Human-Robot Collaboration for Safe Object Transportation Using Force Feedback

J. Ernesto Solanes[a], Luis Gracia[*a], Pau Muñoz-Benavent[a],
Jaime Valls Miro[b], Marc G. Carmichael[b], Josep Tornero[a]

[a]*Instituto IDF, Universitat Politecnica de Valencia, Camino de Vera s/n, 46022 Valencia, Spain (e-mail: luigraca@isa.upv.es).* [*]*Corresponding author*
[b]*Centre for Autonomous Systems (CAS), Faculty of Engineering, University of Technology Sydney (UTS), NSW 2007 Sydney, Australia.*

Abstract

This work presents an approach based on multi-task, non-conventional sliding mode control and admittance control for human-robot collaboration aimed at handling applications using force feedback. The proposed robot controller is based on three tasks with different priority levels in order to cooperatively perform the safe transportation of an object with a human operator. In particular, a high-priority task is developed using non-conventional sliding mode control to guarantee safe reference parameters imposed by the task, e.g., keeping a load at a desired orientation (to prevent spill out in the case of liquids, or to reduce undue stresses that may compromise fragile items). Moreover, a second task based on a hybrid admittance control algorithm is used for the human operator to guide the robot by means of a force sensor located at the robot tool. Finally, a third low-priority task is considered for redundant robots in order to use the remaining degrees of freedom of the robot to achieve a pre-set secondary goal (e.g., singularity avoidance, remaining close to a homing configuration for increased safety, etc.) by means of the gradient projection method. The main advantages of the proposed method are robustness and low computational cost. The applicability and effectiveness of the proposed approach is substantiated by experimental results using a redundant 7R manipulator: the Sawyer collaborative robot.

Keywords: cooperative task, robot system, force control, sliding mode control

1. Introduction

1.1. Motivation

Recent advances in technology and robotics are revolutionizing modern society. Robots are becoming more and more present in the form of unmanned aircraft systems, commonly known as drones, driver-less cars, robot-assisted surgery and rehabilitation systems, robotic prosthetics and exoskeletons, service robots for personal and domestic use, artificial assistants and smart machines, among others.

Possibly, the manufacturing industry in general has been the most benefited by advances in the fields of robotics, control and sensing, bringing in improvements to production processes as well as worker's ergonomics and job quality. Contrary to the old tendency of developing autonomous systems to replace humans by robotic devices, currently the research is more focused on developing robots to work alongside humans and assist them. The reason is that the combination of human cognitive and sensorimotor skills with the technical capabilities of a robot have proven able to solve, facilitate, improve and/or speed up a large variety of complex tasks that neither humans nor robots could successfully afford to do in solitary [1, 2, 3, 4, 5, 6].

A case in point in the manufacturing sector is the manipulation and safe transportation of precarious loads, most notably heavy objects such as car engines, or fragile items such as glass, liquid containers, hazardous materials etc. Moreover, in many cases loads have to be transported and awkwardly deposited in difficult-to-access areas that make it difficult or ergonomically challenging for an operator to keep them in a predetermined position and/or orientation, whilst simultaneously pursuing a higher level assembling or handling assignment. Yet the automatic realization of these type of assignments by a robot is usually discarded due to the limited flexibility afforded by a robot in adapting to changes in the production workspace. Hence, the combination of skillful guidance by the human operator on the one hand, and the sensorimotor stability and strength of the robot on the other can lead to industrially feasible human-robot collaborative solutions for handling applications. Fig. 1 shows an example of a factory operator in the automotive industry being assisted by a robot whilst installing a shock absorber in a car body. In this case, the human is merely guiding the robot tool to meet the exact position where the shock absorber has to be placed, whilst the robot

Fig. 1. Operator and robotic manipulator working side-by-side in Cologne (Germany) Ford's factory (image courtesy of KUKA Robots Ibérica, S.A, Vilanova i la Geltrú, Spain).

is automatically accounting for the shock absorber's weight as it moves [1].

1.2. Literature review

Generally, guidance for the motion of the manipulator is obtained via a wrist-mounted force sensor which evaluates the forces exerted by the human operators. The most commonly used method to convert these measurements into kinematic instructions to the robot is through compliance control, which establishes a direct relationship between the measured forces and the changes in the robot position [7, 8, 9]. Yet other variants and methods can be found in the literature. For instance, authors in [10] presented a method to determine the compliance controller parameters of the physical model using a particle swarm optimization algorithm for a spinal surgery application. In [11] a force tracking method under the impedance control framework was extended to also account for uncertain human limb dynamics. An adaptive controller was developed to deal with point-to-point movements, whereas learning and neural network controls were included to generate periodic and arbitrary continuous trajectories respectively. A hierarchical control system was presented in [12] for the co-manipulation task of folding sheets like fabrics/cloths. The system was based on force and RGB-D feedback at two distinctive control levels. At a higher level, the perception of the human's

---

[1]For more information the reader is referred to
https://media.upv.es/player/?id=cec9a940-8924-11e7-90ea-23686ce0f1be

intention was used to decide on the robot's action, whereas at a lower level the robot reacted to force and RGB-D feedback in following the guidance from the human. In [13] a decision-and-control architecture was proposed for hand-arm systems with "soft robotics" capabilities via dedicated human-machine interfaces. The robot was controlled through a multi-priority Cartesian impedance controller, and the behavior extended with collision detection and reflex reactions. The problem of human-directed position/force control of a robot end-effector interacting with an environment given unknown geometry and stiffness was addressed in [14]. In free space (non-contact) motion, the input was interpreted as a linear velocity command. When contact occurred, a generalized damper-type of impedance control was used for the regulation of force in the constrained direction, while the input from the user would adjust the contact force set point. In [15] a mathematical relation between the velocity of the human-robot interaction point and the force applied by the human operator was established using impedance control for handling tasks, where an adjustable force threshold was used to enable the operator to keep authority over the robot motion. An optimal impedance adaptation was investigated in [16] for interaction control in constrained motions, which lead to an optimal realization of trajectory tracking and force regulation.

Other approaches to human-robot collaborative applications driven by force-control strategies are based on sliding mode control (SMC) theory given its inherent robustness and low computational cost characteristics [17]. For instance, in [18] a non-singular terminal SMC was developed to ensure trajectory tracking precision for the case of a lower limb rehabilitation parallel robot. The device would adjust the gait trajectory online according to the indications from a human-machine interaction force set-up. In [19] a human-robot interaction controller was introduced for a lower extremity exoskeleton whose aim was to improve tracking performance with the development of a fuzzy SMC that considered system uncertainties. This way the controller was able to drive the exoskeleton to shadow the wearer in the presence of weaker interactive driving forces . In [20] a proxy-based SMC was proposed to obtain effective tracking during normal operations for flexible joint manipulators working close to humans, whilst retaining the ability to recover from positional errors in a smooth and damped manner. In [21] a robust SMC was proposed that relied on basic information from the human subject (weight, height, age and gender) to handle model uncertainties due to biomechanical variation of patients using an upper limb rehabilitation robot. An SMC consisting of a proportional-integral-derivative sliding surface and a fuzzy hitting

control law was developed in [22] to guarantee robust tracking performance and reduce the chattering effect for a class of robot-assisted therapeutic exoskeleton. A fuzzy SMC presented in [23] considered a non-linear model for trajectory tracking of micro robots in the human vasculature system. The control of a four-degrees-of-freedom haptic master for robot-assisted minimally invasive surgery in [24] relied on classic SMC practices to achieve the desired torque/force trajectories. It is worth mentioning that SMC has been recently used in the field of robot force control not only to improve controller robustness but also to improve force estimation by means of a sliding perturbation observer (SPO) in order to avoid the use of expensive force sensors. For examples of this approach, see [25, 26].

1.3. Proposal

The work hereby proposed also exploits SMC so that robot and human operator can cooperatively and safely undertake the transportation of objects with the aid of force feedback sensing, guaranteeing the physical integrity of both the transported object and human operator. However, the SMC robot controller in this work offers several distinctive features that sets it apart from other works in the literature. Hence, the introduction of inequality constraints within the SMC framework is a key novelty. Moreover, the strategy also relies on the use of a non-conventional SMC regulator to fulfill the actual constraints. In particular, the strategy constrains a subset of the robot pose coordinates by prudent reference values imposed by the handling operation, e.g., keeping a load at a desired orientation to prevent spill out in the case of liquids, or to reduce undue stresses that may compromise fragile items as in glass transportation. Given the multi-layered nature of the proposed strategy, remaining degrees of freedom in the robot pose are thus left to be guided by the human operator in a lower priority loop using a force sensor located at the robot tool to detect his desired movements to accomplish the task safely and concurrently. A low-priority task for the specific case of redundant robots (as is the case with the examples shown in this work) is also suggested to achieve a pre-set secondary goal, e.g., singularity avoidance, remaining close to a homing configuration for increased safety, etc.

1.4. Structure of the paper

The remainder of the paper is organized as follows. Next section introduces some preliminaries, while Section 3 develops the non-conventional SMC used in this work. The proposed hybrid control approach is presented

in Section 4, while some important remarks about the method are given in Section 5. The feasibility and robustness of the proposed approach is substantiated by experimental results in Section 6 using a redundant 7R manipulator. Finally, some key conclusions drawn from the work are collected in Section 7.

2. Preliminaries

2.1. Kinematics

The robot *pose* $\mathbf{p} = \begin{bmatrix} x & y & z & \alpha & \beta & \gamma \end{bmatrix}^{\mathrm{T}}$ with orientation expressed in roll-pitch-yaw angles and the robot *configuration* $\mathbf{q} = \begin{bmatrix} q_1 & \cdots & q_n \end{bmatrix}^{\mathrm{T}}$, and their derivatives, are related by the following equations:

$$\mathbf{p} = \mathbf{l}(\mathbf{q}) \tag{1}$$

$$\dot{\mathbf{p}} = \frac{\partial \mathbf{l}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{J}\dot{\mathbf{q}} \tag{2}$$

$$\ddot{\mathbf{p}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}}, \tag{3}$$

where vector $\mathbf{l}$ and matrix $\mathbf{J}$ are the kinematic function and analytical Jacobian [27] of the robot, respectively.

In this work, the so-called geometric Jacobian [27] relative to the tool coordinate system $\mathbf{J}_n$ is also required. This matrix transforms the joint velocities $\dot{\mathbf{q}}$ to the linear and angular velocities of the end-effector relative to the tool coordinate system.

On the one hand, the analytical Jacobian $\mathbf{J}$ is required in this work to control a subset of the robot pose coordinates in order to fulfill prudent reference values imposed by the handling operation. On the other hand, the geomentric Jacobian $\mathbf{J}_n$ is required in this work to guide the robot tool by means of the human operator's forces detected by a force sensor located at the end-effector. The relationship between the analytical Jacobian $\mathbf{J}$ and the geometric Jacobian $\mathbf{J}_n$ is given by [28]:

$$\mathbf{J} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{O}_3 \\ \mathbf{O}_3 & \begin{bmatrix} 1 & 0 & \sin(\beta) \\ 0 & \cos(\alpha) & -\sin(\alpha)\cos(\beta) \\ 0 & \sin(\alpha) & \cos(\alpha)\cos(\beta) \end{bmatrix}^{-1} \end{bmatrix} \begin{bmatrix} {}^{\mathbf{0}}\mathbf{R_n} & \mathbf{O}_3 \\ \mathbf{O}_3 & {}^{\mathbf{0}}\mathbf{R_n} \end{bmatrix} \mathbf{J}_n = \mathbf{L}\,\mathbf{R}\,\mathbf{J}_n, \tag{4}$$

where α and β are the roll and pitch angles, respectively, $\mathbf{^0R_n}$ is the rotation matrix between the base and tool coordinate systems, $\mathbf{O}_3$ is the 3×3 zero matrix, $\mathbf{I}_3$ is the 3×3 identity matrix, matrix $\mathbf{L}$ is required to transform the end-effector angular velocities to the derivatives of roll-pitch-yaw angles and $\mathbf{R}$ represents a projection matrix.

