IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 24, 2021, accepted February 16, 2021, date of publication March 9, 2021, date of current version March 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3064151

A Whole-Body Control Framework Based on the Operational Space Formulation Under Inequality Constraints via Task-Oriented Optimization

YISOO LEE[1], SANGHYUN KIM[2], JAEHEUNG PARK[3], (Member, IEEE), NIKOS TSAGARAKIS[4], (Member, IEEE), AND JINOH LEE[5], (Senior Member, IEEE)

[1]Center for Intelligent and Interactive Robotics, Korea Institute of Science and Technology (KIST), Seoul 02792, South Korea
[2]System Engineering Research Division, Department of AI Machinery, Korea Institute of Machinery and Materials (KIMM), Daejeon 34103, South Korea
[3]Graduate School of Convergence Science and Technology, Seoul National University, Suwon-si 16229, South Korea
[4]Humanoids and Human Centered Mechatronics, Istituto Italiano di Tecnologia, 16163 Genoa, Italy
[5]Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany

Corresponding author: Jinoh Lee (jinoh.lee@dlr.de)

This work was supported in part by the Korea Institute of Science and Technology Institutional Programs under Grant 2K02660, in part by the European Union's Horizon 2020 Research and Innovation Programme under Grant 779963 EUROBENCH, in part by the National Research Foundation (NRF) of Korea funded by the Ministry of Education and the Korea Government [Ministry of Science and ICT (MSIT)] under Grant 2019R1A6A3A03033473 and Grant 2021R1A2C3005914, and in part by the National Research Council of Science and Technology under the Research and Development Program of MSIT through the Research Project (Development of Core Machinery Technologies for Autonomous Operation and Manufacturing).

ABSTRACT This paper presents practical enhancements of the operational space formulation (OSF) to exploit inequality constraints for whole-body control of a high degree of freedom robot with a floating base and multiple contacts, such as humanoids. A task-oriented optimisation method is developed to obtain a feasible torque resolution solely for task variables based on the OSF, which effectively reduces the number of optimisation variables. Interestingly, the proposed scheme amends assigned tasks on demand of satisfying inequality conditions, while dynamic consistency among contact-constrained tasks is preserved. In addition, we propose an efficient algorithm structure ameliorating real-time control capability which has been a major hurdle to transplant optimisation methods into the OSF-based whole-body control framework. Control performance, the feasibility of the optimised solution, and the computation time of the proposed control framework are verified through realistic dynamic simulations of a humanoid. We also clarify the pros and cons of the proposed method compared with existing optimisation-based ones, which may offer an insight for practical control engineers to select whole-body controllers necessitated from the desired application.

INDEX TERMS The operational space formulation (OSF), highly redundant robots, whole-body control, inequality constraints.

I. INTRODUCTION

Rendering dexterous motion of a humanoid robot demands that its high number of degrees of freedom (DoFs) with the unactuated floating base can be sophisticatedly controlled with several contacts. The *operational space formulation* (OSF) is one of the classic yet effective methods to deal with the redundant robots [1], which can compute optimised joint torque solutions to achieve multiple tasks in the given task space by assigning priority. Based on the OSF, the hierarchical whole-body control structure guaranteeing dynamic consistency in humanoid robots is established in [2], [3]. Authors in [4] further develop *the operational space-based whole-body control* (OS-WBC) framework for the floating base legged robot operated under multiple contact situation respecting contact constraints.

The OS-WBC framework offers intuitive *task-oriented* resolution, and moreover, rigorous *dynamic consistency*[1] for the under-actuated and contact-constrained robot

The associate editor coordinating the review of this manuscript and approving it for publication was Yangmin Li.

[1]With the most accurate definition in [5], forces contained in the null-space do not affect the operational space force or acceleration while producing joint acceleration energy minimised torque solution; and the resulting operational space inertia matrix eliminates dynamic coupling among hierarchical tasks at the acceleration level.

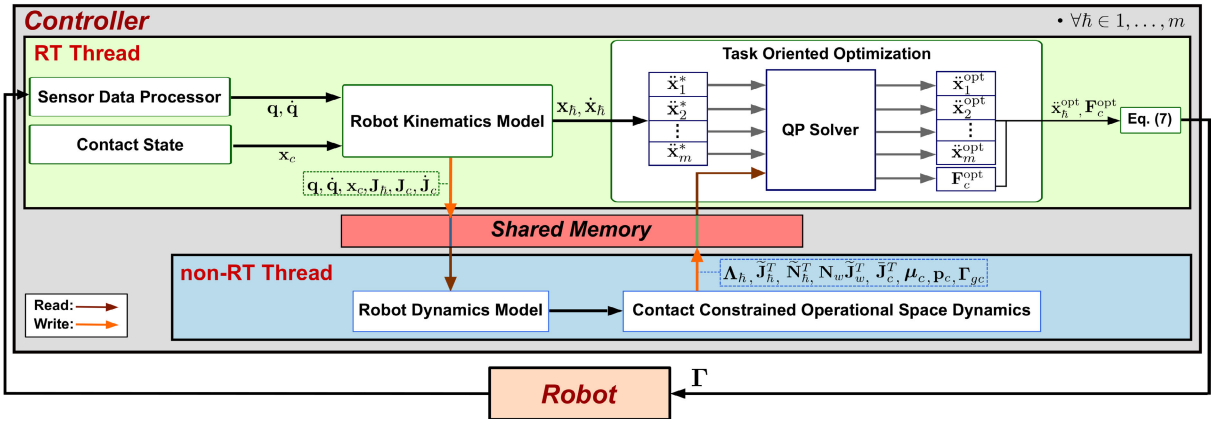


FIGURE 1. A block diagram describes the algorithm structure of the proposed WBC distributed into RT and non-RT threads (See details in Section III).

dynamics through the inertia-weighted pseudo inverse, called dynamically-consistent inverse. Owing to recent advances in torque controlled robot hardware and dynamic modeling technique, the effectiveness of the OS-WBC has been experimentally verified with actual biped robots [6]–[9]. Similarly, computationally efficient methods to solve the inverse dynamics based on the orthogonal decomposition approach are proposed in [10], [11].

However, despite the advantageous nature of the OS-WBC framework, it cannot be explicitly extended to consider *inequality constraints*, which hinders more practical use under WBC scenarios. Many physical constraints in the WBC problem are often to be formulated as inequality conditions and essential to accomplish realistic operation objectives; for example, the legged robot locomotes or balances by creating stable contact wrenches to lie within the Coulomb friction cone and maintaining the center of pressure (CoP) within a support polygon. Since such inequality constraints generally limit the feasible region of solutions, numerical optimisation-based approaches have been employed in WBC to negotiate given control variables such as reference joint torque and acceleration with slack variables.

Among optimisation-based methods, quadratic programming (QP) has recently become popular to control complex robotic systems such as underactuated legged robots, thanks to its straightforwardness for implementing and obtaining an optimised solution under both equality and inequality constraints [12], [13]. In the QP formulation, the control torque solution for multiple tasks is optimised with different weightings, which introduces a *soft* hierarchy among the tasks [14]. On that account, the QP is incorporated with WBC schemes to compute joint torques with the dynamics and kinematics constraints within fast control frequency, e.g., 1kHz, and feasibility of the QP-based WBC approaches is verified by actual experiments with humanoid robots [15]–[19]. The hierarchical quadratic programming (HQP) further focuses on the strictness of the hierarchical execution when conflicts occur among the tasks. This successfully demonstrates to perform WBC tasks with strict hierarchy even under inequality constraints [20]–[23]. Note that, as an expense for the strict priority, iterative optimisation is obliged for each task; this escalates computation effort as a number of tasks increases, yet regardless of a number of inequality constraints [20].

Whereas, optimisation schemes such as QP cannot be easily plugged into the OS-WBC framework mainly due to its inherent large computation of dynamically-consistent parameters in whole-body dynamics. Accordingly, this paper aims to provide an explicit extension of the OS-WBC framework with consideration of inequality constraints. In particular, the proposed controller is complied with the task-oriented design rooted in the OSF, which may benefit those who have traced the original OSF-based controller as seeking a way to control tasks under inequality constraints.

To cope with multiple contact-constrained tasks yet to be efficient enough for a realtime (RT) implementation, the followings are considered in this paper:

- an optimisation method is developed to produce the solution solely for operational-space task commands—task accelerations and forces, which effectively reduce the dimension of the optimisation variables. This gives optimal amendments for holistic task commands on demand of satisfying inequality conditions, yet still preserves the hierarchical structure and dynamic consistency among the optimised tasks. Note that it pursues the task-oriented philosophy of the original OSF, thus called *task-oriented optimisation* in this paper;
- inspired by the practical observation that contact-constrained dynamics changes slower than the sampling frequency of the control loop, we propose an efficient algorithm structure with a multi-thread computation, where the optimisation-based control loop runs in hard RT while the dynamic parameters are updated in a non-RT loop as shown in Fig. 1. This accommodates inherently high computation demands in the OS-WBC framework and thus enables the optimisation-embedded control with RT-robustness; and

- additionally, to clarify differences and pros and cons over other optimisation-based WBCs, we comparatively analyse the conventional QP-based, HQP-based, and proposed method. This is to give better insight into the proposed one as well as to offer a guideline for practical engineers to make a choice of methods suitable for their application.

The rest of the paper is organised as follows: Section II gives a summary of the OS-WBC as a background and Section III proposes the enhanced OS-WBC framework with the task-oriented optimisation and efficient algorithm structure. In Section IV, control performance, the feasibility of the optimised solution, and computation time are validated on a 28 degrees-of-freedom (DoFs) humanoid robot performing multi-contact tasks. Section V tries to answer potential questions by comparing the proposed method with other optimisation-based ones and Section VI finally draws the conclusion.

II. BACKGROUND ON THE OS-WBC

This section introduces the OS-WBC framework [4], [6] as a starting point of this paper.

The rigid-body dynamics equation of the floating base robot, which has $n = (k + 6)$ DoFs, k joints, and c DoFs contact, can be expressed as

$$\mathbf{A}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_c^T\mathbf{F}_c = \mathbf{S}^T\boldsymbol{\Gamma}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ denotes the inertia matrix, $\mathbf{q} \in \mathbb{R}^n$ the joint angle vector, $\mathbf{b} \in \mathbb{R}^n$ the Coriolis/centrifugal force vector, $\mathbf{g} \in \mathbb{R}^n$ the gravity force vector, $\mathbf{J}_c \in \mathbb{R}^{c\times n}$ the contact Jacobian matrix defined as $\dot{\mathbf{x}}_c = \mathbf{J}_c\dot{\mathbf{q}}$, $\mathbf{x}_c \in \mathbb{R}^c$ the contact positions and orientations, $\mathbf{F}_c \in \mathbb{R}^c$ the contact wrench vector, $\mathbf{S} \in \mathbb{R}^{k\times n}$ a selection matrix to exclude un-actuated joints, and $\boldsymbol{\Gamma} \in \mathbb{R}^k$ the actual joint torque vector.

