Explicit safety and compliance on torque controlled robots for physical interaction

Mathieu Celerier, Bastien Muraccioli, Mehdi Benallegue, Yue Hu, Rafael Cisneros Limón, Hiroshi Kaminaga, Gentiane Venture

Explicit safety and compliance on torque controlled robots for physical interaction*

Mathieu Celerier[1], Bastien Muraccioli[1], Mehdi Benallegue[1], Yue Hu[3], Rafael Cisneros Limón[1], Hiroshi Kaminaga[1], Gentiane Venture[1,2]

***Abstract*— This paper presents a framework aimed at improving safety and compliance in dynamic physical interactions for precise torque-controlled rigid-body robots. The framework uses a quadratic program (QP) in the motion generation formulation that explicitly accounts for external forces. Our solution ensures that adherence to feasibility and safety constraints is robust to disturbances, while preserving task compliance. Our tests on a torque-controlled Kinova Gen3 manipulator arm where we simulate external forces by attaching an uncompensated weight (1.25kg) to it's end-effector demonstrated a reduction of 100% in the violation of velocity limit. We also show that it's straightforward to choose between stiffness and compliance for each of the concurrent tasks. Furthermore, by integrating force/torque sensor measurements with a residual-based estimation, we enhanced the accuracy of interaction force estimation on average from 2.0N RMS error using only the residual estimation to 0.7N RMS error with our method. These results highlight the effectiveness of our approach in maintaining reactive safety and compliance in the presence of external forces.**

I. Problem statement

A. About safety and compliance

Robots, commonly used in industrial settings like supply chains, are typically programmed to avoid unintended collisions or interactions with their environment and each other. This is achieved using high-gain position or velocity control to ensure precision and stiffness. These controllers would produce high torque to compensate any deviation from the reference joint trajectory. However, in scenarios where close contact with a dynamic, unpredictable environment is inevitable, traditional position and velocity controls may become too fast and powerful, risking damage to the robot or its surroundings. This is particularly critical in human-robot interaction scenarios, where even small errors can lead to severe or fatal accidents, often in contexts where such power is unnecessary. Therefore, it is necessary to limit the robots' ability to cause damage [1].

Robot control architecture is commonly organized into a layered scheme where each layer depends on the performance and guarantees provided by the lower layers [2]. Safety requirements span these layers, encompassing high-level planning constraints down to mid-level and low-level control mechanisms [3].

At the highest layer, encompassing semantics, planning, and behavior generation, safety mechanisms are responsible for ensuring that task planning and action execution adhere to operational rules and avoid high-level collisions [**?**], [**?**]. Although these mechanisms are critical for overall system safety, they lie outside the focus of this study.

At the mid-level layer we consider task coordination, trajectory generation, and kinematic computations. Here, safety is enforced through strict kinematic constraints, which include monitoring joint positions to prevent exceeding mechanical boundaries, controlling velocities and accelerations to maintain system stability, and ensuring collision avoidance with the robot itself and through restricted zones [**?**], [**?**].

At the lowest layer, which deals with joint-level position control, torque regulation, and current control, safety requirements become more nuanced. Two primary safety guarantees must be maintained.

1) Respect the kinematic constraints imposed by the mid-level layer, both in joint and Cartesian spaces. Achieving this requires precise kinematic control, often implemented using high-gain position control to ensure stiff actuation.
2) Limit the external and internal forces and torques to protect both the robot and its environment [4]. This requires the robot to have some compliance, enabling it to move or deform to adapt to external forces.

B. Issues and challenges

To summarize, the low-level control must achieve precise motor control while also ensuring compliance by limiting interaction forces. Balancing these requirements is challenging because compliance causes the robot to deviate from its planned trajectory when responding to external forces, potentially leading to task-tracking errors or even breaches of safety constraints, such as exceeding joint limits or causing collisions. This inherent conflict between maintaining precise kinematic control and ensuring force/torque compliance poses the main problem in robotic safety [5] that we address in this paper.

[1]M. Celerier, B. Muraccioli, M. Benallegue, R. Cisneros L., H. Kaminaga and G. Venture are with CNRS-AIST JRL (Joint Robotics Laboratory), IRL3218, Department of Information Technology and Human factors, National Institute of Advanced Industrial Science and Technology (AIST), AIST Tsukuba Headquarters and Information Technology Collaborative Research Center, 1-1-1 Umezono, Tsukuba, Ibaraki, 305-8560, Japan, {mathieu.celerier,mehdi.benallegue, rafael.cisneros,hiroshi.kaminaga}@aist.go.jp

[2]G. Venture is also with the Department of Mechanical Engineering, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan, venture@g.ecc.u-tokyo.ac.jp

[3]Y. Hu is with the Department of Mechanical & Mechatronics Engineering, University of Waterloo, 200 University Ave W, Waterloo, ON N2L 3G1, Canada, yue.hu@uwaterloo.ca

*This research was partially funded by the Japan Science and Technology Agency (JST) with JST-Mirai Program, grant number JPMJMI21H4

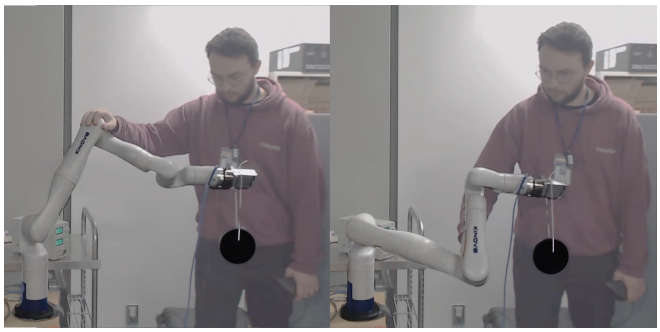

Fig. 1. A demonstration of null-space compliance. The robot remains compliant in the null space of the pose task of the end-effector, its posture can be modified without altering the position or orientation of the end-effector, despite the weight attached at its tip.

