Hierarchical Task-Priority Control for Human-Robot Co-Manipulation

Jonathan Cacace, Fabio Ruggiero, and Vincenzo Lippiello

PRISMA Lab, Department of Electrical Engineering and Information Technology,
University of Naples Federico II, Via Claudio 21, 80125, Naples, Italy,
{jonathan.cacace,fabio.ruggiero,vincenzo.lippiello}@unina.it

Abstract. The extensive distribution of collaborative robots in industrial workplaces allows human operators to decrease the weight and the repetitiveness of their activities. In order to facilitate the role of the human worker during the interaction with these robots, innovative control paradigms, enabling an intuitive human-robot collaborative manipulation, are needed. In this work, a dynamic and hierarchical task-priority control framework is proposed, leveraging a physical interaction with a redundant robot manipulator through a force sensor. The foremost objective of this approach is to exploit the non-trivial null-space of the redundant robot to increase the performance of the co-manipulation and, consequently, its effectiveness. A comparison between the proposed methodology and a standard admittance control scheme is carried out within an industrial use case study consisting of a human operator interacting with a KUKA LBR iiwa arm.

Keywords: Task-priority Control, Human-Robot Interaction, Cobots.

1 Introduction

Collaborative human-robot manipulation is an emerging field pointing at developing innovative service robotic applications in which humans and robots cooperate for the execution of tasks sharing their workspace, as highlighted by Corrales et al (2012). Thanks to the extensive distribution of collaborative robot, also known as *cobots*, a robotic co-worker can assist human operators during the execution of service tasks, like cooperative lifting and transportation of heavy objects, collaborative assembly, and similar. The assistance can be carried out both in an active way, as underlined by Caccavale et al (2016), and in a proactive way, as explained by Zanchettin and Rocco (2017). In this scenario, the human worker typically guides the motion of the robot by physically interacting with its end-effector, whose reaction is compliant to the forces exerted by the human.

In order to operate in safe and adequate conditions, the robot motion should be controlled by taking into account the performance of the manipulator and avoiding instability regions, kinematic singularities, and joint mechanical limitations. If one of these situations occurs, the robot loses the ability to move since the operator forces cannot be translated into suitable joint commands.

In this work, the kinematic redundancy of the manipulator is exploited to design a set of prioritized tasks. These tasks involve the forces exerted by the operator to move the robot accordingly. Such forces are measured through a force sensor mounted at the end-effector. Besides, the tasks also involve several indices related to the manipulator performance such as the robot manipulability, the joint limits, and other quality factors as described by Chiaverini (1997) and Baerlocher and Boulic (1998). While the human operator moves the manipulator end-effector through the force sensor, the proposed formulation maximizes the performance of the robot during such cooperative manipulation. Since the input issued by the human operator is not predictable, the robot motion cannot be optimally planned in advance. In this context, several factors, such as the operator inexperience, fatigue, or stress, may bring the manipulator in a low-performance configuration. As a result, the robot can get stuck or, in worst cases, it becomes unstable and dangerous for a tight interaction with the human. Therefore, robot motion must be adapted at runtime. The task priority formulation can exploit the degrees of freedom (DoFs) needed to accomplish the primary task (e.g., follow the movements of the human operator given through the force sensor) while relying on the non-trivial null-space of the redundant robot, and thus on the remaining DoFs, to satisfy secondary tasks. Besides, in the proposed formulation, dynamic switching of the task priority is proposed to modify the stack structure (e.g., change primary goal and, consequently, the null-space composition of tasks).

The proposed framework is validated through a human-robot interactive task where a human operator cooperates with a KUKA LBR iiwa robot on whose end-effector is mounted a force sensor, as shown in Fig. 1. The effectiveness of the system is also evaluated by comparing the proposed methodology with a standard admittance controller.

The outline of the paper is organized as follows. A brief overview of the related works is presented in Section 2, in which the proposed contribution is pointed out. The hierarchical task priority framework used to implement the null-space control for co-manipulation is described in Section 3. The employed tasks are introduced in Sections 4, while in Section 5 their composition is discussed. Finally, in Section 6, an experimental case study to test the effectiveness of the approach is carried out. Section 7 concludes the paper through a brief discussion of the overall work and future research directions.

