2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
September 24–28, 2017, Vancouver, BC, Canada

Improving the Performance of Biomechanically Safe Velocity Control for Redundant Robots through Reflected Mass Minimization

Nico Mansfeld, Badis Djellab, Jaime Raldúa Veuthey, Fabian Beck, Christian Ott and Sami Haddadin

Abstract— **Ensuring safety is a primary goal in physical human-robot interaction. In various collision experiments it was found that the robot's effective mass, velocity, and geometry are the key parameters which influence the human injury severity during an impact. Recently, a velocity controller was proposed that limits the robot speed to a biomechanically safe value, taking into account the mass and the curvature in the direction of movement for a given point of interest. The mass and the geometry depend on the mechanical design, however, the effective mass also depends on the robot configuration. In this paper, we exploit the redundant degree(s) of freedom of a joint torque controlled seven- and eight-DOF robot to minimize the effective mass without affecting the desired Cartesian end-effector trajectory and with the goal to improve the performance of the safe velocity controller at the same time. Given recent results in robotics injury analysis, we analyze when such a redundancy resolution scheme actually improves safety. For the considered robots, we find reflected mass extrema that can be obtained by null space motions, and propose a real-time, torque-based redundancy resolution scheme, which is finally verified in experiments.**

I. Introduction

Safety in physical human-robot interaction (pHRI) is an important topic in current robotics research and industry. In recent years many contributions were made to this field. In terms of control, there exist collision detection & reaction schemes [1] as well as pre-collision methods to avoid hazardous impacts. In addition to obstacle avoidance [2], [3], a control scheme referred to as the Safe Motion Unit (SMU) was recently proposed, which ensures that a certain level of injury (e. g., a contusion) is not exceeded upon a dynamic collision between a robot and a human [4].

Many of today's collaborative robots have more degrees of freedom (DOF) than necessary to accomplish a desired end-effector task. This property enhances the robot's dexterity and allows to fulfill secondary tasks that are added to the main task. Besides classical secondary tasks such as singularity or joint limit avoidance, several safety-related schemes have been proposed. These works include, e. g., collision avoidance [5], [6], collision detection and reaction [7], the reduction of the impact force [8], [9], [10], [11], or the amount of dissipated energy in blunt inelastic impacts [12]. In [8], [9], contact models were used to determine the relationship of robot parameters and the resulting collision force. It was stated that a reduction of the reflected mass results in lower force. In robotics injury analysis, however, it was shown that a reduction in reflected mass does not always lead to lower injury probability as a saturation effect in reflected mass may take place [13], [14]. Furthermore, the consistency of injury prediction obtained by contact models was shown to be often insufficient w. r. t. the actual medically observed injury [4].

In this work, we follow the data-driven line of research where robot input parameters (mass, velocity, geometry) are related to injury probability instead of using complex, however insufficient models or intermediate physical quantities like force or stress [4]. We aim for improving the performance of biomechanically safe velocity control, i. e., to avoid a possible velocity reduction by the SMU. For this, we elaborate a redundancy resolution scheme that minimizes the reflected mass of a given point of interest (POI). First, we provide an analysis when such a null space strategy would actually improve safety, given recent results in robotics injury analysis. While previous works on reflected mass minimization considered velocity control, we develop a null space controller for robots with joint torque control, that systematically takes joint position limits into account. Using a seven-DOF KUKA/DLR Lightweight Robot (LWR), we combine the null space strategy with the SMU and provide experimental results that show the effectiveness of our approach. Finally, we extend the idea to an eight-DOF system consisting of a LWR and a linear axis and present first results for this robot.

The remainder of this paper is organized as follows. In Section II, we review the concept of the reflected robot mass, the SMU, discuss when a null space effective mass minimization scheme may improve safe velocity control, and describe our contribution. In Section III we describe the proposed optimization scheme. Experimental results for verifying the controller performance in combination with the SMU are presented in Section IV. The extension of the controller to the eight-DOF system is briefly explained in Section V. Finally, Section VI concludes the paper.

II. Preliminaries & Problem Formulation

In this section, we briefly review the reflected mass model, the SMU, and analyze when a null space strategy for minimizing the reflected mass actually improves collision safety. Then, we describe the considered problem and contribution in detail.

A. Robot Model & Reflected Mass

The considered robot dynamics can be expressed as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\mathrm{ext}}, \quad (1)$$

where $\boldsymbol{q} \in \mathbb{R}^n$ are the generalized link coordinates, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the symmetric, positive definite mass matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ is the Coriolis and centrifugal matrix,

Nico Mansfeld, Badis Djellab, Jaime Raldúa Veuthey, Fabian Beck, and Christian Ott are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Wessling, Germany, Sami Haddadin is with the Institute of Automatic Control, Leibniz Universität Hannover (LUH), Hanover, Germany, nico.mansfeld@dlr.de, sami.haddadin@irt.uni-hannover.de

978-1-5386-2682-5/17/$31.00 ©2017 IEEE

and $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ is the gravity torque vector[1]. The joint torques and the external torques are denoted by $\boldsymbol{\tau} \in \mathbb{R}^n$ and $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$.

The so-called reflected or effective mass is the mass perceived during a collision with the robot [17]. The Jacobian matrix $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times n}$ associated to the impact location can be partitioned as

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_v(\boldsymbol{q}) \\ \boldsymbol{J}_\omega(\boldsymbol{q}) \end{bmatrix}. \tag{2}$$

The translational Cartesian velocity at the impact location is $\boldsymbol{v}(\boldsymbol{q}) = \boldsymbol{J}_v(\boldsymbol{q})\dot{\boldsymbol{q}}$. The velocity in normalized Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$ is

$$v_u(\boldsymbol{q}) = \boldsymbol{u}^\mathsf{T}\boldsymbol{v}(\boldsymbol{q}). \tag{3}$$

The relationship between the manipulator joint-space mass matrix $\boldsymbol{M}(\boldsymbol{q})$ and the Cartesian mass matrix $\boldsymbol{\Lambda}(\boldsymbol{q}) \in \mathbb{R}^{6\times 6}$ was derived in [17] and is well known to be

$$\boldsymbol{\Lambda}(\boldsymbol{q}) = \Big(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}^\mathsf{T}(\boldsymbol{q})\Big)^{-1}. \tag{4}$$