2.2. Robot control

This work assumes the existence of a low-level robot controller in charge of achieving a particular joint acceleration from the commanded acceleration $\ddot{\mathbf{q}}_c$, and that its dynamics is fast enough compared to that of $\ddot{\mathbf{q}}_c$. Hence, the relationship:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_c + \mathbf{d}_c \tag{5}$$

holds approximately true, where $\mathbf{d}_c$ represents inaccuracies due to disturbances. Note that the *dynamic model* of the robot system should be taken into account to properly design the mentioned underlying joint controller.

2.3. Task-priority based redundancy resolution

It is useful to consider the task-priority strategy [29] to tackle several (possibly incompatible) objectives simultaneously assigning an order of priority to each one. Thus, a lower-priority task is satisfied only by using the degrees of freedom in the null space of the higher-priority ones [30]. When an exact solution is not possible for a given task at a particular priority level, its error is minimized. The formulation for this approach is detailed below. Let us consider M tasks which consist on calculating the vector $\bar{\mathbf{x}}$ satisfying the following equalities:

$$\mathbf{A}_i \bar{\mathbf{x}} = \mathbf{b}_i, \quad i = 1, \dots, M, \tag{6}$$

where matrix $\mathbf{A}_i$ and vector $\mathbf{b}_i$ of the ith task are assumed known and index i represents the priority order: $i = 1$ for highest priority and $i = M$ to lowest.

The solution $\bar{\mathbf{x}}_M$ that hierarchically minimizes the error of equations in (6) is given by the following recursive formulation, proposed in [31]:

$$\bar{\mathbf{x}}_i = \bar{\mathbf{x}}_{i-1} + (\mathbf{A}_i \mathbf{N}_{i-1})^\dagger (\mathbf{b}_i - \mathbf{A}_i \bar{\mathbf{x}}_{i-1}) \tag{7}$$

$$\mathbf{N}_i = \mathbf{N}_{i-1}(\mathbf{I} - (\mathbf{A}_i \mathbf{N}_{i-1})^\dagger (\mathbf{A}_i \mathbf{N}_{i-1})) \tag{8}$$

$$i = 1, \dots, M, \quad \ddot{\mathbf{q}}_{c,0} = \mathbf{0}, \mathbf{N}_0 = \mathbf{I},$$

where $\mathbf{I}$ and $\mathbf{0}$ denote the identity matrix and zero column vector, respectively, of suitable size, superscript † denotes the Moore-Penrose pseudoinverse and $\overline{\mathbf{x}}_i$ and $\mathbf{N}_i$ are the solution vector and null-space projection matrix, respectively, for the set of first i tasks. The pseudoinverse may be computed via the singular value decomposition (SVD) method [32] and using a tolerance to set to zero the very small singular values in order to avoid extremely large values for the commanded accelerations.

Note that the algebraic problem given by the tasks in (6) is generic from a mathematical point of view and can be solved via the recursive formulation in (7) and (8) regardless of the nature of the real problem to be solved.

3. Non-conventional Sliding Mode Control

The non-conventional SMC proposed in this work to satisfy inequality constraints is described as follows. The state space of the system is divided into the non-allowed region $\mathbf{A}$ and the allowed region $\mathbf{B}$, which are separated by the inequality constraint boundary. Similarly to conventional SMC, when the system sate is in the non-allowed region $\mathbf{A}$ the control action $\mathbf{u} = \mathbf{u}_B$ pushes the system into the allowed region $\mathbf{B}$. But in contrast to conventional SMC, when the system state is in the allowed region $\mathbf{B}$, no control action is applied, i.e., $\mathbf{u} = \mathbf{0}$. Hence, if the system starts in region $\mathbf{A}$, i.e., $\mathbf{x}_A(0)$, it evolves to the boundary of the constraint, which is known as *reaching mode* [17]. Nevertheless, when the system starts in the allowed region $\mathbf{B}$, the system state can "freely" evolve according to some other criterion, e.g., a control law for reference tracking. Therefore, only when the state trajectory tries by itself to leave the allowed region, the non-conventional SMC will make $\mathbf{u}$ switch between $\mathbf{0}$ and $\mathbf{u}_B$ at a theoretically infinite frequency, which can be seen as an ideal sliding mode (SM) behavior [33]. A *continuous equivalent control* [33] can be obtained for the SM phase, i.e., the control required to keep the system on the sliding surface. Hence, SMC produces such control action without explicit knowledge of it and with a low computational cost, which is a typical advantage of SMC strategies [17].

The proposed non-conventional SMC to satisfy inequality constraints is presented in the theorem below.

Theorem 1. *Consider the following dynamical system with n_x states and n_u inputs given by:*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{d}) + \mathbf{g}(\mathbf{x})\, \mathbf{u}, \tag{9}$$

where $\mathbf{x}(t)$ is the state vector, $\mathbf{d}(t)$ is an unmeasured disturbance or model uncertainty, $\mathbf{u}(t)$ is the control input vector (possibly discontinuous), $\mathbf{f}$ is a vector field and $\mathbf{g}$ is a set of vector fields.

Consider also that the system state vector $\mathbf{x}$ is subject to inequality constraints $\phi_i(\mathbf{x}) \leq 0$, $i = 1, \ldots, N$, where $\phi_i(\mathbf{x})$ is the ith inequality constraint function. Thus, the region Φ of the state space compatible with the constraints on state $\mathbf{x}$ is given by:

$$\Phi = \{\mathbf{x} \,|\, \phi_i(\mathbf{x}) \leq 0\}, \tag{10}$$

with $i = 1, \ldots, N$.

Then, assuming that the constraint functions ϕ_i are differentiable, the control action $\mathbf{u}$ that fulfills the variable structure control below guarantees that the system converges to Φ in finite time and remains there henceforth:

$$\mathrm{v2dm}\left(\mathrm{pos}\left(\boldsymbol{\phi}\right)\right) \mathbf{L_g}\boldsymbol{\phi}\, \mathbf{u} = -\mathbf{W}\, \mathrm{pos}\left(\boldsymbol{\phi}\right)\, u^+ \tag{11}$$

$$u^+ > \sum_{i=1}^{n_a} \left(\max(L_f\phi_i, 0)\right)/\mathrm{diag_{min}}(\mathbf{W}), \tag{12}$$

where function $\mathrm{v2dm}(\cdot)$ converts a vector into a diagonal matrix, function $\mathrm{pos}(\cdot)$ represents the positive function (i.e., $\mathrm{pos}(x)$ is equal to 0 if $x < 0$ and equal to 1 if $x > 0$), matrix $\mathbf{L_g}\boldsymbol{\phi}$ contains the row vectors $\mathbf{L_g}\phi_i$ of all inequality constraints, the scalar $L_f\phi_i$ and the row vector $\mathbf{L_g}\phi_i$ denote the Lie derivatives of the inequality constraints in the direction of vector field $\mathbf{f}$ and in the direction of the set of vector fields $\mathbf{g}$, respectively, $\boldsymbol{\phi}$ is a column vector with all the inequality constraint functions ϕ_i, positive scalar u^+ is the so-called switching gain, which can be either constant or varying in time, $\mathbf{W}$ is a diagonal matrix representing the switching gain weights for the inequality constraints, n_a is the number of active inequality constraints, i.e., those with $\phi_i \geq 0$, and function $\mathrm{diag_{min}}(\cdot)$ computes the minimum value of the diagonal elements of a matrix..

Note that the expression $\mathrm{v2dm}(\mathrm{pos}(\boldsymbol{\phi}))$ on the left-side of (11) is used to obtain the trivial scalar equation $0 = 0$ for the non-active inequality constraints (i.e., those with $\phi_i < 0$) and, hence, no degrees of freedom of the system are used by these constraints.

Proof. Firstly, the inequality constraint vector is partitioned into two sub-vectors $\boldsymbol{\phi} = [\boldsymbol{\phi}^{n_a\,\mathrm{T}} \quad \boldsymbol{\phi}^{N-n_a\,\mathrm{T}}]^{\mathrm{T}}$, where the first subvector is composed of the n_a active inequality constraints (i.e., those with $\phi_i \geq 0$) and the second

9

subvector of the remaining non-active inequality constraints (i.e., those with $\phi_i < 0$).

Assuming that $\boldsymbol{\phi}^{n_a}(0) > \mathbf{0}$, the goal of this proof is to show that convergence to point $\boldsymbol{\phi}^{n_a} = \mathbf{0}$ is achieved in finite time.

The column vector $\dot{\boldsymbol{\phi}}$ composed of the constraint function derivatives $\dot{\phi}_i$ is given by

$$
\begin{aligned}
\dot{\boldsymbol{\phi}} &= \frac{\partial \boldsymbol{\phi}^{\mathrm{T}}}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, \mathbf{d}) + \frac{\partial \boldsymbol{\phi}^{\mathrm{T}}}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) \, \mathbf{u} \\
&= L_f \boldsymbol{\phi} + \mathbf{L_g} \boldsymbol{\phi} \mathbf{u}.
\end{aligned}
\tag{13}
$$

Premultiplying (13) by $\mathrm{v2dm}\,(\mathrm{pos}\,(\boldsymbol{\phi}))$ and substituting (11) yields:

$$
\mathrm{v2dm}(\mathbf{z})\dot{\boldsymbol{\phi}} = \mathrm{v2dm}(\mathbf{z})L_f\boldsymbol{\phi} - \mathbf{W}\,\mathbf{z}\,u^+,
\tag{14}
$$

where $\mathbf{z}$ is a column vector with the ith-component $z_i = 1$ if $\phi_i > 0$ and $z_i = 0$ if $\phi_i < 0$.

