



HAL
open science

Impact-Aware Task-Space Quadratic-Programming Control

Yuquan Wang, Niels Dehio, Arnaud Tanguy, Abderrahmane Kheddar

► **To cite this version:**

Yuquan Wang, Niels Dehio, Arnaud Tanguy, Abderrahmane Kheddar. Impact-Aware Task-Space Quadratic-Programming Control. 2020. hal-02741682v2

HAL Id: hal-02741682

<https://hal.science/hal-02741682v2>

Preprint submitted on 11 Nov 2020 (v2), last revised 6 Sep 2023 (v5)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Impact-Aware Task-Space Quadratic-Programming Control

Journal Title
XX(X):1-16
©The Author(s) 2020
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/

SAGE

Yuquan Wang, Niels Dehio, Arnaud Tanguy, and Abderrahmane Kheddar

Abstract

Generating on-purpose impacts with rigid robots is challenging as they may lead to severe hardware failures due to abrupt changes in the velocities and torques. Without dedicated hardware and controllers, robots typically operate at a near-zero velocity in the vicinity of contacts. We assume knowing how much of impact the hardware can absorb and focus solely on the controller aspects. The novelty of our approach is twofold: (i) it uses the task-space inverse dynamics formalism that we extend by seamlessly integrating impact tasks; (ii) it does not require separate models with switches or a reset map to operate the robot undergoing impact tasks. Our main idea lies in integrating post-impact states prediction and impact-aware inequality constraints as part of our existing general-purpose whole-body controller. To achieve such prediction, we formulate task-space impacts and its spreading along the kinematic tree of a floating-base robot with subsequent joint velocity and torque jumps. As a result, the feasible solution set accounts for various constraints due to expected impacts. In a multi-contact situation of under-actuated legged robots subject to multiple impacts, we also enforce standing stability margins. By design, our controller does not require precise knowledge of impact location and timing. We assessed our formalism with the humanoid robot HRP-4, generating maximum contact velocities, neither breaking established contacts nor damaging the hardware.

Keywords

Impact-awareness, on-purpose impact, task-space control, quadratic optimization control, humanoids.

1 Introduction

In robotics, instantaneous impacts occur when robots make contact – intentionally or accidentally – at relatively high velocities (cf. Figure 1). Impacts may cause high impulsive forces and velocity jumps at the contacting surfaces. Impact energy and jumps propagate through the robot’s links in the joints, which might cause severe mechanical damage. Abrupt changes in velocity or torque mean discontinuities that may destabilize closed-loop controllers. Floating-base robots can also witness sudden breaks of pre-existing unilateral contacts resulting in a sudden loss of balance.

When possible, a simple solution is to plan and control new contacts carefully with near-zero contact admissible velocities. Otherwise, new contacts are made with relatively high velocities, e.g., walking or jumping humanoids. In this case, the robot continuous dynamic equations of motions are not a good match for the induced physics, and another set of equations must be considered together with a transition policy (called reset maps). Such a transition policy requires knowledge of additional parameters that depend on the environment (and robot) stiffness, the impact model, the impact localization on the robot (and on the environment), the contact normal, the exact impact time, etc. Acquiring *in-situ*, instantaneously, and reliably these parameters is not always possible in practice.

In fact, a large part of handling properly robotic impacts must be tackled first from a hardware design viewpoint. It is the hardware design that reveals how much impact the robot can achieve without failures. Knowing the hardware capabilities in terms of impact resilience, allows to deal

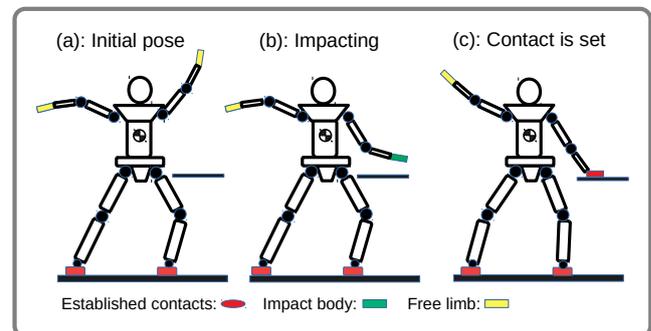


Figure 1. In this schematic sketch, a humanoid robot is impacting a wooden piece as we can commonly find in karate. In stance (a), there are $m_1 = 2$ established contacts (red), $m_2 = 0$ impacting end-effectors (green), and $m_3 = 2$ free limbs (yellow). Stance (b) is represented by $m_1 = 2$, $m_2 = 1$, $m_3 = 1$ and stance (c) by $m_1 = 3$, $m_2 = 0$, $m_3 = 1$.

with impact-aware motion planning and control. Usually, the impact duration is too small to react within the controller time-range (typically 1 to 5 ms, depending on the contact properties and the robot control architecture). Yet, the controller can be designed to regulate the pre-impact velocity safely.

Y. Wang, N. Dehio and A. Kheddar are with the CNRS-University of Montpellier, LIRMM, Interactive Digital Humans group, Montpellier, France. A. Tanguy and A. Kheddar are with the CNRS-AIST Joint Robotics Laboratory, IRL, Tsukuba, Japan. Y. Wang and A. Kheddar are in part with the Beijing Institute of Technology, Beijing, China.

Corresponding author:

Yuquan Wang Email: wyqsndddd@gmail.com

In this paper, we address general-purpose impact-aware tasks, including floating-base robots. Instead of explicitly designing a dedicated controller to handle task-aware impacts, we integrate impact and continuous dynamic models as well as constraints consistently as part of our whole-body multi-objective controller that relies on *quadratic program* (QP) solving (Bouyarmane et al. 2019).

The core idea is to perform a one-step-ahead prediction in every control cycle based on the impact model. That is to say, nearby intended or expected impact, we assume that it will happen in the next iteration. As a result, the controller becomes *aware* of impact-induced jumps and restricts the robot motion to meet impact with (possibly maximum) velocities that can handle the upcoming impact safely, i.e., within the pre-defined hardware resilience and task-dictated acceptable bounds. By our approach, the resulting robot motion is robust to uncertainties in impact time and location. This feature is the main novelty and the most appealing aspect of our approach w.r.t. state-of-the-art. In more details, our main contributions are:

C.1 Using floating-base multi-limb robots predicted impact effects, we gather all subsequent *state jumps* (Sec. 4.4) in the velocity space, the force and torque space, and the centroidal space.

C.2 Approximating post-impact states using Euler-forward method, we propose a *generic template constraint* to bound post-impact state and related constraints (Sec. 5.1).

C.3 Our framework applies also to concurrent impacts. Integrating impact constraints into the QP controller enables *multi-impact-aware whole-body motion generation* (Sec. 5.5), which is validated through experiments on our humanoid robot HRP-4 (Sec. 6).

To sum-up, our novel approach ensures the feasibility of the underlying task-space quadratic programming controller, accounting for hardware safety and feedback state abrupt jumps. It allows a built-in seamless interplay between continuous second-order and impact first order dynamics without branching and reset-maps. It is an alternative (and not a replacement or a competitor) to existing well-established hybrid and or reset map approaches.

2 Background

Analyses in Tsujita et al. (2008); Pashah et al. (2008) revealed that impact durations are of milliseconds order or less, even for low-velocities. Therefore, impact resilience is mitigated first through hardware design. As it is not the main concern of this paper, we report some works in Sec. 2.1.

For a given hardware, impact is then mitigated through the control design. Impact is subsequent to, or even part of, some tasks to be achieved (Sec. 2.2). For example, biped walking or running induces impact that must be treated in gait control with dynamic balance. Instead, kicking a ball is rather a task for which we would like to control the impacts of the feet with the ball together with dynamic balance too.

Our approach exploits on-purpose impact-tasks to increase robot manipulation efficiency (e.g., kicking the ball) and generates a broad spectrum of impulsive forces directly related to task performance (Sec. 2.3).

2.1 Mitigating impact through hardware

Technical knowledge on hardware resilience to impact shall tell how much impact a robot can bear without breaking links, joint mechanisms, embedded electronics, etc. For example, employing *variable stiffness actuator* (VSA) technologies lowers damage risks at impacts. However, it may need more than 10 ms to generate the joint torque that can counterbalance the torque jump (Haddadin et al. 2009). Despite numerous VSA paradigms that have been proposed – refer to the excellent review by Vanderborght et al. (2013) – their force control capabilities suffer from limited bandwidth as pointed in Wensing et al. (2017).

Aiming at reducing initial and post-impact forces, the *proprioceptive actuator* devised for the MIT Cheetah robot’s leg provides a mechanical approach to mitigate impacts without added compliance (Wensing et al. 2017). More recently, Singh and Featherstone (2020) proposed a novel quadruped robot leg design that cancels shock propagation from the floating-base. Another strategy absorbs the impacts at foot-strike through passive springs in the ankle (Reher et al. 2016). Designing smarter hardware specifically for high-impact resilience is challenging, and notable progress has been made.

An outer softcover approach was described in Battaglia et al. (2009), where the thickness of the cover is computed as a function of latency and impact velocity. Generally, shock-absorbing mechanisms or soft soles are added to the ankle/feet of humanoids. Yet further investigations are ongoing to increase the resilience of robots to impacts.

2.2 Handling task-induced impacts

Impacts are sometimes rather seen as perturbations. For example, Pagilla and Yu (2001) consider impact as a transient behavior and uses it for stability analysis rather than for explicit control design. Yet, Stewart (2000) reviewed that locomotion tasks have treated the impulse effects for two decades. The *complementarity dynamical system* (CDS) proposed by Hurmuzlu et al. (2004) offers a general conceptual tool to describe the dynamics of a walking robot, e.g., a biped with one or two contacts; see CDS controllability and stabilizability in Brogliato (2003). Impact-induced state jumps embedded in a reset map are applied to the robot states depending on complementarity condition. However, Grizzle et al. (2014) pointed out that there is no effective control design for CDS.

Rather than dealing with periodic impacts or cyclic behaviors, we are interested in the local feasibility properties before and right after a single impact event. Hence, modeling using Poincaré map (Grizzle et al. 2014), and the controllers developed on top of it do not apply straightforwardly to our aim. Comparing our approach with the impact dynamics model applied for walking (Hurmuzlu et al. 2004; Grizzle et al. 2014) or hybrid control approaches by Rijnen et al. (2017) is pointless. This is because we target general-purpose tasks implying any limb of the robot.

Alternatively, some control strategies simply avoid impacts when contacts are about to be created. Given the impact location, Pagilla and Yu (2001) modified the reference trajectories such that the reference velocity along

the surface normal is zero. The control design based on Zero-tilting Moment Point (ZMP) neglects impact and establishes contacts with close to zero velocity, see e.g. [Kajita et al. \(2010\)](#). [Grizzle et al. \(2014\)](#) summarized that the impact-less reference trajectories are challenging to generate and inefficient to execute. Therefore, in [DeMagistris et al. \(2017\)](#); [Pajon et al. \(2017\)](#) soft soles are considered.

By adding impact-awareness to our task-space control framework (C.3), our approach seamlessly executes the reference motion with any contact velocity as far as it conforms hardware impact-resilience limitations and allowable state jumps (C.2).

2.3 Handling on-purpose impact tasks

Modeling dynamic contact transition using *regularization* models, e.g., mass-spring-damper ([Hu et al. 2007](#); [Stanisic and Fernández 2012](#); [Heck et al. 2016](#)), enables impact-handling by well-established control tools from continuous-time dynamics.

Recently, for dealing with general-purpose impact behaviors with explicit state jumps, once impact is detected [Rijnen et al. \(2017\)](#) proposed a switching controller from pre- to post-impact, based on pre-computed reference spreading. [Konno et al. \(2011\)](#) solved a three-phase non-linear optimization problem to generate reference trajectories as well as the posture at the impact moment to maximize the force jump. These methods require off-line reference trajectory planning and switching, which compromises reactivity. It further conflicts with the concept of implicit trajectory generation in multi-objective task-space control based on real-time sensory feedback.

When reference switching is performed upon impact detection; the latter is always delayed and may be inaccurate. Thus, due to unknown exact impact contact location and timing ([Pagilla and Yu 2001](#)), switching controllers would apply set-points or references. These are pre-defined in the pre-impact mode to be applied during the post-impact mode for a specific time interval, no matter how small it is.

In the contrary, nearby desired impact, our controller triggers impact-aware tasks and constraints. Then it starts assuming the impact to happen in the up-coming time-step iteration. Our impact-aware QP controller (C.3) provides the mapping between the impact-induced state jumps and the pre-impact velocities in each iteration (C.1). This mapping enables the QP controller to regulate the robot motion in real-time such that the impact-aware constraints are fulfilled (C.2). We can find a similar mapping provided in [Partridge and Spong \(2000\)](#) to control the trajectory of a three-link planar robot. Yet our proposed controller does not require off-line trajectory computation and, therefore, robust to impact timing and location. The prediction of post-impact states that we propose (C.1) is based on the impact dynamics model along with the contact normal direction by [Zheng and Hemami \(1985\)](#) without explicitly considering tangential impulse and impact-induced friction. A similar impact dynamics model is introduced by [Hurmuzlu et al. \(2004\)](#) to address bipedal walking [Bergés and Bowling \(2005\)](#); [Grizzle et al. \(2014\)](#); [Wieber et al. \(2016\)](#).

In space robotics, [Nenchev \(2013\)](#) applied joint space momentum conservation to study post-impact robot motion behavior. We do not consider zero-gravity and rather use

the centroidal momentum model proposed in ([Orin et al. 2013](#)). Yet, [Nava et al. \(2016\)](#) pointed out that state-of-the-art momentum controllers may lead to unstable zero dynamics. In [Nenchev \(2018\)](#) a decoupled controller based on relative momentum is proposed. Instead, we only limit the impact-induced centroidal momentum jumps (C.2).