The inverse of the kinetic energy matrix can be decomposed into

$$\boldsymbol{\Lambda}^{-1}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{\Lambda}_v^{-1}(\boldsymbol{q}) & \overline{\boldsymbol{\Lambda}}_{v\omega}(\boldsymbol{q}) \\ \overline{\boldsymbol{\Lambda}}_{v\omega}^\mathsf{T}(\boldsymbol{q}) & \boldsymbol{\Lambda}_\omega^{-1}(\boldsymbol{q}) \end{bmatrix}. \tag{5}$$

The scalar mass perceived at the end-effector given a force in unit direction $\boldsymbol{u}$ is

$$m_u(\boldsymbol{q}) = [\boldsymbol{u}^\mathsf{T}\boldsymbol{\Lambda}_v^{-1}(\boldsymbol{q})\boldsymbol{u}]^{-1}. \tag{6}$$

This quantity is referred to as the reflected mass in direction $\boldsymbol{u}$. For a certain $\boldsymbol{u}$, the reflected mass is influenced by the mass properties of the robot and the current kinematic joint configuration. Possible ways to reduce the reflected mass are a), to reduce the robot weight by design, or b), to modify the joint configuration (if possible). In redundant robots, the configuration for achieving a certain end-effector position is generally not unique. Therefore, one can make use of reconfiguration, i. e., perform self-motions, to minimize the reflected mass in a certain Cartesian direction.

B. When Should the Reflected Mass be Minimized? Influence of Reflected Mass on Injury Severity

A fundamental question that arises with the minimization of the reflected mass is which benefit a reduced mass will have on human injury probability and in which situations one should use a null space minimization scheme to improve collision safety.

The role of mass and velocity on human injury was extensively studied in [13], [14]. For blunt, unconstrained impacts it was shown that a saturation effect in reflected mass takes place when a certain impactor mass is reached. If the robot mass has a considerable weight, then a minimization of the reflected mass will have negligible effect on injury probability. In contrast, the impact velocity always has a significant influence on injury severity. For lower reflected inertias which are inherent to typical pHRI-robots, a change of the reflected mass *does* have an effect on contact force. However, the injury severity is typically low for the considered mass and velocity range during blunt impacts [16]. A minimization of the reflected mass is therefore rather relevant for sharp and edgy contact as shown in [4].

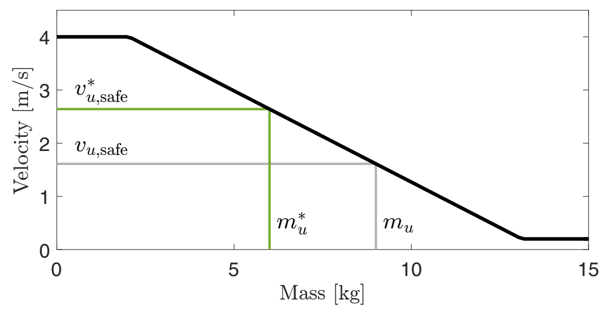


Fig. 1. Effect of mass minimization on biomechanically safe velocity for a spherical impactor with 12.5 mm radius. For the reflected mass m_u the safety curve provides the velocity $v_{u,\text{safe}}$. When reducing the reflected robot mass to m^*_u a larger $v^*_{u,\text{safe}}$ is obtained.

C. Safe Motion Unit

The SMU ensures that a certain level of injury (e. g., a contusion or even no injury at all) will not be exceeded upon a dynamic collision between a robot and a human [4]. For all points of interest along the robot structure (in most applications they are located at the end-effector), the current reflected mass and geometry are related to a biomechanically safe velocity through so-called *safety curves*. A safety curve for a spherical geometry with 12.5 mm radius is exemplarily depicted in Fig. 1. Given the current reflected mass in the Cartesian direction of motion at the point of interest, one can determine the maximal permissible velocity by evaluating the safety curve. The safety curves can be derived from various collision experiments or simulations, e. g., [4], [18], [19]. When activated, the controller monitors the reflected mass and the maximum allowable velocity of each point of interest on the robot structure. If the current task velocity exceeds the (conservative) permissible speed, the controller reduces the velocity such that all safety constraints are met.

D. Contribution

Especially in industrial applications it is desired that the robot moves as fast as possible while still ensuring human safety. The possible velocity reduction imposed by the SMU guarantees safety but deteriorates performance in terms of cycle time. A measure to regain performance, i. e., increase the velocity, is to modify the end-effector geometry and select geometries that allow higher safe velocities given the same reflected mass. Alternatively, the reflected mass in redundant robots may be reduced by suitable use of redundancy.

In this paper, we extend the SMU scheme by developing a null space controller that minimizes the reflected robot inertia without affecting the six-DOF end-effector task in order to further exploit potential performance (speed) increase. Please

[1]The LWR has flexible joint dynamics, where the motor and link side dynamics are coupled via an elastic joint torque. These dynamics are taken into account in the used control framework [15]. For the development of the proposed control scheme, however, the link dynamics is sufficient. Furthermore, in [16] it was shown that for the LWR III (and also other elastic joint robots) the link inertia is decoupled from the motor inertia during collisions, which means that only the link side inertia is required to determine the reflected mass. For sake of clarity, we therefore do not mention the motor side dynamics in this paper.

note that when minimizing the reflected mass, it may be possible that the maximum permissible velocity becomes higher than the desired task speed. In this situation, one can either travel with the nominal velocity or speed up the robot until the maximum feasible velocity of the system is reached. In this paper, we only consider the problem of avoiding a velocity reduction by the SMU. If the SMU does not limit the robot speed, i. e., the mass/velocity pairs are below the safety curve, then mass minimization has no benefit on safety because the motion is regarded as safe already. The mass minimization is therefore only advantageous if the SMU would reduce the desired velocity.

Practically, we mainly consider the seven-DOF LWR III. In Section V we also give first results on the extension to an eight-DOF system consisting of a LWR IV and a linear axis. We assume that the points of interest with sharp or edgy geometry, which will be modeled in the SMU, are located at the end-effector. The robot's links are assumed to be blunt and collisions with these surfaces may be regarded as safe, given the light mass properties and maximum velocity of the robot. The reflected mass minimization shall be added as a redundancy resolution scheme to joint torque control, e. g., the impedance control framework [15]. It may furthermore be integrated into a hierarchical redundancy resolution scheme [20], [21], which is, however, not the scope of this paper. The null space strategy and the SMU shall be usable in a modular fashion. They may either be used independently or simultaneously, while we strive for the latter to maximize performance and ensure safety at the same time.

