The effects of the prediction horizon on the system will be investigated by simulations in Sec. VI.

TABLE I
TASK DEFINITIONS AND CONTROL GAINS IN EXPERIMENTS

Level	Task Definition	Stiffness	Damping
1	EE in x	200 N/m	10 Ns/m
1	EE in y	200 N/m	10 Ns/m
2	Joint position q_1	50 Nm/rad	5 Nms/rad

Two scenarios are implemented to verify the performance of the proposed method w.r.t. control accuracy, optimality, task hierarchy, singularity handling, and constraint handling.

B. Scenario 1: Control Accuracy and Optimality

1) *Setting*: The reference trajectory RT1 (see Fig. 3) is used and the initial configuration is $\mathbf{q}_0 = [0, 90, -90]^T$ deg.

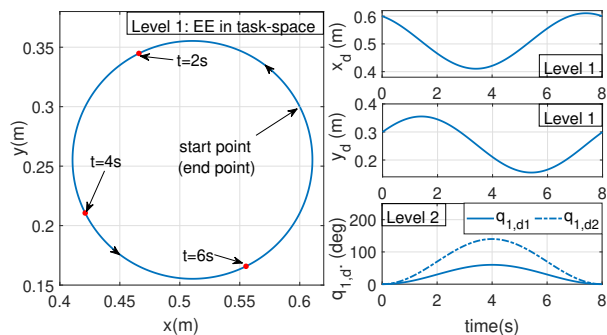


Fig. 3. Reference trajectories RT1 and RT2. x_d and y_d are reference signals for coordinates x and y of the EE in task space, respectively, and the reference geometric curve of the EE is a circle. $q_{1,d1}$ and $q_{1,d2}$ are reference signals for q_1 . RT1 includes x_d , y_d , and $q_{1,d1}$, and RT2 involves x_d , y_d , and $q_{1,d2}$.

- i) *Control Accuracy*. OSF, HQP, TDEOSF, and HIMPC are implemented for comparison. The discrepancy between the identified and the real mathematical model is considered as a disturbance. The robustness of tracking performance is analyzed by investigation of control accuracy.
- ii) *Optimality (qualitative)*. We take into account tracking errors and control signals in the cost function (31). Thus, tracking error and control signals are inspected to investigate the optimality of the control methods. We will first compare HIMPC and TDEOSF and analyze how optimality in HIMPC improves performance. Then, we introduce HIQP, i.e., an acronym of hierarchical incremental QP control which denotes HIMPC with the prediction horizon $N = 1$, and compare this to HIMPC with $N = 5$ to investigate local optimality of the controller. Note that HIQP is merely implemented in Sec. V-B.
- iii) *Optimality (quantitative)*. We will calculate the average cost when applying each of the controllers. With the stage cost for the predictions, see (31), we now introduce the average cost (41) for tasks to evaluate the closed-loop performance during a period $[0, t_s]$ (iterations $N_t = t_s/T_s$) where t_s is the terminal time.

$$C_i = \frac{1}{N_t} \sum_{k=0}^{N_t} (\|e(\mathbf{x}_i(k))\|_{\mathbf{Q}_i}^2 + \|\Delta\tau(k)\|_{\mathbf{R}_i}^2) \quad (41)$$

where C_i is the average cost for task i . Note that $e(\mathbf{x}_i(k))$ in (41) is calculated using measurements, not predictions.

2) *Results*: The results are shown in Figs. 4-8.

i) All of the controllers, OSF, HQP, TDEOSF, and HIMPC, allow to complete the two hierarchic tasks, but the tracking accuracy is different. As shown in Fig. 4 and Table II, tracking performance of the model-based methods (OSF and HQP) is inferior. This is because of the model discrepancy. Tracking performance of OSF and HQP might still be improved with a more accurate model, though its accurate identification is time-consuming and modeling errors are inevitable as we analyzed in Sec. I-A. TDE-based methods (TDEOSF and HIMPC) are model-free ones. Thus, they are not sensitive to the model discrepancy and exhibit strong robustness against disturbances. This allows for high tracking precision.

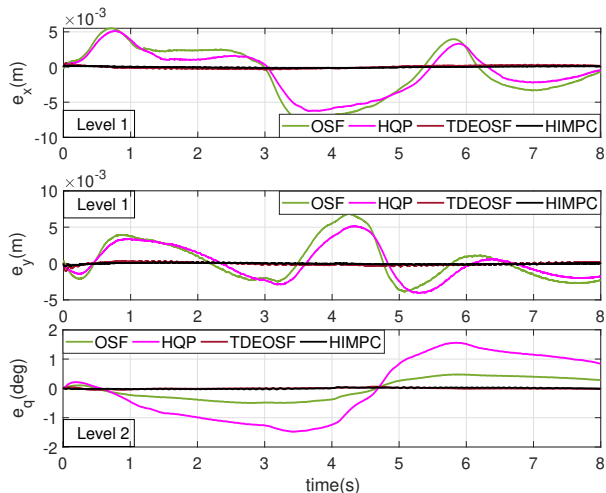


Fig. 4. Experimental results of *Scenario 1*: Tracking errors of OSF, HQP, TDEOSF, and HIMPC.

TABLE II

EXPERIMENTAL RESULTS OF *Scenario 1*: ROOT-MEAN-SQUARE-ERRORS (RMSE) OF DIFFERENT CONTROLLERS

Task	OSF	TDEOSF	HQP	HIMPC
L1: $x_{EE}(m)$	3.81×10^{-3}	9.58×10^{-5}	3.11×10^{-3}	9.07×10^{-5}
L1: $y_{EE}(m)$	2.78×10^{-3}	8.89×10^{-5}	2.39×10^{-3}	9.41×10^{-5}
L2: $q_1(\text{deg})$	0.36	8.02×10^{-3}	1.05	2.12×10^{-2}

ii) For further analysis, the tracking errors of TDEOSF and HIMPC are again displayed in Fig. 5, and we conclude that tracking performance of both methods is satisfactory. The tracking errors of TDEOSF are the smallest. This is because, in HIMPC, the cost function (31) does not only take into account the desired tracking error dynamics but also the control signal to avoid controller oscillations and to achieve energy efficiency and actuator protection. Thus, the superiority of HIMPC is visible when we in addition look at input trajectories. In Fig. 6, it is observed that in comparison to TDEOSF, control signals of HIMPC are smoother. There are many “glitches” in the control signal trajectories of TDEOSF. This is because the control signal of HIMPC is obtained through solving OCPs. In contrast, for TDEOSF, the control signal is calculated from analytic expressions, focusing on addressing tracking errors, while optimality is not considered.

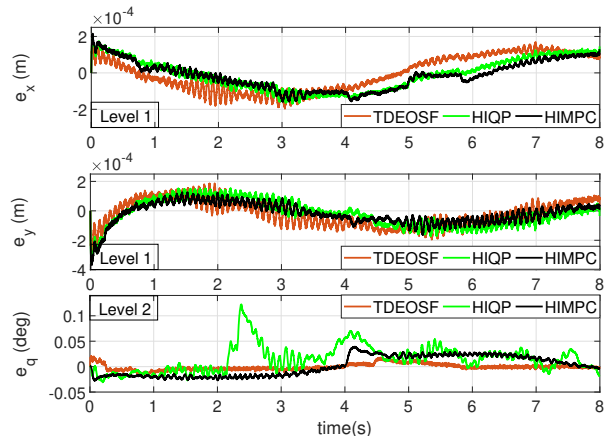


Fig. 5. Experimental results of *Scenario 1*: Tracking errors of TDE-based methods (TDEOSF, HIQP, and HIMPC).

