The Role of Robot Payload in the Safety Map Framework

Mazin Hamad, Nico Mansfeld, Saeed Abdolshah and Sami Haddadin

***Abstract*— In practical robotic applications various types of tools are attached for manipulating objects. Besides adding gravitational load to the robot, which results in larger joint torques, such payloads influence the collision safety characteristics through changing surface curvature properties, reflected mass and effective robot speed along a desired motion direction. In this paper, we evaluate the effect a known, unactuated payload that is attached to the end-effector has on the intantaneous reflected inertial parameters and maximum task velocity of a robot. The proposed mass update approach relies on the analysis of the kinetic energy matrices, while the velocity maximization is tackled by formulating static optimization problems with different constraints on angular motion of the end-effector. Finally, for analyzing the validity of the introduced approach in the framework of Safety Maps, we discuss simulation results of a PUMA 560 robot that has an exemplary payload attached to its end-effector.**

I. Introduction

Physical contact between human and robot may lead to unforeseen and potentially dangerous collisions. Since safety is a key requirement for close physical cooperation, it is crucial to study all machinery aspects that affect the interaction dynamics between humans and robots. In particular, the influence of contact parameters on the human perceived pain and injury severity deserves thorough investigation.

State of the art: In order to mechanically minimize the risk of human injury, even during unforeseeable collisions the velocity shaping controller *Safe Motion Unit* (*SMU*) scales task velocity based on encoded injury biomechanics data [1]. Relying on such a human injury database and extending the *SMU* concept, the *Safety Map* was introduced as a tool for global robot safety evaluation that may serve safer robot design and motion planning [2].

In earlier works, various contact models were used to determine the influence of the robot parameters on resulting collision forces [3], [4]. In this context, minimizing the ratio between manipulability and reflected robot mass has been proposed as an advisable design strategy [5].

Exploiting the redundant degree(s) of freedom, it is possible to minimize the reflected mass without necessarily affecting the desired Cartesian end-effector trajectory [6]. Consequently, it is possible to allow higher task execution velocities while simultaneously respecting safety constraints imposed by the *SMU*.

Mazin Hamad, Saeed Abdolshah, and Sami Haddadin are with the Institute of Robotics and Systems Intelligence, Munich School of Robotics and Machine Intelligence, Technical University of Munich (TUM), Germany. Nico Mansfeld is with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Wessling, Germany, mazin.hamad@tum.de, sami.haddadin@tum.de

Contribution: In this work, we extend the data-driven approach for biomechanically safe velocity control we originally introduced in [1] and the *Safety Map* [7] by considering an additional robot payload. Specifically, we aim for extending the *Safety Map* concept, which is calculated once for a given robot design to handle additional payloads with varying geometry and without the need to recalulate the entire map. For this, we incorporate the payload inertial and geometrical properties into those of the robot by appropriately updating the relevant kinetic energy matrices and Jacobians. When changing the end-effector or grasping an object the inertial properties and kinematics of a robot change. This means that the dynamics of the new system comprised of the robot and the additional payload embodied into its physical structure will be different from the payload-free robot dynamics. As a result, the required information to assess human safety during collisions will have to be recalculated. This can be solved analytically for the reflected mass. In contrast, the corresponding maximum Cartesian velocity is usually not available analytically, and therefore, suitable optimization schemes become necessary. In this paper, we extend our model for capturing human-robot contact dynamics [1] in terms of mass-velocity-curvature effector relations by

1) formulating to update robot inertia properties upon attaching a rigid payload to the end-effector,
2) inferring special properties of payloads that allow quick reuse and extension/update of payload-free system dynamics, and
3) formulating different operational velocity maximization problems for different angular motion conditions for the payload-extended kinematics.

The remainder of the paper is organized as follows. In Section II, we describe the collision model. In Section III, we extend this collision model to include the kinetic energy matrix of a static payload and show how the reflected mass can be updated accordingly. A detailed discussion on different techniques for calculating the resulting maximum robot velocity with an additional payload is then provided in Section IV. Simulation results in Section V verify the algorithms for calculating the updated reflected mass and solving velocity optimization problems within the *Safety Map* framework. Finally, Section VI concludes the paper.

II. PRELIMINARIES

In this section, the instantaneous human-robot contact model and its associated key parameters are shortly reviewed. For this, the end-effector payload and its effect on reflected mass and velocity of the payload-extended robot is considered.

A. Payload-free model

As depicted in Fig. 1 the considered rigid model follows our approach from [1] and [7]. We treat the robot end-effector as an intrinsic payload attached to the flange and the tool to be used or the product being picked as an additional spatially relevant inertial object.

For the collision analysis, the robot is modeled in terms of its configuration dependent scalar reflected mass $m_r(\boldsymbol{q})$ and velocity $\dot{x}_r(\boldsymbol{q})$, as well as its local point of interest (POI), impact curvature c_r and elastic surface properties EP_r [1]. The mass m_r is perceived – in this case by the human – along a chosen Cartesian collision unit vector $\boldsymbol{u}$ [8]. The human may also be represented within the same framework by its reflected impact mass m_h, velocity $\dot{x}_h$, and impact location BP_h on the involved body part along the collision direction. We further assume that the impact direction of motion coincides with the surface normal of the respective body part. This is in alignment with the experimental design in essentially all relevant biomechanics and robotics publications. The absolute relative velocity between human and robot is then given by $\dot{x}_{\text{rel}} = |\dot{x}_r - \dot{x}_h|$.