III. Minimizing Reflected Mass via Self-Motions

In this section, we introduce local, real-time capable optimization methods to minimize the reflected robot inertia.

A. Gradient-Based Minimization

To obtain the null space joint torque that minimizes the reflected inertia, an intuitive idea is to project the gradient of the reflected mass onto the null space of the Jacobian matrix, i. e.,

$$\boldsymbol{\tau}_d = \boldsymbol{\tau}_{\mathrm{prim}} - k\left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\left(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+}\right)^{\mathsf{T}}\right)\nabla m_u(\boldsymbol{q}), \quad (7)$$

where k is a scaling factor, $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ the identity matrix, and $\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+}$ the mass-weighted generalized inverse of the Jacobian matrix to obtain static and dynamic consistency [17]. The output of the primary control law, e. g., a Cartesian impedance controller, is denoted $\boldsymbol{\tau}_{\mathrm{prim}}$. The compensation for gravity is included in this term. The gradient of the reflected mass is given by

$$\nabla m_u(\boldsymbol{q}) = \frac{\partial m_u(\boldsymbol{q})}{\partial \boldsymbol{q}} = \frac{\partial [\boldsymbol{u}^{\mathsf{T}}\boldsymbol{\Lambda}_v^{-1}(\boldsymbol{q})\boldsymbol{u}]^{-1}}{\partial \boldsymbol{q}}. \quad (8)$$

With this approach, that has been used by other authors for velocity control previously, the reflected mass is minimized locally.

Close to extrema the gradient is typically very low and so is the commanded torque. When applying the joint torque controller (7) to the LWR and having a joint configuration that corresponds to an extremum in reflected mass, the robot does not move due to link side friction, even for large k. To systematically overcome such friction effects, we propose a novel method in the following.

B. Attractive Potential for Mass Minimization

In contrast to the previous scheme we propose to use the gradient of an attractive potential that has its maximum at the configuration of minimal reflected mass. It is defined as

$$\boldsymbol{U}(\boldsymbol{q}) = \frac{1}{2}(\boldsymbol{q}^*_{m_u} - \boldsymbol{q})^{\mathsf{T}}\boldsymbol{K}(\boldsymbol{q}^*_{m_u} - \boldsymbol{q}), \quad (9)$$

where $\boldsymbol{q}^*_{m_u}$ is the joint position that corresponds to a minimum in reflected mass. The positive definite, symmetric controller gain matrix is denoted $\boldsymbol{K} = \mathrm{diag}\{k_1, \ldots, k_n\}$. By differentiating (9) w. r. t. the joint position we obtain the desired control torque

$$\boldsymbol{\tau}^*_{m_u} = -\frac{\partial \boldsymbol{U}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \boldsymbol{K}(\boldsymbol{q}^*_{m_u} - \boldsymbol{q}). \quad (10)$$

We project the torque onto the null space of the Jacobian matrix using the mass-weighted pseudoinverse and add it to the output of the primary controller, which yields

$$\boldsymbol{\tau}_d = \boldsymbol{\tau}_{\mathrm{prim}} + \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\left(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+}\right)^{\mathsf{T}}\right)\boldsymbol{\tau}^*_{m_u}. \quad (11)$$

Next, we address the problem of how to obtain the desired position $\boldsymbol{q}^*_{m_u}$. First, we describe how the achievable null space positions can be determined for the LWR. Secondly, we propose an algorithm that finds the desired position efficiently and takes joint position limits into account.

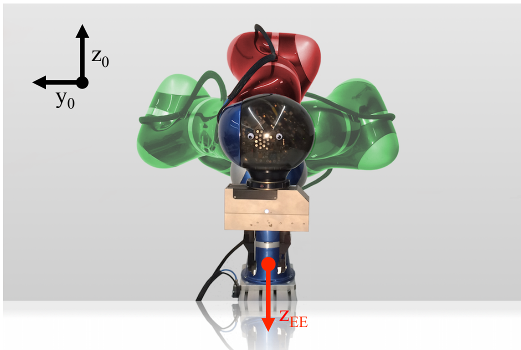
1) Achievable Null Space Joint Positions: Since the task trajectory has six DOF and the LWR has seven DOF, the null space dimension of the Jacobian matrix is one for a non-singular Jacobian matrix. The vector $\boldsymbol{w}(\boldsymbol{q})$, where

$$\boldsymbol{w} : \boldsymbol{q} \mapsto \boldsymbol{\nu} \in \ker(\boldsymbol{J}(\boldsymbol{q})) \subset \mathbb{R}^{n\times 1}, \quad (12)$$

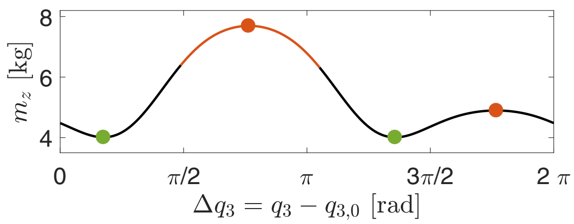
can be regarded as a joint velocity that results in zero operational speed since $\boldsymbol{0} = \boldsymbol{J}(\boldsymbol{q})\boldsymbol{w}(\boldsymbol{q})$. The analytical form of the LWR's kernel was found in [22]. By successively integrating $\boldsymbol{w}(\boldsymbol{q})$ for the initial position $\boldsymbol{q}_{\mathrm{ns}}(0) = \boldsymbol{q}_0$ we obtain all joint positions $\boldsymbol{q}_{\mathrm{ns}}$ that correspond to self-motions (see Fig. 2 (lower)):

$$\boldsymbol{q}_{\mathrm{ns}}(t) = \int_{t_0}^{t} \pm\boldsymbol{w}(\boldsymbol{q}_{\mathrm{ns}}(\tilde{t})) \;+\; \boldsymbol{C}_x\boldsymbol{J}(\boldsymbol{q}_{\mathrm{ns}}(\tilde{t}))^{+}\boldsymbol{e}_x(\boldsymbol{q}_{\mathrm{ns}}(\tilde{t}))\,\mathrm{d}\tilde{t}. \quad (13)$$

