A unified multimodal control framework for human–robot interaction

Andrea Cherubini, Robin Passama, Philippe Fraisse, André Crosnier

A unified multimodal control framework for human-robot interaction

Andrea Cherubini, Robin Passama, Philippe Fraisse, and André Crosnier

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier LIRMM
Université Montpellier - CNRS, 161 Rue Ada, 34392 Montpellier, France
firstname.lastname@lirmm.fr

Abstract

In human-robot interaction, the robot controller must reactively adapt to sudden changes in the environment (due to unpredictable human behaviour). This often requires operating different modes, and managing sudden signal changes from heterogeneous sensor data. In this paper, we present a multimodal sensor-based controller, enabling a robot to adapt to changes in the sensor signals (here, changes in the human collaborator behaviour). Our controller is based on a unified task formalism, and in contrast with classical hybrid vision-force-position control, it enables smooth transitions and weighted combinations of the sensor tasks. The approach is validated in a mock-up industrial scenario, where pose, vision (from both traditional camera and Kinect), and force tasks must be realized either exclusively or simultaneously, for human-robot collaboration.

Keywords: Reactive and Sensor-based Control, Human-Robot Interaction, Visual Servoing.

1. Introduction

Recently, the attention of robotics researchers worldwide has turned towards the field of human-robot interaction (HRI [1, 2, 3, 4, 5]), to enable close collaboration between human and robot [6, 7]. In this context, the robot must infer the user intention, to interact more naturally, from the human perspective [8, 9, 10]. To this end, both visual (e.g., based on Microsoft Kinect^{TM} [11]) and force feedback, have been used [12, 13, 14, 15, 16]. Generally, we believe that direct sensor-based methods, such as visual servoing [17], provide better solutions, for intuitive HRI, than planning techniques, requiring a priori models of the environment and agents [18]. Moreover, force and vision should be used concurrently, since the information they provide is complementary. One pioneer work in this sense is [19], where force and visual control are used to avoid collisions, while tracking human motion during interaction.

However, the authors do not provide a unified solution for integrating the two sensing modalities. Instead, since the vision and force sensors often measure

different physical phenomena, it is preferable to directly combine their data at the control level, rather than to apply multi-sensory fusion, or to design complex state machines. This idea has been initially proposed in [20, 21], by adapting the *hybrid position-force control* paradigm [22]: force constrains some motion directions, while vision drives the remaining degrees of freedom. Later, the authors of [23] have presented a list of hybrid control configurations, and divided the degrees of freedom to be controlled by vision and force. An alternative is *impedance/admittance control* [24], which has been integrated with visual [25] and even tactile [26] control, to account for external forces. Although many techniques for merging vision, force and position control have been designed, the presence of the human in the robot control loop is rarely accounted for.

In our previous work [27], we have started the design of a multimodal framework for human-robot cooperation. The approach is marker-less, and has been validated in a mock-up industrial scenario. However, the following contributions are brought here, with regards to that work:

- a unified formalism, inspired by inverse kinematics [28, 29] guarantees the controller stability, independently from the sensor modality;
- the use of smooth transitions (homotopies) between the sensor-based tasks, and of self-adapting gains, limits the robot accelerations, thus guaranteeing safer operation;
- in contrast with hybrid vision-force control, it is possible to control a same task direction using weighted combinations of different sensors, and sensor-based tasks can be expressed in different reference frames;

Other, minor improvements, with regards to [27], include the introduction of force-based control, guaranteeing safety of HRI, and better accuracy, velocity, and control smoothness. Moreover, our task-oriented approach, **in contrast with** similar ones, such as the stack-of-tasks [29] and constraint-based programming [30, 31], is directly usable in real HRI scenarios (to our knowledge, the method presented in [31] has been used, for now, only for human collision avoidance).

The article is organized as follows. In Section 2, we present our general framework for multimodal control for HRI. In Section 3, relevant variables and sensor-based tasks are defined. Based on these preliminaries, Section 4 shows how the general framework can be instantiated for an industrial case study. Experimental results are reported in Section 5, and summarized in the Conclusion.

2. Control framework

To safely interact with the human, the designed controller must rely on the various sensing modalities present on the robot. These may include cameras (for vision), force/torque sensors, skin (for tact), or proprioception (e.g., for positioning).

A commonly used approach to merge the various sensing modalities directly at the control level is *hybrid sensors control*, e.g. hybrid force/position [22] or hybrid force/vision [20] control. This approach was recently extended in a framework integrating vision, force and tact to realize physical interaction tasks [32]. We hereby recall the formulation of that approach, and propose a more generic one, based on classic *inverse kinematics control* [28].

Let k be the dimension of the operational space associated with the end effector (e.g., $k = 3$ in the case of a planar manipulator). Consider n senses and, for each sense, the task vector $\mathbf{s}_m \in \mathbb{R}^k$, with $m = 1, \ldots, n$. For example, a task associated with the sense of vision could consist in controlling an on-board camera to make it look at a target point, and a task associated with the sense of force could consist in applying a desired wrench with the end effector (e.g., $\mathbf{s}_f = [f_x, f_y, m_z]$ for a planar manipulator). In this work, since position, vision and force are used, $n = 3$. All n tasks have the same size k, and, if the sensor provides less than k measures, it will be sufficient to select the task components corresponding to the actual measures, as will be explained later.

Each task is related to the Cartesian velocity of the end effector, $\mathbf{v} \in \mathbb{R}^k$ by the $k \times k$ matrix $\mathbf{L}_m$ (called *interaction matrix* in the case of visual servoing):

$$\dot{\mathbf{s}}_m = \mathbf{L}_m \mathbf{v}. \tag{1}$$

Stacking the n tasks yields:

$$\dot{\bar{\mathbf{s}}} = \mathbf{L}\mathbf{v}, \quad \text{with } \bar{\mathbf{s}} = \begin{bmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_n \end{bmatrix} \in \mathbb{R}^{kn} \text{ and } \mathbf{L} = \begin{bmatrix} \mathbf{L}_1 \\ \vdots \\ \mathbf{L}_n \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^k. \tag{2}$$

As aforementioned, a combination of tasks defined by different senses (i.e., by components of the different $\mathbf{s}_m$) is realizable, as long as its size is also k. The tasks are selected thanks to n positive definite square diagonal *selection matrices* of size k, denoted $\mathbf{S}_m$, that activate or deactivate a given task component. Then, the k-dimensional hybrid task *to be realized*, is a linear mapping of the complete $\bar{\mathbf{s}}$:

$$\dot{\mathbf{s}} = \mathbf{S}\dot{\bar{\mathbf{s}}}, \qquad \text{with} \qquad \mathbf{S} = [\mathbf{S}_1 \ldots \mathbf{S}_n] \in \mathbb{R}^k \times \mathbb{R}^{kn}. \tag{3}$$

Note that, as outlined above, if the m-th sensor provides less than k measures, the missing components can be deselected by simply setting to zero the corresponding row in $\mathbf{S}_m$. The selection matrices can also be used, as will be shown later, to weigh outputs from different sensors and combine them into a single task.

Merging (3) and (2) gives the open-loop behaviour of the task in function of the end effector velocity:

$$\dot{\mathbf{s}} = \mathbf{S}\mathbf{L}\mathbf{v}. \tag{4}$$

Inverse kinematics control relies on the assumption that matrix $\mathbf{SL}$ is in-

vertible[1]. Then, the optimal[2] solution of (4) ensuring exponential convergence of $\mathbf{s}$ to the desired constant task $\mathbf{s}^*$ is:

$$\mathbf{v} = (\mathbf{SL})^{-1} (\mathbf{s}^* - \mathbf{s}) . \tag{5}$$

Indeed, replacing this into (4) yields:

$$\dot{\mathbf{s}} = \mathbf{s}^* - \mathbf{s}, \tag{6}$$

guaranteeing that $\mathbf{s} = \mathbf{s}^*$ is a stable equilibrium for the closed-loop system.

Let us now compare (5) with the *hybrid sensors control* used in numerous works [20, 22, 23, 32]. This approach consists in assigning each sensing modality to a Cartesian direction in the operational space, and then summing the velocities associated with the selected sensors:

$$\mathbf{v} = \sum_m \mathbf{S}_m \mathbf{v}_m, \tag{7}$$

with some assumption on the selection matrices, e.g., that they are orthogonal, as in [32].

