



HAL
open science

Experimental validation of an energy constraint for a safer collaboration with robots

Lucas Joseph, Vincent Padois, Guillaume Morel

► **To cite this version:**

Lucas Joseph, Vincent Padois, Guillaume Morel. Experimental validation of an energy constraint for a safer collaboration with robots. 2018 International Symposium on Experimental Robotics (ISER 2018), Nov 2018, Buenos Aires, Argentina. hal-01883995

HAL Id: hal-01883995

<https://hal.science/hal-01883995>

Submitted on 29 Sep 2018

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

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

Experimental validation of an energy constraint for a safer collaboration with robots

Lucas Joseph^{1,2,3}, Vincent Padois¹ and Guillaume Morel^{1,3}

¹Sorbonne Université, CNRS UMR 7222, Institut des Systèmes Intelligents et de Robotique, ISIR, F-75005 Paris, France `lastname@isir.upmc.fr`

²General Electric Healthcare, 78530 Buc, France

³INSERM, U1150, Agathe-ISIR, F-75005 Paris, France

Abstract. In this paper a platform dedicated to the measure of the energy dissipated by a robot during an impact is presented. This platform is used to validate a controller that explicitly constrains the robot kinetic energy thanks to a linear quadratic problem formulation. This constraint is used to ensure safety at the control level by computing intrinsically safe commands that comply with ISO norm 15066 on collaborative robot safety. This platform has been designed to validate this approach in case of a collision with an obstacle and is experimented on a KUKA LWR4+ robot.

1 Introduction

The further adoption of robots for collaborative tasks with an operator must go through an enhancement of the user safety. To that extent several research projects have been conducted. They led to the introduction of lightweight robots with a reduced mass and torque sensing capabilities [1], to new variable impedance actuators [2] and solutions to detect and avoid obstacles with exteroceptive sensors [3].

Collaborative robots are designed to work closely with humans. To perform their tasks, these robots have torque/force capacities that make them potentially dangerous for human-robot collaboration. A first measure to improve safety in the vicinity of these robots consists in avoiding unwanted contact with humans. Different obstacle avoidance algorithms have been proposed in the literature. [4] expose a technique called elastic bands that updates the robot trajectory on-line to define a collision-free path. Potential fields methods, first introduced by [5], create virtual forces around obstacles. The magnitude of these forces is linked to the distance between the robot and an obstacle. Whether obstacle avoidance algorithms are implemented by mean of control or through path planning, in cluttered environments with unpredictable human motions, they cannot guaranty that no contact can happened between a fixed base robot and a human.

If contact cannot be avoided, it should at least be safe for the operator. A common approach is to implement algorithms to detect and react to a collision.

Many detection methods exist. They range from comparing the robot actual torques to the theoretical ones necessary to perform its task to observing the change of energy in the system [6]. For a complete review on collision detection algorithms, one may refer to [7]. Once a collision is detected, the work of [8] proposes to stop the robot or to switch its control mode to a zero-gravity torque control. Alternatively, [9] develops a control solution to realise motion and/or force control at any point on the robot and propose to control the force exerted by a robot at the point where a collision occurred.

As effective as these techniques can be, detecting and reacting to a collision is not sufficient to ensure safety. If at the moment of impact the robot is moving too fast or with an important load, the collision by itself might be fatal. Before reacting to a collision, the robot should be actuated in a way that is intrinsically safe. ISO Norm 15066 [10] proposes a new definition of a robot dangerousness during transient contact, *i.e.* an impact. The norm proposes to quantify a robot kinetic energy and provides energy limits to respect in order to avoid severe human injuries.

In a previous work [11], a robot kinetic energy is expressed as a constraint inside a quadratic programming (QP) problem. This constraint ensures that the robot does not exceed a defined limit considered as dangerous. It shows that by correctly expressing the constraint, it is possible to ensure safety in different cases of collision. However, the kinetic energy computed through this method is based on a theoretical dynamic model of the robot and does not guaranty to be representative of the true robot kinetic energy.

To that extent, a platform has been developed to measure the energy transferred by a robot during contact. This platform is used in this work to validate the approach proposed in [11] by comparing the energy measured by the platform to the one computed with the robot model.