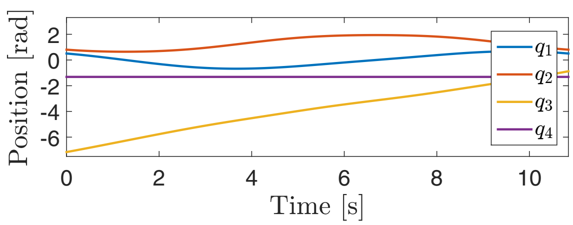
The integration includes a correction term that is based on the end-effector position and orientation error $\boldsymbol{e}_x(\boldsymbol{q}) = \boldsymbol{x}_d(\boldsymbol{q}) - \boldsymbol{x}(\boldsymbol{q})$ between the goal pose and the current pose to compensate for integration drift from a practical point of view [23]. The weighting matrix for the correction term is denoted $\boldsymbol{C}_x$, the Moore-Penrose pseudoinverse of the Jacobian matrix is denoted $\boldsymbol{J}(\boldsymbol{q})^{+}$. We assume that the third entry of $\boldsymbol{w}$ is greater than zero during integration. The resulting null space motion of the LWR, i. e., the rotation of the so-called elbow, is 2π-periodic. The positions $\boldsymbol{q}_{\mathrm{ns}}(t)$ can be associated to reflected masses, given a certain Cartesian direction $\boldsymbol{u}$. In Fig. 2 (lower) it can be observed that q_3 strictly increases over the self-motion. We can therefore use it as a coordinate that represents the self motion, i. e., the elbow rotation. The resulting reflected mass over q_3 plot for a typical pose of the LWR is illustrated in Fig. 2 b). In the figure, it can be observed that the LWR has two minima and two maxima for this end-effector position. For $\boldsymbol{u} = \boldsymbol{z}_{EE} = -\boldsymbol{z}_0$ the minima are the "elbow left" and



(a) Maximum (red) and minima (green) in reflected mass in $\boldsymbol{z}_{EE}$-direction.



(b) Reflected mass in $\boldsymbol{z}_{EE}$-direction. The x-axis represents difference of q_3 w. r. t. its initial value $q_{3,0}$.



(c) Positions of the first four joints.

Fig. 2. Reflected robot mass perceived at the end-effector in $\boldsymbol{z}_{EE}$-direction. The reflected mass over one full elbow rotation, i. e., all possible null space positions, is depicted in b). The red solid line indicates joint positions which are not reachable due to joint limitations. Red and green dots represent maxima and minima in reflected mass. In a), the positions associated to the extrema are applied to the LWR. In the lower figure, the 2π-periodic null space configurations are depicted (joints 5-7 are omitted for sake of readability).

"elbow right" position, the two maxima are the "elbow up" and "elbow down" configuration, respectively. As the LWR has (symmetric) joint limits $|\boldsymbol{q}| \leq \boldsymbol{q}_{\max}$, it is generally possible that not all extrema are reachable. This occurs at $\Delta q_3 \approx 3/4\pi$ rad in Fig. 2 b), for example. Due to the joint position limitations which are represented by the red solid line, the "elbow down" position is not reachable.

2) Finding the Goal Position: For determining the target position $\boldsymbol{q}^*_{m_u}$, we follow the gradient in the mass/q_3 plane until either a minimum in reflected mass or a position constraint is hit. The gradient of the current reflected mass w. r. t. q_3 is obtained by numerical differentiation

$$\left.\frac{\partial m_u(\boldsymbol{q})}{\partial q_3}\right|_{\boldsymbol{q}_{\mathrm{ns}}(t_i)} \approx \frac{m_u(\boldsymbol{q}_{\mathrm{ns}}(t_{i+1})) - m_u(\boldsymbol{q}_{\mathrm{ns}}(t_{i-1}))}{q_3(t_{i+1}) - q_3(t_{i-1})}. \quad (14)$$

For determining the next position in the direction of the gradient descent, we select a Euler forward integration method with constant step size and the same correction term that was used in (13). The integration step is

$$\begin{aligned}\boldsymbol{q}_{\mathrm{ns}}(t_{i+1}) = \boldsymbol{q}_{\mathrm{ns}}(t_i) + &\\ \mathrm{sign}\left(\left.\frac{\partial m_u(\boldsymbol{q})}{\partial q_3}\right|_{\boldsymbol{q}_{\mathrm{ns}}(t_i)}\right) &\frac{\boldsymbol{w}(\boldsymbol{q}_{\mathrm{ns}}(t_i))}{||\boldsymbol{w}(\boldsymbol{q}_{\mathrm{ns}}(t_i))||_2}\Delta t + \\ \boldsymbol{C}_x \boldsymbol{J}(\boldsymbol{q}_{\mathrm{ns}}(t_i))^+ \boldsymbol{e}_x(\boldsymbol{q}_{\mathrm{ns}}(t_i)).& \quad (15)\end{aligned}$$

Here, the gradient of the reflected mass determines the direction of the null space motion, i. e., the velocity $\boldsymbol{w}(\boldsymbol{q})$ given by the kernel of the Jacobian matrix, which is normalized here. The step size is denoted Δt. The third term in (15) corrects the deviation from the desired end-effector pose as in (13).

If no joint position constraints are present, one can simply follow the gradient until the next local minimum is reached. The position bounds, however, necessitate stopping at a limit or even reverse the iteration direction. In our algorithm, we use conservative bounds, i. e., $|\boldsymbol{q}| \leq \boldsymbol{q}_{\mathrm{consv}} < \boldsymbol{q}_{\max}$, to avoid hitting the hardware limits. We illustrate the behavior of the optimization algorithm in the presence of position limits in Fig. 3. There, five cases are depicted that are explained in the following.

1) The current position does not violate bounds, the gradient is followed until the unconstrained local minimum is found.
2) In case the minimum is not reachable, the border of the conservative bounds is selected as the goal position if it is close to a minimum.
3) If the current position violates the conservative joint limits and following the gradient would further violate the constraint, then the direction is reversed and followed until the boundary of the joint limits is reached. This ensures that a reasonable distance to the hardware limits is kept.
4) The current position violates the joint limits but the robot moves from the constraint in direction of the gradient. The gradient is therefore followed until the constraint is not exceeded anymore.
5) In direction of the gradient, the joint limit is hit close to a local maximum. In this case, the algorithm reverses direction to pass the local maximum and move towards the next local minimum.

Please note that the cases mainly show the termination of the optimization process or the behavior at the initial configuration. Several cases may be active during the iteration process until the goal position is reached. Examples:

1) The initial configuration violates the conservative bounds (case 4)). When following the gradient descent, the process terminates at another constraint (case 2)).
2) The initial configuration violates the conservative bounds (case 3)). The direction is reversed until the boundary of the constraint is hit. If this position is close to a local maximum (case 5)), then the iteration continues until the next minimum or constraint is reached.