To handle this conflict, an ideal framework should enable the robot to increase stiffness only when kinematic safety constraints are at stake (e.g., when approaching joint or velocity limits) and remain compliant otherwise. Such a selective-stiffness approach maximizes safety by limiting forces and torques until a constraint is in danger of being violated, at which point the robot becomes sufficiently rigid to prevent a breach. However, activating and deactivating these constraints can introduce nonsmooth dynamics, and sensor noises can produce choppy motions in the vicinity of the constraints [6]. To alleviate this we require robust control solutions capable of handling abrupt changes in stiffness levels.

To increase stiffness, traditional methods often raise the control gains in the kinematic feedback loop. While this can improve the accuracy of planned trajectories, it depends on kinematic tracking errors to generate compensating forces, which leads to greater deviations from the desired references. A more physically grounded and responsive strategy is to estimate and counteract external forces in real time.

External forces can be estimated using two primary methods. The first involves direct force/torque sensors or skin-like sensing systems [7], [8], which provide real-time measurements of external forces but typically cannot cover the entire robot structure. The second one is the residual-based estimation, which derives external forces by subtracting the predicted generalized momentum from measured joint torques [9]. However, residual-based estimation is only reliable at low frequencies because it effectively provides a first-order filtered approximation of the external joint torques. This indicates that for time-varying forces, there is an inevitable delay, leading to reduced accuracy as the frequencies of external forces increase, such as in cases of impact. In this paper, we propose merging both sources of force estimation when available and leveraging each method's strengths without compromising accuracy. This combination enables to get the most precise force estimation regardless of where external forces are applied, even under multiple simultaneous disturbances. It is also a key part of our holistic framework, as it supports significantly higher stiffness when the disturbance is measured by the force sensor.

Finally, such a framework should remain flexible in defining which tasks can be executed with compliance and which require higher stiffness. In many practical scenarios, the robot's primary task does not occupy all of its degrees of freedom, leaving additional (null-space) dimensions that can remain compliant to external forces. This approach allows the robot to achieve high stiffness on critical tasks (e.g., precise positioning or force-sensitive operations), while still granting compliance where it does not compromise task performance or safety. This behavior is referred to as null-space compliance [10]. Some methods have implemented this using passive compliance without external force estimation [11]. However, these approaches lack a framework to adapt selective stiffness dynamically to guarantee safety constraints.

C. Context

Compliance in robotics can be passive, active [12], or a combination of both [13], [14].

- **Passive compliance** uses backdrivable actuators, springs, or compliant materials to naturally absorb external forces, helping the robot adapt to its environment [15].
- **Active compliance** employs real-time control methods, such as torque or impedance control, to modulate the robot's response based on sensed external forces [16].

Our work focuses on collaborative robots (e.g., *Franka Emika Panda*, *Kinova Gen3*, *KUKA LBR iiwa*) that exhibit minimal passive compliance and thus require active compliance control [17], [18].

Many existing safety strategies detect collisions during position control and then switch to compliance control to mitigate impact forces [19]. While these reactive policies are valuable, they do not address all situations requiring high safety levels, especially during sustained physical human-robot interaction. Some studies limit the contact force between the robot and the environment to a safe threshold [20], which requires direct joint-torque control. Such torque-level control allows the robot to tracle tasks with low-stiffness gains to maintain passive compliance and to producing specific forces at selected contact points.

Torque and force safety constraints can also be enforced at the mid-level. For instance, torque-based motion generation can be framed as a Quadratic Problem (QP) that includes equality and inequality constraints for feasibility and safety, including torque and force limits [21]. However, these QP-based approaches typically do not feed external force information back into the mid-level in real-time. As a result, any compliance occurs only as an unintended deviation from the QP's references, meaning that the mid-level constraint guarantees, especially kinematic constraints, may not be fully respected under external disturbances.

The literature shows that torque-controlled robots can indeed become safer and more compliant through various control and sensing strategies. Yet these approaches still don't offer a response to the stiffness versus compliance requirements posed by kinematic and force constraints re-

spectively. In this paper we propose a complete framework that addresses several aspects of this issue.

D. Contributions and Overview.

In summary this paper, we propose a control framework that:

- Facilitates flexible assignment of stiff or compliant behavior to different tasks, leveraging redundancy where available to produce null-space compliance or any other combination of stiffness and complicance. We perform this by making the task-space control stiff by default and expressing compliance as an explicit task objective. Consequently, being compliant no longer means deviating from the mid-level reference trajectory but rather actively tracking a reference that directs the robot to follow the external force's direction.
- Dynamically adapts the robot's stiffness based on current active safety restrictions by handling all safety and feasibility conditions as stiff task-level constraints, while remaining robust against sensor noise and the nonsmooth behavior induced by stiffness transitions.
- Integrates high-bandwidth force sensors with residual-based external force estimation, using spectral fusion. This enhances task stiffness and improves tracking accuracy under external disturbances, wherever they are applied.

THISIS WRONGThe paper is organized as follows: We first present our framework, starting with the improved estimation of the external torques and the QP framework. Then we present the experimental setup and the results, followed by the conclusion.

II. Proposed framework

A. Robot dynamics

The scope of this paper targets serial manipulator robot arms but extensions can be made to other kinds of robots. Let us define an n degrees of freedom (DoF) robot with joint configuration space $\boldsymbol{\mathcal{Q}}$, usually represented by elements of $\mathbb{R}^n$. We call $\mathbf{q} \in \boldsymbol{\mathcal{Q}}$ the joint configuration of the robot, $\dot{\mathbf{q}} \in \mathbb{R}^n$ the joint velocity, $\ddot{\mathbf{q}} \in \mathbb{R}^n$ the joint acceleration, and $\boldsymbol{\tau} \in \mathbb{R}^n$ the commanded torque in joint space.

