SAPHARI

SAFE AND AUTONOMOUS PHYSICAL HUMAN-AWARE ROBOT INTERACTION

SEVENTH FRAMEWORK PROGRAMME

Project funded by the European Community's 7th Framework Programme (FP7-ICT-2011-7)
Grant Agreement ICT–287513

Deliverable D3.1.1

Combined collision avoidance, detection, and reaction

Deliverable due date: 1 April 2014	**Actual submission date:** 25 July 2014
Start date of project: 1 November 2011	**Duration:** 48 months
Lead beneficiary: UNIROMA1	**Revision:** Final

Nature: R	**Dissemination level:** PU

R = Report
P = Prototype
D = Demonstrator
O = Other
	PU = Public
	PP = Restricted to other programme participants (including the Commission Services)
	RE = Restricted to a group specified by the consortium (including the Commission Services)
	CO = Confidential, only for members of the consortium (including the Commission Services)

Executive Summary

This deliverable of work package WP3 summarizes the work performed in Task 3.1 on developing and combining in a single control architecture various techniques for robot collision avoidance in highly dynamic environments and physical collision detection and robot reaction to guarantee safety. The two features of collision avoidance and collision detection/reaction complement themselves in allowing human-robot coexistence, where the robot and the human share closely the same workspace. These features are also the prerequisite for human-robot physical collaboration (which is the subject of a different Task 3.4), which can be integrated in this same framework. The results presented in this document use the sensing capabilities and real?time processing developed in WP4 and provides a layout of the low-level control capabilities used in WP6. Some of the results on collision detection have been used also for kinesthetic teaching within WP5.

Table of contents

1 Introduction....3

2 Collision Avoidance....5

2.1 Collision avoidance in the depth space [UNIROMA1]....5

2.2 Predictive motion planning and control [IOSB]....6

2.3 Dynamical systems modulation [TUM]....8

3 Collision Detection....10

3.1 Residuals....10

3.2 Distinguishing intentional contacts from unforeseen collisions....11

4 Collision Reaction....13

1 Introduction

The proper handling of the different modalities of physical interaction between humans and robots is one of the requirements in service and industrial robotics that the SAPHARI project is addressing. To fulfill this requirement, human safety must always be guaranteed, both when the human and robot are supposed to just work side-by-side without contacts or exchange of forces and when they should strictly collaborate in the execution of complex task. Setting the pathway to a safer collaboration will allow merging the adaptability skills of humans with the precision and high payload capability of robots.

Passive safety was one of the core guidelines of the former FP6 project PHRIENDS (2006-09), mostly devoted to the development of lightweight robot structures with (variable) mechanical compliance, and their low-level control intended to detect collisions and recover performance. In the SAPHARI project, a new physical Human-Robot Collaboration (pHRC) framework has been developed. This framework is conceived as nested layers of consistent behaviors that the robot must guarantee and accomplish: Safety, Coexistence, and Collaboration (see Fig. 1).

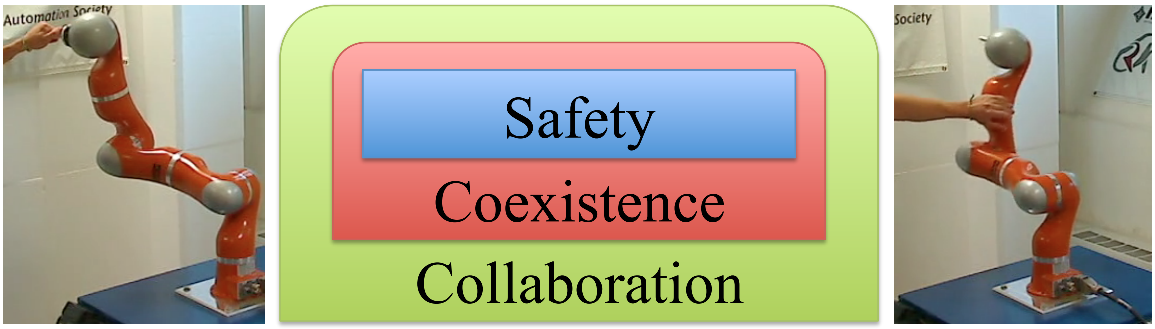


Figure 1: Stack of nested consistent behaviors for pHRC [4].

Safety is the inherent and most important feature of a robot that has to work close to human beings. Classical solutions for preserving safety in industrial environments, i.e., using cages or stopping the robot in the presence of humans, are inappropriate for many pHRI tasks. The latest industrial safety standards and the technical specification ISO/TS15066 in preparation limit the total instantaneous power of a robotic system in operation and determine a maximum speed of moving robots in human environments. However, they may still fall short in some desired professional or personal service applications.

Coexistence is the robot capability of sharing the workspace with other entities, most relevant with humans in which case human safety requirements must be consistently guaranteed (i.e., safe coexistence). An example of coexistence is when a robot and a human operator work together on the same object, without ever requiring mutual contact or coordination of actions.

Collaboration is the robot feature of performing a complex task with direct human interaction and coordination in two different, not mutually exclusive modalities: physical collaboration, where there is an explicit and intentional contact with exchange of forces, and contact-less collaboration, where coordinated actions are guided by an exchange of information (such as through gestures or voice commands). We refer to safe physical collaboration when this activity is consistent with safe coexistence, i.e., when safety and coexistence features are guaranteed during physical collaboration phases.

When collaboration is not the main concern (for this, see the project work in Task 3.4), the safest possible solution is to avoid any undesired contact (collision) with humans or environment obstacles. To this end, in SAPHARI we have explored different solutions for real-time sensor-based collision avoidance, as reported in Sec. 2. Unfortunately, collision avoidance may fail due to the dynamic limitations of sensors and robot motion, e.g., if the human moves faster than the robot can sense or counteract. In this event, it is still possible to detect a physical contacts (Sec. 3), more specifically distinguishing between intentional contacts and unforeseen collisions (Sec. 3.2), so as to or react to collisions by immediately removing the robot from the collision area (Sec. 4).