On our experimental setup, we are able to determine the goal position in real-time, i. e., at 1 kHz control frequency. In order to avoid large step responses in the commanded joint torque, which may occur in case 5), e. g., the number

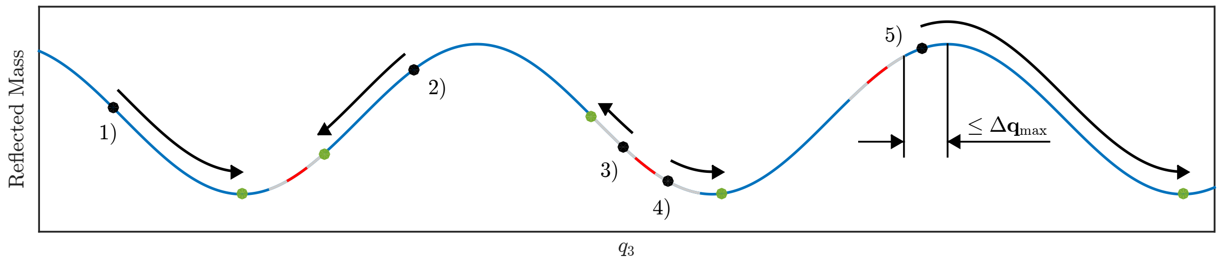


Fig. 3. Finding the desired reflected mass and its associated joint configuration. In the figure, five cases are illustrated that show the behavior of the algorithm. In each case, the black dot indicates the initial reflected mass and the green dot the goal. The conservative position limits are represented by a gray line, the real hardware limits by a red line. All masses/positions on the blue line are reachable without violating constraints. In case 5) the difference $\Delta\boldsymbol{q}_{\max}$ is the difference between the conservative position limits and the local maximum.

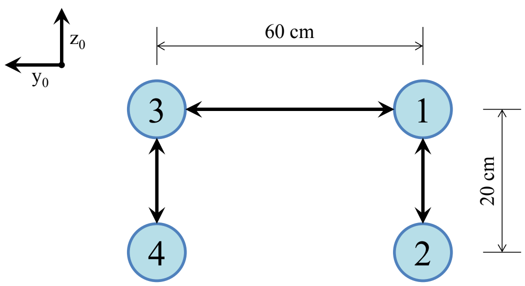


Fig. 4. Schematic of the pick and place task trajectory in the y_0/z_0 plane.

of iterations is limited. For large Δt, chattering may occur as the minimum is generally not found exactly. Therefore, the integration step size must be kept reasonably small. Furthermore, a damping term may be added to (10).

In the next section we show experimental results for a dynamic end-effector trajectory to demonstrate the performance of the proposed controller.

IV. EXPERIMENT

For evaluating the performance of the mass minimization scheme in combination with the SMU, we carry out a pick and place experiment. Although the experiment is relatively simple, the results allow us to determine the benefits and limitations of our approach, as will be discussed later.

The desired Cartesian trajectory is depicted in Fig. 4. The motion sequence is $1 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 1$. The same trajectory was commanded for three control schemes. Firstly, the Cartesian impedance controller was used without SMU or redundancy resolution (see black line in Fig. 5). Secondly, the SMU was activated (blue) and finally SMU & local mass minimization (green).

For the SMU we only assign one point of interest for sake of clarity, namely the end-effector tip. The shape of the closed gripper brackets is very similar to the spherical impactor with 12.5 mm radius, which was analyzed in [4]. Therefore we assign the same safety curve (see Fig. 1) to this POI. The mass/velocity pairs of the LWR III almost always remain below this safety curve. Therefore, the offset of the curve is conservatively shifted such that the effect of the SMU becomes visible. The SMU may reduce the robot speed when the $\boldsymbol{z}_{EE}$-axis of the end-effector (see Fig. 2) is (partly) moving in the direction of travel.

As mentioned previously, the mass minimization is only beneficial when the safety curve would reduce the velocity. However, to evaluate the performance of the mass minimization, it is activated for the entire trajectory. The reflected mass is minimized in the direction of the end-effector movement. For the motions $1 \leftrightarrow 3$ the mass is minimized in $\boldsymbol{y}_{EE} = \boldsymbol{y}_0$-direction, for $1 \leftrightarrow 2$ and $3 \leftrightarrow 4$ in $\boldsymbol{z}_{EE} = -\boldsymbol{z}_0$ direction.

The performance of all controllers can be seen in the attached video. The recorded signals and a classification of the results in terms of safety and performance are depicted in Fig. 5. The results can be interpreted as follows.

Movement $1 \rightarrow 2$*:* In the first movement, the robot is traveling in negative $\boldsymbol{z}_0$-direction. The end-effector $\boldsymbol{z}_{EE}$-axis points in the direction of movement, which is the potentially dangerous direction. The SMU thus reduces the velocity to a biomechanically safe value (see third plot in Fig. 5). The motion with activated SMU is thus slower than the nominal trajectory, however, the reflected mass is not altered. When activating the minimization of the reflected mass (enabled before starting the trajectory), we see that the mass is being reduced in comparison to the original trajectory. Due to lower reflected mass, the safety curve outputs a higher safe velocity. Therefore, the mass minimization allows the robot to travel at the same speed as the nominal trajectory while still ensuring collision safety. As a result, the motion including mass minimization has reached the desired position earlier than the trajectory with SMU only, namely at ≈ 1.2 s instead of ≈ 2.6 s. The robot could now initiate the motion to the next goal position. In Fig. 5, however, this motion is delayed. For better comparison of the results, the motion segments start at the same time for all controllers.

Movement $2 \rightarrow 1$*:* In the second motion segment the robot moves in positive $\boldsymbol{z}_0$-direction. The $\boldsymbol{z}_{EE}$-axis of the end-effector points in the opposite direction of travel. The SMU therefore does not reduce the operational velocity. The velocity is the same for all controllers, while the reflected mass is additionally reduced by the third controller.