In a first section, the QP formulation is explained and the kinetic energy constraint is detailed. In a second section the experimental set-up for the experiments is detailed. In a third section several experiments are realised and experimental results are presented to validate the correct enforcement of the proposed energy constraint. The last section concludes on this work and proposes further improvements that are being investigated.

2 Constrained convex problem formulation

The control of a robot can be expressed as an optimization problem where cost functions define the robot tasks and constraints modelize the robot actuation limits [12]. The optimization problem requires expressing tasks and constraints as functions of the control variable. In this work, the control variable is the joint torque, $\tau \in \mathbb{R}^n$, where n is the robot number of degree of freedom (dof). This section focuses on the expression of a quadratic programming problem to ensure safety by defining a kinetic energy constraint.

2.1 Tasks

The robot tasks are formulated in terms of commanded task space acceleration $\dot{\mathbf{v}}^* \in \mathbb{R}^6$, where $\mathbf{v} \in \mathbb{R}^6$ is the end-effector twist expressed at some point of interest. The tracking of this task is ensured by a PID controller. A robot task can thus be expressed as:

$$\mathbf{T}(\boldsymbol{\tau}) = \dot{\mathbf{v}}^* - \dot{\mathbf{v}}(\boldsymbol{\tau}) \quad (1)$$

Using the robot equation of motion, it is possible to express tasks as a function of the robot torque such that

$$\mathbf{T}(\boldsymbol{\tau}) = \dot{\mathbf{v}}^* - (\dot{J}\dot{\mathbf{q}} + JM^{-1}(\boldsymbol{\tau} - \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{g}(\mathbf{q}))) \quad (2)$$

where \mathbf{q} and $\dot{\mathbf{q}} \in \mathbb{R}^n$ are respectively the joint configuration and its derivative, J and $\dot{J} \in \mathbb{R}^{6 \times n}$ represent the robot Jacobian and its derivative expressed at the end-effector point. $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ is the centrifugal and Coriolis induced joint torque, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ the gravity induced joint torque and $M(\mathbf{q})$ is the $n \times n$ joint-space inertia matrix.

2.2 Constraints

Quadratic programming allows expressing constraints as inequalities. These constraints prevent the robot from reaching forbidden states. The robot intrinsic constraints, *i.e.* its bounds on joint positions, velocities and torques are one of the constraints that should be considered when controlling a robot. These constraints can all be expressed as inequalities and using a proper discretization scheme such as in [12] they can overall be written under the general form

$$C(\mathbf{q}, \dot{\mathbf{q}})\boldsymbol{\tau} \leq \mathbf{h} \quad (3)$$

with $C(\mathbf{q}, \dot{\mathbf{q}})$ the constraint matrix and \mathbf{h} its associated limit vector. The specific case of the energetic constraint is dealt with in section 2.4.

2.3 Quadratic programming formulation

The control of the robot requires finding the optimal command input, $\boldsymbol{\tau}^{opt}$, satisfying the constraints and minimizing the tasks. This problem can be expressed as a quadratic programming optimization which is defined as

$$\begin{aligned} \boldsymbol{\tau}^{opt} = \underset{\boldsymbol{\tau}}{\operatorname{argmin}} \quad & \frac{1}{2} (\|\mathbf{T}(\boldsymbol{\tau})\|^2 + \varepsilon\|\mathbf{r}\|^2) \\ \text{s.t.} \quad & C(\mathbf{q}, \dot{\mathbf{q}})\boldsymbol{\tau} < \mathbf{h} \end{aligned} \quad (4)$$

where $\|\cdot\|$ is the euclidean norm, \mathbf{r} is a regularization term ensuring the uniqueness of the optimized solution and $\varepsilon \ll 1$ is a weighting term allowing monitoring the impact of the regularization on the main tasks performances.

For a kinematically redundant robot the choice of \mathbf{r} is important as it affects the robot behaviour when the defined tasks do not fully constrain the robot. \mathbf{r} is a task using all degrees of freedom. A common regularization task consists in minimizing the control variable, so here $\mathbf{r} = \|\boldsymbol{\tau}\|$. In [11], the controller is implemented on a redundant robot and the regularization task is set to minimize $\|\boldsymbol{\tau} - \mathbf{g}(\mathbf{q})\|^2$. This regularization task minimizes the difference between the computed torque and the gravity induced external torque. It allows performing free motions in the null-space of the robot main tasks by physically interacting with it.

