

Exploiting Robot Redundancy in Collision Detection and Reaction

Alessandro De Luca Lorenzo Ferrajoli
Dipartimento di Informatica e Sistemistica
Università di Roma “La Sapienza”
Via Ariosto 25, 00185 Roma, Italy
deluca@dis.uniroma1.it lorenzo.ferrajoli@tiscali.it

Abstract—We present a method that allows automatic reaction of a robot to physical collisions, while preserving as much as possible the execution of a Cartesian task for which the robot is kinematically redundant. The work is motivated by human-robot interaction scenarios, where ensuring safety is of primary concern whereas preserving task performance is an appealing secondary goal. Unexpected collisions may occur anywhere along the manipulator structure. Their fast detection is realized using our previous momentum-based method, which does not require any external sensing. The reaction torque applied to the joints reduces the effective robot inertia seen at the contact and lets the robot safely move away from the collision area. If we wish, however, to continue the execution of a Cartesian trajectory, robot redundancy can be exploited by projecting the reaction torque into the null space of a dynamic task matrix so as not to affect the original end-effector motion. This leads to the use of the so-called dynamically consistent approach to redundancy resolution, which is further elaborated in the paper. A partial task relaxation strategy can also be devised, with the objective of keeping contact forces below a user-defined safety threshold. Simulation results are reported for the 7R KUKA/DLR lightweight robot arm.

I. INTRODUCTION

The next generation of robots will be asked to strictly cooperate with people in industrial and service applications, implying a shared workspace for the human and robot. Therefore, safety issues in the physical interaction will be of primary concern. In particular, accidental collisions that may harm a human user should be mitigated by a prompt robot reaction while maintaining close cooperation [1], [2]. Developing new key components and control algorithms for robots that have to meet strict safety standards but also deliver useful performance is the main goal of the European project PHRIENDS (Physical Human-Robot Interaction: dependability and Safety).

Current methods of collision prevention, detection, and safe reaction are based upon various types of sensing devices and strategies. In structured environments, impacts between a human and a robot can be avoided by resorting to exteroceptive sensors and/or by imposing forbidden Cartesian workspace areas to the robot. For collision avoidance, recognition of safety-critical conditions is typically based on the processing of multiple external sensor data, such as

This work has been funded by the European Commission's Sixth Framework Programme as part of the project PHRIENDS under grant no. 045359.

those coming from surveillance or on-board vision, or from proximity sensors [3], [4].

To safely handle accidental collisions, a mechanical robot redesign has to be pursued in order to reduce injuries caused at the first impact. To this aim, a combination of lightweight robots with low link inertia [5], soft link covers [6], compliant transmissions to decouple motor from link inertia [7], and variable joint stiffness [8] can be used.

Another important safety issue is the activation of an appropriate robot reaction strategy in the post-impact phase of a collision. Indeed, the first step for implementing a reaction is to be able to detect collisions in a robust and fast way, and in any possible operative situation. The simplest reaction is then to immediately stop the robot, so as to avoid any further harmful impacts [9], [10].

Methods that recognize collisions without the need of external sensors are indeed very appealing. There are several examples of such collision detection schemes [11]–[14]. A successful, physically motivated approach is the disturbance observer presented in [15]. This method is based on the robot generalized momentum, it is fully independent from the particular control law applied to the robot, and is also able to estimate the colliding robot link and the main direction of contact forces. The original idea was later developed in collaboration with DLR and experimentally tested on

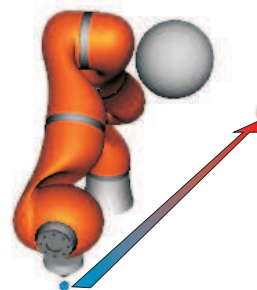


Fig. 1. A Cartesian motion task for the 7-dof KUKA/LWR arm in the presence of a spherical obstacle (unknown to the robot), located approximately halfway of the path and close to the elbow

the DLR-III Lightweight arm, using only proprioceptive sensors [16]. The unique features of this detection method have been useful for the design of a series of reaction strategies. In fact, based on the vector output (called residual) of the observer, the robot basically (over)reacts along the same direction of the collision forces in the Cartesian space by reducing its effective inertia. Up to now, the method has been evaluated mostly with reference trajectories in the joint space. When a collision is detected, the control law switches to the reaction strategy and the (joint-level) task is abandoned.

In this paper, we consider the use of robot redundancy for preserving Cartesian task execution despite the possible occurrence of collisions (see Fig. 1). Redundancy occurs whenever the number of degrees of freedom of the robot is larger than the number of variables needed to describe the robot task [17]¹. By exploiting redundancy, the robot can preserve the full (or at least partial) execution of an end-effector motion task, while still reacting to a detected collision so as to keep the contact forces (and the risk of injuries for a human) below a user-defined safety threshold. In this context, resolution of redundancy should be performed at a dynamic level (see, e.g., [18]–[20]), because of the coupling between the high accuracy requested in trajectory control and the speed of reaction to collisions, which is needed to guarantee safety. When there is not enough freedom to achieve simultaneously both the above goals, a strategy that relaxes only part of the motion task is a viable option. In this way, self-motion capabilities of the redundant robot could be better used to reconfigure the arm away collision.

