Safe Impacts with Soft Contacts Based on Learned Deformations

Niels Dehio, Abderrahmane Kheddar

Safe Impacts with Soft Contacts Based on Learned Deformations

Niels Dehio and Abderrahmane Kheddar

***Abstract*— Safely generating impacts with robots is challenging due to subsequent discontinuous velocity and high impact forces. We aim at increasing the impact velocity (i.e., the robot's relative speed prior to contact) such that impact-tasks like grabbing and boxing are made with the highest allowable speed performance when needed. Previous works addressed this problem for rigid bodies' impacts. This letter proposes a control paradigm for generating impacts with deformable contacts that incorporates hardware and task constraints. Based on data-driven learning of the shock-absorbing soft dynamics and a novel mapping of joint-space limits to contact-space, we devise a constrained model-predictive control to maximize the intentional impact within a feasible, safe level. Our approach is assessed with real-robot experiments on the redundant Panda manipulator, demonstrating high pre-impact velocities of a rigid end-effector on soft objects and an end-effector soft suction-pump on rigid or deformable objects.**

I. Introduction

Our work's long-term context is to enable efficient robotic fast grabbing, tossing, and boxing objects in automated industrial sorting chains. In such applications, the robot shall reach, pick and toss or place (box) objects of different sizes, shapes, and materials from one location to another as fast as possible. The "Amazon Picking Challenge" [1] demonstrated the state of the art grasping capabilities in 2015 and revealed that robots could not compete with humans in terms of successful grasps per hour. Ultimately, gaining speed would not occur only in the "reach", "go-to place/box", or "toss", motion phases as proposed in [2], [3], for example, but also in the "contact transition" phases for grasping and placing or releasing the object. Witness in such works (and many others) that almost all robots slow down drastically (up to zero relative velocity) motion when establishing contact with the environment, e.g., to grasp an object. We instead aim to generate maximum impacts intentionally, thereby speeding up the overall operation time in industrial processes.

There are roughly two categories of objects or environments to contact: rigid and soft. We have already addressed the rigid case in [4], [5]; see also other works dealing with rigid robotic impact in their references section. In this letter, we focus on maximizing the impact velocity considering *deformable* contacts. The softness will consume large parts of the kinetic energy and allows higher pre-impact velocities w.r.t rigid bodies. Our main contribution is a model predictive control scheme that optimizes future deformations to be initiated with the highest possible velocity based on learned, constrained deformation dynamics. For the first time, our formulation maps joint-space hardware limits of a redundant kinematic chain onto the contact-space. Including these inequality constraints into the deformation dynamics, ensures safe operation over a sufficiently long preview-horizon. Compared to related works, our approach's novelty is the explicit computation of the maximum safe impact velocity. It applies to three distinct scenarios:

1) rigid end-effector impacting a soft object,
2) end-effector equipped with soft material (e.g., suction cup) impacting a rigid object, and
3) end-effector equipped with soft material impacting a soft object.

We assessed our approach in various real-robot experiments with Panda manipulators, (c.f Fig 1 and shown in the video `https://youtu.be/juynq6x9JJ8`), and indeed observed high impact velocities without violating hardware limits and remaining in the elastic deformation domain.

Fig. 1: Panda manipulators equipped with a rigid flat end-effector (left) and with a pump plus flexible suction cup (right). In the experiments, we identify the deformation dynamics for the soft dice (left) and the sucker (right) in order to generate maximum impact.

II. Related Works

Soft material undergoing contact forces implies local or global shape changes (deformation) and deformation dynamics [6]. Related works (i) propose deformation models and (ii) devise control schemes that facilitate these models.

Deformation Models: Due to its importance, extensive literature on contact dynamics is available to the community; among them, [7] provides an excellent early overview and [8] a recent survey. The finite element method (FEM) is a widely known numerical method to solve partial differential equations such as those inherent to model continuum mechanics. It simulates material deformations by dividing a non-rigid

The authors are with the CNRS-University of Montpellier, LIRMM, Interactive Digital Humans group, France: niels.dehio@lirmm.fr

A. Kheddar is also with the CNRS-AIST Joint Robotics Laboratory, IRL, Japan.

body into many small parts called finite elements, which behavior is well determined, and assembled into a mesh. This technique emerged in mechanical engineering and has been adopted in the field of robotics, see e.g., [9], [10], [11].

In recent work, [12] proposed a method for soft contact model identification. The approach in [13] combines estimation and control of the material's contact normal stiffness in a single framework. While these works focus on contact point forces and point deformations only, [14] models the entire surface's deformation probabilistically. However, this model does not incorporate resulting contact forces. Similarly, [15] proposes a deformation model for the soft human skin, thereby integrating contact forces proportional to deformations.

Control Schemes: The control of contact transitions with soft material was studied extensively over the last decades, mainly to avoid bouncing, e.g., [16] for soft and rigid contacts. The approaches in [17] and [18] rely on a mass-spring interaction model and derive a continuous control law to stabilize the transition between non-contact and contact phase, without requiring force measurements. The analysis and experiments are, however, limited to a planar 2 DOF robot. More recently, the approaches described in [19], [20] and [21] propose parameter optimization for the more complex mass-spring-damper interaction model. Non-linear tangential forces arising from soft interaction have been controlled with a simple toy robot in simulation [22]. Approaches with floating-base legged robots typically try to impose robust control that compensates for non-modeled soft contact properties, for example, with a quadruped in [23] and with a humanoid in [24], [25].

Whereas the above works focus on implementing robust control schemes, [26] proposed an impact-aware planning method. Yet, none of these works is explicitly aware of (or tries to optimize) the maximum possible impact velocity. Also, to the authors' knowledge, related works on impact or task-space model-predictive control did not consider structural hardware limits defined in joint-space. The first step in this direction has been proposed in [27], [28] for a non-redundant robot leg with three degrees of freedom. Our novel approach overcomes these two shortcomings: First, we propose to map hardware limits onto the contact-space. Second, we maximize the pre-impact end-effector velocity subject to constrained deformation dynamics considering a reasonable long preview horizon for predicting accurate behaviors. Exploiting the shock-absorbing soft material, our contribution enables us to generate maximum but safe impacts intentionally.

III. Dynamic Model

Manipulator dynamics in joint-space are given by

$$\boldsymbol{\tau} + \mathbf{J}^T\mathbf{f} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{h} \tag{1a}$$

$$\text{subject to } \underline{\mathbf{q}} \leq \mathbf{q} \leq \bar{\mathbf{q}} \tag{1b}$$

$$\underline{\dot{\mathbf{q}}} \leq \dot{\mathbf{q}} \leq \bar{\dot{\mathbf{q}}} \tag{1c}$$

$$\underline{\ddot{\mathbf{q}}} \leq \ddot{\mathbf{q}} \leq \bar{\ddot{\mathbf{q}}} \tag{1d}$$

$$\underline{\boldsymbol{\tau}} \leq \boldsymbol{\tau} \leq \bar{\boldsymbol{\tau}} \tag{1e}$$

where $\mathbf{M}$ denotes the joint space inertia, $\ddot{\mathbf{q}}$ are the joint accelerations, $\mathbf{h}$ comprises gravity and Coriolis forces, $\mathbf{J}$ constitutes the end-effector Jacobian which we assume to be full-rank in this letter, and $\mathbf{f}$ is a possible contact force. Symbols $\bar{\star}$, $\underline{\star}$ denote upper and lower limits of a quantity $\star$, respectively. The structural hardware limits must be fulfilled to generate a feasible and safe robot motion.