In this combined treatment of collision avoidance, collision detection, and collision reaction:

- multiple sensing capabilities and real-time processing are those developed in Task 4.1;
- sensor-based collision avoidance has used mainly a Kinect sensor connected to KUKA robot controllers (via the FRI or RSI interface), using the depth information to realize reactive behaviors that anticipate relatively slow human-robot potential collisions;
- for the detection of unescapable/fast collisions, proprioceptive sensing (encoders, motor currents data or torque sensors at the robot joints) has been used in the evaluation of residual signals that monitor the generalised momentum of the robot;
- the reaction strategies considered here are mainly those originally developed in [1,9] for the DLR LWR-III robot, reimplemented in the current robotic systems and control architectures;
- further robot reaction strategies following a recognized intention of collaboration (see, e.g., [13,14]) are not included here being part of Task 3.4, but would fit within the same control architecture;
- complete validations have been conducted on different instances of the KUKA LWR robotic platform, at DLR, IOSB, TUM, UNINA, and UNIROMA1, while few results have been obtained also on a small-size industrial KUKA KR5 robot with closed control architecture, see [8];
- many results are supported also through videos, which are available on the project website (as well as on YouTube channels).

2 Collision Avoidance

Avoiding an undesired collision is indeed the safest approach for human-robot coexistence. Thus, different approaches have been explored in SAPHARI. All of them have been tested on a KUKA LWR arm with 7-DOF.

2.1 Collision avoidance in the depth space [UNIROMA1]

The main information needed by any on-line collision avoidance algorithm is the relative distance between the robot and some obstacle in its workspace, as acquired by exteroceptive sensors either fixed in the environment or mounted on the robot. In this respect, without any extra information about the environment and the obstacles, also occluded points (points behind a detected obstacle) have to be considered as obstacles (see Fig. 2).

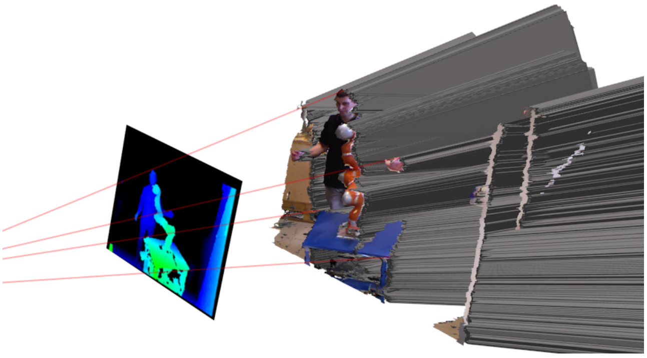


Figure 2: Illustration of the occluded points that forms the gray area generated by a depth sensor.

The performance of any algorithm will depend also on the fast processing capability of the sensor data. In [6, 7], we have proposed a new efficient method for estimating obstacle-to-robot distances that works directly in the depth space associated to a depth sensor (e.g., a Kinect monitoring the HRI scene). This method allows to take into consideration also occluded points and pixel sizes (the so-called frustum associated to a pixel, as shown in Fig. 3).

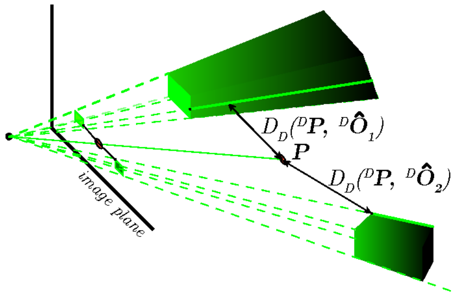

$D_D({}^D\mathbf{P},\ {}^D\hat{\mathbf{O}}_1)$
$\mathbf{P}$
$D_D({}^D\mathbf{P},\ {}^D\hat{\mathbf{O}}_2)$
image plane

Figure 3: Depth space distance evaluation to a point of interest when also the pixel dimension is taken into account. The two possible cases of obstacle depth larger or smaller than the depth of the point of interest are shown.

Once the robot-obstacle distances have been evaluated, they are used to modify on-line the current trajectory of the manipulator so as to avoid collision. Many different approaches for obstacle avoidance can be used [10, 12, 17]. In [6], we have presented a simple but effective collision avoidance algorithm based on a modification of artificial potential fields, including also a pivoting strategy to escape local minima. An example of collision avoidance in close human-robot coexistence is shown in Fig. 4. The robot should avoid the human as well as the other (static) obstacles detected on line in the environment. See [6, 7] for more details.