Movement $1 \rightarrow 3$*:* The motion from position 1 to 3 is along the $\boldsymbol{y}_0/\boldsymbol{y}_{EE}$-axis only. As in the last movement, the end-effector is not pointing in the direction of travel and therefore the SMU has no influence on the robot speed. Please note that the jump in reflected mass from the second to the third movement is due to the change from $\boldsymbol{u} = \boldsymbol{z}_{EE}$ to $\boldsymbol{u} = \boldsymbol{y}_{EE}$. For the first two movements, the reflected mass with activated minimization was always lower than the mass in the other two experiments. During this movement, it can be observed that between 4.3 s and 4.9 s the minimization scheme outputs a larger reflected mass than the nominal

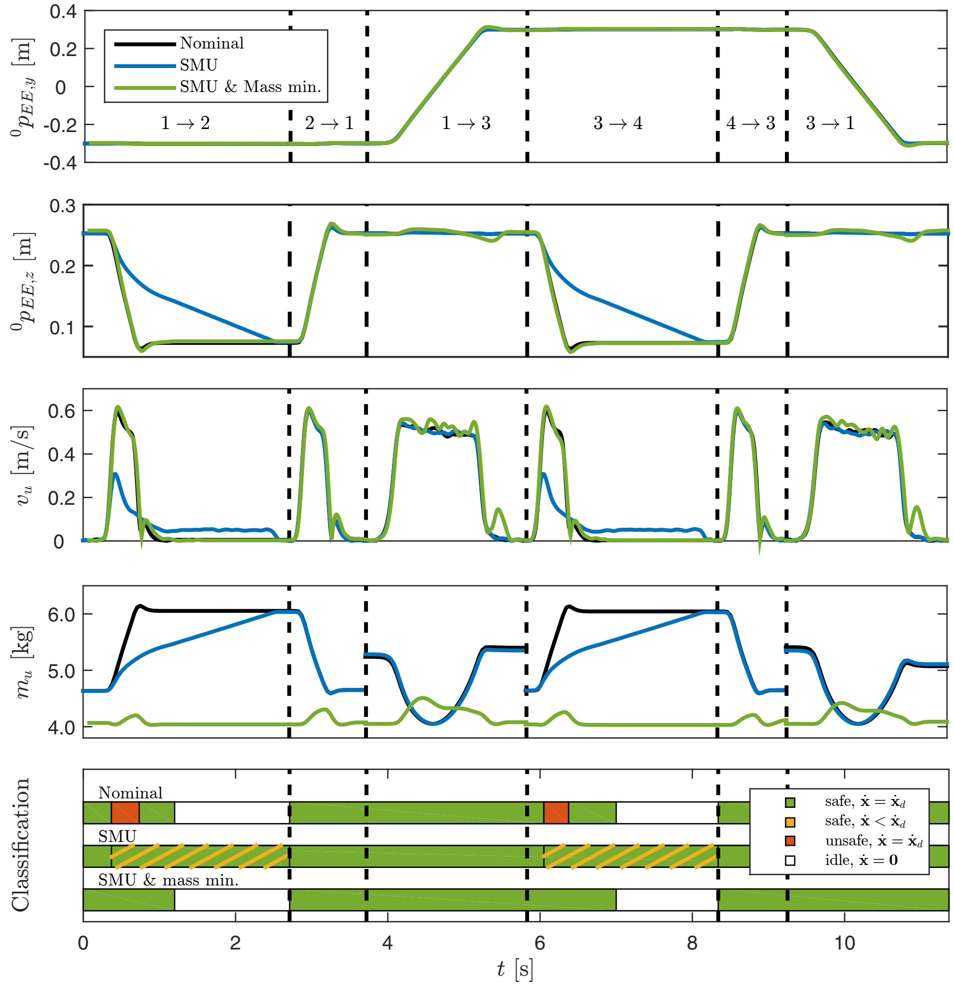


Fig. 5. Performance of SMU and local mass minimization in a pick and place task. The nominal trajectory without SMU or mass minimization is represented by a black line, the result with activated SMU by a blue line, and the trajectory with SMU & mass minimization by a green line. The Cartesian position of the end-effector, denoted ${}^{0}p_{EE,y}$ and ${}^{0}p_{EE,z}$, are depicted in figure one and two, the absolute Cartesian velocity of the end-effector in figure three, and the reflected mass in current $\boldsymbol{u}$-direction figure four. In the top figure, the start and goal position of the each movement are indicated according to Fig. 4. In the bottom figure, a classification of the results in terms of safety and performance is provided. Green segments indicate a safe motion with nominal speed, green/yellow hatched segments a safe motion where the velocity is below the nominal speed, red segments an unsafe motion with nominal speed, and white segments an idle position.

trajectory. This is due to the fact that the initial position in this motion segment is different for both methods. For the trajectory including mass minimization and SMU, the elbow moves from the right to the upper and finally to the left position. For the nominal trajectory the elbow is always in the upper position.

The remaining three movements $3 \rightarrow 4$, $4 \rightarrow 3$, and $3 \rightarrow 1$ are not described in detail because the analysis is identical to the one of the first three motion segments.

Classification of Safety and Performance: In the lower plot in Fig. 5, safety and performance of the three trajectories are classified for this example. In segments $1 \rightarrow 2$ and $3 \rightarrow 4$, the nominal trajectory partially exceeds the biomechanically safe velocity because no safety constraint is taken into account (red). By activating the SMU, we always ensure safety but the performance is decreased, i. e., the robot takes more time to reach the target frame (green/yellow hatched). Finally, the combination of SMU and mass minimization keeps the performance of the original trajectory and ensures safety (green).

A. Discussion

For the considered task, we showed that the proposed method improves the performance of safe velocity control, i. e., the overall cycle time is reduced. However, the method has limitations due its nature of being local.

In the timely evolution of the reflected mass in Fig. 4, one can see at $t \approx 4.5$ s that the optimized reflected mass is higher than the one of the original trajectory. Firstly, this is because the trajectory consists of different motion segments and the elbow position for the nominal trajectory and the one including minimization scheme is different at the initial position of each segment. Secondly, the robot dynamics are limited, which is why the minimum in reflected mass cannot be reached instantaneously. If another POI were present at the end-effector that points in Cartesian $\boldsymbol{y}_0$-direction, the increase in reflected mass *could* result in a reduction of the operational velocity by the SMU.

If only the minimization of the reflected mass was considered without usage of the SMU, one could use a time scaling approach similar to [24] to synchronize the null space strategy with the end-effector trajectory. The difference of the current reflected mass and the next local minimum could serve as a measure to reduce the velocity of the end-effector. The higher the difference in reflected mass, the lower the velocity. This may result in a lower speed, but one would ensure that the minimum in reflected mass is always achieved. However, for sake of clarity and consistency in this paper, we left this for future work.