For the following analysis, let us constrain the end-effector not to change its current direction of motion and force. Define an orthonormal basis $\mathbf{O} = [\mathbf{o}_1, \mathbf{o}_{2\ldots6}]$ (with $\mathbf{O}^T\mathbf{O} = \mathbf{I}$) such that the first vector $\mathbf{o}_1$ corresponds to the normalized end-effector velocity. Decompose the end-effector Jacobian $\mathbf{J}$

$$\mathbf{J}_A = \mathbf{o}_1^T\mathbf{J}, \quad \mathbf{J}_B = \mathbf{o}_{2\ldots6}^T\mathbf{J} \tag{2}$$

where the one-dimensional $\mathbf{J}_A$ is associated with the end-effector's direction of motion and $\mathbf{J}_B$ contains the five other rows. We can decompose the task-space quantities velocity $\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$, acceleration $\ddot{\mathbf{x}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}}$, and force $\mathbf{f}$ similarly. In this paper, we enforce $\dot{\mathbf{x}}_B = \mathbf{0}$, $\ddot{\mathbf{x}}_B = \mathbf{0}$, and $\mathbf{f}_B = \mathbf{0}$ before and during soft contact situations. We will drop the subscript $()_A$ and keep only $()_B$ to simplify the notation.

A. Deformation Dynamics

Soft material deforms under externally applied forces (stress). As a consequence, contact forces are functions of surface deformations [6]. In the following, we limit our analysis to soft materials and associated models with non-varying contact dynamics over time, i.e., obeying Hook's law of deformation. Furthermore, we assume that contact friction constraints are always satisfied during the penetration phase; i.e., there is no slipping (as this is what should be planned for the operations we target). As is common in many related works, we model only a one-dimensional deformation in the contact normal direction.

Consider a reference position $\mathbf{x}$ of a robotic end-effector. With respect to this reference, relative positions and velocities are denoted as $\mathbf{z}, \dot{\mathbf{z}}$, respectively. This notation is advantageous when considering deformations caused by the robot and choosing as reference $\mathbf{x}$ the end-effector position that coincides with the soft material's initial contact position. In that case, $\mathbf{z}$ describes the deformation and $\dot{\mathbf{z}}$ the deformation rate of change. No deformation corresponds to $\mathbf{z} = \mathbf{0}$.

The material's constant, finite stiffness and damping are given by parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}$, which will be identified with the approach described in Sec. V. We employ the mass-spring-damper system defining the resulting force $\mathbf{f}$ as a linear combination of contact deformation $\mathbf{z}$, and it's derivative $\dot{\mathbf{z}}$

$$\mathbf{f} = \boldsymbol{\alpha}\mathbf{z} + \boldsymbol{\beta}\dot{\mathbf{z}} = \mathbf{E}\mathbf{s}_i \tag{3}$$

which can be equivalently formulated with the deformation-dependent state $\mathbf{s} = [\mathbf{z}^T, \dot{\mathbf{z}}^T]^T$ and a constant matrix $\mathbf{E} = [\boldsymbol{\alpha}, \boldsymbol{\beta}]$ representing the material's properties. The continuous-time state-space representation of the linear system dynamics with control input $\mathbf{u}$ writes

$$\dot{\mathbf{s}} = \mathbf{A}_c\mathbf{s} + \mathbf{B}_c\mathbf{u} \tag{4}$$

Given the second-order differential equation $\ddot{\mathbf{z}} = \mathbf{\Lambda}^{-1}\mathbf{f} = \mathbf{\Lambda}^{-1}\boldsymbol{\alpha}\mathbf{z} + \mathbf{\Lambda}^{-1}\boldsymbol{\beta}\dot{\mathbf{z}}$ obtained from (3) by left-multiplication with the inverse of the effective mass[1] $\mathbf{\Lambda}$, results in

$$\mathbf{A}_c = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{\Lambda}^{-1}\boldsymbol{\alpha} & \mathbf{\Lambda}^{-1}\boldsymbol{\beta} \end{bmatrix} \text{ and } \mathbf{B}_c = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \tag{5}$$

Considering a negligible change in the operational apparent mass (see later the effective mass variation curve in Fig. 4), we assume a constant mass $\mathbf{\Lambda}$ for small changes in the robot configuration due to relatively small indentations, and hence, constant $\mathbf{A}_c$.

The discrete-time state-space representation yields

$$\mathbf{s}_{i+1} = \mathbf{A}_d\mathbf{s}_i + \mathbf{B}_d\mathbf{u}_i \tag{6}$$

where subscript i denotes steps at time $t = i\Delta t$ and employing the matrix exponential with sample time Δt

$$\begin{bmatrix} \mathbf{A}_d & \mathbf{B}_d \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \exp\left(\Delta t \begin{bmatrix} \mathbf{A}_c & \mathbf{B}_c \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\right) \tag{7}$$

Because of the spring-damper elements, the control input $\mathbf{u}_i$ does not equal the actual system acceleration. We notice that the discretized acceleration $\ddot{\mathbf{z}}_i$ is linear in terms of state $\mathbf{s}_i$ and control input $\mathbf{u}_i$. Refer to the appendix VII-A for the derivation of matrices $\mathbf{C}$ and $\mathbf{D}$

$$\ddot{\mathbf{z}}_i = \mathbf{C}\mathbf{s}_i + \mathbf{D}\mathbf{u}_i \tag{8}$$

B. Constrained Deformation Dynamics

The above deformation dynamics are constrained when considering a penetration by the robot's end-effector, whose motion is limited by the hardware features. Task-space (or contact-space) bounds are typically configuration-dependent and result from mapping hardware limits, which are provided in joint-space (1b)-(1e). Considering the full six-dimensional task-space associated with the Jacobian $\mathbf{J}$, this mapping procedure is computationally demanding and currently unsuitable for real-time control. As described above, we constrain the end-effector motion to a single axis (the contact normal, e.g., the z-axis parallel to the direction of gravity), simplifying and accelerating the computation.