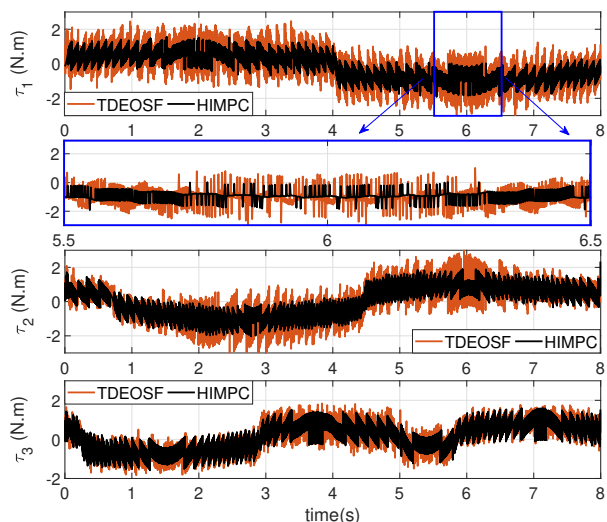


Fig. 6. Experimental results of *Scenario 1*: Control signals of TDEOSF and HIMPC.

A final comparison between HIQP and HIMPC analyzes the local optimality of the controller, and experimental results are shown in Figs. 5 and 7. For HIQP and HIMPC, the tracking errors of task 1 are very similar. However, for task 2, the tracking errors of HIQP are larger than those of HIMPC during some periods (eg. [2s, 3s]). In addition, we observe that the control signals of HIMPC are smoother than those of HIQP (see rectangular symbols in Fig. 7). This is because control signals of HIMPC are obtained by solving OCPs over a longer prediction horizon, while HIQP only takes into account a one-step-ahead prediction resulting in torque peaks and oscillations, especially for a noisy environment.

iii) Fig. 8 shows that the average cost of tasks 1-2 is the lowest for HIMPC, which quantitatively verifies the superior optimality. Although the control signals of HIQP are also obtained through solving constrained OCPs, the average cost of task 2 under the action of HIQP is even higher than that of TDEOSF. Due to its one-step prediction, the HIQP controller is locally optimal. Thus, over a time horizon, the average cost of HIQP may be even higher than that of TDEOSF, though

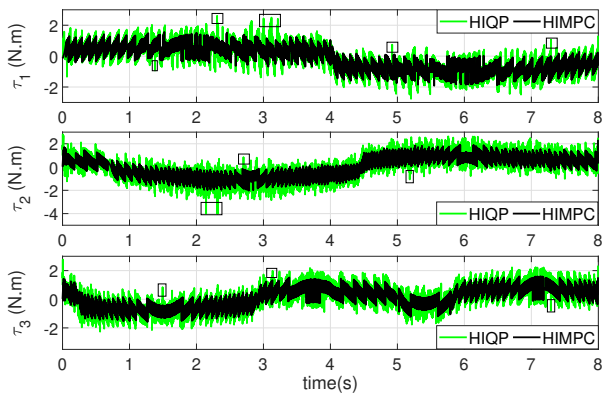


Fig. 7. Experimental results of *Scenario 1*: Control signals of HIQP and HIMPC.

optimality is not considered for TDEOSF controller design.

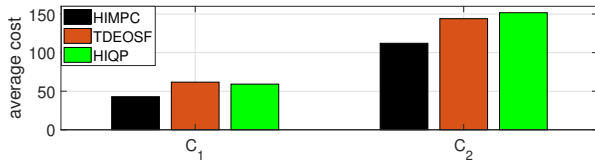


Fig. 8. Experimental results of *Scenario 1*: Average cost (41) of HIMPC, TDEOSF, and HIQP.

C. Scenario 2: Task Hierarchy, Constraint Admissibility, and Singularity Handling

1) *Setting*: The reference trajectories RT2 and RT3 (see Figs. 3 and 9, respectively) are considered in this scenario. The initial configuration is $\mathbf{q}_0 = [0, 90, -90]^T$ deg when RT2 is considered, while for RT3, $\mathbf{q}_0 = [0, 0, 0]^T$ deg is chosen.

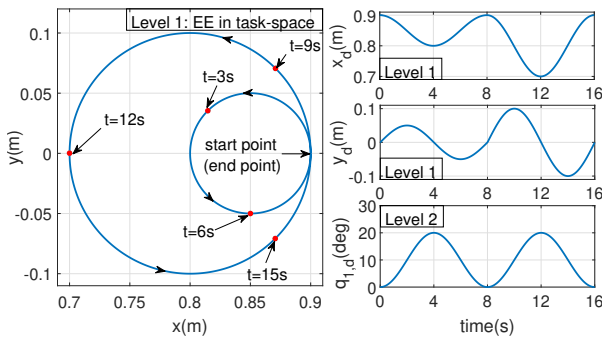


Fig. 9. Reference trajectory RT3. The reference curve of the EE consists of two tangent circles. At $t = 0$ s, 8s, and 16s, the reference configuration of the planar robot manipulator is straightened.

i) *Task Hierarchy and Constraint Admissibility*. As long as the amplitude of the reference trajectory of task 2 (q_1) is small, task 1 and task 2 are non-conflicting. With increasing amplitude, tasks become more and more conflicting. We verify task hierarchy by investigating whether tracking performance of high-priority tasks will be affected when high and low-priority tasks conflict, and the reference trajectory RT2 is employed. RT2 is also used to investigate the capability of HIMPC to guarantee that input and state constraints are not violated.

- ii) *Algorithmic Singularity Handling*. RT2 is used to demonstrate that algorithmic singularity is avoided in HQP and HIMPC. Null-space projection based methods (OSF and TDEOSF) are likely to have algorithmic singularity problems when tasks conflict. For safety, OSF and TDEOSF are implemented in simulations when RT2 is employed.
- iii) *Kinematic Singularity Handling*. Moreover, RT3 is used to verify that kinematic singularity is avoided in HIMPC. For RT3, the robot manipulator is straightened and is kinematically singular at the start, end, and intersection between two circles.

2) *Results*: The results are shown in Figs. 10-15.

i) Tracking errors of HQP and HIMPC are displayed in Fig. 10, and we observe that tracking errors of q_1 (task 2) are large in the time period [3.2s, 6s]. This is because, during this period, tasks conflict. The low-priority task (task 2) has to “sacrifice” itself to guarantee satisfactory tracking performance of the high-priority task. Thus, for HQP and HIMPC, task hierarchy is guaranteed. Besides, because HQP is a model-based method, the tracking error of HQP is higher than that of HIMPC. This further verifies the robustness of HIMPC.

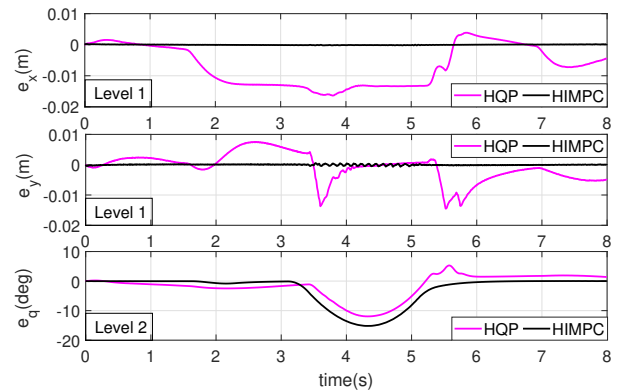


Fig. 10. Experimental results of *Scenario 2*: Tracking errors of HQP and HIMPC (the reference trajectory is RT2).

Besides, *dynamic consistency* can be verified by investigating whether tracking errors of the highest-priority task (task 1) are affected when different reference trajectories (RT1 and RT2) are employed. For the convenience of comparison, tracking errors of task 1 when RT1 and RT2 are employed are summarized in Fig. 11, where one observes that for HQP, tracking performance of task 1 deteriorates when tasks conflict. This is because the nominal \mathbf{M} is used in HQP to construct task priority constraints and *dynamic consistency* is adversely affected by the inertia modeling error. In contrast, for HIMPC, tracking performance of task 1 is nearly not affected no matter whether tasks conflict or not. This is because the TDE method is used to approximate equations of motion of tasks and \mathbf{M} is replaced by a predetermined $\bar{\mathbf{M}}$ in the task priority constraints (compare (34g)). Thus, enhanced *dynamic consistency* of HIMPC is obtained due to the TDE approximation.

