

Impact-Robust Posture Optimization for Aerial Manipulation

Amr Afifi^{*1}, Ahmad Gazar^{*2}, Javier Alonso-Mora², Paolo Robuffo Giordano³, Antonio Franchi^{1,4}

Abstract—We present a novel method for optimizing the posture of kinematically redundant torque-controlled robots to improve robustness during impacts. A rigid impact model is used as the basis for a configuration-dependent metric that quantifies the variation between pre- and post-impact velocities. By finding configurations (postures) that minimize the aforementioned metric, *spikes* in the robot’s state and input commands can be significantly reduced during impacts, improving safety and robustness. The problem of identifying impact-robust postures is posed as a min-max optimization of the aforementioned metric. To overcome the real-time intractability of the problem, we reformulate it as a gradient-based motion task that iteratively guides the robot towards configurations that minimize the proposed metric. This task is embedded within a task-space inverse dynamics (TSID) whole-body controller, enabling seamless integration with other control objectives. The method is applied to a kinematically redundant aerial manipulator performing repeated point contact tasks. We test our method inside a realistic physics simulator and compare it with the nominal TSID. Our method leads to a reduction (up to 51% w.r.t. standard TSID) of post-impact spikes in the robot’s configuration and successfully avoids actuator saturation. Moreover, we demonstrate the importance of kinematic redundancy for impact-robustness using additional numerical simulations on a quadruped and a humanoid robot, resulting in up to 45% reduction of post-impact spikes in the robot’s state w.r.t. nominal TSID.

I. INTRODUCTION

The last two decades have witnessed significant progress in the field of Aerial Manipulation [1]. A key industrial application is Non-Destructive Testing (NDT), where aerial manipulators (AMs) are required to perform multiple point-contact inspections in a fast manner, reducing asset downtime and providing significant economic benefits [2]. However, for robotic systems, including AMs, fast making and breaking contact remains a challenging manipulation skill. In particular, impacts—contacts that occur at non-zero speed—induce a sudden change of momentum over a short time scale

^{*}These authors contributed equally to this work.

This work was partially funded by the ANR-20-CE33-0003 “CAMP” project, the EU AUTOASSESS project, EU Grant agreement ID: 101120732, and by the Dutch Research Council NWO-NWA, within the “Acting under Uncertainty” (ACT) project (Grant No.NWA.1292.19.298))

¹Robotics and Mechatronics Department, Electrical Engineering, Mathematics, and Computer Science (EEMCS) Faculty, University of Twente, 7500 AE Enschede, The Netherlands. amrafifi142@gmail.com, schol@r-franchi.eu.

²Cognitive Robotics (CoR) Department, Delft University of Technology, 2628 CD Delft, The Netherlands. ahmad.gazar@pm.me; j.alonsomora@tudelft.nl.

³CNRS, Univ Rennes, Inria, IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France. E-mail: prg@irisa.fr

⁴Department of Computer, Control and Management Engineering, Sapienza University of Rome, 00185 Rome, Italy. schol@r-franchi.eu

(milliseconds). If not handled properly, impact events can potentially lead to *spikes* in robot actuator commands and in extreme cases to even actuator saturation [3]. Such phenomena can significantly compromise both the safety and robustness of the robot during impact events.

Operational space control is widely used for motion and force control of robotic systems [4]. Task Space Inverse Dynamics (TSID) extends this framework by solving a Quadratic Program (QP) with multiple prioritized tasks in the operational space while respecting the robot’s full dynamics and constraints, making it well-suited for kinematically redundant robots [5]–[9]. However, in its basic formulation, TSID is impact-unaware, as it assumes smooth contact dynamics and does not explicitly model the instantaneous velocity changes caused by impacts. These changes are typically managed through high-frequency feedback loops, which may be insufficient for handling impacts.

The question of how to deal with impacts effectively has been studied in the robotics community, with a focus on fixed-base manipulators and legged robots. In [10], [11], a control technique known as reference spreading is proposed. Reference spreading smooths out the reference target used in the feedback controller by “spreading” signal discontinuities over a short time window of the impact event. A limitation of this approach is that the tracking performance of the controller can degrade if the impact time is uncertain. Another approach in [3] defines a subspace of generalized velocities that remains unaffected by any contact impulse. Feedback control objectives (e.g., velocity tracking errors) are projected onto this impact-invariant subspace, which results in the controller *ignoring* parts of the state that are unpredictable due to impact, while still exerting full authority over the invariant dynamics. Although this approach is robust to any possible contact impulse, it can lead to overly conservative behavior. In [12], an impact-aware TSID formulation is proposed that explicitly models sudden state jumps (in joint velocity and torque) due to impacts. In this QP formulation, constraints are adjusted to anticipate these jumps within robot hardware limits, preventing controller infeasibility post-impact. However, one limitation of this approach lies in the challenge to accurately identify the post-impact velocity and torque bounds without being overly conservative, and it does not minimize the impact effect. The method closest to our proposed approach is presented in [13], where kinematic redundancy is exploited to optimize robot configurations for improved impact handling via differential inverse kinematics and a projected gradient approach. However, this is only suitable for velocity-controlled robots and does not allow for multiple task objectives and the inclusion

of hard constraints.

Impact-handling approaches are still relatively unexplored in aerial manipulation contexts [14]. For AMs, impacts have been addressed by leveraging energetic concepts, for example, in [15], where a passivity-based framework including energy-tanks is used to enable safe interaction with any passive environment. Impacts have also been dealt with via hardware designs, as in [16], [17].

In this work, we tackle impacts by exploiting kinematic redundancy for improving a robot’s **impact-robustness**, that is, to reduce the risk of spikes in both the state and control commands and consequently avoid actuator saturation. Inspired by [13], we adopt a rigid impact model to derive a configuration-dependent metric that quantifies the robot’s susceptibility to impacts. We propose a whole-body TSID controller that includes the search for an impact-robust posture within its cost objective. Different from [13], our QP-based approach is suitable for torque-controlled robots and allows for multiple task objectives and hard constraints applied to complicated, highly dimensional robotic systems. Our method is tailored for anticipated impact tasks, i.e., contact tasks where impacts are expected to occur. Consequently, the robot can assume a “brace for impact” posture before the actual impact. Our results show that our method effectively optimizes postures that reduce the risk of actuator saturation and lead to safer and more robust performance during impacts. Since our approach achieves impact-robustness via redundant degrees of freedom (DoF), our method is not conservative and does not compromise the task-space feedback control action as in [3], [10], [11]. Moreover, our approach does not require the identification of post-impact state bounds as in [12], which can affect the controller’s feasibility.