Assuming each $\mathbf{L}_m$ is invertible, exponential convergence of $\mathbf{s}_m$ to $\mathbf{s}_m^*$, according to (1), is guaranteed by applying:

$$\mathbf{v}_m = \mathbf{L}_m^{-1} (\mathbf{s}_m^* - \mathbf{s}_m) . \tag{8}$$

Plugging (8) into (7), we obtain the *hybrid sensors control* expression:

$$\mathbf{v} = \mathbf{S}\tilde{\mathbf{L}} (\bar{\mathbf{s}}^* - \bar{\mathbf{s}}) \quad \text{with} \quad \tilde{\mathbf{L}} = \begin{bmatrix} \mathbf{L}_1^{-1} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \mathbf{L}_n^{-1} \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^{kn}. \tag{9}$$

This controller is optimal for (4), if and only if (9) coincides with (5):

$$\mathbf{S}\tilde{\mathbf{L}} (\bar{\mathbf{s}}^* - \bar{\mathbf{s}}) = (\mathbf{SL})^{-1} (\mathbf{s}^* - \mathbf{s}) \quad \forall \left(\bar{\mathbf{s}}^*, \dot{\bar{\mathbf{s}}}^*\right) \in \mathbb{R}^2. \tag{10}$$

This is equivalent, considering (3), to:

$$\mathbf{S}\tilde{\mathbf{L}} = (\mathbf{SL})^{-1} \mathbf{S}. \tag{11}$$

In general, this is not the case, but we hereby provide two necessary conditions for it to be true.

Property. *Hybrid sensors control (9) is optimal if the diagonal selection matrices $\mathbf{S}_m$ are all binary and orthogonal, and if the sensor matrices $\mathbf{L}_m$ are all diagonal.*

[1] Otherwise, specific strategies for avoiding singularities, which are out of the scope of this paper, are to be devised.

[2] Throughout the paper, we refer to controllers as *optimal* when they provide the least squares solution to the task, i.e., they minimize the control effort.

Proof. Since all $\mathbf{S}_m$ are binary (hence, idempotent) and orthogonal:

$$\sum_{m=1}^{n} \mathbf{S}_m = \mathbf{I}. \tag{12}$$

Moreover, binary $\mathbf{S}_m$ imply that $\mathbf{S}$ has full rank, so its right pseudoinverse can be derived to show, using (12), that it coincides with its transpose:

$$\mathbf{S}^\dagger = \mathbf{S}^\top \left(\mathbf{S}\mathbf{S}^\top\right)^{-1} = \mathbf{S}^\top \left(\sum_m \mathbf{S}_m^2\right)^{-1} = \mathbf{S}^\top \sum_m \mathbf{S}_m = \mathbf{S}^\top. \tag{13}$$

Then, post-multiplying condition (11) by $\mathbf{S}^\dagger = \mathbf{S}^\top$, we obtain:

$$\mathbf{S}\tilde{\mathbf{L}}\mathbf{S}^\dagger = (\mathbf{S}\mathbf{L})^{-1}\,\mathbf{S}\mathbf{S}^\dagger, \tag{14}$$

which leads to:

$$\mathbf{S}\tilde{\mathbf{L}}\mathbf{S}^\top = (\mathbf{S}\mathbf{L})^{-1}. \tag{15}$$

- The first member of (15) becomes:

$$\mathbf{S}\tilde{\mathbf{L}}\mathbf{S}^\top = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1} \mathbf{S}_m. \tag{16}$$

  By commuting the matrix product (since all $\mathbf{L}_m$ and $\mathbf{S}_m$ are diagonal, and have the same size), and taking advantage of the idempotency of the $\mathbf{S}_m$, we obtain:

$$\mathbf{S}\tilde{\mathbf{L}}\mathbf{S}^\top = \sum_m \mathbf{S}_m^2 \mathbf{L}_m^{-1} = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1}. \tag{17}$$

- The second member of (15) becomes:

$$(\mathbf{S}\mathbf{L})^{-1} = \left(\sum_m \mathbf{S}_m \mathbf{L}_m\right)^{-1}. \tag{18}$$

  Noting s_{im} and l_{im} the i-th elements of $\mathbf{S}_m$ and $\mathbf{L}_m$, respectively:

$$\begin{aligned} (\mathbf{S}\mathbf{L})^{-1} &= \left[\operatorname{diag}\left(\sum_m s_{1m} l_{1m}, \ldots, \sum_m s_{km} l_{km}\right)\right]^{-1} = \\ &= \operatorname{diag}\left(\frac{1}{\sum_m s_{1m} l_{1m}}, \ldots, \frac{1}{\sum_m s_{km} l_{km}}\right). \end{aligned} \tag{19}$$

  Since for each i, exactly one s_i is non-null and equal to 1, this equation can be rewritten:

$$(\mathbf{S}\mathbf{L})^{-1} = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1}. \tag{20}$$

Equations (17) and (20) demonstrate that the first and second members of (15) coincide, and that the property is therefore valid. □

To summarize, hybrid sensors control provides an optimal solution for (4) under two strong assumptions.

1. All the sensor tasks $\mathbf{s}_m$ must be expressed in the same reference frame. This can be stated from in (1), subject to the condition that the $\mathbf{L}_m$ matrices are diagonal.
2. Only one sensor can be used to control each end effector direction. This can be stated from (7), subject to the condition that the $\mathbf{S}_m$ are binary and orthogonal.

These assumptions are mentioned in all works that apply hybrid sensors control. However, they limit its use in practical applications. For instance, merging image-based visual servoing [17], which defines the visual task in the image frame, with force control, usually implemented in the force sensor frame, would infringe the first assumption.

On the other hand, to guarantee stability of the closed-loop system, the classical inverse control scheme (5) only requires that $\mathbf{SL}$ is invertible (a weaker assumption, that is always true if the $\mathbf{S}_m$ are binary, and the $\mathbf{L}_m$ diagonal). Controller (5) can be applied even if the task frames associated with each sensor are different, and even if a task is defined for multiple robots [33], or as a combination of heterogeneous sensor data (as shown in many recent works by Mansard et al. [29], [34], [35]).

Let us now apply the previous result, by expressing the problem in the joint space, rather than in the operational space. The robot joint velocity is denoted $\dot{\mathbf{q}} \in \mathbb{R}^j$, with j the number of degrees of freedom. We assume that $j \geq k$, so that $\mathbf{s}$ can be realized. If $j > k$, redundancy exists, and one can also minimize a scalar cost function $h(\mathbf{q}) \in \mathbb{R}$, while realizing the task $\mathbf{s}$.

Each task is related to the joint velocity by:

$$\dot{\mathbf{s}}_m = \mathbf{J}_m(\mathbf{q}, \mathbf{s}_m)\,\dot{\mathbf{q}}, \tag{21}$$

where

$$\mathbf{J}_m(\mathbf{q}, \mathbf{s}_m) = \frac{\partial \mathbf{s}_m}{\partial \mathbf{q}} \tag{22}$$

is the corresponding *task Jacobian*, of dimension $k \times j$, that depends on both the robot configuration and on the task. By stacking the n tasks, and using (3), we obtain:

$$\dot{\mathbf{s}} = \mathbf{S}\dot{\bar{\mathbf{s}}} = \mathbf{SJ}(\mathbf{q}, \bar{\mathbf{s}})\,\dot{\mathbf{q}}, \tag{23}$$

where:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_n \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^j. \tag{24}$$

The multimodal controller that we propose, for driving $\mathbf{s}$ to $\mathbf{s}^*$ is given by:

$$\dot{\mathbf{q}} = (\mathbf{SJ})^{\dagger}\,\mathbf{\Lambda}\,(\mathbf{s}^* - \mathbf{s}) + \left[\mathbf{I} - (\mathbf{SJ})^{\dagger}(\mathbf{SJ})\right] \bigtriangledown \mathbf{h} \tag{25}$$

In the above equation:

- $(\mathbf{SJ})^{\dagger}$ is the $j \times k$ right pseudoinverse of $\mathbf{SJ}$. We assume that $\mathbf{SJ}$ is full rank during operation, so that the pseudoinverse can be calculated. This was the case throughout the experiments and is a common assumption in inverse kinematics control [34].
- $\mathbf{\Lambda}$ is a positive definite square diagonal matrix of dimension k that determines the convergence rate of $\mathbf{s}$ to $\mathbf{s}^*$;
- the term $\bigtriangledown\mathbf{h} = \frac{\partial h}{\partial \mathbf{q}}$ (i.e., $\bigtriangledown\mathbf{h} = 0$ when $j = k$) is introduced in order to minimize cost function h in case of redundancy.