In Fig. 12 it is shown that input constraints are not violated. During the period [3.5s, 5.2s], the input torques of HIMPC are relatively large. This is because here tasks conflict. The control signal not only needs to guarantee the priority of task 1, but

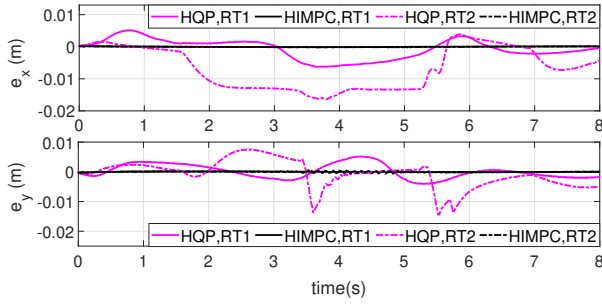


Fig. 11. Experimental results of *Scenario 2*: Tracking errors of task 1 (RT1 and RT2 are employed).

it also regulates tracking performance of low-priority tasks as well as possible. As we discussed in Sec. IV-C, the TDE error and measurement noise are not considered for state predictions in HIMPC, and it results in small prediction errors, see Fig. 13, where joint velocity constraints are slightly violated for short periods. This is common in MPC for uncertain systems with measurement noises. Though, at least we observe in our study that state constraints are not strongly violated. This is because the incremental systems exhibit high approximation accuracy.

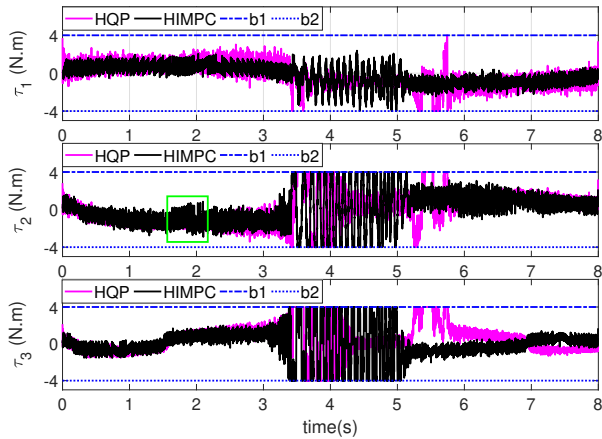


Fig. 12. Experimental results of *Scenario 2*: Torque inputs of three joints with the reference trajectory RT2, where $b_{1,2}$ denotes the corresponding bounds.

ii) As stated in Sec. I-A, null-space projection based methods, such as OSF and TDEOSF, are likely to suffer from algorithmic singularity when tasks conflict. For safety, we did numerical simulations to implement OSF and TDEOSF. Consistent with theoretical analysis, the values of $\mathbf{J}_{2|p}\mathbf{M}^{-1}\mathbf{J}_{2|p}^T$ and $\mathbf{J}_{2|p}\bar{\mathbf{M}}^{-1}\mathbf{J}_{2|p}^T$ ($\mathbf{J}_{2|p}$ is the prioritized Jacobian matrix) under the action of OSF and TDEOSF tend to be 0 at around 3.3s. In other words, $\mathbf{J}_{2|p}$ loses its rank (and is a null vector here). Then, $(\mathbf{J}_{2|p}\mathbf{M}^{-1}\mathbf{J}_{2|p}^T)^{-1}$ and $(\mathbf{J}_{2|p}\bar{\mathbf{M}}^{-1}\mathbf{J}_{2|p}^T)^{-1}$ tend to infinity and undesired (large-value) control signals are obtained. As a result, the system will be unstable. Note that in the context of task definitions (see Table I), $\mathbf{J}_2=[1, 0, 0]$ is a constant vector. Thus, at this moment the system is not kinematically singular, but it is attributed to the prioritized Jacobian matrix (null-space projection idea), i.e., the algorithmic singularity occurs. For HQP and HIMPC, equality constraints

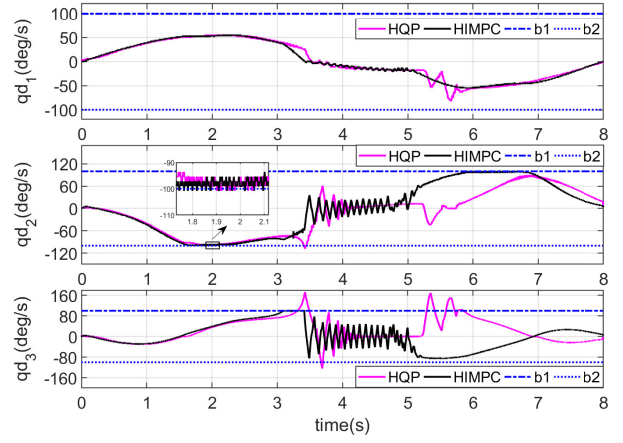


Fig. 13. Experimental results *Scenario 2*: Joint velocities of three joints with the reference trajectory RT2, where $b_{1,2}$ denotes the corresponding bounds.

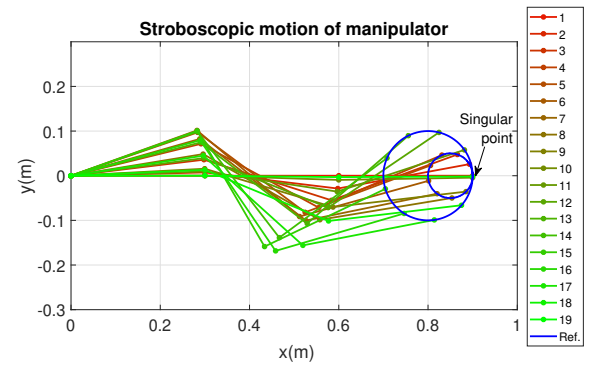


Fig. 14. Experimental results of *Scenario 2*: Stroboscopic motion of the robot manipulator using the proposed HIMPC. The reference trajectory is RT3, and the posture is kinematically singular at the 1st, 10th, and 19th points.

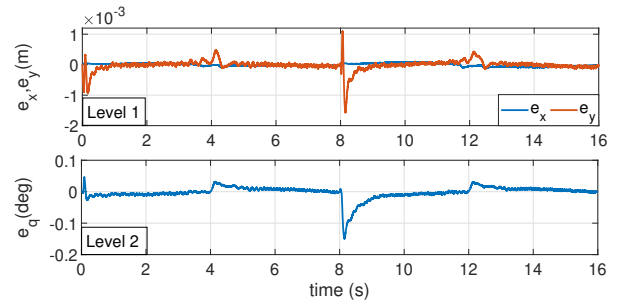


Fig. 15. Experimental results of *Scenario 2*: Tracking errors of HIMPC (the reference trajectory is RT3).

are employed to achieve task hierarchy, and no prioritized inertia matrices are involved. Thus, *algorithmic* singularity is avoided in HQP and HIMPC.

iii) We finally verify that HIMPC can control even if the manipulator passes the kinematically singular configuration in *Scenario 2*, and RT3 (see Fig. 9) is considered. OSF, TDEOSF, and HQP cannot be implemented because of *kinematic* singularity in the initial configuration. The experimental results are shown in Figs. 14 - 15. As shown in Fig. 15, tracking accuracy of HIMPC is still high. It verifies that HIMPC is kinematically singularity-free. Note that tracking errors of tasks (especially

e_y shown in Fig. 15) are slightly larger at the beginning and around 8s. On the one hand, this is because the control degrees of freedom are partially lost when the robot manipulator is kinematically singular, and on the other hand, the reference trajectory is non-smooth at 8s. Thus, this task is challenging and it also evaluates the ability of this method to handle input and state constraints. Nevertheless, HIMPC guarantees that the system passes the singular and also non-smooth region safely.

Finally, we summarize the experimental results in Table III.