Our contributions can be summarized as follows:

- We propose a novel and generic method to improve robustness during impacts by optimizing the postures of kinematically redundant torque-controlled robots. The optimization is formulated as a motion task within a TSID whole-body QP-controller.
- Using a kinematically redundant AM, We test and validate our method in a realistic physics simulator on multiple contact NDT-like scenarios. Our method leads to a significant reduction of post-impact robot motion, w.r.t. nominal TSID, up to 51% and avoids actuator saturation during impact events.
- We confirm the generality of the approach in a multi-contact, high kinematic redundancy setting of legged robots, by incorporating the impact-robustness task in a whole-body controller tailored for legged robots. The validity of the approach is confirmed via numerical simulations on a humanoid and quadruped robot. Again, our method results in a reduction of post-impact motion in the configuration (up to 45% w.r.t. nominal TSID).

II. PRELIMINARIES

In this section, we define our system dynamics modeling and present the feedback control challenges that robots face

when interacting physically with the environment.

A. Floating Base Robot Dynamics

The robots considered in this work are modeled as floating-base systems. The robot configuration is $\mathbf{q} = (\mathbf{q}_b^T, \mathbf{q}_j) \in \mathcal{M} = \mathbb{SE}(3) \times \mathbb{R}^{n_j}$, where \mathbf{q}_b^T represents the floating-base pose, expressed in the inertial frame \mathcal{I} , while \mathbf{q}_j denotes the joint positions of the robot, with n_j Degrees of Freedom (DoF). The generalized robot velocities are denoted as $\boldsymbol{\nu} \in \mathbb{R}^{6+n_j}$. The equations of motion of such a system can be derived using the Euler-Lagrange equations of motion as [18]:

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} + \mathbf{h}(\mathbf{q}, \boldsymbol{\nu}) = \mathbf{G}(\mathbf{q})\mathbf{u} + \mathbf{J}_c(\mathbf{q})^T \mathbf{F}_c. \quad (1)$$

The inertia matrix is denoted as $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{(6+n_j) \times (6+n_j)}$, and $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{6+n_j}$ is the vector capturing the Coriolis, centrifugal, and gravity effects. \mathbf{J}_c represents the contact Jacobian associated with the generalized contact forces \mathbf{F}_c .

B. Aerial Manipulator Actuation Model

The AM considered in this work is composed of a fully actuated hexarotor with fixedly tilted propellers and a 3-DoF robotic arm attached to its base. The AM can be modeled as a floating base system as in Eq. (1) where $n_j = 3$. The actuated control commands are denoted as $\mathbf{u} = [\mathbf{u}_b^T, \boldsymbol{\tau}^T]^T$, where $\mathbf{u}_b \in \mathbb{R}^6$ represent the propellers’ angular velocities of the hexarotor, and $\boldsymbol{\tau} \in \mathbb{R}^3$ being the manipulators’ joint torques. Each propeller angular velocity ω_i , $\forall i = 1, \dots, 6$ is related to its thrust and drag moment, in good approximation, quadratically. For more details, we refer the interested reader to [19]. The AM control inputs are then mapped to the state tuple using the actuation matrix $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^{n \times m}$.

C. Whole Body Control as a QP

Whole-body control of kinematically redundant robots can be formulated using TSID [5]–[8]. Given the current state of the robot (\mathbf{q} and $\boldsymbol{\nu}$), the control objective is to find the set of controls \mathbf{u} , and joint accelerations $\dot{\boldsymbol{\nu}}$, that minimize the residual of a stack of tasks in the operational space subject to the dynamics and physical constraints of the robot. This problem can be formulated as a QP to achieve real-time instantaneous feedback control as follows:

Problem 1 Vanilla TSID

$$\underset{\mathbf{z} := (\mathbf{u}, \dot{\boldsymbol{\nu}})}{\text{minimize}} \sum_{i=1}^{N_{\text{tasks}}} w_i \|e_i(\mathbf{z})\|^2 \quad (2a)$$

subject to

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} + \mathbf{h}(\mathbf{q}, \boldsymbol{\nu}) = \mathbf{G}(\mathbf{q})\mathbf{u} + \mathbf{J}_c(\mathbf{q})^T \mathbf{F}_c, \quad (2b)$$

$$\underline{\mathbf{u}} \leq \mathbf{u} \leq \bar{\mathbf{u}}, \quad \underline{\dot{\boldsymbol{\nu}}} \leq \dot{\boldsymbol{\nu}} \leq \bar{\dot{\boldsymbol{\nu}}}. \quad (2c)$$

The total optimizers are denoted as \mathbf{z} (acceleration and controls). The cost objective (2a) is the summation of the weighted l_2 norm of the task acceleration residual $e_i :=$

$\ddot{\mathbf{y}}_i - \ddot{\mathbf{y}}_i^*$, where $w_i, \forall i = 1, \dots, N_{\text{tasks}}$ is the i th task weight, and

$$\ddot{\mathbf{y}} = \mathbf{J}_i(\mathbf{q})\dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_i(\mathbf{q})\boldsymbol{\nu}, \quad (3a)$$

$$\ddot{\mathbf{y}}_i^* = \underbrace{\ddot{\mathbf{y}}_{\text{des}_i}}_{\mathbf{u}_{\text{ffwd}}} + \underbrace{\mathbf{K}_p(\mathbf{y}_{\text{des}_i} - \mathbf{y}) + \mathbf{K}_d(\dot{\mathbf{y}}_{\text{des}_i} - \dot{\mathbf{y}})}_{\mathbf{u}_{\text{fdbk}}}. \quad (3b)$$

The output task acceleration $\ddot{\mathbf{y}} \in \mathbb{R}^{n_t}$ in (3a) is achieved using the desired output-feedback controller $\ddot{\mathbf{y}}_i^*$ in (3b) comprising a feedforward acceleration command $\ddot{\mathbf{y}}_{\text{des}_i}$, and a feedback control action with $\mathbf{K}_p \succ 0$, and $\mathbf{K}_d \succ 0$ being the positive definite position and velocity task feedback gains respectively. $\mathbf{J}_i(\mathbf{q}) \in \mathbb{R}^{n_t \times n}$ represents the task Jacobian that maps the joint velocities to the operational space velocity, with n_t being the task DoF. Finally, (2c) represent the lower and upper bounds on the controls and joint accelerations, respectively.