We are interested in the configuration-dependent task-space bounds with respect to end-effector (= deformation-) velocity, acceleration, and force in contact normal direction:

- The lower bound on the task-space velocity $\underline{\dot{\mathbf{z}}}$ becomes

$$\begin{aligned} \underline{\dot{\mathbf{z}}} = \min_{\dot{\mathbf{q}}} \quad & \mathbf{J}\dot{\mathbf{q}} \\ \text{s. t. } & \mathbf{J}_B\dot{\mathbf{q}} = \mathbf{0} \\ & \underline{\dot{\mathbf{q}}} \le \dot{\mathbf{q}} \le \bar{\dot{\mathbf{q}}} \end{aligned} \tag{9}$$

  The upper velocity bound $\bar{\dot{\mathbf{z}}}$ is obtained by replacing the $\min$ with $\max$ operator. The final deformation velocity constraint yields $\underline{\dot{\mathbf{z}}} \le \dot{\mathbf{z}} \le \bar{\dot{\mathbf{z}}}$.
- The contact-space acceleration and force bounds are coupled and cannot be treated separately. By choosing scalar weights $\omega_{\mathbf{f}}$, $\omega_{\ddot{\mathbf{z}}}$, we can obtain a feasible tuple (or vertex) consisting of an extreme $\ddot{\mathbf{z}}$ and $\mathbf{f}$ through

$$\begin{aligned} \min_{\ddot{\mathbf{q}},\ddot{\mathbf{z}},\mathbf{f}} \quad & \omega_{\mathbf{f}}\mathbf{f} + \omega_{\ddot{\mathbf{z}}}\ddot{\mathbf{z}} \\ \text{s. t. } & \begin{bmatrix} \mathbf{J}_B, & \mathbf{0}, & \mathbf{0} \\ \mathbf{J}, & -\mathbf{I}, & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{z}} \\ \mathbf{f} \end{bmatrix} = \begin{bmatrix} -\dot{\mathbf{J}}_B\dot{\mathbf{q}}_{\text{cur}} \\ -\dot{\mathbf{J}}\dot{\mathbf{q}}_{\text{cur}} \end{bmatrix} \\ & \begin{bmatrix} \underline{\ddot{\mathbf{q}}} \\ \boldsymbol{\tau} - \mathbf{h} \\ \mathbf{0} \end{bmatrix} \le \begin{bmatrix} \mathbf{I}, & \mathbf{0}, & \mathbf{0} \\ \mathbf{M}, & \mathbf{0}, & -\mathbf{J}_A^T \\ \mathbf{0}, & \mathbf{0}, & \mathbf{I} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{z}}_A \\ \mathbf{f}_A \end{bmatrix} \le \begin{bmatrix} \bar{\ddot{\mathbf{q}}} \\ \bar{\boldsymbol{\tau}} - \mathbf{h} \\ \infty \end{bmatrix} \end{aligned} \tag{10}$$

  Solving this linear program multiple times for different weights, we collect a list of extreme vertices, representing a convex polytope. Refer to [30] for an efficient algorithm to select useful weights. The vertices are converted into a halfspace representation given by matrix $\mathbf{P}$ (halfplanes) and vector $\mathbf{p}$ (offset)

$$\mathbf{P}\begin{bmatrix} \ddot{\mathbf{z}} \\ \mathbf{f} \end{bmatrix} \le \mathbf{p} \tag{11}$$

- The contact deformation $\mathbf{z}$ satisfies scalar lower and upper bounds $\underline{\mathbf{z}} \le \mathbf{z} \le \bar{\mathbf{z}}$. These bounds, however, do not relate to the robot's hardware limits. In order not to break the contact and not to damage the material, these bounds are specified by $\mathbf{0}$, and the maximum allowed penetration depending on the soft material properties (elastic domain constraint).

Due to these contact-space constraints, not all possible deformation trajectories are physically achievable with particular robot hardware, which is an often-ignored fact. However, knowledge of the constrained deformation dynamics is of utmost importance for the model-preview proposed in the next section.

IV. Soft Impact-aware Preview Control

In this section, we first formulate a problem for planning an optimized penetration-trajectory assuming the contact event will happen in the next iteration. In the next step, we reformulate this optimization problem to include also the pre-impact end-effector velocity as a decision variable, which is maximized. By tracking this upper bound with a velocity controller, we can intentionally generate feasible and safe impacts with maximum pre-impact velocity.

A. Penetration Trajectory Planning

Let us formulate an optimization problem to plan a feasible and safe soft penetration over a preview horizon of h steps, starting from the exact moment of the contact transition $\mathbf{z}_1 = \mathbf{0}$ and with the pre-impact velocity $\dot{\mathbf{z}}_{\text{imp}}$. The following constrained problem solver outputs a trajectory of optimized control inputs $\mathbf{u}_1, \ldots, \mathbf{u}_h$ and, hence, indirectly through feedforward simulation also the evolution of deformation states $\mathbf{s}_1, \ldots, \mathbf{s}_h$, as well as predicted accelerations $\ddot{\mathbf{z}}_1, \ldots, \ddot{\mathbf{z}}_h$, and contact forces $\mathbf{f}_1, \ldots, \mathbf{f}_h$. Concatenating the decision variables into a vector $\mathbf{U} = \begin{bmatrix}\mathbf{u}_1, \ldots, \mathbf{u}_h\end{bmatrix}^T$, we

[1]Given the one-dimensional Jacobian $\mathbf{J}$ and the joint-space inertia $\mathbf{M}$, the effective mass is computed as $\mathbf{\Lambda} = (\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T)^{-1}$, see [29].

propose the quadratic program:

$$\begin{aligned} \min_{\mathbf{U}} \quad & \frac{1}{2}\mathbf{U}^T\mathbf{W}\mathbf{U} + \mathbf{U}^T\mathbf{w} \qquad (12) \\ \text{subject to} \quad & \mathbf{s}_1 = \begin{bmatrix}\mathbf{0}\\ \dot{\mathbf{z}}_{\text{imp}}\end{bmatrix},\ \mathbf{s}_i = \begin{bmatrix}\mathbf{z}_i\\ \dot{\mathbf{z}}_i\end{bmatrix} \\ & \mathbf{s}_{i+1} = \mathbf{A}_d\mathbf{s}_i + \mathbf{B}_d\mathbf{u}_i \\ & \underline{\mathbf{z}} \le \mathbf{z}_i = [\mathbf{I},\ \mathbf{0}]\,\mathbf{s}_i \le \bar{\mathbf{z}} \\ & \underline{\dot{\mathbf{z}}} \le \dot{\mathbf{z}}_i = [\mathbf{0},\ \mathbf{I}]\,\mathbf{s}_i \le \bar{\dot{\mathbf{z}}} \\ & \mathbf{P}\begin{bmatrix}\ddot{\mathbf{z}}_i\\ \mathbf{f}_i\end{bmatrix} = \mathbf{P}\begin{bmatrix}\mathbf{C}\mathbf{s}_i + \mathbf{D}\mathbf{u}_i\\ \mathbf{E}\mathbf{s}_i\end{bmatrix} \le \mathbf{p} \\ & \dot{\mathbf{z}}_h = \mathbf{0} \end{aligned}$$

The optimization problem consists of multiple constraints:

- The state is composed of contact deformation and its rate of change: $\mathbf{s}_i = [\mathbf{z}_i^T, \dot{\mathbf{z}}_i^T]^T$.
- Deformation dynamics are satisfied: $\mathbf{s}_{i+1}{=}\mathbf{A}_d\mathbf{s}_i{+}\mathbf{B}_d\mathbf{u}_i$.
- Predicted positions $\mathbf{z}_i$, velocities $\dot{\mathbf{z}}_i$, accelerations $\ddot{\mathbf{z}}_i$, and forces $\mathbf{f}_i$ are restricted to a feasible region, defined by the task-space bounds derived in the previous section. There is no need to constrain the control inputs $\mathbf{u}_i$.
- The trajectory terminates safely by enforcing a zero velocity in the terminal state $\dot{\mathbf{z}}_h = \mathbf{0}$.