TABLE III
COMPARISON BETWEEN THE PROPOSED HIMPC AND STATE-OF-THE-ART CONTROLLERS

Property	OSF	TDEOSF	HQP	HIMPC
High Accuracy	✗	✓	✗	✓
Singularity-Free	✗	✗	✗	✓
Input Constraints	✗	✗	✓	✓
State Constraints	✗	✗	✗	✓

VI. SIMULATION OF A 7-DOF KUKA ROBOT

The effectiveness of HIMPC is now in addition validated by simulations for a higher-DoF robot manipulator that allows for a task hierarchy with more than two task levels.

A. Simulation Setup

We simulate a 7-DoF KUKA LBR iiwa 14 R820 [64] using the Matlab Robotics System Toolbox [65] on a PC with Intel® Core™ CPU (i7 9700K @3.60 GHz). The sampling rate is 1 kHz. A task hierarchy with 3 priority levels (see Fig. 16) is implemented with task definitions and control gains in the target motion dynamics (4) from Table IV. We introduce the tool center point (TCP) and x , y , and z denote its Cartesian coordinates in the body frame, while o_X , o_Y , and o_Z are its orientation w.r.t. the X -, Y - and Z -axes. The initial configuration is $\mathbf{q}_0 = [0, 60, 0, 60, 0, 0, 0]^T$ deg. $\mathbf{Q}_i = 200\mathbf{I}$ and $\mathbf{R}_i = 10\mathbf{I}$ for $i = 1, 2, 3$. Constraints for joint position, velocity, and torques are chosen according to [64]. The constrained OCPs in HIMPC are solved by qpOASES. According to Remark 4, the diagonal matrix is chosen as $\mathbf{M} = \text{diag}(0.1, 0.21, 0.033, 0.042, 0.001, 0.001, 0.0003)$.

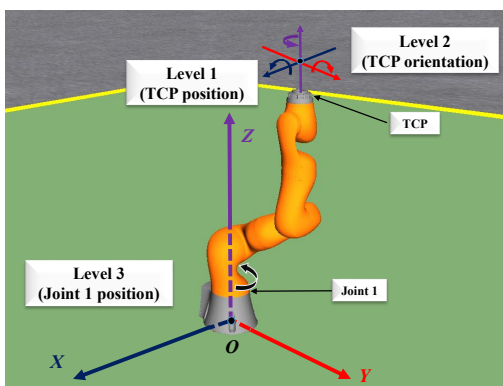


Fig. 16. Simulation setup of the 7-DoF manipulator (KUKA LBR iiwa) with three-level task hierarchy.

TABLE IV
TASK DEFINITIONS AND CONTROL GAINS FOR SIMULATIONS

Level	Task Definition	Stiffness	Damping
1	TCP position in x	200 N/m	10 Ns/m
1	TCP position in y	200 N/m	10 Ns/m
1	TCP position in z	200 N/m	10 Ns/m
2	TCP orientation (o_X)	50 Nm/rad	5 Nms/rad
2	TCP orientation (o_Y)	50 Nm/rad	5 Nms/rad
2	TCP orientation (o_Z)	50 Nm/rad	5 Nms/rad
3	Joint position q_1	50 Nm/rad	5 Nms/rad

The reference trajectory RT4 is considered in the simulation. The reference curve of TCP is a “circle” with a radius of 10 cm (level 1), i.e., reference signals for Cartesian coordinates $x_d = 0.2637 - 0.1\sin(0.5\pi t - 0.5\pi)$, $y_d = -0.1\sin(0.5\pi t)$, and $z_d = 1.096 - 0.1(0.5\pi t)$. The orientation of TCP is assigned to be maintained (level 2), i.e., $o_X = o_Y = o_Z = 0$ deg. Moreover, q_1 is commanded to move in a range of 180 deg (level 3), i.e., $q_{1,d} = 90 + 90\sin(0.5\pi t - 0.5\pi)$ deg. Three scenarios are considered to verify the feasibility of HIMPC for higher-DoF robotic systems, to investigate effects of the prediction horizon on the system, to verify reasonability of approximation of Jacobians over the prediction horizon, and to investigate effects of the measurement resolution on control performance.

B. Scenario 1: Feasibility for Higher-DoF Robotic Systems

1) *Setting*: RT4 is designed such that low-priority tasks conflict with high-priority tasks. Task hierarchy is investigated by measuring whether tracking performance of high-priority tasks is affected by low-priority tasks. Also, it is verified that singularities are successfully avoided. The prediction horizon of HIMPC is $N = 5$ in this scenario. In addition, to further verify robustness of the developed HIMPC against inertia matrix modeling errors, an additional simulation is conducted with a disturbed inertia matrix. Each diagonal element of the inertia matrix is either reduced or increased by 20% of its nominal value, following the order $- - + + - + +$.

2) *Simulation Results of Scenario 1*: The results are displayed in Fig. 17. We observe that the low-priority task “sacrifices” itself to guarantee the task hierarchy, while tracking performance of high-priority tasks is almost not affected. For [3.8s, 4s], tracking errors of Level 2 tasks (especially the orientation of TCP w.r.t. Y-axis) slightly increase. It is mainly because joint positions are constrained. However, for OSF, TDEOSF, and HQP, the system is unstable. It is because, in the initial configuration, the system is kinematically singular, i.e., the Jacobian matrix loses its rank ($\text{rank}(\text{col}(\mathbf{J}_1, \mathbf{J}_2, \mathbf{J}_3)) = 6 < 7$). For HIMPC, we see that tracking errors of the level 3 task (q_1) are slightly larger in the first 0.2 seconds. This is because the robot manipulator is kinematically singular and control degrees of freedom of the system are partially lost during this time period. Nevertheless, the system passes the singular region safely. For a disturbed inertia matrix (see Table V), we see that control accuracy (RMSE) and dynamic consistency are nearly not affected. This is in line with the theoretical analysis in Sec. III-B, i.e., the incremental systems and controller tolerate inertia modeling errors.

TABLE V
SIMULATION RESULTS OF *Scenario 2*: RMSE OF HIMPC FOR SYSTEMS WITH NOMINAL AND DISTURBED INERTIA MATRICES

Case	Level 1			Level 2			Level 3
	x (m)	y (m)	z (m)	α_x (deg)	α_y (deg)	α_z (deg)	q_1 (deg)
Nominal Inertia Matrix	1.849×10^{-4}	4.188×10^{-4}	2.160×10^{-4}	3.686×10^{-2}	2.062×10^{-2}	2.037×10^{-1}	8.773
Disturbed Inertia Matrix	1.856×10^{-4}	4.192×10^{-4}	2.115×10^{-4}	3.388×10^{-2}	2.290×10^{-2}	2.122×10^{-1}	8.796

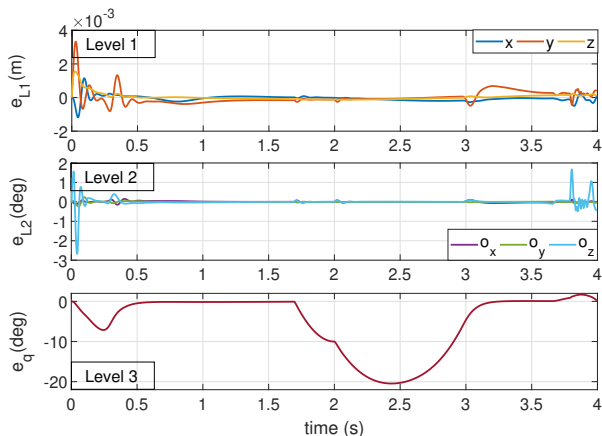


Fig. 17. Simulation results of *Scenario 1*: Tracking errors of HIMPC for the 7-DoF KUKA robot.