Remark 1 *The task residuals in problem (1) in the cost objective are realized in a weighted fashion. An alternative approach can be enforcing a strict hierarchy between tasks, where lower-priority tasks are projected into the null space of the higher-priority tasks [5]. This hierarchical approach can be particularly advantageous for our AM application, as will be presented in the subsequent sections.*

D. Feedback Control Challenges during Impacts

Feedback control of robots in contact with the environment is a challenging task due to the fast, discontinuous nature of impacts that occur in orders of milliseconds [20]. We summarize this challenge here for completeness, as highlighted in [3] as a motivation for our approach. Assuming a robot makes contact with a rigid body, the contact is established instantaneously (i.e., the robot's configuration \mathbf{q} remains constant during the impact event, assuming no contact slippage). Simultaneously, there is an instantaneous change in the robot's velocity due to the impulse $\boldsymbol{\lambda}$. Mathematically, this rigid impact model can be written as follows:

$$\mathbf{q}^+ = \mathbf{q}^- \quad (4a)$$

$$\boldsymbol{\nu}^+ - \boldsymbol{\nu}^- = \mathbf{M}^{-1}(\mathbf{q})\mathbf{J}_c(\mathbf{q})^T \boldsymbol{\lambda}, \quad (4b)$$

where, \mathbf{q}^+ , \mathbf{q}^- , and $\boldsymbol{\nu}^+$, $\boldsymbol{\nu}^-$ are the post- and pre-impact configurations and velocities respectively. Consider the following PD feedback controller similar to (3b):

$$\mathbf{u}_{\text{fdbk}} = \mathbf{K}_p \mathbf{e}_p + \mathbf{K}_d \mathbf{e}_\nu, \quad (5)$$

where \mathbf{e}_p , and \mathbf{e}_ν are the position and velocity errors simultaneously. According to the rigid impact model assumption in (4), the magnitude of the feedback command during the impact is dominated by the change in velocity error since the configuration change is almost zero. This implies the following variational relationship at the impact event:

$$\Delta \mathbf{u}_{\text{fdbk}} \approx \mathbf{K}_d \Delta \mathbf{e}_\nu. \quad (6)$$

In simple words, large changes in the velocity feedback error exhibit large feedback residuals in problem (1), which in turn lead to large control commands in milliseconds

duration (spikes) that can potentially be infeasible for the robot actuators. This can degrade the controller's performance and hinder the robot's safety. The above analysis suggests that feedback alone may not suffice for controlling robots during impacts. We highlight that this analysis is generally applicable to feedback controllers during impact events, whether in joint space or operational space. In the following sections, we propose a method that demonstrates how to incorporate anticipated impacts into feedforward control commands to enhance the robustness of reactive controllers, particularly in kinematically redundant, optimization-based TSID controllers.

III. METHOD: OPTIMIZING IMPACT-ROBUST POSTURES

We aim to minimize the variation between the pre- and post-impact velocities, resulting in smoother control commands and improved robustness during impact events.

Assumption 1 (Anticipated bounded impacts) *We assume anticipated impacts, i.e., impacts that are part of a contact task and for which the prospective contact Jacobian $\mathbf{J}_c(\mathbf{q})$ and its contact normal \mathbf{n} are known a priori, with the impact magnitude λ along \mathbf{n} being a bounded uncertainty $|\lambda| \leq \bar{\lambda}$.*

Given Assumption 1, expanding the impulse vector at the velocity jump with its magnitude and direction becomes:

$$\Delta \boldsymbol{\nu} = \mathbf{M}^{-1}(\mathbf{q})\mathbf{J}_c^T \mathbf{n} \lambda =: \boldsymbol{\Gamma}(\mathbf{q}) \mathbf{n} \lambda, \quad (7a)$$

$$\boldsymbol{\Gamma}(\mathbf{q}) \triangleq \mathbf{M}^{-1}(\mathbf{q})\mathbf{J}_c^T. \quad (7b)$$

To minimize the variation between pre- and post-impact velocities, we formulate the following robust optimization problem:

$$\min_{\mathbf{q}} \max_{|\lambda| \leq \bar{\lambda}} \|\Delta \boldsymbol{\nu}\|^2, \quad (8)$$

Substituting (7a), we obtain

$$\|\Delta \boldsymbol{\nu}\|^2 = \lambda^2 \mathbf{n}^T \underbrace{(\mathbf{J}_c \mathbf{M}^{-2} \mathbf{J}_c^T)}_{\boldsymbol{\Lambda}_c(\mathbf{q})} \mathbf{n}, \quad (9)$$

where $\boldsymbol{\Lambda}_c = (\mathbf{J}_c \mathbf{M}^{-2} \mathbf{J}_c^T)$ is known as the *Dynamic Impact Ellipsoid* [13], which is a form of the effective inertia in the operational-space. The shape of this ellipsoid, computed using its SVD decomposition, determines how much the robot's inertia varies in the contact space. Since the worst-case λ satisfies $|\lambda| = \bar{\lambda}$, the min-max problem reduces to

$$\min_{\mathbf{q}} \bar{\lambda}^2 \mathbf{n}^T \boldsymbol{\Lambda}_c(\mathbf{q}) \mathbf{n}. \quad (10)$$

The above optimization problem chooses a posture \mathbf{q} that maximizes the effective inertia along the impact direction. In other words, the robot is required to adjust its configuration to absorb impacts with larger body inertia in the impact direction.

Problem (10) is a nonlinear optimization problem and can be intractable to solve online. Instead, we propose

approximating it locally as a low-priority posture task within TSID. Define

$$H(\mathbf{q}) := \mathbf{n}^\top \Lambda_c(\mathbf{q}) \mathbf{n}, \quad (11a)$$

$$e_{\text{IM-posture}} = \dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\nu}}_{\text{IM-posture}}^*, \quad (11b)$$

$$\dot{\boldsymbol{\nu}}_{\text{IM-posture}}^* \triangleq -\mathbf{K} \nabla_{\mathbf{q}} H(\mathbf{q}) + \mathbf{K}_p(\mathbf{q}_{\text{des}} - \mathbf{q}) - \mathbf{K}_d \boldsymbol{\nu}, \quad (11c)$$

where \mathbf{q}_{des} is a desired robot configuration. $\mathbf{K} \succ 0$ is a tunable gain. We call $H(\mathbf{q}) \in \mathbb{R}$ the *Impact-Robustness Metric*, which quantifies the effect of impacts along a specific direction on the robot's effective inertia. The higher this number is, the higher the robot's effective inertia in that direction. Finally, $\nabla_{\mathbf{q}} H(\mathbf{q})$ is defined as the gradient of the impact-robustness metric with respect to \mathbf{q} .