Let $V = \mathbf{z}^{\mathrm{T}}\,\mathrm{v2dm}(\mathbf{z})\,\boldsymbol{\phi}$ be a Lyapunov function candidate. Vector $\boldsymbol{\phi}^{n_a}$ can be generically partitioned into two subvectors $\boldsymbol{\phi}^{n_a} = [\boldsymbol{\phi}^{b\,\mathrm{T}} \quad \boldsymbol{\phi}^{n_a-b\,\mathrm{T}}]^{\mathrm{T}}$, where SM occurs in the manifold given by $\boldsymbol{\phi}^b = \mathbf{0}$, whereas the components of vector $\boldsymbol{\phi}^{n_a-b}$ are greater than zero. Obviously, one of these two subvectors may be empty at a certain time. Since vectors $\mathbf{z}^{n_a-b} = \mathbf{1}$ and $\mathbf{z}^{N-n_a} = \mathbf{0}$ are constant, the time derivative of V results in:

$$
\begin{aligned}
\dot{V} = \frac{d}{dt}\left(\begin{bmatrix} \mathbf{z}^b \\ \mathbf{1} \\ \mathbf{0} \end{bmatrix}^{\mathrm{T}} \mathrm{v2dm}\left(\begin{bmatrix} \mathbf{z}^b \\ \mathbf{1} \\ \mathbf{0} \end{bmatrix} \right) \right) \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\phi}^{n_a-b} \\ \boldsymbol{\phi}^{N-n_a} \end{bmatrix} \\
+ \mathbf{z}^{\mathrm{T}}\,\mathrm{v2dm}(\mathbf{z})\,\dot{\boldsymbol{\phi}} = \mathbf{z}^{\mathrm{T}}\,\mathrm{v2dm}(\mathbf{z})\,\dot{\boldsymbol{\phi}}.
\end{aligned}
\tag{15}
$$

Substituting (14) in (15) yields:

$$
\dot{V} = \mathbf{z}^{\mathrm{T}}\,\mathrm{v2dm}(\mathbf{z})\,L_f\boldsymbol{\phi} - \mathbf{z}^{\mathrm{T}}\,\mathbf{W}\,\mathbf{z}\,u^+.
\tag{16}
$$

Since $\mathbf{z}^{N-n_a} = \mathbf{0}$ and the components of vector $\mathbf{z}^{n_a}$ range from 0 to 1, the upper bound of the first term in (16) is given by $z_i^{n_a} = 1$ if $L_f\phi_i^{n_a} > 0$ and

10

$z_i^{n_a} = 0$ if $L_f \phi_i^{n_a} < 0$, that is:

$$\mathbf{z}^{\mathrm{T}} \, \text{v2dm}(\mathbf{z}) \, L_f \boldsymbol{\phi} \leq \sum_{i=1}^{n_a} \left(\max(L_f \phi_i, 0) \right). \tag{17}$$

Assuming that $u^+ > 0$, the second term in (16) is negative, since matrix $\mathbf{W}$ is positive definite, and its upper bound is given by:

$$- \mathbf{z}^{\mathrm{T}} \, \mathbf{W} \, \mathbf{z} \, u^+ \leq -\text{diag}_{\min}(\mathbf{W}) \, \|\mathbf{z}\|_2^2 \, u^+$$
$$\text{where } \|\mathbf{z}\|_2 \geq 1 \ \forall \, \boldsymbol{\phi} > \mathbf{0}, \tag{18}$$

because if vector $\boldsymbol{\phi}^{n_a - b}$ is not empty at least one component of vector $\mathbf{z}$ is equal to 1.

From (17) and (18), the upper bound of the time derivative of the Lyapunov function V results in:

$$\dot{V} \leq \sum_{i=1}^{n_a} \left(\max(L_f \phi_i, 0) \right) - \text{diag}_{\min}(\mathbf{W}) \, u^+. \tag{19}$$

Therefore, if u^+ fulfills (12) the Lyapunov function decays at a finite rate, it vanishes and collective SM in the intersection of the n_a active inequality constraints occurs after a finite time interval. $\qquad \square$

Chattering. Discrete-time implementations of any practical SMC makes the system leave the ideal SM and oscillate with finite frequency and amplitude inside a band around $\boldsymbol{\phi} = \mathbf{0}$, which is called *chattering* [17]. The upper bound for the chattering band $\triangle \boldsymbol{\phi}$ of the proposal can be obtained using the Euler-integration of the discontinuous control action given by Eq. (11):

$$\triangle \boldsymbol{\phi} = T_s \, |\mathbf{L_g} \boldsymbol{\phi} \, \mathbf{u}| = T_s \, u^+ \, \text{dm2v}(\mathbf{W}), \tag{20}$$

where T_s is the sampling time of the robot system and function $\text{dm2v}(\cdot)$ gives a column vector with the diagonal elements of a square matrix.

4. Proposed method

4.1. Overview of the method

The objective of this work is to obtain a hybrid position/force robot control so that the robot and the human operator cooperatively manipulate and

transport loads/objects. In particular, some coordinates of the robot pose are controlled to achieve the reference values imposed by the transport task, e.g., to keep a reference orientation, to maintain verticality, etc. Meanwhile, the remaining coordinates of the robot pose are guided by the human operator using a *force sensor* located at the robot tool to detect his desired driving forces.

Fig. 2 shows the block diagram of the proposed method. Three tasks with different priority levels are considered. Level 1 (high-priority task) is developed using the SMC theories described in Section 3 in order to fulfill a set of inequality constraints to ensure the required transport conditions for the object. Level 2 (medium-priority task) is based on a hybrid admittance control algorithm including a switching term that is used for the robot to track the human operator's forces. Note that the forces exerted by the human operator are detected by a force sensor located at the robot tool regardless of the point of application, whether in the handle located at the end-effector, or directly through the actual object being transported—labeled in Fig. 2 with numbers 1 and 2 inside a circle, respectively. Finally, Level 3 (low-priority task) is considered for the case of redundant robots in order to use the remaining degrees of freedom of the robot to achieve secondary goals, e.g., singularity avoidance, keeping the robot configuration close to an arbitrary rest position such as home for increased safety, etc.

The input to all three levels is the robot state $\{\mathbf{q}, \dot{\mathbf{q}}\}$ obtained from the robot controller. Moreover, Level 2 has also as input the vector $\mathbf{F}$ of forces and torques measured by a sensor located at the robot tool, which has already been filtered by the sensor electronics. Each level yields an acceleration equality of the form $\mathbf{A}_i \overline{\mathbf{x}} = \mathbf{b}_i$ (6) whose square error must be minimized, where the vector to be established $\overline{\mathbf{x}}$ corresponds to the commanded joint acceleration vector $\ddot{\mathbf{q}}_c$. For this purpose, the task priority redundancy resolution given by Eqs. (7) and (8) is used to obtain the commanded acceleration $\ddot{\mathbf{q}}_{c,3}$, which is integrated and sent to the robot controller. Finally, the robot controller performs a low-level control loop to track the commanded velocity $\dot{\mathbf{q}}_c$ using the current angles $\mathbf{q}$ and torques $\boldsymbol{\tau}$ measured by the joint sensors. Thus, the difference between the commanded and actual joint velocities is bounded and given by the error $\mathbf{d}_c$ of the low-level joint controller.
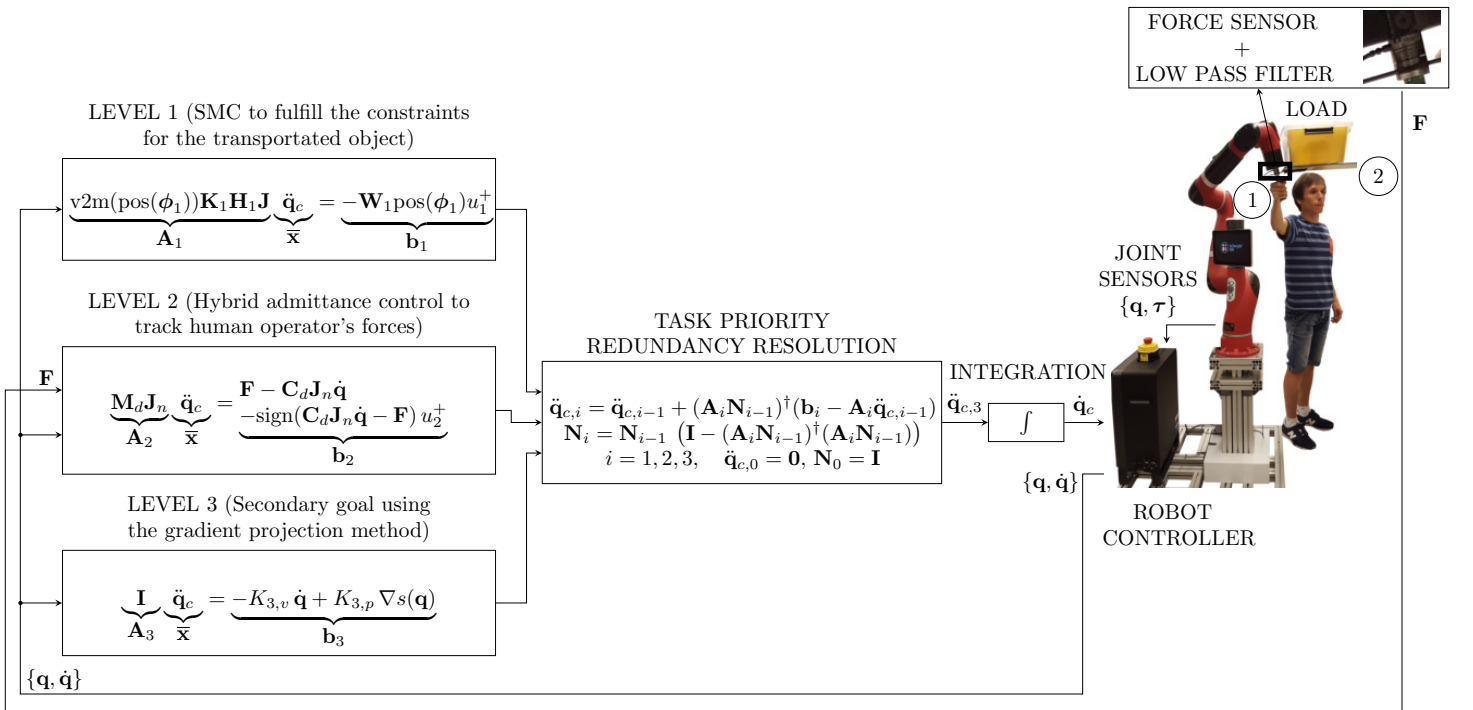
12

Fig. 2. Block diagram of the method.

LEVEL 1 (SMC to fulfill the constraints for the transported object)

$$\underbrace{\text{v2m}(\text{pos}(\boldsymbol{\phi}_1))\mathbf{K}_1\mathbf{H}_1\mathbf{J}}_{\mathbf{A}_1}\underbrace{\ddot{\mathbf{q}}_c}_{\overline{\mathbf{x}}} = \underbrace{-\mathbf{W}_1\text{pos}(\boldsymbol{\phi}_1)u_1^+}_{\mathbf{b}_1}$$

LEVEL 2 (Hybrid admittance control to track human operator's forces)

$$\underbrace{\mathbf{M}_d\mathbf{J}_n}_{\mathbf{A}_2}\underbrace{\ddot{\mathbf{q}}_c}_{\overline{\mathbf{x}}} = \underbrace{\begin{array}{c}\mathbf{F}-\mathbf{C}_d\mathbf{J}_n\dot{\mathbf{q}}\\-\text{sign}(\mathbf{C}_d\mathbf{J}_n\dot{\mathbf{q}}-\mathbf{F})\,u_2^+\end{array}}_{\mathbf{b}_2}$$

LEVEL 3 (Secondary goal using the gradient projection method)

$$\underbrace{\mathbf{I}}_{\mathbf{A}_3}\underbrace{\ddot{\mathbf{q}}_c}_{\overline{\mathbf{x}}} = \underbrace{-K_{3,v}\,\dot{\mathbf{q}}+K_{3,p}\,\nabla s(\mathbf{q})}_{\mathbf{b}_3}$$

TASK PRIORITY REDUNDANCY RESOLUTION

$$\ddot{\mathbf{q}}_{c,i} = \ddot{\mathbf{q}}_{c,i-1}+(\mathbf{A}_i\mathbf{N}_{i-1})^{\dagger}(\mathbf{b}_i-\mathbf{A}_i\ddot{\mathbf{q}}_{c,i-1})$$
$$\mathbf{N}_i = \mathbf{N}_{i-1}\left(\mathbf{I}-(\mathbf{A}_i\mathbf{N}_{i-1})^{\dagger}(\mathbf{A}_i\mathbf{N}_{i-1})\right)$$
$$i = 1,2,3,\quad \ddot{\mathbf{q}}_{c,0}=\mathbf{0},\,\mathbf{N}_0=\mathbf{I}$$