The dynamic model of a manipulator robot arm under external forces can be written as follows [22]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}^e \tag{1}$$

where $\mathbf{M}(\cdot) \in \mathbb{R}^{n\times n}$ and $\mathbf{C}(\cdot) \in \mathbb{R}^{n\times n}$ are the inertia and Coriolis matrices respectively such that $\dot{\mathbf{M}}(\cdot) = \mathbf{C}(\cdot)+\mathbf{C}(\cdot)^T$ and $\mathbf{g}(\cdot) \in \mathbb{R}^n$ is the gravity vector. The external torque[1] $\boldsymbol{\tau}^e \in \mathbb{R}^n$ is the sum of joint torques induced by applied external forces and moments $\mathbf{F}_i \in \mathbb{R}^6$ and defined as:

$$\boldsymbol{\tau}^e = \sum \mathbf{J}_i(\mathbf{q})^T \mathbf{F}_i, \tag{2}$$

[1]In this paper we assume exteral forces and moments to be applied by the environment on the robot, but they can be seen as forces applied by the robot on the environment, which lead to slight changes in (1).

where $\mathbf{J}_i(\mathbf{q}) \in \mathbb{R}^{6\times n}$ is the Jacobian of the contact point [10]. To simplify the reading of the following parts of this paper we define $\ddot{\mathbf{q}}^e$ as the joint acceleration due to applied external forces as $\ddot{\mathbf{q}}^e = \mathbf{M}^{-1}\boldsymbol{\tau}^e$.

B. External torque estimation

To be able to include in our model the torques due to external forces first we need to measure/estimate them. Two methods can be used to retrieve external torques. First, we can place one or more F/T sensors on each point where physical contact with the robot is desired (usually the end-effector in the case of a manipulator arm robot). In other words, they measure directly $\mathbf{F}_i$ in (2).

The other method is to use an observer such as the residual-based method [9], which can work regardless of the contact location. Concretely, the residual is the estimation of the external joint torque and is defined with the following differential equation:

$$\mathbf{r} = K_I \left[\mathbf{p}(t) - \int_0^t \left(\boldsymbol{\tau} + \mathbf{C}^T\dot{\mathbf{q}} - \mathbf{g} + \mathbf{r} \right) ds - \mathbf{p}(0) \right], \tag{3}$$

with $\mathbf{r}$ the residual, K_I a positive gain which defines the estimation dynamics, and $\mathbf{p}$ the generalized momentum of the robot defined by:

$$\mathbf{p}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}. \tag{4}$$

The residual-based method ensures the following first-order response between the actual value of external torques and their estimation:

$$\frac{r_i(s)}{\tau_i^e(s)} = \frac{K_I}{K_I + s} \tag{5}$$

with $i \in 1, .., N-1$ the joint index. Due to this dynamic acting as a low-pass filter with a cut-off frequency of $\frac{K_i}{2\pi}$, the higher frequencies of external forces are filtered out.

We introduce a third method, hereinafter referred to as "our method". This approach merges the two previously presented methods, combining residual-based estimation with an F/T sensor to compensate for the loss of higher frequencies. These frequencies contain crucial information for dynamic environmental interaction, such as during impacts.

To do so, we first filter our F/T sensor measurements such that they are only used for the frequencies missing in the residual by applying the following filter described in the Laplace domain:

$$\begin{aligned} \boldsymbol{\tau}_{filtered}(s) &= \left(1 - \frac{K_I}{K_I + s}\right) \boldsymbol{\tau}_{sensor}(s) \\ &= \frac{s}{K_I + s} \boldsymbol{\tau}_{sensor}(s), \end{aligned} \tag{6}$$

where $\boldsymbol{\tau}_{sensor} = \sum \mathbf{J}_l(\mathbf{q})^T \mathbf{F}_l$ with $\mathbf{J}_l(\mathbf{q})^T$ the Jacobian matrix associated with the link l which denotes the link equipped with F/T sensor providing the measurement $\mathbf{F}_l$. From this signal, we can define the estimation of external forces combining both (5) and (6) from respectively the residual and the F/T sensor as:

$$\hat{\boldsymbol{\tau}}^e = \mathbf{r} + \boldsymbol{\tau}_{filtered}. \tag{7}$$

The consequence of this formulation is that if all the external forces are measured by the F/T sensors ($\boldsymbol{\tau}_{sensor} = \boldsymbol{\tau}^e$), the estimation retrieves the value of the external force in the complete bandwidth of the F/T sensor:

$$\begin{aligned}\hat{\boldsymbol{\tau}}^e &= \frac{K_I}{K_I + s}\boldsymbol{\tau}^e + (1 - \frac{K_I}{K_I + s})\boldsymbol{\tau}^e, \\ &= \boldsymbol{\tau}^e.\end{aligned} \tag{8}$$

and in the case that some external forces are missing from the F/T sensor, their sum will be estimated through the residual method.

In other words, with our proposed method we obtain perfect estimations of the external torques applied through the F/T sensors without any reduction in the quality of the estimation of the residual method for other external forces, thus gaining the best of each method.

C. QP-based motion generation

The motion generation has to produce reference trajectories to be tracked by the robot. In the proposed framework we need to generate a reference joint acceleration $\ddot{\mathbf{q}}^r$. This reference needs to account for hierarchically sorted goals. Here, they are described in decreasing hierarchical order.

1) Safety and feasibility constraints: The primary objective is to adhere to feasibility and safety constraints. Feasibility constraints, including collision avoidance, joint torque, and position limits, ensure trajectory generation capability. Violations can lead to potentially hazardous behaviors. Safety constraints aim to mitigate risks by limiting robot capabilities, such as joint velocity and interaction force limits.

We give here the expressions of the most common already existing constraints:

a) Torque limits: This constraint is simply written as:

$$\boldsymbol{\tau}^- \leq \boldsymbol{\tau}^r \leq \boldsymbol{\tau}^+, \tag{9}$$

with $\boldsymbol{\tau}_j^-$ and $\boldsymbol{\tau}_j^+$ respectively the lower and upper torque limit, and $\boldsymbol{\tau}^r$ the torque to be sent to the robot's internal controller, which is linearly dependent on the decision variable $\ddot{\mathbf{q}}^r$:

$$\boldsymbol{\tau}^r = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}^r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) - \hat{\boldsymbol{\tau}}^e, \tag{10}$$

which when combined with (1) under the assumption that $\hat{\boldsymbol{\tau}}^e = \boldsymbol{\tau}^e$, leads to the ideal joint acceleration tracking $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}^r$.