2.4 The kinetic energy constraint

In its discrete form, the kinetic energy, $E_{c,k}$, of the robot end-effector can be written:

$$E_{c,k} = \frac{1}{2} \mathbf{v}_k^T \Lambda_k(\mathbf{q}_k) \mathbf{v}_k \quad (5)$$

where, $\Lambda_k(\mathbf{q}_k) = (J_k(\mathbf{q}_k)M_k^{-1}(\mathbf{q}_k)J_k^T(\mathbf{q}_k))^{-1} \in \mathbb{R}^{6 \times 6}$ is the operational-space inertia matrix and $\mathbf{v}_k \in \mathbb{R}^6$ its current velocity.

Considering a time period, Δt , between two discrete time k and $k + 1$, the robot velocity at the next time step can be approximated to

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \dot{\mathbf{v}}_k^c \Delta t, \quad (6)$$

where $\dot{\mathbf{v}}_k^c$ is the current control acceleration. Considering that the modification of the configuration of the robot during the time interval Δt is small enough, it can be shown that the difference of kinetic energy between two successive discrete times can be written:

$$E_{c,k+1} = E_{c,k} + \underbrace{\left(\mathbf{v}_k \Delta t + \frac{1}{2} \dot{\mathbf{v}}_k^c \Delta t^2 \right)^T}_{\Delta \mathbf{x}} \underbrace{\Lambda_k \dot{\mathbf{v}}_k^c}_{\mathbf{F}_k^c}. \quad (7)$$

On the one hand, $\Delta \mathbf{x}$ represents the expected variation of pose given an initial operational state $\{\mathbf{x}_k; \mathbf{v}_k\}$ and the commanded acceleration trajectory $\dot{\mathbf{v}}_k^c$. On the other hand, $\mathbf{F}_k^c = \Lambda_k \dot{\mathbf{v}}_k^c$, is the equivalent control wrench.

This formulation can be interpreted as the provisional kinetic energy in Δt seconds. This energy is expressed as a function of the robot torque using the equation of motion. It is then inserted as a constraint in the optimization problem so that $E_{c,k+1}(\boldsymbol{\tau}) \leq E_c^{lim}$. In [11], it is shown that this constraint can be used to limit the energy that can be transferred from the robot to an obstacle during transient contact and also to limit wrenches applied by the robot when it entraps an obstacle against a fixed surface. However, further validations are required to asses the correct respect of this energy limit.

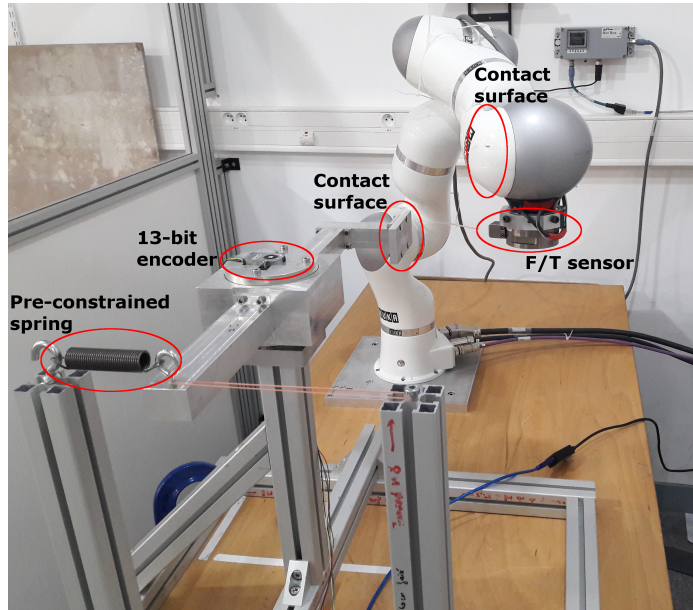


Fig. 1: Platform to measure the energy transferred by a robot during an impact. The elongation of the spring is measured by an encoder to determine the potential energy accumulated when an impact occurs.