C. Scenario 2: Investigating Effects of the Prediction Horizon and Verifying Reasonability of Constraint Simplifications

1) *Setting*: In order to investigate effects of the length of the prediction horizon on the system, larger prediction horizons ($N = 10, 20, 30$) are used for HIMPC in this scenario. To further investigate reasonability of constraint simplifications and verify computational efficiency, a nonlinear HIMPC (NHIMPC) scheme is introduced, where the Jacobian matrices over the prediction horizon are updated by current joint state predictions, i.e., simplification in (29) and (33) are suspended in NHIMPC, which is also the only difference between HIMPC and NHIMPC, aiming for objective of comparison simulations. The constrained OCPs in the NHIMPC scheme is solved by an efficient NLP toolbox, CasADi [66], where the SQP method is employed. The computational complexity of the algorithms is investigated by monitoring the computing time for each prediction step.

2) *Simulation Results of Scenario 2*: For the investigated tasks (see Fig. 16), the Jacobian of task 3 is constant. Thus, the simplifications in (29) and (33) affect performance of Level 1 and 2 tasks mostly. Due to space limitations, we will only display tracking errors of x in this scenario. The simulation results are shown in Figs. 18-19 and Tables VI-VII.

As expected from the theory of MPC without terminal ingredients, the tracking performance becomes better with the rise of prediction horizons, see Fig. 18. A comparison between HIMPC and NHIMPC is shown in Table VI. Although the Jacobian is updated over the prediction horizon, RMSE of NHIMPC and HINPC are almost equal. Besides, during time periods [0.16s, 0.2s] and [0.4s, 0.5s] (see Fig. 19), tracking performance of NHIMPC is even worse. Due to the updated

TABLE VI
SIMULATION RESULTS OF *Scenario 2*: RMSE OF HIMPC AND NHIMPC (10^{-4} m)

Controllers	$N = 10$	$N = 20$	$N = 30$
HIMPC	1.710	1.613	1.542
NHIMPC	1.701	1.598	1.525

Jacobian, convexity of the constrained OCPs in NHIMPC is not guaranteed. For non-convex programming problems, it may happen that only a local optimum is found that might be much worse than the global one. In addition, the computing time of the non-convex programming problems are significantly higher (cf. Table VII). As shown in Fig. 20, the average computing time of HIMPC increases exponentially with the rise of the prediction horizon, and it is much less than that of NHIMPC. Thus, the simplification in (29) not only guarantees convexity of the constrained OCPs, but also decreases computational complexity of the algorithm. It is worth to notice that the average computing time of HIMPC with $N = 5$ (8.05×10^{-3} s) is longer than the sampling period (1ms). It means the sampling time is to be higher to guarantee the real-time control for the considered 7-DoF manipulator with 3 task levels in the current controller implementation. To apply HIMPC on higher-DoF systems, yet to keep a small sampling period (e.g., 1ms) in the same hardware setup, more practical implementation effort will be devoted to developing faster QP solvers [67] and approximation solutions.

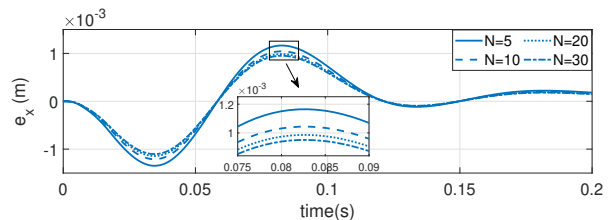


Fig. 18. Simulation results of *Scenario 2*: Tracking errors of TCP positions in X -axis under the action of HIMPC with $N = 5, 10, 20, 30$.

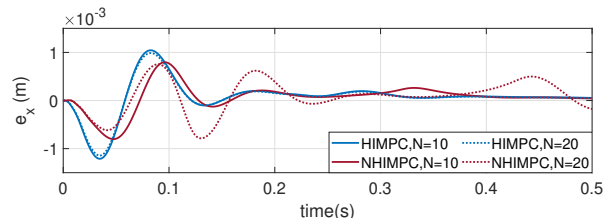


Fig. 19. Simulation results of *Scenario 2*: Tracking errors of TCP positions in X -axis under the action of HIMPC and NHIMPC with $N = 10, 20$.

TABLE VII
SIMULATION RESULTS OF *Scenario 2*: COMPUTING TIME OF
HIMPC AND NHIMPC WITH $N = 5$

Controllers	Average time	Median time	Maximum time
HIMPC	8.05×10^{-3} s	9.04×10^{-3} s	1.63×10^{-2} s
NHIMPC	0.37 s	0.39 s	0.54 s

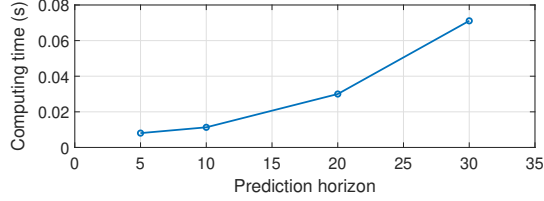


Fig. 20. Simulation results of *Scenario 2*: Average computing time of HIMPC per iteration with different prediction horizons.

D. Scenario 3: Investigating Effects of the Measurement Resolution on Control Performance

1) *Setting*: The prediction horizon $N = 5$, and a series of measurement resolution from 0.5×10^{-3} deg to 1×10^{-2} deg is selected. For a coarse resolution, the noisy response is triggered and control signal chattering will be observed. The noisy response is analyzed by investigating smoothness of control signal curves. The smoothness of curves is measured using the concept of curve smoothness, which is defined as: $S = \sqrt{(\sum_{k=0}^{N_t} \|\mathbf{c}(k) - \bar{\mathbf{c}}(k)\|^2) / N_t}$, where $\bar{\mathbf{c}}(k)$ denotes the control signal for the benchmark, generated with the resolution of 1×10^{-5} deg in this simulation.

2) *Simulation Results of Scenario 3*: The results are shown in Table VIII. The curve smoothness decreases when the resolution is coarser. Because of space limitation, we take τ_1 for example. We learn that when the curve smoothness is greater than 10 N·m, the control signal chattering is observed (see Fig. 21). In the example, to avoid excessive wear of actuators and guarantee great control performance, it is suggested that the encoder measures the joint angle more precisely than 5×10^{-3} deg. For instance, one can consider the encoder with a resolution of 16 bit attached after the gear, or that with 10 bit at the motor shaft with a gear ratio of 100.

TABLE VIII
SIMULATION RESULTS OF *Scenario 3*: CURVE SMOOTHNESS VALUES
OF CONTROL SIGNAL CURVES WITH DIFFERENT RESOLUTIONS

Resolution (10^{-3} deg)	0.5	1	2	5	10
Curve Smoothness (N·m)	0.93	2.20	4.54	11.62	22.81

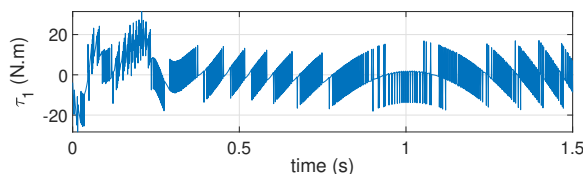


Fig. 21. Simulation results of *Scenario 3*: Control signals with the measurement resolution of 5×10^{-3} deg.

VII. CONCLUSION

In this article, a hierarchical incremental model predictive control (HIMPC) is proposed for robot manipulators to execute multiple tasks simultaneously. To reduce dependency on an accurate mathematical model, equations of motion and system dynamics are approximated by incremental systems using time-delay estimation (TDE). It improves the robustness of the controller in terms of high tracking accuracy and enhanced dynamics consistency. For HIMPC, a series of constrained optimal control problems (OCPs) is developed, where task hierarchy is achieved by equality constraints, which are set based on the *dynamic consistency* principle instead of using null-space projections. Besides, inversion of terms that include Jacobians is not needed and all Hessian matrices of the constrained OCPs are positive definite. As a result, the proposed HIMPC is algorithmically and kinematically singularity-free. In addition, HIMPC results in a linear MPC. Compared with nonlinear MPC, the computational complexity of HIMPC dramatically decreases, which enables the controller to run with a sampling time of 1 ms. Finally, simulations and experiments were conducted, and the efficacy of HIMPC is validated.