The COPRA library[2] allows us to formulate (12) as a quadratic problem conveniently. The quadratic cost to be optimized is given by the symmetric and positive-definite matrix $\mathbf{W}$ and a vector $\mathbf{w}$. For example, minimizing the error between the current and desired contact force allows force tracking without force sensor information $\sum_{i=1}^{h} \omega_{\mathbf{f}_i} \left\|\mathbf{f}_{\text{des}} - [\mathbf{E}\ \ \mathbf{0}]\,\mathbf{s}_i\right\|^2$. Another reasonable objective is to minimize the control effort $\sum_{i=1}^{h} \left\|\mathbf{u}_i\right\|^2$.

B. Determining the maximum Pre-Impact Velocity

The optimization problem (12) depends linearly on the pre-impact velocity. Hence, it can be reformulated w.r.t. $\dot{\mathbf{z}}_{\text{imp}}$

$$\begin{aligned} \min_{\mathbf{U}} \quad & \mathbf{U}^T\mathbf{W}\mathbf{U} + \mathbf{U}^T\left[\mathbf{y} + \mathbf{Y}\dot{\mathbf{z}}_{\text{imp}}\right] \qquad (13) \\ \text{subject to} \quad & \mathbf{X}\mathbf{U} = \mathbf{k} - \mathbf{K}\dot{\mathbf{z}}_{\text{imp}} \\ & \mathbf{Z}\mathbf{U} \le \mathbf{v} - \mathbf{V}\dot{\mathbf{z}}_{\text{imp}} \end{aligned}$$

with matrices $\mathbf{X}, \mathbf{Z}, \mathbf{K}, \mathbf{V}$ and vectors $\mathbf{k}, \mathbf{v}$ implementing the constraints in (12) and with $\mathbf{w} = \mathbf{y} + \mathbf{Y}\dot{\mathbf{z}}_{\text{imp}}$ for the cost. However, as we want to explicitly optimize the pre-impact velocity, we need to consider $\dot{\mathbf{z}}_{\text{imp}}$ as an additional decision variable $\tilde{\mathbf{U}} = \left[\dot{\mathbf{z}}_{\text{imp}}^T, \mathbf{U}^T\right]^T$. Reformulating (13) and adding a new objective through matrix $\mathbf{R}$ and vector $\mathbf{r}$ yields (refer to the appendix VII-B for the derivation)

$$\begin{aligned} \min_{\tilde{\mathbf{U}}} \quad & \tilde{\mathbf{U}}^T\tilde{\mathbf{W}}\tilde{\mathbf{U}} + \tilde{\mathbf{U}}^T\tilde{\mathbf{w}} \qquad (14) \\ \text{subject to} \quad & [\mathbf{K}, \mathbf{X}]\,\tilde{\mathbf{U}} = \mathbf{k} \\ & [\mathbf{V}, \mathbf{Z}]\,\tilde{\mathbf{U}} \le \mathbf{v} \end{aligned}$$

with

$$\tilde{\mathbf{W}} = \begin{bmatrix}\mathbf{R}\mathbf{R}^T{+}\mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T, & \mathbf{Y}\\ \mathbf{Y}^T, & \mathbf{W}\end{bmatrix} \text{ and } \tilde{\mathbf{w}} = \begin{bmatrix}\mathbf{r}\\ \mathbf{y}\end{bmatrix}$$

[2]https://github.com/jrl-umi3218/copra

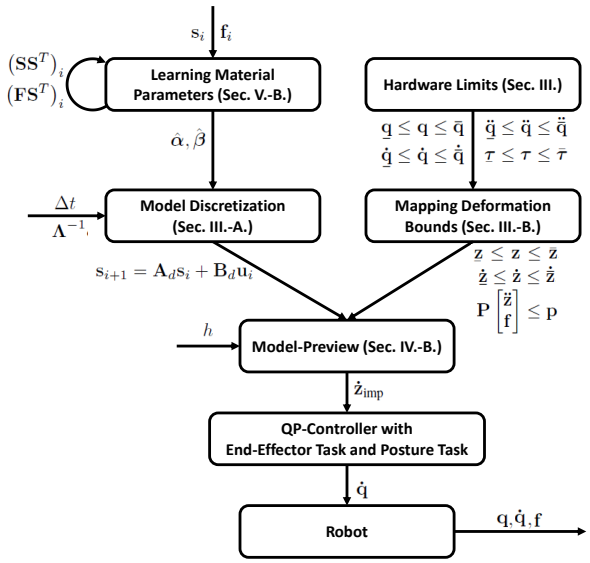

Fig. 2: Illustration of the proposed control scheme (before making contact) for intentional impact maximization.

Note that $\tilde{\mathbf{W}}$ is symmetric and positive-definite for any non-zero matrix $\mathbf{R}$ (with appropriate dimensions, in our case $\mathbf{R}\mathbf{R}^T$ is scalar), as shown in appendix VII-C. We extended the COPRA library to implement the new optimization problem.

Given the maximum feasible pre-impact velocity $\dot{\mathbf{z}}_{\text{imp}}$, there may exist infinite solutions in terms of subsequent control inputs $\mathbf{u}_1, \ldots, \mathbf{u}_h$ that all result in a feasible, safe penetration behavior. In fact, this kind of redundancy is resolved by imposing a secondary objective in the nullspace of the primary objective (pre-impact velocity maximization). In (14), this secondary objective is already given by the previous $\mathbf{W}$, $\mathbf{Y}$, and $\mathbf{y}$ (i.e., minimizing the force error and/or control effort). The primary objective is achieved by selecting a small $\mathbf{R} > 0$ (only required for positive-definiteness of $\tilde{\mathbf{W}}$) and choosing $\mathbf{r} \to -\infty$. It is well-known that a weighted combination of two quadratic costs realizes a nullspace projection if choosing weights with different orders of magnitudes.

We can also ignore the secondary objective and treat (14) as a linear program with the cost $[-1, 0_1, \ldots, 0_h]\,\tilde{\mathbf{U}}$, which is faster to compute. In practice, the concrete choice of control inputs is irrelevant as we are only interested in the maximum pre-impact velocity. Hence, we do not need to impose a specific secondary cost.

C. Final Control Scheme

Without knowledge of contact location or contact time, our idea is to expect the impact event to happen in the next time step, in the direction of end-effector motion. Consequently, we solve (14) in each control cycle to obtain the maximum safe pre-impact velocity. The robot hardware and the task achievement are secured as long as the end-effector motion respects this pre-impact velocity bound. Accordingly, we generate the maximum safe intentional impact by tracking the optimized $\dot{\mathbf{z}}_{\text{imp}}$ with the end-effector.