2 Related Works

Within human-robot co-manipulation systems, human operators can usually physically interact with a robot by applying forces on its end-effector directly. In these applications, the operator forces are translated into robot motion commands imposing a desired dynamic behavior between the external forces and the robot motion: this is possible thanks to an admittance control scheme, for instance. Several works addressed the development of intuitive interaction strate-

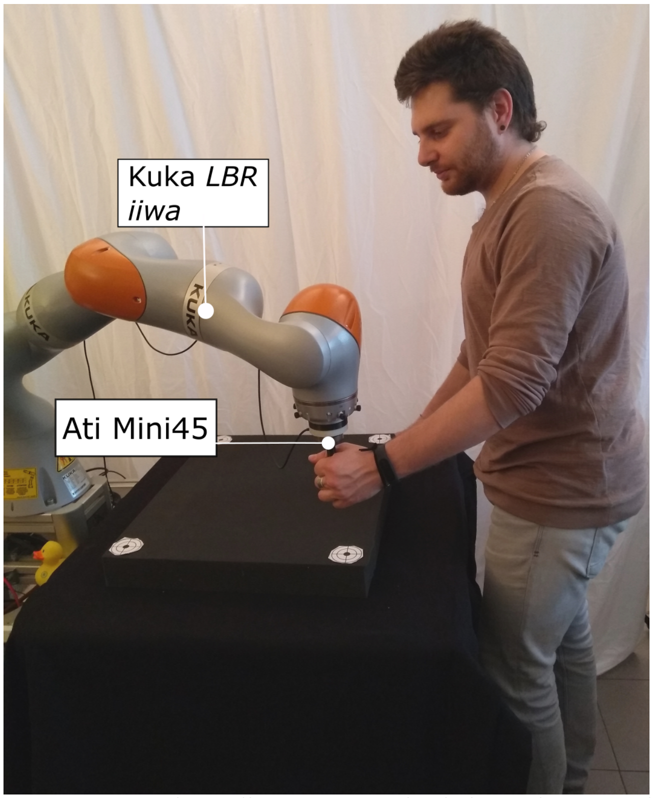


Fig. 1: The human operator physically interacts with a lightweight manipulator through the *ATI Mini45* force sensor.

gies to improve the performance of the co-manipulation starting from the fundamental works of Newman (1992) and Gullapalli et al (1992).

Different indices are proposed in the literature to evaluate the performance of the manipulator in a given configuration. Yoshikawa (1985) introduced a manipulability index, while Dubey and Luh (1988) introduced the end-effector velocity ratio (MVR) index. Landi et al (2017) presented a parameter to calculate the deviations of the manipulator from the nominal behavior, and therefore identify control vulnerabilities.

Several works addressed the problem of increasing manipulator performance during human-robot co-manipulation tasks. The following approaches can be distinguished. Within variable admittance approaches, the gains of the admittance controller are regulated on-line to provide an intuitive interaction between the robotic arm and the operator. In particular, Labrecque and Gosselin (2018) regulated the controller gains to accommodate and optimize the system response to any potential dynamics, while Landi et al (2017) adjusted the damping of the admittance controller to avoid undesired and unstable robot configurations. Dimeas et al (2018) proposed instead a haptic feedback approach to prevent the operator from bringing the manipulator in a low-performance region of the workspace. Finally, similarly to the proposed approach, the redundancy of the manipulator, and thus its non-trivial null-space, was exploited by Ficuciello et al (2014) to ensure a decoupled apparent inertia at the end effector. Differently from Ficuciello et al (2014), the proposed technique defines a stack of tasks to optimize different quality indices, whose priority can be modified based on the current behavior of the co-manipulation task.

Task priority control of redundant systems is anyway extensively explored in the literature. Applications span from the fundamental work by Nakamura et al (1987) to more recent utilization in fine object manipulation, aerial and underwater scenarios by Caccavale et al (2013), Lippiello et al (2016), and Casalino et al (2012), respectively. Task composition was exploited by Rocchi et al (2015) and Mansard et al (2009) to control collaborative working of humanoid robots, while a framework providing a flexible and composable way to define robot control tasks was proposed by Aertbeliën and De Schutter (2014). Finally, in the context of collaborative applications and similarly to the proposed approach, Agravante et al (2014) designed a task composition methodology coupling a visual servoing and an impedance controller. Differently from what proposed in this work, the compliant behaviour does not seem introduced to accommodate the human control input, but to achieve the desired collaborative task. Therefore, at the best of authors' knowledge, no application in which the human operator physically leads the motion of the robot can be found with hierarchical task-priority control. Hence, the novelty introduced by this work is indeed the application of the renowned null-space task priority formulation within a cobot system.