In the following we drive the model starting from the robot equation of motion. The rigid body dynamics of a robot manipulator is well-known to be

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

where $\boldsymbol{q} \in \mathbb{R}^n$ denotes the generalized robot coordinates, $\boldsymbol{M}_r(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ the symmetric, positive definite mass matrix, $\boldsymbol{C}_r(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ the Coriolis and centrifugal matrix, and $\boldsymbol{g}_r(\boldsymbol{q}) \in \mathbb{R}^n$ the gravity torque vector. The motor joint torque is denoted $\boldsymbol{\tau} \in \mathbb{R}^n$ and the external joint torques $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$. For sake of clarity, we omit any non-conservative force vectors such as the friction torques $\boldsymbol{\tau}_f$.
The robot's Cartesian mass matrix is $\boldsymbol{\Lambda}_r(\boldsymbol{q}) \in \mathbb{R}^{6\times 6}$. Its inverse is called pseudo kinetic energy matrix [8] and can be related to the robot joint space mass matrix by

$$\boldsymbol{\Lambda}_r^{-1}(\boldsymbol{q}) = \boldsymbol{J}_r(\boldsymbol{q})\boldsymbol{M}_r^{-1}(\boldsymbol{q})\boldsymbol{J}_r^{\mathsf{T}}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{\Lambda}_{r,v}^{-1}(\boldsymbol{q}) & \overline{\boldsymbol{\Lambda}}_{r,v\omega}(\boldsymbol{q}) \\ \overline{\boldsymbol{\Lambda}}_{r,v\omega}^{\mathsf{T}}(\boldsymbol{q}) & \boldsymbol{\Lambda}_{r,\omega}^{-1}(\boldsymbol{q}) \end{bmatrix}. \tag{2}$$

The Jacobian matrix associated with the impact location is $\boldsymbol{J}_r(\boldsymbol{q}) = [\boldsymbol{J}_{r,v}^{\mathsf{T}}(\boldsymbol{q}), \boldsymbol{J}_{r,\omega}^{\mathsf{T}}(\boldsymbol{q})]^{\mathsf{T}} \in \mathbb{R}^{6\times n}$, with $\boldsymbol{J}_{r,v}(\boldsymbol{q})$ and $\boldsymbol{J}_{r,\omega}(\boldsymbol{q})$ being the Jacobians for translational and angular motions, respectively. The so-called robot reflected mass $m_{r,u}(\boldsymbol{q}) \in \mathbb{R}$ and velocity $v_{r,u}(\boldsymbol{q}) \in \mathbb{R}$ in normalized Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$ are

$$\begin{aligned} m_{r,u}(\boldsymbol{q}) &= \left[\boldsymbol{u}^{\mathsf{T}}\boldsymbol{\Lambda}_{r,v}^{-1}(\boldsymbol{q})\boldsymbol{u}\right]^{-1} \\ v_{r,u}(\boldsymbol{q}) &= \boldsymbol{u}^{\mathsf{T}}\boldsymbol{J}_{r,v}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{u}^{\mathsf{T}}\boldsymbol{v}_r(\boldsymbol{q}). \end{aligned} \tag{3}$$

$\boldsymbol{\Lambda}_{r,v}^{-1}(\boldsymbol{q})$, $\boldsymbol{\Lambda}_{r,\omega}^{-1}(\boldsymbol{q})$ and $\overline{\boldsymbol{\Lambda}}_{r,v\omega}^{\mathsf{T}}(\boldsymbol{q}) \in \mathbb{R}^{3\times 3}$ are the translational, rotational and coupling pseudo kinetic energy matrix, respectively.

The goal is now to update the robot dynamic capabilities in terms of reflected mass and velocity, i. e. the coordinates of the *Safety Map*, to consider additional spatially relevant rigid payloads. These capabilities are usually generated under considerations of relevant POIs along the robot body or its end-effector, yet typically without any considerations for any extra payload. However, an additional payload may introduce new POIs, at which the reflected mass and maximum Cartesian velocity need to be evaluated along the chosen motion direction $\boldsymbol{u}$. We aim to efficiently solve this problem for the payload-extended robot by developing update laws from the payload-free robot dynamic capabilities that are computationally more efficient than recalculating from scratch.

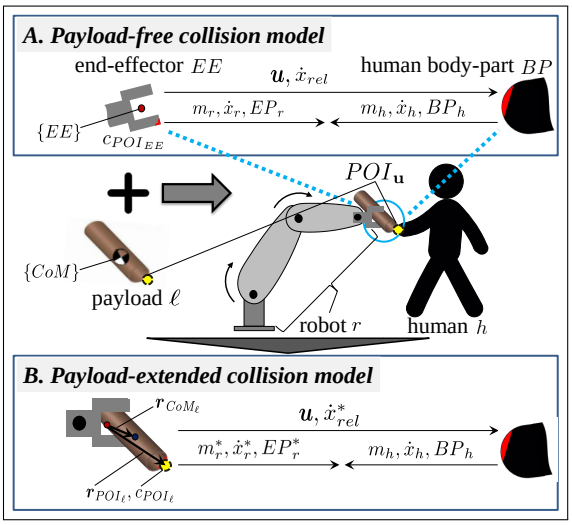

Fig. 1. Extending the payload-free model for representing the instantaneous contact dynamic properties of the impactor and human under collision.