System (23), controlled by (25), is globally asymptotically stable with respect to the k selected tasks. Indeed, plugging (25) into (23) yields:

$$\dot{\mathbf{s}} = \mathbf{\Lambda}\left(\mathbf{s}^* - \mathbf{s}\right). \tag{26}$$

Thus, since $\mathbf{\Lambda}$ is a positive definite diagonal matrix, $\mathbf{s} = \mathbf{s}^*$ is a stable equilibrium for the closed-loop system. Also, note that minimization of h has no effect on the convergence rate of the task.

For constant gain matrix $\mathbf{\Lambda}$, convergence of the task will be exponential according to (26). Thus, since (25) is a proportional feedback controller, the joint velocities will also follow an exponential trend, an unwanted behaviour which may lead to abrupt velocity changes at task transitions (i.e., when the error suddenly increases). A simple solution to this, is the use, for each task, of an adaptive gain matrix, function of the task error $\mathbf{s}^* - \mathbf{s}$, inspired by [36]:

$$\mathbf{\Lambda}\left(\mathbf{s}\right) = \mathbf{\Lambda}^* \left[\mathrm{e}^{-\alpha||\mathbf{s}^*-\mathbf{s}||} + \beta\left(1 - \mathrm{e}^{-\alpha||\mathbf{s}^*-\mathbf{s}||}\right)\right]. \tag{27}$$

In (27), $\mathbf{\Lambda}^*$ is the diagonal gain matrix applied when $\mathbf{s}$ **is close to** $\mathbf{s}^*$, and $\alpha \geq 0$ and $\beta \in]0, 1]$ are two scalar parameters such that, as the task error norm $||\mathbf{s}^* - \mathbf{s}||$ increases, $\mathbf{\Lambda}$ exponentially decreases (with slope dependent on α) to $\beta\mathbf{\Lambda}^*$, for very large task error. This exponential trend compensates that of the error signal, thus generating a less variable control input $\dot{\mathbf{q}}$, as will be shown by the experiments. The values of α, β, and $\mathbf{\Lambda}^*$ are tuned empirically, so that the robot joint velocities stay roughly constant during operation.

In the next Section, we we first define the reference frames and the main variables of the framework and then, for each of three sensor-based tasks (position, vision and force), we give the expression of $\mathbf{s}$ and that of the corresponding Jacobian $\mathbf{J}$.

3. Sensor-based tasks

3.1. Definitions

The reference frames used in our work are (see Fig. 1): the robot base (B), camera (C), end effector (E), and image (I) frames. Reference frame B is fixed in the world, whereas C, E and I move with the robot. The pose of A in frame

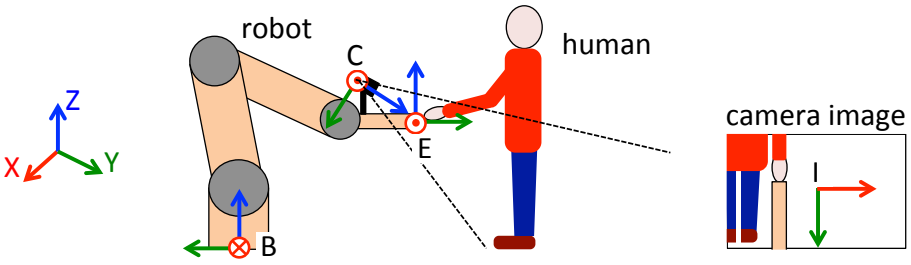

Figure 1: Reference frames used in our multimodal framework for human-robot interaction.

B is defined as: ${}^{B}\mathbf{p}_{A} = \left[{}^{B}\mathbf{t}_{A}, {}^{B}\theta\mathbf{u}_{A}\right]^{\top} \in \mathbb{SE}(3)$, with ${}^{B}\theta\mathbf{u}_{A}$ the angle/axis vector [37].

For the camera, we use the normalized perspective model. A 3D point with coordinates $\left({}^{C}X, {}^{C}Y, {}^{C}Z\right)$ in the camera frame, projects in the image as a 2D point with coordinates:

$$x = \frac{{}^{C}X}{{}^{C}Z}, \qquad y = \frac{{}^{C}Y}{{}^{C}Z}. \tag{28}$$

We assume that the pose of the camera in the end effector, ${}^{E}\mathbf{p}_{C}$, is constant and known through a preliminary calibration step.

For human-robot collaboration, we use $n = 3$ tasks: positioning, visual, and force task. Each one has dimension $k = 6$, in order to control all 6 degrees of freedom of the end effector. We will hereby detail each task.

3.2. Positioning task

The objective of positioning is to control the end effector pose in the base frame. Hence, the positioning task is:

$$\mathbf{s}_{p} = {}^{B}\mathbf{p}_{E}. \tag{29}$$

This can be estimated at each iteration, by applying the robot forward kinematics to the measured articular variables, $\mathbf{q}$.

For this task, the Jacobian in (21) is simply:

$$\mathbf{J}_{p} = \frac{\partial {}^{B}\mathbf{p}_{E}}{\partial \mathbf{q}}. \tag{30}$$

This Jacobian can be computed, at run time, by applying the technique presented in [28].

3.3. Visual task

The objective of the visual task, is to drive the end effector to a desired pose with respect to a visible target. To this end, we apply the two and one-half-dimensional (2 1/2 D) visual servo paradigm originally introduced in [38].

This method combines the advantages of image-based and position-based visual servoing schemes, while trying to avoid their shortcomings [17]. In fact, the task is defined by a combination of image features and 3D characteristics:

$$\mathbf{s}_v = \begin{bmatrix} x & y & \log {}^{C}Z & {}^{C^*}\theta\mathbf{u}_C \end{bmatrix}^\top. \tag{31}$$

In this equation, x and y are the image coordinates of the target characterized by (28), ${}^{C}Z$ is the target depth in the camera frame, and ${}^{C^*}\theta\mathbf{u}_C$ gives the relative rotation between the current and desired poses of the camera.

The Jacobian corresponding to the 2 1/2 D task is [38]:

$$\mathbf{J}_v = \mathbf{L}_s \, {}^{C}\mathbf{V}_B \frac{\partial^B \mathbf{p}_C}{\partial \mathbf{q}}. \tag{32}$$

In this expression, $\mathbf{L}_s$ is the interaction matrix relating the task evolution to the camera velocity in frame C:

$$\mathbf{L}_s = \begin{bmatrix} \mathbf{L}_{11}\left(x, y, {}^{C}Z\right) & \mathbf{L}_{12}\left(x, y\right) \\ 0 & \mathbf{L}_{22}\left({}^{C^*}\theta\mathbf{u}_C\right) \end{bmatrix}, \tag{33}$$

while ${}^{C}\mathbf{V}_B$ is the spatial motion transform matrix from frame B to frame C:

$${}^{C}\mathbf{V}_B = \begin{bmatrix} {}^{C}\mathbf{R}_B & \left[{}^{C}\mathbf{t}_B\right]_\times {}^{C}\mathbf{R}_B \\ \mathbf{0} & {}^{C}\mathbf{R}_B \end{bmatrix}. \tag{34}$$

The complete expressions of $\mathbf{L}_{11}$, $\mathbf{L}_{12}$, and $\mathbf{L}_{22}$ are given in [17], and $[\mathbf{t}]_\times$ is the skew-symmetric matrix associated with vector $\mathbf{t}$. Jacobian $\mathbf{J}_v$ can be calculated at each iteration, since $\mathbf{L}_s$ depends on $\mathbf{s}$, ${}^{C}\mathbf{V}_B$ on the pose of B in C (determined via forward kinematics ${}^{B}\mathbf{p}_E$ plus constant known ${}^{E}\mathbf{T}_C$), and $\partial^B \mathbf{p}_C / \partial \mathbf{q}$ can be calculated again using the technique presented in [28].