Note that the mapping between task-space and joint space is assumed locally constant during the preview. This implies that the Jacobian, joint-space inertia, gravity compensation term, and Coriolis forces are assumed with small variation during the penetration process, which is a feasible assumption for small deformations and a short horizon. Further, note that our approach is perfectly suited for position-controlled robots as joint torque measurements are not required.

The final control scheme (c.f Fig 2) requires the material's stiffness and damping. The next section explains how to identify these parameters autonomously. Recursive learning allows us to update the material model during the penetration phase online.

V. Learning Material Deformation Properties

The soft material properties need to be learned (and updated) in a data-driven way unless precise knowledge is apriori available [12]. In order to collect the required data, the robot has to penetrate the deformable contact surfaces. Such exploratory motion can be generated without knowledge of the actual deformation model by tracking a pre-planned deformation trajectory that excites the soft contact properties.

A. Offline-Learning

Given the robot's joint configuration, in each i-th control cycle, the contact state $\mathbf{s}_i$ is recorded via forward kinematics, and the contact force $\mathbf{f}_i$ is obtained through a wrist-mounted force-torque sensor or via external joint torque measurements ($\mathbf{f}_i = \left(\mathbf{J}_i^T\right)^+ \boldsymbol{\tau}_i$). All K recorded data points are concatenated into matrix form

$$\mathbf{S} = \begin{bmatrix}\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_K\end{bmatrix} \text{ and } \mathbf{F} = \begin{bmatrix}\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_K\end{bmatrix} \tag{15}$$

In view of (3), employing linear least-square regression with scalar model error

$$\mathbf{e} = \frac{1}{K}\left\|\mathbf{F}^T - \mathbf{S}^T\left[\boldsymbol{\alpha}, \boldsymbol{\beta}\right]^T\right\|^2 \tag{16}$$

allows us to estimate the material's stiffness and damping $\hat{\mathbf{E}} = \begin{bmatrix}\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\end{bmatrix}$ through

$$\hat{\mathbf{E}} = \begin{bmatrix}\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\end{bmatrix} = \mathbf{F}\mathbf{S}^\dagger = \mathbf{F}\mathbf{S}^T\left(\mathbf{S}\mathbf{S}^T\right)^{-1} \tag{17}$$

where $()^\dagger$ denotes the Moore-Penrose pseudoinverse that minimizes the Euclidean norm (the model error $\mathbf{e}$). A regularization term may be added to ensure numerical stability.

For new states $\mathbf{S}$, the corresponding contact forces $\hat{\mathbf{F}}$ are predicted through simple multiplication with the learned $\hat{\mathbf{E}}$

$$\hat{\mathbf{F}} = \begin{bmatrix}\hat{\mathbf{f}}_1, & \hat{\mathbf{f}}_2, & \ldots, & \hat{\mathbf{f}}_K\end{bmatrix} = \hat{\mathbf{E}}\mathbf{S} \tag{18}$$

We can expect $\mathbf{f} = \hat{\mathbf{f}}$, and hence $\mathbf{F} = \hat{\mathbf{F}}$, when assuming zero sensor noise and the correct choice of $\hat{\mathbf{E}}$ (and, of course, assuming that the linear model represents the reality).

B. Online-Learning via Recursive Formulation

Given a large amount of training data, (17) can only be applied for off-line learning because of the time-consuming matrix inversion. We here extend [12] by employing a recursive formulation that is suitable for model updates in real-time. Given a new data point consisting of $\mathbf{s}_i$ and $\mathbf{f}_i$, we update in each control cycle $i > 0$ the intermediate terms (denoted in subscripts)

$$\left(\mathbf{S}\mathbf{S}^T\right)_i = \left(\mathbf{S}\mathbf{S}^T\right)_{i-1} + \mathbf{s}_i\mathbf{s}_i^T \tag{19}$$

$$\left(\mathbf{F}\mathbf{S}^T\right)_i = \left(\mathbf{F}\mathbf{S}^T\right)_{i-1} + \mathbf{f}_i\mathbf{s}_i^T \tag{20}$$

with initialization $\left(\mathbf{S}\mathbf{S}^T\right)_0 = \mathbf{0}$ and $\left(\mathbf{F}\mathbf{S}^T\right)_0 = \mathbf{0}$. Note that the dimensionality of these intermediate terms is low and stays constant. The material model is improved within the real-time control loop by

$$\left(\hat{\mathbf{E}}\right)_i = \left(\mathbf{F}\mathbf{S}^T\right)_i \left(\mathbf{S}\mathbf{S}^T\right)_i^{-1} \tag{21}$$

These recursive updates are computationally not demanding and yield identical results with (17).

VI. Real-Robot Experiments

Setup: The experimental platform is a 7 degrees-of-freedom Panda manipulator from FrankaEmika, controlled at 1 ms update rate based on our existing QP-control framework [31]. We attached a 3D-printed end-effector with a flat circular contact surface (diameter 0.13 m). The robot's hardware limits are specified on the manufacturer's website[3]. For reproducibility, we ordered a similar dice as chosen by [12] and also selected the vertical axis as the direction of motion, typical for bin-picking scenarios. The soft dice weights 0.1 kg and has 0.16 m edge length. In additional experiments, we attach to the end-effector a pump with two different flexible suction cups (length 0.03 m and 0.05 m) and use various other rigid and soft objects, c.f Fig 1.

Learning Procedure: Material properties are learned by tracking a predefined (sinusoidal) penetration trajectory using position-control and high gains followed-up by five predefined impacts with varying pre-impact velocities (ranging from 0.15 m/s to 0.35 m/s). The final database for the dice contains more than ten seconds of penetration data. Fig 3 plots the deformation states $\mathbf{z}_i$, $\dot{\mathbf{z}}_i$ observed through forward kinematics and associated contact forces $\mathbf{f}_i$ obtained via Panda's built-in force sensor signal. We also identify the soft material parameters for two different flexible suction cups attached to the pump and the soft objects (please refer also to the video `https://youtu.be/juynq6x9JJ8`).