The paper is organized as follows. After some preliminaries, our work on momentum-based collision detection and reaction is recalled in Sect. III. Dynamic redundancy resolution is presented in Sect. IV, and an alternative derivation of the original idea of dynamic consistency introduced in [20] is provided, showing also some possible extensions. The basic strategy for combining task preservation and collision reaction is given in Sect. V. Finally, simulation results are reported in Sect. VI for the 7R KUKA/LWR lightweight arm.

II. PRELIMINARIES

We consider robot manipulators as open kinematic chains of rigid bodies with n joints and generalized coordinates $\mathbf{q} \in \mathbb{R}^n$. The dynamic model is

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}, \quad (1)$$

where $\mathbf{M}(\mathbf{q})$ is the symmetric, positive definite robot inertia matrix (including reflected actuator inertias), the Coriolis and centrifugal terms are factorized using the matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ of Christoffel symbols, $\mathbf{g}(\mathbf{q})$ is the gravity vector, $\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})$ includes friction/dissipative terms, and $\boldsymbol{\tau}$ is the motor torque.

From the skew-symmetry of matrix $\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ it follows that

$$\dot{\mathbf{M}}(\mathbf{q}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}}). \quad (2)$$

¹The increasing diffusion of robotic systems with a large number of joints, such as dual-arm robots and humanoids, is raising new interest in problems and solutions related to redundancy.

During normal operation, the robot arm may collide with a standing or moving person/obstacle in its workspace. This collision could occur at any location along the robot structure. Although situations of multiple simultaneous collisions can also be handled, we will assume in the following the case of a single collision for simplicity. Let $\mathbf{V}_K = \mathbf{J}_K(\mathbf{q})\dot{\mathbf{q}}$ be the linear and angular velocity at the (unknown) contact point, where $\mathbf{J}_K(\mathbf{q})$ is the associated geometric Jacobian (also unknown in advance), and \mathbf{F}_K be the collision forces and moments exerted by the environment on the robot.

When a collision occurs, the robot dynamics (1) becomes

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} + \boldsymbol{\tau}_K =: \boldsymbol{\tau}_{\text{tot}}, \quad (3)$$

where the joint torque $\boldsymbol{\tau}_K$ associated to \mathbf{F}_K is given by

$$\boldsymbol{\tau}_K = \mathbf{J}_K^T(\mathbf{q})\mathbf{F}_K. \quad (4)$$

The generalized momentum of the robot [15] is defined as

$$\mathbf{p} = \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}. \quad (5)$$

Using (1) and (2), its time evolution is given by

$$\dot{\mathbf{p}} = \boldsymbol{\tau}_{\text{tot}} + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}). \quad (6)$$

The dynamics of the generalized momentum is thus decoupled component-wise with respect to the torques acting on the right-hand side of eq. (3).

III. COLLISION DETECTION AND REACTION

Following [16], we define the n -dimensional *residual* vector

$$\mathbf{r}(t) = \mathbf{K}_I \left[\mathbf{p}(t) - \int_0^t (\boldsymbol{\tau} + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{r}) ds \right] \quad (7)$$

with $\mathbf{r}(0) = \mathbf{0}$, a diagonal matrix $\mathbf{K}_I > \mathbf{0}$, and where $\mathbf{p}(t)$ is the robot generalized momentum at time $t \geq 0$, as defined in (5), with $\mathbf{p}(0) = \mathbf{0}^2$. Vector \mathbf{r} can be computed using the measured state $(\mathbf{q}, \dot{\mathbf{q}})$ and the commanded torque $\boldsymbol{\tau}$.

From eqs. (3), (6), and (7), the dynamics of \mathbf{r} is

$$\dot{\mathbf{r}} = -\mathbf{K}_I \mathbf{r} + \mathbf{K}_I \boldsymbol{\tau}_K, \quad (8)$$

or, component-wise in the Laplace domain,

$$\frac{r_j(s)}{\tau_{K,j}(s)} = \frac{K_{I,j}}{s + K_{I,j}}, \quad j = 1, \dots, n,$$

with the n decoupled transfer functions having unitary gains. Therefore, \mathbf{r} is ‘observing’ (a filtered version of) the joint torque resulting from the collision. Note that the computation of the residual \mathbf{r} can be made for any torque input $\boldsymbol{\tau}$ (i.e., any control command applied to the robot) and that the resulting dynamics (8) is independent from this input.

During free motion, $\mathbf{r} = \mathbf{0}$ up to measurement noise and unmodeled disturbances. In response to a generic collision, one or more components of \mathbf{r} will raise exponentially (with time constants $1/K_{I,j}$) reaching peak values that depend on the severity of the impact force \mathbf{F}_K and on its location

²We assume that the robot is initially at rest. Otherwise, we should subtract also $\mathbf{p}(0) \neq \mathbf{0}$ in the definition of $\mathbf{r}(t)$.

(through $\mathbf{J}_K^T(\mathbf{q})$). When contact is lost, all components of \mathbf{r} will rapidly return to zero. Because of these features, the residual \mathbf{r} is useful for detecting collisions without the need of extra sensors beyond those always on board of the robot (joint encoders). Further collision identification properties of \mathbf{r} are discussed in [16].