B. Payload-extended model

Assume that a rigid payload of mass m_ℓ and inertia tensor ${}^{CoM}\boldsymbol{\mathcal{I}}_\ell$ is attached to the operational point of the robot end-effector at a point with Cartesian position vector $\boldsymbol{r}_{CoM_\ell} = [r_1, r_2, r_3]^{\mathsf{T}}$ from $\{EE\}$ to its center of mass *CoM* (with its cross-product operator matrix as $\hat{\boldsymbol{r}}_{CoM_\ell} = [0, -r_3, r_2; r_3, 0, -r_1; -r_2, r_1, 0]$). The payload inertia matrix expressed in $\{CoM\}$ is

$${}^{CoM}\boldsymbol{M}_\ell = \begin{bmatrix} m_\ell \mathbf{I} & \mathbf{0} \\ \mathbf{0} & {}^{CoM}\boldsymbol{\mathcal{I}}_\ell \end{bmatrix}, \tag{4}$$

where $\mathbf{I}$ and $\mathbf{0}$ are the identity and zero matrices of appropriate dimension. With ${}^{CoM}\boldsymbol{v}_\ell$ and ${}^{CoM}\boldsymbol{\omega}_\ell$ denoting the translational and angular velocities at *CoM*, the end-effector velocities become

$$\begin{bmatrix} \boldsymbol{v}_r \\ \boldsymbol{\omega}_r \end{bmatrix} = {}^{EE}\boldsymbol{J}_{\ell CoM}\, {}^{CoM}\begin{bmatrix} \boldsymbol{v}_\ell \\ \boldsymbol{\omega}_\ell \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \hat{\boldsymbol{r}}_{CoM_\ell} \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} {}^{CoM}\begin{bmatrix} \boldsymbol{v}_\ell \\ \boldsymbol{\omega}_\ell \end{bmatrix}. \tag{5}$$

Then, the payload kinetic energy matrix can be expressed as

$$\begin{aligned} \boldsymbol{\Lambda}_\ell &= {}^{EE}\boldsymbol{J}_{\ell\, CoM}^{-\mathsf{T}}\, {}^{CoM}\boldsymbol{M}_\ell\, {}^{EE}\boldsymbol{J}_{\ell\, CoM}^{-1} \\ &= \begin{bmatrix} m_\ell \boldsymbol{I} & -m_\ell \hat{\boldsymbol{r}}_{CoM_\ell} \\ -m_\ell \hat{\boldsymbol{r}}_{CoM_\ell}^{\mathsf{T}} & {}^{CoM}\boldsymbol{\mathcal{I}}_\ell + m_\ell \hat{\boldsymbol{r}}_{CoM_\ell}^{\mathsf{T}} \hat{\boldsymbol{r}}_{CoM_\ell} \end{bmatrix}. \end{aligned} \tag{6}$$

In analogy to (2), the payload-extended joint space mass matrix becomes

$$\boldsymbol{M}_\ell(\boldsymbol{q}) = \boldsymbol{J}_r^{\mathsf{T}}(\boldsymbol{q})\boldsymbol{\Lambda}_\ell \boldsymbol{J}_r(\boldsymbol{q}). \tag{7}$$

Reflected Mass of Payload-extended Robot at POI: Since the reflected mass calculated along a given direction at any payload POI is the same as at the end-effector due to being one rigid body, it is sufficient to calculate it once at $\{EE\}$ along the given $\boldsymbol{u}$-direction. The joint space inertia matrix for the total system then becomes

$$\begin{aligned}\boldsymbol{M}_r^*(\boldsymbol{q}) = \boldsymbol{M}_r(\boldsymbol{q}) + \boldsymbol{M}_\ell(\boldsymbol{q}) &= \boldsymbol{J}_r^\mathsf{T}(\boldsymbol{q})\left[\boldsymbol{\Lambda}_r^{-1}(\boldsymbol{q})+\boldsymbol{\Lambda}_\ell\right]\boldsymbol{J}_r(\boldsymbol{q}) \\ &= \boldsymbol{J}_r^\mathsf{T}(\boldsymbol{q})\boldsymbol{\Lambda}_r^*(\boldsymbol{q})\boldsymbol{J}_r(\boldsymbol{q}).\end{aligned} \tag{8}$$

The overall operational space kinetic energy matrix of the payload-extended robot is given by

$$\boldsymbol{\Lambda}_r^*(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{\Lambda}_{r,v}(\boldsymbol{q})+m_\ell\boldsymbol{I} & \boldsymbol{\Lambda}_{r,v\omega}(\boldsymbol{q})-m_\ell\hat{\boldsymbol{r}}_{CoM_\ell} \\ \left[\boldsymbol{\Lambda}_{r,v\omega}(\boldsymbol{q})-m_\ell\hat{\boldsymbol{r}}_{CoM_\ell}\right]^\mathsf{T} & \boldsymbol{\Lambda}_{r,\omega}(\boldsymbol{q})+\boldsymbol{\mathcal{I}}_\ell+m_\ell\hat{\boldsymbol{r}}_{CoM_\ell}^\mathsf{T}\hat{\boldsymbol{r}}_{CoM_\ell} \end{bmatrix}. \tag{9}$$

According to [8], the instantaneous perceived reflected mass of the payload-extended robot in a normalized Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$ is calculated from the translational part of (9) as

$$m_{r,\boldsymbol{u}}^*(\boldsymbol{q}) = m_{CoM,\boldsymbol{u}}^*(\boldsymbol{q}) = \left[\boldsymbol{u}^\mathsf{T}\boldsymbol{\Lambda}_{r,v}^{*-1}\boldsymbol{u}\right]^{-1} = m_{POI,\boldsymbol{u}}^*(\boldsymbol{q}). \tag{10}$$

Hence, $\boldsymbol{\Lambda}_{r,v}^{*-1}$ provides a description of the response of any point on the end-effector/payload to an external force [8].