3.4. Force task

The objective of force control is to regulate the external wrench $\mathbf{h}$ (force and torque vectors $\mathbf{f}$ and $\mathbf{m}$), at the contact point between robot and human, to a desired value. This is essential to guarantee safe interaction with the environment and with the human operator. Without loss of generality, in this work, such external wrench is expressed in the end effector frame E.

To realize the force task, we apply an admittance controller [24], where the deviation of the end effector motion due to the interaction with the environment is related to the contact wrench, through an equivalent mass-spring-damper system with adjustable parameters.

Here, we consider a simple spring system, with null mass and damping, and positive definite diagonal square stiffness matrix $\mathbf{K}$, such that:

$${}^{E}\mathbf{h}_E - {}^{E}\mathbf{h}^*_E = -\mathbf{K}\left({}^{E}\mathbf{p}_E - {}^{E}\mathbf{p}^*_E\right) = \mathbf{K}\, {}^{E}\mathbf{p}^*_E. \tag{35}$$

Then, the force task is defined as: $\mathbf{s}_f = {}^{E}\mathbf{h}_E$. Deriving the above equation yields the Jacobian corresponding to this task:

$$\mathbf{J}_f = -\mathbf{K}\, \frac{\partial^E \mathbf{p}_E}{\partial \mathbf{q}}. \tag{36}$$

Having defined $\mathbf{K}$, $\mathbf{J}_f$ can again be calculated with the technique from [28].

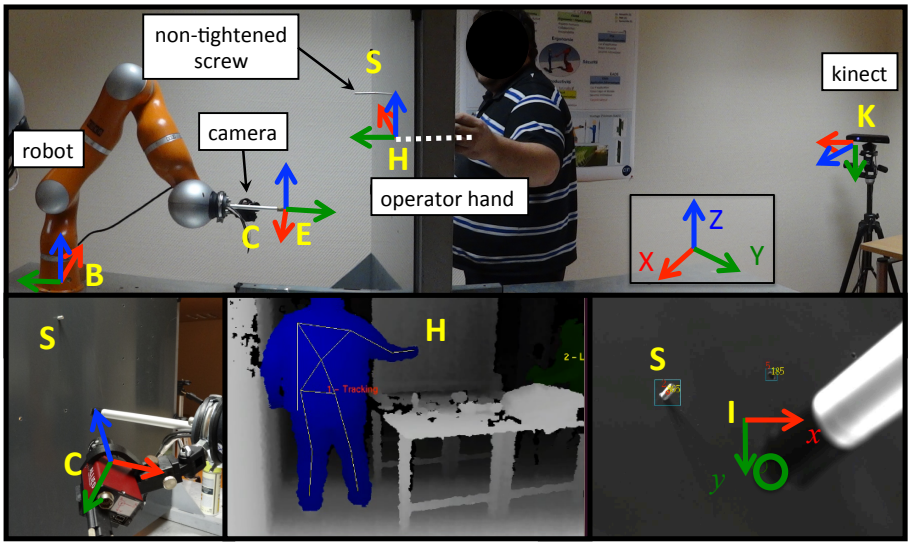

Figure 2: Collaborative screwing case study. Top: experimental setup. Bottom left: view of the camera and end effector. Bottom center: Kinect image. Bottom right: camera image.

4. A case study: collaborative screwing

4.1. Experimental setup and assumptions

To validate our controller, we focus on a case study, where a robot aids a human operator in a screwing operation. In Fig. 2, we show the setup, along with the frames defined in Sect. 3.1; the screw is denoted S.

Human and robot operate on the opposite sides of a flank, where a series of screws must be inserted. The required operations are respectively:

- for the human: to insert the screws in the holes,
- for the robot: to tighten a bolt on each of the inserted screws, while the human maintains it on the flank.

Since the focus here is mainly on our **multimodal control framework**, we do not implement the physical screwing action; instead, we consider a screw to be tightened, when the end effector touches it with proper alignment. To realize the collaborative screwing operation, we utilize a Kinect, that outputs an RGB-D image of the work scene from a fixed pose, and a black and white camera mounted on the robot. These sensors are respectively dedicated to detecting and tracking the human hand motion, and to tracking newly inserted screws on the flank. Finally, to properly align end effector and screw, an estimation of the external forces is necessary. This force estimation will be explained in Sect. 5.

Our work assumptions are that the flank is perpendicular to the Y axis of the base frame, at known distance from B, and that the Kinect pose in the base frame has been coarsely calibrated.

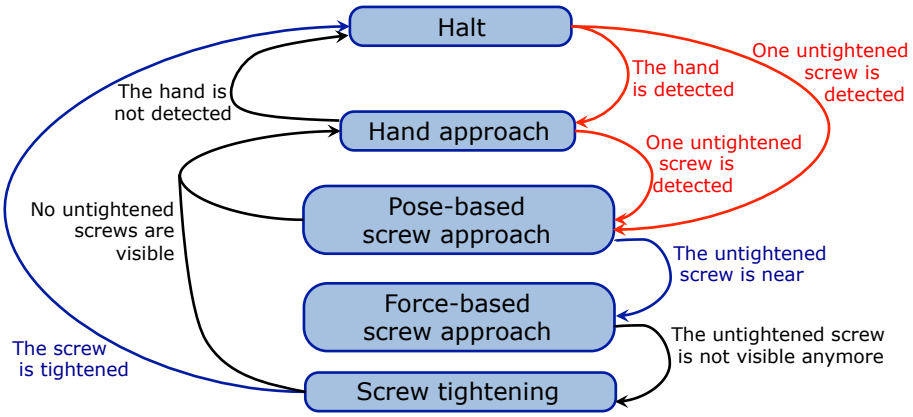

Figure 3: Collaborative screwing state machine, selecting the appropriate control mode, according to the sensed data.

To avoid luminosity variations in the image, we maintain the camera orientation with respect to the flank constant throughout operation. Since ${}^{E}\mathbf{p}_{C}$ is constant, we have decided to do this by keeping the end effector perpendicular to the flank, with the axes of frame E placed as in Fig. 2. Hence, we impose the desired rotation matrix from end effector to base to be:

$$ {}^{B}\mathbf{R}_{E}^{*} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \tag{37} $$

In the rest of this Section, we will detail the strategy that has been used to realize collaborative screwing, with controller (25).

4.2. Multimodal control strategy

To realize the collaborative screwing task, we utilize four modes, and halting, which simply consists in setting $\dot{\mathbf{q}} = \mathbf{0}$. The Jacobian used in (25) is:

$$ \mathbf{J} = \begin{bmatrix} \mathbf{J}_p \\ \mathbf{J}_v \\ \mathbf{J}_f \end{bmatrix}, \tag{38} $$

with $\mathbf{J}_p$, $\mathbf{J}_v$, and $\mathbf{J}_f$ defined respectively in (30), (32), and (36).

The modes are operated by the state machine in Fig. 3. As the figure shows, the transitions can be activated either by detection (red) or loss (black) of sensed information, or by success of the mode (blue). The detection/loss of information is determined by sensors processing. Instead, a mode is successful when:

$$ \sum_{i=1}^{6} w_i ||s_i^* - s_i|| < \sigma, \tag{39} $$

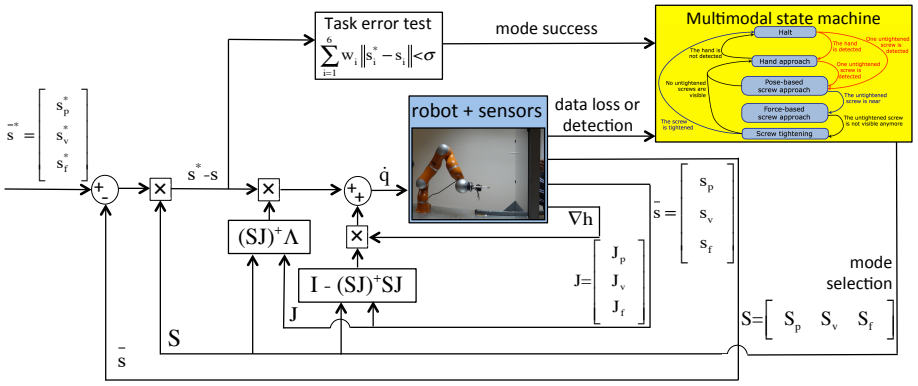

Figure 4: Multimodal framework block diagram.

with $\mathbf{w} = [w_1 \dots w_6] \in \mathbb{R}^6$ a vector of positive weights, and σ a scalar threshold.