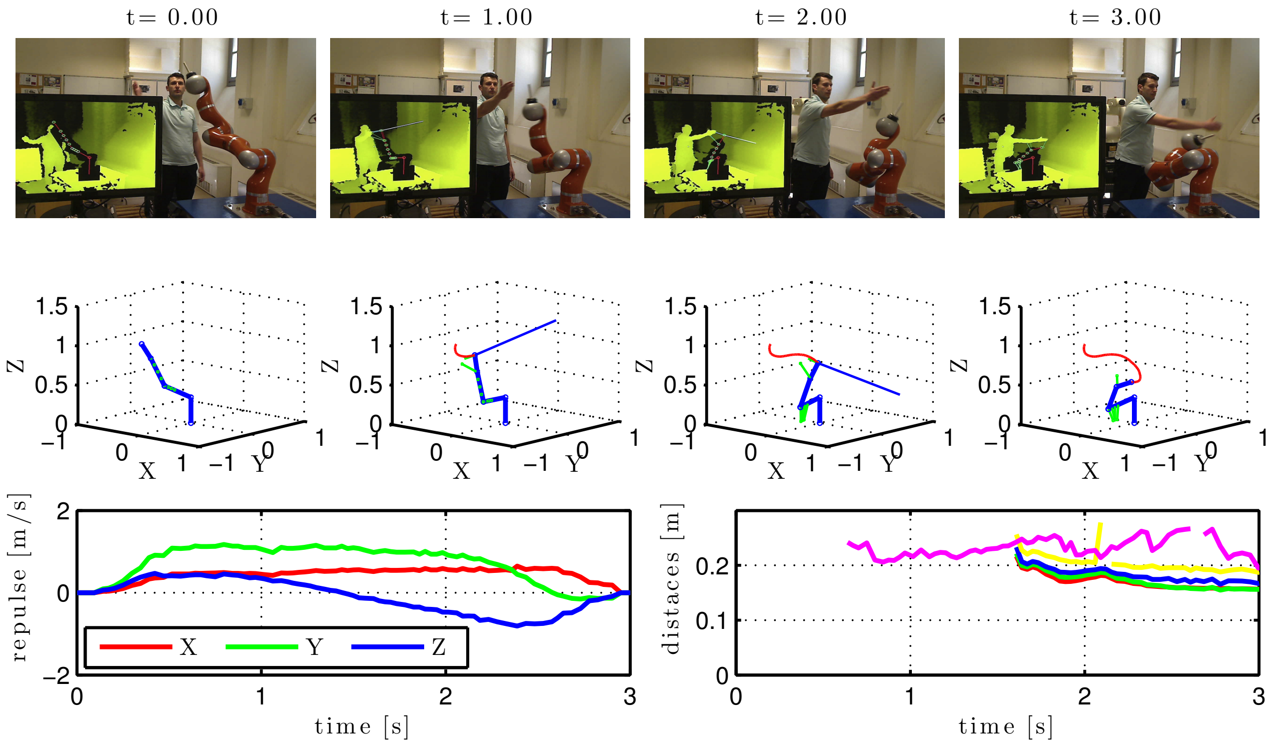


Figure 4: Example of collision avoidance using the depth space, with a human operator trying to touch the robot end-effector. The first and second rows refer to four instants of a typical experiment ($t = 0, 1, 2, 3$ s). Snapshots of the experiment are shown in the first row, while the second row illustrates the main quantities of interest: end-effector trajectory [red]; distances between control points and nearest obstacles [green]; end-effector repulsive velocity [blue]. The last row shows the evolution of the components of the end-effector repulsive velocity [left] and of the minimum distances for a number of control points on the robot surface [right].

2.2 Predictive motion planning and control [IOSB]

Based on information on static obstacles, an initial robot trajectory from the current to the goal position is calculated by means of an A*-search algorithm in the configuration space, where a discretization of valid configurations is computed offline and augmented with time (Fig. 5).

The three-dimensional work space of the robot is divided into cells with a fixed edge length. The Cartesian space occupied by the robot in a certain configuration is determined by approximating the robot by several cubes (Fig. 6)

Moving obstacle are detected using a external sensor, and their motion is predicted using a Kalman filter. This prediction, together with the robot representation, is used to update the configuration-time map. Thus, the robot trajectory is periodically updated to take unforeseen changes into account, similar to a model predictive control approach. Therefore, it is essential that the trajectory is computed in real-time.

Figure 7 shows a simulation of the collision avoidance obtained with this approach. For further details please refer to Milestone MS10 [15].

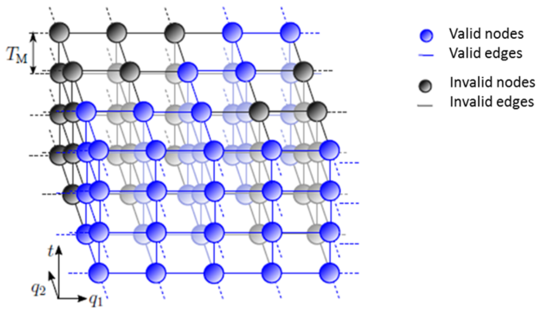


Figure 5: Discrete configuration-time space for a 2-DOF robot.

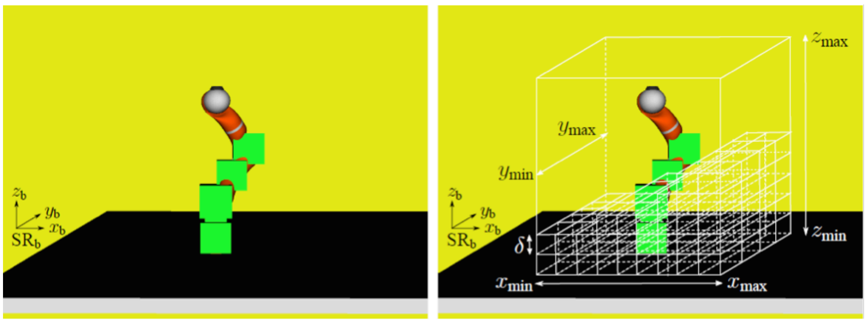


Figure 6: Modeling of the Cartesian space: Approximation of the robot by cubes [left] and sub-dividing the robot workspace into cubic cells [right].

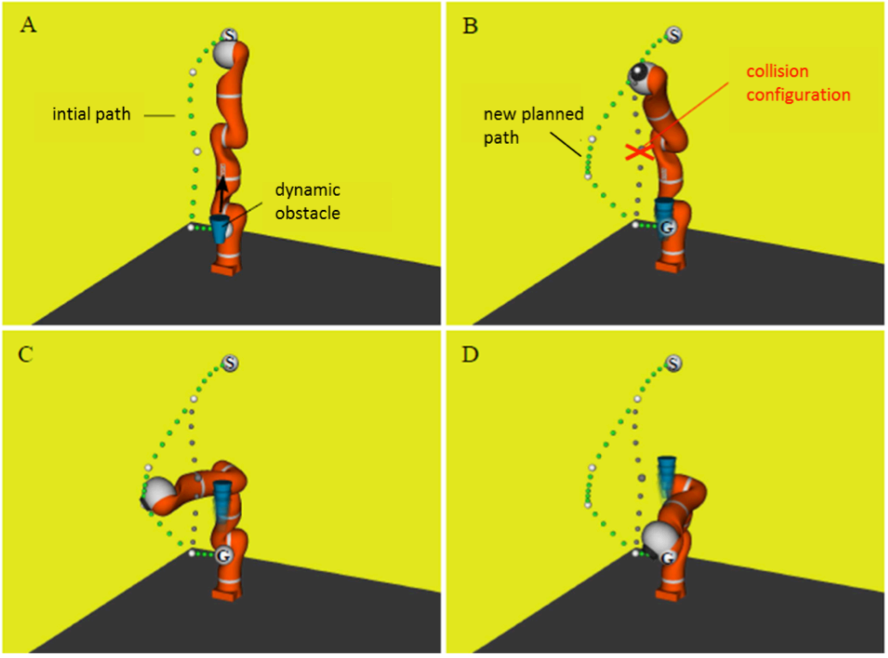


Figure 7: Example of collision avoidance: (A) initial path, (B) predicted collision and updated path, (C ,D) collision-free robot motion.

2.3 Dynamical systems modulation [TUM]

Another approach to reactive collision avoidance is based on the assumption that the robot motion is generated using a first-order, in general time-variant, Dynamical System (DS):

$$\dot{\boldsymbol{p}} = \boldsymbol{f}(\boldsymbol{p}, t), \tag{1}$$

where $\boldsymbol{p} \in \mathbb{R}^d$ and $\dot{\boldsymbol{p}} \in \mathbb{R}^d$ represent the robot end-effector position and velocity, respectively. Driving robots with a DS has several advantages in terms of robustness to external perturbations, such as unexpected contacts or changes in the goal/initial position. The DS structure allows, for example, to easily implement a human-based velocity scaling algorithm for safe HRI [19].