With a stationary contact constraint $\ddot{\mathbf{x}}_c{=}\dot{\mathbf{x}}_c{=}\mathbf{0}$, the *contact constrained dynamics* of the given operational space coordinate $\mathbf{x}$ can be described as follows:

$$\Lambda\ddot{\mathbf{x}} + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F},$$

with

$$\begin{cases} \Lambda = \{\mathbf{J}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{J}_c^T\bar{\mathbf{J}}_c^T)\mathbf{J}^T\}^{-1}, \\ \boldsymbol{\mu} = \bar{\mathbf{J}}^T\mathbf{b} - \Lambda\dot{\mathbf{J}}\dot{\mathbf{q}} + \Lambda\mathbf{J}\mathbf{A}^{-1}\mathbf{J}_c^T\Lambda_c\dot{\mathbf{J}}_c\dot{\mathbf{q}}, \\ \mathbf{p} = \bar{\mathbf{J}}^T\mathbf{g}, \end{cases} \tag{2}$$

where Λ, $\boldsymbol{\mu}$, $\mathbf{p}$, and $\mathbf{F}$ denote the operational space inertia matrix, the operational space Coriolis/centrifugal force vector, the operational space gravity force vector, and the operational space wrench vector, respectively, while $\mathbf{J}$ denotes the Jacobian matrix for the given operational space task and $\bar{\bullet}$ is the dynamically consistent inverse of $\bullet$. Here, the weighted pseudo-inverse is employed for the kinematic redundancy, i.e., DoFs of the task $\mathbf{x}$ is smaller than k. This paper considers the weighting matrix $\mathbf{W} = \mathbf{S}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{J}_c^T\bar{\mathbf{J}}_c^T)\mathbf{S}^T$, which minimises the joint acceleration energy.

In the underactuated robotic system, one can compute $\boldsymbol{\Gamma}$ to produce the given force $\mathbf{F}$ as follows [24]:

$$\boldsymbol{\Gamma} = (\overline{\bar{\mathbf{J}}^T\mathbf{S}^T})\mathbf{F} = \widetilde{\mathbf{J}}^T\mathbf{F}, \tag{3}$$

The hierarchical control structure then can be formulated with the null-space projection matrix defined as $\widetilde{\mathbf{N}}^T = \mathbf{I} - \widetilde{\mathbf{J}}^T\widetilde{\bar{\mathbf{J}}}^T$. Accordingly, the controller with m hierarchical tasks can be formulated as

$$\Gamma_t = \widetilde{\mathbf{J}}_1^T\mathbf{F}_1^* + \sum_{i=2}^{m}\{\prod_{j=1}^{i-1}(\widetilde{\mathbf{N}}_j^T\widetilde{\mathbf{J}}_i^T\mathbf{F}_i^*)\}, \tag{4}$$

where $\Gamma_t \in \mathbb{R}^k$ denotes the tasks control torque vector, $\mathbf{F}_\hbar^*$ the reference force vector at the operational space, while the subscript indicates the task priority as $\hbar \in 1, \ldots, m$. From (2), one can then control the task with the reference acceleration $\ddot{\mathbf{x}}_\hbar^*$ to create $\mathbf{F}_\hbar^*$ as follows:

$$\mathbf{F}_\hbar^* = \boldsymbol{\Lambda}_\hbar\ddot{\mathbf{x}}_\hbar^* + \boldsymbol{\mu} + \mathbf{p}. \tag{5}$$

The contact wrench can be as desired without disturbing the given tasks, when contact redundancy exists, i.e., $c > 6$, by creating the contact wrench distribution torque $\boldsymbol{\Gamma}_c \in \mathbb{R}^k$ with the modification of contact wrench components in the null-space of $\mathbf{W}$ as follows:

$$\boldsymbol{\Gamma}_c = \mathbf{N}_w\widetilde{\mathbf{J}}_w^T\mathbf{F}_c^*, \tag{6}$$

where $\mathbf{N}_w \in \mathbb{R}^{k\times(c-6)}$ denotes the null-space projection matrix of $\mathbf{W}$, $\widetilde{\mathbf{J}}_w^T \in \mathbb{R}^{(c-6)\times(c-6)}$ is the dynamically-consistent inverse of $\mathbf{S}_c\mathbf{J}_c^T\mathbf{S}^T\mathbf{N}_w$, $\mathbf{S}_c \in \mathbb{R}^{(c-6)\times c}$ the selection matrix corresponding to modified contact wrench components, and $\mathbf{F}_c^* \in \mathbb{R}^{(c-6)}$ the reference contact wrench distribution vector.

Finally, with given control references $\ddot{\mathbf{x}}_\hbar^*$ and $\mathbf{F}_c^*$, the reference joint torque Γ is computed as follows:

$$\Gamma = \underbrace{\widetilde{\mathbf{J}}_1^T\boldsymbol{\Lambda}_1\ddot{\mathbf{x}}_1^* + \sum_{i=2}^{m}\{\prod_{j=1}^{i-1}(\widetilde{\mathbf{N}}_j^T\widetilde{\mathbf{J}}_i^T\boldsymbol{\Lambda}_i\ddot{\mathbf{x}}_i^*)\}}_{\Gamma_t,\ \text{Eq. (4)}} + \Gamma_{gc} + \underbrace{\mathbf{N}_w\widetilde{\mathbf{J}}_w^T\mathbf{F}_c^*}_{\Gamma_c,\ \text{Eq. (6)}}, \tag{7}$$

where $\Gamma_{gc} \in \mathbb{R}^k$ is the compensation torque of gravity and Coriolis/centrifugal forces, which includes the operational force terms $(\boldsymbol{\mu} + \mathbf{p})$.

From the robot dynamics and the control solution, the applying wrench at the contact link can be computed as

$$\mathbf{F}_c = \bar{\mathbf{J}}_c^T\mathbf{S}^T\Gamma - \boldsymbol{\mu}_c - \mathbf{p}_c, \tag{8}$$

where $\boldsymbol{\mu}_c$, $\mathbf{p}_c$ are vectors of the Coriolis/centrifugal force and the gravity force projected on the contact space, respectively.

III. THE PROPOSED WBC FRAMEWORK

This section presents an enhanced OS-WBC framework which offers optimised control torques for hierarchical tasks associated with inequality constraints while achieving both dynamic consistency and low computation cost in RT control loop in implementation.

A. TASK-ORIENTED OPTIMISATION

1) PROBLEM SETTING

The OS-WBC is formulated as the following optimisation problem:

$$\underset{\ddot{\mathbf{x}}_\hbar^{\text{opt}},\mathbf{F}_c^{\text{opt}}}{\text{minimise}} \sum_{i=1}^{m+1} \frac{1}{2} \boldsymbol{\chi}_i^T \mathbf{H}_i \boldsymbol{\chi}_i, \tag{9}$$

where the vector $\boldsymbol{\chi}_i$ are given as

$$\begin{cases} \boldsymbol{\chi}_\hbar = \ddot{\mathbf{x}}_\hbar^{\text{opt}} - \ddot{\mathbf{x}}_\hbar^*, \quad \hbar \in 1, \ldots, m, & (10) \\ \boldsymbol{\chi}_{m+1} = \mathbf{S}_f \mathbf{F}_c^{\text{opt}}, & (11) \end{cases}$$

where $\mathbf{S}_f$ denotes the selection matrix excluding the elements corresponding to the normal direction of the contact plane, and $\mathbf{H}_i$ denotes a weighting matrix for the optimisation, set as a positive constant diagonal matrix. The key idea of this formulation is to obtain optimised operational-space task commands $\ddot{\mathbf{x}}_\hbar^{\text{opt}}$ and $\mathbf{F}_c^{\text{opt}}$ as the output of optimisation, which means that the assigned tasks themselves are amended on demand of satisfying constraints, thus called *task-oriented optimisation* in this study.

More specifically, in (10), the minimisation of $\boldsymbol{\chi}_\hbar$ gives the optimal $\ddot{\mathbf{x}}_\hbar^{\text{opt}}$ as close as the computed reference acceleration command in (5) for executing the desired tasks within feasible solutions. Equation (11) minimises the moment and tangential forces in the contact wrench components to establish stable contacts, which finds the most stable CoP position and the least slipping condition within feasible solutions.

2) EQUALITY CONSTRAINTS

First, the OS-WBC torque condition (7) is set as the equality constraint as

$$\Gamma = \widetilde{\mathbf{J}}_1^T \boldsymbol{\Lambda}_1 \ddot{\mathbf{x}}_1^{\text{opt}} + \sum_{i=2}^{m} \{ \prod_{j=1}^{i-1} (\widetilde{\mathbf{N}}_j^T \widetilde{\mathbf{J}}_i^T \boldsymbol{\Lambda}_i \ddot{\mathbf{x}}_i^{\text{opt}}) \} + \mathbf{N}_w \widetilde{\mathbf{J}}_w^T \mathbf{F}_c^{\text{opt}} + \Gamma_{gc}. \tag{12}$$

Second, the contact wrench condition $\mathbf{F}_c$ is set as another equality constraint, given by substituting (12) into (8) as

$$\mathbf{F}_c = \bar{\mathbf{J}}_c^T \mathbf{S}^T \Big[\widetilde{\mathbf{J}}_1^T \boldsymbol{\Lambda}_1 \ddot{\mathbf{x}}_1^{\text{opt}} + \sum_{i=2}^{m} \{ \prod_{j=1}^{i-1} (\widetilde{\mathbf{N}}_j^T \widetilde{\mathbf{J}}_i^T \boldsymbol{\Lambda}_i \ddot{\mathbf{x}}_i^{\text{opt}}) \} + \Gamma_{gc} \Big] + \mathbf{F}_c^{\text{opt}} - \boldsymbol{\mu}_c - \mathbf{p}_c. \tag{13}$$

Third, the following equality constraint is applied to preserve the contact wrench distribution by $\mathbf{F}_c^{\text{opt}}$, preventing the resultant contact wrench to be changed:

$$\begin{bmatrix} \mathbf{R}_{c_1} & \mathbf{O} & \cdots & \mathbf{R}_{c_N} & \mathbf{O} \\ \hat{\mathbf{P}}_{c_1} & \mathbf{R}_{c_1} & \cdots & \hat{\mathbf{P}}_{c_N} & \mathbf{R}_{c_N} \end{bmatrix} \mathbf{F}_c^{\text{opt}} = \mathbf{G} \mathbf{F}_c^{\text{opt}} = \mathbf{0}, \tag{14}$$

where $\mathbf{G} \in \mathbb{R}^{6 \times c}$ denotes the grasp map matrix for N-number of plane contacts; c_j the j-th contact link; $\mathbf{R}_{c_j} \in \mathbb{R}^{3 \times 3}$; the rotation matrix of c_j with respect to the given coordinate; and $\hat{\mathbf{P}}_{c_j} \in \mathbb{R}^{3 \times 3}$ the skew symmetric matrix of the distance vector from the given coordinate to position of c_j.