Our complete framework is summarized in Fig. 4. In the rest of this section, we will focus on each of the four modes, by specifying the selection matrices $\mathbf{S}_p$, $\mathbf{S}_v$ and $\mathbf{S}_f$, the desired task vector $\mathbf{s}^*$, and the activation condition (39).

4.3. Hand approaching mode

If the human operating hand is detected by the Kinect, its position is fed to a controller that moves the robot so that the camera has a good view of the area where the human is operating. Since only the positioning task (29) is necessary, the selection matrices are:

$$\mathbf{S}_p = \mathbf{I} \qquad \mathbf{S}_v = \mathbf{S}_f = \mathbf{0}. \tag{40}$$

From (3) and (29), we can infer the desired task vector:

$$\mathbf{s}^* = \mathbf{S}_p \mathbf{s}_p^* = {}^B\mathbf{p}_E^* \in \mathbb{SE}(3). \tag{41}$$

To derive $\mathbf{s}_p^*$, we introduce the Kinect (K), and operating hand (H) frames (see Fig. 2). The origin of H is the orthogonal projection of the operator hand on the flank, while its orientation is that of B. The hand position in the kinect frame is estimated using OpenNI[3], and then orthogonally projected on the flank[4] to obtain ${}^K\mathbf{X}_H$, which can then be transformed to ${}^B\mathbf{X}_H$ in the base frame. Then, our aim is to place H at a fixed desired position in the camera frame:

$${}^C\mathbf{X}_H^* = \begin{bmatrix} {}^C X_H^* & {}^C Y_H^* & {}^C Z_H^* & 1 \end{bmatrix}^\top, \tag{42}$$

to increase the chances of visualizing in the image the future inserted screw. Also, since we set ${}^H\mathbf{R}_B = \mathbf{I}$, ${}^B\mathbf{R}_E^*$ according to (37), and since ${}^C\mathbf{R}_E$ is constant

[3] http://www.openni.org

[4] It is trivial to derive the flank plane equation in K from ${}^B\mathbf{T}_K$ and ${}^B Y_H$.

and known, the desired orientation of the camera with respect to the hand, ${}^{H}\mathbf{R}^{*}_{C}$ can be derived. Combining ${}^{C}\mathbf{X}^{*}_{H}$ and ${}^{H}\mathbf{R}^{*}_{C}$, we can obtain the desired camera to hand transformation, ${}^{H}\mathbf{T}^{*}_{C}$. This can now be used to determine ${}^{B}\mathbf{T}^{*}_{E}$, and therefore $\mathbf{s}^{*}_{p}$, to be used in (25), along with:

$$\mathbf{S} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \tag{43}$$

Since the hand approaching mode should be activated and deactivated only by perceived data, the task convergence need not be monitored, and we can set $\sigma = 0$ and $\mathbf{w} \neq \mathbf{0}$, so that (39) is never true in this mode.

4.4. Pose-based screw approaching mode

If a screw is detected in the image (see Fig. 2, bottom right), its position determines s_v according to (31), so that the end effector is driven in front of it. To this end, we exploit the screw position as viewed from the on-board camera (to infer x and y), along with the measures of the robot articular positions for forward kinematics (to infer ${}^{C}Z$ and ${}^{C^*}\theta\mathbf{u}_C$). The details on the image processing algorithms used to detect and track the screws are given in [27].

The shift from hand to screw approaching can lead to abrupt joint accelerations. This problem, which is common when switching between manipulation primitives, has been recently tackled using on-line trajectory generation [39]. Here, we exploit homotopy to better manage the transition. We define $t > 0$ the *screw age* (i.e., the time since it has been detected). The visual task selection matrix is then designed to smoothly vary from $\mathbf{0}$ to $\mathbf{I}$, as t tends to a tuned scalar T:

$$\mathbf{S}_v = \begin{cases} \frac{1-\cos(\pi t/T)}{2}\ \mathbf{I} & \text{if } \ t < T, \\ \mathbf{I} & \text{otherwise.} \end{cases} \tag{44}$$

The other task selection matrices are set to:

$$\mathbf{S}_p = \mathbf{I} - \mathbf{S}_v, \qquad \mathbf{S}_f = \mathbf{0}, \tag{45}$$

so that, in controller (25):

$$\mathbf{S} = \begin{bmatrix} \mathbf{I} - \mathbf{S}_v & \mathbf{S}_v & \mathbf{0} \end{bmatrix}. \tag{46}$$

In practice, the visual task is gradually activated by $\mathbf{S}_v$, while concurrently the hand position task is deactivated by $\mathbf{S}_p$.

As proved in Sect. 2, the advantage of our framework is that such a smooth transition can be easily implemented without compromising the controller stability, since, in contrast with hybrid sensors control, the selection matrices do not have to be binary. This is also a fundamental advantage with respect to the method proposed in [39].

From (3), we can infer the desired task vector:

$$\mathbf{s}^{*} = \begin{cases} (\mathbf{I} - \mathbf{S}_v)\, \mathbf{s}^{*}_{p} + \mathbf{S}_v \mathbf{s}^{*}_{v} & \text{if } \ t < T, \\ \mathbf{s}^{*}_{v} & \text{otherwise.} \end{cases} \tag{47}$$

As mentioned, $\mathbf{s}^*$ varies from a task dependent on both hand and screw, to purely vision-based screw approaching task $\mathbf{s}_v^*$, that we define as:

$$\mathbf{s}_v^* = \begin{bmatrix} x_S^* & y_S^* & \log {}^C Z_S^* & \mathbf{0} \end{bmatrix}^\top . \tag{48}$$

This $\mathbf{s}_v^*$ corresponds to driving the screw to image position (x_S^*, y_S^*) at desired depth ${}^C Z_S^*$, while zeroing the orientation error between C and C^*.

Let us now explain how $\mathbf{s}_v^*$ is derived. The image position (x_S^*, y_S^*) (circle in Fig. 2, bottom right) is set so that end effector and screw are aligned at the end of this mode. We set the end effector Cartesian position to have a desired translation with respect to the screw:

$${}^E\mathbf{t}_S^* = \begin{bmatrix} 0 & 0 & {}^E Z_S^* \end{bmatrix}^\top , \tag{49}$$

so that ${}^E Z_S^* > 0$ is as small as possible, without end effector occlusion. Then, from the known ${}^C\mathbf{T}_E$, and from ${}^E\mathbf{X}_S^*$, we can derive ${}^C\mathbf{X}_S^*$, and, from that: ${}^C Z_S^*$, $x_S^* = {}^C X_S^* / {}^C Z_S^*$, and $y_S^* = {}^C Y_S^* / {}^C Z_S^*$. For rotations, as usual we servo ${}^B\mathbf{R}_E^*$ according to (37). Then, ${}^{C^*}\theta\mathbf{u}_C$ can be calculated from known ${}^C\mathbf{R}_E$ and ${}^E\mathbf{R}_B^*$.

The following mode (force-based screw approaching), is triggered when the visual error with respect to the screw is small enough. Hence, we set:

$$\begin{cases} \mathbf{w} \neq 0 \ \ \forall t > 0, \\ \sigma = 0 \ \text{ if } \ t < T, \\ \sigma > 0 \ \text{ otherwise}, \end{cases} \tag{50}$$

so that (39) is verified only after time T, when the hand task is deactivated.

4.5. Force-based screw approaching mode

Once the screw is near enough, force control is activated, to make the end effector compliant in case of contact, while advancing. We activate this mode just before contact, because, in the absence of external contacts, the force signal to noise ratio can lead to inaccurate positioning.