Cartesian Velocity of Payload-Extended Robot at POI: The relation between joint velocities $\dot{\boldsymbol{q}}$ and operational space velocities $\dot{\boldsymbol{x}}_\ell$ at a payload POI with impact curvature c_{POI_ℓ} and expressed at the end-effector frame $\{EE\}$ is

$$\begin{aligned}\dot{\boldsymbol{x}}_{POI_\ell} = \begin{bmatrix}\boldsymbol{v}_{POI_\ell} \\ \boldsymbol{\omega}_{POI_\ell}\end{bmatrix} &= {}^{EE}\boldsymbol{J}_{POI_\ell}^{-1}\boldsymbol{J}_r(\boldsymbol{q})\dot{\boldsymbol{q}} \\ = \begin{bmatrix}\boldsymbol{I} & -\hat{\boldsymbol{r}}_{POI_\ell} \\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{J}_{r,v}(\boldsymbol{q}) \\ \boldsymbol{J}_{r,\omega}(\boldsymbol{q})\end{bmatrix}\dot{\boldsymbol{q}} &= \begin{bmatrix}\boldsymbol{J}_{r,v}(\boldsymbol{q})-\hat{\boldsymbol{r}}_{POI_\ell}\boldsymbol{J}_\omega(\boldsymbol{q}) \\ \boldsymbol{J}_{r,\omega}(\boldsymbol{q})\end{bmatrix}\dot{\boldsymbol{q}} \\ &= \begin{bmatrix}\boldsymbol{J}_{r,v}^*(\boldsymbol{q}) \\ \boldsymbol{J}_{r,\omega}^*(\boldsymbol{q})\end{bmatrix}\dot{\boldsymbol{q}} = \boldsymbol{J}_r^*(\boldsymbol{q})\dot{\boldsymbol{q}},\end{aligned} \tag{11}$$

where $\hat{\boldsymbol{r}}_{POI_\ell}$ is the cross-product operator matrix for the position vector of $\hat{\boldsymbol{r}}_{POI_\ell}$ from $\{EE\}$.

In order to calculate the maximum velocity at the payload POI, one could e. g. obtain a graphical solution using manipulability polytopes or a kinematics optimization problem has to be solved as in [9]. Considering the payload-extended robot, such optimization problems involves the velocity Jacobians at the desired point of interest. Accordingly, the maximum velocity of the payload-extended robot at the payload POI along $\boldsymbol{u}$ is given by optimizing for the maximum of

$$\boldsymbol{v}_{r,\boldsymbol{u}}^*(\boldsymbol{q}) = \boldsymbol{u}^\mathsf{T}\boldsymbol{J}_v^*(r,\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{u}^\mathsf{T}\boldsymbol{v}_r^*(\boldsymbol{q}). \tag{12}$$

Note that the two components of the payload-extended translational Jacobian $\boldsymbol{J}_v^*(r,\boldsymbol{q})\dot{\boldsymbol{q}}$ are not indepdendent from each other. Thus, the optimized kinematics of the payload-free robot at $\{EE\}$ cannot be directly used to obtain the maxmium velocity for the payload-extended robot at POI.

III. Reflected Mass Update with Payload

In order to efficiently update the reflected mass of the robot upon the addition of a known payload, we seek at least an approximate analytical formulation. Thus, let us revisit relevant matrix inversion identities for the sum of two matrices and consider some basic assumptions on the invertibility of sub- and intermediate matrices beforehand.

Identity I: Since $\boldsymbol{\Lambda}_r^*(\boldsymbol{q})$ is a symmetric block matrix, it can be expressed as a structure similar to a block-diagonal matrix [10] as

$$\begin{aligned}\boldsymbol{\Lambda}_r^*(\boldsymbol{q}) &= \begin{bmatrix}\boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^\mathsf{T} & \boldsymbol{D}\end{bmatrix}, \begin{cases}\boldsymbol{A} = \boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q}) + m_\ell\boldsymbol{I} \\ \boldsymbol{B} = \boldsymbol{\Lambda}_{r,\boldsymbol{v}\omega}(\boldsymbol{q}) - m_\ell\hat{\boldsymbol{r}} \\ \boldsymbol{C} = \left[\boldsymbol{\Lambda}_{r,\boldsymbol{v}\omega}(\boldsymbol{q}) - m_\ell\hat{\boldsymbol{r}}\right]^\mathsf{T} \\ \boldsymbol{D} = \boldsymbol{\Lambda}_{r,\omega}(\boldsymbol{q}) + \boldsymbol{\mathcal{I}}_\ell + m_\ell\hat{\boldsymbol{r}}^\mathsf{T}\hat{\boldsymbol{r}}\end{cases} \\ &= \begin{bmatrix}\boldsymbol{I} & \boldsymbol{B}\boldsymbol{D}^{-1} \\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{A}-\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^\mathsf{T} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{B}\boldsymbol{D}^{-1} \\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}^\mathsf{T}.\end{aligned} \tag{13}$$

Given that $\boldsymbol{A}$, $\boldsymbol{D}$ and both Schur complements $\boldsymbol{A}-\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^\mathsf{T}$ and $\boldsymbol{D}-\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B}$ are invertible except for singular configurations, the inverse $\boldsymbol{\Lambda}_r^{*-1}(\boldsymbol{q})$ is [8], [11]

$$\begin{aligned}\boldsymbol{\Lambda}_r^{*-1}(\boldsymbol{q}) &= \begin{bmatrix}(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^\mathsf{T})^{-1} & -\boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{D}-\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B})^{-1} \\ -(\boldsymbol{D}-\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1} & (\boldsymbol{D}-\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\end{bmatrix} \\ &\doteq \begin{bmatrix}\boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{*-1}(\boldsymbol{q}) & \overline{\boldsymbol{\Lambda}_{r,\boldsymbol{v}\omega}^*}(\boldsymbol{q}) \\ \overline{\boldsymbol{\Lambda}_{r,\boldsymbol{v}\omega}^*}^\mathsf{T}(\boldsymbol{q}) & \boldsymbol{\Lambda}_{r,\omega}^{*-1}(\boldsymbol{q})\end{bmatrix}.\end{aligned} \tag{14}$$