Recently, [Jia et al. \(2019\)](#) studied flying object batting, where a closed-form 2D impact dynamics model is applied. It generates the desired impulse by enumerating five possibilities. In the 3D cases, the closed-form solution is only available if we can control the initial sliding direction to an invariant subset ([Jia and Wang 2017](#)). Note that according to the energy conservation principle, it is not possible to treat impact dynamics along with the two tangential directions and normal direction separately. One caveat of this issue is mathematical modeling and computational complexity.

The Zero-tilting Moment Point (ZMP) is widely used as a balance criterion for biped walking on the planar grounds, see ([Vukobratović and Borovac 2004](#)). Recently, the concept is extended to the multi-contact case in [Caron et al. \(2017\)](#). For tasks that require large impulsive forces, e.g., a nailing task discussed in [Tsuji et al. \(2008\)](#) and a wooden piece breaking task performed in [Konno et al. \(2011\)](#), ZMP is used to analyze the dynamic balance of each robot configuration instance. Introducing impact-aware constraints (C.2) to bound the impact-induced state jumps (C.1) including ZMP, Center-of-Mass (COM) velocity, and the centroidal momentum, our proposed method allows more reliable and robust motion generation.

To our best knowledge, intentionally generating high impacts tasks with humanoids is studied only in very few works and for specific scenarios, e.g., [Konno et al. \(2011\)](#), with a focus rather on planning. We aim to extend state-of-the-art task-space multi-objective and multi-sensory whole-body controllers formulated as a quadratic program (QP) to encompass impact tasks (C.3). We adopt and further build on our initial concept proposed in simulation for fixed-base robots in [Wang and Kheddar \(2019\)](#) and the preliminary extension to the floating-base robot in [Wang et al. \(2019\)](#). Concerning these and previously cited contributions, this article provides (i) a deeper conceptualization of our approach with detailed discussion on multi-impact cases; (ii) additional impact-aware constraints (C.1) framed in a generic impact-aware template (C.2); (iii) improved impact effect estimation and richer experiments.

3 Impact-unaware Constraint-based QP-Control

More than a decade ago, control formulations based on a quadratic program (QP) has been proposed for redundant manipulators, see e.g. [Zhang et al. \(2004\)](#). It was also proposed in computer graphics animation, see e.g. ([Abe et al. 2007](#); [de Lasa and Hertzmann 2009](#)). In robotics, ([Kanoun et al. 2009](#); [Decré et al. 2009](#)) QP formulations integrated inequality constraints. It has become the predominant scheme for operating redundant robots (e.g. humanoids) ([Righetti et al. 2013](#); [Kuindersma et al. 2016](#); [Liu et al. 2016](#); [Bouyarmane and Kheddar 2018](#)) to cite a few. The approach allows pursuing multiple prioritized objectives

simultaneously while satisfying strict bounds. It has been applied both to position- and torque-controlled robots.

In this paper, we extend to impact-awareness tasks our controller detailed in [Bouyarmane et al. \(2019\)](#). In this section, we recall the main constraints used in continuous second order dynamics with our notations. All variables are referring to the current time step, during which the QP decision (i.e., control) variables are computed. We introduce kinematics and dynamics constraints in [Sec. 3.1](#); joint space, contact space, and centroidal space constraints in [Secs. 3.2, 3.3, and 3.4](#), respectively. We summarize the QP controller without considering the impacts in [Sec. 3.5](#), and highlight impact-related problems in [Sec. 3.6](#).

3.1 Robot model

Consider a kinematic tree structure with $m = m_1 + m_2 + m_3$ independent end-effectors such that:

1. m_1 end-effectors are established contacts;
2. m_2 end-effectors for which we expect an upcoming impact (or dynamic contact transition), and
3. m_3 free limbs whose momentum jump can mitigate an external impulse propagating from other end-effectors.

We denote these three sets of end-effectors with σ_{m_1} , σ_{m_2} , and σ_{m_3} respectively. [Figure 1](#) illustrates an exemplary impact task with varying m_1, m_2, m_3 over time.

For legged robots, we can represent the floating-base, which holds a configuration in $SE(3)$, by six virtual joints. Thus the generalized joint position writes $\mathbf{q} \in \mathbb{R}^{(n+6)}$. A floating-base robot with m_1 established contacts is governed with the following equation of motion:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}\boldsymbol{\tau} + \mathbf{J}_{\sigma_{m_1}}^\top \mathbf{f}_{\sigma_{m_1}} \quad (1)$$

where $\mathbf{B} \in \mathbb{R}^{(n+6) \times n}$ is the actuated joints selection matrix; $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{(n+6) \times (n+6)}$ is the inertia matrix; the vector $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{(n+6)}$ gathers both Coriolis and gravitational effects; vector $\boldsymbol{\tau} \in \mathbb{R}^n$ is the actuated joint torques. We collect the stacked Jacobian $\mathbf{J}_{\sigma_{m_1}}$:

$$\mathbf{J}_{\sigma_{m_1}} = [\mathbf{J}_1^\top, \dots, \mathbf{J}_{m_1}^\top]^\top \in \mathbb{R}^{3m_1 \times (n+6)}.$$

Remark 3.1. *The m_2 end-efforts are not part of the second-order dynamics; hence they are not accounted in balance prior to impact.*

3.2 Joint space constraints

Given the torque limits $\boldsymbol{\tau} \leq \boldsymbol{\tau}_k \leq \bar{\boldsymbol{\tau}}$, the joint space dynamics (1) results in

$$\begin{bmatrix} \mathbf{I} \\ -\mathbf{I} \end{bmatrix} \mathbf{M}\ddot{\mathbf{q}}_k \leq \begin{bmatrix} \mathbf{B}\bar{\boldsymbol{\tau}} \\ -\mathbf{B}\boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ -\mathbf{I} \end{bmatrix} (\mathbf{J}^\top \mathbf{f}_{k-1} - \mathbf{N}_{k-1}). \quad (2)$$

Other joint constraints, such as position and velocity limits

$$\begin{aligned} \underline{\mathbf{q}} &\leq \mathbf{q}_k \leq \bar{\mathbf{q}} \\ \underline{\dot{\mathbf{q}}} &\leq \dot{\mathbf{q}}_k \leq \bar{\dot{\mathbf{q}}}, \end{aligned}$$

and others such as collision avoidance, constraints in the Cartesian space... do not express directly in the robot state

acceleration. By approximating the derivative using Euler backward method, they can be written as a function of the decision variable $\ddot{\mathbf{q}}_k$ and the measured joint positions \mathbf{q}_{k-1} and velocities $\dot{\mathbf{q}}_{k-1}$; and implementation in closed-loop as described in [Djeha et al. \(2020\)](#).

3.3 Contact space constraints

For rigid contacts, we set zero relative acceleration as

$$\mathbf{J}_i \ddot{\mathbf{q}}_k + \dot{\mathbf{J}}_i \dot{\mathbf{q}}_k = \mathbf{0}, \quad (3)$$

and non-sling of any i -th contact wrench $\mathbf{W}_i = [\mathbf{f}^\top, \boldsymbol{\tau}^\top]^\top$, expressed at the origin of its local contact frame, by

$$\|\mathbf{f}_t\| < \mu \mathbf{f}_n \text{ and } \mathbf{f}_n > 0,$$

where μ denotes the friction coefficient, $\mathbf{f}_t, \mathbf{f}_n$ denote the tangential and normal forces. If needed, the center of pressure (CoP) is limited within a rectangular contact area $\{(x, y) : x \in [-X, X], y \in [-Y, Y]\}$, i.e., fulfilling:

$$|\tau_x| \leq Y \mathbf{f}_z, |\tau_y| \leq X \mathbf{f}_z \text{ and } \tau_z \in [\tau_{z,\min}, \tau_{z,\max}].$$

We can re-formulate the above inequalities using a half-plane representation:

$$\mathcal{C}_i \mathbf{W}_i \leq 0. \quad (4)$$

The details of \mathcal{C}_i are given in [Caron et al. \(2015\)](#) or similarly in [\(Grizzle et al. 2014, Sec. 3.3\)](#). As long as (3) and (4) hold, the contact is set without slipping or taking-off.

3.4 Centroidal space constraints

In the multi-contact setting, we define the Zero-tilting Moment Point (ZMP) location* as $\mathbf{z} \in \mathbb{R}^3$:

$$\mathbf{z} = \frac{\mathbf{n} \times \boldsymbol{\tau}_O}{\mathbf{n} \cdot \mathbf{f}_O} = \frac{\hat{\mathbf{n}} \boldsymbol{\tau}_O}{\mathbf{n}^\top \mathbf{f}_O}, \quad (5)$$

where the $\hat{\cdot}$ operator defines the skew-symmetric matrix. The wrench $\mathbf{W}_O = [\mathbf{f}_O^\top, \boldsymbol{\tau}_O^\top]^\top$ denotes the net contact wrench measured in the inertial frame \mathcal{F}_O whose origin is typically the center between the feet contacts ([Caron et al. 2017](#)). The ZMP is restricted by its feasible area $\mathbf{z} \in \mathcal{S}_s$ or $\mathbf{z} \in \mathcal{S}_z$. In the simple case with coplanar contacts (for example double support stance on flat ground) the bounding polygon is defined by the support polygon \mathcal{S}_s , i.e., the convex hull of the ground contact points. [Caron et al. \(2017\)](#) derived a feasible multi-contact ZMP area \mathcal{S}_z for the more general case. Introducing A_z and \mathbf{a}_z , the convex half-plane representation of \mathcal{S}_z becomes:

$$A_z \mathbf{z}_k \leq \mathbf{a}_z.$$

As pointed by [Sugihara \(2009\)](#), the ZMP is a measure of the force. Thus given the ZMP definition (5), we rearrange the above constraint with respect to the external wrench:

$$\mathcal{G}_z \mathbf{W}_O \leq 0. \quad (6)$$

Using the unit vector $[0, 0, 1]^\top$ (the reverse direction of the gravity) as the surface normal $\mathbf{n} \in \mathbb{R}^3$, \mathcal{G}_z writes as: $\mathcal{G}_z = [-\mathbf{a}_z \mathbf{n}^\top \quad A_z \hat{\mathbf{n}}]$.

*According to [Caron et al. \(2017\)](#), we can project the ZMP on surfaces with different heights, thus we employ the generic definition $\mathbf{z} \in \mathbb{R}^3$. In this paper, we choose the ground surface, i.e., $z_z = 0$.

When the robot is subject to external forces, Sugihara (2009) showed that the horizontal COM velocity $\dot{c}_{x,y}$ has to remain inside a convex 2D polygon $\mathcal{S}_{\dot{c}}$ in order to ensure the dynamic balance. This area $\mathcal{S}_{\dot{c}}$ relates to both the feasible ZMP area \mathcal{S}_z and the COM position $c_{x,y}$. The constraint $\dot{c}_{x,y} \in \mathcal{S}_{\dot{c}}$ is transformed into the half-plane representation, introducing $\mathcal{G}_{\dot{c}_{x,y}}$ and $h_{\dot{c}_{x,y}}$:

$$\mathcal{G}_{\dot{c}_{x,y}} \dot{c}_{x,y} \leq h_{\dot{c}_{x,y}}. \quad (7)$$

Due to the kinematic and actuation limits, the robot controller should minimize the angular momentum $\mathcal{L}_c \in \mathbb{R}^3$ (Lee and Goswami 2012; Wiedebach et al. 2016). Suppose the angular momentum is bounded by $\mathcal{L}_c \leq \bar{\mathcal{L}}_c$, we take the angular part $A_{\omega c}(\mathbf{q}_k)$ from the centroidal momentum matrix $A_c(\mathbf{q}_k)$ proposed by Orin et al. (2013) to formulate:

$$A_{\omega c}(\mathbf{q}_k) \dot{\mathbf{q}} \leq \bar{\mathcal{L}}_c. \quad (8)$$

3.5 Impact-unaware whole-body QP controller

The whole-body controller is formulated as a quadratic program (QP) with desired task objectives and strict constraints (Bouyarmine et al. 2019). Prioritized tasks shall be met at best according to the associated cost function. Constraints shall be met strictly.

$$\min_{\nu} \sum_{i \in \mathcal{I}_o} w_i \|e_i(\nu)\|^2$$

s.t. **Joint space constraints:**

Joint position, velocity, and torque (2),

Contact space constraints:

Holding contact position: (3),

Fulfilling contact wrench cone: (4),

Centroidal space constraints:

ZMP constraint: (6),

COM velocity constraint: (7),

Angular momentum constraint: (8),

non-cited constraints, e.g., collision avoidance...

where $\sum_{i \in \mathcal{I}_o} w_i \|e_i(\nu)\|^2$ scalarizes multiple task objectives included in a set \mathcal{I}_o , w_i weights the i -th task and $e_i(\nu)$ denotes the task error. $e_i(\nu)$ is linear in terms of the QP decision variables ν , see the details in Bouyarmine et al. (2019). Both the generalized joint accelerations $\ddot{\mathbf{q}}_k$ and the generating vectors of the contact wrench cone $\mathbf{f}_\lambda(k)$ are optimized, hence $\nu_k := \{\ddot{\mathbf{q}}_k, \mathbf{f}_\lambda(k)\}$.

3.6 Problem description

Impacts result in instantaneous jumps in the contacting end-effector's velocity, which affects many constraints expressed in terms of joint velocity such as robot state or kinematic bounds, ZMP, angular momentum, COM velocity, etc. As a result, the QP feasible search space, defined by such constraints, may suddenly shrink to an empty set. In this case, the QP is infeasible for the next control iteration. An illustrative toy example is given later in Sec. 5.6. In the other hand, the controller must embed constraints related to hardware impact resilience.