The weighted l_2 norm for the impact-robustness task residual $e_{\text{IM-posture}}$ can be incorporated within TSID as part of the objective function in Eq. 2a, in addition to an appropriate weighting choice that represents its priority. The proposed task residual will iteratively reconfigure the robot's posture to achieve the local minimization of $H(\mathbf{q})$. Since we consider impacts which are part of a contact task, the robot can choose, e.g., based on a finite state machine, to activate this task residual (introduce a non-zero weight) before attempting the impact action. This highlights the flexibility of the method in being applied only when required by the task

IV. APPLICATION TO AERIAL MANIPULATION

In this section, we apply the proposed method to an AM. We focus on inspection tasks where AMs are required to make fast repeated contacts with the environment.

A. Aerial Manipulator Stack of Tasks

As mentioned in remark 1, different from problem (1), we adapt a strict hierarchy of tasks inside TSID that prioritizes the reduced-attitude (tilt) control of the fully-actuated multi-rotor as the highest priority task, the EE as a second priority, and the robot's posture as the third priority. Unlike earlier works on the control of fully actuated kinematically redundant AMs [21] [22], we prioritize the tilt angle control, as it has a strong influence on the robot's stability.

1) *Reduced Attitude Task*: We assign a desired tilt of the hexarotor base as a first priority task. This is achieved using the control command $\dot{\boldsymbol{\omega}}_{xy}^* \in \mathbb{R}^2$, specifying the angular acceleration of the hexarotor around its tilt axis (i.e., the x-y directions). The desired angular acceleration is then related to the joint accelerations $\dot{\boldsymbol{\nu}}$ through the Jacobian task matrix $\mathbf{J}_{\text{tilt}} \in \mathbb{R}^{2 \times 6+n_j}$ as:

$$\dot{\boldsymbol{\omega}}_{xy}^* = \mathbf{J}_{\text{tilt}} \dot{\boldsymbol{\nu}}, \quad (12a)$$

$$\mathbf{J}_{\text{tilt}} \triangleq \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (12b)$$

The reduced attitude error $e_{\text{red}} \in \mathbb{R}^2$ is a measure of the error between the aerial robot's body frame's z-axis and the z-axis of the desired frame which is calculated based on the shortest rotation that aligns both z-axes [23]. The desired reduced attitude controller is then calculated as

$$\dot{\boldsymbol{\omega}}_{xy}^* = \mathbf{K}_{\text{red}} e_{\text{red}} - \mathbf{K}_\omega \boldsymbol{\omega}_{xy}. \quad (13)$$

2) *End-Effector Pose Task*: The second priority task controls the EE pose. We consider a desired EE impedance command [24] $\dot{\boldsymbol{\nu}}_e^* \in \mathbb{R}^6$, with the following task kinematics expressed at the acceleration level as

$$\dot{\boldsymbol{\nu}}_e^* = \mathbf{J}_e(\mathbf{q}) \dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_e(\mathbf{q}) \boldsymbol{\nu}, \quad (14)$$

where $\mathbf{J}_e(\mathbf{q}) \in \mathbb{R}^{6 \times (6+n_j)}$ is the EE task Jacobian. We impose that the EE pose task does not affect the reduced attitude task, by projecting it in its null-space as follows:

$$\dot{\boldsymbol{\nu}}_e^* = \mathbf{J}_e(\mathbf{q}) \mathbf{N}_{\text{tilt}} \dot{\boldsymbol{\nu}} + \mathbf{J}_e(\mathbf{q}) \mathbf{J}_{\text{tilt}}^\dagger \dot{\boldsymbol{\omega}}_{xy} + \dot{\mathbf{J}}_e(\mathbf{q}) \boldsymbol{\nu}, \quad (15)$$

$$\mathbf{N}_{\text{tilt}} \triangleq (\mathbf{I} - \mathbf{J}_{\text{tilt}}^\dagger \mathbf{J}_{\text{tilt}}), \quad (16)$$

where \mathbf{N}_{tilt} is the null-space projection matrix in the reduced attitude task [4], \mathbf{I} is the identity matrix with the respected size. The superscript \dagger denotes the Moore-Penrose inverse.

3) *Posture Task*: The third priority task comprises a desired posture of the AM. This task is incorporated as the least square error in the cost objective of the OCP. Notice that this cost is equivalent to projecting the task in the null space of the higher-priority task.

Finally, we can write down the nominal TSID QP that solves for the above stack of tasks.

Problem 2 Nominal TSID for Aerial Manipulation

$$\underset{\mathbf{u}, \dot{\boldsymbol{\nu}}}{\text{minimize}} \quad \|\dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\nu}}_{\text{Nom-posture}}^*\|_{\mathbf{W}}^2 \quad (17a)$$

subject to

$$\mathbf{M}(\mathbf{q}) \dot{\boldsymbol{\nu}} + \mathbf{h}(\mathbf{q}, \boldsymbol{\nu}) = \mathbf{G}(\mathbf{q}) \mathbf{u} + \mathbf{J}_c(\mathbf{q})^T \hat{\mathbf{F}}_c, \quad (17b)$$

$$\dot{\boldsymbol{\omega}}_{xy}^* = \mathbf{J}_{\text{tilt}} \dot{\boldsymbol{\nu}}, \quad (17c)$$

$$\dot{\boldsymbol{\nu}}_e^* = \mathbf{J}_e(\mathbf{q}) \mathbf{N}_{\text{tilt}} \dot{\boldsymbol{\nu}} + \mathbf{J}_e(\mathbf{q}) \mathbf{J}_{\text{tilt}}^\dagger \dot{\boldsymbol{\omega}}_{xy} + \dot{\mathbf{J}}_e(\mathbf{q}) \boldsymbol{\nu}, \quad (17d)$$

$$\underline{\mathbf{u}} \leq \mathbf{u} \leq \bar{\mathbf{u}}, \quad \underline{\dot{\boldsymbol{\nu}}} \leq \dot{\boldsymbol{\nu}} \leq \bar{\dot{\boldsymbol{\nu}}}, \quad (17e)$$

where the desired nominal posture controller is

$$\dot{\boldsymbol{\nu}}_{\text{Nom-posture}}^* = \mathbf{K}_p(\mathbf{q}_{\text{des}} - \mathbf{q}) - \mathbf{K}_d \boldsymbol{\nu}. \quad (18)$$

$\mathbf{W} \succeq 0$ denotes the posture task weight, and $\hat{\mathbf{F}}_c$ represents the measured contact forces.