Consequently, $\boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{*-1}(\boldsymbol{q}) = \left(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^\mathsf{T}\right)^{-1}$ can be further simplified [10] to

$$\begin{aligned}\boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{*-1}(\boldsymbol{q}) &= \boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}\left(\boldsymbol{D}-\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B}\right)^{-1}\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1} \\ &= \boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{\Lambda}_{r,\omega}^{*-1}(\boldsymbol{q})\boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}.\end{aligned} \tag{15}$$

Moreover, applying a special case of the binomial inverse theorem, namely Hua's identity [12], leads to

$$\begin{aligned}\boldsymbol{A}^{-1} &= \left(\boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q}) + m_\ell\boldsymbol{I}\right)^{-1} \\ &= \boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{-1}(\boldsymbol{q}) - \left[\boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q}) + \boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q})\frac{1}{m_\ell}\boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q})\right]^{-1}.\end{aligned} \tag{16}$$

By inserting (16) into (15), one gets an update expression for the reflected mass with isolated quantification of the payload contribution. Now, this mass update expression can only be further simplified under special conditions. For example, considering $\boldsymbol{A} \gg \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{B}^\mathsf{T}$ in (14) (or equivalently, $\boldsymbol{D} \gg \boldsymbol{B}^\mathsf{T}\boldsymbol{A}^{-1}\boldsymbol{B}$), a computationally cheaper update expression with direct quantification of the payload contribution to the updated reflected mass would be obtained

$$\boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{*-1}(\boldsymbol{q}) = \boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{-1}(\boldsymbol{q}) \underbrace{-\boldsymbol{\Lambda}_{r,\boldsymbol{v}}^{-1}(\boldsymbol{q})\left[\boldsymbol{I} + \frac{1}{m_\ell}\boldsymbol{\Lambda}_{r,\boldsymbol{v}}(\boldsymbol{q})\right]^{-1}}_{\text{special payload contribution}}. \tag{17}$$

However, this approximation is valid only for very special types of payloads that have a large *inertia to mass ratio* around $\{EE\}$ along $\boldsymbol{u}$. However, typically this is neither physically realistic nor applicable in most commercial grippers.

IV. Velocity Update with Payload

As mentioned in Sec. II-B, to obtain the maximum Cartesian velocity of the payload-extended robot at the payload POI along the direction of motion, generally an optimization problem has to be solved. Firstly, we introduce the according optimization problems to obtain the maximum velocity at a chosen POI. Specifically, we deal with the cases of the angular end-effector motion being constrained to be identically zero, i. e. pure translational motion, or not. On this

basis, we then discuss different alternatives to obtain the maximum velocity at any payload POI lying on the robot motion direction.

A. Problem 1: Translation ($\boldsymbol{\omega}_r = \mathbf{0}$)

We determine the maximum possible velocity in a normalized Cartesian direction $\boldsymbol{u}$ under consideration of joint velocity constraints by formulating a static optimization problem. First, the Cartesian velocity shall satisfy $\dot{\boldsymbol{x}}_r = [\boldsymbol{v}_r^\mathsf{T}, \boldsymbol{\omega}_r^\mathsf{T}]^\mathsf{T} = [v_r \boldsymbol{u}^\mathsf{T}, 0, 0, 0]^\mathsf{T}$. Let us now consider following problem with mixed constraints in standard form.

$$\begin{aligned} &\max_{\tilde{x}_i \in \mathbb{R}} f(\tilde{\mathbf{x}}) \\ &\text{subject to:} \\ &g_1(\tilde{\mathbf{x}}) \leq b_1, \ \dots, g_k(\tilde{\mathbf{x}}) \leq b_k \\ &h_1(\tilde{\mathbf{x}}) = c_1, \ \dots, h_m(\tilde{\mathbf{x}}) = c_m \end{aligned} \tag{18}$$

The optimization variable is the vector $\tilde{\mathbf{x}} = [v_r, \dot{q}_1, \dots \ , \dot{q}_n]^\mathsf{T}$ comprised of the reflected Cartesian velocity and associated joint velocities $\dot{\boldsymbol{q}} \in \mathbb{R}^n$. The associated cost function is

$$f(\tilde{\mathbf{x}}) = v_r \rightarrow \max. \tag{19}$$

From $\dot{\boldsymbol{x}}_r = \boldsymbol{J}_r(\boldsymbol{q})\dot{\boldsymbol{q}}$ we obtain the m=3+3=6 equality constraints $h_1(\tilde{\mathbf{x}}), \ \dots, h_6(\tilde{\mathbf{x}})$

$$\mathbf{0} = \boldsymbol{J}_r(\boldsymbol{q})\dot{\boldsymbol{q}} - [v_r \boldsymbol{u}^\mathsf{T}, 0, 0, 0]^\mathsf{T}. \tag{20}$$

The physical requirement for a positive speed together with the symmetric joint velocity limits define the k=1+2n inequality constraints $g_1(\tilde{\mathbf{x}}), \ \dots, g_{1+2n}(\tilde{\mathbf{x}})$

$$\begin{aligned} v_r &\geq 0 \\ -\dot{\boldsymbol{q}}_{\max} \leq \dot{\boldsymbol{q}} &\leq \dot{\boldsymbol{q}}_{\max} \end{aligned} \tag{21}$$

For simplicity and without loss of generality, we neglect the limitations on the available joint torque. Also, one may assume that k_0 inequality constraints are binding while the rest $k-k_0$ are not. This is generally true for robotic manipulators since, to achieve the maximum velocity, some joints has to be saturated while restricting the rest. This means that the other joints may not saturate simultaneously due to the optimized kinematic structure. Let us further assume that the Jacobian of the equality constraints and binding inequality constraints, i. e. saturating joint velocities, at the local maximum $\tilde{\mathbf{x}}^*$ of $f(\tilde{\mathbf{x}})$ have full rank. Then, according to [13], $\tilde{\mathbf{x}}^*$ can be calculated that complies with the constraint set given by the k inequalities and m equalities.