INTEGRATION $\quad \ddot{\mathbf{q}}_{c,3}$

$\int$ $\quad \dot{\mathbf{q}}_c$

$\mathbf{F}$

$\{\mathbf{q},\dot{\mathbf{q}}\}$

FORCE SENSOR + LOW PASS FILTER

LOAD

$\mathbf{F}$

JOINT SENSORS $\{\mathbf{q},\boldsymbol{\tau}\}$

$\{\mathbf{q},\dot{\mathbf{q}}\}$

ROBOT CONTROLLER

13

4.2. Level 1: SMC to fulfill the constraints for the transported object

4.2.1. Lie derivatives

In order to use the theory in Section 3, a dynamical system in the form of Eq. (9) is considered with the state vector $\mathbf{x} = \begin{bmatrix} \mathbf{q}^{\mathrm{T}} & \dot{\mathbf{q}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, the disturbance vector $\mathbf{d} = \mathbf{d}_c$ and the input vector $\mathbf{u} = \ddot{\mathbf{q}}_c$. Hence, the model is a double integrator, and from (5) the state equation results in:

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{O} & \mathbf{I} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{0} \\ \mathbf{d}_c \end{bmatrix} + \begin{bmatrix} \mathbf{O} \\ \mathbf{I} \end{bmatrix} \mathbf{u}, \tag{21}$$

and, therefore, the Lie derivatives for the constraint function ϕ_i are given by:

$$\mathbf{L_g}\phi_i = \nabla\phi_i^{\mathrm{T}} \mathbf{g} = (\partial\phi_i/\partial\dot{\mathbf{q}})^{\mathrm{T}} \tag{22}$$

$$L_f\phi_i = \nabla\phi_i^{\mathrm{T}} \mathbf{f} = (\partial\phi_i/\partial\mathbf{q})^{\mathrm{T}} \dot{\mathbf{q}} + (\partial\phi_i/\partial\dot{\mathbf{q}})^{\mathrm{T}} \mathbf{d}_c. \tag{23}$$

4.2.2. Modified constraints

Approaching the constraints at high speed is not advisable because, in general, large joint accelerations $\ddot{\mathbf{q}}$ would be required to slow down the robot motion in order to keep it on the constraint boundary. Therefore, the actual constraints σ_i will be modified to include the speed of movement as follows:

$$\phi_i = \sigma_i + K_i\dot{\sigma}_i = \sigma_i + K_i\,(\partial\sigma_i/\partial\mathbf{q})^{\mathrm{T}} \dot{\mathbf{q}} \le 0, \tag{24}$$

where K_i is a free design parameter that determines the rate of approach to the boundary of the original inequality constraint.

4.2.3. Constraints for the transported object

A typical constraint imposed by the transport task is to keep a reference orientation for the object when the robot and the human operator cooperatively transport it. To accomplish this, the following three inequality constraints are defined:

$$\sigma_{1,\alpha}(\mathbf{p}) = |\alpha - \alpha_{ref}| - \alpha_{max} \le 0 \tag{25}$$

$$\sigma_{1,\beta}(\mathbf{p}) = |\beta - \beta_{ref}| - \beta_{max} \le 0 \tag{26}$$

$$\sigma_{1,\gamma}(\mathbf{p}) = |\gamma - \gamma_{ref}| - \gamma_{max} \le 0, \tag{27}$$

where $\{\alpha, \beta, \gamma\}$ and $\{\alpha_{ref}, \beta_{ref}, \gamma_{ref}\}$ are the actual and reference values, respectively, for the roll-pitch-yaw orientation angles and $\{\alpha_{max}, \beta_{max}, \gamma_{max}\}$

14

are the maximum absolute deviation allowed for these angles.

4.2.4. Acceleration equality for Level 1

Taking into account (22) and (24)–(27), the Lie derivative $\mathbf{L_g}\phi_1$ for the first level, which is required for the SMC in (11), is given by:

$$\mathbf{L_g}\phi_1 = (\partial\phi_1/\partial\dot{\mathbf{q}})^{\mathrm{T}} = \mathbf{K}_1 \, (\partial\boldsymbol{\sigma}_1/\partial\mathbf{q})^{\mathrm{T}} = \mathbf{K}_1\mathbf{H}_1\mathbf{J}, \qquad (28)$$

where $\boldsymbol{\sigma}_1$ and ϕ_1 are column vectors composed of the three inequality constraints above, $\mathbf{K}_1$ is a diagonal matrix composed of all approaching parameters $K_{1,i}$ for the first level and:

$$\mathbf{H}_1 = \begin{bmatrix} 0 & 0 & 0 & \mathrm{sign}(\alpha - \alpha_{ref}) & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathrm{sign}(\beta - \beta_{ref}) & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathrm{sign}(\gamma - \gamma_{ref}) \end{bmatrix}, \qquad (29)$$

where $\mathrm{sign}(\cdot)$ represents the *sign function*.

Substituting (28) in (11) yields the following acceleration equality:

$$\mathrm{v2dm}\left(\mathrm{pos}\left(\phi_1\right)\right) \mathbf{K}_1 \, \mathbf{H}_1 \, \mathbf{J} \, \ddot{\mathbf{q}}_c = -\mathbf{W}_1 \, \mathrm{pos}\left(\phi_1\right) \, u_1^+ \;\; \rightarrow \;\; \mathbf{A}_1\ddot{\mathbf{q}}_c = \mathbf{b}_1, \qquad (30)$$

where $\mathbf{W}_1$ and u_1^+ are the switching gain weight matrix and switching gain, respectively, for the first level and $\mathbf{A}_1$ and $\mathbf{b}_1$ are the matrix and vector for the first task in (6).

4.3. Level 2: Hybrid admittance control to track human operator's forces

Similar to other works in the domain of human-robot interaction [7, 34], admittance control is considered to track the human operator's forces, but an additional switching term will be introduce to avoid calculating the derivative of the Jacobian. Traditional admittance control for human-robot interaction is described by:

$$\mathbf{M}_d \, \dot{\mathbf{v}}_n + \mathbf{C}_d \, \mathbf{v}_n = \mathbf{F}, \qquad (31)$$

where $\mathbf{v}_n = \mathbf{J}_n\dot{\mathbf{q}}$ is the vector of linear and angular velocities of the robot end-effector relative to the tool coordinate system, $\mathbf{F}$ is the external force vector (i.e., the forces exerted by the operator) relative to the tool coordinate system and the controller gains $\mathbf{M}_d$ and $\mathbf{C}_d$ are 6×6 diagonal matrices representing the virtual inertia and virtual damping, respectively. Note that

15

this admittance control omits the virtual stiffness since "restoring" forces are not desirable during the human-robot collaboration [7, 34].

Replacing the relation $\mathbf{v}_n = \mathbf{J}_n \dot{\mathbf{q}}$ in (31) and solving for $\ddot{\mathbf{q}}$ yields:

$$\mathbf{M}_d \, \mathbf{J}_n \, \ddot{\mathbf{q}} = \mathbf{F} - \mathbf{C}_d \, \mathbf{J}_n \, \dot{\mathbf{q}} - \mathbf{M}_d \, \dot{\mathbf{J}}_n \, \dot{\mathbf{q}}. \tag{32}$$

In order to avoid computing the derivative of the Jacobian $\mathbf{J}_n$ in (32), the following hybrid control law is proposed for the acceleration equality in Level 2:

$$\mathbf{M}_d \, \mathbf{J}_n \, \ddot{\mathbf{q}}_c = \mathbf{F} - \mathbf{C}_d \, \mathbf{J}_n \, \dot{\mathbf{q}} - \mathrm{sign}(\mathbf{C}_d \, \mathbf{J}_n \, \dot{\mathbf{q}} - \mathbf{F}) \, u_2^+ \quad \rightarrow \quad \mathbf{A}_2 \ddot{\mathbf{q}}_c = \mathbf{b}_2, \tag{33}$$

where $\mathbf{A}_2$ and $\mathbf{b}_2$ are the matrix and vector for the second task in (6), and the last switching term is used to compensate for the term $\mathbf{M}_d \, \dot{\mathbf{J}} \, \dot{\mathbf{q}}$ when the steady state given by $\mathbf{C}_d \, \mathbf{v}_n = \mathbf{F}$ has been reached, see (31).

4.4. Level 3: Secondary goal using the gradient projection method

This level is considered only for the case of *redundant* robots (e.g., the one used in the experiments in Section 6) since otherwise there are no remaining degrees of freedom at this level. A secondary goal (e.g. assuming manipulator configurations as close as possible to some reference, or those further from obstacles or singularity, etc.) can be achieved using robot *self-motions*, where the robot configuration is modified but the robot pose does not change. For this purpose, the most common approach in the literature is the so-called *gradient projection* method (GPM) [35], which minimizes a configuration-dependent scalar, the performance index s, by means of its gradient vector $\nabla s(\mathbf{q})$, which can be considered as a virtual force that attempts to push the configuration of the robot away from a critical region. In particular, the following acceleration equality is considered:

$$\ddot{\mathbf{q}}_c = -K_{3,v} \, \dot{\mathbf{q}} + K_{3,p} \, \nabla s(\mathbf{q}) \quad \rightarrow \quad \mathbf{A}_3 \ddot{\mathbf{q}}_c = \mathbf{b}_3, \tag{34}$$

where $K_{3,v}^{-1}$ and $K_{3,p}$ are design parameters representing the time constant of the joint velocities and the gain of the mentioned virtual force, respectively, and $\mathbf{A}_3$ and $\mathbf{b}_3$ are the matrix and vector for the third task in (6).

Different options have been proposed in the literature for selecting performance index s. For instance, the weighted square distance to a *reference*

configuration $\mathbf{q}_{ref}$ is used in [35] as performance index:

$$s_{ref}(\mathbf{q}) = -\frac{1}{2}(\mathbf{q}_{ref} - \mathbf{q})^{\mathrm{T}}\mathbf{W}_s(\mathbf{q}_{ref} - \mathbf{q}) \tag{35}$$

$$\nabla s_{ref}(\mathbf{q}) = \mathbf{W}_s(\mathbf{q}_{ref} - \mathbf{q}), \tag{36}$$

where $\mathbf{W}_s$ is a diagonal weighting matrix. The reference configuration $\mathbf{q}_{ref}$ can be chosen for joint-limit avoidance, obstacle avoidance, etc.

Other performance indexes in the literature are defined to avoid singular configurations using: the robot *manipulability* [36]; the *condition number* of the Jacobian [37]; the *local condition index* (LCI) (i.e., the reciprocal of the condition number) of the Jacobian [38]; etc. Furthermore, a performance index based on *artificial potential fields* is considered in [39] to keep the robot away from the obstacles.

5. Additional remarks

5.1. Control action

In this work the joint accelerations are considered as the SM discontinuous control action, which yields two advantages: the joint velocities are continuous (smoother control) and it allows to reach smoothly the boundary of the inequality constraints. If the actual control action are the joint velocities, a pure integrator can be applied to the discontinuous control signal to compute the actual continuous control action. Similarly, if the actual control action are the joint positions, a double integrator can be applied between both signals.

It is worth mentioning that, if the robot can be commanded using joint torques, which is not the case of most industrial robots, the proposed approach can be adapted for robot dynamic control computing the joint torques from the commanded joint accelerations using the inverse dynamic model of the robot. However, from a practical point of view, it could be advisable to use the kinematic control to take advantage of the low-level joint control provided by the robot manufacturer which is typically designed (and fully optimized) tanking into account a more complete robot dynamic model, including frictions, dead-zones, etc.

5.2. Time derivatives

The proposed approach requires the derivatives of the roll-pitch-yaw orientation angles $\{\alpha, \beta, \gamma\}$ for the SMC in the Level 1. As in many other

applications, the simplest way to deal with this issue consists in using numerical differentiation, e.g., the well-known backward Euler approximation. However, some kind of filtering should be previously applied to the actual variable when non-negligible noise is present. It is important to remark that the low-pass filter used for noise reduction must not limit the bandwidth of the control law. That is, the bandwidth of the control law should not exceed the bandwidth of the low-pass filter. In particular, for the proposed SMC in the first level, the theoretical frequency of the control law signal is equal to $(2T_s)^{-1}$ Hertz and, hence, the filter attenuation at this frequency should be relatively small.