3) INEQUALITY CONSTRAINTS

To preserve rigid and stable contacts, inequality constraints are given as

$$\left.\begin{array}{rl} f_{c_j,z} & < \varepsilon \\ \mu_s |f_{c_j,z}| & > (f_{c_j,x}^2 + f_{c_j,y}^2)^{1/2} \\ \mu_s |f_{c_j,z}| & > m_{c_j,z} \\ -\dfrac{m_{c_j,y}}{f_{c_j,z}} & \in [x_p^{\min}, x_p^{\max}] \\ \dfrac{m_{c_j,x}}{f_{c_j,z}} & \in [y_p^{\min}, y_p^{\max}] \end{array}\right\} \forall j \in 1, \ldots, N, \tag{15}$$

where $f_{c_j,\bullet}$ denotes the $\bullet$-axis contact force at c_j, x and y the tangential direction at the contact, and z the normal direction to the contact plane, while $m_{c_j,\bullet}$ is the contact moment at c_j along $\bullet$-axis (x, y, z-axes), $\varepsilon < 0$ the threshold value, $\mu_s > 0$ the static friction coefficient, and $[x_p^{\min}, x_p^{\max}]$, $[y_p^{\min}, y_p^{\max}]$ boundaries of the CoP in the contact line or plane. Note that the contact conditions (15) are described in each local coordinate at the contact. In addition, one can also consider limits of the task reference and joint torque as inequality constraints as follows:

$$\ddot{\mathbf{x}}_\hbar^{\text{opt}} \in [\ddot{\mathbf{x}}_\hbar^{\min}, \ddot{\mathbf{x}}_\hbar^{\max}], \quad \Gamma \in [\Gamma^{\min}, \Gamma^{\max}],$$

where $\ddot{\mathbf{x}}_\hbar^{\min}$, $\ddot{\mathbf{x}}_\hbar^{\max}$ are vectors of the task acceleration limit, and $\Gamma^{\min}$, $\Gamma^{\max}$ the vectors of the joint torque limit. Also, additional constraints can be added in the optimisation formulation while keeping the aforementioned constraints (12)-(15).

Finally, the task-oriented optimisation problem can be formulated as a general QP form to be solved by a numerical QP solver. The whole-body control torque then can be obtained by substituting the optimisation results ($\ddot{\mathbf{x}}_\hbar^{\text{opt}}$, $\mathbf{F}_c^{\text{opt}}$) into (7) as $\ddot{\mathbf{x}}_\hbar^* \triangleq \ddot{\mathbf{x}}_\hbar^{\text{opt}}$, $\mathbf{F}_c^* \triangleq \mathbf{F}_c^{\text{opt}}$. It is worthwhile to notice that by setting hierarchical inverse dynamics resolution from the OS-WBC formulation as equality constraints (12) and (13), the acquired torque solution always complies with a hierarchy, where the higher priority tasks are not affected by lower priority tasks even with under-actuation and multiple contacts (i.e., dynamically consistent), while the task commands are optimally rectified to satisfy other inequality constraints.

Remark 1: In the proposed control framework, not only - acceleration level tasks but also force-level tasks can be controlled. The force control of i-th priority task can be simply implemented by replacing the acceleration references regarding terms $\ddot{\mathbf{x}}_i^{\text{opt}}$ and $\ddot{\mathbf{x}}_i^*$ in (9) and (10) with force reference terms $\mathbf{F}_i^{\text{opt}}$ and $\mathbf{F}_i^*$, respectively; and replacing $\boldsymbol{\Lambda}_i \ddot{\mathbf{x}}_i^{\text{opt}}$ in (12) and (13) with $\mathbf{F}_i^{\text{opt}}$. Therefore, both the acceleration-level tasks and force-level tasks can be controlled in the proposed control framework. Note that pre- and post-multiplications of $\boldsymbol{\Lambda}_i^{-T}$ and $\boldsymbol{\Lambda}_i^{-1}$ with $\mathbf{H}_i$ can be performed to remove the influence of the difference in the dimension of acceleration and force when solving the QP optimisation.

Remark 2: Acceleration- and force-level control can be implemented in the same priority task adopting the hybrid motion/force control method [1] in the proposed control

framework. When the control reference vector of i-th priority task is composed with acceleration references and force references, $\ddot{\mathbf{x}}_i^{\text{opt}}$ and $\ddot{\mathbf{x}}_i^*$ in (9), (10), (12), and (13) can be respectively replaced with,

$$\mathbf{\Lambda}_i\mathbf{\Omega}\mathbf{f}_i^* + \tilde{\mathbf{\Omega}}\mathbf{f}_i^*, \text{ and } \mathbf{\Lambda}_i\mathbf{\Omega}\mathbf{f}_i^{\text{opt}} + \tilde{\mathbf{\Omega}}\mathbf{f}_i^{\text{opt}},$$

where $\mathbf{\Omega}$ and $\tilde{\mathbf{\Omega}}$ are the generalized task specification matrices defined in [1] which selecting components for acceleration- and force-level, respectively.

Remark 3: When the tasks are controlled with $\ddot{\mathbf{x}}_\hbar^{\text{opt}}$, the control force $\mathbf{F}_\hbar^{opt}$ can be expressed as follows:

$$\begin{aligned}\mathbf{F}_\hbar^{opt} &= \mathbf{\Lambda}_\hbar\ddot{\mathbf{x}}_\hbar^{opt} + \boldsymbol{\mu} + \mathbf{p} \\ &= \mathbf{\Lambda}_\hbar\ddot{\mathbf{x}}_\hbar^* + \mathbf{\Lambda}_\hbar(\ddot{\mathbf{x}}_\hbar^{opt} - \ddot{\mathbf{x}}_\hbar^*) + \boldsymbol{\mu} + \mathbf{p} \\ &= \mathbf{F}_\hbar^* + \mathbf{\Lambda}_\hbar(\ddot{\mathbf{x}}_\hbar^{opt} - \ddot{\mathbf{x}}_\hbar^*).\end{aligned}$$

This means that in addition to the force created by the feedback control term, $\mathbf{F}_\hbar^*$ in (5), the proposed method creates a correction force $\mathbf{\Lambda}_\hbar(\ddot{\mathbf{x}}_\hbar^{opt} - \ddot{\mathbf{x}}_\hbar^*)$ to satisfy inequality constraints during the task-oriented dynamically consistent optimisation.

4) WEIGHTING MATRIX

In the optimisation, weighting matrices, $\mathbf{H}_i$, play a critical role to obtain suitable control commands. In this paper, we propose to set $\mathbf{H}_i$ as $\mathbf{H}_\hbar \gg \mathbf{H}_{m+1}$ for each objective function, which induces that the optimisation for the hierarchical task execution, (10), becomes dominant. The weightings for tasks $\mathbf{H}_\hbar$ are set in consideration of the assigned priority as $\mathbf{H}_1 \geq \cdots \geq \mathbf{H}_m$. The QP optimisation result is then determined based on the difference in the value of the diagonal components of the weighting matrices. To give a clear difference in the level of task reference amendment between tasks, the difference in the diagonal values should be large enough. In general, it is enough to set 10^2 to 10^3 times between the weightings. For example, $\mathbf{H}_1 = 10^6\mathbf{I}$ and $\mathbf{H}_2 = 10^3\mathbf{I}$ can be applied when there are two hierarchical tasks.

B. PRACTICAL CONSIDERATION OF REALTIME CONTROL

1) COMPUTATIONAL COST REDUCTION

In this study, we propose to utilize a complete orthogonal decomposition (COD)-based calculation approach to reduce the computational cost of analytically achieved variables. The conventional OS-WBC [4], [6] are utilizing the singular value decomposition (SVD) which requires heavy computation cost to calculate the dynamically consistent inverse matrices in (3) and (6), and $\mathbf{N}_w$ in (6). The COD can serve the same role as the SVD, yet with lower computation effort when decomposing a rank-deficient matrix [25].

Based on the COD with Householder QR factorization, an arbitrary rank-deficient matrix $\mathbf{Y} \in \mathbb{R}^{\mathrm{m}\times\mathrm{n}}$ is decomposed as follows:

$$\mathbf{Y} = \mathbf{QRV}^T = \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\mathbf{R}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{v}_1^T \\ \mathbf{v}_2^T\end{bmatrix}, \tag{16}$$

where $\mathbf{Q} \in \mathbb{R}^{\mathrm{m}\times\mathrm{m}}$ and $\mathbf{V}^T \in \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$ are orthonormal matrices, and $\mathbf{R} \in \mathbb{R}^{\mathrm{m}\times\mathrm{n}}$ is a rank-deficient upper triangular matrix composed with $\mathbf{R}_1$ that is a full-rank upper triangular matrix. The unitary matrix $\mathbf{v}_2$ in (16) is the projector on the null-space of $\mathbf{Y}$. Therefore, the values in the vector space spanned by the column vectors of $\mathbf{v}_2$ will not affect that of $\mathbf{v}_1$. As a result, $\mathbf{v}_2$ can be utilized as $\mathbf{N}_w$ in (6), allowing better computational efficiency compared with that of SVD.

The dynamically-consistent inverse of the rank-deficient positive semi-definite matrix with the weighting $\mathbf{W}$ can be obtained through COD. From the decomposed matrix (16), since $\mathbf{Q}$ and $\mathbf{V}^T$ are unitary matrices and $\mathbf{R}_1$ is an invertible matrix, the pseudo-inverse of $\mathbf{Y}$ can be obtained as follows:

$$\mathbf{Y}^\dagger = \begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2\end{bmatrix}\begin{bmatrix}\mathbf{R}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{Q}_1^T \\ \mathbf{Q}_2^T\end{bmatrix} = \mathbf{v}_1\mathbf{R}_1^{-1}\mathbf{Q}_1^T.$$

Then, the least square solution for inverse problem of $\mathbf{Y}\boldsymbol{\alpha} = \boldsymbol{\beta}$ can be described as

$$\boldsymbol{\alpha} = \mathbf{Y}^\dagger\boldsymbol{\beta} + \mathbf{v}_2\boldsymbol{\alpha}_2 = \mathbf{v}_1\mathbf{R}_1^{-1}\mathbf{Q}_1^T\boldsymbol{\beta} + \mathbf{v}_2\boldsymbol{\alpha}_2, \tag{17}$$

where $\boldsymbol{\alpha}_2$ is any arbitrary vector that is projected in the null-space of the $\mathbf{Y}$. The minimum solution can be attained when the following partial derivative of the $\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{W}\boldsymbol{\alpha}$ becomes zero [26]:

$$\begin{aligned}\frac{\partial}{\partial\boldsymbol{\alpha}_2}\left(\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{W}\boldsymbol{\alpha}\right) &= \frac{1}{2}\frac{\partial(\mathbf{Y}^\dagger\boldsymbol{\beta} + \mathbf{v}_2\boldsymbol{\alpha}_2)^T\mathbf{W}(\mathbf{Y}^\dagger\boldsymbol{\beta} + \mathbf{v}_2\boldsymbol{\alpha}_2)}{\partial\boldsymbol{\alpha}_2} \\ &= \mathbf{v}_2^T\mathbf{W}(\mathbf{Y}^\dagger\boldsymbol{\beta} + \mathbf{v}_2\boldsymbol{\alpha}_2) = 0.\end{aligned} \tag{18}$$

From (18), $\boldsymbol{\alpha}_2$ is obtained as follows:

$$\boldsymbol{\alpha}_2 = -(\mathbf{v}_2^T\mathbf{W}\mathbf{v}_2)^\dagger\mathbf{v}_2^T\mathbf{W}\mathbf{Y}^\dagger\boldsymbol{\beta}. \tag{19}$$

Then, substituting (19) into (17) yields

$$\boldsymbol{\alpha} = \{\mathbf{I} - \mathbf{v}_2(\mathbf{v}_2^T\mathbf{W}\mathbf{v}_2)^\dagger\mathbf{v}_2^T\mathbf{W}\}\mathbf{Y}^\dagger\boldsymbol{\beta} = \bar{\mathbf{Y}}\boldsymbol{\beta}, \tag{20}$$

where $\bar{\mathbf{Y}}$ denotes the dynamically-consistent inverse of $\mathbf{Y}$ with $\mathbf{W}$, defined in Section II.