To avoid collisions, a local deformation of the motion path is introduced. This deformation is obtained by modulating the DS with a suitable modulation matrix $\boldsymbol{M}(\boldsymbol{p})$ by:

$$\dot{\boldsymbol{p}}(t) = \boldsymbol{M}(\boldsymbol{p})\boldsymbol{f}(\boldsymbol{p}, t)\ . \tag{2}$$

Matrix $\boldsymbol{M}$ reduces the robot velocity along the normal to the obstacle surface, splitting the motion on the tangential directions. This generates an escaping motion that preserves the equilibrium point of the modulated DS [16]. On the obstacle surface, the velocity in the normal direction is zero. Hence, the robot will never penetrate a convex obstacle [11] and, under certain assumptions on the degree of concavity [17], neither a concave obstacle.

To calculate the modulation matrix in (2), we assume that the normal vector $\hat{\boldsymbol{n}}(\bar{\boldsymbol{p}})$ to the obstacle surface is defined for all the points $\bar{\boldsymbol{p}}$ of the surface. Then, a tangential hyperplane can be defined at each point on the surface. Let the matrix $\boldsymbol{V}(\bar{\boldsymbol{p}}) = [\hat{\boldsymbol{n}}(\bar{\boldsymbol{p}})\ \ \hat{\boldsymbol{v}}_1(\bar{\boldsymbol{p}})\ \cdots\ \hat{\boldsymbol{v}}_{d-1}(\bar{\boldsymbol{p}})]$, be an orthonormal basis of the d-dimensional space, where $[\hat{\boldsymbol{v}}_1(\bar{\boldsymbol{p}})\ \cdots\ \hat{\boldsymbol{v}}_{d-1}(\bar{\boldsymbol{p}})]$ is a base of the tangential hyperplane. Introduce $\Phi' = \Phi(\bar{\boldsymbol{p}}) - \alpha$ as the distance between the robot and the surface of the obstacle, including a positive scalar α as a *safety margin*, where $\bar{\boldsymbol{p}}$ is the point of minimum distance. Define a diagonal matrix $\boldsymbol{E}(\bar{\boldsymbol{p}}) = diag\,\{\lambda_1(\bar{\boldsymbol{p}}), \ldots, \lambda_d(\bar{\boldsymbol{p}})\}$, with:

$$\begin{aligned}
\lambda_1 &= \begin{cases} 1 - \dfrac{1}{(\Phi'+1)^{1/\rho}} & \dot{\boldsymbol{p}}^T\bar{\boldsymbol{p}} < 0 \text{ or } m = 1 \\ 1 & \dot{\boldsymbol{p}}^T\bar{\boldsymbol{p}} \geq 0 \text{ and } m = 0 \end{cases} \\
\lambda_i &= 1 + \frac{1}{(\Phi'+1)^{1/\rho}} \qquad\qquad i = 2, 3, \ldots, d.
\end{aligned} \tag{3}$$

In (3), the positive scalar ρ is the *reactivity* parameter, used to change the magnitude of the modulation, and the boolean variable $m = 0, 1$ is used to interrupt the modulation ($m = 1$) after passing the obstacle ($\dot{\boldsymbol{p}}^T\bar{\boldsymbol{p}} < 0$). Finally, the modulation matrix can be calculated as

$$\boldsymbol{M}(\bar{\boldsymbol{p}}) = \boldsymbol{V}(\bar{\boldsymbol{p}})\boldsymbol{E}(\bar{\boldsymbol{p}})\boldsymbol{V}(\bar{\boldsymbol{p}})^{-1}\ . \tag{4}$$

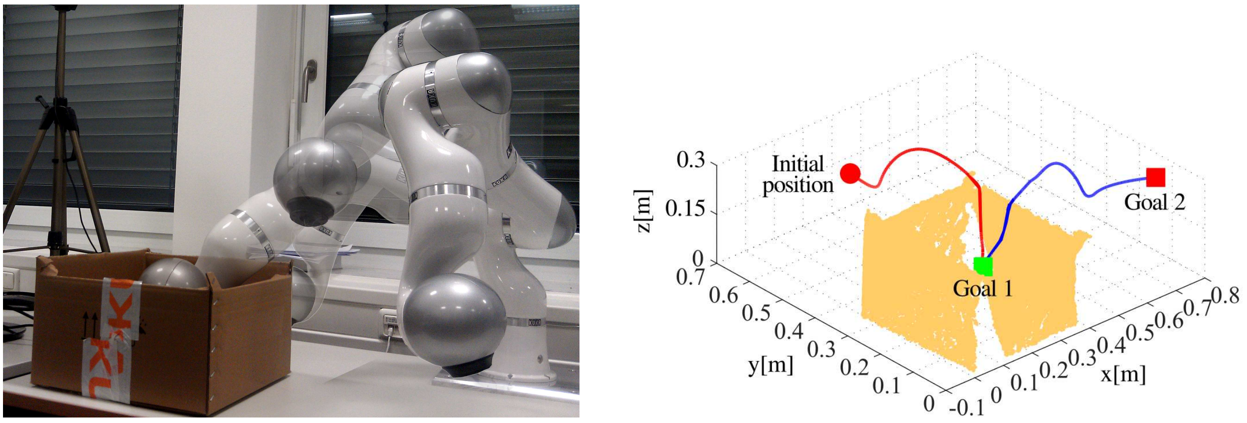
The described distance-based modulation can be directly applied in the same way, no matter how many obstacles exist in the work space. We simply calculate the distance from the closest object and the normal at the point of minimum distance. So, the number of objects affect the performance of the algorithm only mildly.

In the case of moving obstacles, the modulation in (2) does not guarantee impenetrability. Indeed, increasing the *reactivity* or the *safety margin* does not guarantee either to find a collision-free path to the goal in dynamic environments [18]. Let us consider one moving obstacle (when multiple obstacles exist in the work space, consider simply the closest or most dangerous one), with translational and angular velocities $\dot{\boldsymbol{p}}_t$ and $\dot{\boldsymbol{p}}_a$. In order to guarantee impenetrability, the modulated system becomes:

$$\dot{\boldsymbol{p}} = \boldsymbol{M}(\bar{\boldsymbol{p}})(\boldsymbol{f}(\boldsymbol{p}, t) - \dot{\boldsymbol{p}}_O) + \dot{\boldsymbol{p}}_O = \boldsymbol{M}(\bar{\boldsymbol{p}})\boldsymbol{f}(\boldsymbol{p}, t) + (\boldsymbol{I} - \boldsymbol{M}(\bar{\boldsymbol{p}}))\dot{\boldsymbol{p}}_O \tag{5}$$

where $\dot{\boldsymbol{p}}_O = \dot{\boldsymbol{p}}_t + \dot{\boldsymbol{p}}_a \times \bar{\boldsymbol{p}}$, $\boldsymbol{I}$ is the d-dimensional identity matrix and $\boldsymbol{M}(\bar{\boldsymbol{p}})$ is calculated using (4). The term $\boldsymbol{M}(\boldsymbol{f} - \dot{\boldsymbol{p}}_O)$ is a modulation in the obstacle coordinate system, that guarantees the impenetrability in the current instant. The additional term $\dot{\boldsymbol{p}}_O$ puts the system in the robot coordinate system and guarantees collisions avoidance in the following time instant.