5.3. Advantages of the proposed method

The first advantage of the proposed multitask SMC is *complementarity*: one task is in charge of achieving the reference values required by the transport task, while another task is in charge of controlling the robot force in order to track the human operator's forces when both cooperatively transport the object.

Furthermore, the main advantages of the proposed non-conventional SMC to satisfy the inequality constraints in Level 1 are:

- *Smoothness*: firstly, the joint velocities are continuous since the SM control action are the joint accelerations; and, secondly, the constraints boundary is reached progressively depending on a free design parameter, i.e., the velocity perpendicular to the constraint manifold is progressively reduced to zero.

- *Robustness*: the SMC algorithm is robust against the Lie derivatives $L_f \phi_i$ since they are collinear [17] with the discontinuous control action. Therefore, it is not affected by the terms included in $L_f \phi_i$, such as: the inaccuracies $\mathbf{d}_c$ of the low-level control loop; the derivative of the robot Jacobian; etc.

- *Low computational cost*: the SMC algorithm in Level 1 only uses partial information of the system model, i.e., the Lie derivatives $L_f \phi_i$ are not needed (derivative of the robot Jacobian, inaccuracies of the low-level control loop, etc.), only the Lie derivatives $\mathbf{L_g} \phi_i$ are required. In particular, the non-conventional SMC in Level 1, given by (30), only requires the robot Jacobian and the roll-pitch-yaw orientation angles of the robot tool.

Similarly, the hybrid admittance control in Level 2, given by (33), only requires the robot Jacobian, joint speeds and force sensor measurement. Note that the conventional admittance control given by (32) requires the derivative of the robot Jacobian but, as before, this scenario is avoided by the hybrid control law (33) given the switching term.

Therefore, the proposed approach can be efficiently implemented in a few program lines, resulting in reduced computational needs (see the Appendix at the end of the manuscript for more details).

As per traditional SMC controllers, the main disadvantage of the proposed method is the *chattering* drawback, see Section 3. However, given sufficiently large sampling rates (as is the case in the experimental work shown in the next section), the issue of chattering can be often neglected.

6. Real experimentation

The setup used for the experiments consists of (see Fig. 3): a Sawyer collaborative robot; a force/torque sensor Nano25 attached to the robot end-effector; a tool that combines a joystick and a flat rectangular plastic tray of 190x95x3mm attached to the sensor; and a shot glass with liquid.

The controller is implemented in an external PC (Intel Core i5-3470 processor at 3.2GHz) using *Ubuntu 16.04* as O.S., *ROS Lunar* distribution, *Intera 5 SDK* from Rethink Robotics, and the *netft_rdt_driver* ROS package provided by ATI Industrial Automation. The Sawyer robot, force sensor and external PC are connected to a router and communicate via UDP protocol.

6.1. Experiment conditions

i) The force sensor signal is filtered using a low-pass filter with a cut-off frequency of 73 Hz, which is implemented in the sensor electronics.

ii) The control period T_s is set to 20 milliseconds.

iii) Parameters used for Level 1 (Section 4.2): the reference orientation for the flat rectangular object at the tool is perpendicular to the robot Z-axis, i.e., roll and pitch angles are constrained to $\alpha_{ref} = 180°$ and $\beta_{ref} = 0$, respectively, whereas the yaw angle γ remains unconstrained and can be guided by the human operator in Level 2; $\alpha_{max} = \beta_{max} = 1°$, $\mathbf{K}_1 = 0.05\,\mathbf{I}$, $u_1^+ = 0.005$ and $\mathbf{W}_1 = \mathbf{I}$.

Fig. 3. Experimental setup: 7R serial manipulator with a force sensor rigidly attached to the robot end-effector, a tool that combines a joystick and a flat rectangular plastic tray of 190x95x3mm attached to the sensor and a shot glass with liquid.

iv) Parameters used for Level 2 (Section 4.3): $\mathbf{M}_d =$ diag($\begin{bmatrix} 6 & 6 & 6 & 0.25 & 0.25 & 0.25 \end{bmatrix}$), $\mathbf{C}_d =$ diag($\begin{bmatrix} 25 & 25 & 25 & 1 & 1 & 1 \end{bmatrix}$) and $u_2^+ = 0.01$, where diag($\cdot$) gives a diagonal matrix from a vector.

v) Parameters used for Level 3 (Section 4.4):

- First experiment: $K_{3,v} = 5$, $K_{3,p} = 1$, performance index s_{ref} in (35) for "pushing" the robot to a reference configuration given by $\mathbf{q}_{ref} = \begin{bmatrix} 0.17° & -67.15° & -0.12° & 124.73° & 0.12° & 32.32° & 190.22° \end{bmatrix}^\mathrm{T}$ and weighting matrix $\mathbf{W}_s = \mathbf{I}$.

- Second experiment: $K_{3,v} = 5$, $K_{3,p} = 100$, performance index s_{lci} equal to the LCI (inverse of the condition number) of the robot Jacobian. The gradient ∇s_{lci} to be used in (34) is computed numerically using a similar procedure to that detailed in [40].
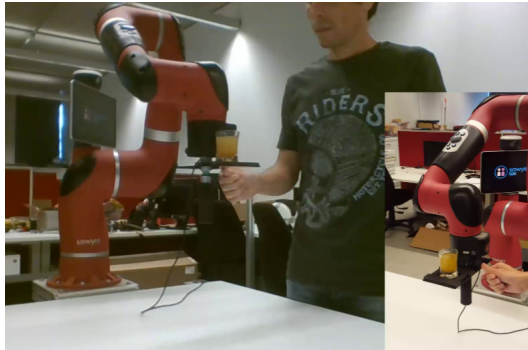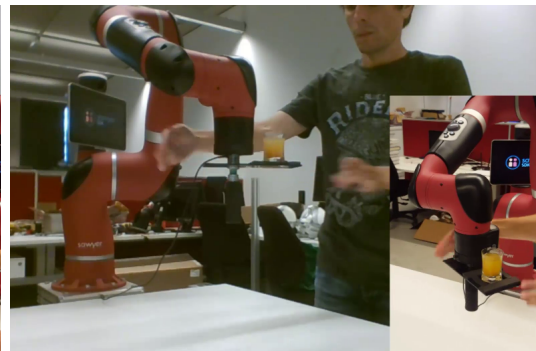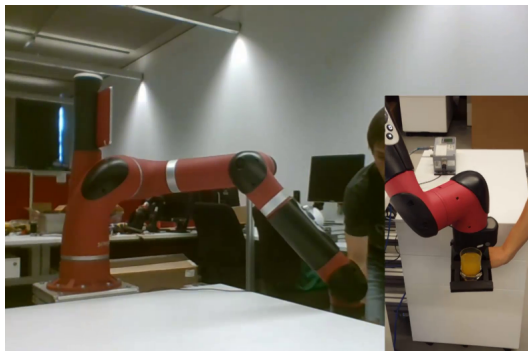
(a) video 0m34s, graph 22s   **(b)** video 0m35s, graph 23s
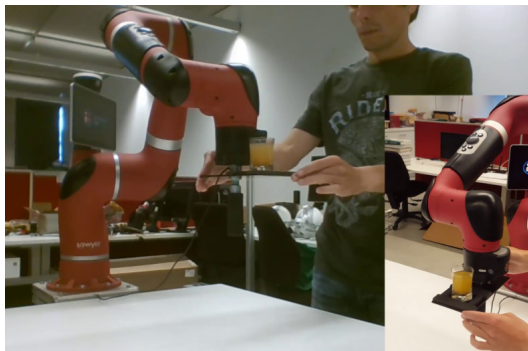
(c) video 1m18s, graph 66s   **(d)** video 1m20s, graph 68s

(e) video 1m32s, graph 80s   **(f)** video 2m02s, graph 110s

Fig. 4. Frames of the first experiment video. The time instant is indicated for each frame.
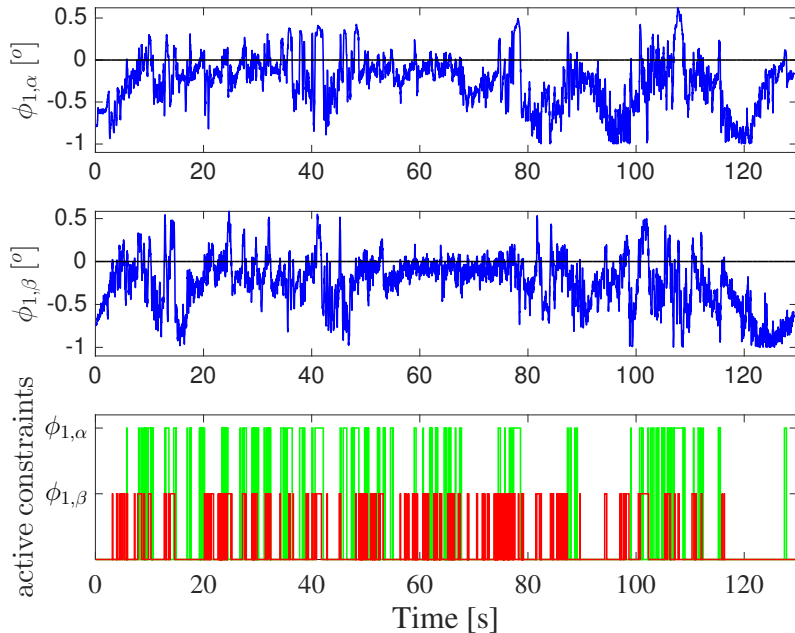
Fig. 5. First experiment: top, constraint function for the tool roll angle α; middle, constraint function for tool pitch angle β; bottom, constraints activation.

6.2. Results

In order to verify the performance of the proposed approach, a first experiment has been conducted with the shot glass with liquid placed on the flat object in order to show the smoothness of the proposed method. For this experiment it has been used for the secondary goal in Level 3 the performance index s_{ref} in (35). The video of the experiment can be played at https://media.upv.es/player/?id=72e36890-2b0f-11e8-b43a-51b816915a74. Fig. 4 shows several instants of the video where it can be seen how the user easily guides the flat rectangular object with the shot glass using one finger (Fig. 4a and Fig. 4b), rotates it (Fig. 4c and Fig. 4d) and takes it to the limits of the robot workspace (Fig. 4e and Fig. 4f).

Fig. 5 shows the performance of the first experiment in terms of constraint functions and the activation of the inequality constraints. In particular, the roll or pitch constraints in Level 1 are active around 30% of the experiment time, whereas the maximum deviation for these angles is just about 0.5 degrees, see Fig. 6, which means that the orientation for the flat object is almost perpendicular to the Z-axis all the time, as required. Note that the
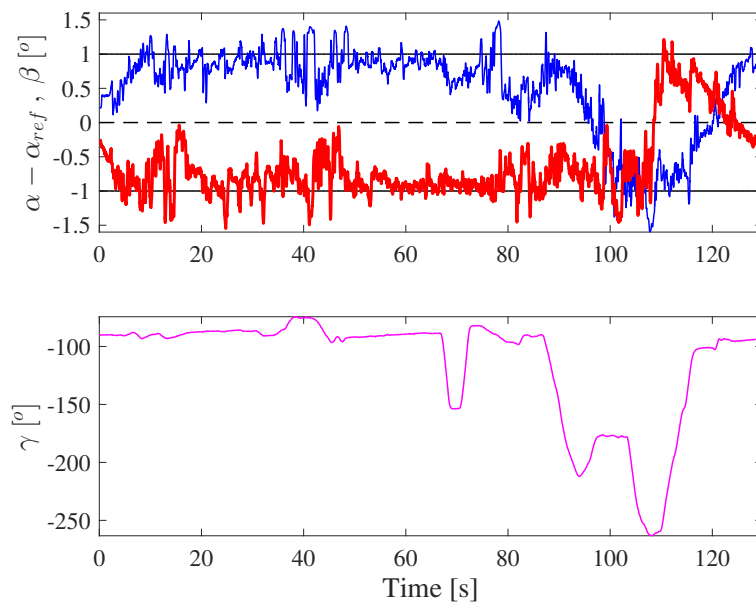
Fig. 6. First experiment: angles of the robot tool. Top: constrained roll α (thin-blue line) and pitch β (thick-red line) angles, where the continuous horizontal lines indicate the boundary of the allowed region, whereas the dashed horizontal line represents the reference value for the angles. Bottom: yaw angle γ, which is modified to track the operator's torques.