2) EFFICIENT ALGORITHM STRUCTURE

In this subsection, we establish an efficient programming method for the proposed WBC framework which renders parallel computing with two-threads to ensure RT safety in control. This takes advantage of multiple core computers—widespread in recent humanoid applications. While the COD-based calculation shown in the previous subsection effectively reduces computational load, the more DoFs with many hierarchies the robot has, the more challenging it is to complete the entire calculation within the allocated sampling period of an RT control loop.

In the proposed method, one can observe that the contact constrained dynamics is to be calculated, shown in Section II, which requires expensive computational cost. Whereas, interestingly, the parameter variation of those dynamic variables is relatively slower than the sampling time since it can be assumed that the robot dynamics is continuous and the

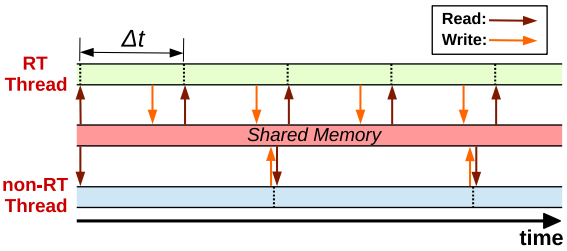
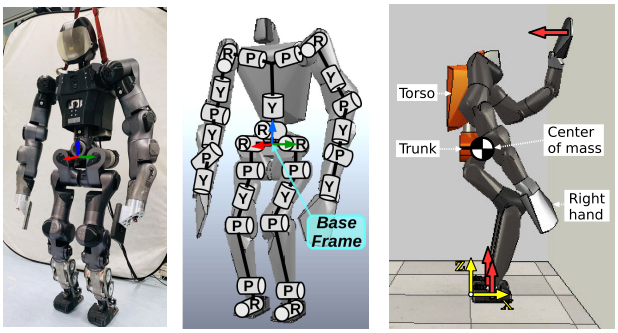


FIGURE 2. Computing process through shared memory, where Δt is the sampling period of the RT thread, while the brown colored arrow denotes read operation, and the orange colored arrow denotes write operation.



(a) (b) (c)

FIGURE 3. (a) The humanoid robot COMAN+, (b) its schematic kinematic structure, and (c) description of the global axis (yellow colored arrows), the controlled links (white colored dotted arrows), and contact normal directions (red colored arrows) in the simulation.

actual motion of the robot is not excessively fast, i.e., high-frequency update rate (e.g. above 1kHz) for those dynamic parameters is not necessary [27].

Therefore, the proposed WBC is mainly implemented in two RT and non-RT threaded software structure as depicted in Fig. 1. Hard RT requirements ensuring a control loop with the high-frequency update rate are run in the RT thread, which includes control command calculation with the task-oriented optimisation with a QP solver, sensory data acquisition, kinematics parameters calculation, and synchronous communication between the low-level motor controllers. On the other hand, the dynamic parameters in the contact constrained operational space are calculated in the non-RT thread, realised by cross-domain datagram protocol (XDDP) for the asynchronous communication between two threads, guaranteeing a lock-free inter-process communication [28]. This includes the operational space dynamics variables $(\Lambda_{\hbar}, \widetilde{\mathbf{J}}_{\hbar}^{T}, \widetilde{\mathbf{N}}_{\hbar}^{T})$, contact space dynamics variables $(\mathbf{N}_w\widetilde{\mathbf{J}}_w^T, \bar{\mathbf{J}}_c^T, \boldsymbol{\mu}_c, \mathbf{p}_c)$, and gravity and Coriolis/centrifugal force compensation torque (Γ_{gc}). For the sake of convenience, we call these variables computed in the non-RT thread as contact-constrained dynamics parameters in this study.

For the data exchange between the two threads, the shared memory is utilized as illustrated in Fig. 2. Both threads can perform calculations based on the most recent data recorded in shared memory without considering synchronization between them since both threads can access the shared memory, simultaneously. In every sampling period of the RT control loop, the RT thread reads shared memory, calculates the required variables after reading, writes them into the shared memory at the end of the calculation, and then waits for the initiation of the next control loop. The non-RT thread similarly works as the RT one, yet unlike the RT thread, computation results are updated whenever the writing to the shared memory is completed in the earliest time intervals.

This multi-threaded algorithm structure thus allows the proposed WBC framework to acquire the robust RT computation, where it solves only a single QP optimisation for the holistic hierarchical tasks within a sampling period of the RT thread. On the other hand, the other hierarchical optimisation method, i.e., HQP, has to solve multiple QP optimisations corresponding to the number of hierarchy levels for a single control loop.

Remark 4: The proposed multi-thread approach has little effect in the conventional HQP- and QP-based WBCs. This is because the multi-thread approach reduces computations required for calculating dynamics parameters for QP formulation. In the case of the conventional QP- and HQP-based methods, their QP formulation applies joint space dynamics in which parameters ($\mathbf{A}$, $\mathbf{b}$, and $\mathbf{g}$) can be computed very fastly. On the other hand, the QP formulation of the proposed method applies the contact constrained operational space dynamics which requires a large amount of computation. Thus, the multi-thread approach can effectively reduce computational cost only for the proposed WBC.

IV. VERIFICATION WITH NUMERICAL EXPERIMENTS

The proposed WBC framework is numerically validated by demonstrating control of a human-sized humanoid robot COMAN+ [29], shown in Fig. 3. The height of the robot is 1.6 m, weight is 67 kg, and the robot has 28 DoFs — 6 DoFs in each leg, 7 DoFs in each arm, and 2 DoFs in the waist. For realistic implementation and verification, the physics-based dynamic simulator V-REP with the physics engine Vortex is employed in the computer with 3.4 GHz quad-core processors and 32 GByte memory (without extra GPUs). The QP optimisation is performed by the solver implemented in the open-source library qpOASES.

Four scenarios #1-4 are considered to verify the control performance of the proposed method. In the first two scenarios, five hierarchical tasks (T1-T5) are assigned as summarized in Table 1, while three contacts are to be maintained with a plane contact condition between two feet and the ground, and between the left hand and wall as presented in Fig. 3c. The robot is controlled with a floating base, i.e., underactuated condition and two degrees-of-redundancy after the task assignment. The other two scenarios #3 and #4 have different task configurations, and details can be found in each relevant subsection. For all the scenarios, $\mathbf{H}_{m+1} = \mathbf{I}$ and $\mathbf{H}_{\hbar}$ are different for each scenario.

For control, acceleration commands of the operational-space task $(\ddot{\mathbf{x}}^*)$ are created to track desired position and velocity trajectories, $\mathbf{x}^d, \dot{\mathbf{x}}^d$, via the following PD-control

TABLE 1. Configuration of Hierarchical tasks (Scenarios #1-2).

Priority	Tasks	DoFs
1	T1–CoM position (x, y, z)	3
2	T2–Right hand position (x, y, z)	3
3	T3–Trunk orientation (roll, pitch, yaw)	3
4	T4–Torso orientation (roll, yaw)	2
5	T5–Right hand orientation (roll, pitch, yaw)	3

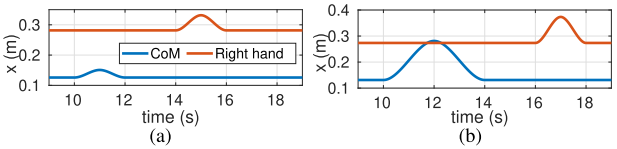


FIGURE 4. The reference trajectory of the CoM positioning (T1) and right hand positioning (T2) for (a) Scenarios #1, and (b) Scenarios #2.

scheme:

$$\ddot{\mathbf{x}}^* = k_p(\mathbf{x}^d - \mathbf{x}) + k_v(\dot{\mathbf{x}}^d - \dot{\mathbf{x}}),$$

where k_p, k_v denote proportional and derivative gains, respectively, which are chosen as k_p=100 and k_v=20 for all the tasks. The coefficients for inequality constraints (15) are set as ε=-30.0 N, μ_s=0.5, $[x_p^{\min}, x_p^{\max}]$=[−0.075, 0.075] m, $[y_p^{\min}, y_p^{\max}]$=[−0.05, 0.05] m for feet contacts, and ε=-30.0 N, μ_s=0.05, $[x_p^{\min}, x_p^{\max}]$=[−0.005, 0.005] m, $[y_p^{\min}, y_p^{\max}]$=[−0.005, 0.005] m for hand contact, and $[\boldsymbol{\Gamma}^{\min}, \boldsymbol{\Gamma}^{\max}]$ with actual actuator limits (e.g., ±200 Nm for knees). The sampling frequency is set to 1 kHz for the RT thread.

A. SCENARIO #1: CONTROL ACCURACY AND DYNAMIC CONSISTENCY

1) TASK SETTING

In this scenario, a center of mass (CoM) and right-hand motions are controlled along the x-axis as shown in Fig. 4(a): CoM for ±25 mm during t = 10-12 s, and the right hand for ±50 mm t = 14-16 s, while the motion in other directions is controlled to maintain the initial position and orientation. Here, the task is designed to be achieved without saturating inequality constraints. The proposed controller thus produces little amendment of acceleration references in the operational space in (10).

2) CONTROL ACCURACY

By investigating the error norms of the tasks shown in Fig. 5, one can confirm that the robot is accurately controlled by the proposed control. During t=9-19 s, the average position control error norms of CoM and right hand (T1,T2) are 0.045 mm and 0.281 mm, respectively. The average orientation control error norms of the trunk, torso, and right hand (T3-T5) are 0.0003, 0.0006, and 0.0192 rad, respectively.

3) DYNAMIC CONSISTENCY

To examine the effectiveness of the dynamic consistency of the method, the same tasks are controlled as those in Scenarios #1,2 while $\widetilde{\mathbf{J}}^T$ is obtained through the pseudo-inverse instead of the dynamically-consistent inverse. This replacement by the pseudo-inverse not only deteriorates the control performance but also affects a hierarchical control structure: as shown in Fig. 6, the control error norms drastically increase compared with the results with dynamically consistent inverse (Fig. 5); in addition, unwanted dynamic coupling behavior is observed in Fig. 7a, where the control error of x-axis CoM motion affects the other directions during t≈10-12.4 s. Also, during t≈14-18 s, unintended CoM motion occurs due to the influence of right-hand motion, i.e., the lower priority task. On the other hands, in the proposed method shown in Fig. 7b, the x-axis CoM motion is slightly perturbed, yet it is not affected by the lower priority task of right-hand motion, during t≈14-16 s. The dynamic consistency is thus satisfied.