In practice, collision detection occurs when there exists at least an index j , with $j \in \{1, \dots, n\}$, for which $|r_j| > r_{\text{low},j}$. Detection threshold $r_{\text{low},j}$ should be chosen in terms of the noise characteristics of measurements, including also uncertainty in system dynamics. Dynamic thresholding can be used for avoiding false detection due to spurious spikes in noisy signals, as shown in [21].

As soon as one of the detection thresholds is exceeded, the control law in use for executing the original robotic task should be switched to one able to restore a safe operative condition. Several strategies can be devised based on the directional information contained in the residual vector \mathbf{r} .

In [16], the following simple *reflex strategy* has been proposed and experimentally verified:

$$\boldsymbol{\tau} = \mathbf{g}(\mathbf{q}) + \mathbf{K}_R \mathbf{r}, \quad (\text{diagonal}) \mathbf{K}_R > \mathbf{0}. \quad (9)$$

Beside compensating gravity [1], the idea is to let the robot (over)react to the external collision force along the same resulting direction, as seen at the level of joint torques. By combining eqs. (3) and (9), and assuming for the sake of analysis the limit case of $\mathbf{r} = \boldsymbol{\tau}_K$ (for large \mathbf{K}_I), the robot dynamics becomes

$$(\mathbf{I} + \mathbf{K}_R)^{-1} (\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})) = \boldsymbol{\tau}_K. \quad (10)$$

The robot inertial and dissipative terms, as seen by the collision torque $\boldsymbol{\tau}_K$, are thus scaled by a factor larger than unity — a lighter pushable robot is obtained. As a result, the robot bounces back in a direction (implicitly defined by eq. (9) and the robot dynamics (3)) which is the most advantageous for escaping contacts, at least locally. While this reaction guarantees a safe behavior, the original task that was being executed at the collision time is completely abandoned.

IV. DYNAMIC REDUNDANCY RESOLUTION

Let $\mathbf{x} \in \mathbf{R}^m$ be the variables needed to describe the task for the robot end-effector and suppose that the robot is kinematically redundant with respect to the task, i.e. $m < n$. The kinematic relationships between joint and task space are given by the following equations:

$$\mathbf{x} = \mathbf{f}(\mathbf{q}) \quad \dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad \ddot{\mathbf{x}} = \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}}, \quad (11)$$

where $\mathbf{J}(\mathbf{q}) = \partial \mathbf{f}(\mathbf{q}) / \partial \mathbf{q}$ is the analytical Jacobian associated to the task.

Because of redundancy, inverse kinematics possesses an infinity of solutions that can be conveniently explored at a differential level. Since we are interested in dynamic issues, redundancy will be resolved at the acceleration level. Let \mathbf{G} be a *generalized inverse* of the Jacobian \mathbf{J} , i.e., satisfying

$$\mathbf{J}(\mathbf{q})\mathbf{G}(\mathbf{q})\mathbf{J}(\mathbf{q}) = \mathbf{J}(\mathbf{q}), \quad \forall \mathbf{q}. \quad (12)$$

Common choices for \mathbf{G} include the pseudoinverse or any weighted right inverse of the Jacobian \mathbf{J} . At a given robot state $(\mathbf{q}, \dot{\mathbf{q}})$, all possible joint accelerations realizing a desired $\ddot{\mathbf{x}}$ can be written as

$$\ddot{\mathbf{q}} = \mathbf{G}(\mathbf{q})(\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}}) + (\mathbf{I} - \mathbf{G}(\mathbf{q})\mathbf{J}(\mathbf{q}))\ddot{\mathbf{q}}_0 \quad (13)$$

where $(\mathbf{I} - \mathbf{G}\mathbf{J})$ is a projection matrix in the null space of \mathbf{J} and $\ddot{\mathbf{q}}_0$ is an arbitrary joint acceleration. By construction, $\ddot{\mathbf{q}}_0$ will never contribute to the end-effector acceleration $\ddot{\mathbf{x}}$.

Setting $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})$, from eqs. (1) and (13), all control torques realizing the task acceleration $\ddot{\mathbf{x}}$ are of the form

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\mathbf{G}(\mathbf{q})(\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}}) + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{M}(\mathbf{q})(\mathbf{I} - \mathbf{G}(\mathbf{q})\mathbf{J}(\mathbf{q}))\ddot{\mathbf{q}}_0. \quad (14)$$

Moreover, since any joint acceleration $\ddot{\mathbf{q}}_0$ can be generated by a unique torque as $\ddot{\mathbf{q}}_0 = \mathbf{M}^{-1}(\mathbf{q})(\boldsymbol{\tau}_0 - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}))$, equation (14) can be further elaborated as

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{M}(\mathbf{q})\mathbf{G}(\mathbf{q}) \left[\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} \right] + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \\ &\quad + \mathbf{M}(\mathbf{q})(\mathbf{I} - \mathbf{G}(\mathbf{q})\mathbf{J}(\mathbf{q}))\mathbf{M}^{-1}(\mathbf{q})(\boldsymbol{\tau}_0 - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})) \\ &= \mathbf{M}(\mathbf{q})\mathbf{G}(\mathbf{q}) \left[\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q})\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \right] \\ &\quad + \mathbf{M}(\mathbf{q})(\mathbf{I} - \mathbf{G}(\mathbf{q})\mathbf{J}(\mathbf{q}))\mathbf{M}^{-1}(\mathbf{q})\boldsymbol{\tau}_0. \end{aligned} \quad (15)$$

It is apparent in both eqs. (14) and (15) that the torque command has two separate components: one that is needed to impose a desired task acceleration $\ddot{\mathbf{x}}$, when the robot is in the current state $(\mathbf{q}, \dot{\mathbf{q}})$, and another that does not affect $\ddot{\mathbf{x}}$ in any case, which can be parametrized by the acceleration $\ddot{\mathbf{q}}_0$ or by the torque $\boldsymbol{\tau}_0$, respectively. We note that this result holds for *any* choice of the generalized inverse \mathbf{G} .