In order to deal with impact-induced state jumps, the main idea is to adjust the pre-impact contact velocity. Here, adjusting means that we shall find a compromise between what is possible to be achieved in terms high-velocities (ideally, the maximum possible for a desired high impact) and the QP controller state-jump and hardware feasibility set. In other words, we propose additional constraints that allow the controller to steer the robot to generate impacts that ensure a feasible closed-loop QP right after the impact.

In the following, we explicit how to predict impact-induced state jumps (based on the assumption of rigid contact surfaces) in Sec. 4 and then explain how to formulate impact-aware constraints in Sec. 5.

4 Predicting impact-induced state jumps

The prediction driven from control goals amounts to the least squares solution of two sets of equalities: (i) the contact velocity restitution, (ii) the impact dynamics.

In Sec. 4.1, we apply the velocity restitution model proposed by Zheng and Hemami (1985) used in Hurmuzlu and Marghitu (1994); Konno et al. (2011). The nominal impact dynamics writes:

$$\mathbf{M} \Delta \dot{\mathbf{q}} = \mathbf{J}^\top \boldsymbol{\nu}, \quad (10)$$

also used in Grizzle et al. (2001); Addi and Rodić (2010); Konno et al. (2011); Grizzle et al. (2014); Featherstone (2014); Wang and Kheddar (2019). In a multi-contact setting, we shall account for two issues:

1. The coupling between different end-effectors using the operational-space extension of (10), see Featherstone (2014); Chang and Khatib (2000).
2. For humanoids we might also use *centroidal momentum* constraint (18) based on the centroidal momentum matrix (CMM) Orin et al. (2013) and applied as a basis for computing the balanced motion in Kuindersma et al. (2016); Wiedebach et al. (2016). In Wensing and Orin (2016) the computation of the CMM from the joint space is given.

We summarize the prediction as a least squares solution in Sec. 4.3. In order to exploit the prediction for impact-tasks, we specify important impact-induced state jumps as functions of the QP decision variables in Sec. 4.4.

4.1 Contact velocity restitution

Let us consider one of the m_2 robot end-effectors undergoing a desired impact task. The scalar coefficient of restitution $c_r > 0$ represents the relationship between the pre-impact end-effector velocity $\dot{\mathbf{x}}^- \in \mathbb{R}^3$ and the post-impact velocity $\dot{\mathbf{x}}^+ \in \mathbb{R}^3$ assuming no sliding

$$\dot{\mathbf{x}}_{k+1}^+ = -c_r P_n \dot{\mathbf{x}}_{k+1}^- + (I - P_n) \dot{\mathbf{x}}_{k+1}^-, \quad (11)$$

where $P_n = \mathbf{n}\mathbf{n}^\top \in \mathbb{R}^{3 \times 3}$ is the projection operator onto the contact normal $\mathbf{n} \in \mathbb{R}^3$. Note that angular velocity terms are neglected. In this model, the pre-impact velocity is not necessarily aligned with the contact normal. The resulting

three-dimensional end-effector velocity jump becomes[†]

$$\Delta \dot{\mathbf{x}}_{k+1} = \dot{\mathbf{x}}_{k+1}^+ - \dot{\mathbf{x}}_{k+1}^- = \underbrace{-(1+c_r)P_n}_{P_\Delta} \dot{\mathbf{x}}_{k+1}^-. \quad (12)$$

We approximate the next iteration pre-impact velocity as:

$$\begin{aligned} \dot{\mathbf{x}}_{k+1}^- &= \dot{\mathbf{x}}_{k+1} = J_{k+1} \dot{\mathbf{q}}_{k+1} \\ &= (J_k + \dot{J}_k \Delta t + \ddot{J}_k \Delta t^2 + \dots)(\dot{\mathbf{q}}_k + \ddot{\mathbf{q}}_k \Delta t) \\ &= \underbrace{J_k \dot{\mathbf{q}}_k}_{\dot{\mathbf{x}}_k} + \underbrace{J_k \ddot{\mathbf{q}}_k \Delta t + \dot{J}_k \dot{\mathbf{q}}_k \Delta t}_{\dot{\mathbf{x}}_k \Delta t} + \underbrace{\ddot{J}_k \dot{\mathbf{q}}_k \Delta t^2 + \dots}_{\approx 0} \end{aligned}$$

Neglecting all terms in Δt^n where $n > 1$ since $\Delta t \leq 5$ ms, we expand (12) to express the jump $\Delta \dot{\mathbf{x}}_{k+1}$ with respect to the QP decision variable $\dot{\mathbf{q}}_k$:

$$\Delta \dot{\mathbf{x}}_{k+1} = P_\Delta (J_k \dot{\mathbf{q}}_k + J_k \ddot{\mathbf{q}}_k \Delta t + \dot{J}_k \dot{\mathbf{q}}_k \Delta t),$$

where $\dot{\mathbf{q}}_k$ is obtained from the current robot state, and Δt denotes the sampling period. Neglecting the time step notation and define the block diagonal projection matrix $P_{\sigma_{m_2}} \in \mathbb{R}^{3m_2 \times 3m_2}$, we gather the velocity jumps at the next iteration of the m_2 impacting end-effectors as:

$$\Delta \dot{\mathbf{x}}_{\sigma_{m_2}} = P_{\sigma_{m_2}} (J_{\sigma_{m_2}} \dot{\mathbf{q}} + J_{\sigma_{m_2}} \ddot{\mathbf{q}} \Delta t + \dot{J}_{\sigma_{m_2}} \dot{\mathbf{q}} \Delta t) \quad (13)$$

4.2 Impact dynamics

An impact generate a contact force. The latter is integrated within the short time-range δt , altogether with the second order dynamics, and results in an impulse $\boldsymbol{\iota}$, such that:

$$\Delta \mathbf{f} = \frac{1}{\delta t} \boldsymbol{\iota}, \quad (14)$$

where $\Delta \mathbf{f}$ is the contact force jumps as in (Konno et al. 2011)-(eq. 23) and in (Grizzle et al. 2014)-(eq. 45).

End-effectors that would experience state jumps have their Jacobians stacked in $\mathbf{J}_{\sigma_m} \in \mathbb{R}^{3m \times (n+6)}$. Considering the robot's joint space equations of motion (1) with the generalized force $B\boldsymbol{\tau} = \mathbf{J}_{\sigma_m}^\top \mathbf{f}_{\sigma_m}$, we obtain

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} \mathbf{J}_{\sigma_m}^\top \mathbf{f}_{\sigma_m} + \boldsymbol{\alpha}, \quad (15)$$

where $\boldsymbol{\alpha}$ denotes the *bias force* (i.e., free acceleration) that vanishes after integrating (15) over the impact duration δt ,

$$\Delta \dot{\mathbf{q}} = \mathbf{M}^{-1} \mathbf{J}_{\sigma_m}^\top \boldsymbol{\iota}_{\sigma_m}.$$

Left-multiplying the above equation by \mathbf{J}_{σ_m} , we map this relationship onto the end-effectors operational space:

$$\Delta \dot{\mathbf{x}}_{\sigma_m} = \Upsilon \boldsymbol{\iota}_{\sigma_m},$$

where $\Upsilon = \mathbf{J}_{\sigma_m} \mathbf{M}^{-1} \mathbf{J}_{\sigma_m}^\top$. Note that Υ characterizes the impulse propagation among the m end-effectors. Its expansion includes three blocks:

$$\begin{bmatrix} \Delta \dot{\mathbf{x}}_{\sigma_{m_1}} \\ \Delta \dot{\mathbf{x}}_{\sigma_{m_2}} \\ \Delta \dot{\mathbf{x}}_{\sigma_{m_3}} \end{bmatrix} = \underbrace{\begin{bmatrix} \Upsilon_{\sigma_{m_1} \sigma_{m_1}}, & \Upsilon_{\sigma_{m_1} \sigma_{m_2}}, & \Upsilon_{\sigma_{m_1} \sigma_{m_3}} \\ \Upsilon_{\sigma_{m_2} \sigma_{m_1}}, & \Upsilon_{\sigma_{m_2} \sigma_{m_2}}, & \Upsilon_{\sigma_{m_2} \sigma_{m_3}} \\ \Upsilon_{\sigma_{m_3} \sigma_{m_1}}, & \Upsilon_{\sigma_{m_3} \sigma_{m_2}}, & \Upsilon_{\sigma_{m_3} \sigma_{m_3}} \end{bmatrix}}_{\Omega} \begin{bmatrix} \boldsymbol{\iota}_{\sigma_{m_1}} \\ \boldsymbol{\iota}_{\sigma_{m_2}} \\ \mathbf{0} \end{bmatrix} \quad (16)$$

where a block $\Upsilon_{\sigma_i \sigma_j} \in \mathbb{R}^{3m_i \times 3m_j}$ gathers matrices $\mathbf{J}_k \mathbf{M}^{-1} \mathbf{J}_l$ of the articulated bodies inertia matrices when

$k = l$ and the cross-coupling inertia when $k \neq l$. As $\boldsymbol{\iota}_{\sigma_{m_3}} = \mathbf{0}$, we only need the block Ω from Υ .

As in (13), $\Delta \dot{\mathbf{x}}_{\sigma_{m_1}} \in \mathbb{R}^{3m_1}$ gathers the provisional velocity jumps of the m_1 already existing contacts which will have reaction impulse forces $\boldsymbol{\iota}_{\sigma_{m_1}} \in \mathbb{R}^{3m_1}$, $\Delta \dot{\mathbf{x}}_{\sigma_{m_3}} \in \mathbb{R}^{3m_3}$ denotes the free end-effectors (i.e., no external force or impulse is acting on them), for which we aim to monitor the velocity jumps. Recall that $m = m_1 + m_2 + m_3$.

Multiple solutions for joint velocity jumps $\Delta \dot{\mathbf{q}}$ can fulfill $\Delta \dot{\mathbf{x}}_{\sigma_m} = \mathbf{J}_{\sigma_m} \Delta \dot{\mathbf{q}}$ in (16). Yet, each $\dot{\mathbf{q}}$ determines a unique centroidal momenta $\mathbf{h}_c \in \mathbb{R}^6$. Thus we introduce the centroidal space impact dynamics to specify the centroidal momenta jump $\Delta \mathbf{h}_c$ uniquely.

Assuming m_1 established contacts and m_2 impacts, we define the combined set: $\sigma_e = \sigma_{m_1} \cup \sigma_{m_2}$, the aggregation of external forces determines the centroidal momentum derivative:

$$\dot{\mathbf{h}}_c = A_c \dot{\mathbf{q}} = \begin{bmatrix} M \mathbf{g} \\ 0 \end{bmatrix} + J_{cp_{\sigma_e}}^\top \mathbf{f}_{\sigma_e}, \quad (17)$$

where M is the total mass of the robot, A_c is the centroidal momentum matrix (CMM), and $\mathbf{p}_i \in \mathbb{R}^3$ denotes the i -th contact location (the CoP position).

The centroidal frame external force $\mathbf{f}_i = R_{cp_i} \mathbf{f}_i^m \in \mathbb{R}^3$ for $i \in \sigma_e$ is calculated from the measured force \mathbf{f}_i^m . Each $J_{cp_i}^\top \in \mathbb{R}^{6 \times 3}$ of the stacked Jacobians $J_{cp_{\sigma_e}}$, that calculates the resultant wrench in the centroidal frame \mathcal{F}_c , is given by: $J_{cp_i}^\top := [I^\top, \overrightarrow{cp_i}^\top \times^\top]^\top$.

Integrating (17) over the impact duration δt , the centroidal momentum jump equals to the sum of the external impulses:

$$\Delta \mathbf{h}_c = A_c \Delta \dot{\mathbf{q}} = J_{cp_{\sigma_e}}^\top \boldsymbol{\iota}_{\sigma_e} \quad (18)$$

with the constant term $M \mathbf{g} \delta t$ being negligible.

4.3 Impulse distribution quadratic program

To solve the post-impact joint velocity jumps and impulses[‡], we define them as optimization variable:

$$\mathbf{u} = [\Delta \dot{\mathbf{q}}^\top, \boldsymbol{\iota}_{\sigma_{m_1}}^\top, \boldsymbol{\iota}_{\sigma_{m_2}}^\top]^\top = [\Delta \dot{\mathbf{q}}^\top, \boldsymbol{\iota}_{\sigma_e}^\top]^\top.$$

We would like the robot to apply multiple impacts along with given directions in the next control iteration that we would like to steer with the following set of equalities:

Impact dynamics (16) and (18):

$$\begin{aligned} J_{\sigma_m} \Delta \dot{\mathbf{q}} &= \Omega \boldsymbol{\iota}_{\sigma_e}, \\ A_c \Delta \dot{\mathbf{q}} &= J_{cp_{\sigma_e}} \boldsymbol{\iota}_{\sigma_e}, \end{aligned} \quad (19)$$

Contact velocity restitution (13):

$$J_{\sigma_{m_2}} \Delta \dot{\mathbf{q}} = \Delta \dot{\mathbf{x}}_{\sigma_{m_2}}.$$

We reformulate (19) as a quadratic program

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2} \mathbf{u}^\top \mathbf{u} \\ \text{s.t.} \quad & B \mathbf{u} = \mathbf{b} \end{aligned} \quad (20)$$

[†]The model (11) can be replaced by an experimental one.