3 Dynamic Task Priority Framework

In this section, the dynamic task priority framework used to implement the co-manipulation system is explained. A similar formulation was already introduced by Buonocore et al (2015) and Lippiello et al (2016) in different application domains.

Let n be the number of DoFs of the robot manipulator. To exploit kinematic redundancy of the system, the dimension μ_0 of the main task for the robot end-effector must be less than n. In this way, it is possible to achieve some secondary tasks in terms of priority, while preserving the principal task, exploiting the extra DoFs. Hence, in a redundant manipulator, the same position of the end-effector can be reached through different joint configurations, letting the robotic arm to be suitably reconfigured by using internal motions.

Let $\boldsymbol{q} = \left[q_1 \dots q_n\right]^\top \in \mathbb{R}^n$ be the arm joint vector describing the arm configuration, where q_i is the i-th joint variable, with $i = 1, \dots, n$. The goal of the kinematic control framework is to accomplish the primary and, eventually, the secondary tasks generating the control velocity vector $\dot{\boldsymbol{q}} \in \mathbb{R}^n$ (i.e., the velocity command of the robot joints).

Let $\boldsymbol{\sigma}_0 := \boldsymbol{f}_0(\boldsymbol{q}) \in \mathbb{R}^{\mu_0}$ be the main task. The following differential relationship holds:

$$\dot{\boldsymbol{\sigma}}_0 := \frac{\partial \boldsymbol{f}_0(\boldsymbol{q})}{\partial \boldsymbol{q}} \dot{\boldsymbol{q}} = \boldsymbol{J}_0(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{1}$$

where $\boldsymbol{J}_0(\boldsymbol{q}) \in \mathbb{R}^{\mu_0 \times n}$ represents the main task Jacobian. In order to fulfill the regulation problem of $\boldsymbol{\sigma}_0$ to the desired value $\boldsymbol{\sigma}_0^*$, let $\widetilde{\boldsymbol{\sigma}}_0 = \boldsymbol{\sigma}_0^* - \boldsymbol{\sigma}_0$ be the main task error. By inverting (1) the following velocity command can be considered

$$\dot{\boldsymbol{q}}^* = \boldsymbol{J}_0(\boldsymbol{q})^\dagger \boldsymbol{\Lambda}_0 \widetilde{\boldsymbol{\sigma}}_0, \tag{2}$$

where $\boldsymbol{\Lambda}_0 \in \mathbb{R}^{\mu_0 \times \mu_0}$ is a positive-definite gain matrix, and $\boldsymbol{J}_0(\boldsymbol{q})^\dagger$ is the generalized pseudo-inverse of $\boldsymbol{J}_0(\boldsymbol{q})$. The following error dynamics is achieved folding (2) into (1)

$$\dot{\widetilde{\boldsymbol{\sigma}}}_0 = -\boldsymbol{\Lambda}_0 \widetilde{\boldsymbol{\sigma}}_0, \tag{3}$$

obtaining an asymptotic convergence of the main task to its desired value.

Since $\mu_0 < n$, the null-space of $\boldsymbol{J}_0(\boldsymbol{q})$ is non-trivial. A further task can be thus defined as $\boldsymbol{\sigma}_1 := \boldsymbol{f}_1(\boldsymbol{q}) \in \mathbb{R}^{\mu_1}$, with $\mu_1 \leq n$, from which the following joint velocity command can be designed

$$\dot{\boldsymbol{q}}^* = \boldsymbol{J}_0(\boldsymbol{q})^\dagger \boldsymbol{\Lambda}_0 \widetilde{\boldsymbol{\sigma}}_0 + \boldsymbol{N}_0(\boldsymbol{q}) \boldsymbol{J}_1(\boldsymbol{q})^\dagger \boldsymbol{\Lambda}_1 \tilde{\boldsymbol{\sigma}}_1, \tag{4}$$