b) Velocity limits: We need to enforce the following velocity limit:

$$\dot{\mathbf{q}}^- \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}^+, \tag{11}$$

where $\dot{\mathbf{q}}^-$ and $\dot{\mathbf{q}}^+$ are the lower and upper velocity bounds. We write this constraint in terms of $\ddot{\mathbf{q}}^r$, which becomes:

$$\lambda_v \left(\dot{\mathbf{q}}^- - \dot{\mathbf{q}}\right) \leq \ddot{\mathbf{q}}^r \leq \lambda_v \left(\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}\right), \tag{12}$$

with λ_v a positive gain. We can prove that if this constraint is constantly respected, the velocity constraint will be respected too [23].

c) Joint position limits: This refers to the joint limit constraint

$$\mathbf{q}^- \leq \mathbf{q} \leq \mathbf{q}^+ \tag{13}$$

where $\mathbf{q}^-$ and $\mathbf{q}^+$ are the lower and upper bound for joint positions. Using a Velocity-Damper [23] we can write the following constraints:

$$\begin{cases} \dot{q}_j \leq \xi \frac{d_j - d_s}{d_i - d_s} & \text{if } \left(q_j^+ - q_j\right) < d_i \\ -\xi \frac{d_j - d_s}{d_i - d_s} \leq \dot{q}_j & \text{if } \left(q_j - q_j^-\right) < d_i \end{cases} \tag{14}$$

where the subscript j denotes the joint number and d_j is the distance to the joint limit defined as

$$\begin{cases} d_j = \left(q_j^+ - q_j\right) & \text{if } \left(q_j^+ - q_j\right) < d_i \\ d_j = \left(q_j - q_j^-\right) & \text{if } \left(q_j - q_j^-\right) < d_i \end{cases} \tag{15}$$

and where d_i and d_s are distances to the joint limit. d_i represents the distance under which the constraint will have an effect. d_s is the minimum acceptable distance to the joint limit. The control should prevent the distance to the joint limit from being smaller than this value. And ξ a fixed positive gain that can be automatically recomputed [24] each time d_j cross under d_i.

$$\xi = -\frac{d_i - d_s}{d_j - d_s}\dot{d}_j + \xi_{off} \tag{16}$$

where ξ_{off} is a constant given by the user allowing the velocity to increase a bit or stay constant before diminishing. We then write this constraint in terms of $\ddot{q}^r$ from the reference velocity of the first-order Velocity-Damper:

$$\begin{cases} \ddot{q}_j^r \leq -\lambda_\theta \left(\overline{\dot{q}_j^d} - \dot{q}_j\right) & \text{if } \left(q_j^+ - q_j\right) < d_i \\ -\lambda_\theta \left(\underline{\dot{q}_j^d} - \dot{q}_j\right) \leq \ddot{q}_j^r & \text{if } \left(q_j - q_j^-\right) < d_i \end{cases} \tag{17}$$

where λ_θ is a positive gain and $\overline{\dot{q}_j^d}$ and $\underline{\dot{q}_j^d}$ are the reference velocities from the Velocity-Damper for the upper and lower joint limits, respectively:

$$\begin{cases} \overline{\dot{q}_j^d} = \xi \frac{d_j - d_s}{d_i - d_s} & \text{if } \left(q_j^+ - q_j\right) < d_i \\ \underline{\dot{q}_j^d} = -\xi \frac{d_j - d_s}{d_i - d_s} & \text{if } \left(q_j - q_j^-\right) < d_i \end{cases} \tag{18}$$

Similarly, it can be shown that respecting this constraint at each iteration guarantees that the joint limit constraint can be respected. The gains λ_v and λ_θ are parameters commonly set to $\frac{1}{T_c}$ with T_c the control timestep duration. Although the method for choosing these gains are out of the scope of this study, these has been defined as parameters to allow greater flexibility in selecting the gain and thus the dynamic of the constraint.

Other feasibility and safety constraints can be expressed in more or less similar ways such as collision avoidance constraints, interaction forces constraints, etc.

2) Operational space tasks: The second goal is to realize the desired tasks in the operational space. These are the actual tasks that the robot needs to achieve, commonly the pose of an end-effector in the Cartesian space.

A task is commonly defined by an error function $\mathbf{e}_k(\mathbf{q}, t) = \mathbf{x}(\mathbf{q}) - \mathbf{x}^o(t)$ that we want to minimize, where

the subscript k denotes the number of the task, and $\mathbf{x}^o(t)$ is the feedforward objective task position. This is classically done by targeting converging second-order dynamics:

$$\ddot{\mathbf{e}}_k^* = -\lambda_{k,1}\dot{\mathbf{e}}_k(\mathbf{q},\dot{\mathbf{q}},t) - \lambda_{k,2}\mathbf{e}_k(\mathbf{q},t) \tag{19}$$

where $\lambda_{k,1}$ and $\lambda_{k,2}$ are positive gains and $\ddot{\mathbf{e}}_k^*$ is the ideal error acceleration. We need then to find feasible values of the task acceleration $\ddot{\mathbf{e}}_k^d(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}^d,t)$ that tracks those dynamics.

The following expression shows the affine mapping between $\ddot{\mathbf{e}}_k^d$ and $\ddot{\mathbf{q}}^d$:

$$\ddot{\mathbf{e}}_k^d(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}^d,t) = \mathbf{J}_k\ddot{\mathbf{q}}^d + \dot{\mathbf{J}}_k\dot{\mathbf{q}} - \ddot{\mathbf{x}}^o, \tag{20}$$

where $\ddot{\mathbf{x}}^o$ is the feedforward objective acceleration of the task, and $\mathbf{J}_k$ is the k-th task Jacobian matrix. However, the second-order decay of (19) is not always feasible, because of the higher priority constraints or just because the task is out of the reachable space of the robot. A classic solution is then to minimize the second-order tracking error. In other words, we want the acceleration reference to be the solution to this optimization

$$\ddot{\mathbf{q}}^d = \arg\min_{\ddot{\mathbf{q}}^r}(\|\mathbf{J}_k\ddot{\mathbf{q}}^r + \dot{\mathbf{J}}_k\dot{\mathbf{q}} - \ddot{\mathbf{x}}^o - \ddot{\mathbf{e}}_k^*\|^2), \tag{21}$$

subject to feasibility and safety constraints. We track multiple tasks using weighted optimization:

$$\ddot{\mathbf{q}}^d = \arg\min_{\ddot{\mathbf{q}}^r}(\sum_k w_k\|\ddot{\mathbf{e}}_k^r(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}^r,t) - \ddot{\mathbf{e}}_k^*\|^2), \tag{22}$$

where $\ddot{\mathbf{e}}_k^r = \mathbf{J}_k\ddot{\mathbf{q}}^r + \dot{\mathbf{J}}_k\dot{\mathbf{q}} - \ddot{\mathbf{x}}^o$ and w_k is the weight of the k-th task.