[‡]All the impulses are defined in the inertial frame. The centroidal frame keeps the same orientation of the inertial frame. Thus we can use $\boldsymbol{\iota}$ for both (16) and (18).

with block matrices

$$B = \begin{bmatrix} J_{\sigma_m}, & -\Omega \\ A_c, & -J_{cp\sigma_e} \\ J_{\sigma_{m_2}}, & 0 \end{bmatrix}, \text{ and } \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \Delta \dot{\mathbf{x}}_{\sigma_{m_2}} \end{bmatrix}.$$

The least-squares solution of (20), i.e., the joint velocity jump $\Delta \dot{\mathbf{q}}^*$ and the impulse $\iota_{\sigma_e}^*$ at the time step t_{k+1} , can be obtained analytically:

$$\mathbf{u}^* = \begin{bmatrix} \Delta \dot{\mathbf{q}}^* \\ \iota_{\sigma_e}^* \end{bmatrix} = B^\# \mathbf{b} = \begin{bmatrix} K_{\Delta \dot{\mathbf{q}}} \\ K_L \end{bmatrix} \Delta \dot{\mathbf{x}}_{\sigma_{m_2}}, \quad (21)$$

where $B^\#$ is the pseudo-inverse of B :

$$B^\# = B^\top (BB^\top)^{-1} = \begin{bmatrix} \cdots, & K_{\Delta \dot{\mathbf{q}}} \\ \cdots, & K_L \end{bmatrix}.$$

Only the right $3m_2$ columns (related to the $3m_2$ lower, nonzero elements of \mathbf{b}) of $B^\#$ are of interest. Note that we can decompose $K_{\Delta \dot{\mathbf{q}}} \in \mathbb{R}^{(n+6) \times 3m_2}$, $K_L \in \mathbb{R}^{3(m_1+m_2) \times 3m_2}$ into individual impacting end-effector contributions:

$$K_{\Delta \dot{\mathbf{q}}} = [K_{\Delta \dot{\mathbf{q}}_1}, K_{\Delta \dot{\mathbf{q}}_2}, \dots, K_{\Delta \dot{\mathbf{q}}_{m_2}}],$$

$$K_L = \begin{bmatrix} K_{1,1}, & \cdots, & K_{1,m_2} \\ \vdots & & \vdots \\ K_{(m_1+m_2),1}, & \cdots, & K_{(m_1+m_2),m_2} \end{bmatrix}.$$

Remark 4.1. The QP (20) is an estimator that does not restrict any contact to a fixed position. Thus we can observe $\Delta \dot{\mathbf{x}}_i \neq \mathbf{0}$ for $i = 1, \dots, m_1$ regardless of whether an end-effector has established contact or not.

To sustain established contacts, we construct the contact wrench cone constraint (36) (which is the impact-aware version of (4) to be introduced in Sec. 5.3) using the predicted $\iota_{\sigma_{m_1}}^*$. Depending on the feasibility of (36), the QP controller can reduce the contact velocity and hence also the propagated impulse $\iota_{\sigma_{m_1}}^*$.

4.4 Whole-body impact effects

Here, we summarize usual state jumps in a generic form. The quantity jump $\Delta \lambda_{k+1}$ relates to joint accelerations $\ddot{\mathbf{q}}_k$ and is linearly decomposed as

$$\Delta \lambda_{k+1} = \mathcal{J}_{\Delta \lambda} \ddot{\mathbf{q}}_k \Delta t + \mathcal{C}_{\Delta \lambda} \dot{\mathbf{q}}_k, \quad (22)$$

where the matrices $\mathcal{J}_{\Delta \lambda}$ and $\mathcal{C}_{\Delta \lambda}$ reflects the impact model and the impulse propagation through the kinematic tree. The coefficient of restitution contributes to $\mathcal{J}_{\Delta \lambda}$ and $\mathcal{C}_{\Delta \lambda}$ as a scalar multiplier by $(1 + c_r)$. Hence, higher coefficients lead to higher jumps. We specify $\mathcal{J}_{\Delta \lambda}$ and $\mathcal{C}_{\Delta \lambda}$ on a case-by-case basis in the rest of this section. These expressions are used in Sec. 5 to formulate impact-aware constraints.

Joint position jumps We assume as in Zheng and Hemami (1985); Konno et al. (2011); Wang and Kheddar (2019), that there are no joint position jumps during impact $\Delta \mathbf{q}_{k+1} = \mathbf{0}$.

Joint velocity jumps given in (21) are reformulated in terms of joint accelerations by substituting (13)

$$\begin{aligned} \Delta \dot{\mathbf{q}}_{k+1} &= K_{\Delta \dot{\mathbf{q}}} \Delta \dot{\mathbf{x}}_{\sigma_{m_2}} \\ &= \underbrace{K_{\Delta \dot{\mathbf{q}}} P_{\sigma_{m_2}} J_{\sigma_{m_2}}}_{\mathcal{J}_{\Delta \dot{\mathbf{q}}}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{K_{\Delta \dot{\mathbf{q}}} P_{\sigma_{m_2}} (J_{\sigma_{m_2}} + \dot{J}_{\sigma_{m_2}} \Delta t)}_{\mathcal{C}_{\Delta \dot{\mathbf{q}}}} \dot{\mathbf{q}}_k. \end{aligned} \quad (23)$$

End-effector force jumps Substituting (13) in the predicted impulse (21), we can predict the force jump defined by (14), in terms of joint accelerations:

$$\begin{aligned} \Delta \mathbf{f}_{\sigma_e, k+1} &= \frac{1}{\delta t} K_L \Delta \dot{\mathbf{x}}_{\sigma_{m_2}} \\ &= \underbrace{\frac{1}{\delta t} K_L P_{\sigma_{m_2}} J_{\sigma_{m_2}}}_{\mathcal{J}_{\Delta \mathbf{f}}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{\frac{1}{\delta t} K_L P_{\sigma_{m_2}} (J_{\sigma_{m_2}} + \dot{J}_{\sigma_{m_2}} \Delta t)}_{\mathcal{C}_{\Delta \mathbf{f}}} \dot{\mathbf{q}}_k. \end{aligned} \quad (24)$$

Impulsive joint torque jumps are uniquely defined by the end-effector force jumps through

$$\Delta \boldsymbol{\gamma} = J_{\sigma_e}^\top \Delta \mathbf{f}_{\sigma_e},$$

where we used the symbol $\Delta \boldsymbol{\gamma}$ to distinguish from the joint motor torque $\boldsymbol{\tau}$ in (2). Hence in view of (24), we predict the impulsive joint torque jumps as:

$$\Delta \boldsymbol{\gamma}_{k+1} = \underbrace{J_{\sigma_e}^\top \mathcal{J}_{\Delta \mathbf{f}}}_{\mathcal{J}_{\Delta \boldsymbol{\gamma}}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{J_{\sigma_e}^\top \mathcal{C}_{\Delta \mathbf{f}}}_{\mathcal{C}_{\Delta \boldsymbol{\gamma}}} \dot{\mathbf{q}}_k \quad (25)$$

End-effector velocity jumps Given the joint velocity jump $\Delta \dot{\mathbf{q}}$, (23), the end-effector velocity jumps are:

$$\begin{aligned} \Delta \dot{\mathbf{x}}_{\sigma_m, k+1} &= J_{\sigma_m} \Delta \dot{\mathbf{q}}_{k+1} \\ &= \underbrace{J_{\sigma_m} \mathcal{J}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{J}_{\Delta \dot{\mathbf{x}}}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{J_{\sigma_m} \mathcal{C}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{C}_{\Delta \dot{\mathbf{x}}}} \dot{\mathbf{q}}_k. \end{aligned} \quad (26)$$

Angular momentum jump Given (23) and the angular part $A_{\omega c}(\mathbf{q}_k)$ of the centroidal momentum matrix $A_c(\mathbf{q}_k)$, we obtain $\Delta \mathcal{L}_c(\mathbf{q}_{k+1})$ as:

$$\begin{aligned} \Delta \mathcal{L}_c(\mathbf{q}_{k+1}) &= A_{\omega c}(\mathbf{q}_k) \Delta \dot{\mathbf{q}}_{k+1}, \\ &= \underbrace{A_{\omega c}(\mathbf{q}_k) \mathcal{J}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{J}_{\Delta \mathcal{L}_c}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{A_{\omega c}(\mathbf{q}_k) \mathcal{C}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{C}_{\Delta \mathcal{L}_c}} \dot{\mathbf{q}}_k, \end{aligned} \quad (27)$$

and we set $A_{\omega c}(\mathbf{q}_{k+1}) = A_{\omega c}(\mathbf{q}_k)$.

Planar COM Velocity Jump $\Delta \dot{c}_{x,y} \in \mathbb{R}^2$, i.e., in the x and y directions writes

$$\begin{aligned} \Delta \dot{c}_{x,y}(k+1) &= \frac{1}{M} A_{vc}(\mathbf{q}_k) \Delta \dot{\mathbf{q}}_{k+1}, \\ &= \underbrace{\frac{1}{M} A_{vc}(\mathbf{q}_k) \mathcal{J}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{J}_{\Delta \dot{c}_{x,y}}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{\frac{1}{M} A_{vc}(\mathbf{q}_k) \mathcal{C}_{\Delta \dot{\mathbf{q}}}}_{\mathcal{C}_{\Delta \dot{c}_{x,y}}} \dot{\mathbf{q}}_k, \end{aligned} \quad (28)$$

where A_{vc} takes the corresponding rows from A_c , and we use the same approximation as (27).

ZMP Jump We use the predicted force jumps (24) to calculate the resultant wrench:

$$\begin{aligned} \Delta \mathbf{W}_O(k+1) &= J_{O p_{\sigma_e}}^\top \Delta \mathbf{f}_{\sigma_e} \\ &= \underbrace{J_{O p_{\sigma_e}}^\top \mathcal{J}_{\Delta \mathbf{f}_{\sigma_e}}}_{\mathcal{J}_{\Delta \mathbf{W}_O}} \ddot{\mathbf{q}}_k \Delta t + \underbrace{J_{O p_{\sigma_e}}^\top \mathcal{C}_{\Delta \mathbf{f}_{\sigma_e}}}_{\mathcal{C}_{\Delta \mathbf{W}_O}} \dot{\mathbf{q}}_k, \end{aligned} \quad (29)$$

where each element of the stacked Jacobian $J_{O p_{\sigma_e}}$ writes:

$$J_{O p_i}^\top = [I^\top, \overrightarrow{O p_i} \times^\top]^\top.$$

Injecting the wrench jump $\Delta \mathbf{W}_O = [\Delta \mathbf{f}_O^\top, \Delta \boldsymbol{\tau}_O^\top]^\top$ into the ZMP definition (5), we obtain the ZMP jump:

$$\Delta \mathbf{z}_{k+1} = \frac{\hat{n} \Delta \boldsymbol{\tau}_O}{\mathbf{n}^\top \mathbf{f}_O + \Delta \mathbf{f}_O}. \quad (30)$$

5 Impact-aware QP control based on Predicted State Jumps

Based on the predicted impact-induced state jumps, we propose novel impact-aware constraints that guarantee the feasibility of the QP controller after impact when it occurs. They extend the constraints formulation detailed in Sec. 3.

5.1 Constraining generic quantity jumps

We constrain post-impact state of the generic quantity λ

$$D\lambda_{k+1}^+ \leq \bar{\lambda} \quad (31)$$

with the matrix D representing half-planes and the offset vector $\bar{\lambda}$ representing the upper and lower bounds. Substituting the Euler forward method $\lambda_{k+1}^- = \lambda_k + \dot{d}_k \Delta t$ and the impact model $\lambda_{k+1}^+ = \lambda_{k+1}^- + \Delta \lambda_{k+1}$ we obtain

$$D(\lambda_k + \dot{d}_k \Delta t + \Delta \lambda_{k+1}) \leq \bar{\lambda}. \quad (32)$$

We reformulate the constraint regarding the QP decision variable \dot{q}_k by using the generic expression for the impact-induced jump $\Delta \lambda_{k+1}$ (22)

$$D\mathcal{J}_{\Delta\lambda}\ddot{q}_k\Delta t \leq \bar{\lambda} - D(\lambda_k + \dot{d}_k\Delta t + \mathcal{C}_{\Delta\lambda}\dot{q}_k).$$

Note that this constraint is easily adapted such that only actuated joints \mathbf{q} are constrained. Further note that it is typically challenging to obtain an accurate measurement of \dot{d}_k . Therefore, we approximate* the pre-impact state as $\lambda_{k+1}^- \approx \lambda_k$, and consequently, ignore the term $\dot{d}_k\Delta t \simeq 0$:

$$D\mathcal{J}_{\Delta\lambda}\ddot{q}_k\Delta t \leq \bar{\lambda} - D(\lambda_k + \mathcal{C}_{\Delta\lambda}\dot{q}_k). \quad (33)$$

In the following, we consider joint velocity, joint torque, fulfilling contact wrench cone, centroidal angular momentum, COM velocity, and ZMP. All these derivations follow this generic template.

5.2 Constraining joint space jumps

5.2.1 Joint velocity Choosing joint velocities as quantity $\lambda := \dot{q}$ in (33) with $D := [I, -I]^T$ and $\bar{\lambda} := \bar{q}$ yields:

$$\begin{bmatrix} I \\ -I \end{bmatrix} (I + \mathcal{J}_{\Delta\dot{q}}) \ddot{q}_k \Delta t \leq \begin{bmatrix} \bar{q} \\ -\bar{q} \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} (I + \mathcal{C}_{\Delta\dot{q}}) \dot{q}_k, \quad (34)$$

where $\mathcal{J}_{\Delta\dot{q}}$ and $\mathcal{C}_{\Delta\dot{q}}$ are defined by collecting the corresponding rows from $\mathcal{J}_{\Delta\dot{q}}$ and $\mathcal{C}_{\Delta\dot{q}}$.

5.2.2 Impulsive joint torque Injecting the impulsive joint torque jumps $\lambda := \Delta\gamma$ into (33) with $D := [I, -I]^T$ and $\bar{\lambda} := \overline{\Delta\gamma}$ leads to:

$$\begin{bmatrix} I \\ -I \end{bmatrix} \mathcal{J}_{\Delta\gamma} \ddot{q}_k \Delta t \leq \begin{bmatrix} \overline{\Delta\gamma} \\ -\overline{\Delta\gamma} \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} (\mathcal{C}_{\Delta\gamma} \dot{q}_k), \quad (35)$$

where $\Delta\gamma_k = 0$ since the impact is assumed to take place at the time step $k + 1$.