TABLE I
SOLVING OPTIMIZATION PROBLEM I.

Form the Lagrangian: $L(\tilde{\mathbf{x}}, \lambda_1, ..., \lambda_k, \mu_1, ..., \mu_m)$
$= f(\tilde{\mathbf{x}}) - \sum_{i=1}^{k} \lambda_i [g_i(\tilde{\mathbf{x}}) - b_i] - \sum_{i=1}^{m} \mu_i [h_i(\tilde{\mathbf{x}}) - c_i]$
There exists an optimal set of Lagrange multipliers:
${\lambda_1}^*, ..., {\lambda_k}^*$ and ${\mu_1}^*, ..., {\mu_m}^*$
such that:
1) $\frac{\partial L}{\partial \tilde{x}_i}(\tilde{\mathbf{x}}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = 0$ for all $i \in \{1, ..., n\}$
2) $\lambda_i^* [g_i(\tilde{\mathbf{x}}) - b_i] = 0$ for all $i \in \{1, ..., k\}$
3) $h_i(\tilde{\mathbf{x}}^*) = c_i$ for all $i \in \{1, ..., m\}$
4) $g_i(\tilde{\mathbf{x}}^*) \leq b_i$ for all $i \in \{1, ..., k\}$
5) $\lambda_i^* \geq 0$ for all $i \in \{1, ..., k\}$

By solving the optimization problem e.g. via linear programming one obtains $v_{r,\boldsymbol{u},\max} = v_{r,\max}\boldsymbol{u}$. Due to limited joint torque dynamics of a given system this maximum velocity is not always practically feasible. However, a found solution should thus be a conservative upper estimate.

B. Problem 2: Translation and rotation ($\boldsymbol{\omega}_r$ not constrained)

Now, we relieve the angular velocity $\boldsymbol{\omega}_r$ from being zero. This means that it may be used to further increase v_r along $\boldsymbol{u}$. Then, the reduced equality constraints $h_1(\tilde{\mathbf{x}}), \ \dots, h_3(\tilde{\mathbf{x}})$ for the optimization problem would become

$$\mathbf{0} = \boldsymbol{J}_{r,\boldsymbol{v}}(\boldsymbol{q})\dot{\boldsymbol{q}} - v_r \boldsymbol{u}. \tag{22}$$

Please note that within the *Safety Map* framework $\boldsymbol{\omega}$ is not entirely free, but confined to a desired rotational unit vector $\boldsymbol{v}$ (similar to $\boldsymbol{u}$ in translation), meaning $\dot{\boldsymbol{x}}_r = [v_r \boldsymbol{u}^\mathsf{T}, \omega_r \boldsymbol{v}^\mathsf{T}]^\mathsf{T}$. However, in this paper we will give more freedom to the optimization in order to better show the full effect the rotational velocity vector has on the maximization of translational velocity.

The cost function and the inequality constraints remain the same. With this very similar problem formulation (except for k=3), the resulting translational velocity $v_{r,\boldsymbol{u},\max}$ generally becomes larger than the one determined by optimization problem 1.

Operational Velocity of the Payload: Let us assume a POI that lies along the robot direction of motion $\boldsymbol{u}$, see Fig. 1. For determining the influence of this POI on the safe operational velocity along $\boldsymbol{u}$ ($v_{\boldsymbol{u}}^{\text{safe}}$), one can use the maximum end-effector velocity. To do so, the maximization has to be carried out first at the the robot end-effector. Then, the payload is considered as a lever arm between $\{EE\}$ and $POI_{\boldsymbol{u}}$, which makes it cheaper to calculate the maximum velocity at POI for all point of interests along $\boldsymbol{u}$ ($v_{POI_{\boldsymbol{u}}}$). Hence, their possible restricting effects on $v_{\boldsymbol{u}}^{\text{safe}}$ can be quickly assessed. A more sensible but expensive approach is to carry out the velocity optimization at each $POI_{\boldsymbol{u}}$ directly. For the first option, we can find the maximum velocity at the robot end-effector and the associated joint velocity $\dot{\boldsymbol{q}}$ via optimization problem 1 or 2. Then, we can determine the

Cartesian velocities of the payload point of interest using (11).

Depending on the angular velocity of the payload and the geometry at $\boldsymbol{v}_{POI_{\boldsymbol{u}}}$, the velocity of $POI_{\boldsymbol{u}}$ may be larger or smaller than the velocity of the robot alone. Regarding the second possibility, i.e. directly optimizing the velocity at $POI_{\boldsymbol{u}}$, the optimized translational velocities of the payload are generally larger than the velocity at robot end-effector. Moreover, the direct optimization scheme can be used for arbitrary points of interest.