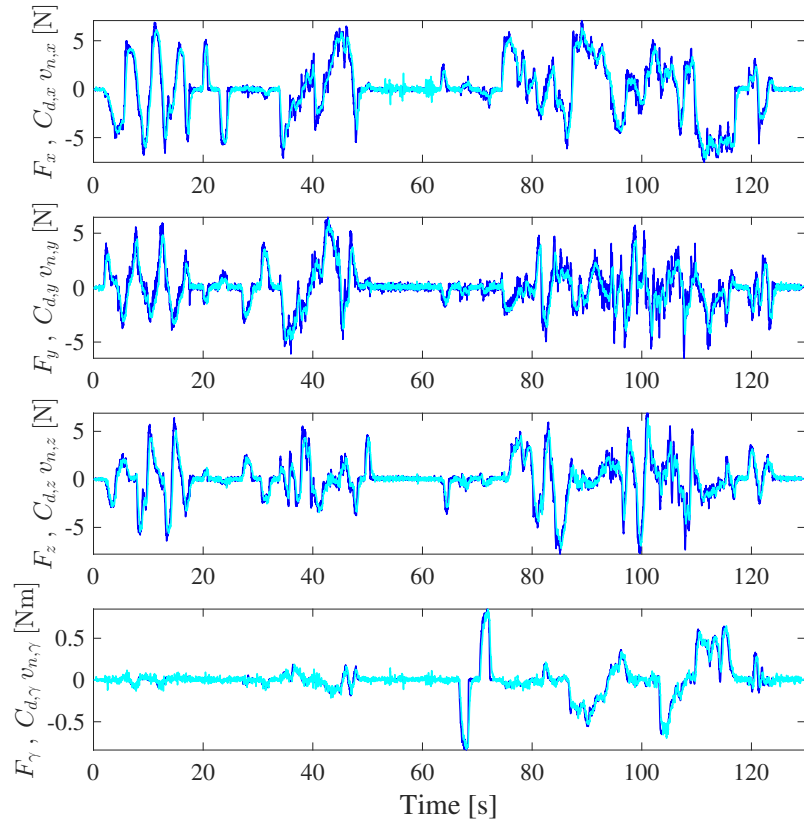**Fig. 7.** First experiment: human operator's forces and tool velocities multiplied by the virtual damping $\mathbf{C}_d$. From top to bottom: linear X-, Y- and Z-axes and angular Z-axis relative to the tool coordinate system. The force signals are in dark-blue, whereas the tool velocities are in light-cyan. Note that the tool velocities follow the force signals so closely that they can be hardly distinguished.
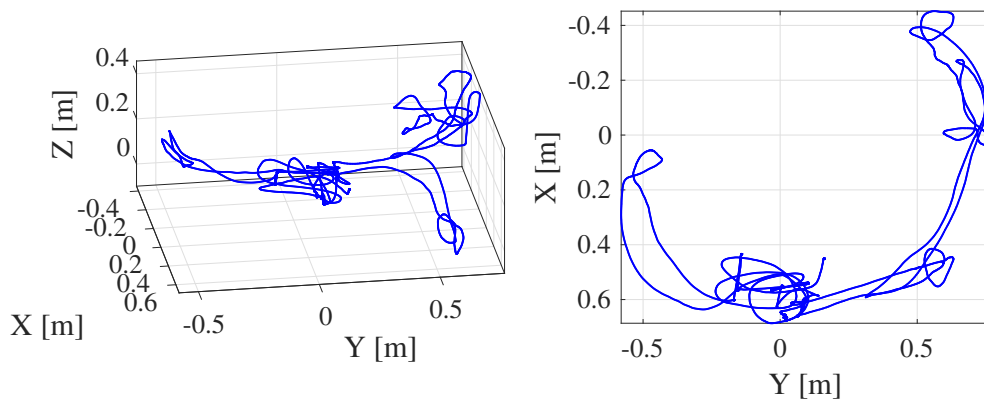
24

Fig. 8. First experiment: trajectory followed by the robot tool. Left: 3D view of the tool trajectory. Right: top view of the tool trajectory.

constraint for angle β is activated more than that for angle α, which is due to the torque in the pitch axis exerted by the weight and inertia of the shot glass with liquid.

Fig. 7 shows the human operator's forces, which are successfully tracked by the tool velocities: note that the steady state of the admittance control (31) in Level 2 corresponds to $\mathbf{C}_d\,\mathbf{v}_n = \mathbf{F}$. Fig. 8 shows the trajectory followed by the robot tool, where it can be seen that a large area has been covered by the robot during the experiment: around 1.3 meters in both X- and Y-axes and 0.6 meters in Z-axis.

Fig. 9 shows the control commands computed during the experiment. Note that all three levels contribute to the commanded joint accelerations, although those for Level 1 are smaller, since only small corrections are needed to keep the reference orientation of the flat object, as mentioned above. Fig. 9 also shows the commanded accelerations and velocities as a function of frequency, where it can be seen that the commanded velocities are relatively smooth since their high-frequency harmonics (from 2.5Hz to 25Hz) are negligible, see the bottom graph, which is not the case for the commanded accelerations, see the second last graph, due to the non-negligible contribution of the SMC in Level 1. In particular, the average value over the interval 2.5Hz-25Hz of the harmonics for $\ddot{\mathbf{q}}_c$ is 0.00791, whereas for $\dot{\mathbf{q}}_c$ is 0.00019, i.e., the magnitude for velocities is around 42 times lower than that of the accelerations.

A *second experiment* has been conducted in order to show the capability of the method to use another criterion for the secondary
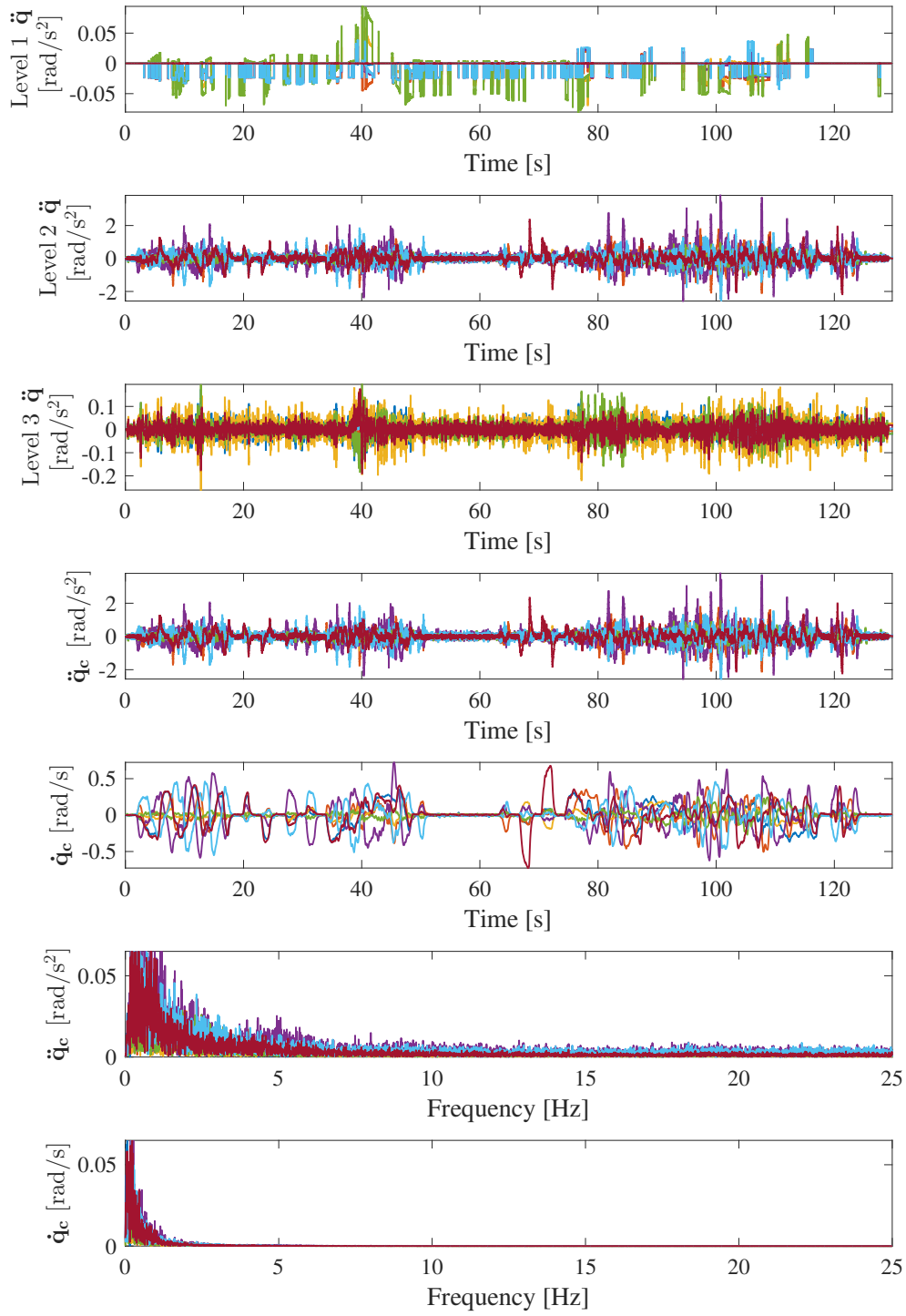
25

Fig. 9. First experiment: contribution of each priority level to the commanded joint accelerations in the first three graphs; commanded joint accelerations in the fourth graph and commanded joint velocities sent to the robot controller in the fifth graph; commanded joint accelerations and velocities as a function of frequency in the bottom two graphs.
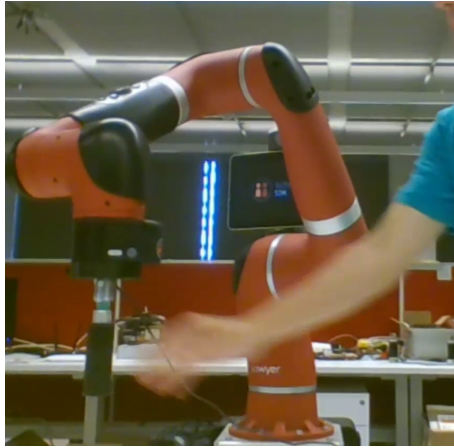
goal in Level 3, see Section 4.4. In particular, in order to avoid singularity it has been used as performance index s_{lci} the LCI (inverse of the condition number) of the robot Jacobian, which ranges from 0 (singularity) to 1. The video of the experiment can be played at https://media.upv.es/player/?id=ea468510-4933-11e8-bf15-17cde44d787f. Fig. 10 shows several instants of the video where it can be seen that, when the user has just changed the tool position (Fig. 10a, Fig. 10c and Fig. 10e), the robot reconfigures itself (Fig. 10b, Fig. 10d and Fig. 10f) to achieve a configuration with better LCI. This behavior can be clearly observed in Fig. 11: when the user stops guiding the tool, i.e., force vector **F** reaches the zero value in the bottom graph, marked with vertical dashed-green lines, the robot improves the performance index (thick-red segments in the top graph) using self-motion, i.e., changing the joint positions but not modifying the tool pose, see the second and third graphs.