The above developments allow to recover, as a special case, the interesting result of a dynamically consistent redundancy resolution method introduced in [20]. There, the starting point was the availability of a dynamic control strategy defined at the task level (say, for the robot end-effector) by means of a generalized force $\mathbf{F} \in \mathbf{R}^m$ that imposes a desired acceleration $\ddot{\mathbf{x}}$. This force would be then realized simply by the joint torque $\mathbf{J}^T(\mathbf{q})\mathbf{F}$, but in the presence of redundancy there is the possibility of having additional torque components at the joint level. Let \mathbf{H} be a *generalized inverse* of the Jacobian transpose \mathbf{J}^T , i.e., satisfying

$$\mathbf{J}^T(\mathbf{q})\mathbf{H}(\mathbf{q})\mathbf{J}^T(\mathbf{q}) = \mathbf{J}^T(\mathbf{q}), \quad \forall \mathbf{q}, \quad (16)$$

and note that $\mathbf{G} = \mathbf{H}^T$ satisfies eq. (12). Accordingly, any torque at the robot joint level can be decomposed as

$$\boldsymbol{\tau} = \mathbf{J}^T(\mathbf{q})\mathbf{F} + (\mathbf{I} - \mathbf{J}(\mathbf{q})^T\mathbf{H}(\mathbf{q}))\boldsymbol{\tau}_0 \quad (17)$$

where $\boldsymbol{\tau}_0$ is an arbitrary joint torque that cannot be generated by a task force $\mathbf{F} \in \mathbf{R}^m$ (geometrically, the projection matrix $(\mathbf{I} - \mathbf{J}^T\mathbf{H})$ defines $[\mathcal{R}(\mathbf{J}^T)]^\perp$). The analysis in [20] shows that there is a unique choice of \mathbf{H} in eq. (17) such that

the torque τ_0 will not affect the acceleration $\ddot{\mathbf{x}}$ (*dynamic consistency*). This is the inertia-weighted left inverse of \mathbf{J}^T

$$\begin{aligned}\mathbf{H}_M(\mathbf{q}) &= \left(\mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q})\mathbf{J}^T(\mathbf{q})\right)^{-1}\mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q}) \\ &=: \mathbf{\Lambda}(\mathbf{q})\mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q}),\end{aligned}\quad (18)$$

where $\mathbf{\Lambda}$ is the (symmetric) robot inertia in the task space.

By choosing \mathbf{G} in eq. (15) as $\mathbf{G} = \mathbf{H}_M^T = \mathbf{M}^{-1}\mathbf{J}^T\mathbf{\Lambda}$, we find the structure of the command torque to be used for dynamic redundancy resolution

$$\begin{aligned}\boldsymbol{\tau} &= \mathbf{J}^T(\mathbf{q})\mathbf{\Lambda}(\mathbf{q})(\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q})\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})) \\ &\quad + (\mathbf{I} - \mathbf{J}^T(\mathbf{q})\mathbf{H}_M(\mathbf{q}))\boldsymbol{\tau}_0,\end{aligned}\quad (19)$$

which is in fact in the form of eq. (17).

V. A COMBINED COLLISION REACTION AND TASK PRESERVATION STRATEGY

In the following, we will combine the results of Sect. III and IV so as to implement a strategy that is able to react safely to collisions while continuing to execute as much as possible the original task.

Let $\mathbf{x}_d(t) \in \mathcal{C}^2$ be the Cartesian reference trajectory of the end-effector, including both position and orientation in terms of a set of Euler angles (the full task dimension is $m = 6$), and denote the task error as $\mathbf{e} = \mathbf{x}_d - \mathbf{x}$.

As long as there is no collision detected, we will use an input-output linearizing and decoupling control in the task space based on eq. (19). However, we will cancel first gravity from the picture³. Thus, the control law will be

$$\boldsymbol{\tau} = \mathbf{J}^T(\mathbf{q})\mathbf{F}_{\text{cart}} + \mathbf{g}(\mathbf{q}) + (\mathbf{I} - \mathbf{J}^T(\mathbf{q})\mathbf{H}_M(\mathbf{q}))\boldsymbol{\tau}_0, \quad (20)$$

where \mathbf{F}_{cart} is given by

$$\begin{aligned}\mathbf{F}_{\text{cart}} &= \mathbf{\Lambda}(\mathbf{q}) \left[\ddot{\mathbf{x}}_d + \mathbf{K}_P\mathbf{e} + \mathbf{K}_D\dot{\mathbf{e}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} \right] \\ &\quad + \mathbf{\Lambda}(\mathbf{q})\mathbf{J}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q}) (\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})).\end{aligned}\quad (21)$$

In eq. (21), \mathbf{K}_P and \mathbf{K}_D are both positive definite and symmetric (typically diagonal) matrices. To avoid excessive self-motion of the redundant arm, a damping action is injected in the joint space by choosing in eq. (20) $\boldsymbol{\tau}_0 = -\mathbf{K}_{V,0}\dot{\mathbf{q}}$, with diagonal $\mathbf{K}_{V,0} > 0$. Being filtered by $\mathbf{I} - \mathbf{J}^T\mathbf{H}_M$, this damping will not affect the trajectory tracking accuracy. As a matter of fact, in the closed-loop system we have

$$\ddot{\mathbf{e}} + \mathbf{K}_D\dot{\mathbf{e}} + \mathbf{K}_P\mathbf{e} = \mathbf{0}, \quad (22)$$

so that the tracking error converges exponentially to zero.