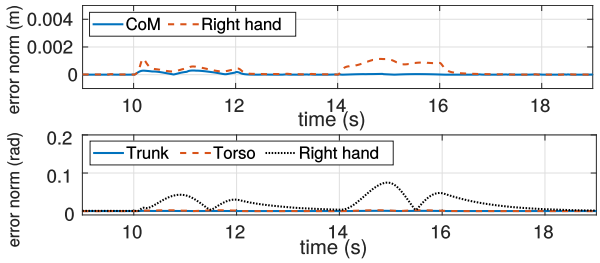


FIGURE 5. Plots of control performance with the proposed method in Scenario #1: position control results (upper) and orientation control results (lower).

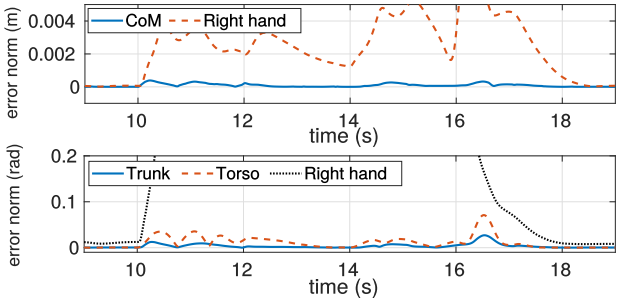


FIGURE 6. Plots of control performance with the pseudo-inverse method in Scenario #1: position tasks T1,T2 (upper), and orientation tasks T3-5 (lower). The peak value of the right hand orientation error is 0.988 rad.

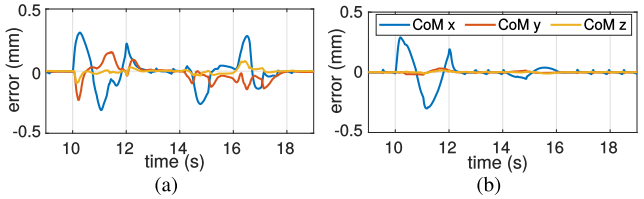
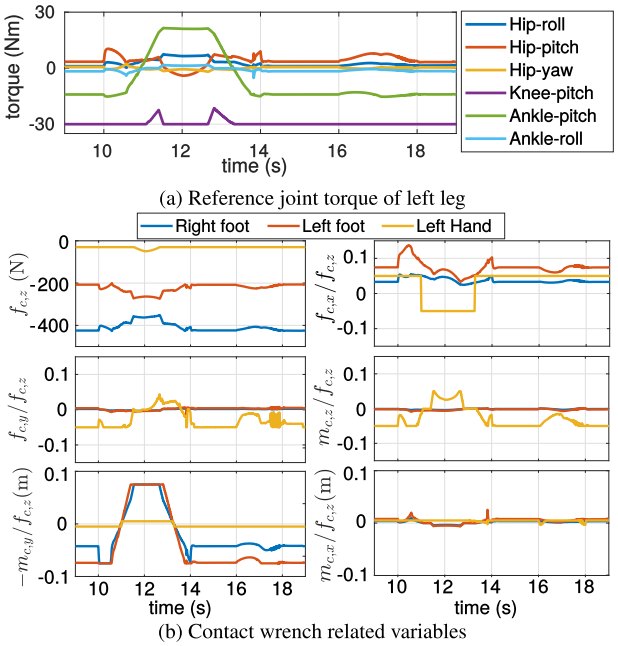


FIGURE 7. CoM control errors in Scenario #1: (a) pseudo-inverse based resolution and (b) dynamically-consistent resolution in the proposed WBC.

B. SCENARIO #2: FEASIBILITY UNDER INEQUALITY CONSTRAINTS

1) TASK SETTING

This simulation scenario demonstrates faster and wider ranged motions than the previous simulations in Scenario #1



(a) Reference joint torque of left leg

(b) Contact wrench related variables

FIGURE 8. optimisation results of the inequality constraints related variables when priority considered weighting is applied in Scenario #2.

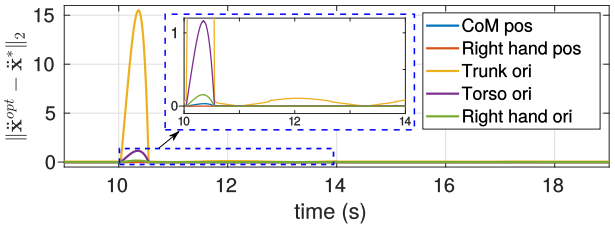


FIGURE 9. Optimal amendments of each tasks $\|\ddot{\mathbf{x}}^{\mathbf{opt}} - \ddot{\mathbf{x}}^{*}\|_2$ when priority considered weighting is applied in Scenario #2.

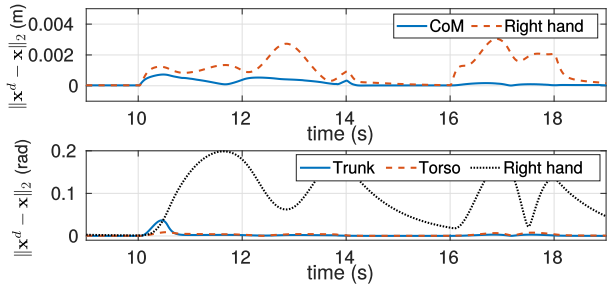


FIGURE 10. Differences between actual response and desired trajectory when priority considered weighting is applied in Scenario #2: position tasks T1,T2 (upper), and orientation tasks T3-5 (lower).

to validate the optimisation solution when variables are saturated at the constraint boundaries. As seen in Fig. 4b, CoM and the right hand are commanded to move ±150 mm during t = 10-14 s and ±100 mm during t = 16-18 s along the x-axis, respectively, while the other directions are regulated at the initial position and orientation. Additionally, the torque limits of the left leg joints are set narrower as [-30, 30] Nm.

2) PRIORITY CONSIDERED WEIGHTING

In this case, $\ddot{\mathbf{x}}_{\hbar}^{\text{opt}}$ is optimally amended from $\ddot{\mathbf{x}}_{\hbar}^{*}$ complying with inequality constraints. The weighting matrices for the task-oriented optimisation are composed with two levels as

$$\mathbf{H}_1 = \mathbf{H}_2 = 10^7\mathbf{I} \quad \gg \quad \mathbf{H}_3 = \mathbf{H}_4 = \mathbf{H}_5 = 10^3\mathbf{I}, \quad (21)$$

which offer optimised solutions for the first two tasks ($\ddot{\mathbf{x}}_{1,2}^{\text{opt}}$) to be kept as close to the desired commands ($\ddot{\mathbf{x}}_{1,2}^{*}$) as possible, while those for the lower priority tasks to be heavily amended. (The task weighting is further studied in the following subsection.)

Fig. 8 shows optimisation results of the generated torque reference of the left leg and contact wrench related variables. As given in inequality conditions, the created knee-pitch joint torque is limited at −30 Nm in Fig. 8a and it causes the magnitude of $f_{c,z}$ at the left foot to become smaller than that of the right foot as seen in Fig. 8b. In the same figure, one can also observe that the CoP in the x-axis ($-m_{c,y}/f_{c,z}$) of both feet are saturated to the assigned limit during t≈10-14 s via CoM motion control, and all the constraints on the left-hand contact are successfully limited, respecting the narrow range of the corresponding inequality constraints.

As intended with the task weight setting (21), at t≈ 10.04-10.55 s in Fig. 9, task commands for three lowest priority tasks $\ddot{\mathbf{x}}_{\hbar}^{*}$ are significantly amended, i.e., the difference between $\ddot{\mathbf{x}}_{\hbar}^{\text{opt}}$ and $\ddot{\mathbf{x}}_{\hbar}^{*}$ for T3-T5 becomes larger. In accordance with the task reference amendment, the gap between response ($\mathbf{x}_{\hbar}$) induced from the optimisation and the desired trajectory ($\mathbf{x}_{\hbar}^{d}$) for T3-T5 increases as presented in Fig. 10. This is an inevitable trade-off to obtain a feasible solution satisfying constraints. Nevertheless, the proposed method is capable of controlling the modifications by weightings assignment as users' intention. After t = 10.55 s, the task reference amendment is not significant, therefore $\mathbf{x}_{\hbar}^{d} - \mathbf{x}_{\hbar}$ indicates the control error itself. Note that this error is originated from the faster and wider motion assigned for the comparison, not from the optimisation performance, although it appears to be larger than previous result in Fig. 5

3) IDENTICAL WEIGHTING

As seen in the previous result, the selection of the weighting matrix in the cost function is not trivial in optimisation-based methods. Hence, we further investigate the impact of the task weighting by conducting comparative simulation with different weighting. Here, the weighting matrices are uniformly set for all tasks as $\mathbf{H}_{\hbar} = 10^7\mathbf{I}$.

The optimal task command $\ddot{\mathbf{x}}_{\hbar}^{\text{opt}}$ amended from $\ddot{\mathbf{x}}_{\hbar}^{*}$ and corresponding control results are shown in Figs. 11 and 12, respectively. The uniform task weighting leads to a feasible solution minimising $\sum_{i=1}^{5}(\ddot{\mathbf{x}}_{i}^{\text{opt}} - \ddot{\mathbf{x}}_{i}^{*})$ whose magnitude is approximately 7.5 times smaller than that with priority-considered weighting (21), while CoM and right-hand position errors (T1,T2) become larger while the trunk and torso orientation errors become smaller. Therefore, one can confirm that the proposed method offers flexibility to determine a strategy on how to treat constraints with consideration of respective tasks by designing $\mathbf{H}_{\hbar}$.

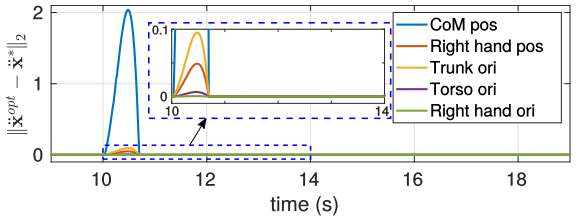


FIGURE 11. Optimal amendments of each tasks $\|\ddot{\mathbf{x}}^{\mathbf{opt}} - \ddot{\mathbf{x}}^{*}\|_2$ when identical weighting is applied in Scenario #2.

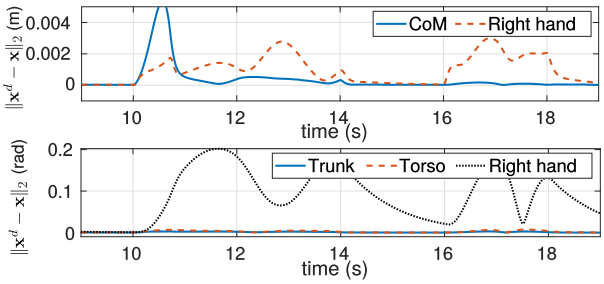


FIGURE 12. Differences between actual response and desired trajectory when identical weighting is applied in Scenario #2: position tasks T1,T2 (upper), and orientation tasks T3-5 (lower).