3) Null-space task: In the case of redundant robots, there is variability in the way a task can be achieved. This is usually an advantage because it provides better performances despite the activation of constraints. However, this variability may create discontinuities in the decision variable over time. Therefore, it is recommended to give as a third goal a reference to the acceleration in null space. The simplest way to do it is by giving a reference value $\ddot{\mathbf{q}}^*$ to the acceleration itself.

This reference value can be for example set to give a reference posture (posture task) or to act just like a damping. We will show later an alternative solution to make it produce null space compliance. This reference is treated as a task, numbered 0, with the Jacobian being the identity and w_0 being the weight.

Note that the hierarchy between the operational space tasks and this null-space task is set through weighting rather than strict constraints which may slightly decrease the quality of the operational task tracking. Nevertheless, it allows the prevention of problems related to the divergence of $\ddot{\mathbf{q}}^*$ in case of task singularity while being generally a good approximation of hierarchy[25].

D. The Quadratic Program

1) Stiff QP: If we follow strictly the definitions presented above, the resulting structure of the QP is the following:

$$\begin{aligned}\ddot{\mathbf{q}}^d = \arg\min_{\ddot{\mathbf{q}}^r}(&\sum_k w_k\|\ddot{\mathbf{e}}_k^r - \ddot{\mathbf{e}}_k^*\|^2 + w_0\|\ddot{\mathbf{q}}^r - \ddot{\mathbf{q}}^*\|^2);\\ &\text{such that}\\ &\mathbf{A}_c\ddot{\mathbf{q}}^r \le \mathbf{b}_c,\end{aligned} \tag{23}$$

where $\mathbf{A}_c$ and $\mathbf{b}_c$ gather all the feasibility and safety constraints.

Additional details on the implementation of this QP motion solver with torque-control are available in [26], where it is also explained how to minimize the velocity error when the torque tracking quality is bad.

The main difference between this QP and a regular one is the presence of the estimated external torques $\hat{\boldsymbol{\tau}}^e$ in the definition of the torque $\boldsymbol{\tau}^r$ that is sent to the robot. This means that the torque compensates for the external torques. Therefore, since we consider that we have a good estimation of the external torque, this should make the robot perfectly stiff.

While it is desirable for the constraints to be stiff and reactive to external torques, we need to produce a compliant behavior whenever stiffness is not necessary, namely in the null space and sometimes in the operational-space task itself.

2) QP with compliance: Since the QP is compensating the external torques by default, we need to write explicitly the compliance as a target for our tasks. Making tasks explicitly compliant amounts to taking into account the effect of external forces while still aiming to achieve their goal. This is done by replacing the QP of (23) with this new optimization problem

$$\begin{aligned}\min_{\ddot{\mathbf{q}}^r}(&\textstyle\sum_k w_k\|\ddot{\mathbf{e}}_k^r - \ddot{\mathbf{e}}_k^* - \gamma_k\mathbf{J}\hat{\ddot{\mathbf{q}}}^e\|^2 + w_0\|\ddot{\mathbf{q}}^r - \ddot{\mathbf{q}}^* - \gamma_0\hat{\ddot{\mathbf{q}}}^e\|^2);\\ &\text{such that}\\ &\mathbf{A}_c\ddot{\mathbf{q}}^r \le \mathbf{b}_c,\end{aligned} \tag{24}$$

where $\ddot{\mathbf{q}}^*$ is the ideal joint acceleration $\hat{\ddot{q}}^e = \mathbf{M}^{-1}\hat{\boldsymbol{\tau}}^e$ is the estimated effect of external forces on the accelerations, and $\gamma_k \in [0;1]$ is a variable defining the compliance of the k-th task. The idea is to explicitly require the task to deviate when influenced by an external force. This approach adds a layer of complexity to the numerical implementation of control tasks. However, it provides the flexibility to decide whether a control task should be compliant or not in various scenarios, and to what degree.

It's important to note that not all tasks necessitate compliance. In fact, for some tasks, stiffness might be a requirement. For example, if we set $\gamma_k = 0$ and $\gamma_0 = 1$, only the null-space task will be compliant, leaving the operational tasks stiff. This is known as null-space compliance.

Our approach, while similar to Sadeghian et al.'s [10], differs notably. They used a strict hierarchy, in contrast to our weighted optimization. Their method accounts for inertial effects, which we tested but found no improvement in compliance. Additionally, their approach has weaker safety guarantees. We believe our method is simpler, more flexible,

and less complex mathematically. It allows for easy changes to the control mode during operation.

If we set $\gamma_k = 1$, the operational space task becomes compliant, typically requiring its transformation into an impedance task. However, any combination of compliance and stiffness is possible. Section V-C experimentally explores the effect of the compliance parameter γ on different compliance setups.

When all tasks are set to be compliant, the robot would behave much like any torque-controlled robot, without considering external forces. But as soon as a safety constraint is activated, our method ensures that external forces are taken into account while managing this constraint. This highlights the flexibility and adaptability of our approach.

III. General experimental setup

A. Hardware presentation

Experiments were conducted on a Kinova Gen 3 robot, a 7 DoF manipulator with 4 limitless joints. The robot was interfaced with our open-source control framework[2] based on *mc_rtc*[3] using Kinova's *Kortex* API and controlled at 1 KHz. An ATI Mini58 F/T sensor was added to the end-effector and connected via *EtherCAT* to an Intel NUC. Wrenches were published through ROS to the control PC and retrieved by our framework within 1 ms. External torques were estimated using the residual method and merged with force sensor measurements using the filter presented in this paper, with gains described in the results section.