Consider next the occurrence of a collision. The momentum-based observer (7) will output a vector \mathbf{r} and collision will be detected when

$$\text{CD} = \max \left\{ \frac{|r_1|}{r_{\text{low},1}}, \dots, \frac{|r_n|}{r_{\text{low},n}} \right\} > 1, \quad (23)$$

³This is slightly different from the original approach in [20]. Moreover, one could also cancel by feedback all non-acceleration dependent terms $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$ and obtain a simpler expression for the Cartesian controller (21) yielding the same closed-loop dynamics (22) in the task space.

where each joint has a detection threshold $r_{\text{low},i} > 0$, for $i = 1, \dots, n$. As soon as $\text{CD} > 1$, a reflex torque $\mathbf{K}_{R,0}\mathbf{r}$, with diagonal $\mathbf{K}_{R,0} > 0$, will be included in the null-space term $\boldsymbol{\tau}_0$:

$$\boldsymbol{\tau}_0 = \begin{cases} -\mathbf{K}_{V,0}\dot{\mathbf{q}}, & \text{if } \text{CD} \leq 1 \\ \mathbf{K}_{R,0}\mathbf{r} - \mathbf{K}_{V,0}\dot{\mathbf{q}}, & \text{if } \text{CD} > 1. \end{cases} \quad (24)$$

The robot configuration will then be forced away from the collision area, whereas the robot end-effector continues to track the original trajectory.

However, if the collision is not removed by this self-motion and the contact forces increase too much (due to the specific task being performed and/or to the intentional motion of the (human) obstacle), the robot should relax the trajectory execution task or even abandon it. For this, it is useful to introduce two further thresholds, $r_{\text{relax}} > \max_{i=1, \dots, n} r_{\text{low},i}$ and $r_{\text{abort}} > r_{\text{relax}}$, that will be used to rule the robot behavior in reaction to a collision, and to define

$$r_{\text{CD}} = \max_{i=1, \dots, n} |r_i|. \quad (25)$$

This signal will be continuously monitored during robot operation. If r_{CD} exceeds r_{relax} the original task is partly relaxed, typically keeping still under control the positional variables and abandoning orientation. The part of the original task that is being kept is defined by $\mathbf{x}_{\text{rel}} \in \mathbf{R}^{m_r}$, with $m_r < m$, which is a subvector of \mathbf{x} . Partial task relaxation, with its associated threshold r_{relax} , is intended for increasing the dimension of the robot null-space, so as to allow a more effective ‘internal’ reaction to the collision. Nonetheless, if r_{CD} exceeds also r_{abort} the task is completely abandoned. This upper threshold avoids damages both for a human in the workspace and for the robot itself. When the robot abandons the task, it bounces back from the collision area and then rapidly stops because of a large joint-space damping term.

This reasoning is summarized in the following control expressions (which hold for $\text{CD} > 1$):

$$\boldsymbol{\tau} = \begin{cases} \mathbf{J}^T\mathbf{F}_{\text{cart}} + (\mathbf{I} - \mathbf{J}^T\mathbf{H}_M)\boldsymbol{\tau}_0 + \mathbf{g}, & \text{if } r_{\text{CD}} \leq r_{\text{relax}} \\ \mathbf{J}_{\text{rel}}^T\mathbf{F}_{\text{rel}} + (\mathbf{I} - \mathbf{J}_{\text{rel}}^T\mathbf{H}_{M,\text{rel}})\boldsymbol{\tau}_0 + \mathbf{g}, & \text{if } r_{\text{relax}} < r_{\text{CD}} \leq r_{\text{abort}} \\ \mathbf{K}_R\mathbf{r} - \mathbf{K}_V\dot{\mathbf{q}} + \mathbf{g}, & \text{if } r_{\text{CD}} > r_{\text{abort}}, \end{cases} \quad (26)$$

where $\mathbf{K}_R > 0$, $\mathbf{K}_V > \mathbf{K}_{V,0}$, $\boldsymbol{\tau}_0$ is given by (24), and all quantities with the subscript ‘rel’ are computed similarly to those for \mathbf{x} (see, e.g., eqs. (11), (18), and (21)), but now in association with \mathbf{x}_{rel} and $\mathbf{e}_{\text{rel}} = \mathbf{x}_{\text{rel},d} - \mathbf{x}_{\text{rel}}$.

VI. SIMULATION RESULTS

The approach has been extensively tested by simulation (performed in Simulink) on a dynamic model of the KUKA/DLR lightweight arm with $n = 7$ rotational dofs, neglecting the presence of joint elasticity and dissipative effects.