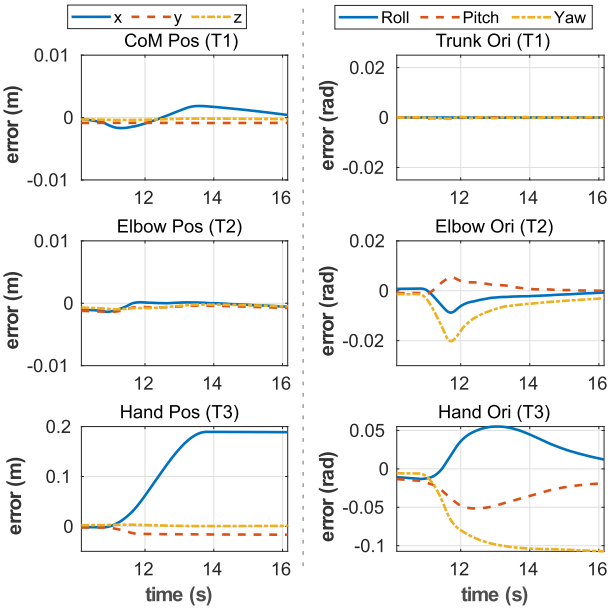


FIGURE 13. Control error in Scenario #3. Plots in the left column are the position control errors and plots in the right column are the orientation control errors.

TABLE 2. Configuration of Hierarchical tasks (Scenario #3).

Priority	Tasks	DoFs
1	T1–CoM position (x, y, z) & Trunk orientation (roll, pitch, yaw)	6
2	T2–Left elbow position & orientation (x, y, z, roll, pitch, yaw)	6
3	T3–Left hand position & orientation (x, y, z, roll, pitch, yaw)	6
4	T4–Joint angle (Posture)	28

C. SCENARIO #3: CONFLICTING TASKS

1) TASK SETTING

To verify the efficacy of the hierarchical control structure of the proposed control framework, we conducted a numerical experiment with four tasks as specified in Table 2, where two plane contacts are to be maintained between the feet and the ground. Note that we purposely set the third priority task, T3, to conflict against the second priority task, T2: the x-axis position of the left hand is commanded to track the cubic-spline trajectory for 0.2 m forward during 3 s from $t = 11$ s while y-, z-axis position, and 3 DoFs orientation are controlled to keep the initial position and orientation, where 6 DoFs task T3 cannot be controlled normally since there are only three DoFs (yaw, pitch, and yaw joints in order) outboard from the elbow link that is controlled by T2. For clarity of investigation, the tasks are designed not to be saturated with inequality constraints. Thus, the task commands are nearly the same after the task-oriented optimisation, i.e., $\ddot{x}_i^{\text{opt}} \approx \ddot{x}_i^{*}$. The weighting matrices are uniformly set for all tasks as $\mathbf{H}_\hbar = 10^7\mathbf{I}$.

2) RESULTS

As seen in Fig. 13, the higher priority tasks T1 and T2 are controlled with small errors while the lower priority task T3 conflicting against T2 does not follow the given trajectory. This means that the proposed controller can have the conflicting task T3 sacrificed in order to respect the task priority, as expected by the null-space projection approach-based hierarchical control structure. The control error of T3, especially for hand position in the x-axis, greatly increases during t=11-14 s since the hand cannot move to track the trajectory due to the limited DoFs assigned for T3. The slight increases in control error of T1 and T2 after t=11 s are caused by modeling errors from the joint friction forces and the contact dynamics error. Therefore, this verifies that the proposed method can guarantee the strict priority when controlling the optimised multiple tasks ($\ddot{\mathbf{x}}_i^{\text{opt}}$).

D. SCENARIO #4: INVESTIGATION OF COMPUTATIONAL COST

To evaluate the RT-robustness of the proposed WBC framework, several simulations are conducted by changing the number of contact planes as well as the number of prioritized tasks.

The algorithm computation time is measured in respective RT and non-RT threads, and averaged after 1000 times execution. As proposed in Section III-B2, RT thread computation time is taken for solving the QP formulation of the task-oriented optimisation and calculating kinematics parameters for $\mathbf{J}_\hbar$ and $\mathbf{J}_c$; and non-RT thread computation time is taken for calculation of the contact-constrained dynamics parameters and joint space dynamics parameters $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{g}$. One can observe in Fig. 14 that the proposed method guarantees the control frequency of 1 kHz without time violation computing the task-oriented optimisation. It is completed within 0.5 ms even if the contact DoFs that is proportion to the computational time increase to 24.

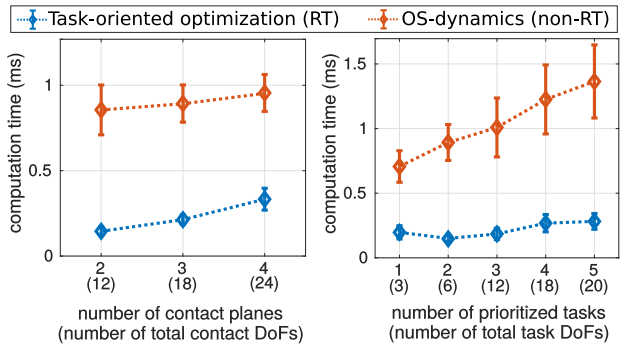


FIGURE 14. Average computation time and its standard deviation of the task-oriented optimisation in RT thread and the contact-constrained dynamics parameters calculation regarding the number of contact planes (left) and the number of prioritized tasks (right).

Besides, in the non-RT thread loop, computation of the contact-constrained dynamics parameters is completed within 2 ms, and is then written in the shared memory. While both the number of contacts and number of prioritized tasks increase its computation time, the latter affects more, because the calculation of $\widetilde{\mathbf{J}}_{\hbar}^{T}$ for each hierarchy is dominant in overall computation. Indeed, the controller in 1 kHz RT thread can update the dynamic parameters from the shared memory in less than 3 loops.

Incidentally, we also confirm that the COD-based computation method, presented in Section III-B1, provides faster calculation speed compared with the SVD-based approach. For example, in the simulation with 5 prioritized tasks and two-feet contacts, the COD-based computation is 64% more efficient as it takes 1.55 ms while the SVD-based one takes 4.27 ms. This is a reasonable result since the computational cost of the Householder QR factorization is $O(\mathfrak{m}\mathfrak{n}^2 - \mathfrak{n}^3/3)$ and the SVD is $O(\mathfrak{m}\mathfrak{n}^2 + \mathfrak{n}^3)$ for an $\mathfrak{m} \times \mathfrak{n}$ thin matrix.

V. COMPARATIVE DISCUSSION

This section provides a closer investigation of the proposed method compared with the conventional QP- and HQP-based WBC approaches. In fact, although each method has its own advantages, those for the existing optimisation-based methods have not been clearly addressed in previous literature. We, therefore, strive to draw an intuition from the pros and cons of each method for control engineers to make an appropriate choice necessitated from the application. For fair comparisons, the optimisation problem with the QP solver is set with the common conditions of the robot dynamics (1), stationary contact constraint ($\ddot{\mathbf{x}}_c = \mathbf{0}$), and Jacobian-based approach ($\ddot{\mathbf{x}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}}$).

A. COMPARISON WITH QP-BASED WBC

The proposed method can provide a strict task hierarchy when there are no task amendments as shown in Section IV-C, while the QP-based WBC always provides a soft hierarchy between the tasks.

For intuitive comparison, a typical QP-based WBC algorithm, e.g., [14], is formulated as a similar optimisation problem form to the proposed method as follows:

$$\underset{\ddot{\mathbf{q}}^{\text{opt}},\boldsymbol{\Gamma}^{\text{opt}},\mathbf{F}_c^{\text{opt}}}{\text{minimise}} \sum_{i=1}^{3} \frac{1}{2} \boldsymbol{X}_i^T \boldsymbol{\Upsilon}_i \boldsymbol{X}_i, \tag{22}$$

which is subject to (1), $\mathbf{J}_c\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c\dot{\mathbf{q}} = 0$, $\ddot{\mathbf{x}}_{\hbar}^{\text{opt}} = \mathbf{J}_{\hbar}\ddot{\mathbf{q}}^{\text{opt}} + \dot{\mathbf{J}}_{\hbar}\dot{\mathbf{q}}$, and (15), where $\ddot{\mathbf{q}}^{\text{opt}} \in \mathbb{R}^n$ is the optimised joint acceleration vector, and $\boldsymbol{\Gamma}^{\text{opt}} \in \mathbb{R}^k$ is the optimised joint torque vector.

In this formulation, the vector $\boldsymbol{X}_1$ is given as a sum of the task reference amendment, $\boldsymbol{X}_2$ is the same as (11), $\boldsymbol{X}_3 = \Gamma^{\text{opt}}$, and the weighting matrices are set as $\boldsymbol{\Upsilon}_1 \gg \boldsymbol{\Upsilon}_2 \gg \boldsymbol{\Upsilon}_3$. Here, $\boldsymbol{X}_1^T\boldsymbol{\Upsilon}_1\boldsymbol{X}_1$ is set to provide physical meaning of $\sum_{j=1}^{m} w_j(\ddot{\mathbf{x}}_j^{\text{opt}} - \ddot{\mathbf{x}}_j^*)$ for the hierarchical control, where w_j is the weight number for the j-th priority task. The hierarchy can then be achieved by assigning weights $w_{\hbar}$. This weighted sum approach creates a clear difference in a way to deal with hierarchical tasks compared to the proposed method. A weighted sum approach generates a soft hierarchy by optimising the holistic task references with $\boldsymbol{X}_1^T\boldsymbol{\Upsilon}_1\boldsymbol{X}_1$, while the proposed method generates a strict hierarchy by minimising each hierarchical task reference individually under the dynamically-consistent control framework (12), (13).