3 Experimental setup

This work proposes to use an external device to quantify the energy transferred from the robot to an obstacle during an impact. This platform, depicted in Figure 1, measures the potential energy stored in a spring when a contact occurs. If the friction effects are negligible, this potential energy is equal the energy transferred by the robot at the impact.

This platform consists in a rigid structure on which is mounted a rotating beam. A spring is attached on one side of the beam and is pre-constrained. The robot enters in collision with the other side. Ball bearings are used to reduce friction at best. A 13-bit encoder is placed at the structure centre of rotation and a micro-controller queries its position at 500 Hz. The resulting elongation, Δx , of the spring is obtained using geometric relations. The spring stiffness, k , is measured beforehand using a micrometric table and a dynamometer with a 0.05N precision. The potential energy, E_p loaded in the spring is equal to $\frac{1}{2}k\Delta x^2$. The robot and the platform are mounted on a fix and rigid table. A bracing is mounted on the beam carrying the encoder to reduce any deformation of the structure during the collision.

The control framework proposed in section 2 is tested on a KUKA LWR4+. Torque commands are sent to the robot at 1 kHz using the KUKA Fast Research Interface through an OROCOS component [13]. The QP problem is implemented

using the qpOASES library [14]. When contact occurs, strong inertial efforts are applied on the robot. An ATI Gamma sensor is used to record these forces. A threshold is applied to determine the instant of collision. The kinetic energy of the robot at that instant is compared to the maximum potential energy stored in the platform spring.

4 Experimental results

This section provides experimental results obtained with the platform described previously. First the measurements of the platform are compared with the model based kinetic energy of the robot. In a second part, experiments are realised to verify the respect of the kinetic energy constraint.

In the two experiments, the robot moves along a straight line with a target position located behind the platform. The robot trajectory tracking task is performed using a PID controller. When a contact is detected, the task is cancelled so that it only remains the regularization task, leaving the robot in gravity compensation mode. This is to ensure that the PID controller does not integrate the error resulting from the collision and thus inject more energy to the platform.

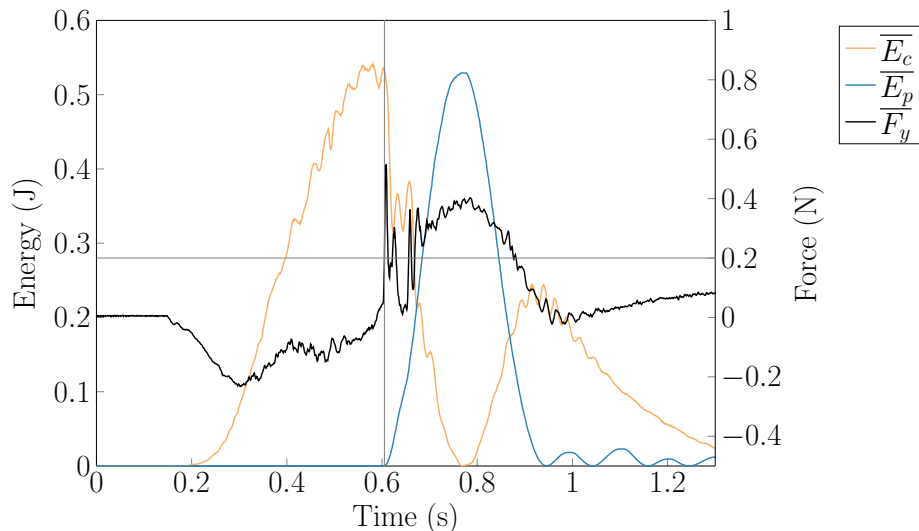


Fig. 2: Energy dissipation during an impact without a constraint on the kinetic energy.