The effectiveness of the proposed approach can be observed in two validation experiments with a KUKA LWR. In the first experiment the robot has to reach two goal positions, one of which is located at the center of a box of size 40cm×35cm×20cm. One side of the box is open (Fig. 8(a)). Starting from a point outside the box, the robot is guided to the first goal $\boldsymbol{g}_1$ inside it by the system $\dot{\boldsymbol{p}}(t) = k(\boldsymbol{g}_1 - \boldsymbol{p}(t)),\ k = 2$. Then, starting from $\boldsymbol{g}_1$ the robot comes out of the box and reaches the second goal $\boldsymbol{g}_2$ of $\dot{\boldsymbol{p}}(t) = k(\boldsymbol{g}_2 - \boldsymbol{p}(t)),\ k = 2$. A collisions free path is found by modulating this switching linear DS, as represented in Fig. 8(b).



(a) Task execution

(b) 3D view

Figure 8: The task of going into and out of a box. Given the point cloud (yellow points) of the box and two goal positions, the robot is driven into (red line) and out of (blue line) a box, by modulating a switching linear DS.

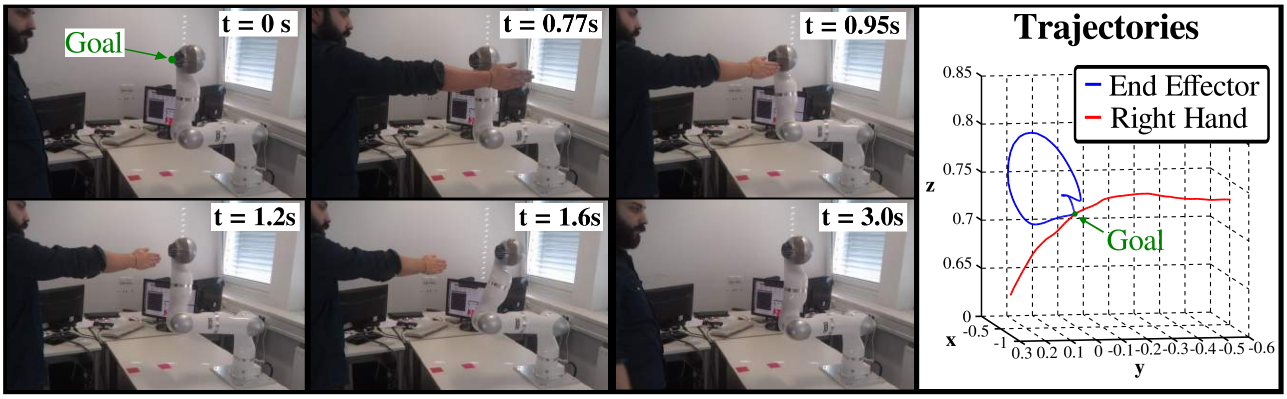


Figure 9: The robot has to avoid collisions with the human and return to the initial position. The norm of the estimated hand velocity comes from 0.45 to $0.6\ m/s$.

In the second experiment the robot has to keep the end-effector in a fixed position while the user tries to hit it. The robot trajectory is generated integrating the linear DS $\dot{\boldsymbol{p}} = 3(\boldsymbol{g} - \boldsymbol{p})$, where $\boldsymbol{g}$ is the initial reference position. The human is tracked at $30Hz$ using an RBG-D camera and the *OpenNI* library (www.openni.org). A Kalman filter is used to reduce the noise on the hand position estimation and to estimate the hand velocity. To implement the filter, we assumed a constant velocity in each time step. The robot is removed from the sensor depth map using a shader-based filter (github.com/jhu-lcsr-forks/realtime_urdf_filter). The robot configuration at six different time instants, together with the robot end-effector and the human hand trajectories, are shown in Fig. 9.

3 Collision Detection

Detection of physical collisions is the innermost feature for a safe control of the robot behavior, since collision avoidance cannot be always guaranteed in unpredictable dynamic environments. To be useful, real-time collision detection must be very efficient, in order to allow as soon as possible a fast robot reaction. This limits the use of exteroceptive sensors, such as cameras, due to their low bandwidth. Moreover, achieving detection based only on basic proprioceptive sensors is very appealing in terms of on-board availability (without workspace restrictions) and limited costs.

3.1 Residuals

We have used the residual-based method originally proposed in [5] for estimating the effect of external forces arising in a collision during robot motion. This method needs an accurate knowledge of the dynamic model terms, but uses only robot joint position measurements, as given, e.g., by encoders. For a robot with rigid links and joints, the motor torque $\boldsymbol{\tau}$ applied to the robot is also needed, as commanded by the user, e.g., by imposing motor currents, and fully independently of its origin (feedforward, feedback, model-based or PID law, and so on). A powerful result is that the same detection and isolation features are obtained also in the case of robots wirh flexible joints (of constant or variable stiffness), by replacing the commanded motor torque $\boldsymbol{\tau}$ with the joint elastic torque $\boldsymbol{\tau}_J$, as measured by joint torque sensors [1].

For a rigid robot, based on its generalized momentum

$$\boldsymbol{p} = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{6}$$

we define the residual vector $\boldsymbol{r} \in \mathbb{R}^n$ as follows:

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

with $\boldsymbol{r}(0) = \boldsymbol{0}$, a gain matrix $\boldsymbol{K}_I = diag\left\{K_{I,1}, \ldots, K_{I,n}\right\} > 0$, Coriolis/centrifugal matrix $\boldsymbol{C}$ (of Christoffel symbols) and gravity vector $\boldsymbol{g}$. It is easy to show that each component r_i, $i = 1, \ldots, n$, is an independent, first-order, unity-gain filtered version of the component $\tau_{ext,i}$ of the unknown joint torque $\boldsymbol{\tau}_{ext}$ resulting from a collision that occurs at any place along the robot structure.In the ideal condition of $\boldsymbol{K}_I \to \infty$, which means in practice that the gains should be taken as large as possible, we have

$$\boldsymbol{r} \simeq \boldsymbol{\tau}_{ext}. \tag{8}$$

Thus, a physical collision will be detected as soon as some norm $\|\boldsymbol{r}\| > r_{thres}$, being $r_{thres} > 0$ a suitable scalar threshold used to prevent false detection due to measurement noise and/or model uncertainties acting on $\boldsymbol{r}$. Note also that when a contact/collision is over, the residual $\boldsymbol{r}$ will return to zero at an exponential rate.