For the case study shown in Fig. 1, we compare the robot behavior under variations of the threshold levels for task relaxation and abandoning. The reference motion is a

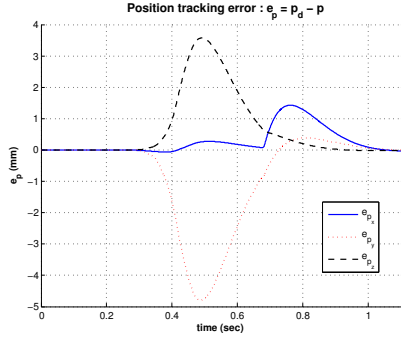


Fig. 2. No task relaxation: Position errors

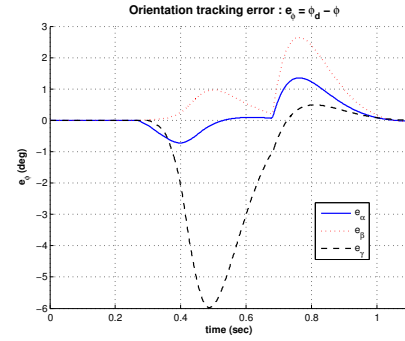


Fig. 3. No task relaxation: Orientation errors

linear trajectory on a horizontal plane of length $L = 0.89$ m, executed with a trapezoidal acceleration profile in $T = 0.9$ s, and with constant end-effector orientation ($m = 6$). For a better visualization, two intervals of 0.1 s are included before and after the actual motion. The robot has only $n - m = 1$ degree of redundancy for the complete position/orientation task, but may switch to $n - m_r = 4$ degrees if the orientation task is relaxed. The obstacle is a radially elastic sphere of 8 cm diameter, located approximately halfway and above the path. Its uniform stiffness is 10^4 N/m.

The used control parameters were: $K_I = 100 \cdot I$ for the residual in (7), $K_{R,0} = 50 \cdot I$ and $K_{V,0} = \text{diag}\{25, 20, 15, 10, 7.5, 6.5, 5\}$ for the null-space reaction and damping in (24), $K_R = 150 \cdot I$ and $K_V = 2K_{V,0}$ for the case of task abandoning in (26). The gains of the Cartesian PD in (21) were chosen as $K_{P,i} = 170$, $K_{D,i} = 20$, for all $i = 1, \dots, 6$. The detection thresholds $r_{\text{low},i}$ in eq. (23) were set at 1% of the maximum torques available at each joint, which are 200 Nm (joints 1,2), 100 Nm (joints 3,4,5), and 40 Nm (joint 6,7).

In the first test, no thresholds r_{relax} and r_{abort} were imposed and the full task is executed up to its end. Figures 2–5 show the position and orientation tracking errors, the evolution of the residuals, and the control torques. In this case, there were three major subsequent impacts on link 4, between $t = 0.25$ and $t = 0.5$ s, with a peak collision force of 55 N and a speed of 2.3 m/s at the contact point. The last isolated impact, which is a sort of robot bouncing after the major collisions, occurs around $t = 0.7$ s on link 5; since the contact point of this last collision is close to the preceding joint, the effect on residual r_5 is almost unnoticeable. In correspondence to an impact and associated robot reaction, we note a discontinuity of the control torques.

In the second test, we introduced a relaxation threshold $r_{\text{relax}} = 3$ Nm for the orientation (see Figs. 6–9). In this case, there were three clear impacts only on link 4, with a halved peak contact force of 27 N which occurs at a contact speed of 1.1 m/s. This reduction is evident from the smaller values of the residuals in Fig. 8. Note that the collision for link 5 of the previous case no longer occurs. Indeed, the orientation error in Fig. 7 grows five time bigger than before and reaches a peak of about 30° . The controller is anyway

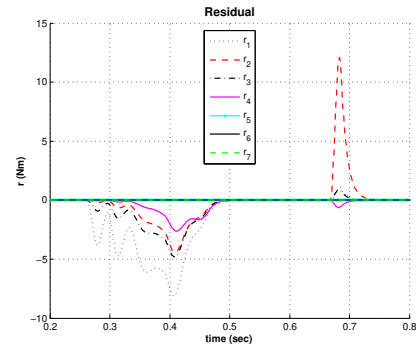


Fig. 4. No task relaxation: Residuals (expanded view for $t \in [0.2, 0.8]$ s)