B. QP control implementation

We implemented our control in the framework *mc_rtc* which allows us to define our tasks and a set of constraints to ensure the safety and feasibility of the control. These are converted by the framework into a set of objectives and constraints for a QP. Once we build this QP, it will solve for $\ddot{\mathbf{q}}^{\mathbf{r}}$, which is then used to compute the desired torque $\boldsymbol{\tau}^r$ using (10). From this, our *mc_rtc interface*, which allows Kinova's API and mc_rtc to communicate, retrieves the computed reference torques and sends them to the low-level control of the robot.

IV. Results

A. Force sensing fusion

An important aspect of our method is to have a good estimation of external forces.

1) Experimental Setup: The robot was placed in a configuration where one joint was primarily influenced by the force applied to the end-effector, where the F/T sensor was placed. Position control was used for accurate repeatability of initial conditions. A weight (1.25 kg) was attached to the end-effector, and the system switched to torque control after reaching the initial configuration. The task then switch to compliant mode, allowing the robot to fall quickly without compensating for the weight's force. This sequence was repeated 74 times, retrieving signals from the F/T sensor, residual-based estimation, and our estimation method. RMSE was computed for each trial with respect to the F/T sensor, the ground truth in our case, and statistics were obtained from Table. I. To demonstrate our method's utility in sudden force scenarios, an experiment was conducted where a suspended weight was dropped onto the F/T sensor. This was repeated with two different residual gains, $K_I = 10$ and $K_I = 0.5$, resulting in different estimation dynamics.

TABLE I
Evaluation of RMS error over 74 trials

	Min.	Max.	Avg.	Std.
Our method [N]	0.5	0.8	0.7	< 0.1
Residual estimation [N]	1.7	2.6	2.0	0.2

2) Results: We examined the root-mean-squared error distribution for both methods per trial, as shown in Fig. 3. A Kruskal-Wallis ANOVA on this data yielded $H = 111.75$ and $p = 4 \cdot 10^{-26}$. This indicates our method significantly reduces the F/T sensor error compared to the residual at high gains ($K_I = 10$). The gain chosen for the residual as high gain is the highest gain we can use in our experimental setup, such that dynamical modeling errors such as structural vibration do not interfere with the estimation. Increasing the gain would cause these modeling errors to appear in the estimation as high frequency noise and would be amplified by the reactive feedback loop.

Fig. 2 presents results for two different gains during sudden impacts. We visualize the residual, as it is the main parameter of these two experiments for which we can clearly see the difference in dynamics. The filtered F/T sensor signals compensate for the dynamics of the residual, dropping when the weight is removed at low gain and staying low at high residual gain due to better external force tracking.

The combination of the filtered torque and the residual (Eq. 7) shows our method closely follows the raw F/T sensor measurements for forces with both high and low residual dynamics. The visible static error in both residual gain values is due to poorly modeled static friction and torque sensor bias. Despite this, our method can estimate external forces by combining residual-base estimation with F/T sensor measurements, even when the dynamics of the residual are slow compared to the force's dynamics.

B. Safety

1) Experimental Setup: Here we describe how we tested safety under external forces. We selected one constraint to test the cases where the effect of external forces is included explicitly in our QP and the cases where they are not. The constraint that we decided to test is the velocity limit. To test this, we used the same setup as the one presented for force estimation with a repeated drop of a weight attached to the F/T sensor. We repeated this test both with and without external forces being included in our QP, each 24 times. When the external force is included, we expect the QP to produce the required torque to compensate for these forces

[2]https://github.com/mathieu-celerier/mc-kinova-sim-superbuild

[3]https://jrl-umi3218.github.io/mc_rtc/

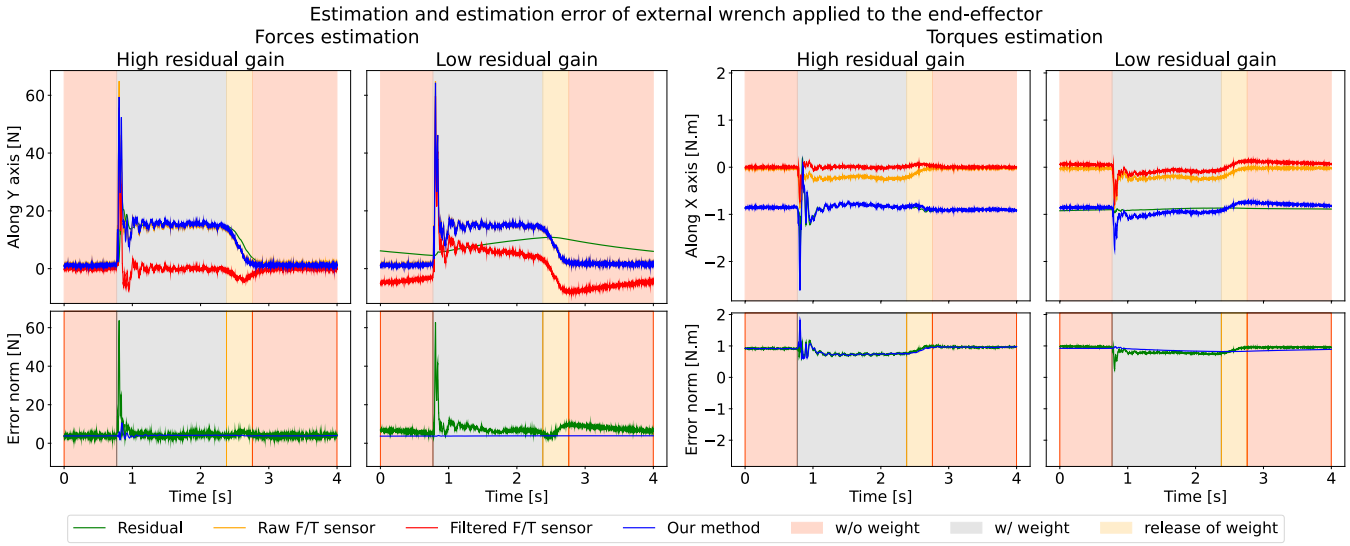

Fig. 2. Estimation of joint torques due to external forces expressed as a wrench at the end-effector. Two different values of the residual gain were used. On the left, with a low residual gain of $K_I = 0.5$, and on the right, a high residual gain of $K_I = 10$. In *yellow*, our ground truth coming from the F/T sensor measurement. In *green*, the estimation of external forces with residual-based estimation. In *red*, the F/T sensor measurement filtered such that it can be merged with the residual. Finally in *blue*, our method combining residual estimation and F/T sensor measurement.