Impacts propagate through mechanical linkages to reach the gear-box, the torque sensor (if any)... for each joint. Hence impact must be mitigated to not damage any of the previous items, including the mechanical linkages or covers... In fact, it is not important what could be the exact

source of the limitation as, at the end, one will take the conservative (min/max) shock that is allowed for each joint or for the robot outers. When impact-task is needed, our QP controller embed concurrently the two set of constraints, the stall torque limits are set for the second order dynamics constraints, and the impact-torque limits are set for the impact-aware constraints. By doing this, higher tolerance of torques are set by (35).

5.3 Fulfilling contact wrench cone

Considering in (33) all the m_1 established contacts using $\lambda := \mathbf{W}$, $\bar{\lambda} := \mathbf{0}$ and $D := \mathcal{C}_f$ yields:

$$\mathcal{C}_f \mathcal{J}_{\Delta f} \ddot{q}_k \Delta t \leq -\mathcal{C}_f (\mathbf{W}_k + \mathcal{C}_{\Delta f} \dot{q}_k), \quad (36)$$

where the current \mathbf{W}_k is measurable. \mathcal{C}_f collects the corresponding force columns from \mathcal{C} , see (4).

5.4 Constraining centroidal space state jumps

Suppose the angular momentum is bounded by $\mathcal{L}_c \leq \bar{\mathcal{L}}_c$, it fits(33) with $D := I$ and $\bar{\lambda} := \bar{\mathcal{L}}_c$:

$$\mathcal{J}_{\Delta\mathcal{L}_c} \ddot{q}_k \Delta t \leq \bar{\mathcal{L}}_c - (\mathcal{L}_{c_k} + \mathcal{C}_{\Delta\mathcal{L}_c} \dot{q}_k). \quad (37)$$

The horizontal COM velocity injected into (33) with $\lambda := \dot{c}_{x,y}$, $D := \mathcal{G}_{\dot{c}_{x,y}}$ and $\bar{\lambda} := \mathbf{h}_{\dot{c}_{x,y}}$ holds:

$$\mathcal{G}_{\dot{c}_{x,y}} \mathcal{J}_{\Delta\dot{c}_{x,y}} \ddot{q}_k \Delta t \leq \mathbf{h}_{\dot{c}_{x,y}} - \mathcal{G}_{\dot{c}_{x,y}} (\dot{c}_{x,y}(k) + \mathcal{C}_{\Delta\dot{c}_{x,y}} \dot{q}_k) \quad (38)$$

Considering the ZMP $\lambda := z$ in (33) with $D := \mathcal{G}_z$ and $\bar{\lambda} := \mathbf{0}$ results in:

$$\mathcal{G}_z \mathcal{J}_{\Delta\mathbf{W}_O} \ddot{q}_k \Delta t \leq -\mathcal{G}_z (\mathbf{W}_O(k) + \mathcal{C}_{\Delta\mathbf{W}_O} \dot{q}_k), \quad (39)$$

where $\mathbf{W}_O(k)$ is computed based on force measurements.

5.5 Impact-aware whole-body QP controller

Given the derived impact-aware constraints (34-39), we extend the baseline QP controller (9):

$$\min_{\nu} \sum_{i \in \mathcal{I}_o} w_i \|e_i(\nu)\|^2$$

s.t. **Joint space constraints:**

Post-impact joint velocity: (34) ,

Post-impact impulsive joint torque: (35) ,

Joint position, velocity, and torque (2),

Contact space constraints:

Post-impact contact wrench cone: (36),

Fulfilling contact wrench cone: (4),

Centroidal space constraints:

Post-impact angular momentum: (37) ,

Post-impact COM velocity: (38) ,

Post-impact ZMP: (39) ,

Other constraints, e.g., collision avoidance: (32). (40)

*For impact-aware joint velocity constraint (with $\lambda := \dot{q}$), we choose \dot{d}_k as the QP decision variable \ddot{q}_k .

Impact-aware constraints modulate the contact velocity keeping (40) feasible while maximizing impact-tasks performances. Experiments in Sec. 6, show that the impact-aware QP controller (40) permits high contact velocities.

The high contact velocity tasks are completed upon the detection of the impact. We can stop using the impact-aware constraints by switching the QP controller from (40) to (9). Since the constraints associated with (40) are more conservative than (9), the switch from (40) to (9) does not lead to infeasible solutions.

Remark 5.1. *Triggering impact-aware constraints during robot motion can result in an infeasible QP. Indeed, previewing one-step ahead may lead to non-feasible deceleration. In our experiments with HRP-4, the time step is 5 ms, which might be not enough for the robot to decelerate instantly to meet impact-aware bounds, see the visualized jump of the feasible search space boundaries before-and-after the impact-awareness activation in Figure 2.*

This problem is common to other usual constraints in which deceleration is needed to meet the bounds as zero velocity and acceleration, see Del Prete (2018); Djeha et al. (2020), and we discuss it in future work.

5.6 Illustrative example

Consider a planar manipulator with two rotational joints to illustrate the joint velocity constraint. The link length is $l_1 = l_2 = 0.5$ m. The joint velocity limits $\dot{\mathbf{q}} = [0.9, 0.6]^T$ rad/s and $\underline{\dot{\mathbf{q}}} = -\dot{\mathbf{q}}$ are implemented as:

$$\begin{bmatrix} I \\ -I \end{bmatrix} \Delta t \ddot{\mathbf{q}}_k \leq \begin{bmatrix} \dot{\mathbf{q}} \\ -\dot{\mathbf{q}} \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} \dot{\mathbf{q}}_{k-1}. \quad (41)$$

We chose the joint configuration $\mathbf{q}_k = [0, 0, 0.2\pi]^T$ rad for demonstration. The current end-effector velocity is $\dot{\mathbf{x}}_k = [0, 0.3]^T$ m/s. Note that the robot is non-redundant, and there exists only one solution in joint space that performs a desired end-effector motion: $\dot{\mathbf{q}} = J^{-1}\dot{\mathbf{x}}$ and its derivative $\ddot{\mathbf{q}} = J^{-1}\ddot{\mathbf{x}} - J^{-1}\dot{J}\dot{\mathbf{q}}$. We choose our real robot controller sampling time $\Delta t = 5$ ms.

In this example, the aim is to move the end-effector in the positive y -direction. In the next control cycle, we aim to achieve an artificially high end-effector acceleration $\ddot{\mathbf{x}}_{\text{des}} = [0, 0, 120.0]^T$ m/s² as exclusive control task (and equal weighting for x - and y -direction). In other words, we can optimally track the end-effector task when minimizing the Euclidean norm $\|J\dot{\mathbf{q}}_k + \dot{J}\dot{\mathbf{q}}_k - \ddot{\mathbf{x}}_{\text{des}}\|^2$. Black contour lines and the grey-to-white fading in Fig 2 represent this objective function. We obtain the QP

$$\begin{aligned} \ddot{\mathbf{q}}_k = \underset{\ddot{\mathbf{q}}_k}{\text{argmin}} \quad & \ddot{\mathbf{q}}_k^\top J^\top J \ddot{\mathbf{q}}_k + 2\ddot{\mathbf{q}}_k^\top J^\top (J\dot{\mathbf{q}}_k - \ddot{\mathbf{x}}_{\text{des}}) \\ \text{s.t. (41) [and (34) for impact-awareness]} \end{aligned} \quad (42)$$

because the scalar term $(\dot{\mathbf{q}}_k^\top J^\top - \ddot{\mathbf{x}}_{\text{des}}^\top)(J\dot{\mathbf{q}}_k - \ddot{\mathbf{x}}_{\text{des}})$ is constant and does not affect the optimization process.

Dotted lines in Figure 2 depict the classical joint velocity constraint without impact-awareness (41). The resulting feasible polytope in terms of joint accelerations constitutes a rectangle with an offset from the origin.

Next, let us expect an impact at the end-effector in the next iteration. The contact normal corresponds to the

y -axis, and we select the coefficient of restitution $c_r = 0.02$. The convex bounds of the impact-aware joint velocity constraint (34) are indicated by dashed lines. Note that this constraint corresponds to a polyhedron with parallel bounds. In practice, we need to superimpose both constraints to obtain a shrunken, convex polytope, shown in blue solid lines. Extreme vertices are indicated by star symbols[†].

The optimal impact-aware solutions found by (42) and send as references to the robot are marked in Figure 2 by colored circles: employing (41) only as the impact-unaware solution (red) and the impact-aware solution (green) with (41) and (34). We cannot fully achieve the desired end-effector acceleration in both cases. The impact-aware solution is more conservative because of the additional constraint. The post-impact joint velocity is given by

$$\dot{\mathbf{q}}_{k+1}^+ = (I + \mathcal{J}_{\Delta\dot{\mathbf{q}}}) (\Delta t \ddot{\mathbf{q}}_k + \dot{\mathbf{q}}_k). \quad (43)$$

Without impact-awareness $\dot{\mathbf{q}}_{k+1}^+ = [-0.618, 1.345]^T$ rad/s violates the second joint's velocity limit. Instead, with impact-awareness $\dot{\mathbf{q}}_{k+1}^+ = [-0.281, 0.600]^T$ rad/s. The deceleration ensures the satisfaction of the joint velocity constraint in the next iteration.

In Wang and Kheddar (2019) we report a comparison with a simulated KR5 manipulator including a physics engine linked to our controller. It highlights consequences of impact-aware and impact-unaware QP control when a contact velocity of 0.72 m/s is used to crash against a wall.

6 Experiments

We assess our impact-aware QP controller (40) on the humanoid HRP-4. Two sets of experiments are performed; each has different highlights. We encourage the readers to check the experiment videos[‡].

Experiment 1 Pushing a concrete wall with the maximum contact velocity –see the illustration in Figure 3 and experiment setup in Sec. 6.1, with the following highlights:

- (H-1.1) Impact-aware control updating the contact velocity in real-time instead of the desired, infeasible 0.8 m/s. At the impact time, the contact velocity is measured to be 0.35 m/s.
- (H-1.2) High contact velocity jump and subsequent high contact force jump (137 N).
- (H-1.3) Robustness to the uncertain location of the wall along the pre-defined contact normal.
- (H-1.4) All the impact-aware constraints (34-39) are fulfilled despite the relatively high contact velocity and contact force jumps.

Experiment 2 Grabbing an empty cardboard box, see the illustration in Figure 5 and experiment setup in Sec. 6.2, with the following highlights:

- (H-3.1) Swift box-grabbing motion without stopping for establishing contacts.
- (H-3.2) Two simultaneous impacts.

[†]The figure was generated using [Matlab scripts](#) to convert between halfspace- and vertex-representation.

[‡]Online: <https://youtu.be/v1Jfy8-jiwE>

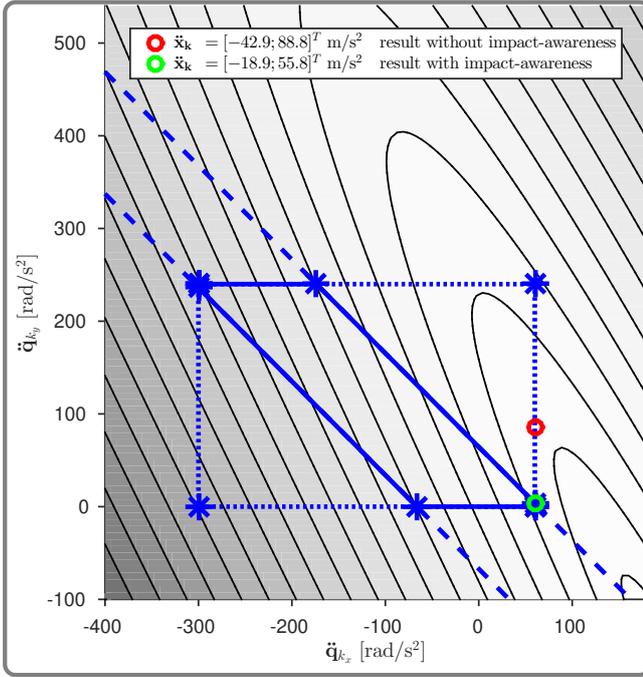


Figure 2. Feasible polytope of joint velocity constraints for a 2-DoF robot: constraint without impact-awareness (dotted lines); constraint with impact-awareness (dashed lines); and superposition of both constraints (blue solid lines). Star symbols indicate the polytopes' extreme vertices. The task objective function is represented by black contour lines and the grey-to-white fading. Circles indicate solutions of the QP (42) without and with the proposed impact-awareness.

For *Experiment 1* we first present the detailed experiment setup and confirm the highlights (H-1.1) and (H-1.2). Then we check the feasibility (H-1.4) in joint space (34-35) and and centroidal space (38-39) in different subsections.

In the internal visualization related to experiment video of *Option C*, the wall location does not match the reality. However, the controller is not affected as it is independent of the exact impact location (H-1.3).

For *Experiment 2*, the highlights (H-3.1) and (H-3.2) are demonstrated by the contact velocity profile in Fig. (9(a)-9(b)) and the contact force profile Fig. (9(c)-9(d)).

6.1 Pushing a wall

We designed *Experiment 1* in order to show that the impact-aware constraints enable the QP to autonomously determine the maximum allowable contact velocity online.

In Sec. 6.1.4, we compare three ZMP constraint formulations: each restricts the ZMP differently:

Option A : the feet's support polygon: $z \in S_s$.

Option B : the multi-contact ZMP area: $z \in S_z$.

Option C : the static COM equilibrium area: $z \in S_c$.

We conclude that *Option C* leads to the highest contact velocity mentioned in (H-1.1).

6.1.1 Experimental setup We present the technical details structured in the following three aspects.

Robot configuration The humanoid is initialized with a standing posture in double coplanar supports and commanded its right gripper's palm to hit a wall in front of it with an infeasible high reference contact velocity (0.8 m/s),

see Figure 6(e). The hand was prepared by mounting a 3D printed rigid plastic palm of 3 cm thickness.