where $\boldsymbol{J}_1(\boldsymbol{q}) \in \mathbb{R}^{\mu_1 \times n}$ is the Jacobian matrix of the defined task and it is built similarly to (1), $\tilde{\boldsymbol{\sigma}}_1$ is the error with respect to a desired value $\boldsymbol{\sigma}_1^*$ for the task, $\boldsymbol{\Lambda}_1 \in \mathbb{R}^{\mu_1 \times \mu_1}$ is a positive-definite gain matrix, and $\boldsymbol{N}_0(\boldsymbol{q}) = \boldsymbol{I}_n - \boldsymbol{J}_0(\boldsymbol{q})^\dagger \boldsymbol{J}(\boldsymbol{q})$ is the projector into the null-space of $\boldsymbol{J}_0(\boldsymbol{q})$, with $\boldsymbol{I}_n \in \mathbb{R}^{n \times n}$ the identity matrix.

Adding further secondary tasks, Antonelli (2009) generalizes the framework to the case of η prioritized tasks with the following general velocity command

$$\dot{\boldsymbol{q}}^* = \boldsymbol{J}_0(\boldsymbol{q})^\dagger \boldsymbol{\Lambda}_0 \widetilde{\boldsymbol{\sigma}}_0 + \sum_{i=1}^{\eta} \boldsymbol{N}_{0|\ldots|i-1}(\boldsymbol{q}) \boldsymbol{J}_i(\boldsymbol{q})^\dagger \boldsymbol{\Lambda}_i \widetilde{\boldsymbol{\sigma}}_i, \tag{5}$$

where $\boldsymbol{N}_{0|\ldots|i}(\boldsymbol{q})$ is the projector onto the null space of the *augmented Jacobian* $\boldsymbol{J}_{0|\ldots|i}(\boldsymbol{q})$ of the i-th task, with $i = 0, \ldots, \eta - 1$, which are respectively defined as follows

$$\boldsymbol{J}_{0|\ldots|i}(\boldsymbol{q}) = \left[\boldsymbol{J}_0(\boldsymbol{q})^\top \cdots \boldsymbol{J}_i(\boldsymbol{q})^\top\right]^\top \tag{6}$$

$$\boldsymbol{N}_{0|\ldots|i}(\boldsymbol{q}) = \boldsymbol{I}_n - \boldsymbol{J}_{0|\ldots|i}(\boldsymbol{q})^\dagger \boldsymbol{J}_{0|\ldots|i}(\boldsymbol{q}). \tag{7}$$

The proposed priority-based task formulation (5) ensures that the execution of the main task is not affected by the remaining tasks in the assigned order. In other words, the execution of the lower priority tasks is subordinated to the execution of the higher priority ones. In this context, the lower priority task will be satisfied if enough and qualified DoFs are available, while the complete fulfillment of the main task is instead always guaranteed. Notice that the convergence of the task errors depends on the annihilation and independence properties of the task Jacobians as defined by Antonelli (2009).

3.1 Dynamic modification of the task composition

As anticipated within the sections above, the considered tasks are put in a prioritized stack that can be on-line modified either by insert/removing a task in/from the stack, or by switching the priority order of one or more tasks. In order to address possible discontinuities of the control input that could be generated by the modification of the stack, a time-vanishing smoothing term is adopted to manage the transitions. Specifically, suppose that the transition phase starts at $t = 0$ and the r-th task of the stack must be deactivated and substituted by the $(r+1)$-th task. During the transition, the velocity command is computed like Caccavale et al (2013) as

$$\dot{\boldsymbol{q}}^*(t) = \dot{\boldsymbol{q}}^*_{r+1}(t) + e^{-\frac{t}{\tau}} \left(\dot{\boldsymbol{q}}^*_r(0) - \dot{\boldsymbol{q}}^*_{r+1}(0) \right), \tag{8}$$

where τ is a positve time constant determining the transition phase duration, and $\dot{\boldsymbol{q}}^*_k$ is the velocity command corresponding to the k-th task of the stack. When t becomes sufficiently higher than τ, the r-th task stack is entirely removed, and a new transition can start. Similarly, equation (8) can be employed to switch the priority between two tasks in the stack. Consider that equation (8) only affects the computed commanded velocity during the transition of different stacks after their composition. The main aim of this equation is to make smoother the activation/deactivation of given tasks.