V. Results for the PUMA 560

A. Simulation Setup

In Fig. 2 the influence of a rigid payload on the reflected mass and velocity is illustrated for the PUMA 560 robot. Note that the proposed velocity optimization schemes may also be applied to redundant robots. The payload mass and inertia tensor about the main axes are chosen to be $m_\ell = 2\,\mathrm{kg}$ and ${}^{EE}\boldsymbol{\mathcal{I}}_\ell = diag([0.2, 0.3, 0.4])\ \mathrm{kg\,m^2}$, respectively. The position vector of the payload center of mass to end-effector is $\hat{\boldsymbol{r}}_{CoM_\ell} = [0.05, 0.01, 0.015]^\mathsf{T}\ \mathrm{m}$. An exemplary payload point of interest with $c_{POI_\ell}{=}c_{POI_{EE}}$ along the robot motion direction $\boldsymbol{u}$ was chosen ($POI_{\boldsymbol{u}}$), see Fig. 1. Furthermore, the reflected mass and velocity at $\{EE\}$ and $POI_{\boldsymbol{u}}$ are depicted within the *Safety Map* framework.

First, the influence of the payload on the reflected mass is shown, then its effect on the maximum operational velocity is discussed. For this purpose, results from both approaches to velocity maximization are illustrated. In the upper row, the reachable discretized Cartesian cuboids (colored grey, each has a side length of $25\,cm$) are shown for the main projections. In the lower row, the *Safety Map* for the payload-extended robot was obtained by updating the reflected mass and considering velocity optimization with $\boldsymbol{\omega}_r{=}\mathbf{0}$ and $\boldsymbol{\omega}_r$ being free. For the given payload parameters and using (15) for updating the reflected mass, we may save $\approx 32.4\%$ of the time that one would need to recalculate it from scratch. This result was obtained after implementing the described update expression for the reflected mass in C-language and using the optimized Meschach library of routines to perform the involved matrix computations [14]. The development PC has a GNU GCC Compiler running on a MS Windows 10© machine with Intel Core i7-7700 CPU of 3.60 GHz and 16.0 GB of RAM memory.

B. Discussion

Attaching a rigid payload to the PUMA 560 increases the reflected mass in any Cartesian direction $\boldsymbol{u}$, see Fig. 2, lower row. Under the assumption that the curvatures of end-effector and payload at point of contact are geometrically similar ($c_{POI_\ell}{=}c_{POI_{EE}}$), this additional payload reduces the collision safety of the robot. In the *Safety Map*, the payload contribution may therefore result more likely in an intersection of the robot mass/velocity range with the human injury data representation. As for real-time safety control with the *SMU*, a higher reflected mass at constant impact curvature leads to a further velocity decrease, which in turn increases the cycle time of a given task.

For calculating the inverse of the updated kinetic energy matrix for a general payload, one needs to either invert one $6{\times}6$ matrix for every $\boldsymbol{q}$ and then select the required $3{\times}3$ translational part, or alternatively carry out one inversion, five multiplications and one addition (all involving $3{\times}3$ matrices). However, as discussed for some special payloads that have large inertia to mass ratio, a computationally cheaper but approximate update law can be used. This simplification needs only one inversion, two additions, one multiplication and one scaling (again, all calculations are involving $3{\times}3$ matrices).

When considering translational velocities in the strong sense ($\boldsymbol{\omega}_r{=}\mathbf{0}$), then the payload-extended robot *Safety Map* representation simply shifts in direction of the reflected mass, as the maximum velocity at each location on the payload is the same as the velocity at the robot end-effector. For maximum operational velocity in the weak sense with no constraint on angular motion ($\boldsymbol{\omega}_r{=}\mathbf{0}$), it can be seen in Fig. 2 (lower row), that our exemplary payload POI leads to a velocity increase. The physical interpretation is the payload POI is located at a lever arm rotating around $\{EE\}$. The magnitude of this angular motion times the distance of this payload POI to $\{EE\}$ ($= \boldsymbol{r}_{POI_{\boldsymbol{u}}}$) generates additional translational velocity, see Fig. 2 (lower row). Furthermore, optimization problem 2 provides the largest translational velocity compared to optimization problem 1 or a pseudo-inverse solution. In any case, using the joint velocity that was determined for maximum velocity at the robot end-effector also for calculating the velocity at the POI_ℓ through (11) would lead to false results. This is because the angular velocity contribution at the POI is not taken into account in the optimization, and thus its effects on maximum velocity is underestimated.

The velocity optimization at POI is therefore more consequent. However, the relationship between optimal velocity at POI_ℓ and at the robot end-effector is difficult to determine. One reason for this is that the solution relies on numerical optimization, which is typically nonintuitively interpretable. Therefore, the question whether there is a way to inexpensively reuse the solution of the payload-free optimization problem for the payload-extended case needs further investigation and is entangled to the more constrained rotational velocity within the *Safety Map* framework.

VI. Conclusion

In this work, the influence of an inertial payload with spatial relevance on the reflected inertial parameters and maximum operational velocity in a desired motion direction is analyzed. We derived analytical update laws for the robot reflected mass that can be employed to quantify the payload contribution to the kinetic energy matrix. This allows reusing apriori inertial information, i.e. previously generated for the payload-free case. We also investigated the problem of finding the maximum robot operational velocity in the robot motion direction of the payload-free and payload-extended