Remark 2 Problems (1) and (2) do not account for post-impact velocities, relying instead on fast feedback loops to address impact disturbances during contact establishment, which leads to control spikes as presented in Sec. II-D. This remains a challenging research question for robots performing contact-rich tasks, such as loco-manipulation [3], [12].

To address this drawback in nominal TSID feedback controllers, we leverage the developments in Sec. III to include the impact-robustness residual in the cost function as a remedy to account for post-impact impulses that can be difficult for only feedback to handle. We formulate the following OCP:

Problem 3 Impact-Robust TSID for Aerial Manipulation

$$\underset{\mathbf{u}, \dot{\boldsymbol{\nu}}}{\text{minimize}} \quad \|\dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\nu}}_{\text{IM-posture}}^*\|_{\mathbf{W}}^2 \quad (19a)$$

subject to

$$(17b), (17c), (17d), (17e).$$

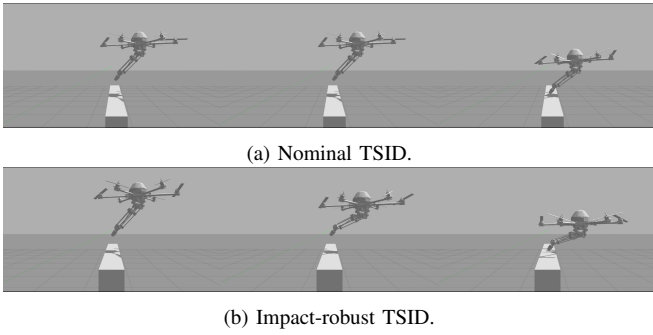


Fig. 1: Left to right: snapshots of robot configurations in the vertical contact scenario.

The desired impact posture acceleration command becomes:

$$\dot{\mathbf{v}}_{\text{IM-posture}}^* = \dot{\mathbf{v}}_{\text{Nom-posture}}^* - \mathbf{K} \nabla_{\mathbf{q}} H(\mathbf{q}). \quad (20)$$

Notice that the only difference between the impact-robust TSID problem above and the (2) is the gradient of the impact-robustness term in the desired posture acceleration. Thanks to this term, the impact-robust TSID controller utilizes the robot's kinematic redundancy to iteratively optimize postures that are robust to anticipated impacts.

Remark 3 *Since the impact direction is known a priori (Assumption (1)), $\nabla_{\mathbf{q}} H(\mathbf{q})$ can be computed offline using an off-the-shelf automatic differentiation tools, which results in a negligible computational overhead online w.r.t. the nominal TSID in problem (2) (an extra matrix-vector multiplication).*

V. RESULTS

We evaluate our method in the Gazebo physics-based simulator¹ using the Open Dynamics Engine (ODE)² on a fully actuated hexarotor equipped with a 3-DoF robotic arm. The open-source framework telekyb3³ interfaces the robot software stack with the simulator, incorporating Gazebo plugins to model rotor dynamics via a first-order motor model. To enhance simulation fidelity, we model propeller-induced vibrations on the hexarotor body, which affect the simulated IMU measurements. For controller development, we use the Whole-Body toolbox [25] to compute dynamic quantities (e.g., inertial, Coriolis, and gravitational matrices) of the aerial manipulator model. The derivative of the impact-robustness metric with respect to the robot configuration, $\nabla_{\mathbf{q}} H(\mathbf{q})$, is computed using ADAM⁴ and CasADi auto-differentiation [26], which is essential for the impact-robustness residual task in Problem 3. Both nominal and impact-robust TSID whole-body control problems are solved using the open-source QP solver qpOASES [27].

A. Aerial Manipulation Scenarios and Robustness metrics

The AM is tasked with making and breaking contacts repeatedly with a known contact surface normal using its

¹<https://gazebo.org/home>

²<https://www.ode.org/>

³<https://git.openrobots.org/projects/telekyb3>

⁴<https://github.com/ami-iit/adam>

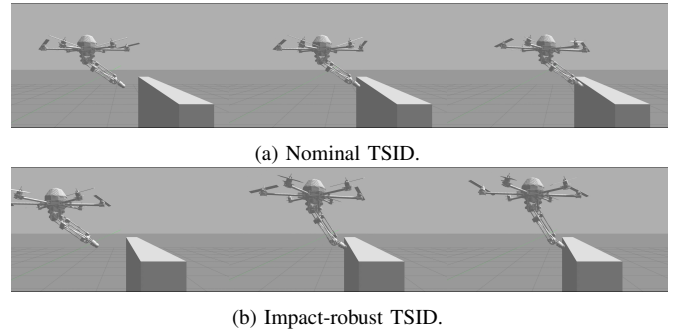


Fig. 2: Left to right: snapshots of robot configurations in the lateral contact scenario.

EE, similar to a contact inspection task. In particular, we devise the following contact scenarios:

- **Vertical pushing:** five repeated vertical contacts (Fig. 1).
- **Lateral pushing:** five repeated lateral contacts (Fig. 2).

The robot's whole-body controller is tasked with maintaining zero tilt of the floating base of the hexarotor as its highest priority task (Sec. IV-A.1). Moreover, the robot controls a desired EE pose as the second priority (Sec. IV-A.2). Finally, the robot is assigned a desired posture as the third priority task (Sec. IV-A.3). We compare the robustness of our Impact-robust TSID controller in problem (3) that optimizes for the impact-robustness residual (20) against the nominal TSID controller problem (2) that is impact agnostic. We use the following metrics to quantify the robustness of both controllers during the impact events:

- **Constraint saturation:** We consider the saturation of the control commands during impact events as a critical robustness metric, since constraint saturation leads to unsafe motions.
- **Motion during impacts:** this metric quantifies the change in robot configuration during impact events, computed as the integral of the l_2 norm of the actual robot velocity $\boldsymbol{\nu}_a$

$$\mathbf{q}_{\text{impact}}^{\text{total}} = \int_{t=0}^{t=T} \|\boldsymbol{\nu}_a(t)\| dt, \quad (21)$$

where T is the sum of impact durations.