Generating Impact: We compute six vertices/halfplanes for (11) in each iteration employing the algorithm described in [30], which is sufficient to obtain an accurate estimate of the force-acceleration dependency. The optimization problem (14) is parameterized with $h = 20$ steps and the discretization sample time $\Delta t = 25$ ms, resulting in the preview horizon $T = 0.5$ s and 204 inequality constraints. We solve

[3] `https://frankaemika.github.io/docs/control_parameters.html#constants`

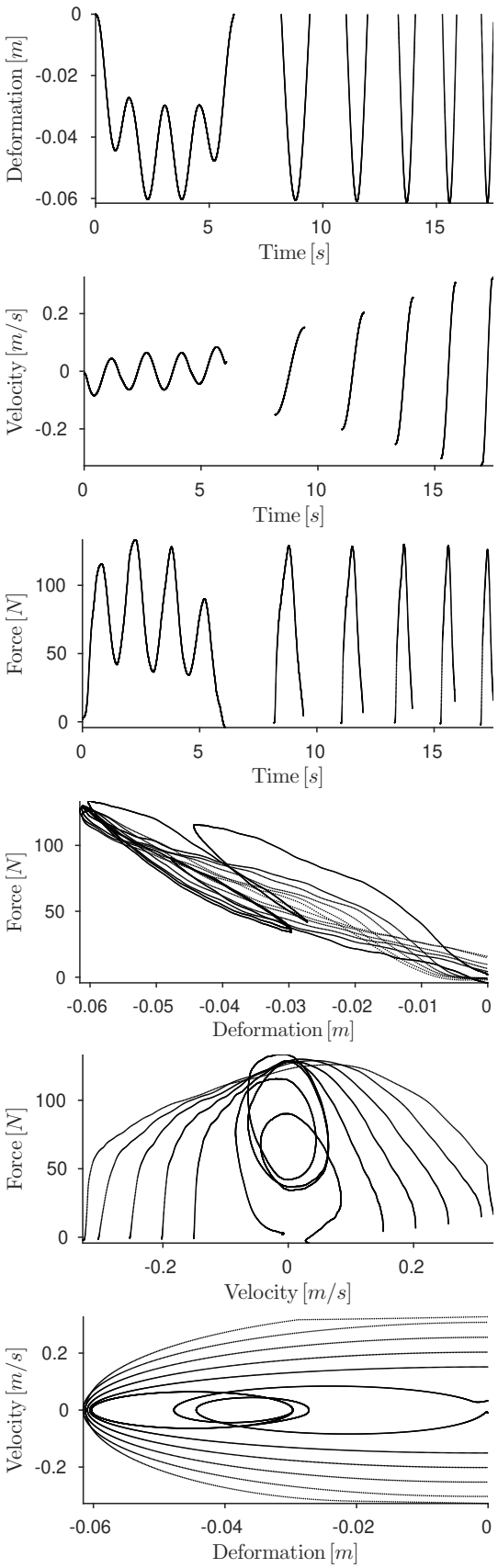

Fig. 3: Recorded data points for *dice* model identification.

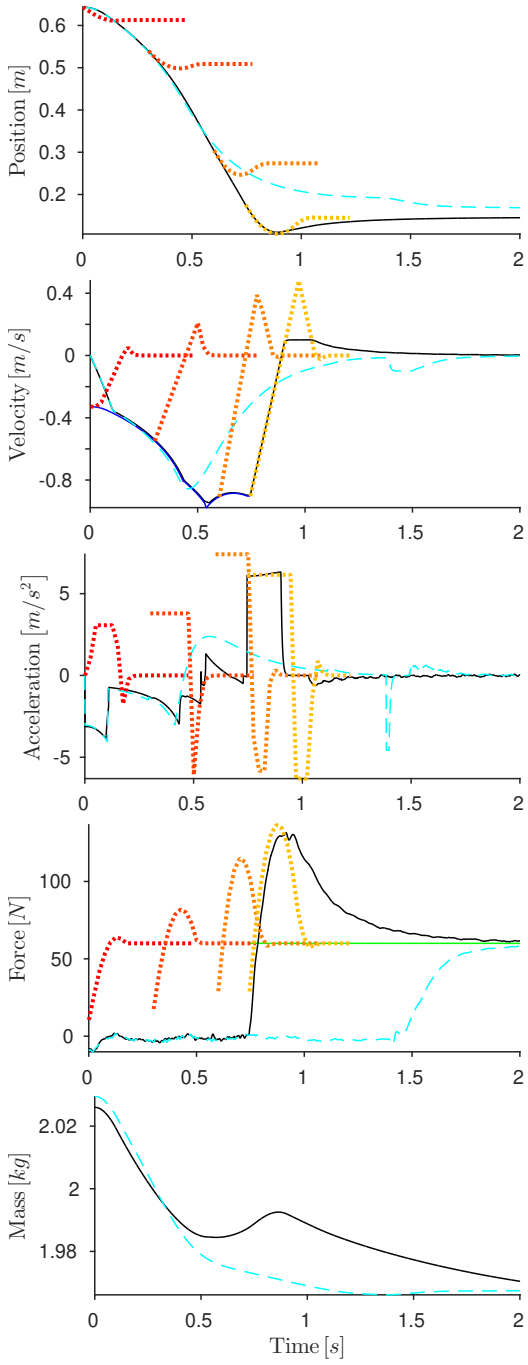

Fig. 4: Intentionally generating the maximum impact for the soft dice employing the proposed impact-aware controller (solid black lines). All values are represented in task-space. The dark-blue curve shows the maximum feasible pre-impact velocity at a given time step that is tracked (before making contact) and green the desired contact force of 60 N (after making contact). Planned trajectories are plotted at 0, 0.3, and 0.6 s and when detecting the contact at 0.75 s with color fading from red into orange (dotted lines). During the whole experiment, the effective mass varies by 0.05 kg. For comparison, the baseline approach with contact event at 1.38 s is shown with dashed cyan lines. The x-axis denotes time in all sub-figures.

it in every control cycle at 1 Khz. These different frequencies are legitimate, as the optimization result constitutes an upper bound on the pre-impact velocity and should be as up-to-date as possible. A primary end-effector velocity task tracks the maximum possible pre-impact velocity. Besides, a secondary task minimizes the joint-accelerations to account for the redundancy, thereby avoiding elbow self-motion, which could affect the end-effector's effective mass (Fig 2). We select 2 N as a threshold for contact detection. Afterward, we switch to an admittance controller [31] that tracks the reference force 60 N. We decided not to track the planned trajectory, as the admittance controller also incorporates the force sensor signal and thereby accounts for material model inaccuracies. More research is needed to incorporate the force measurements into the state space representation (4).

We conduct multiple maximum impact-experiments with various soft objects, starting in different configurations, with varying desired final contact forces. We here discuss only one experiment with the soft dice due to space limitations – please refer to the video for the other experiments. The robot end-effector starts approximately 0.46 m above the contact. Analyzing the logged data, hardware limits are satisfied in all trials. After 0.75 s, we detect the contact with a pre-impact end-effector velocity of 0.898 m/s. Steady-state is reached after approximately 0.8 s of deformation, thereby confirming the chosen preview duration. The final estimated force is close to the desired force. Fig 4 also shows the planned trajectories at a few exemplary iterations (every 0.3 s and at the contact event). As discussed above, these are not necessarily unique: the admittance controller realizes a (slightly) different motion that takes longer to converge.

For baseline comparison, we choose a near-zero contact velocity approach by setting the contact location as a static target. Fig 4 reveals that both controllers accelerate similarly at the beginning of the experiment. As expected, the baseline controller slows down and contacts the surface after 1.38 s. It is important to note that the baseline-admittance task converges faster due to the different initial conditions. However, the exact contact location may not always be known a priori. We conclude from this experiment that contacting a soft material with the maximum feasible impact may not necessarily result in the overall fastest task achievement. Safe, time-optimal planning and control of combined pre-impact *and* post-impact behavior for deformable contacts are left for future work.