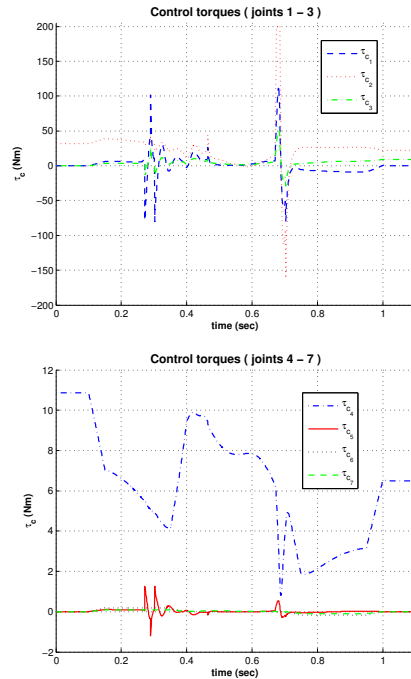


Fig. 5. No task relaxation: Control torques

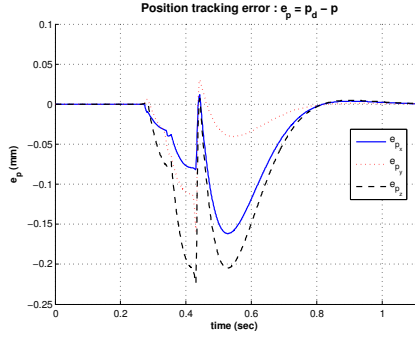


Fig. 6. Orientation task relaxation: Position errors

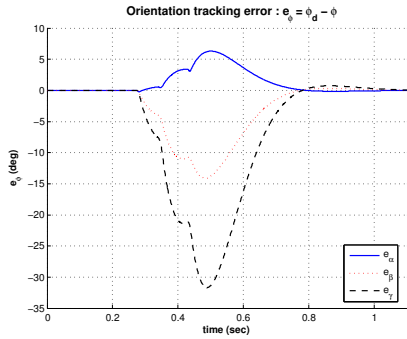


Fig. 7. Orientation task relaxation: Orientation errors

able to recover approximately zero orientation error before the end of the trajectory. On the other hand, the position error is reduced by a factor of 20, thanks to the freedom gained by the robot in accommodating reaction torques within a larger dynamic null space.

In the third and final test, we additionally introduced the threshold $r_{\text{abort}} = 4$ Nm for completely abandoning the task in case of excessive contact forces (see Figs. 10–12). Only the first two impacts occur (on link 4) and the behavior is exactly the same as in the second case until $t \approx 0.35$ s. As before, the robot relaxes task orientation after the first impact. However, since at the second impact the residual r_1 exceeds now r_{abort} , the control switches to the last expression in eq. (26), without considering the Cartesian task anymore. The robot moves away from the collision area and stops rather quickly (at $t = 0.44$ s) because of the velocity dissipation term. The position error in Fig. 10 continues to grow for a while, because the reference trajectory has not been stopped and the robot is standing still. The steady-state torques in Fig. 12 are those provided by gravity compensation.

All three tests are shown in the accompanying video clip.

VII. CONCLUSIONS

Building on our previous results on momentum-based, sensorless detection and safe robot reaction to collisions, we considered the use of kinematic redundancy of a manipulator for preserving continuous execution of a Cartesian trajectory despite of unexpected/undesired collisions along

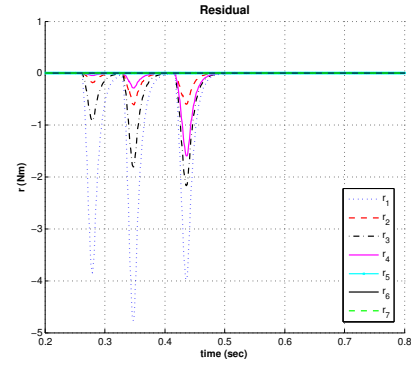


Fig. 8. Orientation task relaxation: Residuals (expanded view for $t \in [0.2, 0.8]$ s)

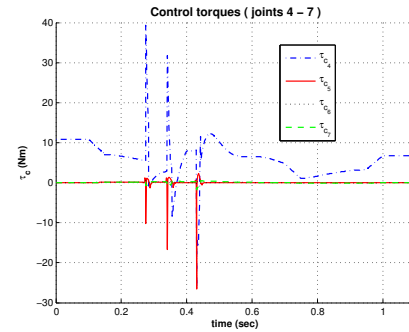
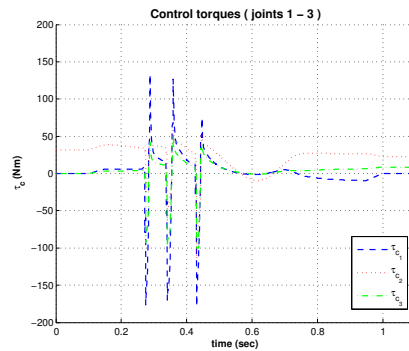


Fig. 9. Orientation task relaxation: Control torques

the structure. A redundancy resolution scheme was defined at the torque level, confining the robot reaction to a detected collision within a suitable dynamic null space not affecting task acceleration. A trade-off has been identified between limiting the contact forces and preserving accuracy in trajectory tracking. The idea of partial task relaxation proved to be useful for keeping the contact forces below a safety threshold without fully abandoning the original task.

An interesting aspect not detailed in this paper concerns the developed implementation of the observer computation in eq. (7). By a suitable modification of the recursive Newton-Euler algorithm for inverse dynamics (which would not be able to compute numerically, e.g., the term $C^T(q, \dot{q})\dot{q}$), we

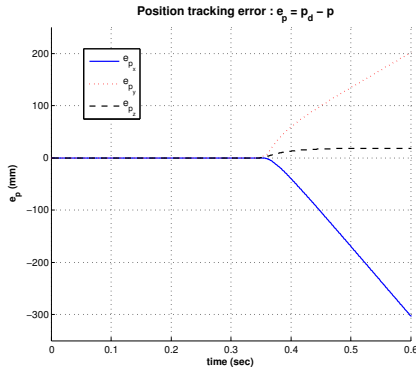


Fig. 10. Full task relaxation: Position errors

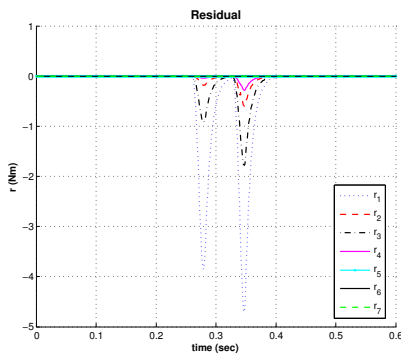


Fig. 11. Full task relaxation: Residuals

obtained a reduction by at least an order of magnitude of the time needed to evaluate the dynamic terms in the residual.