B. Aerial Manipulation Simulations

We plot the impact-robustness metric $H(\mathbf{q})$ in Fig. 3, which serves as a feedforward term in the desired impact-robustness residual (Eq. (20)). Recall that this term drives the robot's whole-body reconfiguration to achieve impact-robust postures. Starting from the same initial configuration for both controllers, the impact-robust TSID controller adjusts the robot's posture to minimize $H(\mathbf{q})$ while maintaining the desired EE pose in both test scenarios. As shown in Fig. 3, the impact-robust TSID significantly reduces $H(\mathbf{q})$ for both the vertical and lateral contact scenarios compared to the nominal TSID, proactively accounting for impact disturbances. This reconfiguration can be visualized in Figs. 1 and

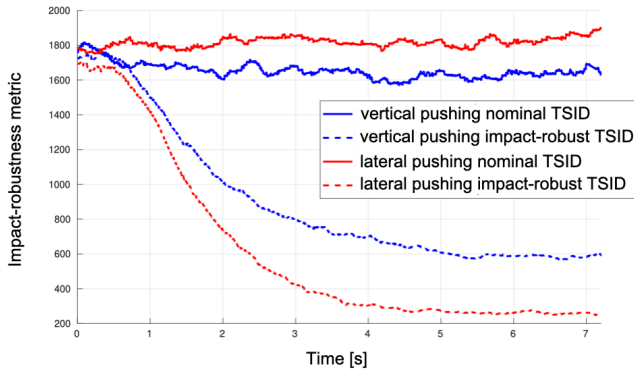


Fig. 3: The evolution of the impact-robustness metric $H(\mathbf{q})$ for the vertical and lateral contact scenarios.

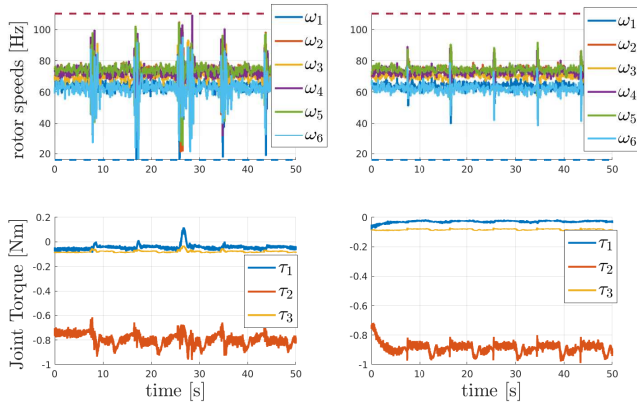


Fig. 4: AM's whole-body controls (rotors' angular velocities, and joint torques) for the vertical pushing scenario. Left: nominal TSID controller saturates the angular velocities control limits during impacts. Right: impact-robust TSID controller optimizes smoother and safer control inputs without saturation.

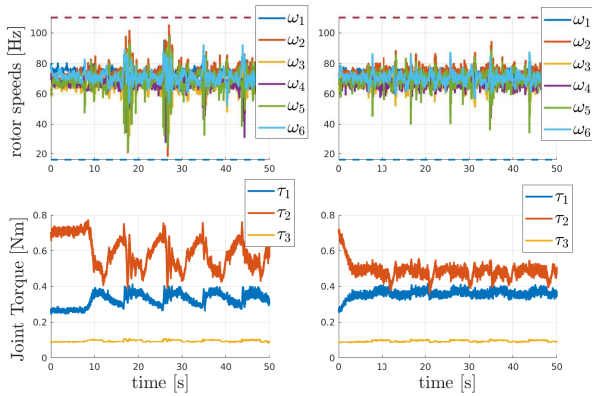


Fig. 5: AM's whole-body controls (rotors' angular velocities, and joint torques) for the lateral pushing scenario. Left: nominal TSID controller. Right: impact-robust TSID controller.

2 before the impact event. Thanks to this reconfiguration, it becomes evident that the impact-robust TSID absorbs impacts more gracefully than the nominal TSID (see the multimedia attachment for reference).

Test Scenario	$q_{\text{impact}}^{\text{total}}$	nominal TSID	$q_{\text{impact}}^{\text{total}}$	impact-robust TSID
Vertical push		10.4		5.1
Lateral push		9.2		5.5
Quadruped		0.411		0.342
Humanoid		0.305		0.166

TABLE I: Comparison of the integral of the l_2 norm of the joint velocities over impact duration $q_{\text{impact}}^{\text{total}}$.

As a consequence of minimizing the impact-robustness metric $H(\mathbf{q})$ in problem (3), the impact-robust TSID controller avoids saturating control limits (rotor angular velocities and joint torques) and generates smoother control commands compared to the nominal TSID, as shown in Fig. 4 and Fig. 5 for the vertical and lateral contact scenarios, respectively. In contrast, the nominal TSID frequently saturates rotor angular velocity limits and produces aggressive control actions during impacts, since it is agnostic to impacts and relies solely on feedback. Such control saturation compromises robot safety and may cause motion failures in real-world applications.

Finally, we quantified the change in robot configuration during impact events $q_{\text{impact}}^{\text{total}}$ in (21) as our second safety quantification measure. As reported in Table I, impact-robust TSID compensates better for impact disturbances, which in turn resulted in less motion during contact events (51%, and 40.2%) w.r.t. to nominal TSID for the vertical and lateral pushing scenarios, respectively. The joint configuration profiles are plotted in Fig. 6 for the lateral pushing scenario for completeness, which shows the change in joint configurations during the repeated impacts (pink rectangles). As shown in the figure, the robot state varies less during impact for the impact-robust TSID w.r.t. the nominal TSID, thanks to the impact-residual minimization of the posture task (20).

C. Application to contact-rich redundant robots

We complement our results with additional numerical simulations performed on highly dimensional multi-contact scenarios for legged robots. We test our approach on the humanoid robot Talos [28] and the open-source quadruped robot Solo [29]. Such simulations aim to emphasize the role kinematic redundancy plays in improving impact robustness, which becomes more evident for such systems. Due to the high dimensionality of these systems and the multi-contact nature, we added the following tasks to both the nominal TSID problem (2) and the impact-robust TSID problem (3), which are essential for stabilizing the robot.

- Center of mass position task: To regulate the configuration of the floating base.
- Friction pyramid constraints: We additionally optimize the contact forces to give more DoFs to the solver, and constrain them to be inside the friction pyramids to ensure dynamic feasibility of the contact forces.
- Non-holonomic contact tasks: To constrain the kinematic motion at the robots' end effectors in contact.
- A posture task: to regulate a desired robot posture in the joint space.