The desired wrench on the end effector, in the end effector frame, is:

$${}^E\mathbf{h}_E^* = \begin{bmatrix} 0 & 0 & {}^E f_{E,Z}^* & 0 & 0 & 0 \end{bmatrix}^\top . \tag{51}$$

Through force control (35), ${}^E f_{E,Z}^* < 0$ makes the end effector progress forward. All other components are zeroed to make the end effector compliant.

As long as the screw is visible, the end effector can be driven towards it by using visual control. Then, the task selection matrices are:

$$\mathbf{S}_p = \mathbf{0} \qquad \mathbf{S}_v = \begin{bmatrix} \mathbf{I}_2 & \mathbf{0}_{2\times 4} \\ \mathbf{0}_{4\times 2} & \mathbf{0}_4 \end{bmatrix} \qquad \mathbf{S}_f = \begin{bmatrix} \mathbf{0}_2 & \mathbf{0}_{2\times 4} \\ \mathbf{0}_{4\times 2} & \mathbf{I}_4 \end{bmatrix}. \tag{52}$$

Therefore, in controller (25):

$$\mathbf{S} = \begin{bmatrix} \mathbf{0}_{2\times 6} & \mathbf{I}_2 & \mathbf{0}_{2\times 6} & \mathbf{0}_{2\times 4} \\ \mathbf{0}_{4\times 6} & \mathbf{0}_{4\times 2} & \mathbf{0}_{4\times 6} & \mathbf{I}_4 \end{bmatrix}, \tag{53}$$

and the desired task is:

$$\mathbf{s}^* = \mathbf{S}_v \mathbf{s}_v^* + \mathbf{S}_f \mathbf{s}_f^* = \begin{bmatrix} x_s^* & y_s^* & {}^E f_Z^* & 0 & 0 & 0 \end{bmatrix}^\top . \tag{54}$$

The transition from this mode to the following is triggered by the loss of the screw, when it is too near to be visible in the image. Then, we set $\sigma = 0$ and $\mathbf{w} \neq \mathbf{0}$, so that (39) is never true.

4.6. Screw tightening mode

When the screw is so near that it is not visible any more, the last mode is activated. This relies solely on force control:

$$\mathbf{S}_p = \mathbf{0} \qquad \mathbf{S}_v = \mathbf{0} \qquad \mathbf{S}_f = \mathbf{I}. \tag{55}$$

Therefore, in controller (25):

$$\mathbf{S} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \tag{56}$$

and the desired force task is:

$$\mathbf{s}^* = \mathbf{s}_f^* = \begin{bmatrix} 0 & 0 & {}^E f_{E,Z}^* & 0 & 0 & 0 \end{bmatrix}^\top . \tag{57}$$

Clearly, if tightening was also to be realized (although this is not the case here, as mentioned in Sect. 4.1), the desired moment around Z, ${}^E m_{E,Z}^*$ should also be non-null. To verify that the screw is tightened, we check the force error according to (39), with tuned weights $\mathbf{w} \neq 0$ and threshold $\sigma > 0$.

5. Experiments

To validate our framework, we have run a series of experiments with a lightweight KUKA LWR IV robot in the scenario illustrated in Fig. 2. Since a tightening tool is not mounted on the end effector, we have used a cylindrical tool of external diameter 14 mm to verify the precision of our method. The LWR is redundant with respect to the end effector operational space dimension (it has $j = 7$ degrees of freedom, whereas $k = 6$). Thus, we use the extra degree of freedom to guarantee joint limit avoidance. To this end, in (25), we use a scalar, configuration dependent, cost function [40]:

$$h(\mathbf{q}) = \frac{1}{2} \sum_{i=1}^{7} \left(\frac{q_i - q_{i,mid}}{q_{i,M} - q_{i,m}} \right)^2 , \tag{58}$$

with $[q_{i,m}, q_{i,M}]$ the available range for joint i and $q_{i,mid} = (q_{i,M} + q_{i,m})/2$ its midpoint. The values of $\dot{\mathbf{q}}$ computed via (25) are fed to the Reflexxes online trajectory generation library[5] for smoothing. To get the interaction wrench

[5] www.reflexxes.com

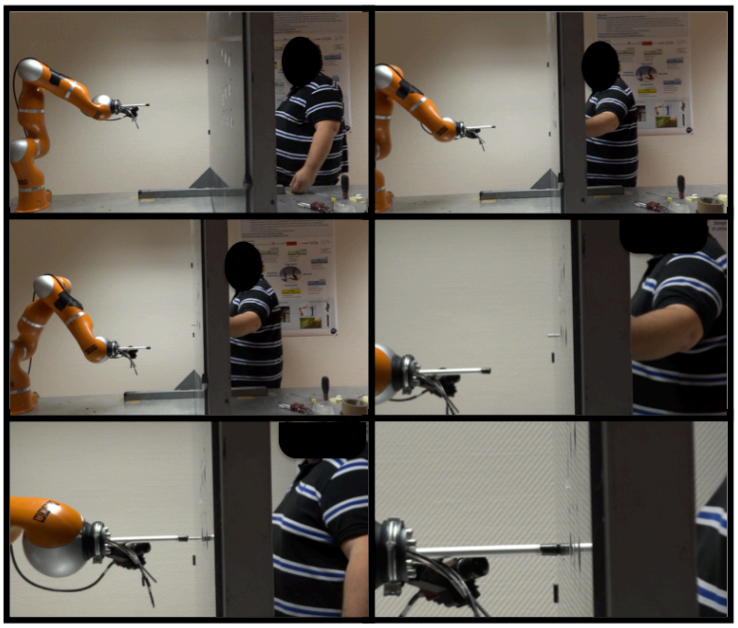

Figure 5: Six consecutive snapshots of the first experiment of collaborative screwing.

${}^{E}\mathbf{h}_{E}$, instead of mounting a force sensor on the end effector, we have decided to use the estimated external wrench signal provided by the robot controller through the FRI Interface[6]. The camera mounted on the end effector is a Stingray F201B from Allied Vision Technologies, with resolution 1024×768 pixels, and we used the ViSP library [41] for visualization purposes. The image processing pipeline takes approximately 60 ms. Thus, although the skeleton processing on the Kinect is slightly faster, we fix the control loop rate at 15 Hz.

To highlight the novel contributions of our framework, i.e., force control, and the use of homotopy and adaptive gains, various experiments were run. These are shown in the video attached to this paper.

In a preliminary experiment, (see Fig. 5), three screws are touched with the tip of the tool, using only the hand approaching (HA) and pose-based screw approaching (SA) modes. In this experiment, the homotopy between these two modes is deactivated, and the gain matrix $\mathbf{\Lambda}$ is independent from $\mathbf{s}$. In Fig. 6, we have plotted the components of the error $\mathbf{e} = \mathbf{s}^* - \mathbf{s}$ (top) and of the joint velocities $\dot{\mathbf{q}}$ (bottom). The numbers correspond to the inserted screws (1 to 3). It is clear from the curves that the transitions between modes are abrupt in terms of $\dot{\mathbf{q}}$. This is because homotopy and adaptive gains are not used.

Let us now focus on the second, complete experiment (see Fig. 7), where all modes, as well as homotopy and adaptive gains, were applied. This time, we

[6] http://cs.stanford.edu/people/tkr/fri/html/

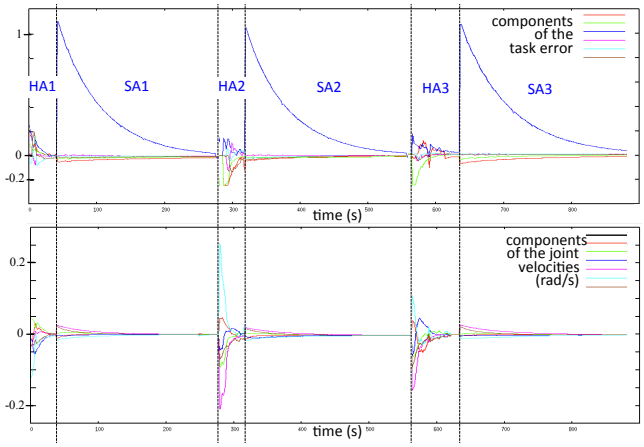

Figure 6: The six components of the error $\mathbf{e} = \mathbf{s}^* - \mathbf{s}$ (top) and the seven components of the joint velocities $\dot{\mathbf{q}}$ (bottom) during the first experiment.