In the experiment, the changes in $\boldsymbol{u}$ were abrupt, which lead to significant reconfigurations of the LWR's elbow position. In practice, one could try to include transitions such that $\boldsymbol{u}$ changes smoothly and the minimization scheme has more time to reconfigure the elbow. Furthermore, one could use instants where the end-effector manipulates an object or is in a steady state to kinematically reconfigure the elbow such that a minimum in reflected mass is achieved already at the start of the next dynamic movement.

Lastly, if the trajectory was available offline, one could formulate an optimal control problem that, e. g., minimizes the final time while taking the safety curves as inequality constraints into account. However, this solution has limitations as well because it is not sensor-based.

V. Extension to LWR Mounted on Linear Axis

In this section, we provide first results on the extension of the method to an eight-DOF robot. It consists of a LWR which is mounted on a linear axis. The generalized coordinates are $\boldsymbol{q} = [q_x, q_1, q_2, q_3, q_4, q_5, q_6, q_7]^\mathsf{T}$, where q_x is the position of the linear axis and $q_i, i = 1, \ldots, 7$ are the positions of the LWR. For a six-DOF task the robot has two redundant degrees of freedom when the Jacobian matrix is non-singular. Again, we seek for a joint configuration that corresponds to a minimum in reflected mass and does not alter the end-effector task. Having once found the goal configuration, we may apply the control law (10), (11) to the impedance-controlled system [15], [25].

The following description of our approach and the simulation results is kept very brief. More details on the theory and experimental results will follow in a subsequent paper.

In Section III-B, we showed that q_3 may represent the null space elbow rotation, because it (locally) strictly increases/decreases over the 2π-periodic self-motion. Also for this robot, we seek for (two) coordinates, which represent the self-motion and can be understood intuitively. These coordinates shall be the position of the linear axis and the 3rd LWR joint.

From singular value decomposition (SVD) of $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times 8}$ we obtain the orthogonal matrix $\boldsymbol{V}(\boldsymbol{q}) = [\boldsymbol{V}_1(\boldsymbol{q}) \,|\, \boldsymbol{V}_2(\boldsymbol{q})]$, where $\boldsymbol{V}_1(\boldsymbol{q})$ and $\boldsymbol{V}_2(\boldsymbol{q})$ constitute the orthonormal bases for the range space of $\boldsymbol{J}(\boldsymbol{q})^\mathsf{T}$ and the null space of $\boldsymbol{J}(\boldsymbol{q})$, respectively. The matrix $\boldsymbol{V}_2(\boldsymbol{q}) = [\boldsymbol{v}_{21}(\boldsymbol{q}), \boldsymbol{v}_{22}(\boldsymbol{q})]$ contains two null space vectors that are orthogonal w. r. t. each other.

We now seek to decouple the motion of the manipulator and the linear axis as done in [26]. Firstly, we want to find a null space vector $\boldsymbol{w}(\boldsymbol{q}) = [0, *, \ldots, *] \in \ker(\boldsymbol{J}(\boldsymbol{q}))$ where the first entry is zero and the other entries are arbitrary. By projecting any joint velocity onto this part of the null space, we obtain zero velocity in the linear axis, i. e., only a motion of the LWR is considered. The vector $\boldsymbol{w}(\boldsymbol{q})$ can be calculated by a linear combination of $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$, namely

$$\boldsymbol{w}(\boldsymbol{q}) = \boldsymbol{v}_{21}(\boldsymbol{q}) - \frac{v_{21,1}(\boldsymbol{q})}{v_{22,1}(\boldsymbol{q})} \boldsymbol{v}_{22}(\boldsymbol{q}), \tag{16}$$

where $v_{21,1}(\boldsymbol{q})$ and $v_{22,1}(\boldsymbol{q})$ are the first elements of $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$, respectively. If $v_{22,1}(\boldsymbol{q}) = 0$, then we may select $\boldsymbol{w}(\boldsymbol{q}) = \boldsymbol{v}_{22}(\boldsymbol{q})$. The second null space vector $\boldsymbol{w}_\perp(\boldsymbol{q})$ shall be independent from $\boldsymbol{w}(\boldsymbol{q})$, i. e., orthogonal. To obtain $\boldsymbol{w}_\perp(\boldsymbol{q})$, we use the Gram-Schmidt process to orthogonalize $\boldsymbol{w}(\boldsymbol{q})$ and either $\boldsymbol{v}_{21}(\boldsymbol{q})$ or $\boldsymbol{v}_{22}(\boldsymbol{q})$. While a velocity that is projected onto $\boldsymbol{w}(\boldsymbol{q})$ halts the linear axis, the projection onto $\boldsymbol{w}_\perp(\boldsymbol{q})$ maximizes its velocity. To sum up, we obtained two vectors which are orthogonal w. r. t. each other and span the entire null space of the Jacobian matrix like $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$.

We can now systematically integrate $\boldsymbol{w}(\boldsymbol{q})$ and $\boldsymbol{w}_\perp(\boldsymbol{q})$ and use q_x and q_3 to represent the self-motions. In Fig. 6 we display the reflected robot mass in Cartesian y_0-direction over q_x and q_3. All points illustrated in the q_x/q_3 plane correspond to configurations that do not alter the end-effector position. Using this grid, we may calculate the gradient descent of the reflected mass and determine the next local minimum. This procedure was applied in iterative form to minimize the reflected mass for a dynamic trajectory. The initial configuration was similar to the one depicted in Fig. 2 a) (red configuration). While keeping the same orientation and Cartesian x_0- and z_0-position, the end-effector moves 1.5 m in Cartesian y_0-direction. The timely evolution of the reflected mass with and without reflected mass minimization is depicted in Fig. 6. The results indicate that the reflected mass is minimized effectively and our approach is promising also for this system. The simulated motion is shown in the attached video. In the video, also the controller behavior in the presence of external joint torques is shown for the real system.

VI. Conclusion

In this work, we considered the problem of minimizing the reflected robot mass by exploiting redundancy of a LWR manipulator and combining this redundancy resolution joint torque control scheme with the Safe Motion Unit. The minima and maxima in reflected mass were determined for static end-effector poses of the LWR. Then, a real-time capable local minimization scheme was proposed that provides a desired joint torque based on an attractive field spanned between the current joint position and the position associated to the next local minimum in reflected mass. The controller performance was verified experimentally. Using the experimental results, we highlighted the advantages and limitations of our approach. Finally, the method was extended to a system that has two redundant DOF.