VII. Conclusion and Future Work

In order to speed up industrial automation processes that require contact such as grabbing, contact transitions should be accelerated, and hence, contact formations shall be made with impact under robot and task integrity. When evaluating the effect of the pre-impact end-effector velocity for soft contacts, constrained deformation dynamics must be considered, describing the system behavior after making high-velocity contact. This is a significant difference compared to impacts with a rigid material, where we have shown in our previous work that, although conservative, a single-step ahead prediction is sufficient for safe impact-aware control [5].

In this letter, we learn a contact force model for deformable contacts based on exploratory penetration data with impacts that is recursively updated during penetration phases. Next, we propose to map hardware limits onto the contact-space to obtain constrained deformation dynamics. This allows us to plan constrained task-space trajectories for intentional impacts: In the experiments, we maximize pre-impact velocities without compromising the robot limitation while avoiding reaching the plastic domain. Our novel control scheme is independent of whether the soft material is attached to the end-effector (e.g., a deformable suction cup) or part of the environment (e.g., a deformable sponge) and operates at a 1 KHz update rate on the redundant Panda manipulator.

We are currently extending our multi-arm object manipulation approach [32] to cope with multiple soft impacts. Future work will also tackle the planning of pre-impact robot nullspace postures, which can significantly influence the effective mass at the end-effector [29].

APPENDIX

A. Task-space acceleration $\ddot{\mathbf{z}}_i$ depends linearly on $\mathbf{s}_i$ and $\mathbf{u}_i$

Let us denote the matrix elements of matrices $\mathbf{A}_d, \mathbf{B}_d$ as

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{d}_{1,1} & \mathbf{d}_{1,2} \\ \mathbf{d}_{2,1} & \mathbf{d}_{2,2} \end{bmatrix} \text{ and } \mathbf{B}_d = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix}$$

Hence, $\dot{\mathbf{z}}_{i+1} = \mathbf{d}_{2,1}\mathbf{z}_i + \mathbf{d}_{2,2}\dot{\mathbf{z}}_i + \mathbf{e}_2\mathbf{u}_i$. The acceleration $\ddot{\mathbf{z}}_i$ is linear in terms of deformation state $\mathbf{s}_i$ and control input $\mathbf{u}_i$

$$\begin{aligned} \ddot{\mathbf{z}}_i &= \frac{1}{\Delta t}\left(\dot{\mathbf{z}}_{i+1} - \dot{\mathbf{z}}_i\right) = \frac{1}{\Delta t}\left[\mathbf{d}_{2,1}\mathbf{z}_i + \left(\mathbf{d}_{2,2} - 1\right)\dot{\mathbf{z}}_i + \mathbf{e}_2\mathbf{u}_i\right] \\ &= \left[\tfrac{\mathbf{d}_{2,1}}{\Delta t}, \quad \tfrac{\mathbf{d}_{2,2}-1}{\Delta t}\right]\begin{bmatrix}\mathbf{z}_i \\ \dot{\mathbf{z}}_i\end{bmatrix} + \left[\tfrac{\mathbf{e}_2}{\Delta t}\right]\mathbf{u}_i = \mathbf{C}\mathbf{s}_i + \mathbf{D}\mathbf{u}_i \end{aligned}$$

B. Reformulation of quadratic costs

In the following, we reformulate the quadratic cost with (14) and without (12) the velocity $\dot{\mathbf{z}}_{\text{imp}}$ as a decision variable

$$\begin{aligned} &\frac{1}{2}\tilde{\mathbf{U}}^T\begin{bmatrix}\mathbf{R}\mathbf{R}^T + \mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T, & \mathbf{Y} \\ \mathbf{Y}^T, & \mathbf{W}\end{bmatrix}\tilde{\mathbf{U}} + \tilde{\mathbf{U}}^T\begin{bmatrix}\mathbf{r} \\ \mathbf{y}\end{bmatrix} \\ &= \frac{1}{2}\dot{\mathbf{z}}_{\text{imp}}^T\left[\mathbf{R}\mathbf{R}^T + \mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T\right]\dot{\mathbf{z}}_{\text{imp}} + \frac{1}{2}\mathbf{U}^T\mathbf{W}\mathbf{U} + \mathbf{U}^T\mathbf{Y}\dot{\mathbf{z}}_{\text{imp}} \\ &\quad + \dot{\mathbf{z}}_{\text{imp}}^T\mathbf{r} + \mathbf{U}^T\mathbf{y} \\ &= \frac{1}{2}\mathbf{U}^T\mathbf{W}\mathbf{U} + \mathbf{U}^T\left[\mathbf{y} + \mathbf{Y}\dot{\mathbf{z}}_{\text{imp}}\right] \\ &\quad + \frac{1}{2}\dot{\mathbf{z}}_{\text{imp}}^T\left[\mathbf{R}\mathbf{R}^T + \mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T\right]\dot{\mathbf{z}}_{\text{imp}} + \dot{\mathbf{z}}_{\text{imp}}^T\mathbf{r} \end{aligned}$$

where the terms $\frac{1}{2}\dot{\mathbf{z}}_{\text{imp}}^T\left[\mathbf{R}\mathbf{R}^T + \mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T\right]\dot{\mathbf{z}}_{\text{imp}}$ and $\dot{\mathbf{z}}_{\text{imp}}^T\mathbf{r}$ are constant if $\dot{\mathbf{z}}_{\text{imp}}$ is treated constant during optimization.

C. Matrix $\tilde{\mathbf{W}}$ is symmetric and positive-definite

Consider a Cholesky decomposition of matrix $\mathbf{W} = \mathbf{L}\mathbf{L}^T$ where $\mathbf{L}$ is a lower triangular matrix. Accordingly, it yields also $\mathbf{W}^{-1} = (\mathbf{L}^T)^{-1}\mathbf{L}^{-1}$. We here proof that matrix $\tilde{\mathbf{W}}$

defined in (14) is symmetric and positive-definite by showing its Cholesky decomposition for any non-zero matrix $\mathbf{R}$

$$\begin{aligned}\tilde{\mathbf{W}} &= \begin{bmatrix}\mathbf{R}, & \mathbf{Y}(\mathbf{L}^T)^{-1}\\ \mathbf{0}, & \mathbf{L}\end{bmatrix}\begin{bmatrix}\mathbf{R}, & \mathbf{Y}(\mathbf{L}^T)^{-1}\\ \mathbf{0}, & \mathbf{L}\end{bmatrix}^T\\ &= \begin{bmatrix}\mathbf{R}\mathbf{R}^T + \mathbf{Y}(\mathbf{L}^T)^{-1}\mathbf{L}^{-1}\mathbf{Y}^T, & \mathbf{Y}(\mathbf{L}^T)^{-1}\mathbf{L}^T\\ \mathbf{L}\mathbf{L}^{-1}\mathbf{Y}^T, & \mathbf{L}\mathbf{L}^T\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{R}\mathbf{R}^T + \mathbf{Y}\mathbf{W}^{-1}\mathbf{Y}^T, & \mathbf{Y}\\ \mathbf{Y}^T, & \mathbf{W}\end{bmatrix}\end{aligned}$$

ACKNOWLEDGMENT

The authors wish to thank Pierre Gergondet for his technical support and Julien Roux for providing a C++ implementation of the algorithm described in [30].