Parameter configuration The QP controller runs at $\Delta t = 5$ ms, and the four force-torque sensors –mounted at the ankles and the wrists, are acquired together with other sensors at each controller loop. Since exact impact duration cannot be measured if less than Δt and impacts are recorded by the force sensors, we assume the impact duration δt to be at most 5 ms and set it to this conservative value. The friction coefficient of the established contacts is 0.7. The coefficient of restitution is $c_r = 0.02$.

Remark 6.1. Uncertain coefficient of restitution \tilde{c}_r leads to an uncertain prediction of the end-effector velocity jump $\Delta \tilde{x}_{k+1}$ due to (12), and accordingly, to uncertain predictions $\tilde{u}^* = [\Delta \tilde{q}^{*T}, \tilde{v}^{*T}]^T$ due to (21). Given \tilde{c}_r is a scalar, the errors

$$e_{u^*} = \begin{bmatrix} e_{\Delta \tilde{q}^*} \\ e_{\tilde{v}^*} \end{bmatrix} = \begin{bmatrix} \Delta \tilde{q}^* - \Delta \tilde{q}^* \\ \tilde{v}^* - v^* \end{bmatrix}$$

are proportional to the error

$$e_{c_r} = (1 + \tilde{c}_r) - (1 + c_r) = \tilde{c}_r - c_r.$$

Through the analysis from several trial-runs, we choose the coefficient of restitution $c_r = 0.02$, which leads to a reasonable prediction of contact force jump Δf , see Figure 6(f). We can observe Δf from the ATI sensors more precisely than estimating $\Delta \dot{q}$ from encoders.

Task description Figure 4(a) reports the finite state machine (FSM) used to modulate and change the behavior of the robot. The *Start* state initializes the right palm to be parallel to the wall. The impact-aware QP (40) is activated in the *Impact* state. It then adjusts the task-steered contact velocity (0.8 m/s), see Figure 6(e). The transition from *Impact* state to *Admittance* state depends upon the *impact detection criterion*, i.e., 20 N threshold measured by the force sensor. During the *Admittance* state, the robot regulates the contact force to 15 N for 14 seconds, thereby stabilizing the contact. Then the *Detach* state releases the contact and the *Reset* state returns to the initial configuration.

It is a good practice to regulate the contact wrenches to stabilize the under-actuated robot center of mass dynamics, e.g. Caron et al. (2019). Yet, in order to observe impact-induced state jumps without interference, especially the contact force jumps and the ZMP jumps, we decided not to apply admittance behavior for the established contacts (feet) in the experiments. Instead, we apply task-space position control for the feet.

6.1.2 Contact velocity regulation Given the infeasible high reference contact velocity 0.8 m/s displayed by purple dashed line in Figure 6(e), the QP controller (40) updates the feasible contact velocity (displayed in orange line) concerning the impact-awareness in each time-step. Thus we do not rely on a pre-specified contact location (H-1.3).

The measured contact force jump is close to its prediction in Figure 6(f), which indicates a reasonable coefficient of restitution c_r . see Remark. 6.1.

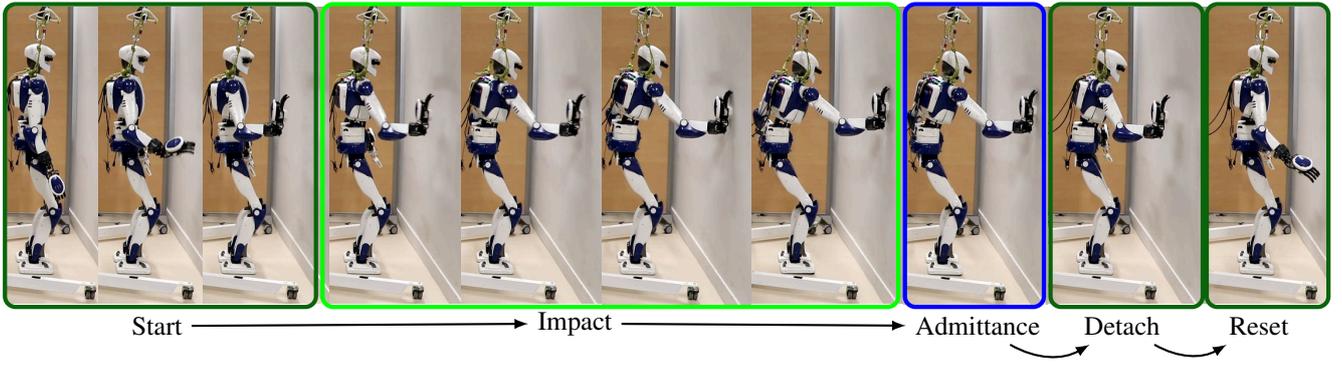


Figure 3. Snapshot of *Experiment 1* with *Option C*: the HRP-4 robot hit the wall with contact velocity 0.35 m/s.

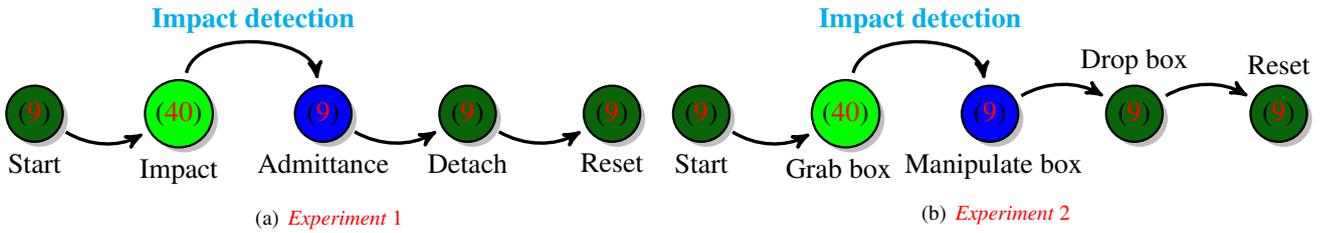


Figure 4. Finite state machines of *Experiment 1* and *Experiment 2* with the corresponding snapshots in Figure 3 and 5 respectively. The impact-aware QP (40) is applied in the green-highlighted *Impact* states. The contact (or contacts for box-grabbing) are set, and admittance tasks are activated in the blue-highlighted states *Admittance* and *Manipulate box*.

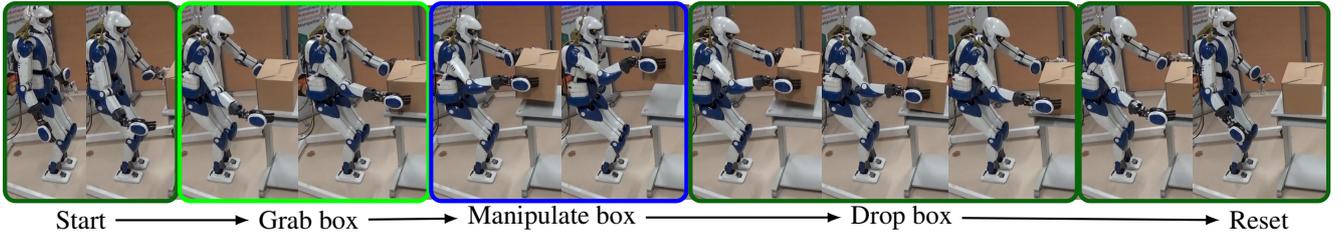


Figure 5. Snapshot of *Experiment 2*: the HRP-4 robot grabbing a box with contact velocities 0.15 m/s at both hands.

6.1.3 Joint space constraints We present plots from two out of all the joints, namely the right arm shoulder roll joint and the elbow joint. In Figure 6(a) and Figure 6(b) the measured joint velocities fulfill the limits (32). To improve the visibility while scaling the figures properly, we ignored the well fulfilled joint velocity bounds $\{-2.052, 2.052\}$ rad/s in Figure 6(a) and $\{-2.649, 2.649\}$ rad/s in Figure 6(b).

The HRP-4 is not equipped with joint torque sensors. Thus we obtain the torque jumps applying the well-known relation: $\Delta\gamma = J^T \Delta f$. Since the HRP-4 robot is not backdrivable and has fragile components, we selected $\overline{\Delta\gamma} = 0.4\bar{\tau}$ and $\underline{\Delta\gamma} = 0.4\underline{\tau}$ for constraint (35). In Figure 6(c) and Figure 6(d), the impulsive joint torque jumps are well-bounded in line with (32).

6.1.4 ZMP constraint The feasible ZMP area is recently extended to general multi-contact settings, yet considering static contacts. Despite the well-established push recovery strategies for planar bipedal maneuvers, Stephens (2007); Sugihara (2009), in the case of an impact event resulting in a discontinuous change of the multi-contact situation, it is unclear which bounds apply to the ZMP. Running comparative experiment trials, we investigated three formulations of

the ZMP bounding polygon illustrated in Figure 10. These are injected into the impact-aware ZMP constraint (39).

Option A: The ZMP is restricted to remain within the support polygon \mathcal{S}_s defined by the feet contacts: $z \in \mathcal{S}_s$. In other words, the classical ZMP area for coplanar contacts is applied (Vukobratović and Borovac 2004), and the contact to be established by the hand is ignored. Figure 8 plots the results. This choice leads to a conservative contact velocity 0.11 m/s as shown in Figure 8(a) and a small contact force jump 37 N displayed in Figure 8(b). Note that $z \in \mathcal{S}_s$ is fulfilled, see Figure 8(c). The higher the contact velocity, the higher the ZMP jump. Given that in Figure 8(d) the predicted Δz_x overlaps the bound of \mathcal{S}_s , we conclude the robot applied the maximum contact velocity while fulfilling $z \in \mathcal{S}_s$.

Option B: We restrict the ZMP within the multi-contact ZMP area \mathcal{S}_z defined by the feet contacts and the hand contact as formulated in Caron et al. (2017): $z \in \mathcal{S}_z$. Notice that we apply an optimized version that meets real-time requirements. Compared to *Option A*, we observe a slightly higher contact velocity 0.17 m/s. Due to the similarity, we do not plot figures for this trial run.

Option C: Finally, given the contacts of feet and the hand, we restrict the ZMP within the static COM equilibrium area

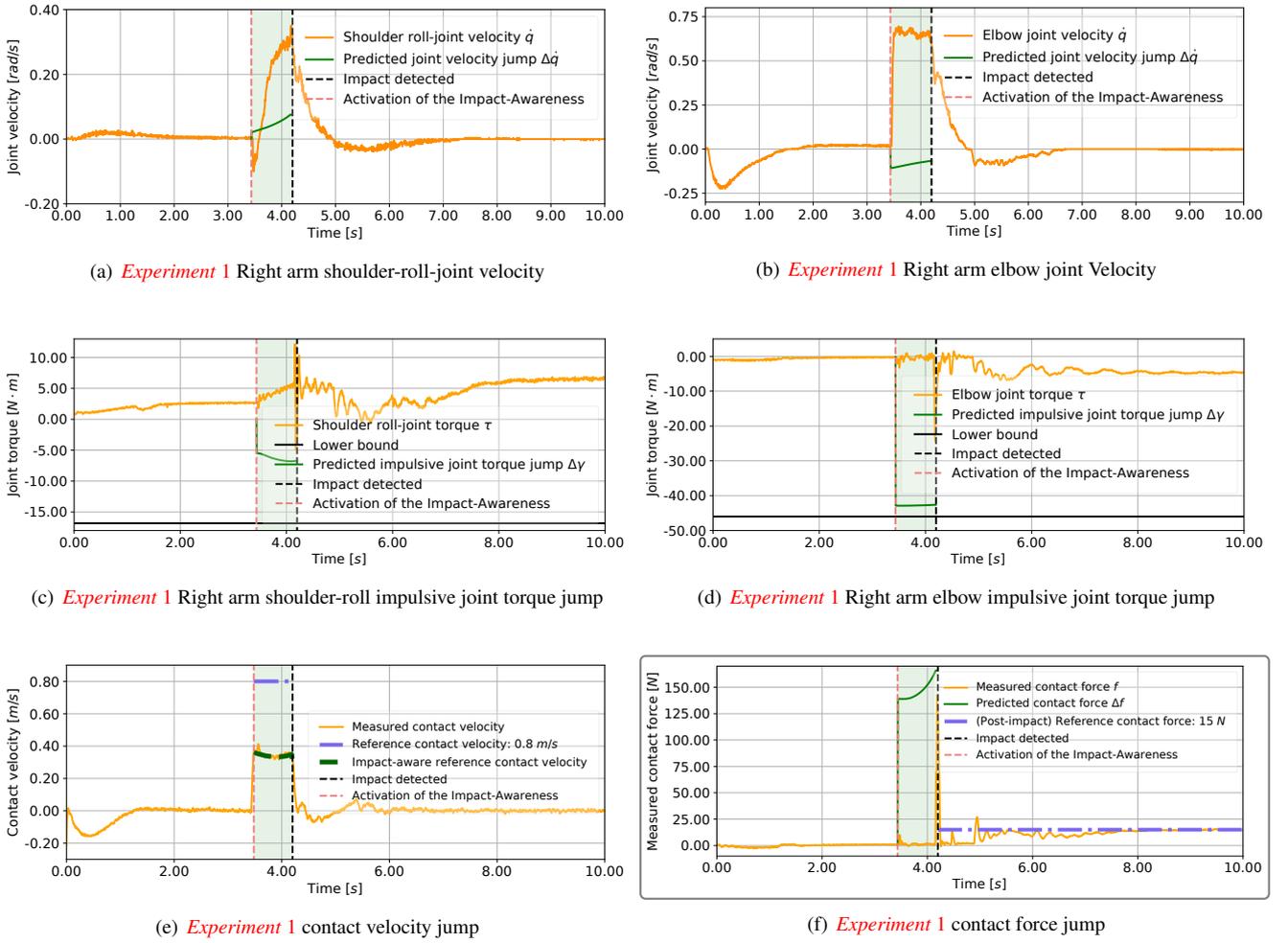


Figure 6. Impact-aware joint space and contact space constraints of *Experiment 1* with *Option C*, see the discussions from Sec. 6.1.2 to Sec. 6.1.3.