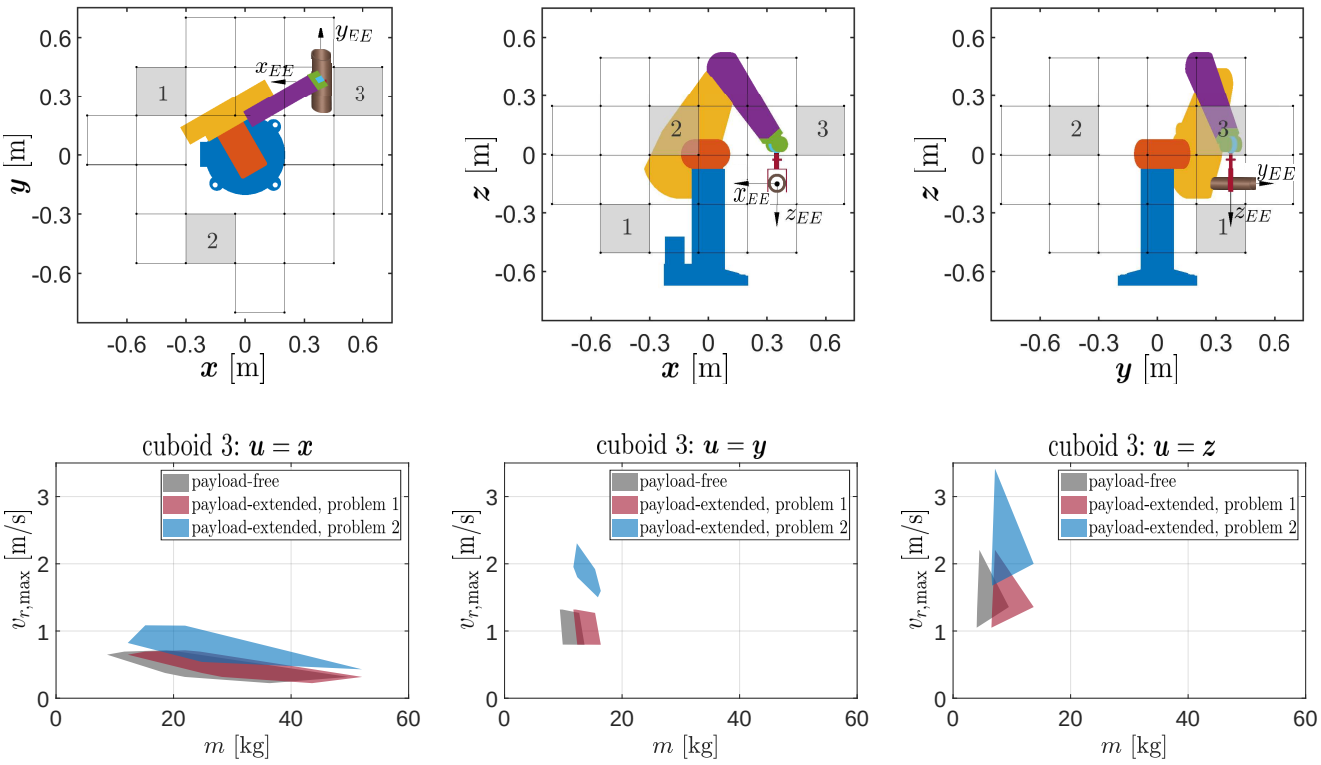

Fig. 2. PUMA 560: Influence of a rigid end-effector payload (with an exemplary POI along $\boldsymbol{u}$) on the reflected mass and maximum task velocity for the cuboid 3. Motions along the three basic Cartesian directions $\boldsymbol{u} = \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ are considered.

cases under joint velocity constraints.This was done based on static optimization problems. We implemented and verified the approach for the PUMA 560 robot in simulation and within the *Safety Map* framework.

Acknowledgment

This work was supported by the European Union's Horizon 2020 research and innovation programme as part of the project ILIAD under grant no. 732737, and by the Alfried Krupp von Bohlen und Halbach Foundation.

References

[1] 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," *Int. J. of Robotics Research*, vol. 31, pp. 1578–1602, 2012.

[2] N. Mansfeld, M. Hamad, M. Becker, A. G. Marin, and S. Haddadin, "Safety map: A unified representation for biomechanics impact data and robot instantaneous dynamic properties," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1880–1887, July 2018, nominated for ICRA 2018 Best Paper Award on Human-Robot Interaction.

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

[4] M. W. Gertz, J.-O. Kim, and P. K. Khosla, "Exploiting redundancy to reduce impact force," in *Intelligent Robots and Systems' 91.'Intelligence for Mechanical Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on*. IEEE, 1991, pp. 179–184.

[5] S.-D. Lee, B.-S. Kim, and J.-B. Song, "Human–robot collision model with effective mass and manipulability for design of a spatial manipulator," *Advanced Robotics*, vol. 27, no. 3, pp. 189–198, 2013.

[6] 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 *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 5390–5397.

[7] N. Mansfeld, M. Hamad, M. Becker, A. G. Marin, and S. Haddadin, "Safety map: A unified representation for biomechanics impact data and robot instantaneous dynamic properties," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1880–1887, July 2018.

[8] O. Khatib, "Inertial properties in robotics manipulation: An object-level framework," *The International Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995.

[9] J. Lee, "A study on the manipulability measures for robot manipulators," in *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems. Innovative Robotics for Real-World Applications. IROS'97*, vol. 3. IEEE, 1997, pp. 1458–1465.

[10] J. Gallier, "The schur complement and symmetric positive semidefinite (and definite) matrices," 2017.

[11] R. A. Horn, C. R. Johnson, and R. A. M. a. Horn, *Topics in matrix analysis*, 1st ed. Cambridge ; New York : Cambridge University Press, 1994, sequel to: Matrix analysis.

[12] P. M. Cohn, *Further algebra and applications*. Springer Science & Business Media, 2011.

[13] J. Wilde, "Constrained optimization," 2013, [Online; accessed 23-July-2019, available at http://www.columbia.edu/~md3405/Constrained_Optimization.pdf].

[14] D. E. Stewart and Z. Leyk, *Meschach: Matrix computations in C*. Centre for Mathematics and its Applications, Australian National University, 1994, vol. 32.