4 Definition of Possible Tasks for a Cobot System

In this section, the possible tasks of interest for a cobot system are presented. In particular, two different kinds of tasks are discussed: positioning and performance tasks. The latter group includes all the tasks affecting the arm manipulability, while the former tasks are related to the control of the position and orientation of the robot end-effector.

4.1 End-effector position task

The goal of this task is to hold the position of the robot end-effector in a desired configuration. Let $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times n}$ be the Jacobian matrix of the robotic arm, which can be defined as follows

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_p(\boldsymbol{q}) \\ \boldsymbol{J}_o(\boldsymbol{q}) \end{bmatrix}, \tag{9}$$

where $\boldsymbol{J}_p(\boldsymbol{q}) \in \mathbb{R}^{3\times n}$ and $\boldsymbol{J}_o(\boldsymbol{q}) \in \mathbb{R}^{3\times n}$ represent the parts of the Jacobian mapping the joint velocities to the linear and rotational velocities of the end-effector, respectively. Let $\boldsymbol{p}(\boldsymbol{q}) \in \mathbb{R}^3$ and $\boldsymbol{p}^* \in \mathbb{R}^3$ be the current and desired position of the end-effector in the robot base frame, respectively. The corresponding task function is defined as $\boldsymbol{\sigma}_p = \boldsymbol{p}^* - \boldsymbol{p}(\boldsymbol{q})$. Hence, in order to fulfill the task, the reference $\boldsymbol{\sigma}^*_p = \boldsymbol{0}$ is chosen. The corresponding Jacobian matrix for the end-effector positioning task is $\boldsymbol{J}_p(\boldsymbol{q})$.

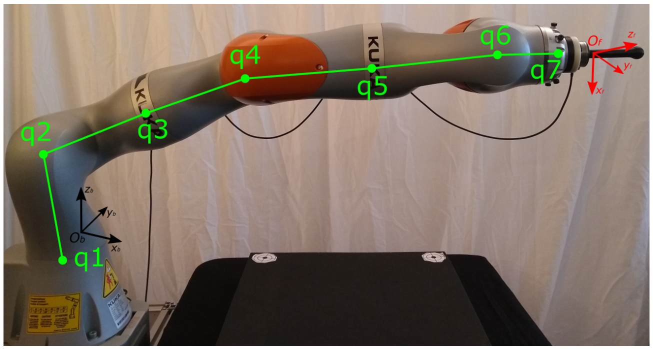


Fig. 2: KUKA LBR iiwa robot joints depicted in green. In black, the robot base frame. In red, the end-effector frame.

4.2 End-effector orientation task

The purpose of this task is to control the orientation of the robot end-effector. In the proposed scenario, the robot should maintain its orientation in a configuration convenient with the considered co-manipulation task. For instance, considering the proposed setup, the end-effector of the robot is commanded to maintain the robot flange facing downwards, as depicted in Fig.1. Let $\boldsymbol{R}^* \in SO(3)$ and $\boldsymbol{R}(\boldsymbol{q}) \in SO(3)$ be the rotation matrices representing the desired and current orientation of the robot end-effector, respectively. The corresponding unit quaternions are denoted by $\{\eta_o(\boldsymbol{q}), \boldsymbol{\epsilon}_o(\boldsymbol{q})\}$ and $\{\eta_o^*, \boldsymbol{\epsilon}_o^*\}$, respectively. The orientation error can be computed as follows

$$\boldsymbol{e}_o(\boldsymbol{q}) = \eta_o(\boldsymbol{q})\boldsymbol{\epsilon}_o^* - \eta_o^*\boldsymbol{\epsilon}_o(\boldsymbol{q}) - \boldsymbol{S}(\boldsymbol{\epsilon}_o^*)\boldsymbol{\epsilon}_o(\boldsymbol{q}), \tag{10}$$

where $\boldsymbol{S}(\cdot)$ is the skew-symmetric matrix. With reference to Fig. 2, the sought goal is to keep the z-axis of the end-effector facing downwards, neglecting the rotation around such an axis. Therefore, only the first two components of the error vector calculated in (10) are employed in the following experiments. Hence, the task function is designed as $\boldsymbol{\sigma}_o = \boldsymbol{\Sigma}\boldsymbol{e}_o$, with $\boldsymbol{\Sigma} = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \end{bmatrix}$, while the desired task variable is $\boldsymbol{\sigma}_o^* = \boldsymbol{0}$. The corresponding task Jacobian matrix is $\boldsymbol{J}_o' = \boldsymbol{\Sigma}\boldsymbol{J}_o \in \mathbb{R}^{2\times n}$.