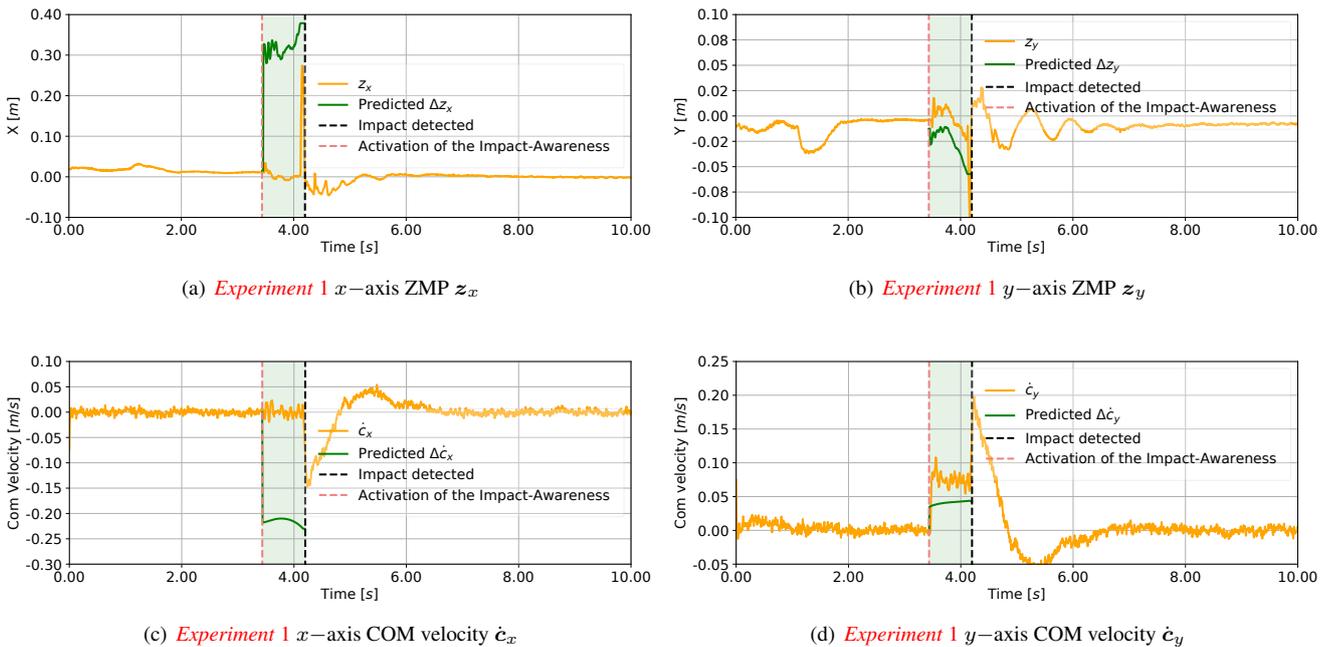


Figure 7. Impact-aware centroidal space constraints of *Experiment 1* with *Option C*, see the discussions from Sec. 6.1.4 to Sec. 6.1.5.

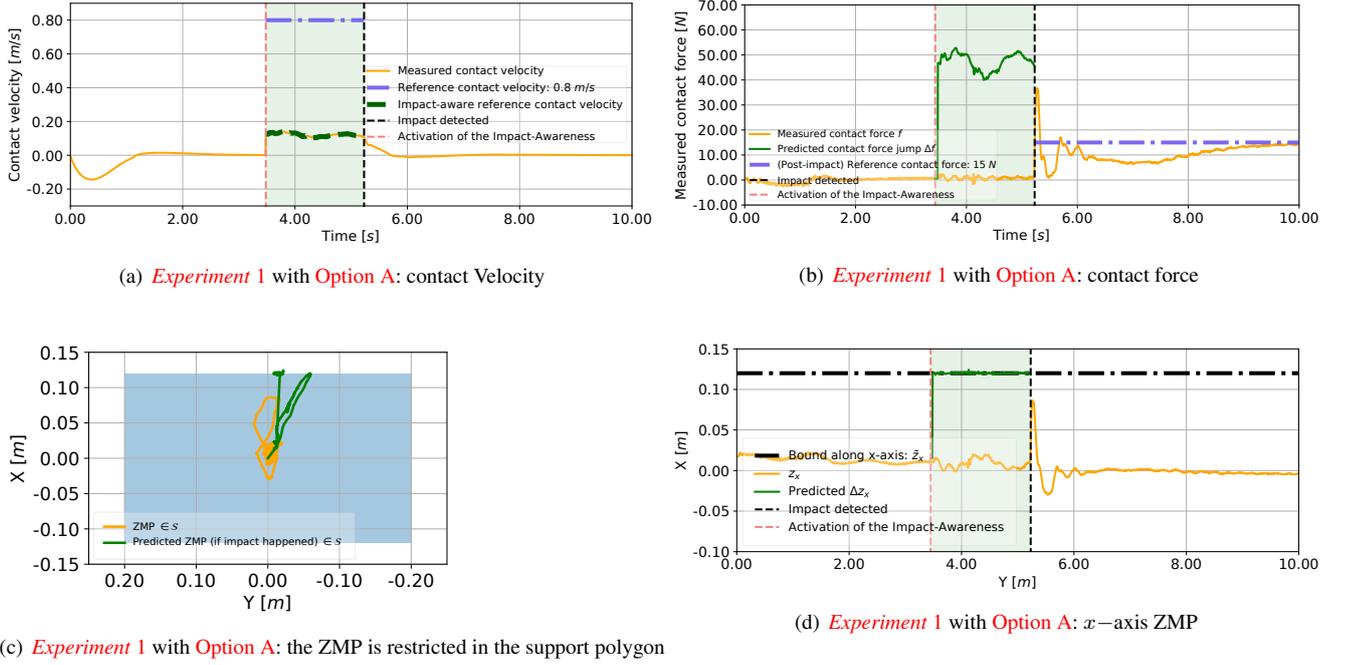


Figure 8. Impact-aware constraints of *Experiment 1* with Option A, see the discussions in Sec. 6.1.4.

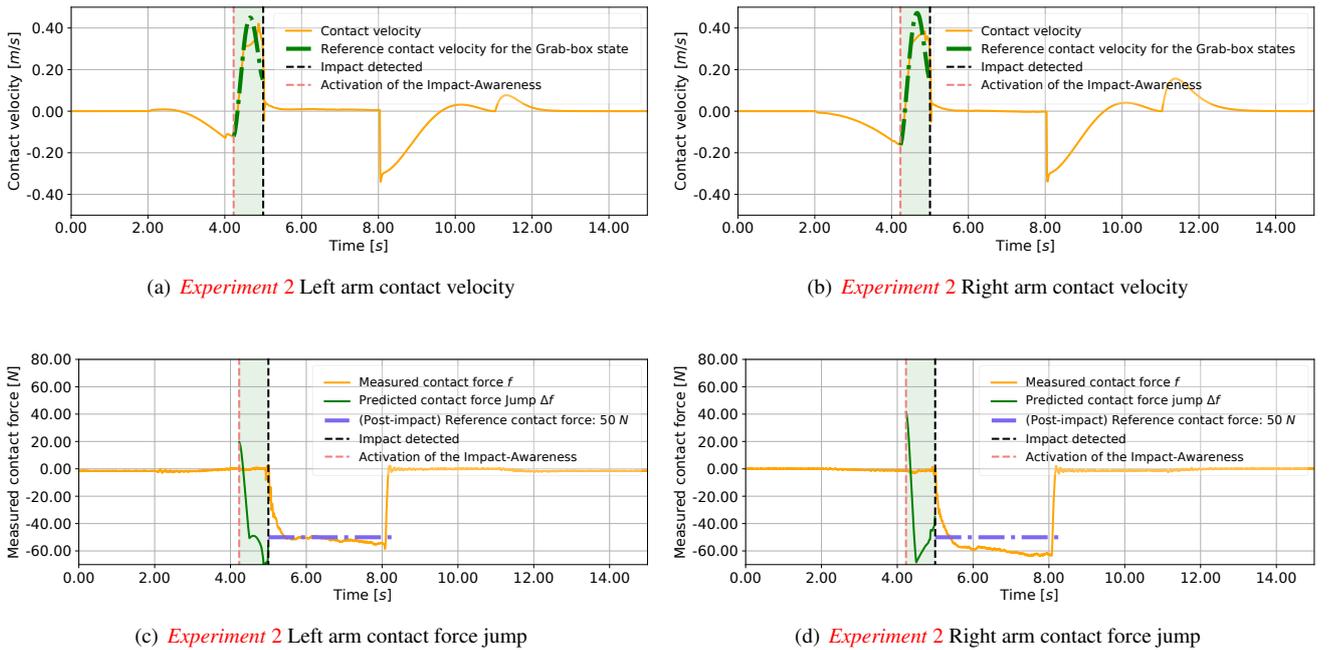


Figure 9. Contact velocities and contact force jumps of *Experiment 2*, see the discussions from Sec. 6.2.1 to Sec. 6.2.2.

\mathcal{S}_c : $z \in \mathcal{S}_c$, for more details on \mathcal{S}_c , refer to [Bretl and Lall \(2008\)](#). We employ an optimized algorithm to compute \mathcal{S}_c . Figure 11 illustrates the evolution over time of the real ZMP (5) and the predicted ZMP $z + \Delta z$ (30). Figure 7 reports further results. The robot applies the contact velocity 0.35 m/s, which is the highest among all trials (H-1.1), observing the associated contact force jump 137 N (H-1.2). We note that all constraints are fulfilled, the hardware does not break and the robot maintains balance (H-1.4).

6.1.5 COM velocity constraints We did not defined bound for the COM velocity in Fig. (7(c)-7(d)), hence we only plot

the actual value against the predicted impact-induced state jumps. The important observation is that the actual jump is close to the prediction. Given the LIPM dynamics, the ZMP jump Δz and the COM velocity jump $\Delta \dot{c}$ takes the opposite sign. We can observe the correspondance between Figure 7(a)-7(b) and Figure 7(c)-7(d).

For plots in Sec. 6.1.4 and Sec. 6.1.5, there exists a general observation that along with the contact normal direction (the x -axis plots), predictions of the floating-base state jumps can bound the actual jumps, and the discrepancies are relatively small. Since our impact model (12) considers only

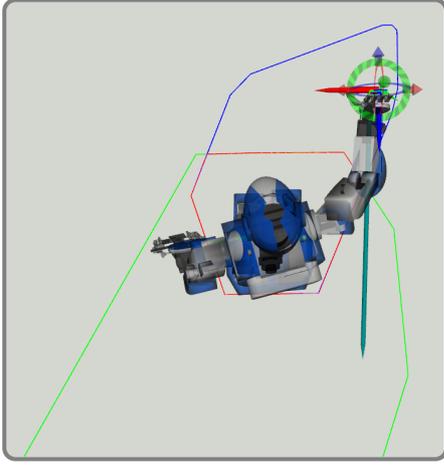


Figure 10. Multi-contact ZMP area \mathcal{S}_z , that defines **Option B** (light green polygon), and the multi-contact static COM equilibrium area \mathcal{S}_c , that defines **Option C** (blue polygon). The light blue arrow on the right hand indicates the contact force jump. The red area indicates the intersection: $\mathcal{S}_z \cap \mathcal{S}_c$.

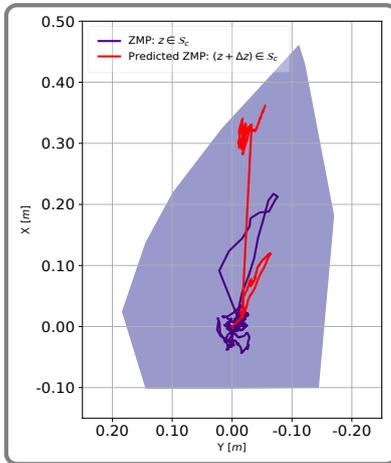


Figure 11. In *Experiment 1*, the ZMP is restricted in the multi-contact COM area \mathcal{S}_c which is updated in every control cycle according to the current posture. This polygon deviates from Figure. 10 as it shows a slightly different stance.

one-dimensional/directional impact, predictions of lateral direction (y -axis plots) are not as good. We consider this acceptable for this study since the magnitude of the lateral direction (y -axis plots) jumps are much smaller.

6.2 Box-grabbing with a swift motion

In *Experiment 2*, the HRP-4 robot lifts a cardboard box using swift motion without reducing its hands' speed at contacts.

6.2.1 Experimental setup

Robot configuration The initial robot configuration is the same as in *Experiment 1* in front of a table on which a cardboard box is put. We mounted a felt on the plastic palm to increase the friction with the cardboard. Due to the payload limitation of the HRP-4 (< 500 g due to the weakness of the wrist) and also the limited contact area and friction coefficient between the left palm and the box, the latter was empty, i.e. not loaded.

Task description Figure 4(b) shows that *Experiment 2* follows a similar FSM as *Experiment 1*.

The *Start* state raises the two palms; then the *Grab box* state moves the two palms following pre-defined trajectories considering an approximate location and size of the box. The two hands contacted the box with velocities at 0.15 m/s, and all the constraints are respected. Upon the same *impact detection criterion*, the *Manipulate box* state activates two admittance tasks that regulate the contact forces to 50 N. The *Drop box* state moves the two palms following another set of pre-defined trajectories to release the contacts and then resume the initial robot configuration in the *Reset* state.