A main advantage of SMC techniques over classical continuous approaches is its inherent robustness, see Section 5.3. In this regard the above experiments have revealed how the robot is able to automatically adapt the tool position to the operator's forces while maintaining the required tool orientation regardless of the possible inaccuracies accounted for at the low-level joint controllers and, in general, all the reasonable perturbations and unmodeled dynamics (non-linearities, friction forces, etc.) that could be expected in a practical setting for the collaborative task at hand, as shown by the results presented here (please also refer to the video links included).

7. Conclusions

An approach for human-robot collaboration in transportation applications has been presented using multi-task, non-conventional sliding mode control, admittance control and the gradient projection method. The proposed method represents a hybrid position/force scheme since some coordinates of the robot pose were controlled via a position feedback loop, while other coordinates were controlled via a force feedback loop.

Three tasks with different priority levels were defined in the controller to cooperatively perform the safe transportation of an object with a human operator. The first high-priority task was developed using non-conventional sliding mode control to guarantee safe reference parameters imposed by the task, e.g., keeping a load at a desired orientation (to prevent spill out in the case of liquids, or to reduce undue stresses that may compromise frag-

goal in Level 3, see Section 4.4. In particular, in order to avoid singularity it has been used as performance index s_{lci} the LCI (inverse of the condition number) of the robot Jacobian, which ranges from 0 (singularity) to 1. The video of the experiment can be played at https://media.upv.es/player/?id=ea468510-4933-11e8-bf15-17cde44d787f. Fig. 10 shows several instants of the video where it can be seen that, when the user has just changed the tool position (Fig. 10a, Fig. 10c and Fig. 10e), the robot reconfigures itself (Fig. 10b, Fig. 10d and Fig. 10f) to achieve a configuration with better LCI. This behavior can be clearly observed in Fig. 11: when the user stops guiding the tool, i.e., force vector **F** reaches the zero value in the bottom graph, marked with vertical dashed-green lines, the robot improves the performance index (thick-red segments in the top graph) using self-motion, i.e., changing the joint positions but not modifying the tool pose, see the second and third graphs.

A main advantage of SMC techniques over classical continuous approaches is its inherent robustness, see Section 5.3. In this regard the above experiments have revealed how the robot is able to automatically adapt the tool position to the operator's forces while maintaining the required tool orientation regardless of the possible inaccuracies accounted for at the low-level joint controllers and, in general, all the reasonable perturbations and unmodeled dynamics (non-linearities, friction forces, etc.) that could be expected in a practical setting for the collaborative task at hand, as shown by the results presented here (please also refer to the video links included).

7. Conclusions

An approach for human-robot collaboration in transportation applications has been presented using multi-task, non-conventional sliding mode control, admittance control and the gradient projection method. The proposed method represents a hybrid position/force scheme since some coordinates of the robot pose were controlled via a position feedback loop, while other coordinates were controlled via a force feedback loop.

Three tasks with different priority levels were defined in the controller to cooperatively perform the safe transportation of an object with a human operator. The first high-priority task was developed using non-conventional sliding mode control to guarantee safe reference parameters imposed by the task, e.g., keeping a load at a desired orientation (to prevent spill out in the case of liquids, or to reduce undue stresses that may compromise frag-
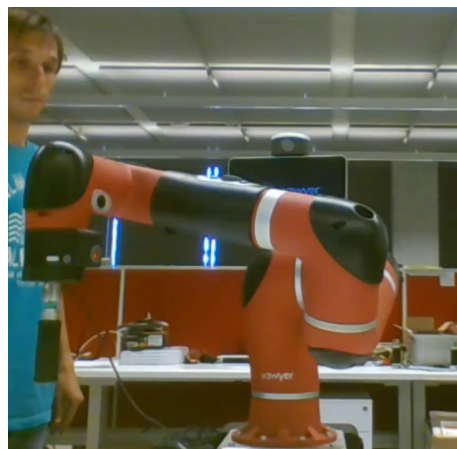
(a) video 15s, graph 3s
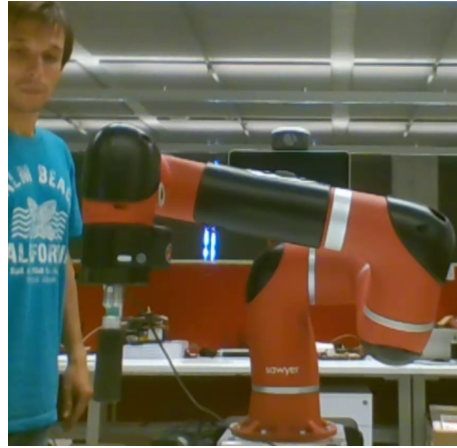
(b) video 42s, graph 30s

(c) video 48s, graph 36s

(d) video 1m11s, graph 59s

(e) video 1m16s, graph 64s

(f) video 1m30s, graph 78s

28

Fig. 10. Frames of the second experiment video. The time instant is indicated for each frame.
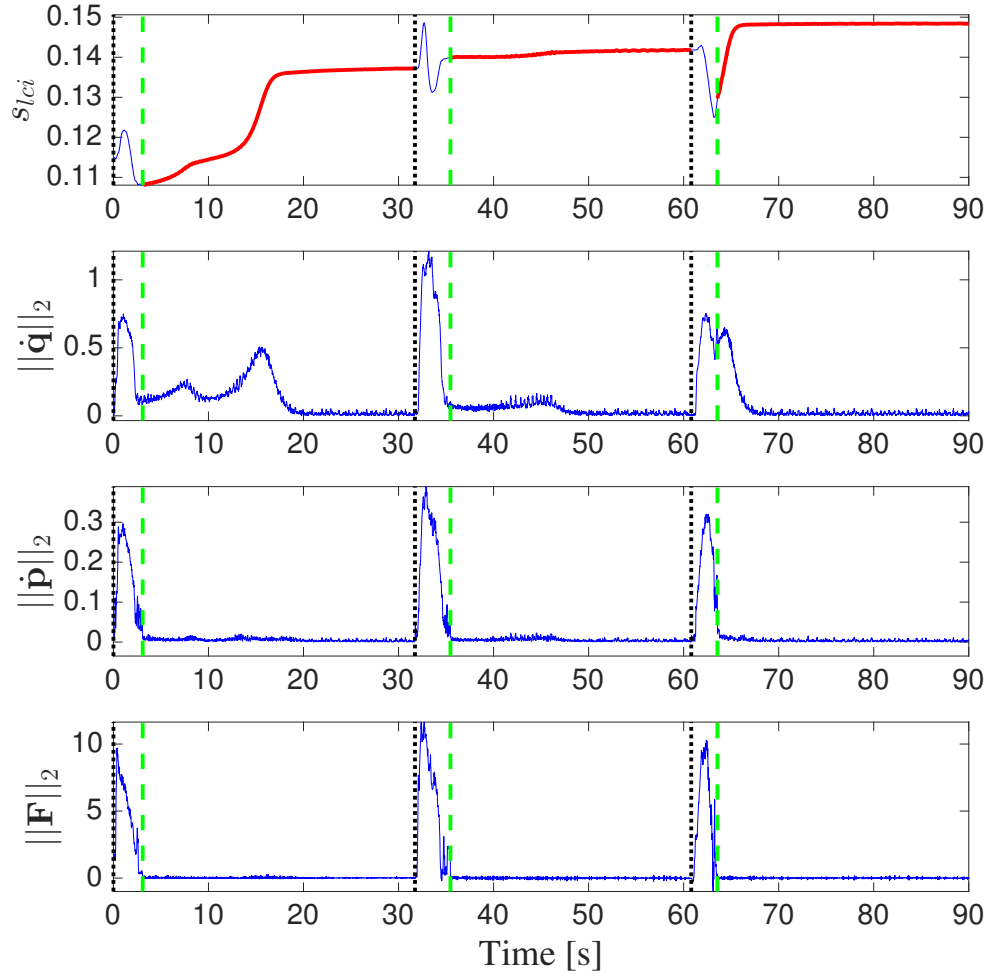
Fig. 11. Signals for the second experiment. Top graph: performance index s_{lci} for Level 3, where thin-blue line is used when the user is guiding the robot and thick-red line when no forces are detected. Second and third graphs: norm of the joint and tool velocities, respectively. Bottom graph: norm of the human operator's forces. The vertical lines indicate the time instants when the human operator starts (dotted-black) and stops (dashed-green) guiding the robot.

29

ile items). The second medium-priority task, which is based on a hybrid admittance control algorithm that includes a switching term, was used for the human operator to guide the robot by means of a force sensor located at the robot tool. Finally, the third low-priority task was considered to use the remaining degrees of freedom of redundant robots to achieve a pre-set secondary goal (e.g., singularity avoidance, remaining close to a homing configuration for increased safety, etc.) by means of the gradient projection method.

The main advantages of the proposed approach are robustness and low computational cost (refer to Appendix), while its main limitation, like in other sliding mode control applications, is the chattering drawback. The feasibility and effectiveness of the proposed approach was substantiated by experimental results using a redundant 7R manipulator: the Rethink Robotics Sawyer collaborative robot.

Although the motivation of this work was the cooperative transportation of an object, the proposed hybrid robot control can also be used to tackle other industrial applications where the human operator guides the robot under the constraint of some coordinates to reference values. For instance, in a drilling application, the robot can be guided by the operator to an a-priori unknown position in order to operate a drill, whilst the tool orientation is constrained to remain perpendicular to a (pre-set or sensed on-the-fly) drilling surface.

Appendix. Computer Implementation

The pseudo-code of the proposed method is shown below. The algorithm is executed at a sampling time of T_s seconds and uses the following auxiliary functions:

- Kinematic function $\mathbf{l}(\mathbf{q})$ and analytical $\mathbf{J}$ and geometric $\mathbf{J}_n$ Jacobians.

- Moore-Penrose pseudoinverse $(\cdot)^{\dagger}$, using a tolerance to set to zero the very small singular values, see Section 2.

- Robot sensors: $GetRobotStateAndForce()$, which returns the current robot state $\{\mathbf{q}, \dot{\mathbf{q}}\}$ and the force $\mathbf{F}$ detected by the sensor, which has already been filtered by the sensor electronics.

- Actuators: $SendToJointControllers(\dot{\mathbf{q}}_c)$, which sends the current commanded joint velocity vector to the joint controllers.

The computation time per iteration of the algorithm using compiled C code in a computer with Intel Core i5-3470 processor at 3.2 GHz clock frequency was around 15 microseconds for the experiment in Section 6.