Figure 2 compares the energy of the robot and the platform when a collision occurs. The orange curve represents $\overline{E_c}$, the average kinetic energy of the robot during the experiment over 10 trials. The blue curve represents the average potential energy stored in the platform spring, $\overline{E_p}$. In dark is the average force

along the y-axis, \overline{F}_y , recorded by the ATI sensor. It can be seen that prior to contact, the force recorded by the ATI is close to zero. When the collision occurs, a spike in the force is recorded. At the same instant a sudden drop of the robot kinetic energy can be observed as well as an accumulation of potential energy stored in the spring. Overall, the robot average kinetic energy at the instant of contact is 0.516 J. The average potential energy stored in the spring rises up to 0.528 J. It results a difference of 3.17 % and a standard deviation over 10 trials of 0.012 J between the model based kinetic energy and the measurement of the platform. This experiment thus validates the model-based computation of the kinetic energy.

In a second experiment, depicted on Figure 3, the kinetic energy limit, E_c^{lim} , is set at 0.2 J, below the 0.5 J required to correctly execute the task. The horizon time step Δt is set at 15 ms. Once again, the experiments are realised 10 times. The red curve represents the kinetic energy limit. The green curve represents the average predicted kinetic energy, $\overline{E}_{c,k+1}$, computed with the robot dynamic model. It is the energy, expressed in (7), that is constrained inside the QP solver. The predicted energy being expressed relatively to a horizon of time Δt , it represents the energy that the robot would have in that horizon of time. The QP solver finds torque commands to ensure that it never exceeds the limit E_c^{lim} (in red). Consequently, the resulting real kinetic energy is inferior to the predicted one. When the collision occurs, this current kinetic energy is on average equal to 0.168 J. The potential energy stored in the spring after impact rises up to 0.170 J. Over 10 trials, the standard deviation of the error is 0.002 J.

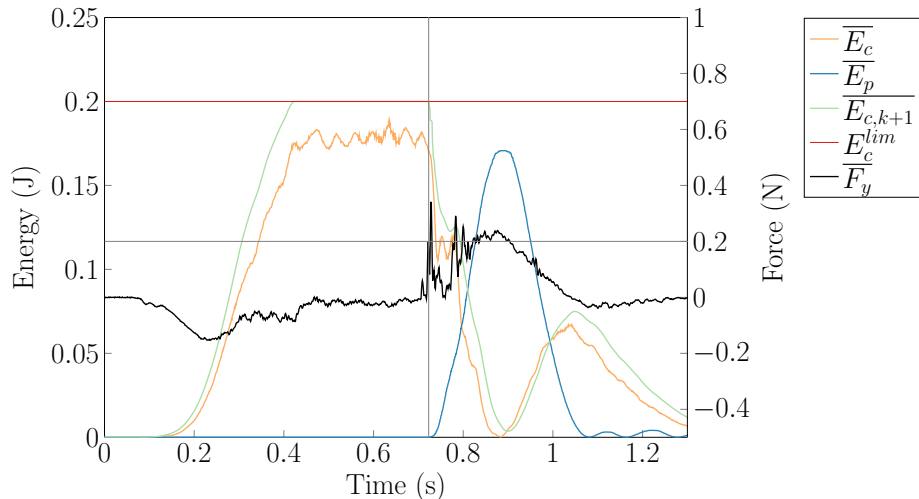


Fig. 3: Energy dissipation during an impact with a limited kinetic energy of 0.2 J.

The resulting energy that is constrained through this method is always inferior to the limit and validates the proposed solution. To be closer to the limit, it would require choosing a horizon of time Δt as close to the robot control period as possible. However, a too small time step could induce important variation of the robot commanded torques at the next time step which may lead to oscillation and commands that are not feasible by the actuators. Constraint compatibility should be taken into account as thoroughly detailed in [15].

5 Conclusion

Torque sensing robots offer new ways to interact with a robot and to detect collisions. However, depending on the robot current velocity and the tools it carries, the detection of a collision may not be sufficient to ensure safety. Ensuring safety thus also requires considering the case of unwanted contact. To alleviate this problem, ISO Norm 15066 proposes to limit the energy that can be transferred from the robot to a human body.

This work proposes a control solution that directly constrains the robot kinetic energy through a constrained convex optimization formulation. This kinetic energy constraint is expressed according to the theoretical model of the robot and does not mandatorily represents its real kinetic energy. To validate this model, a platform is designed to record the energy transferred from the robot during an impact. The results show that the model based kinetic energy is similar to the real one. This platform is then used to show the correct enforcement of the kinetic energy constraint. It experimentally proves that the formulation of a kinetic energy constraint can efficiently prevent the robot from transferring a dangerous amount of kinetic energy during an impact, thus improving the safety of a robotic solution.