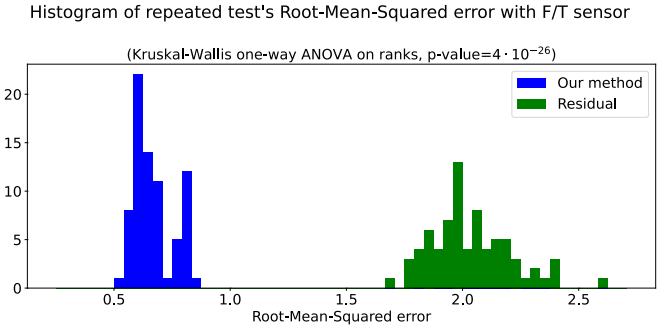

Fig. 3. Histogram of all RMS errors acquired for each method over all trials. In *blue* results on the RMSE between our method and F/T sensor's ground truth. In *green* results on the RMSE between the residual estimation and the F/T sensor's ground truth

TABLE II

OVERRUN OVER EACH TRIAL WITH OR WITHOUT EXTERNAL FORCES

External forces	Avg. overrun	Max. overrun
with	-7.3 %	-4.1 %
without	46.2 %	49.7 %

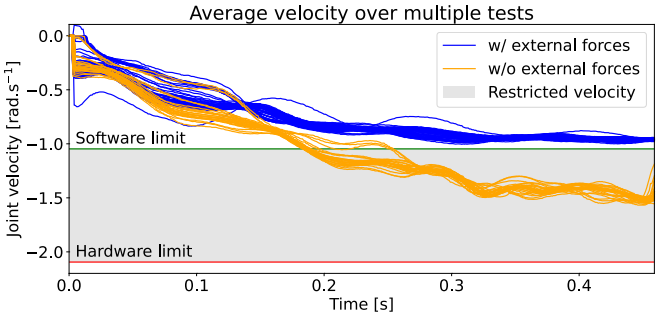

Fig. 4. Measurement of joint velocity under external forces. The *blue* and *yellow* lines represent velocities with and without external forces, respectively. It shows average velocities over 25 trials (*solid lines*), and each individual trials (*semi-transparent lines*). The *green* and *red* lines show the velocity limits defined in our QP and the robot's internal limit, respectively. The *grey* shaded region signifies the unreachable area.

and slow the motion such that it stays within the safe limits. On the other hand, when the external force is not included, we expect the QP to produce only the necessary torque to slow the motion of the robot as if no forces were applied.

2) Results: The results are displayed in Fig. 4. We can see when external forces are not included that the joint starts accelerating to the velocity limit due to the force applied, quickly reaching the limit without stopping and then slowing down due to damping being present on our posture task. With our method, when external forces are included in the constraint, we can see that the joint starts accelerating in a similar way but slows down appropriately so that it never exceeds the limit. Table II summarizes the results obtained over the trials including the average overrun of the velocity with respect to the limit defined in our control. Looking at the figure and the results in this table it's clear that by including external forces we are able to ensure safety limits, where without it's possible to go far past the limits which may become dangerous, lead to damage to the robot or triggering the robot's safeties leading to a complete stop of it.

C. Compliance

1) Experimental Setup: As our method includes explicitly the external forces thus compensating them, we need to explicitly define the compliance of a task, if we don't want external forces to be compensated by the task. When using the proposed method, it's possible to define a 6D task in the Cartesian space to fix the position of the end-effector, which is set as non-compliant. We can then define a posture task with a significantly lower weight as our null-space task. With this setup, we can achieve common scenarios of

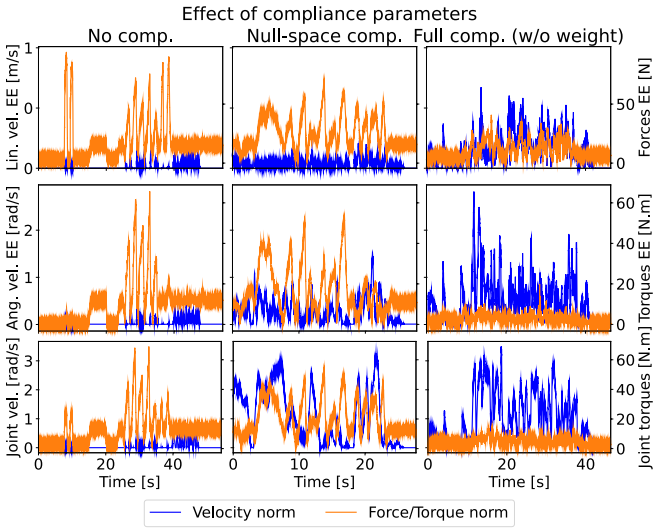

Fig. 5. Measurement of 6D cartesian and posture task velocities (*blue*) and the external forces/torques (*orange*) applied to them under three compliance setups. In the non-compliance setup, both tasks are stiff with $\gamma = 0$. In the null-space compliance setup, only the posture task is compliant with $\gamma = 1$. In the full compliance setup, both tasks are compliant with $\gamma = 1$, allowing for total robot manipulation.

compliance or not by setting the compliance parameter γ. With $\gamma = 0$ for both tasks, the control is set in non-compliant mode, in that case, external forces are compensated, and the motion of the robot is very small. From non-compliant mode, we can set $\gamma = 1$ on the posture task by doing that we can achieve null-space compliance for which external forces only get compensated for the 6D cartesian task at the end effector allowing free motion in the null-space with very small motion at the end effector. Finally, from the null-space compliance setup, we can achieve full compliance by setting $\gamma = 1$ to 6D cartesian task, doing so allows free manipulation of the robot in any degree of freedom.

2) Results: Fig. 5 summarizes the effect of compliance setups by showing the linear and angular velocity norm of the 6D task and joint velocity norm. This was tested with a weight attached at the end-effector of the robot where it can be sensed by F/T frame and with additional external forces applied manually, we only removed the weight for full compliance as it was making the robot fall.