In the experimental part, we modeled only one POI on the robot structure so far. In future work we will consider several POI and integrate the SMU with the mass minimization more tightly. Also, the benefit of both the SMU and mass minimization on collision safety will be verified by conducting suitable collision experiments. Furthermore, we will work on the generalization of the optimization scheme to systems with more DOF and integrate it into a hierarchical

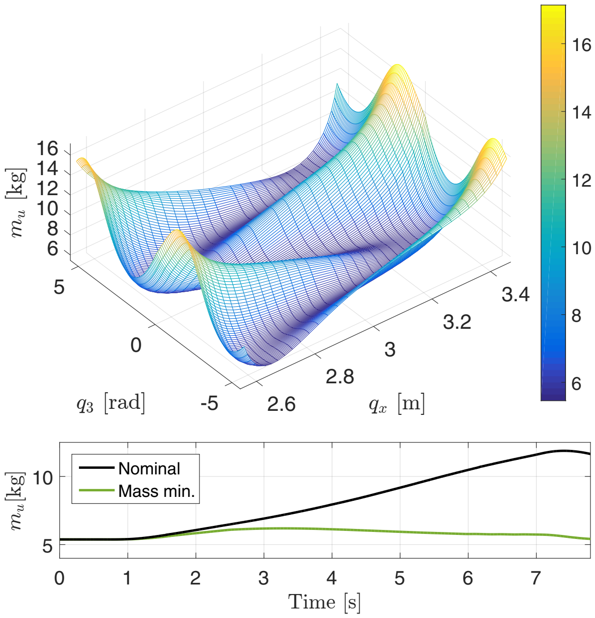


Fig. 6. Reflected mass in Cartesian y_0-direction of the eight-DOF system over the position of the linear axis and the third LWR joint (upper). Here, a configuration similar to the one depicted in Fig. 2 a) (red configuration) was selected. The timely evolution of reflected mass for a movement in Cartesian y_0-direction is depicted in the lower figure. The black and green line represent the reflected mass without and with enabled minimization scheme, respectively.

redundancy resolution scheme that also considers singularity avoidance, for example.

Acknowledgment

This work has been partially funded by the European Commission's Seventh Framework Programme as part of the project EuRoC under grant no. 608849 and the Alfried Krupp von Bohlen und Halbach Foundation. We would like to thank André Hansen for the illustration in Fig. 2.

References

[1] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2006)*. IEEE, 2006, pp. 1623–1630.

[2] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.

[3] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 874–883, 2007.

[4] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "On making robots understand safety: Embedding injury knowledge into control," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1578–1602, 2012.

[5] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *The international journal of robotics research*, vol. 4, no. 3, pp. 109–117, 1985.

[6] M. P. Polverini, A. M. Zanchettin, and P. Rocco, "Real-time collision avoidance in human-robot interaction based on kinetostatic safety field," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*. IEEE, 2014, pp. 4136–4141.

[7] A. De Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2008)*. IEEE, 2008, pp. 3299–3305.

[8] I. D. Walker, "The use of kinematic redundancy in reducing impact and contact effects in manipulation," in *IEEE International Conference on Robotics and Automation (ICRA1990)*. IEEE, 1990, pp. 434–439.

[9] M. W. Gertz, J.-O. Kim, and P. K. Khosla, "Exploiting redundancy to reduce impact force," in *IEEE/RSJ International Workshop on Intelligent Robots and Systems*. IEEE, 1991, pp. 179–184.

[10] I. D. Walker, "Impact configurations and measures for kinematically redundant and multiple armed robot systems," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 5, pp. 670–683, 1994.

[11] J. Kim, W. Chung, and Y. Youm, "Normalized impact geometry and performance index for redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA2000)*. IEEE, 2000, pp. 1714–1719.

[12] R. Rossi, M. P. Polverini, A. M. Zanchettin, and P. Rocco, "A pre-collision control strategy for human-robot interaction based on dissipated energy in potential inelastic impacts," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2015)*. IEEE, 2015, pp. 26–31.

[13] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "The role of the robot mass and velocity in physical human-robot interaction - part I: Unconstrained blunt impacts," in *IEEE International Conference on Robotics and Automation (ICRA2008), Pasadena, USA*, 2008, pp. 1331–1338.

[14] ——, "The role of the robot mass and velocity in physical human-robot interaction - part II: Constrained blunt impacts," in *IEEE International Conference on Robotics and Automation (ICRA2008), Pasadena, USA*, 2008, pp. 1339–1345.

[15] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *The International Journal of Robotics Research*, vol. 26, no. 1, pp. 23–39, 2007.

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

[17] O. Khatib, "Inertial properties in robotic manipulation: an object-level framework," *Int. Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995.

[18] R. Behrens and N. Elkmann, "Study on meaningful and verified thresholds for minimizing the consequences of human-robot collisions," in *IEEE International Conference on Robotics and Automation (ICRA2014)*. IEEE, 2014, pp. 3378–3383.

[19] J.-J. Park, J.-B. Song, and S. Haddadin, "Collision analysis and safety evaluation using a collision model for the frontal robot–human impact," *Robotica*, vol. 33, no. 07, pp. 1536–1550, 2015.

[20] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 3–15, 1987.

[21] B. Siciliano and J.-J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Fifth International Conference on Robots in Unstructured Environments (ICAR1991)*. IEEE, 1991, pp. 1211–1216.

[22] C. Ott, *Cartesian impedance control of redundant and flexible-joint robots*. Springer, 2008.

[23] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *The International Journal of Robotics Research*, vol. 10, no. 4, pp. 410–425, 1991.

[24] S. Haddadin, A. Albu-Schäffer, A. D. Luca, and G. Hirzinger, "Collision detection & reaction: A contribution to safe physical human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2008), Nice, France*, 2008, pp. 3356–3363.

[25] A. Dietrich, K. Bussmann, F. Petit, P. Kotyczka, C. Ott, B. Lohmann, and A. Albu-Schäffer, "Whole-body impedance control of wheeled mobile manipulators," *Autonomous Robots*, vol. 40, no. 3, pp. 505–517, 2016.

[26] N. Mansfeld, F. Beck, A. Dietrich, and S. Haddadin, "Interactive, intuitively interpretable null space motions for redundant mobile manipulators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2017)*. IEEE, 2017.