The presented results are a first step toward the goal of enhancing safety without necessarily compromising performance. We are currently planning experiments on the real KUKA/DLR lightweight arm, within the partnership of the PHRIENDS project, for which we need to extend the treatment to the presence of elasticity at the robot joints. We are also considering an alternative strategy where collision reaction is handled with higher priority w.r.t. task execution.

REFERENCES

- [1] J. Heinzmann and A. Zelinsky, "Quantitative safety guarantees for physical human-robot interaction", *Int. J. of Robotics Research*, vol. 22, no. 7/8, pp. 479–504, 2003.
- [2] O. Khatib, K. Yokoi, O. Brock, K.-S. Chang, and A. Casal, "Robots in human environments: Basic autonomous capabilities", *Int. J. of Robotics Research*, vol. 18, no. 7, pp. 684–696, 1999.
- [3] D. M. Ebert and D. D. Henrich, "Safe human-robot-cooperation: Image-based collision detection for industrial robots", in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2002, pp. 1826–1831.
- [4] V. Lumelsky and E. Cheung, "Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulator", *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 23, no. 1, pp. 194–203, 1993.
- [5] G. Hirzinger, A. Albu-Schäffer, M. Hähle, I. Schaefer, and N. Sporer, "On a new generation of torque controlled light-weight robots", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2001, pp. 3356–3363.

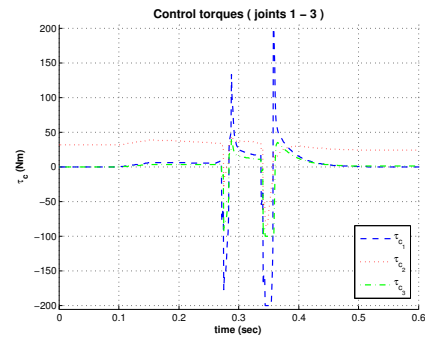


Fig. 12. Full task relaxation: Control torques for joints 1 to 3

- [6] K. Ikuta, H. Ishii, and M. Nokata, "Safety evaluation method of design and control for human-care robots", *Int. J. of Robotics Research*, vol. 22, no. 7/8, pp. 281–297, 2003.
- [7] A. Albu-Schäffer and G. Hirzinger, "A globally stable state-feedback controller for flexible joint robots", *Advanced Robotics*, vol. 15, no. 8, pp. 799–814, 2001.
- [8] A. Bicchi and G. Tonietti, "Fast and soft arm tactics: Dealing with the safety-performance tradeoff in robot arms design and control", *IEEE Robotics and Automation Mag.*, vol. 11, no. 2, pp. 22–33, 2004.
- [9] K. Suita, Y. Yamada, N. Tsuchida, K. Imai, H. Ikeda, and N. Sugimoto, "A failure-to-safety 'kyozon' system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1995, pp. 3089–3096.
- [10] S. Takakura, T. Murakami, and K. Ohnishi, "An approach to collision detection and recovery motion in industrial robot", in *Proc. 15th Annual Conf. of IEEE Industrial Electronics Society (IECON89)*, Boston, MA, 1989, pp. 421–426.
- [11] H.-B. Kuntze, Ch. W. Frey, K. Giesen, and G. Milighetti, "Fault tolerant supervisory control of human interactive robots", in *Proc. IFAC Work. on Advanced Control and Diagnosis (ACD03)*, Duisburg, D, 2003, pp. 55–60.
- [12] K. Kosuge and T. Matsumoto, "Collision detection of manipulator based on adaptive control law", in *Proc. IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, 2001, pp. 117–122.
- [13] S. Morinaga and K. Kosuge, "Collision detection system for manipulator based on adaptive impedance control law", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 1080–1085.
- [14] S. Morinaga and K. Kosuge, "Compliant motion control of manipulator's redundant dof based on model-based collision detection system", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2004, pp. 5212–5217.
- [15] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 1011–1016.
- [16] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight robot arm", in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 1623–1630.
- [17] L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators*, Springer, London, 2nd edition, 2000.
- [18] J. M. Hollerbach and K. C. Suh, "Redundancy resolution of manipulators through torque optimization", *IEEE J. of Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 1987.
- [19] I. D. Walker, "Impact configurations and measures for kinematically redundant and multiple armed robot systems", *IEEE Trans. on Robotics and Automation*, vol. 10, no. 5, pp. 670–683, 1994.
- [20] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework", *Int. J. of Robotics Research*, vol. 14, no. 1, pp. 29–36, 1995.
- [21] A. De Luca and R. Mattone, "An adapt-and-detect actuator FDI scheme for robot manipulators", in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2004, pp. 4975–4980.