V. Conclusion

We have demonstrated transparent compliance for torque-controlled robots while ensuring respect for safety constraints like velocity limits. This was achieved through explicit consideration of the external torque within the QP-based motion solver, allowing us to set an acceleration reference that complies with external forces. We posit that this method can be extended to other controllers such as hierarchical solvers and projection-based approaches.

Our method, which combines force torque sensor readings with residual-based estimation, has demonstrated good performance. It enhance the bandwidth of external force measurements and retains force estimation at any contact point. We believe it could enhance control in human-robot interactions, ergonomics, and user perception.

Potential extensions of our compliance control include handling robots with poor torque tracking. Our method builds a reference acceleration that complies with forces, providing an explicit compliant trajectory. This could improve kinematic tracking for better compliance perception, especially when forces are measured through a force/torque sensor.

Future work may focus on integrating force limit constraints to enhance control safety in stiff tasks, transitioning to hierarchical control for better feasibility, and adapting the framework for floating base robots like legged ones to handle contact force constraints and friction considerations.

While our method ensures a set of constraints at a more foundational level , task-level extensions such as task scaling could improve robot behavior when deviating from target trajectories due to external forces, preventing unpredictable robot motion.

References

[1] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1507–1527, 2009.

[2] D. Kortenkamp, R. Simmons, and D. Brugali, *Robotic Systems Architectures and Programming*. Cham: Springer International Publishing, 2016, pp. 283–306.

[3] I. O. for Standardization, "Iso standard no. 10218," Robots and robotic devices — Safety requirements for industrial robots, 2011, retrieved from https://www.iso.org/standard/51330.html.

[4] M. V. Minniti, R. Grandia, K. Fäh, F. Farshidian, and M. Hutter, "Model predictive robot-environment interaction control for mobile manipulation tasks," in *IEEE ICRA*. IEEE, 2021.

[5] A. Calanca, R. Muradore, and P. Fiorini, "A review of algorithms for compliant control of stiff and fixed-compliance robots," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, pp. 613–624, 2016.

[6] M. Schumacher, J. Wojtusch, P. Beckerle, and O. von Stryk, "An introductory review of active compliant control," *Robotics and Autonomous Systems*, vol. 119, pp. 185–200, Sept. 2019.

[7] T. Hoshi and H. Shinoda, "Robot skin based on touch-area-sensitive tactile element," in *IEEE ICRA*. IEEE, 2006.

[8] G. Cheng, E. Dean-Leon, F. Bergner, J. R. G. Olvera, Q. Leboutet, and P. Mittendorfer, "A comprehensive realization of robot skin: Sensors, sensing, control, and applications," *Proceedings of the IEEE*, 2019.

[9] A. De Luca, A. Albu-Schaffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in *IEEE/RSJ IROS*. IEEE, 2006.

[10] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, "Task-space control of robot manipulators with null-space compliance," *IEEE Transactions on Robotics*, 2013.

[11] M. Benallegue, R. Cisneros, A. Benallegue, A. Tanguy, A. Escande, M. Morisawa, and F. Kanehiro, "On compliance and safety with torque-control for robots with high reduction gears and no joint-torque feedback," in *IEEE/RSJ IROS*. IEEE, 2021.

[12] W. Wang, R. N. Loh, and E. Y. Gu, "Passive compliance versus active compliance in robot-based automated assembly systems," *Industrial Robot: An International Journal*, 1998.

[13] Y. Huang, S. Li, and Q. Huang, "Robust impedance control for seas," *Journal of the Franklin Institute*, 2020.

[14] E. Spyrakos-Papastavridis, P. R. Childs, and J. S. Dai, "Passivity preservation for variable impedance control of compliant robots," *IEEE/ASME Transactions on Mechatronics*, 2019.

[15] R. V. Ham, T. G. Sugar, B. Vanderborght, K. W. Hollander, and D. Lefeber, "Compliant actuator designs," *IEEE Robotics & Automation Magazine*, 2009.

[16] T. Lefebvre, J. Xiao, H. Bruyninckx, and G. De Gersem, "Active compliant motion: a survey," *Advanced Robotics*, 2005.

[17] M. Polic, M. Car, F. Petric, and M. Orsag, "Compliant plant exploration for agricultural procedures with a collaborative robot," *IEEE Robotics and Automation Letters*, 2021.

[18] R. Natarajan, G. L. Johnston, N. Simaan, M. Likhachev, and H. Choset, "Torque-limited manipulation planning through contact by interleaving graph search and trajectory optimization," *arXiv preprint arXiv:2210.08627*, 2022.

[19] Z.-J. Li, H.-B. Wu, J.-M. Yang, M.-H. Wang, and J.-H. Ye, "A position and torque switching control method for robot collision safety," *Intl. Journal of Automation and Computing*, 2018.

[20] E. Magrini, F. Flacco, and A. De Luca, "Control of generalized contact motion and force in physical human-robot interaction," in *IEEE ICRA*. IEEE, 2015.

[21] Y. Zhang, S. S. Ge, and T. H. Lee, "A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2004.

[22] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," in *IEEE/RSJ IROS*. IEEE, 2016.

[23] B. Faverjon and P. Tournassoud, "A local based approach for path planning of manipulators with a high number of degrees of freedom," in *IEEE ICRA*. IEEE, 1987.

[24] J. Vaillant, A. Kheddar, H. Audren, F. Keith, S. Brossette, A. Escande, K. Bouyarmane, K. Kaneko, M. Morisawa, P. Gergondet, *et al.*, "Multi-contact vertical ladder climbing with an hrp-2 humanoid," *Autonomous Robots*, 2016.

[25] K. Bouyarmane and A. Kheddar, "On weight-prioritized multitask control of humanoid robots," *IEEE Transactions on Automatic Control*, 2017.

[26] R. Cisneros, M. Benallegue, A. Benallegue, M. Morisawa, H. Audren, P. Gergondet, A. Escande, A. Kheddar, and F. Kanehiro, "Robust humanoid control using a qp solver with integral gains," in *IEEE/RSJ IROS*. IEEE, 2018.