As mentioned, for a robot with elastic joints (as the KUKA LWR arm), a similar residual can be built based on the so-called link-side dynamics of the robot. The following residual has the same properties as (7):

$$\boldsymbol{r}(t) = \boldsymbol{K}_I \left(\boldsymbol{p}(t) - \int_0^t \left(\boldsymbol{\tau}_J + \boldsymbol{C}^T(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{r} \right) ds \right), \tag{9}$$

More details can be found in [1, 5, 9].

3.2 Distinguishing intentional contacts from unforeseen collisions

According to our framework for pHRC, safety must be guaranteed even when a contact between human and robot occurs. To establish the pathway toward physical collaboration, the robot has to distinguish, after a detection phase, between an unforeseen (thus, undesired) collision and an intentional physical contact (which may signalize the human desire to collaborate), and then it should react accordingly. Indeed, also in this case detection and reaction have to be as fast as possible.

In [8], we have proposed a signal-based method that used two filters working in parallel on the robot motor currents in order to separate *soft* from *hard* type of contacts. In that case, we worked on the motor currents since no knowledge of the robot dynamics was available and there was no possibility of imposing torque or current commands.

The idea was to apply a High-Pass Filter (HPF) and a Low-Pass Filter (LPF) to the motor currents. In fact, in most robot tasks the desired motion is smooth and repetitive in nature and the frequency content of the associated commands (in feedback of feedforward mode) is limited and predictable in advance. On the other hand, noise as well as the effect of hard (and thus unintentional) collisions typically appear in the high-frequency range of closed-loop control signals. A LPF cleans the signals from high-frequency noise, and possibly from the effect of hard collisions, while retaining the command frequencies needed for executing the motion task in a limited bandwidth. On the other hand, soft contacts between the robot and a human (intended for collaboration) may be still recognized in the filtered signal. A HPF removes components that are slowly varying in time, down to constant offsets. The filtered signal will still be very noisy, but is mostly sensitive to the effect of hard impacts (i.e., undesired/unexpected collisions). Therefore, by applying simultaneously a HPF and a LPF it should be possible in principle to distinguish between an intentional contact and an unforeseen collision. This guess was confirmed by the detailed results presented in [8].

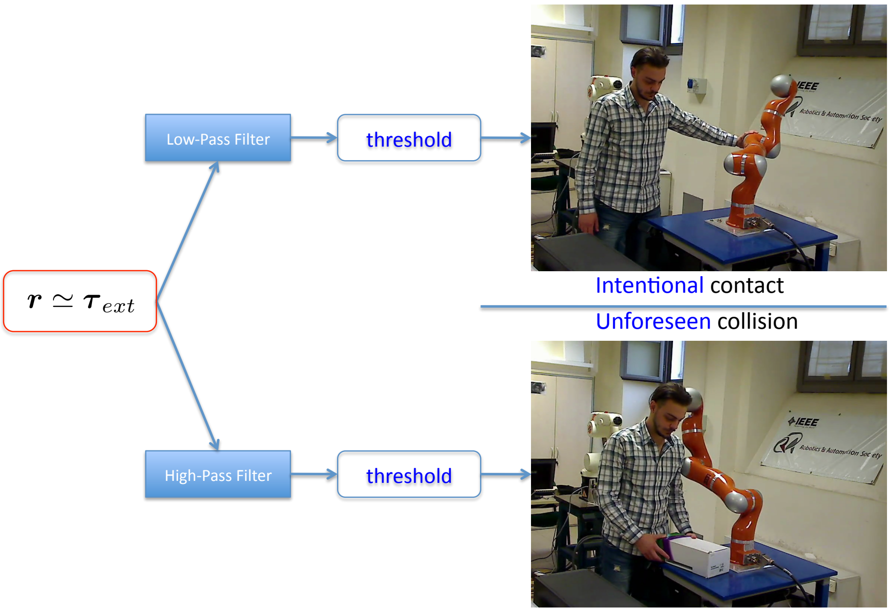


Figure 10: By using signal processing by HPF and LPF, unforeseen collisions can be distinguished from intentional contacts.

The same concept of double filtering could be extended also to the residual signal, as generated by (7) or (9 and sketched in Fig. 10. This signal contains in fact much more structured (and model-based) information with respect to the motor currents and is well suited to separate behaviors based on frequency contents, especially in position-controlled robots like the KUKA LWR. However, we found out that the most reliable results with a residual-based approach were obtained by exploiting the (low-pass) filtering action of the residual itself. This is described in the following (yet unpublished results).

From (7), the dynamics of $\boldsymbol{r}$ is, component-wise in the Laplace domain,

$$W_j(s) = \frac{r_j(s)}{\tau_{ext,j}(s)} = \frac{K_{I,j}}{s + K_{I,j}} = \frac{1}{1 + s\lambda_j}, \qquad j = 1, \ldots, n, \tag{10}$$

which represent n decoupled *low-pass* transfer functions with unitary gains. The filter constants $\lambda_j = 1/K_{I,j}$ corresponds to a *cut-off* frequencies $f_{cut,j} = 1/2\pi\lambda_j = K_{I,j}/2\pi$. Therefore, the bandwidth of the $j - th$ filter is proportional to the corresponding value of $K_{I,j}$ in the gain matrix $\boldsymbol{K}_I$: the larger is the gain, the easier is to recognize *fast* contacts, such as undesired collisions.