Algorithm executed at sampling time of T_s seconds

1 $[\mathbf{q}, \dot{\mathbf{q}}, \mathbf{F}]$ =GetRobotStateAndForce();

2 $\mathbf{p} = \mathbf{l}(\mathbf{q})$;   // Eq. (1)

3 $\dot{\mathbf{p}} = \mathbf{J}\dot{\mathbf{q}}$;   // Eq. (2)

4 $\boldsymbol{\phi}_1 = \begin{bmatrix} |\alpha - \alpha_{ref}| - \alpha_{\max} + K_{1,\alpha}\,\mathrm{sign}(\alpha - \alpha_{ref})\,\dot{\alpha} \\ |\beta - \beta_{ref}| - \beta_{\max} + K_{1,\beta}\,\mathrm{sign}(\beta - \beta_{ref})\,\dot{\beta} \\ |\gamma - \gamma_{ref}| - \gamma_{\max} + K_{1,\gamma}\,\mathrm{sign}(\gamma - \gamma_{ref})\,\dot{\gamma} \end{bmatrix}$; // Eqs. (24),(25)-(27)

5 $\mathbf{A}_1 = \mathrm{v2dm}\,(\mathrm{pos}\,(\boldsymbol{\phi}_1))\,\mathbf{K}_1\mathbf{H}_1\mathbf{J}$;   // Eq. (30)

6 $\mathbf{b}_1 = -\mathbf{W}_1\,\mathrm{pos}\,(\boldsymbol{\phi}_1)\,u_1^+$;   // Eq. (30)

7 $\mathbf{A}_2 = \mathbf{M}_d\,\mathbf{J}_n$;   // Eq. (33)

8 $\mathbf{b}_2 = \mathbf{F} - \mathbf{C}_d\,\mathbf{J}_n\,\dot{\mathbf{q}} - \mathrm{sign}(\mathbf{C}_d\,\mathbf{J}_n\,\dot{\mathbf{q}} - \mathbf{F})\,u_2^+$;   // Eq. (33)

9 $\mathbf{A}_3 = \mathbf{I}$;   // Eq. (34)

10 $\mathbf{b}_3 = -K_{3,v}\,\dot{\mathbf{q}} + K_{3,p}\,\nabla s(\mathbf{q})$;   // Eq. (34)

11 $\ddot{\mathbf{q}}_{c,1} = \mathbf{A}_1^\dagger \mathbf{b}_1$;   // Eq. (7), $i = 1$

12 $\mathbf{N}_1 = \mathbf{I} - \mathbf{A}_1^\dagger \mathbf{A}_1$;   // Eq. (8), $i = 1$

13 $\ddot{\mathbf{q}}_{c,2} = \ddot{\mathbf{q}}_{c,1} + (\mathbf{A}_2\mathbf{N}_1)^\dagger(\mathbf{b}_2 - \mathbf{A}_2\ddot{\mathbf{q}}_{c,1})$;   // Eq. (7), $i = 2$

14 $\mathbf{N}_2 = \mathbf{N}_1(\mathbf{I} - (\mathbf{A}_2\mathbf{N}_1)^\dagger(\mathbf{A}_2\mathbf{N}_1))$;   // Eq. (8), $i = 2$

15 $\ddot{\mathbf{q}}_{c,3} = \ddot{\mathbf{q}}_{c,2} + (\mathbf{A}_3\mathbf{N}_2)^\dagger(\mathbf{b}_3 - \mathbf{A}_3\ddot{\mathbf{q}}_{c,2})$;   // Eq. (7), $i = 3$

16 $\dot{\mathbf{q}}_c = \dot{\mathbf{q}}_{c,prev} + T_s\,\ddot{\mathbf{q}}_{c,3}$;   // Integration

17 SendToJointControllers($\dot{\mathbf{q}}_c$);

18 $\dot{\mathbf{q}}_{c,prev} = \dot{\mathbf{q}}_c$;   // For next iteration

References

[1] R. Meziane, M. J.-D. Otis, H. Ezzaidi, Human-robot collaboration while sharing production activities in dynamic environment: Spader system, Robotics and Computer-Integrated Manufacturing 48 (2017) 243–253.

[2] I. Maurtua, A. Ibarguren, J. Kildal, L. Susperregi, B. Sierra, Human-robot collaboration in industrial applications, International Journal of Advanced Robotic Systems 14 (2017) 1729881417716010.

[3] Y. Ansari, M. Manti, E. Falotico, Y. Mollard, M. Cianchetti, C. Laschi, Towards the development of a soft manipulator as an assistive robot for personal care of elderly people, International Journal of Advanced Robotic Systems 14 (2017) 1729881416687132.

[4] L. Rozo, D. Bruno, S. Calinon, D. G. Caldwell, Learning optimal controllers in human-robot cooperative transportation tasks with position and force constraints, in: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1024–1030.

[5] P. Tsarouchi, A.-S. Matthaiakis, S. Makris, G. Chryssolouris, On a human-robot collaboration in an assembly cell, International Journal of Computer Integrated Manufacturing 30 (2017) 580–589.

[6] A. Mohammed, B. Schmidt, L. Wang, Active collision avoidance for human-robot collaboration driven by vision sensors, International Journal of Computer Integrated Manufacturing 30 (2017) 970–980.

[7] F. Dimeas, N. Aspragathos, Online stability in human-robot cooperation with admittance control, IEEE Trans. on Haptics 9 (2016) 267–278.

[8] S. Nikoleizig, A. Vick, J. Krüger, Compensating human feedback oscillation in compliance control for industrial robots, in: 2017 3rd Int. Conf. on Control, Automation and Robotics (ICCAR), pp. 221–224.

[9] A. M. Khan, D.-w. Yun, K. M. Zuhaib, J. Iqbal, R.-J. Yan, F. Khan, C. Han, Estimation of desired motion intention and compliance control for upper limb assist exoskeleton, International Journal of Control, Automation and Systems 15 (2017) 802–814.

[10] Z. Jiang, Y. Sun, L. Lei, Y. Hu, C. Xiao, J. Zhang, Compliance control based on particle swarm optimization approach for physical human-robot interaction, in: 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR), pp. 117–122.

[11] Y. Li, S. S. Ge, Force tracking control for motion synchronization in human-robot collaboration, Robotica 34 (2016) 1260–1281.

[12] P. N. Koustoumpardis, K. I. Chatzilygeroudis, A. I. Synodinos, N. A. Aspragathos, Human Robot Collaboration for Folding Fabrics Based on Force/RGB-D Feedback, Springer, Cham, pp. 235–243.

[13] J. Vogel, S. Haddadin, B. Jarosiewicz, J. Simeral, D. Bacher, L. Hochberg, J. Donoghue, P. van der Smagt, An assistive decision-and-control architecture for force-sensitive hand-arm systems driven by human-machine interfaces, The International Journal of Robotics Research 34 (2015) 763–780.

[14] L. Lu, J. T. Wen, Human-directed robot motion/force control for contact tasks in unstructured environments, in: 2015 IEEE Int. Conf. on Automation Science and Engineering (CASE), pp. 1165–1170.

[15] S. Jlassi, S. Tliba, Y. Chitour, An event-controlled online trajectory generator based on the human-robot interaction force processing, Industrial Robot: An International Journal 41 (2014) 15–25.

[16] C. Wang, Y. Li, S. S. Ge, K. P. Tee, T. H. Lee, Continuous critic learning for robot control in physical human-robot interaction, in: 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013), pp. 833–838.

[17] C. Edwards, S. Spurgeon, Sliding Mode Control: Theory and Applications, Taylor & Francis, UK, 1st edition, 1998, ch. 1, pp. 1-18.

[18] J. Zhou, Z. Zhou, Q. Ai, Impedance control of the rehabilitation robot based on sliding mode control, in: X. Li (Ed.), Mechanical Engineering and Control Systems (MECS2015), pp. 135–140.

[19] X. Jin, S. Zhu, X. Zhu, Q. Chen, X. Zhang, Single-input adaptive fuzzy sliding mode control of the lower extremity exoskeleton based on human-robot interaction, Advances in Mechanical Engineering 9 (2017) DOI: 10.1177/1687814016686665.

[20] N. Kashiri, N. G. Tsagarakis, M. Van Damme, B. Vanderborght, D. G. Caldwell, Proxy-Based Sliding Mode Control of Compliant Joint Manipulators, Springer International Publishing, Cham, pp. 241–257.

[21] D. Yun, A. M. Khan, R.-J. Yan, Y. Ji, H. Jang, J. Iqbal, K. M. Zuhaib, J. Y. Ahn, J. Han, C. Han, Handling subject arm uncertainties for upper

limb rehabilitation robot using robust sliding mode control, Int. J. of Precision Engineering and Manufacturing 17 (2016) 355–362.

[22] Q. Wu, X. Wang, F. Du, Q. Zhu, Fuzzy sliding mode control of an upper limb exoskeleton for robot-assisted rehabilitation, in: 2015 IEEE International Symposium on Medical Measurements and Applications (MeMeA) Proceedings, pp. 451–456.

[23] A. Mitra, L. Behera, Development of a fuzzy sliding mode controller with adaptive tuning technique for a mri guided robot in the human vasculature, in: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), pp. 370–377.

[24] J.-S. Oh, S.-H. Choi, S.-B. Choi, Control of repulsive force in a virtual environment using an electrorheological haptic master for a surgical robot application, Smart Materials and Structures 23 (2014) DOI: 10.1088/0964-1726/23/1/015010.

[25] N. Rahman, M. C. Lee, Reaction force separation method of surgical tool from unknown dynamics and disturbances by fuzzy logic and perturbation observer of smcspo algorithm, in: The SICE Annual Conference 2013, pp. 2536–2541.

[26] N. Rahman, M. Lee, Roboticslab based simulation for extraction of reaction force in multi dof surgical robot dynamic system and analysis, in: 2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), pp. 560–564.

[27] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics: Modelling, Planning and Control, Springer-Verlag, London, UK, 2009, ch. 3.1 and ch. 3.6, pp. 105-106 and 128-132.

[28] P. Corke, Robotics, Vision and Control: Fundamental Algorithms in MATLAB, Springer-Verlag, Berlin, Germany, 2nd edition, 2017, ch. 8.1.2 and ch. 8.1.3, pp. 232-233.

[29] S. Chiaverini, G. Oriolo, I. Walker, Kinematically redundant manipulators, Springer Handbook of Robotics (2008) pp. 245–268.

[30] Y. Nakamura, H. Hanafusa, T. Yoshikawa, Task-priority based redundancy control of robot manipulators, The Int. Journal of Robotics Research 6 (1987) 3–15.

[31] B. Siciliano, J. Slotine, A general framework for managing multiple tasks in highly redundant robotic systems, in: 1991 Proc. of the Fifth Int. Conf. on Advanced Robotics (ICAR'91), Pisa, Italy, pp. 1211–1216.

[32] G. Golub, C. Van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996, ch. 8.6, pp. 448-457.

[33] V. Utkin, J. Guldner, J. Shi, Sliding Mode Control in Electro-Mechanical Systems, Taylor & Francis, London, 2nd edition, 2009, ch. 1 and ch. 2, pp. 1-40.

[34] C. Landi, F. Ferraguti, L. Sabattini, C. Secchi, C. Fantuzzi, Admittance control parameter adaptation for physical human-robot interaction, in: 2017 IEEE International Conference on Robotics and Automation (ICRA'17), pp. 2911–2916.

[35] A. Liégeois, Automatic supervisory control of the configuration and behavior of multibody mechanisms, IEEE Trans. on Systems, Man and Cybernetics 7 (1977) 868–871.

[36] G. Marani, J. Kim, J. Yuhl, W. K. Chung, A real-time approach for singularity avoidance in resolved motion rate control of robotic manipulators, in: 2002 Proceedings of the IEEE Int. Conf. on Robotics and Automation, volume 2, Washington, DC, USA, pp. 1973–1978.

[37] L. Huo, L. Baron, The joint-limits and singularity avoidance in robotic welding, Industrial Robot: An Int. Journal 35 (2008) 456–464.

[38] S. Patel, T. Sobh, Manipulator performance measures-a comprehensive literature survey, J. of Intelligent & Robotic Systems 77 (2015) 547–570.

[39] J. Juang, Collision avoidance using potential fields, Industrial Robot 25 (1998) 408–415.

[40] F. Dimeas, V. Moulianitis, N. Aspragathos, Manipulator performance constraints in human-robot cooperation, Robotics and Computer-Integrated Manufacturing 50 (2018) 222–233.