When the robot kinetic energy is not sufficient to realise the tasks, the QP solver finds torque solutions reducing the trajectory tracking performances. This induces a decrease of the robot velocity. The only way to maximise the robot velocity while satisfying this constraint is to minimize the robot operational inertia matrix in the direction of motion. In the work of Mansfeld *et.al.* [16], a gradient descent method is used to reactively compute, at each time step, a joint configuration minimizing the robot perceived mass in the direction of motion. Such algorithms could be used in the control framework presented in this paper and enforced using the regularisation task. It is possible to define a regularisation task that pushes the robot toward a desired position minimizing its perceived mass towards an obstacle. Ongoing work is being performed to test this implementation and yields promising results.

Acknowledgement

This work has been partially funded by ANRT under CIFRE grant n° 2015/0288.

References

1. A. Albu-Schäffer, S. Haddadin, Ch. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger. The DLR lightweight robot: design and control concepts for robots in human environments. *Industrial Robot: An International Journal*, pages 376–385, 2007.
2. Antonio Bicchi and Giovanni Tonietti. Fast and ”soft-arm” tactics. *IEEE Robotics and Automation Magazine*, pages 22–33, 2004.
3. Fabrizio Flacco, Torsten Kroeger, Alessandro De Luca, and Oussama Khatib. A Depth Space Approach for Evaluating Distance to Objects. *Journal of Intelligent & Robotic Systems*, pages 1–21, 2014.
4. S. Quinlan and O. Khatib. Elastic bands: connecting path planning and control. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 802–807, 1993.
5. Oussama Khatib. Inertial properties in robotic manipulation: An object-level framework. *International Journal of Robotics and Research*, pages 19–36, dec 1995.
6. Alessandro De Luca, Alin Albu-Schäffer, Sami Haddadin, and Gerd Hirzinger. Collision detection and safe reaction with the DLR-III lightweight manipulator arm. *IEEE International Conference on Intelligent Robots and Systems*, pages 1623–1630, 2006.
7. Sami Haddadin, Alessandro De Luca, and Senior Member. Robot Collisions : Detection , Isolation , and Identification. In *IEEE Transaction on Robotics*, pages 1292–1312, 2017.
8. Sami Haddadin, Alin Albu-Schäffer, Alessandro De Luca, and Gerd Hirzinger. Evaluation of collision detection and reaction for a human-friendly robot on biological tissues. *6th IARP/IEEE-RAS/EURON Workshop on Technical Challenges for Dependable Robots in Human Environments*, 2008.
9. Emanuele Magrini, Fabrizio Flacco, and Alessandro De Luca. Control of generalized contact motion and force in physical human-robot interaction. *ICRA*, pages 2298–2304, 2015.
10. ISO/TS 15066. Robots and robotic devices collaborative robots, 2016.
11. Lucas Joseph, Vincent Padois, and Guillaume Morel. Towards X-ray medical imaging with robots in the open: safety without compromising performances. *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
12. Sébastien Rubrecht, Vincent Padois, Philippe Bidaud, Michel De Broissia, and Max Da Silva Simoes. Motion safety and constraints compatibility for multibody robots. *Autonomous Robots*, pages 333–349, 2012.
13. Herman Bruyninckx. OROCOS: design and implementation of a robot control software framework. *ICRA*, 2002.
14. Hans Joachim Ferreau, Christian Kirches, Andreas Potschka, Hans Georg Bock, and Moritz Diehl. qpOASES: a parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, pages 327–363, 2014.
15. Anis Meguenani. *Safe control of robotic manipulators in dynamic contexts*. Theses, Université Pierre et Marie Curie - Paris VI, October 2017.
16. Nico Mansfeld, Badis Djellab, Jaime Rald, Fabian Beck, Christian Ott, and Sami Haddadin. Improving the Performance of Biomechanically Safe Velocity Control for Redundant Robots through Reflected Mass Minimization. *IROS*, pages 5390–5397, 2017.