To distinguish in efficient way the nature of the contact, we have considered two residual vectors $\boldsymbol{r}_L$ and $\boldsymbol{r}_H$ working in parallel and having different bandwidths, as obtained from two different diagonal gain matrices $\boldsymbol{K}_L$ and $\boldsymbol{K}_H$ respectively, with $K_{H,j} > K_{L,j}$, for $j = 1, \ldots, n$. *Soft* contacts will excite in the same way both residuals $\boldsymbol{r}_L$ and $\boldsymbol{r}_H$. On the other hand, *hard* and *fast* contacts will be excite $\boldsymbol{r}_H$ more than $\boldsymbol{r}_L$. For sake of simplicity, consider a generic joint j and the associated residual components $r_{L,j}$ and $r_{H,j}$. Their frequency responses $W_{L,j}(f)$ and $W_{H,j}(f)$ are shown in Figure 11. Denoting by f_{cont} the generic frequency content in a human-robot contact, if f_{cont} belongs to both residuals bandwidths, i.e., $f_{cont} \leq f_{cut,L,j} (< f_{cut,H,j})$, the contact will be considered as intentional (green area in Fig. 11). Otherwise, if $f_{cut,L,j} < f_{cont} \leq f_{cut,H,j}$, the contact will be considered as an undesired collision (red area in Fig. 11). Finally, if $f_{cont} > f_{cut,H,j}$ the contact will not be perceived, at least instantaneously, by the robot.

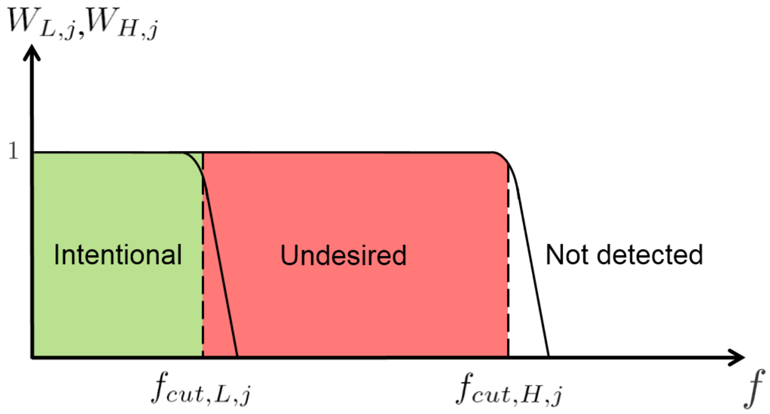


Figure 11: Frequency responses of $\boldsymbol{r}_L$ and $\boldsymbol{r}_H$ and classification of contact types.

At this stage, in order to uniquely discriminate the two type of contacts, we define a signal $\sigma(t)$ that indicates the presence or not of an undesired collision. It is computed as

$$\sigma(t) = \begin{cases} 1 & \text{if } \exists j \in \{1, \ldots, n\} : \left| \frac{\bar{r}_{H,j}(t)}{r_{L,j}(t)} \right| \geq \sigma_{th,j} \\ 0 & \text{else,} \end{cases} \tag{11}$$

where

$$\bar{r}_{H,j}(t) = \begin{cases} r_{H,j}(t) & \text{if } |r_{H,j}(t)| \geq r_{th,j} \\ 0 & \text{else.} \end{cases}$$

Since at the cut-off frequency $f_{cut,L,j}$ we have an attenuation of 3dB, to be consistent with Fig. 11, we have chosen as threshold for classification a value $\sigma_{th,j} \geq 10^{-3/20} \simeq 1.41$. Moreover, since a residual with higher cut-off frequency is more sensitive to torque measurement noise, a threshold $r_{th,j} > 0$ has been introduced to prevent false collision detections during robot motion.

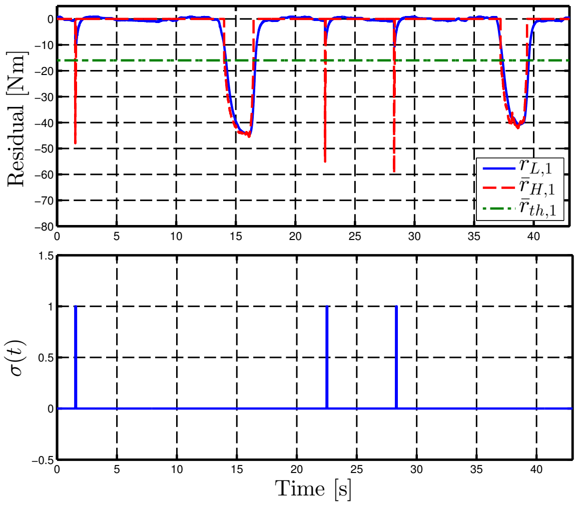


Figure 12: Experiment for distinguishing the nature of contacts. The components of $\boldsymbol{r}_L$ and $\bar{\boldsymbol{r}}_H$ associated to the first robot joint and their threshold $r_{th,1} = 16$ Nm are shown at the top, while signal $\sigma(t)$ evaluated from eq. (11) is shown at the bottom. Undesired collisions are correctly detected at $t = 1.6$ s, $t = 22.5$ s and $t = 28.3$ s.

This method has been tested on the KUKA LWR (type 4+) and the best performance was obtained using the following parameters: $k_L = 5$, $k_H = 60$, $\sigma_{th,j} = 1.8$ and $r_{th,j} = 16$ (for all j). The robot is performing an hexagonal trajectory with its end-effector when a human enters accidentally in the workspace, without any intention to collaborate (see the situation depicted at the bottom-right of Fig. 10). Figure 12 shows the components of residual vectors $\boldsymbol{r}_L$ and $\bar{\boldsymbol{r}}_H$ associated to the first joint and the signal $\sigma(t)$. When the robot hits the human, an undesired collision is detected (at the three instants $t = 1.6$ s, $t = 22.5$ s and $t = 28.3$ s) and the *reflex* reaction strategy is used to drive away the robot from the human and to then to stop it. After a few seconds in which the robot remains at rest, the task is resumed. Note that, when the human wants to collaborate with the robot and touches it softly (as in the situation depicted at the top-right of Fig. 10) $\sigma(t)$ remains at zero as it should be in case for *soft* contacts. This happens for the contact events starting around $t = 13.6$ s and $t = 37.1$ s in Fig. 12.

4 Collision Reaction

Once an undesired physical collision has been detected the robot switches as fast as possible from the control law associated to normal task execution to a reaction control law. A series of alternative reaction strategies have been considered and implemented, see Fig. 13, mostly based on the value of the residual $\boldsymbol{r}$ and on the use of different thresholds.