Future research will be devoted to studying HIMPC with strict state constraints and it will be addressed by the computationally efficient data-driven disturbance estimation algorithm and the stochastic MPC method.

APPENDIX A PARAMETERS IN (35)

$$\begin{aligned}
 \mathbf{Q}_i &:= \bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i \bar{\mathbf{B}}_{1i} + \bar{\mathbf{R}}_i; \mathcal{L}_i := 2\bar{\mathbf{B}}_{1i}^T \bar{\mathbf{Q}}_i (\bar{\mathbf{A}}_1 \bar{\mathbf{x}}_i(k) - \mathbf{X}_{id}); \\
 \mathbf{Q}_i &:= \text{diag}(\bar{\mathbf{K}}_i^T \mathbf{Q}_i \bar{\mathbf{K}}_i, \dots, \bar{\mathbf{K}}_i^T \mathbf{Q}_i \bar{\mathbf{K}}_i); \bar{\mathbf{B}}_{1i} := \\
 &\begin{bmatrix} \hat{\mathbf{B}}_{1i}^0 & & & & & \\ \mathbf{A}_1 \hat{\mathbf{B}}_{1i}^0 & \hat{\mathbf{B}}_{1i}^1 & & & & \\ \mathbf{A}_1^2 \hat{\mathbf{B}}_{1i}^0 & \mathbf{A}_1 \hat{\mathbf{B}}_{1i}^1 & \hat{\mathbf{B}}_{1i}^2 & & & \\ \dots & & & & & \\ \mathbf{A}_1^{N-1} \hat{\mathbf{B}}_{1i}^0 & \mathbf{A}_1^{N-2} \hat{\mathbf{B}}_{1i}^1 & \mathbf{A}_1^{N-3} \hat{\mathbf{B}}_{1i}^2 & \dots & \hat{\mathbf{B}}_{1i}^{N-1} & \end{bmatrix}; \\
 \mathbf{X}_{id} &:= \text{col}(\bar{\mathbf{x}}_{id}(k+1), \dots, \bar{\mathbf{x}}_{id}(k+N)); \\
 \bar{\mathbf{R}}_i &:= \text{diag}(\mathbf{R}_i, \dots, \mathbf{R}_i); \bar{\mathbf{K}}_i := [\mathbf{K}_i, \mathbf{O}]; \\
 \bar{\mathbf{A}}_1 &:= \text{col}(\mathbf{A}_1, \mathbf{A}_1^2, \dots, \mathbf{A}_1^N); \mathbf{C}_{i1} := \text{col}(-\mathbf{c}_{i1}, \mathbf{c}_{i1}); \\
 \mathbf{D}_{i1} &:= \begin{bmatrix} -\bar{\mathbf{L}}_1 \bar{\mathbf{q}}(k) + \bar{\mathbf{q}}_{\min} \\ \bar{\mathbf{L}}_1 \bar{\mathbf{q}}(k) - \bar{\mathbf{q}}_{\max} \end{bmatrix}; \mathbf{c}_{i1} := \\
 &\begin{bmatrix} \mathbf{L}_1 \mathbf{B}_2 \\ \mathbf{L}_1 \mathbf{A}_2 \mathbf{B}_2 & \mathbf{L}_1 \mathbf{B}_2 \\ \mathbf{L}_1 \mathbf{A}_2^2 \mathbf{B}_2 & \mathbf{L}_1 \mathbf{A}_2 \mathbf{B}_2 & \mathbf{L}_1 \mathbf{B}_2 \\ \dots & & & \\ \mathbf{L}_1 \mathbf{A}_2^{N-1} \mathbf{B}_2 & \mathbf{L}_1 \mathbf{A}_2^{N-2} \mathbf{B}_2 & \mathbf{L}_1 \mathbf{A}_2^{N-3} \mathbf{B}_2 & \dots & \mathbf{L}_1 \mathbf{B}_2 \end{bmatrix}; \\
 \bar{\mathbf{L}}_1 &:= \text{col}(\mathbf{L}_1 \mathbf{A}_2, \mathbf{L}_1 \mathbf{A}_2^2, \dots, \mathbf{L}_1 \mathbf{A}_2^N); \\
 \bar{\mathbf{q}}_{\min} &:= \text{col}(\mathbf{q}_{\min}, \dots, \mathbf{q}_{\min}); \\
 \bar{\mathbf{q}}_{\max} &:= \text{col}(\mathbf{q}_{\max}, \dots, \mathbf{q}_{\max}); \mathbf{L}_1 := [\mathbf{I}, \mathbf{O}, \mathbf{O}, \mathbf{O}]; \\
 \mathbf{C}_{i2} &:= \text{col}(-\mathbf{c}_{i2}, \mathbf{c}_{i2}); \mathbf{D}_{i2} := \begin{bmatrix} -\bar{\mathbf{L}}_2 \bar{\mathbf{q}}(k) + \bar{\mathbf{q}}_{\min} \\ \bar{\mathbf{L}}_2 \bar{\mathbf{q}}(k) - \bar{\mathbf{q}}_{\max} \end{bmatrix}; \\
 \bar{\mathbf{L}}_2 &:= \text{col}(\mathbf{L}_2 \mathbf{A}_2, \mathbf{L}_2 \mathbf{A}_2^2, \dots, \mathbf{L}_2 \mathbf{A}_2^N); \mathbf{L}_2 := [\mathbf{O}, \mathbf{I}, \mathbf{O}, \mathbf{O}]; \\
 \bar{\dot{\mathbf{q}}}_{\min} &:= \text{col}(\dot{\mathbf{q}}_{\min}, \dots, \dot{\mathbf{q}}_{\min}); \\
 \bar{\dot{\mathbf{q}}}_{\max} &:= \text{col}(\dot{\mathbf{q}}_{\max}, \dots, \dot{\mathbf{q}}_{\max}); \mathbf{C}_{i3} := \text{col}(-\bar{\mathbf{I}}, \bar{\mathbf{I}}); \mathbf{c}_{i2} := \\
 &\begin{bmatrix} \mathbf{L}_2 \mathbf{B}_2 \\ \mathbf{L}_2 \mathbf{A}_2 \mathbf{B}_2 & \mathbf{L}_2 \mathbf{B}_2 \\ \mathbf{L}_2 \mathbf{A}_2^2 \mathbf{B}_2 & \mathbf{L}_2 \mathbf{A}_2 \mathbf{B}_2 & \mathbf{L}_2 \mathbf{B}_2 \\ \dots & & & \\ \mathbf{L}_2 \mathbf{A}_2^{N-1} \mathbf{B}_2 & \mathbf{L}_2 \mathbf{A}_2^{N-2} \mathbf{B}_2 & \mathbf{L}_2 \mathbf{A}_2^{N-3} \mathbf{B}_2 & \dots & \mathbf{L}_2 \mathbf{B}_2 \end{bmatrix};
 \end{aligned}$$

$$\mathbf{D}_{i3} := \begin{bmatrix} -\bar{\tau}_0 + \bar{\tau}_{\min} \\ \bar{\tau}_0 - \bar{\tau}_{\max} \end{bmatrix}; \bar{\mathbf{I}} := \begin{bmatrix} \mathbf{I} & & & \\ & \mathbf{I} & & \\ & & \dots & \\ & & & \mathbf{I} & \dots & \mathbf{I} \end{bmatrix};$$

$$\bar{\boldsymbol{\tau}}_0 := \text{col}(\boldsymbol{\tau}_0, \dots, \boldsymbol{\tau}_0); \bar{\boldsymbol{\tau}}_{\min} := \text{col}(\boldsymbol{\tau}_{\min}, \dots, \boldsymbol{\tau}_{\min});$$

$$\bar{\boldsymbol{\tau}}_{\max} := \text{col}(\boldsymbol{\tau}_{\max}, \dots, \boldsymbol{\tau}_{\max}); \mathbf{a}_{ip}^\top := \bar{\mathbf{J}}_{p,k}; \mathbf{b}_{ip} := \bar{\mathbf{J}}_{p,k} \mathbf{U}_p^*;$$