6.2.2 Results We plotted the contact velocities for the left and right arms in Figure 9(a) and Figure 9(b), respectively. In both cases, the contact velocities are close to the reference 0.15 m/s. Thus we say the contacts are established without either slowing down or following pre-defined deceleration trajectories (H-3.1). Comparing the impact detection time, we can find that the two impacts are simultaneous (H-3.2).

The impact-induced contact force jumps of left and right arms are presented in Figure 9(c) and Figure 9(d), which are smaller compared to *Experiment 1* due to the slower reference contact velocities.

Like *Experiment 1*, the contact force jumps are well predicted (with a bit conservative margin). On the other hand different from *Experiment 1*, the reference contact velocities are well tracked (instead of reducing from 0.8 m/s to 0.35 m/s) as the impact-induced state jumps will not break the impact-aware constraints (36-39).

Note that demonstrating a QP failure due to a pre-impact velocity that was too large would result in hardware failure or a severe fall, which we can not afford with our robot.

7 Conclusion and future work

The main contribution of this work is in enabling task-space QP control schemes to achieve robot motions with impact tasks safely. Our enables physical contact established at non-zero speed by optimizing pre-impact velocities under impact-aware constraints. Indeed, in the vicinity of impact, the controller encompasses in a one-iteration ahead, the jumps that would be induced by a desired impact (shall it occur) and enforce all the constraints to be aware of subsequent jumps in the state velocity and contact forces constraint. As part of the prediction process, impact propagation along the constrained kinematic tree is also considered. Our current framework include hardware impact-resilience and can sustain pre-existing contacts. It applies to floating-base robots in general multi-contact settings considering balance.

Our approach requires only the contact surface normal to be known. We achieved high contact velocities, and large force jumps in the experiments conducted with an HRP-4 humanoid. To our best knowledge, this is the first approach that performs impact-aware whole-body control based on constrained quadratic optimization.

Now that we confirmed the main concept and encapsulated impact-aware tasks in our general multi-objective and multi-sensory task-space QP control, we plan to investigate the following shortcomings: (i) off-line and on-line identification of impact pertinent parameters to refine the controller performance together with a thorough study of sensitivity and robustness w.r.t parameters' uncertainties; (ii) investigate more advanced impact models that include friction in 3D;

(iii) sliding and tangential impacts so as, for example, to be able to grab objects that are moving; (iv) develop more refined impact propagation models, namely those based on energy spreading and assess them through ground-truth instrumentations; (v) triggering impact-aware constraints with guaranteed feasibility (other than adding slack variable as decision variable or using strict hierarchy schemes).

Acknowledgment

This work was funded by the H2020 EU project I-AM. Yuquan Wang is partially supported by the National Natural Science Foundation of China (U1613216) and Shenzhen Fundamental Research Grant (JCYJ20180508162406177) through the Chinese University of Hong-Kong, Shenzhen.

The authors wish to thank Pierre Gergondet for his continuous support in setting up the `mc_rtc` controller and Stéphane Caron for many fruitful discussions regarding the implementation of multi-contact ZMP area.

References

- Abe Y, Da Silva M and Popović J (2007) Multi-objective control with frictional contacts. In: *ACM SIGGRAPH/Eurographics symposium on Computer animation*. pp. 249–258.
- Addi K and Rodić A (2010) Impact dynamics in biped locomotion analysis: Two modelling and implementation approaches. *AIMS Journal on Mathematical Biosciences and Engineering* 7(3): 479–504.
- Battaglia M, Blanchet L, Kheddar A, Kajita S and Yokoi K (2009) Combining haptic sensing with safe interaction. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 231–236.
- Bergés P and Bowling A (2005) Impact forces in legged robot locomotion. In: *IEEE International Conference on Robotics and Automation*. pp. 3745–3751.
- Bouyarmane K, Chappellet K, Vaillant J and Kheddar A (2019) Quadratic programming for multirobot and task-space force control. *IEEE Transactions on Robotics* 35(1): 64–77.
- Bouyarmane K and Kheddar A (2018) On weight-prioritized multitask control of humanoid robots. *IEEE Transactions on Automatic Control* 63(6): 1632–1647.
- Bretl T and Lall S (2008) Testing static equilibrium for legged robots. *IEEE Transactions on Robotics* 24(4): 794–807.
- Brogliato B (2003) Some perspectives on the analysis and control of complementarity systems. *IEEE Transactions on Automatic Control* 48(6): 918–935.
- Caron S, Kheddar A and Tempier O (2019) Stair climbing stabilization of the hrp-4 humanoid robot using whole-body admittance control. In: *IEEE International Conference on Robotics and Automation*. pp. 277–283.
- Caron S, Pham QC and Nakamura Y (2015) Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas. In: *IEEE International Conference on Robotics and Automation*. pp. 5107–5112.
- Caron S, Pham QC and Nakamura Y (2017) Zmp support areas for multi-contact mobility under frictional constraints. *IEEE Transactions on Robotics* 33(1): 67–80.
- Chang KS and Khatib O (2000) Operational space dynamics: Efficient algorithms for modeling and control of branching mechanisms. In: *IEEE International Conference on Robotics and Automation*. San Francisco, USA, pp. 850–856.
- de Lasa M and Hertzmann A (2009) Prioritized optimization for task-space control. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 5755–5762.
- Decré W, Smits R, Bruyninckx H and De Schutter J (2009) Extending iTaSC to support inequality constraints and non-instantaneous task specification. In: *IEEE International Conference on Robotics and Automation*. pp. 964–971.
- Del Prete A (2018) Joint position and velocity bounds in discrete-time acceleration/torque control of robot manipulators. *IEEE Robotics and Automation Letters* 3(1): 281–288.
- DeMagistris G, Pajon A, Miossec S and Kheddar A (2017) Optimized humanoid walking with soft soles. *Robotics and Autonomous Systems* 95: 52–63.
- Djeha M, Tanguy A and Kheddar A (2020) Adaptive-gains enforcing constraints in closed-loop QP control. *IEEE Robotics and Automation Letters* 5(4): 6504–6511.
- Featherstone R (2014) *Rigid body dynamics algorithms*. Springer.
- Grizzle JW, Abba G and Plestan F (2001) Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *IEEE Transactions on automatic control* 46(1): 51–64.
- Grizzle JW, Chevallereau C, Sinnet RW and Ames AD (2014) Models, feedback control, and open problems of 3d bipedal robotic walking. *Automatica* 50(8): 1955–1988.
- Haddadin S, Albu-Schäffer A and Hirzinger G (2009) Requirements for safe robots: Measurements, analysis and new insights. *The International Journal of Robotics Research* 28(11-12): 1507–1527.
- Heck D, Saccon A, Van de Wouw N and Nijmeijer H (2016) Guaranteeing stable tracking of hybrid position-force trajectories for a robot manipulator interacting with a stiff environment. *Automatica* 63: 235–247.
- Hu G, Makkar C and Dixon WE (2007) Energy-based nonlinear control of underactuated euler-lagrange systems subject to impacts. *IEEE Transactions on Automatic Control* 52(9): 1742–1748.
- Hurmuzlu Y, Génot F and Brogliato B (2004) Modeling, stability and control of biped robots—a general framework. *Automatica* 40(10): 1647 – 1664.
- Hurmuzlu Y and Marghitu DB (1994) Rigid body collisions of planar kinematic chains with multiple contact points. *The international journal of robotics research* 13(1): 82–92.
- Jia YB, Gardner M and Mu X (2019) Batting an in-flight object to the target. *The International Journal of Robotics Research* 38(4): 451–485.
- Jia YB and Wang F (2017) Analysis and computation of two body impact in three dimensions. *Journal of Computational and Nonlinear Dynamics* 12(4): 041012.
- Kajita S, Morisawa M, Miura K, Nakaoka S, Harada K, Kaneko K, Kanehiro F and Yokoi K (2010) Biped walking stabilization based on linear inverted pendulum tracking. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 4489–4496.
- Kanoun O, Lamiroux F, Wieber PB, Kanehiro F, Yoshida E and Laumond JP (2009) Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots. In: *IEEE International Conference on Robotics and Automation*. pp. 2939–2944.

- Konno A, Myojin T, Matsumoto T, Tsujita T and Uchiyama M (2011) An impact dynamics model and sequential optimization to generate impact motions for a humanoid robot. *The International Journal of Robotics Research* 30(13): 1596–1608.
- Kuindersma S, Deits R, Fallon M, Valenzuela A, Dai H, Permenter F, Koolen T, Marion P and Tedrake R (2016) Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonomous Robots* 40(3): 429–455.
- Lee SH and Goswami A (2012) A momentum-based balance controller for humanoid robots on non-level and non-stationary ground. *Autonomous Robots* 33(4): 399–414.
- Liu M, Lober R and Padois V (2016) Whole-body hierarchical motion and force control for humanoid robots. *Autonomous Robots* 40(3): 493–504.
- Nava G, Romano F, Nori F and Pucci D (2016) Stability analysis and design of momentum-based controllers for humanoid robots. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 680–687.
- Nenchev DN (2013) Reaction null space of a multibody system with applications in robotics. *Mechanical Sciences* 4(1): 97–112.
- Nenchev DN (2018) The momentum equilibrium principle: Foot contact stabilization with relative angular momentum/velocity. In: *IEEE-RAS International Conference on Humanoid Robots*. pp. 17–24.
- Orin DE, Goswami A and Lee SH (2013) Centroidal dynamics of a humanoid robot. *Autonomous robots* 35(2-3): 161–176.
- Pagilla PR and Yu B (2001) A stable transition controller for constrained robots. *IEEE/ASME transactions on mechatronics* 6(1): 65–74.
- Pajon A, Caron S, De Magistris G, Miossec S and Kheddar A (2017) Walking on gravel with soft soles using linear inverted pendulum tracking and reaction force distribution. In: *IEEE-RAS International Conference on Humanoid Robots*. p. 432437.
- Partridge CB and Spong MW (2000) Control of planar rigid body sliding with impacts and friction. *The International Journal of Robotics Research* 19(4): 336–348.
- Pashah S, Massenzio M and Jacquelin E (2008) Prediction of structural response for low velocity impact. *International Journal of Impact Engineering* 35(2): 119–132.
- Reher J, Cousineau EA, Hereid A, Hubicki CM and Ames AD (2016) Realizing dynamic and efficient bipedal locomotion on the humanoid robot durus. In: *IEEE International Conference on Robotics and Automation*. pp. 1794–1801.
- Righetti L, Buchli J, Mistry M, Kalakrishnan M and Schaal S (2013) Optimal distribution of contact forces with inverse-dynamics control. *The International Journal of Robotics Research* 32(3): 280–298.
- Rijnen M, de Mooij E, Traversaro S, Nori F, van de Wouw N, Saccon A and Nijmeijer H (2017) Control of humanoid robot motions with impacts: Numerical experiments with reference spreading control. In: *IEEE International Conference on Robotics and Automation*. pp. 4102–4107.
- Singh BRP and Featherstone R (2020) Mechanical shock propagation reduction in robot legs. *IEEE Robotics and Automation Letters* 5(2): 1183–1190.
- Stanisic RZ and Fernández ÁV (2012) Adjusting the parameters of the mechanical impedance for velocity, impact and force control. *Robotica* 30(4): 583–597.
- Stephens B (2007) Humanoid push recovery. In: *IEEE-RAS International Conference on Humanoid Robots*. pp. 589–595.
- Stewart DE (2000) Rigid-body dynamics with friction and impact. *SIAM review* 42(1): 3–39.
- Sugihara T (2009) Standing stabilizability and stepping maneuver in planar bipedalism based on the best com-zmp regulator. In: *International Conference on Robotics and Automation*. pp. 1966–1971.
- Tsujita T, Konno A, Komizunai S, Nomura Y, Owa T, Myojin T, Ayaz Y and Uchiyama M (2008) Analysis of nailing task motion for a humanoid robot. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Nice, France, pp. 1570–1575.
- Vanderborght B, Albu-Schäffer A, Bicchi A, Burdet E, Caldwell DG, Carloni R, Catalano M, Eiberger O, Friedl W, Ganesh G et al. (2013) Variable impedance actuators: A review. *Robotics and autonomous systems* 61(12): 1601–1614.
- Vukobratović M and Borovac B (2004) Zero-moment point thirty five years of its life. *International journal of humanoid robotics* 1(01): 157–173.
- Wang Y and Kheddar A (2019) Impact-friendly robust control design with task-space quadratic optimization. In: *Proceedings of Robotics: Science and Systems*, volume 15. Freiburg, Germany, p. 32.
- Wang Y, Tanguy A, Gergondet P and Kheddar A (2019) Impact-aware multi-contact motion generation with a quadratic optimization controller. In: *IEEE-RAS International Conference on Humanoid Robots*. Toronto, Canada.
- Wensing PM and Orin DE (2016) Improved computation of the humanoid centroidal dynamics and application for whole-body control. *International Journal of Humanoid Robotics* 13(01): 1550039.
- Wensing PM, Wang A, Seok S, Otten D, Lang J and Kim S (2017) Proprioceptive actuator design in the mit cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots. *IEEE Transactions on Robotics* 33(3): 509–522.
- Wieber PB, Tedrake R and Kuindersma S (2016) Modeling and control of legged robots. In: *Springer handbook of robotics*. Springer, pp. 1203–1234.
- Wiedebach G, Bertrand S, Wu T, Fiorio L, McCrory S, Griffin R, Nori F and Pratt J (2016) Walking on partial footholds including line contacts with the humanoid robot atlas. In: *IEEE-RAS International Conference on Humanoid Robots*. pp. 1312–1319.
- Zhang Y, Ge SS and Lee TH (2004) 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)* 34(5): 2126–2132.
- Zheng YF and Hemami H (1985) Mathematical modeling of a robot collision with its environment. *Journal of Field Robotics* 2(3): 289–307.