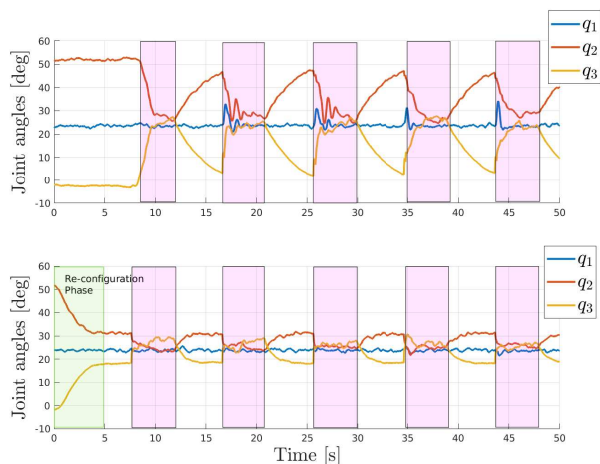


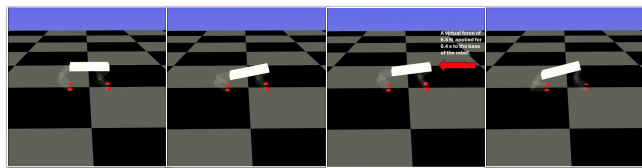
Fig. 6: Evolution of the joint angles profiles during impact events (pink rectangles). Top: nominal TSID. Bottom: impact-robust TSID (ours).

Similarly, we compare our impact-robust TSID approach to the nominal TSID for disturbance compensation for stabilizing an initial configuration of the robot. First, we simulate an impulse force of 6.5 N for 400 ms in the sagittal direction of the base of the quadrupedal robot Solo as shown in Fig. 7. This is an interesting case since this version of Solo legs has only 8 DoF, which limits the base rotation around the sagittal axis. Despite such a kinematic disadvantage, the impact-robust TSID controller was able to reconfigure the base upwards along the tangential direction to compensate for the force in the sagittal direction (check top row Fig. 7). Similar to the AM, we report how much the norm of the robot configuration changed during impact events $\|q_{\text{impact}}^{\text{total}}\|$ for both controllers, as shown in Table I. Thanks to the impact-robust residual that drives such reconfiguration before impact (20), the impact-robust TSID controller allowed the robot to move less (16.8% w.r.t. the nominal TSID).

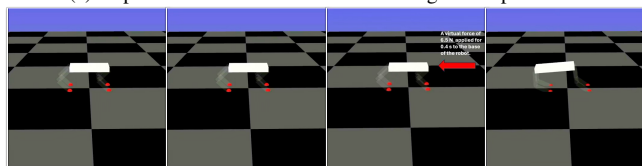
This metric is even more evident in the humanoid robot scenario due to the high kinematic redundancy of Talos. This time we simulate a higher impact of 32 N for 400 ms. As shown in Fig. 8, the impact-robust TSID used the robot’s kinematic redundancy as an advantage to reconfigure its upper-body to better compensate for the impact disturbance before the impact. On the other hand, the nominal TSID maintained its initial configuration as it is agnostic to the impact event. $\|q_{\text{impact}}^{\text{total}}\|$ is even more evident here, resulting in 45.6% w.r.t. the nominal TSID controller. This shows that impact-robust TSID can be used as a complementary robustness tool that is easily extendable to highly dimensional multi-contact systems.

VI. DISCUSSION

In this section, we outline the strengths and weaknesses of our method and compare it with state-of-the-art approaches. A key strength of our method is its implementation simplicity. By leveraging the rigid impact model residual, the impact-robustness metric is incorporated as an additional



(a) Impact-robust TSID moves less during the impact event.



(b) Nominal TSID moves more during the impact event.

Fig. 7: Snapshots of Solo reacting to an impulse force disturbance (red arrow) at the base of 6.5 N for 400 ms.

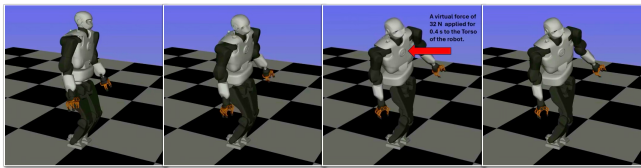
feedforward term within the desired posture acceleration task. Our method addresses impact discontinuities in a closed-loop feedback manner within a hierarchical TSID, utilizing gradient descent on the impact residual. In contrast, reference spreading handles this issue offline by smoothly blending pre- and post-impact reference states, which does not reduce the actual impulse but rather smooths the controller’s tracking target.

Another key advantage of our method is that it’s not conservative. Different from the work of [12] that enforces extra hard constraints to avoid infeasibilities of the post-impact acceleration, impact-robust TSID tackles the problem by finding impact-robust postures softly through a task residual in the cost prior to the impact. At the same time, a limitation of our method arises when the robot’s redundancy is exhausted, as the impact-robustness task is mapped to the null space of higher-priority tasks. Moreover, we differentiate our method from the work of [3], which takes a robust control approach by mapping the robust controls to an impact-invariant space. Despite being agnostic to impact magnitude, direction, and timing, this approach can be quite conservative and becomes hard to scale for highly dimensional systems like full humanoid robots. Our approach, on the other hand, assumes an impact direction as a trade-off for performance. Thanks to this assumption, impact-robust TSID does not add a computational overhead online w.r.t nominal TSID as the gradient of the impact-robustness term is computed offline.

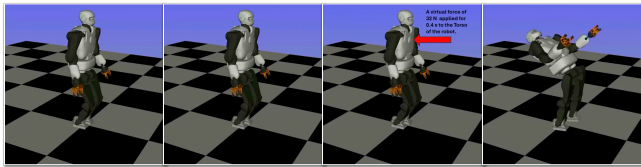
Finally, we view this work as a complementary robustness tool that can be used in combination with the state-of-the-art methods as [3], [9], [12], [30].

VII. CONCLUSIONS

In this work, we presented a novel approach to optimize impact-robust postures as a task residual inside TSID for kinematically redundant robots. Our results show that the optimized posture effectively reduces spikes during impact events, which avoids control saturations and infeasibilities. Consequently, this leads to smoother robot state profiles (joint positions and velocities) that minimize robot motion during impacts, resulting in safer and more stable robot



(a) Impact-robust TSID optimizes for a posture with lower upper-body momentum to compensate for the impact.



(b) Nominal TSID exerts higher upper-body angular momentum to compensate for the impact.

Fig. 8: Snapshots of Talos reacting to an impulse force disturbance (red arrow) at the torso of 32 N for 400 ms.

motions. Our method complements state-of-the-art methods in impact handling whenever kinematic redundancy is not exhausted, which is useful for highly dimensional multi-contact robotic systems. Future work includes extending this approach to consider uncertainty in the contact normal and applying our approach to real robot experiments.