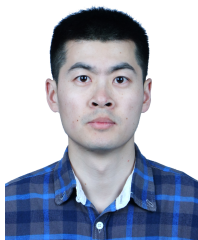
$$\bar{\mathbf{J}}_{p,k} := \text{diag}(\bar{\mathbf{J}}_{p,k}^0 \bar{\mathbf{M}}^{-1}, \dots, \bar{\mathbf{J}}_{p,k}^{N-1} \bar{\mathbf{M}}^{-1});$$

$$\mathbf{U}_p^* := \text{col}(\Delta \boldsymbol{\tau}_{p,k|k}^*, \dots, \Delta \boldsymbol{\tau}_{p,k+N-1|k}^*).$$

REFERENCES

- [1] F. Flacco, A. De Luca, and O. Khatib, "Control of redundant robots under hard joint constraints: Saturation in the null space," *IEEE Trans. on Robot.*, vol. 31, no. 3, pp. 637–654, Jun. 2015.
- [2] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 1006–1028, Jul. 2014.
- [3] K. Al Khudir *et al.*, "Stable torque optimization for redundant robots using a short preview," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 2046–2053, Apr. 2019.
- [4] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE J. Robot. Autom.*, vol. 3, no. 1, pp. 43–53, Feb. 1987.
- [5] —, "Inertial properties in robotic manipulation: An object-level framework," *Int. J. Robot. Res.*, vol. 14, no. 1, pp. 19–36, Feb. 1995.
- [6] J. Nakanishi *et al.*, "Operational space control: A theoretical and empirical comparison," *Int. J. Robot. Res.*, vol. 27, no. 6, pp. 737–757, Jun. 2008.
- [7] A. Dietrich, C. Ott, and J. Park, "The hierarchical operational space formulation: Stability analysis for the regulation case," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 1120–1127, Apr. 2018.
- [8] C. Ott, A. Dietrich, and A. Albu-Schäffer, "Prioritized multi-task compliance control of redundant manipulators," *Automatica*, vol. 53, pp. 416–423, Mar. 2015.
- [9] A. Dietrich and C. Ott, "Hierarchical impedance-based tracking control of kinematically redundant robots," *IEEE Trans. Robot.*, vol. 36, no. 1, pp. 204–221, Feb. 2020.
- [10] P. H. Chang and J. W. Jeong, "Enhanced operational space formulation for multiple tasks by using time-delay estimation," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 773–786, Aug. 2012.
- [11] J. Lee *et al.*, "Robust and adaptive dynamic controller for fully-actuated robots in operational space under uncertainties," *Auton. Robots*, vol. 43, no. 4, pp. 1023–1040, Apr. 2019.
- [12] —, "Robust and adaptive whole-body controller for humanoids with multiple tasks under uncertain disturbances," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Stockholm, Sweden, May 2016, pp. 5683–5689.
- [13] D. Nicolis, F. Allevi, and P. Rocco, "Robust impedance shaping of redundant teleoperators with time-delay via sliding mode control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst. (IROS)*, Macau, China, Nov. 2019, pp. 2740–2747.
- [14] L. Saab *et al.*, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, Aug. 2013.
- [15] J. D. M. Osorio *et al.*, "Unilateral constraints for torque-based whole-body control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Las Vegas, USA, Jan. 2020, pp. 7623–7628.
- [16] E. M. Hoffman *et al.*, "Multi-priority Cartesian impedance control based on quadratic programming optimization," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, Australia, May 2018, pp. 309–315.
- [17] J. Lee *et al.*, "MPC-based hierarchical task space control of underactuated and constrained robots for execution of multiple tasks," in *Proc. IEEE Conf. Decis. Control (CDC)*, Jeju, Korea, Dec. 2020, pp. 5942–5949.
- [18] D. Nicolis, F. Allevi, and P. Rocco, "Operational space model predictive sliding mode control for redundant manipulators," *IEEE Trans. Robot.*, vol. 36, no. 4, pp. 1348–1355, Aug. 2020.
- [19] J. Hollerbach and K. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 308–316, Aug. 1987.
- [20] A. Karami *et al.*, "Hierarchical tracking task control in redundant manipulators with compliance control in the null-space," *Mechatronics*, vol. 55, pp. 171–179, Nov. 2018.
- [21] K. P. Tee and R. Yan, "Adaptive operational space control of redundant robot manipulators," in *Proc. American Control Conf. (ACC)*, San Francisco, USA, Jul. 2011, pp. 1742–1747.
- [22] J. Peters and S. Schaal, "Learning to control in operational space," *Int. J. Robot. Res.*, vol. 27, no. 2, pp. 197–212, Feb. 2008.
- [23] K. Youcef-Toumi and S.-T. Wu, "Input/output linearization using time delay control," *Trans. ASME J. Dyn. Syst., Meas. Control*, vol. 114, no. 1, pp. 10–19, Mar. 1992.
- [24] T. C. Hsia and L. Gao, "Robot manipulator control using decentralized linear time-invariant time-delayed joint controllers," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Cincinnati, USA, May 1990, pp. 2070–2075.
- [25] M. Jin *et al.*, "Practical nonsingular terminal sliding-mode control of robot manipulators for high-accuracy tracking control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3593–3601, Sep. 2009.
- [26] J. Lee *et al.*, "Microscale precision control of a computer-assisted transoral laser microsurgery system," *IEEE/ASME Trans. Mechatron.*, vol. 25, no. 2, pp. 604–615, Apr. 2020.
- [27] J. Lee, P. H. Chang, and M. Jin, "Adaptive integral sliding mode control with time-delay estimation for robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 64, no. 8, pp. 6796–6804, Aug. 2017.
- [28] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Trans. Robot. Autom.*, vol. 13, no. 3, pp. 398–410, Jun. 1997.
- [29] A. Dietrich, A. Albu-Schäffer, and G. Hirzinger, "On continuous null space projections for torque-based, hierarchical, multi-objective manipulation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Saint Paul, USA, May 2012, pp. 2978–2985.
- [30] O. Kanoun, F. Lamiroux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 785–792, Aug. 2011.
- [31] M. Faroni *et al.*, "Predictive inverse kinematics for redundant manipulators with task scaling and kinematic constraints," *IEEE Trans. Robot.*, vol. 35, no. 1, pp. 278–285, Feb. 2019.
- [32] M. Faroni, M. Beschi, and N. Pedrocchi, "Inverse kinematics of redundant manipulators with dynamic bounds on joint movements," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6435–6442, Oct. 2020.
- [33] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, Dec. 2014.
- [34] L. Grüne, "Economic receding horizon control without terminal constraints," *Automatica*, vol. 49, no. 3, pp. 725–734, Mar. 2013.
- [35] J. Köhler, M. A. Müller, and F. Allgöwer, "Nonlinear reference tracking: An economic model predictive control perspective," *IEEE Trans. Automat. Control*, vol. 64, no. 1, pp. 254–269, Jan. 2019.
- [36] K. Bouyarmane and A. Kheddar, "On weight-prioritized multitask control of humanoid robots," *IEEE Trans. Automat. Control*, vol. 63, no. 6, pp. 1632–1647, Jun. 2018.
- [37] L. Hewing, J. Kabzan, and M. N. Zeilinger, "Cautious model predictive control using Gaussian process regression," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 6, pp. 2736–2743, Nov. 2020.
- [38] X. Xu *et al.*, "Learning-based predictive control for discrete-time nonlinear systems with stochastic disturbances," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6202–6213, Dec. 2018.
- [39] G. P. Incremona, A. Ferrara, and L. Magni, "MPC for robot manipulators with integral sliding modes generation," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 3, pp. 1299–1307, Jun. 2017.
- [40] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, Mar. 2006.
- [41] R. J. Vanderbei and D. F. Shanno, "An interior-point algorithm for nonconvex nonlinear programming," *Comput. Optim. Appl.*, vol. 13, no. 1, pp. 231–252, Apr. 1999.
- [42] C. Büskens and H. Maurer, "SQP-methods for solving optimal control problems with control and state constraints: Adjoint variables, sensitivity analysis and real-time control," *J. Comput. Appl. Math.*, vol. 120, no. 1-2, pp. 85–108, Aug. 2000.
- [43] A. Astudillo *et al.*, "Position and orientation tunnel-following NMPC of robot manipulators based on symbolic linearization in sequential convex quadratic programming," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2867–2874, Apr. 2022.
- [44] M. V. Minniti *et al.*, "Adaptive CLF-MPC with application to quadrupedal robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 1, pp. 565–572, Jan. 2022.
- [45] Y. Wang *et al.*, "Incremental model predictive control exploiting time-delay estimation for a robot manipulator," *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 6, pp. 2285–2300, Nov. 2022.