Beside stopping the robot, the basic reflex reaction once the collision has been detected is to apply a reaction torque, including gravity compensation, of the form

$$\boldsymbol{\tau}_R = \boldsymbol{K}_R\,\boldsymbol{r} + \boldsymbol{g}(\boldsymbol{q}), \qquad \boldsymbol{K}_R > 0 \text{ (typically, diagonal)} \tag{12}$$

for the rigid robot case. For the KUKA LWR, because of its joint elasticity, the reaction command is modified as

$$\boldsymbol{\tau}_R = \boldsymbol{K}_R\,\boldsymbol{r} + \bar{\boldsymbol{g}}(\boldsymbol{\theta}), \tag{13}$$

where $\bar{\boldsymbol{g}}$ is a quasi-static approximation of the required gravity compensation which uses only motor measurements $\boldsymbol{\theta}$ and guarantees passivity (see [2]). This torque is then fed via the FRI as a user commanded torque (in impedance mode), that will be realized by the low-level KUKA controller using joint torque sensing. The result is that the robot flies away from the collision area (disregarding the original task).

Not all of the reaction strategies abandon completely the original task execution. For instance, a time scaling of the original joint or Cartesian trajectory can be realized [9] for mild collisions, so that the robot stops or moves back along the previous path and recovers forward motion as soon as the collision is over. So, the trajectory is scaled in time but the originally planned path is never left by the robot, unless a very critical collision is detected. In [3], the redundancy of the robot with respect to the original task (e.g., the end-effector tracking a desired trajectory) is exploited by accommodating the robot reaction command in a suitable dynamic null space. As a result, the Cartesian task execution is preserved in full or partially, provided that the residual signal r does not reach a higher safety threshold. Otherwise, the persistent contact may lead to a potential human injury, and then the task is eventually abandoned. More details can be found in [1, 3, 4, 9].

Finally, we remark that a variety of reaction strategies is being developed within SAPHARI in response to the human intention to collaborate with the robot, as recognized through intentional physical contact, haptic touch, voice, or gestures. These are not included in the present document.

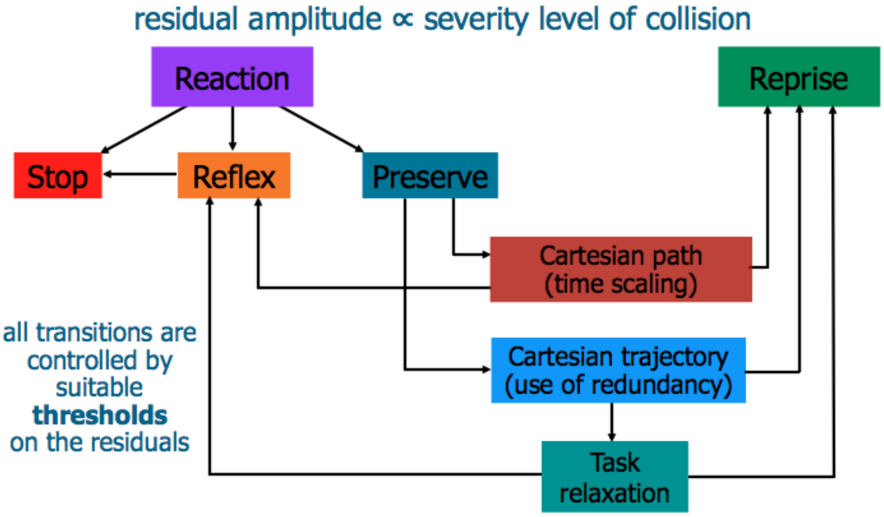

residual amplitude ∝ severity level of collision
Reaction
Reprise
Stop
Reflex
Preserve
Cartesian path (time scaling)
Cartesian trajectory (use of redundancy)
Task relaxation
all transitions are controlled by suitable **thresholds** on the residuals

Figure 13: Portfolio of robot reaction strategies (excluding those intended for collaboration)

References

[1] 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*, pages 1623–1630, 2006.

[2] A. De Luca and W. Book. Robots with flexible elements. In B. Siciliano and O. Khatib, editors, *Springer Handbook of Robotics*, pages 287–319. Springer, 2008.

[3] A. De Luca and L. Ferrajoli. Exploiting robot redundancy in collision detection and reaction. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3299–3305, 2008.

[4] A. De Luca and F. Flacco. Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration. In *Proc. IEEE Int. Conf. on Biomedical Robotics and Biomechatronics*, pages 288–295, 2012.

[5] A. De Luca and R. Mattone. Sensorless robot collision detection and hybrid force/motion control. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 999–1004, 2005.

[6] F. Flacco, T. Kroger, A. De Luca, and O. Khatib. A depth space approach to human-robot collision avoidance. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 338–345, May 2012.

[7] F. Flacco, T. Kroger, A. De Luca, and O. Khatib. A depth space approach for evaluating distance to objects. *(submitted to) Journal of Intelligent and Robotic Systems*, 2014.

[8] M. Geravand, F. Flacco, and A. De Luca. Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 3985–3992, 2013.

[9] S. Haddadin, A. Albu-Schäffer, A. De Luca, and G. Hirzinger. Collision detection and reaction: A contribution to safe physical human-robot interaction. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3356–3363, 2008.

[10] S. Haddadin, S. Belder, and A. Albu-Schaeffer. Dynamic motion planning for robots in partially unknown environments. In *IFAC World Congress (IFAC2011)*, Milan, Italy, September 2011.

[11] S. M. Khansari-Zadeh and Aude Billard. A dynamical system approach to realtime obstacle avoidance. *Autonomous Robots*, 32(4):433–454, 2012.

[12] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. of Robotics Research*, 5(1):90–98, 1986.

[13] E. Magrini, F. Flacco, and A. De Luca. Estimation of contact forces using a virtual force sensor. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, IL, USA, September 2014.

[14] E. Magrini, F. Flacco, and A. De Luca. Human-robot collaboration using estimated contact forces. In *ICRA 2014 Works. on Experiences of Research on Robot Assistance for Industrial Environments*, Hong Kong, PRC, June 2014.

[15] SAPHARI project. MIlestone MS10: Collision prevention by visual feedback. Technical report, UNINA, November 2013.

[16] M. Saveriano and D. Lee. Distance based dynamical system modulation for reactive collision avoidance. In *DGR-Tage*, 2013.

[17] M. Saveriano and D. Lee. Point cloud based dynamical system modulation for reactive avoidance of convex and concave obstacles. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5380–5387, 2013.

[18] M. Saveriano and D. Lee. Distance based dynamical system modulation for reactive avoidance of moving obstacles. In *Proc. of the IEEE International Conference on Robotics and Automation*, 2014.

[19] M. Saveriano and D. Lee. Safe motion generation and online reshaping using dynamical systems. In *IEEE International Conference on Ubiquitous Robots and Ambient Intelligence*, 2014 (submitted).