To clearly reveal the difference between the two methods regarding the hierarchical control aspect, both optimisation problems can be reformulated as the general QP form as follows:

$$\underset{\boldsymbol{\rho}}{\text{minimise}} \frac{1}{2}\boldsymbol{\rho}^T\mathbf{G}\boldsymbol{\rho} + \boldsymbol{\ell}^T\boldsymbol{\rho}, \tag{23}$$

where $\boldsymbol{\rho}$ is the decision variable vector, $\mathbf{G}$ is the given matrix for the quadratic part of the objective, and $\boldsymbol{\ell}$ is the given vector for the linear part of the objective. Here, $\mathbf{G}$ of the proposed method can be expressed as

$$\mathbf{G} = \begin{bmatrix} w_1\mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & w_2\mathbf{I} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & w_m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & w_{m+1}\mathbf{S}_f^T\mathbf{S}_f \end{bmatrix}, \tag{24}$$

while the conventional QP-based method is given by

$$\mathbf{G} = \begin{bmatrix} \sum_{j=1}^{m} w_j\mathbf{J}_j^T\mathbf{J}_j & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & w_{m+1}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & w_{m+2}\mathbf{S}_f^T\mathbf{S}_f \end{bmatrix}. \tag{25}$$

As seen above, the decision variable of the QP-based approach includes the joint acceleration vector $\ddot{\mathbf{q}}^{\text{opt}}$ which expressed by the weighted sum of the tasks—the element (1,1) in (25). Since the decision variable is composed of joint acceleration term $\ddot{\mathbf{q}}^{\text{opt}}$, it is difficult to construct a hierarchical control structure with the operational space formulation (12) and (13). On the other hand, the operational space formulations can be directly applied as equality constraints in the proposed method since all task reference vectors of the entire hierarchy $\ddot{\mathbf{x}}_{\hbar}^{\text{opt}}$ are included in the decision variable. Accordingly, the proposed method firstly minimises reference amendment of all the tasks, so the task reference

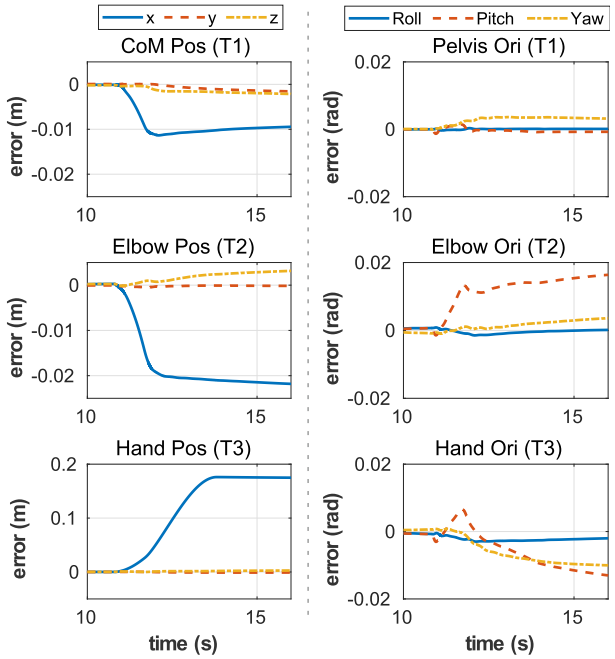


FIGURE 15. Control error when QP-based method is conducted for Scenario #3. Plots in the left column are the position control errors and plots in the right column are the orientation control errors.

amendment will be zero when there is a feasible solution. Furthermore, lower priority tasks do not affect higher priority tasks even when task references are amended according to constraints (12) and (13) with the dynamically consistent control structure. Therefore, the proposed method can effectively control the conflicting tasks with the desired priority.

On the other hand, the QP-based WBC obtains soft hierarchy through minimising the weighted sum of tasks references amendment. In this way, a solution from the QP can produce better control performance for the higher priority tasks by sacrificing control performance of lower priority tasks when there are conflicting tasks. The QP-based WBC can create a hierarchy simply, without performing a calculation of a null-space projection matrix like our method or without solving multiple QPs like the HQP-based method, but lower priority tasks can always affect higher priority tasks due to the soft hierarchy.

To show the robot behavior under the soft hierarchy of the QP-based WBC, Scenario #3 in Section IV-C is conducted with the above QP-based WBC formulation with weights $w_1 = 10^9$, $w_2 = 10^6$, $w_3 = 10^3$, $w_4 = 1$. As shown in Fig. 15, control errors of higher priority tasks are lower as intended by hierarchy. However, unlike the proposed method result shown in Fig. 13, the higher priority tasks are affected by lower priority task T3 so the control error of T1 and T2 increases proportionally to that of T3. This is in contrast to the results in Section IV-C, where the control error of the lower priority task has little effect on the higher priority tasks. This comparison result supports that our proposed method can be more effectively utilized when various tasks should be controlled under strict priority compared to the QP-based WBC approach.

Meanwhile, similar to our method, few studies are utilizing an analytically achieved null-space projection matrix for hierarchical task control. QP optimisation is incorporated with analytic null-space projection matrices in [30], however, dynamic consistency cannot be exploited due to the absence of the inertia weighting matrix in the computation of the null-space projection. Recently, the method in [30] is further extended with consideration of dynamic consistency [31], however, it is not straightforward to apply to the underactuated robot with the floating-base, e.g., humanoids. Interestingly, the authors in [32] suggest computing acceleration in the configuration space based on the desired acceleration commands instead of using operational-space dynamics calculation. While this method is computationally efficient and it can provide feasible solutions satisfying task priorities, the operational space dynamics is disregarded.

B. COMPARISON WITH HQP-BASED WBC

The HQP-based WBC offers adequate improvements to overcome afore discussed limitations of the QP-based, i.e., a generic approach producing inverse dynamics solutions satisfying strict task hierarchy and inequality constraints. It is worth to notice that both HQP-based and the proposed WBCs can guarantee dynamic consistency when a feasible solution can be obtained without amending task references, i.e., when $\ddot{\mathbf{x}}_h^{\text{opt}} = \ddot{\mathbf{x}}_h^*$. Interesting, in this case, the HQP-based WBC gives the same solution as the dynamically consistent OS-WBC [20], [33].

However, when solutions in WBCs meet inequality constraints, the task references are ought to be amended via optimisation, where differences between the HQP and the proposed methods become apparent. The proposed method finds optimal amendments for each task ($\ddot{\mathbf{x}}_h^{\text{opt}} \neq \ddot{\mathbf{x}}_h^*$), where the level of amendments is determined by the corresponding task-oriented weighting matrix, $\mathbf{H}_i$ in (9), and the relationship between the task and saturated inequality constraints. Possibly, the lower priority task saturating inequality constraints may produce amendments for the higher priority task in this optimisation process. On the other hand, the HQP-based WBC can keep the strict hierarchy even when task references are amended since optimisation is solved for each prioritized task under inequalities and QP results of higher priority tasks are set as an equality constraint in lower priority QP optimisation to prevent affecting higher priority tasks.

Therefore, unlike the HQP-based method, the proposed method softens the task hierarchy when inequality constraints have to be bounded for a feasible solution as seen in Section IV-B, which aims at searching optimal amendments for entire tasks to achieve the best goal performance. The priority in the task optimisation is indirectly managed by the weighting matrix $\mathbf{H}_i$. In other words, the task hierarchy strictness of the proposed method is determined by whether or not a feasible solution is bounded by inequality constraints. This characteristic in terms of keeping task hierarchy can be defined as 'semi-strict' hierarchy since it is different compared with QP-based WBCs, which provide soft-hierarchy,

TABLE 3. Average computation time of WBCs in [millisecond].

Test condition		HQP-based WBC	Proposed method	
N° tasks	DoFs	RT thread[1]	RT thread	non-RT thread[2]
1	3	1.37	**0.21**	(0.82)
2	6	2.53	**0.17**	(1.01)
3	12	3.84	**0.20**	(1.12)
4	18	5.18	**0.29**	(1.33)
5	20	6.34	**0.30**	(1.36)

[1] The HQP algorithm is implemented as [33]. Note that the computation time may be reduced if another HQP-based methods such as [20], [23], [34], [35] are applied.
[2] The contact-constrained operational space dynamics variables are separately calculated in the non-RT thread, which does not affect realtime robustness (See Figs. 1,2).

and HQP-based WBCs, which provide strict-hierarchy. Nevertheless, one can notice once the task reference is optimised, the hierarchy of the amended tasks is strictly satisfied as controlled via the dynamically consistent hierarchical control structure of (12). It can be considered as controlling the robot with conventional OS-WBC in [4] with instantaneously re-planned task and contact wrench references from optimisation. In the same perspective, the HQP-based WBC [20], [33] can also be considered as dynamically consistent even when they have to amend task references for a feasible solution.

In this light, the HQP-based WBC gives a clear benefit to the application demanding strict and persistent hierarchical tasks under inequality conditions. Whereas, it is also known that the HQP-based method requests relatively long computational time due to repeated QP optimisations, while the proposed method is designed to perform fast calculations exploiting a single QP optimisation.

To quantitatively investigate computational costs, comparative simulations are conducted for the HQP-based and the proposed WBC with the same simulation environment settings shown in Section IV, while tasks are assigned to maintain the initial home posture in the double-support state, i.e. $c = 12$ as seen in Fig. 3b. Five different cases are examined by increasing the number of task hierarchy from one to five tasks, where total DoFs increase from 3 to 20, accordingly. The algorithm computation time is measured and averaged after the algorithm is executed 1000 times in the control loop. For the HQP-based method, the algorithm introduced in [33] is implemented in a single RT-thread, and the simulation settings are entirely identical to those in Section IV-D. Note that the computation time is taken for formulating and solving the optimisation problems in the QP form as well as for calculation of kinematics and dynamics parameters $\mathbf{J}_{\hbar}$, $\mathbf{J}_c$, $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{g}$.

Table 3 presents the result of the average computation time for the HQP-based and the proposed method. One can observe that the proposed method takes less computation time owing to a single QP optimisation and the multi-threaded structure, thus is advantageous when the number of tasks increases. Note that the computation time of the HQP-based WBC can be reduced by utilizing a null-space projection technique proposed in [34], or motion-space decomposition methods introduced in [20], [23]. It may be further reduced if the method in [35] is applied for the inverse dynamics calculation. Nevertheless, it is still challenging to implement a large number of hierarchical tasks of a high DoFs robot into the HQP-based WBC running in a fast RT control loop, such as 1-2kHz commonly found in recent robotics literature; for instance, in [23], a 25 DoFs robot under double support state takes about 3 ms for solving five QP optimisations for two prioritized tasks and other constraints.

VI. CONCLUDING REMARK AND OPEN PROBLEM

This paper introduces an enhanced WBC framework for controlling a high number of hierarchical tasks with consideration of inequality constraints, by amalgamating the OS-WBC with the task-oriented optimisation and practical computation techniques to reduce computational cost in the RT control loop. The provided optimal solution rectifies task acceleration or force references to avoid constraint violations and possess flexibility for users to determine task-oriented amendments. Throughout diverse simulation scenarios, we validate the control performance of the proposed method regarding accuracy, dynamic consistency, task hierarchy, and computation efficiency. Based on the comparative study in the previous section, a guideline for the selection among WBCs can be summarized as follows:

- the QP-based WBC algorithm is relatively simple and easy to implement than HQP-based and proposed methods; and it is effective when the soft hierarchy is not disadvantageous in such case that multiple tasks are well-planned without any conflicts.
- the HQP-based WBC is the most generic method for any applications to manage multiple tasks obeying strict hierarchy.
  However, relatively heavy computational cost can be a burden when the number of tasks and DoFs of the robot increase; and
- the proposed method can be a good choice for those who are familiar with the classical OSF-based methods [1]–[4] and seek a way to control a high number of hierarchical tasks under inequality constraints, yet with less computing power. Whereas, programming work considering the multi-thread and shared

memory requires a relatively complex implementation process.