This work was funded by the H2020 EU project I-AM.

REFERENCES

[1] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, "Analysis and observations from the first amazon picking challenge," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 172–188, 2018.

[2] H. Pham and Q.-C. Pham, "Critically fast pick-and-place with suction cups," in *IEEE Int. Conf. on Robotics and Automation*, 2019, pp. 3045–3051.

[3] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser, "Tossingbot: Learning to throw arbitrary objects with residual physics," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1307–1319, 2020.

[4] Y. Wang and A. Kheddar, "Impact-friendly robust control design with task-space quadratic optimization," in *Proceedings of Robotics: Science and Systems*, 2019.

[5] Y. Wang, N. Dehio, A. Tanguy, and A. Kheddar, "Impact-aware task-space quadratic-programming control," *arxiv*, 2020. [Online]. Available: https://arxiv.org/abs/2006.01987

[6] N. Xydas and I. Kao, "Modeling of contact mechanics and friction limit surfaces for soft fingers in robotics, with experimental results," *Int. Journal of Robotics Research*, vol. 18, no. 9, pp. 941–950, 1999.

[7] G. Gilardi and I. Sharf, "Literature survey of contact dynamics modelling," *Mechanism and Machine Theory*, vol. 37, no. 10, pp. 1213–1239, 2002.

[8] F. Nadon, A. J. Valencia, and P. Payeur, "Multi-modal sensing and robotic manipulation of non-rigid objects: A survey," *Robotics*, vol. 7, no. 4, 2018.

[9] K. Bouyarmane and A. Kheddar, "FEM-based static posture planning for a humanoid robot on deformable contact support," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2011, pp. 487–492.

[10] G. De Magistris, A. Pajon, S. Miossec, and A. Kheddar, "Humanoid walking with compliant soles using a deformation estimator," in *IEEE International Conferance on Robotics and Automation*, 2016, pp. 1757–1762.

[11] E. Coevoet, A. Escande, and C. Duriez, "Soft robots locomotion and manipulation control using fem simulation and quadratic programming," in *Int. Conf. on Soft Robotics*, 2019, pp. 739–745.

[12] M. Azad, V. Ortenzi, H.-C. Lin, E. Rueckert, and M. Mistry, "Model estimation and control of compliant contact normal force," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2016, pp. 442–447.

[13] G. Ganesh, N. Jarrass, S. Haddadin, A. Albu-Schäffer, and E. Burdet, "A versatile biomimetic controller for contact tooling and haptic exploration," in *IEEE Int. Conf. on Robotics and Automation*, 2012, pp. 3329–3334.

[14] S. Caccamo, P. Gler, H. Kjellström, and D. Kragic, "Active perception and modeling of deformable surfaces using gaussian processes and position-based dynamics," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2016, pp. 530–537.

[15] Y. Yoshiyasu, K. Ayusawa, E. Yoshida, Y. Matsumoto, and Y. Endo, "Forward dynamics simulation of human figures on assistive devices using geometric skin deformation model," in *Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, 2015, pp. 2442–2445.

[16] W. L. Xu, J. D. Han, and S. K. Tso, "Experimental study of contact transition control incorporating joint acceleration feedback," *IEEE/ASME Transactions on Mechatronics*, vol. 5, no. 3, pp. 292–301, 2000.

[17] G. Hu, C. Makkar, and W. E. Dixon, "Energy-based nonlinear control of underactuated euler-lagrange systems subject to impacts," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1742–1748, 2007.

[18] C.-H. Liang, S. Bhasin, K. Dupree, and W. E. Dixon, "A force limiting adaptive controller for a robotic system undergoing a noncontact-to-contact transition," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 6, pp. 1330–1341, 2009.

[19] R. Zotovic Stanisic and Á. Valera Fernández, "Adjusting the parameters of the mechanical impedance for velocity, impact and force control," *Robotica*, vol. 30, no. 4, pp. 583–597, 2012.

[20] D. Heck, A. Saccon, N. van de Wouw, and H. Nijmeijer, "Guaranteeing stable tracking of hybrid positionforce trajectories for a robot manipulator interacting with a stiff environment," *Automatica*, vol. 63, pp. 235 – 247, 2016.

[21] V. Samy, K. Bouyarmane, and A. Kheddar, "Analysis of a simple model for post-impact dynamics active compliance in humanoids falls with nonlinear optimization," in *IEEE Int. Conf. on Simulation, Modeling, and Programming for Autonomous Robots*, 2018, pp. 62–67.

[22] M. Azad and M. Mistry, "Balance control strategy for legged robots with compliant contacts," in *IEEE Int. Conf. on Robotics and Automation*, 2015, pp. 4391–4396.

[23] S. Fahmi, M. Focchi, A. Radulescu, G. Fink, V. Barasuol, and C. Semini, "Stance: Locomotion adaptation over soft terrain," *arxiv*, 2019. [Online]. Available: https://arxiv.org/abs/1904.12306

[24] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *Int. Journal of Robotics Research*, vol. 35, no. 12, pp. 1522–1543, 2016.

[25] G. Mesesan, J. Englsberger, G. Garofalo, C. Ott, and A. Albu-Schäffer, "Dynamic walking on compliant and uneven terrain using dcm and passivity-based whole-body control," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2019.

[26] T. Stouraitis, L. Yan, J. Moura, M. Gienger, and S. Vijayakumar, "Multi-modal trajectory optimization for impact-aware manipulation," *arxiv*, 2020. [Online]. Available: https://arxiv.org/abs/2006.13374

[27] R. Orsolino, M. Focchi, C. Mastalli, H. Dai, D. G. Caldwell, and C. Semini, "Application of wrench-based feasibility analysis to the online trajectory optimization of legged robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3363–3370, 2018.

[28] R. Orsolino, M. Focchi, S. Caron, G. Raiola, V. Barasuol, D. G. Caldwell, and C. Semini, "Feasible region: An actuation-aware extension of the support region," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1239–1255, 2020.

[29] N. Mansfeld, B. Djellab, J. R. Veuthey, F. Beck, C. Ott, and S. Haddadin, "Improving the performance of biomechanically safe velocity control for redundant robots through reflected mass minimization," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 5390–5397.

[30] T. Bretl and S. Lall, "Testing static equilibrium for legged robots," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 794–807, 2008.

[31] K. Bouyarmane, K. Chappellet, J. Vaillant, and A. Kheddar, "Quadratic programming for multirobot and task-space force control," *IEEE Transactions on Robotics*, vol. 35, no. 1, pp. 64–77, 2019.

[32] N. Dehio, J. Smith, D. L. Wigand, G. Xin, H.-C. Lin, J. J. Steil, and M. Mistry, "Modeling and Control of Multi-Arm and Multi-Leg Robots: Compensating for Object Dynamics during Grasping," in *IEEE Int. Conf. on Robotics and Automation*, 2018, pp. 294–301.