Figure 7: Left: the tightening task. Right: consecutive snapshots of the second experiment.

verify that the cylindrical tool successfully encircles new, non-tightened screws (see photo on the left of Fig. 7). A high accuracy is required, since the screw external diameter and tool internal diameters are respectively 5 and 9 mm. Although, the specifications are more strict than in the previous experiment, force control facilitates the insertion, by correcting slight orientation errors. Hence, we are confident that, with a tightening tool, the approach should also work. Convergence of the hand approaching and screw approaching modes has been discussed just above. Let us now focus on the final mode, when force control intervenes. In Fig. 8, we have ${}^E f_{E,Z}$ and ${}^B Y_E$ during this final mode. The plots start at time t = 60 seconds, when the tool comes into contact with the flank. Correspondingly, ${}^E f_{E,Z}$ decreases from the null value, until the desired value is reached (in (51), we set ${}^E f^*_{E,Z} = -25$ N). Then, at $t \approx 65$ s, the end effector stops (see bottom graph). After a few seconds, a user (see snapshots in Fig. 7 and video) moves by hand the robot last joint. The forces are detected (see ${}^E f_{E,Z}$ in Fig. 8), and admittance control induces the small variations of ${}^B Y_E$. This experiment shows that the framework is capable of force stabilization, for safe human-robot interaction.

Finally, we ran experiments with and without the adaptive gains for Λ (experiments noted AG and FG), and with and without the smooth homotopy from hand to screw approaching (noted H and NH). To compare the experiments, in Fig. 9, we have plotted the norm of the joint velocity $|\dot{\mathbf{q}}|$, for three experiments: NH+FG, NH+AG, and H+AG. Since the change mainly concerns the hand and

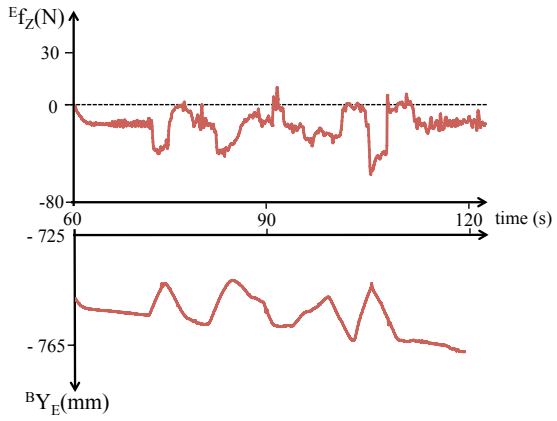

Figure 8: Evolution of ${}^{E}f_{E,Z}$ (top) and ${}^{B}Y_{E}$ (bottom) over time, during force-based screw approach and tightening.

the beginning of screw approaching, we have only plotted the curves for these phases. The mode change is visible in the plots, and occurs after approximately 5 seconds. The joint velocity norm has been chosen, since it is a good indicator of the movement smoothness. In the first experiment (NH+FG, green curve), which reproduces the approach used in [27], strong variations appear during the HA mode; these come from the variability of the positioning error $\mathbf{s}_p^* - \mathbf{s}_p$, due to hand motion. Replacing the fixed gain with an adaptive one yields the cyan curve. As the curves show, the use of an adaptive gain, reduces the shaky motion. Another improvement is obtained by adding an homotopy of duration

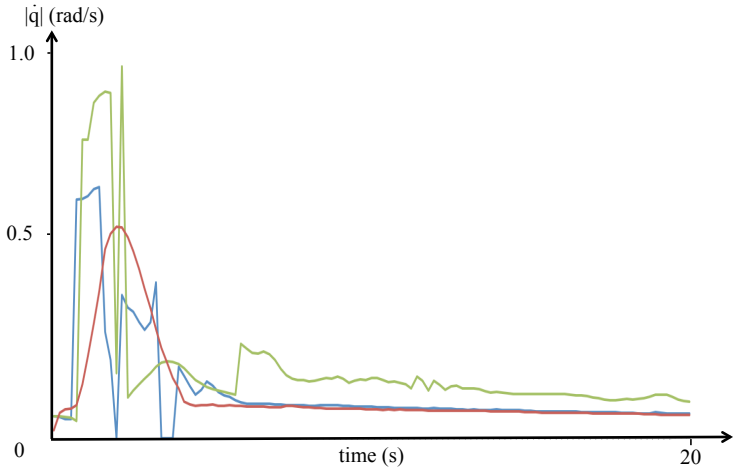

Figure 9: Evolution of $|\dot{\mathbf{q}}|$ over time, during hand and screw approaching, using NH+FG (green), NH+AG (cyan), and H+AG (red).

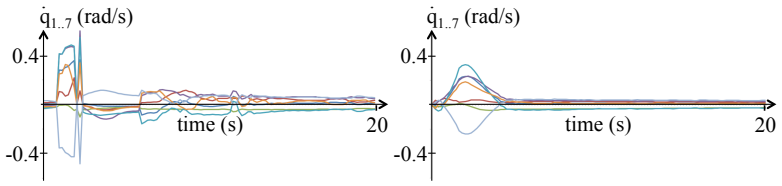

Figure 10: Evolution of $\dot{\mathbf{q}}$ components over time, during hand and screw approaching, using NH+FG (left), and H+AG (right).

Table 1: Comparison between $|\dot{\mathbf{q}}|$ and $|\ddot{\mathbf{q}}|$ in the four configurations.

configuration	FG	AG	NH	H
$\|\dot{\mathbf{q}}\|$ (rad s^{-1})	0.091	0.067	0.079	0.079
$\|\ddot{\mathbf{q}}\|$ (rad s^{-2})	0.132	0.082	0.145	0.068

$T = 1$ second when the screw is seen. In this case (red curve), the transition from hand to screw tracking is much smoother. To further demonstrate the smooth transitions obtained thanks to the adaptive gains and homotopies, in Fig. 10, we have compared the joint velocity components obtained with NH+FG (left), and with H+AG (right). As the figure shows, the curves are much smoother, and less control effort is required, with the new approach. We have also compared the average values of $|\dot{\mathbf{q}}|$ and $|\ddot{\mathbf{q}}|$ over each experiment. These, shown in Table 1, confirm the cited properties. For both metrics, AG outscores FG, by realizing the same operation in the same time with less velocity and acceleration. Also, as expected, homotopy reduces $|\ddot{\mathbf{q}}|$ (from 0.145 to 0.068 rad s^{-1}) since it realizes a smoothing effect on the joint velocities, but has no influence on $|\dot{\mathbf{q}}|$. Reducing $|\ddot{\mathbf{q}}|$ is crucial for safe human-robot interaction, since most robot safety metrics (see [42]), depend on accelerations measured at impacts.

In summary, the approach with both homotopy and adaptive gains (H+AG) should be selected. The advantages are numerous: less energy is required, the motion is smoother (facilitating image processing), and faster operation can be obtained. In fact, although, for the purpose of these comparisons, the gains were all tuned so that the duration of the experiments be the same, in other experiments we have fine tuned the gains of H+AG, to achieve screw approaching in 40 seconds, i.e., approximately 80% faster than in [27].

6. Conclusions

In this paper, we have generalized the multimodal framework for human-robot interaction originally introduced in [27]. The generalized framework can operate by activating or deactivating various tasks, according to the sensed data and to the needs of the application. This is of particular interest when numerous sensing devices are to be used for control, as is often the case in HRI. Typically,

in this work, we have applied the framework to a collaborative screw tightening experiment, where vision, kinect, position and force data must be alternatively controlled. To avoid abrupt accelerations, important features such as adaptive gains and homotopy are included in the framework.

This preliminary work opens numerous avenues for future research. In the future, we plan to use our framework for full human-robot cooperation, with direct physical interaction. It would then be possible to verify its robustness to unknown dynamic parameters, resulting from the interaction with the human body (e.g., arm and hand).

Acknowledgements

Work supported by the ANR (French National Agency) ICARO project.

References

[1] A. Bicchi, M. Peshkin and J. Colgate, "Safety for physical human-robot interaction", *Springer Handbook of Robotics*, B. Siciliano, O. Khatib (Eds.), Springer, 2008, pp. 1335-1348.