It is worthwhile to recall the previous discussion that the proposed WBC framework can give a more strict hierarchy compared to the QP-based WBC with soft hierarchy, but the tasks are optimally amended when inequality constraints meet during a task execution; and it is proven in Section IV-B2 that the proposed method can still bestow the priority for task amendments by users' choice of weightings. However, it is also true that the lower-priority task can affect the higher priority tasks during the optimisation process in the proposed method, thus is so-called 'semi-strict' hierarchy, while the HQP-based method keeps the very strict hierarchy in the sense of the original task priority concept that sacrifices the lower priority task to manage the inequality constraints. Therefore, there is still an open question: ''which is better when inequality condition constrains task executions: finding the solution by always sacrificing the lower priority task as HQP method does? or first amending the tasks and redistributing the prioritised tasks as the proposed method does?'' This interesting problem should be studied in future work.

Incidentally, the proposed WBC framework has been already implemented and experimented in the actual humanoid COMAN+ with RT-OS system and multi-threading, called XbotCore [28] (refer to supplementary media of this paper for details of experimental setup and results.) Another future work will aim at exploiting control schemes coping with modeling error and disturbance for robustness and control accuracy, e.g., [36], [37].

REFERENCES

[1] 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.

[2] O. Khatib, L. Sentis, J. Park, and J. Warren, ''Whole-body dynamic behavior and control of human-like robots,'' *Int. J. Humanoid Robot.*, vol. 1, no. 1, pp. 29–43, Mar. 2004.

[3] L. Sentis and O. Khatib, ''Synthesis of whole-body behaviors through hierarchical control of behavioral primitives,'' *Int. J. Humanoid Robot.*, vol. 2, no. 4, pp. 505–518, Dec. 2005.

[4] J. Park and O. Khatib, ''Contact consistent control framework for humanoid robots,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2006, pp. 1963–1969.

[5] H. Bruyninckx and O. Khatib, ''Gauss' principle and the dynamics of redundant and constrained manipulators,'' in *Proc. ICRA. Millennium Conf., IEEE Int. Conf. Robot. Automat. Symposia*, vol. 3, Apr. 2000, pp. 2563–2568.

[6] Y. Lee, S. Hwang, and J. Park, ''Balancing of humanoid robot using contact force/moment control by task-oriented whole body control framework,'' *Auto. Robots*, vol. 40, no. 3, pp. 457–472, Mar. 2016.

[7] D. Kim, Y. Zhao, G. Thomas, B. R. Fernandez, and L. Sentis, ''Stabilizing series-elastic point-foot bipeds using whole-body operational space control,'' *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1362–1379, Dec. 2016.

[8] Y. Lee, H. Lee, S. Hwang, and J. Park, ''Terrain edge detection for biped walking robots using active sensing with vCoP-position hybrid control,'' *Robot. Auto. Syst.*, vol. 96, pp. 41–57, Oct. 2017.

[9] Y. Lee and J. Park, ''Reactive bipedal walking method for torque controlled robot,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 395–402.

[10] M. Mistry, J. Buchli, and S. Schaal, ''Inverse dynamics control of floating base systems using orthogonal decomposition,'' in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 3406–3412.

[11] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, ''Optimal distribution of contact forces with inverse-dynamics control,'' *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 280–298, Mar. 2013.

[12] C. Collette, A. Micaelli, C. Andriot, and P. Lemerle, ''Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts,'' in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2008, pp. 3187–3193.

[13] F. Aghili, ''Quadratically constrained quadratic-programming based control of legged robots subject to nonlinear friction cone and switching constraints,'' *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 6, pp. 2469–2479, Dec. 2017.

[14] K. Bouyarmane and A. Kheddar, ''On weight-prioritized multitask control of humanoid robots,'' *IEEE Trans. Autom. Control*, vol. 63, no. 6, pp. 1632–1647, Jun. 2018.

[15] B. Henze, M. A. Roa, and C. Ott, ''Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios,'' *Int. J. Robot. Res.*, vol. 35, no. 12, pp. 1522–1543, Oct. 2016.

[16] S. Kuindersma, F. Permenter, and R. Tedrake, ''An efficiently solvable quadratic program for stabilizing dynamic locomotion,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Jun. 2014, pp. 2589–2594.

[17] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, ''Optimization-based full body control for the DARPA robotics challenge,'' *J. Field Robot.*, vol. 32, no. 2, pp. 293–312, Mar. 2015.

[18] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, ''Design of a momentum-based control framework and application to the humanoid robot atlas,'' *Int. J. Humanoid Robot.*, vol. 13, no. 1, Mar. 2016, Art. no. 1650007.

[19] G. Mesesan, J. Englsberger, G. Garofalo, C. Ott, and A. Albu-Schäffer, ''Dynamic walking on compliant and uneven terrain using DCM and passivity-based whole-body control,'' in *Proc. IEEE-RAS 19th Int. Conf. Humanoid Robots (Humanoids)*, Oct. 2019, pp. 25–32.

[20] N. Mansard, ''A dedicated solver for fast operational-space inverse dynamics,'' in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 4943–4949.

[21] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Souères, and J.-Y. Fourquet, ''Dynamic whole-body motion generation under rigid contacts and other unilateral constraints,'' *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, Apr. 2013.

[22] P. M. Wensing and D. E. Orin, ''Generation of dynamic humanoid behaviors through task-space control with conic optimization,'' in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 3103–3109.

[23] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti, ''Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid,'' *Auto. Robots*, vol. 40, no. 3, pp. 473–491, Mar. 2016.

[24] J. Park, ''The relationship between controlled joint torque and end-effector force in underactuated robotic systems,'' *Robotica*, vol. 29, no. 4, pp. 581–584, Jul. 2011.

[25] S. Kim, K. Jang, S. Park, Y. Lee, S. Y. Lee, and J. Park, ''Continuous task transition approach for robot controller based on hierarchical quadratic programming,'' *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1603–1610, Apr. 2019.

[26] J. W. Demmel, *Applied Numerical Linear Algebra*, vol. 56. Philadelphia, PA, USA: SIAM, 1997.

[27] O. Khatib, ''Real-time obstacle avoidance for manipulators and mobile robots,'' in *Autonomous Robot Vehicles*. New York, NY, USA: Springer, 1986, pp. 396–404.

[28] L. Muratore, A. Laurenzi, E. M. Hoffman, and N. G. Tsagarakis, ''The XBot real-time software framework for robotics: From the developer to the user perspective,'' *IEEE Robot. Autom. Mag.*, vol. 27, no. 3, pp. 133–143, Sep. 2020.

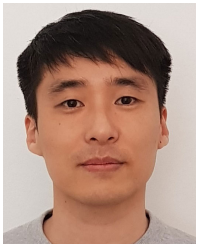
[29] *The European Project 'Cognitive Interaction in Motion—CogIMon*. Accessed: 2019. [Online]. Available: https://cogimon.eu/

[30] M. Liu, R. Lober, and V. Padois, ''Whole-body hierarchical motion and force control for humanoid robots,'' *Auto. Robots*, vol. 40, no. 3, pp. 493–504, Mar. 2016.

[31] N. Dehio and J. J. Steil, ''Dynamically-consistent generalized hierarchical control,'' in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 1141–1147.

[32] D. Kim, J. Lee, J. Ahn, O. Campbell, H. Hwang, and L. Sentis, ''Computationally-robust and efficient prioritized whole-body controller with contact constraints,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.

[33] L. Saab, N. Mansard, F. Keith, J.-Y. Fourquet, and P. Souères, "Generation of dynamic motion for anthropomorphic systems under prioritized equality and inequality constraints," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 1091–1096.
[34] M. de Lasa, I. Mordatch, and A. Hertzmann, "Feature-based locomotion controllers," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–10, Jul. 2010.
[35] 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, Jun. 2014.
[36] J. Lee, H. Dallali, M. Jin, D. G. Caldwell, and N. G. Tsagarakis, "Robust and adaptive dynamic controller for fully-actuated robots in operational space under uncertainties," *Auto. Robots*, vol. 43, no. 4, pp. 1023–1040, Apr. 2019.
[37] 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.



YISOO LEE received the B.S. and M.S. degrees in naval architecture and ocean engineering from Seoul National University, South Korea, in 2008 and 2010, respectively, and the Ph.D. degree from the Department of Intelligent Convergence Systems, Seoul National University, in 2017. From 2017 to 2018, he was a Postdoctoral Researcher with Seoul National University. From 2018 to 2020, he held a postdoctoral position at the Humanoids & Human Centered Mechatronics Laboratory, Istituto Italiano di Tecnologia, Italy. He is currently a Senior Researcher with the Korea Institute of Science and Technology (KIST). His research interests include humanoid robot control, robot locomotion, manipulation, and optimal control.

SANGHYUN KIM received the B.S. degree in mechanical engineering and the Ph.D. degree in intelligent systems from Seoul National University, South Korea, in 2012 and 2020, respectively. He was a Postdoctoral Researcher with The University of Edinburgh, U.K., in 2020. He is currently a Senior Researcher with the Korea Institute of Machinery and Materials, South Korea. His main research interest includes optimal control for mobile manipulators.

JAEHEUNG PARK (Member, IEEE) received the B.S. and M.S. degrees in aerospace engineering from Seoul National University, South Korea, in 1995 and 1999, respectively, and the Ph.D. degree in aeronautics and astronautics from Stanford University, USA, in 2006. From 2006 to 2009, he was a Postdoctoral Researcher and later a Research Associate with the Stanford Artificial Intelligence Laboratory. From 2007 to 2008, he worked part-time with Hansen Medical Inc., a medical robotics company, USA. Since 2009, he has been a Professor with the Department of Intelligent Convergence Systems, Seoul National University. His research interests include robot–environment interaction, contact force control, robust haptic teleoperation, multicontact control, whole-body dynamic control, biomechanics, and medical robotics.

NIKOS TSAGARAKIS (Member, IEEE) received the D.Eng. degree in electrical and computer science engineering from the Polytechnic School, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1995, and the M.Sc. degree in control engineering and the Ph.D. degree in robotics from the University of Salford, Salford, U.K., in 1997 and 2000, respectively. Since 2013, he has also been working as a Visiting Professor with the Center for Robotics Research, Department of Informatics, King's College University, London, U.K. He is currently a Tenured Senior Scientist with the Istituto Italiano di Tecnologia, Genoa, Italy, with overall responsibility for humanoid design and human centered mechatronics development. He was a Technical Editor of the IEEE/ASME TRANSACTIONS ON MECHATRONICS. He is currently on the Editorial Board of the IEEE ROBOTICS AND AUTOMATION LETTERS.

JINOH LEE (Senior Member, IEEE) received the B.S. degree *(summa cum laude)* in mechanical engineering from Hanyang University, Seoul, South Korea, in 2003, and the M.Sc. and Ph.D. degrees in mechanical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012. From 2012 to 2017, he held a postdoctoral position at the Department of Advanced Robotics, Istituto Italiano di Tecnologia (IIT), Genoa, Italy, where he also held a research scientist position, from 2017 to 2020. He is currently a Research Scientist with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Weßling, Germany. 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. He is a member of the IEEE Robotics and Automation, the IEEE Control Systems, and the IEEE Industrial Electronics Societies. He is a Senior Editor of the IROS Conference Paper Review Board (CPRB).