- [46] T. Faulwasser *et al.*, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1505–1511, Jul. 2016.
- [47] S. Roy *et al.*, "Overcoming the underestimation and overestimation problems in adaptive sliding mode control," *IEEE/ASME Trans. Mechatron.*, vol. 24, no. 5, pp. 2031–2039, Oct. 2019.
- [48] Z. Zhang, M. Leibold, and D. Wollherr, "Integral sliding-mode observer-based disturbance estimation for Euler-Lagrangian systems," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 6, pp. 2377–2389, Nov. 2020.
- [49] S. Gros *et al.*, "From linear to nonlinear MPC: Bridging the gap via the real-time iteration," *Int. J. Control*, vol. 93, no. 1, pp. 62–80, Jan. 2020.
- [50] C. Mastalli *et al.*, "Crocodyl: An efficient and versatile framework for multi-contact optimal control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Paris, France, May 2017, pp. 2536–2542.
- [51] A. Zanelli *et al.*, "FORCES NLP: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs," *Int. J. Control*, vol. 93, no. 1, pp. 13–29, Jan. 2020.
- [52] H. J. Ferreau *et al.*, "qpOASES: A parametric active-set algorithm for quadratic programming," *Math. Program. Comput.*, vol. 6, no. 4, pp. 327–363, Nov. 2014.
- [53] K. Cheshmi *et al.*, "NASOQ: Numerically accurate sparsity-oriented QP solver," *ACM Trans. Graph.*, vol. 39, no. 4, pp. 1–17, Aug. 2020.
- [54] S. Yu *et al.*, "Continuous finite-time control for robotic manipulators with terminal sliding mode," *Automatica*, vol. 41, no. 11, pp. 1957–1964, Nov. 2005.
- [55] C. Shen, Y. Shi, and B. Buckham, "Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5796–5805, Jul. 2018.
- [56] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*. 3rd ed. Menlo Park, CA: Addison-Wesley, 1998.
- [57] M. Liu, Y. Tan, and V. Padois, "Generalized hierarchical control," *Auton. Robot.*, vol. 40, no. 1, pp. 17–31, Jan. 2016.
- [58] D. He, L. Wang, and J. Sun, "On stability of multiobjective NMPC with objective prioritization," *Automatica*, vol. 57, pp. 189–198, Jul. 2015.
- [59] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge Univ. Press, 2004.
- [60] H. Jeong and I. Lee, "Optimization for whole body reaching motion without singularity," *Int. J. Precis. Eng. Manuf.*, vol. 23, no. 6, pp. 639–651, Jun. 2022.
- [61] D. Di Vito, C. Natale, and G. Antonelli, "A comparison of damped least squares algorithms for inverse kinematics of robot manipulators," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6869–6874, Jul. 2017.
- [62] P. Falugi and D. Q. Mayne, "Getting robustness against unstructured uncertainty: A tube-based MPC approach," *IEEE Trans. Automat. Control*, vol. 59, no. 5, pp. 1290–1295, May 2014.
- [63] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Syst. Mag.*, vol. 36, no. 6, pp. 30–44, Dec. 2016.
- [64] KUKA Robot Group, *LBR iiwa 7 R800, LBR iiwa 14 R820 Specification, 5th edition*. Augusburg, Germany, 2015.
- [65] P. Corke, "Robot manipulator capability in MATLAB: A tutorial on using the robotics system toolbox [Tutorial]," *IEEE Robot. Autom. Mag.*, vol. 24, no. 3, pp. 165–166, Sep. 2017.
- [66] J. A. Andersson *et al.*, "CasADi – A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, Mar. 2019.
- [67] A. Bambade *et al.*, "PROX-QP: Yet another quadratic programming solver for robotics and beyond," in *Proc. Robotics: Sci. Syst. (RSS)*, New York, USA, Jun. 2022.



Yongchao Wang received the M.Sc. degree in control science and engineering from Xidian University, China, in 2016. He is currently pursuing the Ph.D. degree in automatic control with the Chair of Automatic Control Engineering, Technical University of Munich, Germany. He is also a Research Associate with the Chair of Automatic Control Engineering, Technical University of Munich. His research interests include adaptive control, robust nonlinear control, optimal control, backstepping, and robotics.



Yang Liu received the B.Sc. degree in automation from East China University of Science and Technology, China, in 2019. He is currently pursuing the M.Sc. degree in electrical and computer engineering with the Chair of Automatic Control Engineering, Technical University of Munich, Germany. His research interests include model predictive control, optimal control and robotics.



Marion Leibold received the Diploma degree in applied mathematics and the Ph.D. degree in motion planning and control of legged robots from Faculty of Electrical Engineering and Information Technology, Technical University of Munich, Germany, in 2002 and 2007, respectively.

She is currently a Senior Researcher with the Chair of Automatic Control Engineering, Faculty of Electrical Engineering and Information Technology, Technical University of Munich. Her research interests include optimal control and nonlinear control theory, and the applications to robotics.



Martin Buss (Fellow, IEEE) received the Diploma Engineering degree in electrical engineering from the Technical University of Darmstadt, Darmstadt, Germany, in 1990, the Ph.D. degree in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1994.

From 1994 to 1995, he was a Post-Doctoral Researcher with the Department of System Engineering, Australian National University, Canberra, Australia. From 1995 to 2000, he was a Senior Research Assistant and a Lecture with the Chair of Automatic Control Engineering, Faculty of Electrical Engineering and Information Technology, Technical University of Munich. From 2000 to 2003, he was a Full Professor, the Head of Control Systems Group, and the Deputy Director of the Institute of Energy and Automation Technology, Faculty IV, Electrical Engineering and Computer Science, Technical University of Berlin, Berlin, Germany. Since 2003, he has been a Full Professor with the Chair of Automatic Control Engineering, Faculty of Electrical Engineering and Information Technology, Technical University of Munich, where he has been with the Medical Faculty since 2008. His research interests include automatic control, mechatronics, multimodal human system interfaces, and optimization.



Jinoh Lee (Senior Member, IEEE) received the B.S. degree in mechanical engineering from Hanyang University, Seoul, South Korea, in 2003 (awarded Summa Cum Laude), and the M.Sc. and Ph.D. degrees in mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012. He held Postdoctoral position (2012–2017) and Researcher position (2017–2020) at the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genoa, Italy. He is currently a Research Scientist with the

Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany, and is Adjunct Professor with Mechanical Engineering Department, KAIST. His professional is about robotics and control engineering, which include manipulation of highly redundant robots such as dual-arm and humanoids, robust control of nonlinear systems and compliant robotic system control for safe human-robot interaction. Dr. Lee is a member of the IEEE Robotics and Automation, the IEEE Control Systems, and the IEEE Industrial Electronics Societies. He is also a Senior Editor of the IROS Conference Paper Review Board (CPRB) and IEEE International Conference on Biomedical Robotics and Biomechanics (BioRob).