[2] A. De Santis, B. Siciliano, A. De Luca and A. Bicchi, "An atlas of physical human-robot interaction", 2008, *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253-270.

[3] A. De Luca and F. Flacco, "Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration", *IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics BIOROB*, 2012.

[4] L. Sentis and O. Khatib, "A Whole-Body Control Framework for Humanoids Operating in Human Environments", *IEEE Int. Conf. on Robotics and Automation ICRA*, 2006.

[5] M. S. Erden and A. Billard, "End-point Impedance Measurements at Human Hand during Interactive Manual Welding with Robot", *IEEE Int. Conf. on Robotics and Automation ICRA*, 2014.

[6] X. Li and C. C. Cheah, "Human-Guided Robotic Manipulation: Theory and Experiments", *IEEE Int. Conf. on Robotics and Automation ICRA*, 2014.

[7] M. S. Erden, B. Maric, "Assisting manual welding with robot", *Robotics and Computer Integrated Manufacturing*, 2011, vol. 27, pp. 818-828.

[8] G. Grunwald, G. Schreiber, A. Albu-Schaffer and G. Hirzinger, "Touch: The direct type of human interaction with a redundant service robot", *IEEE Int. Workshop on Robot and Human Interactive Communication, ROMAN*, 2001.

[9] M. S. Erden and T. Tomiyama, "Human intent detection and physically-interactive control of a robot without force sensors", *IEEE Trans. on Robotics*, 2010, vol. 26, no. 2, pp. 370-382.

[10] O. Khatib, E. Demircan, V. DeSapio, L. Sentis, T. Besier, S. Delp, "Robotics-based Synthesis of Human Motion", *Journal of Physiology - Paris*, 2009, Vol. 103(3-5), pp. 211-219.

[11] Microsoft Corporation, 1 Microsoft Way, Redmond, WA 98052-7329, USA, "Microsoft Kinect homepage. http://xbox.com/Kinect (accessed: Aug. 19, 2014)", *Internet*, 2014.

[12] Y. Maeda, T. Hara and T. Arai, "Human-robot cooperative manipulation with motion estimation" *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2001, Vol. 4, pp. 2240-2245.

[13] L. Villani and J. De Schutter, "Force Control", *Springer Handbook of Robotics*, B. Siciliano, O. Khatib (Eds.), Springer, 2008, pp. 161-184.

[14] V. Lippiello, B. Siciliano, L. Villani, "Interaction Control of Robot Manipulators Using Force and Vision", *Int. Journal of Optomechatronics*, 2008, vol. 2, no. 3, 257274.

[15] F. Flacco, T. Kröger, A. De Luca and O. Khatib, "A Depth Space Approach to Human-Robot Collision Avoidance", *IEEE Int. Conf. on Robotics and Automation ICRA*, 2012, pp. 338-345.

[16] J. A. Corrales, G. J. Garcia Gomez, F. Torres Medina and V. Perdereau, "Cooperative Tasks between Humans and Robots in Industrial Environments", *International Journal of Advanced Robotic Systems*, 2012, Vol. 9 No. 94, pp. 1-10.

[17] F. Chaumette and S. Hutchinson, "Visual servo control", *IEEE Robotics and Automation Magazine*, Vol. 13, no. 4, 2006, pp. 82–90 and Vol. 14, no. 1, 2007, pp. 109–118.

[18] S. M. La Valle, "Planning Algorithms", Cambridge University Press, 2006.

[19] A. De Santis, V. Lippiello, B. Siciliano and L. Villani, "Human-Robot Interaction Control Using Force and Vision", *Advances in Control Theory and Applications*, 2007, Vol. 353, pp. 51–70.

[20] B. J. Nelson, "Improved force control through visual servoing", *American Control Conference*, 1995.

[21] K. Hosoda, K. Igarashi, and M. Asada, "Adaptive hybrid visual servoing/force control in unknown environment", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 1996.

[22] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators", *ASME Journal of Dynamic Systems, Measurement, and Control*, 1981, Vol. 103, p. 126–133.

[23] J. Baeten, H. Bruyninckx and J. De Schutter, "Integrated vision/force robotic servoing in the task space formalism", *International Journal of Robotics Research*, 2003, vol. 22, no.10-11, pp. 941–954.

[24] N. Hogan, "Impedance control: an approach to manipulation: parts I-III", *ASME Journal of Dynamic Systems, Measurement, and Control*, 1985, Vol. 107, p. 1–24.

[25] G. Morel, E. Malis and S. Boudet, "Impedance based combination of visual and force control", *IEEE Int. Conf. on Robotics and Automation ICRA*, 1998.

[26] M. Prats, P. J. Sanz, A. P. Del Pobil, "Vision-tactile-force integration and robot physical interaction", *IEEE Int. Conf. on Robotics and Automation ICRA*, 2009.

[27] A. Cherubini, R. Passama, A. Meline, A. Crosnier and P. Fraisse, "Multimodal control for human-robot cooperation", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2013.

[28] B. Siciliano, L. Sciavicco, L. Villani and G. Oriolo, "Robotics: Modelling, Planning and Control", Springer, 2009.

[29] N. Mansard, O. Stasse, P. Evrard and A. Kheddar, "A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks", *Int. Conf. on Advanced Robotics, ICAR*, 2009.

[30] J. De Schutter, T. De Laet, J. Rutgeerts, W. Decr, R. Smits, E. Aertbelin, K. Claes and H. Bruyninckx, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty", *Int. Journal of Robotics Research*, 2007, Vol. 26, no. 5, pp. 433-455.

[31] N. M. Ceriani, A. M. Zanchettin, P. Rocco, A. Stolt and A. Robertsson, "A constraint-based strategy for task-consistent safe human-robot interaction", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2013.

[32] M. Prats, P. J. Sanz and A. P. Del Pobil, "Reliable non-prehensile door opening through the combination of vision, tactile and force feedback", *Autonomous Robots*, 2010, Vol. 29, no. 2, pp. 201–218.

[33] G. Antonelli, F. Arrichiello and S. Chiaverini, "The NSB control: a behavior-based approach for multi-robot systems", Paladyn Journal of Behavioral Robotics, vol. 1, no. 1, pp. 48-56, 2010.

[34] N. Mansard, O. Khatib and A. Kheddar, "A Unified Approach to Integrate Unilateral Constraints in the Stack of Tasks", *IEEE Trans. on Robotics*, 2009, Vol. 25, no. 3.

[35] J. Park, J. Lee, and N. Mansard, "Intermediate Desired Value Approach for Task Transition of Robots in Kinematic Control", *IEEE Trans. on Robotics*, 2012, Vol. 28, no. 6, pp. 1260 – 1277 .

[36] L. Saab, O. Ramos, F. Keith, N. Mansard, P. Soueres and J-Y. Fourquet, "Dynamic Whole-Body Motion Generation under Rigid Contacts and other Unilateral Constraints", *IEEE Trans. on Robotics*, 2013, Vol. 29 (2), pp. 346–362.

[37] K. Waldron and J. Schmiedeler, "Kinematics", *Springer Handbook of Robotics*, B. Siciliano, O. Khatib (Eds.), Springer, 2008, pp. 9-33.

[38] E. Malis, F. Chaumette and S. Boudet, "2-1/2D visual servoing", *IEEE Trans. Robot. Automat.*, 1999, Vol. 15, no. 2, pp. 238-250.

[39] T. Kröger and B. Finkemeyer, "Robot motion control during abrupt switchings between manipulation primitives", *Workshop on Mobile Manipulation at the IEEE Int. Conf. on Robotics and Automation ICRA*. 2011.

[40] A. Liegeois, "Automatic supervisory control of configurations and behavior of multibody mechanisms", *IEEE Trans. on Systems, Man, and Cybernetics*, 1977, vol. 7, no. 6, pp. 868-871.

[41] E. Marchand, F. Spindler and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills", *IEEE Robotics and Automation Magazine, Special Issue on "Software Packages for Vision-Based Control of Motion"*, 2005, Vol. 12, no. 4, pp. 40–52.

[42] J. Versace, "A review of severity index", *Proceedings of Stapp Car Crash Conference*, 1971, pp. 149170.