REFERENCES

- [1] A. Ollero, M. Tognon, A. Suarez, D. Lee, and A. Franchi, “Past, present, and future of aerial robotic manipulators,” *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 626–645, 2022.
- [2] A. Ollero, A. Suarez, C. Papaioannidis, I. Pitas, J. M. Marredo, V. D. Hoang, E. Ebeid, V. Kratky, M. Saska, C. Hanoune, A. Afifi, A. Franchi, C. Vourtsis, D. Floreano, G. Vasiljevic, S. Bogdan, A. Caballero, F. Ruggiero, V. Lippielo, C. Matilla, G. Cioffi, D. Scaramuzza, J. R. M. d. Dios, B. C. Arrue, C. Martin, K. Zurad, C. Gaitan, J. Rodriguez, A. Muñoz, and A. Viguria, “Multi-aerial robotic system for power line inspection and maintenance: Comparative analysis from the aerial-core final experiments,” *IEEE Transactions on Field Robotics*, vol. 2, pp. 549–573, 2025.
- [3] W. Yang and M. Posa, “Impact invariant control with applications to bipedal locomotion,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5151–5158.
- [4] O. Khatib, “A unified approach for motion and force control of robot manipulators: The operational space formulation,” *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.
- [5] L. SENTIS and O. KHATIB, “Synthesis of whole-body behaviors through hierarchical control of behavioral primitives,” *International Journal of Humanoid Robotics*, vol. 02, no. 04, pp. 505–518, 2005.
- [6] L. Righetti, J. Buchli, M. Mistry, and S. Schaal, “Inverse dynamics control of floating-base robots with external constraints: A unified view,” in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1085–1090.
- [7] M. Mistry and L. Righetti, “Operational space control of constrained and underactuated systems,” in *Robotics*, ser. Robotics: Science and Systems. MIT Press Journals, 2012, pp. 225–232.
- [8] A. Del Prete, F. Nori, G. Metta, and L. Natale, “Prioritized motion–force control of constrained fully-actuated robots: “task space inverse dynamics”,” *Robotics and Autonomous Systems*, vol. 63, pp. 150–157, 2015.
- [9] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. D. Prete, “Optimization-based control for dynamic legged robots,” *IEEE Transactions on Robotics*, vol. 40, pp. 43–63, 2024.
- [10] M. Rijnen, E. de Mooij, S. Traversaro, F. Nori, N. van de Wouw, A. Saccon, and H. Nijmeijer, “Control of humanoid robot motions with impacts: Numerical experiments with reference spreading control,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4102–4107.

- [11] J. Van Steen, G. Van Den Brandt, N. van de Wouw, J. Kober, and A. Saccon, “Quadratic programming-based reference spreading control for dual-arm robotic manipulation with planned simultaneous impacts,” *IEEE Transactions on Robotics*, vol. 40, pp. 3341–3355, 2024.
- [12] Y. Wang, N. Dehio, A. Tanguy, and A. Kheddar, “Impact-aware task-space quadratic-programming control,” *The International Journal of Robotics Research*, vol. 42, no. 14, pp. 1265–1282, 2023.
- [13] I. Walker, “Impact configurations and measures for kinematically redundant and multiple armed robot systems,” *IEEE Transactions on Robotics and Automation*, vol. 10, no. 5, pp. 670–683, 1994.
- [14] G. Indukumar, A. Saccon, A. Franchi, and C. Gabellieri, “A study on impact-aware aerial robots colliding with the environment at non-vanishing speed,” in *2025 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2025, pp. 162–169.
- [15] M. Brunner, L. Giacomini, R. Siegart, and M. Tognon, “Energy tank-based policies for robust aerial physical interaction with moving objects,” in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2054–2060.
- [16] P. De Petris, M. Nissov, M. Harms, and K. Alexis, “Design, modeling, control and experimental evaluation of flexy: a novel compliant quadrator,” in *2024 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2024, pp. 31–38.
- [17] T. Bartelds, A. Capra, S. Hamaza, S. Stramigioli, and M. Fumagalli, “Compliant aerial manipulators: Toward a new generation of aerial robotic workers,” *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 477–483, 2016.
- [18] R. Featherstone, *Rigid Body Dynamics Algorithms*. Berlin, Heidelberg: Springer-Verlag, 2007.
- [19] M. Hamandi, F. Usai, Q. Sablé, N. Staub, M. Tognon, and A. Franchi, “Design of multirotor aerial vehicles: A taxonomy based on input allocation,” *The International Journal of Robotics Research*, vol. 40, no. 8–9, pp. 1015–1044, 2021.
- [20] D. E. Stewart and J. C. Trinkle, “An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction,” *International Journal for Numerical Methods in Engineering*, vol. 39, no. 15, pp. 2673–2691, 1996.
- [21] M. Ryll, D. Bicego, and A. Franchi, “A truly redundant aerial manipulator exploiting a multi-directional thrust base,” *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 138–143, 2018.
- [22] G. Nava, Q. Sablé, M. Tognon, D. Pucci, and A. Franchi, “Direct force feedback control and online multi-task optimization for aerial manipulators,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 331–338, 2020.
- [23] D. Brescianini and R. D’Andrea, “Tilt-prioritized quadcopter attitude control,” *IEEE Transactions on Control Systems Technology*, vol. 28, no. 2, pp. 376–387, 2018.
- [24] N. Hogan, “Impedance control: An approach to manipulation,” in *1984 American Control Conference*, 1984, pp. 304–313.
- [25] F. Romano, S. Traversaro, D. Pucci, and F. Nori, “A whole-body software abstraction layer for control design of free-floating mechanical systems,” *Journal of Software Engineering for Robotics*, 2017.
- [26] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi – A software framework for nonlinear optimization and optimal control,” *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.
- [27] H. Ferreau, C. Kirches, A. Potschka, H. Bock, and M. Diehl, “qpOASES: A parametric active-set algorithm for quadratic programming,” *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [28] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Souères, N. Mansard, F. Lamiraux, J.-P. Laumond, L. Marchionni, H. Tome, and F. Ferro, “Talos: A new humanoid research platform targeted for industrial applications,” in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 689–695.
- [29] F. Grimminger, A. Meduri, M. Khadiv, J. Viereck, M. Wüthrich, M. Naveau, V. Berenz, S. Heim, F. Widmaier, T. Flayols, J. Fiene, A. Badri-Spröwitz, and L. Righetti, “An open torque-controlled modular robot architecture for legged locomotion research,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3650–3657, 2020.
- [30] J. J. van Steen, N. van de Wouw, and A. Saccon, “Robot control for simultaneous impact tasks via quadratic programming-based reference spreading,” in *2022 American Control Conference (ACC)*, 2022, pp. 3865–3872.