4.3 Interaction force task

This task intends to implement the compliance behavior of the manipulator exploiting the data generated by the force sensor. Roughly speaking, this task pursues a zero-force control allowing the position of the robot end-effector to be freely moved by the human operator.

Let $\boldsymbol{f}^s \in \mathbb{R}^3$ be the interaction force vector in the sensor frame and $\boldsymbol{R}_s(\boldsymbol{q}) \in \mathbb{R}^{3\times 3}$ the rotation matrix between the sensor frame and the robot base frame. The task function is defined as $\boldsymbol{\sigma}_f = \boldsymbol{R}_s(\boldsymbol{q})\boldsymbol{f}^s$, while the desired value is $\boldsymbol{\sigma}_f^* = \boldsymbol{0}$. In this way, the manipulator moves the position of its end-effector until the human operator does not interact with the robot. The corresponding Jacobian matrix of the task is $\boldsymbol{J}_p(\boldsymbol{q}) - \boldsymbol{S}(\boldsymbol{p}_s(\boldsymbol{q}) - \boldsymbol{p}(\boldsymbol{q}))\boldsymbol{J}_o(\boldsymbol{q})$, with $\boldsymbol{p}_s(\boldsymbol{q}) \in \mathbb{R}^3$ is the position of the tip of the sensor force in the robot base frame. However, to simplify the computations, the approximated task Jacobian $\boldsymbol{J}_p(\boldsymbol{q})$ is considered for this task.

4.4 Joint limits avoidance task

This task is responsible to avoid mechanical joint limits. In order to bring the joints as far as possible from their limits, the robot is solicited to keep a configuration in the middle of such a range. Therefore, the desired joints configuration can be computed as $\boldsymbol{q}^* = \boldsymbol{q}_L + \frac{1}{2}(\boldsymbol{q}_H - \boldsymbol{q}_L)$, where $\boldsymbol{q}_L \in \mathbb{R}^n$ and $\boldsymbol{q}_H \in \mathbb{R}^n$ are the minimum and maximum mechanical limits, respectively. The task function σ_l corresponding to the joint limits task is defined as

$$\sigma_l(\boldsymbol{q}) = \frac{1}{2n} \sum_{i=1}^{n} \left(\frac{q_i - q_i^*}{q_{H_i} - q_{L_i}} \right)^2, \tag{11}$$

with q_i^*, q_{H_i}, and q_{L_i} being the i-th component of $\boldsymbol{q}^*$, $\boldsymbol{q}_H$ and $\boldsymbol{q}_L$, respectively. Equation (11) represents a weighted sum of the error of each joint with respect to its desired configuration in the middle of the mechanical range. The desired value for the task is $\boldsymbol{\sigma}_l^* = 0$, while the corresponding Jacobian matrix $\boldsymbol{J}_l \in \mathbb{R}^{1\times n}$ is equal to

$$\boldsymbol{J}_l = \frac{1}{n} \left[\frac{q_1 - q_1^*}{(q_{H_1} - q_{L_1})^2} \cdots \frac{q_n - q_n^*}{(q_{H_n} - q_{L_n})^2} \right]. \tag{12}$$

4.5 Manipulability task

The goal of this task is to prevent the arm to reach kinematic singularity configurations. For this scope, the manipulability index introduced in Siciliano et al (2009) might be considered. Given an arbitrary arm configuration $\boldsymbol{q}$, the manipulability index is calculated as

$$w(\boldsymbol{q}) = \sqrt{\det\left(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^\top\right)}. \tag{13}$$

However, to save computational resources, simpler manipulability indices exist. In this work, the solution adopted by Caccavale et al (2013) is considered. The manipulability task function can be thus designed in the following way

$$\sigma_m(\boldsymbol{q}) = \frac{1}{2} \sum_{i=2}^{n-1} \sin^2(q_i). \tag{14}$$

The first and last joints are neglected for the considered robot since they do not cause singularity issues. Notice that, for the considered robot manipulator, the above equation has the same local minima of (13), with low values when the robot configuration is close to kinematic singularities.

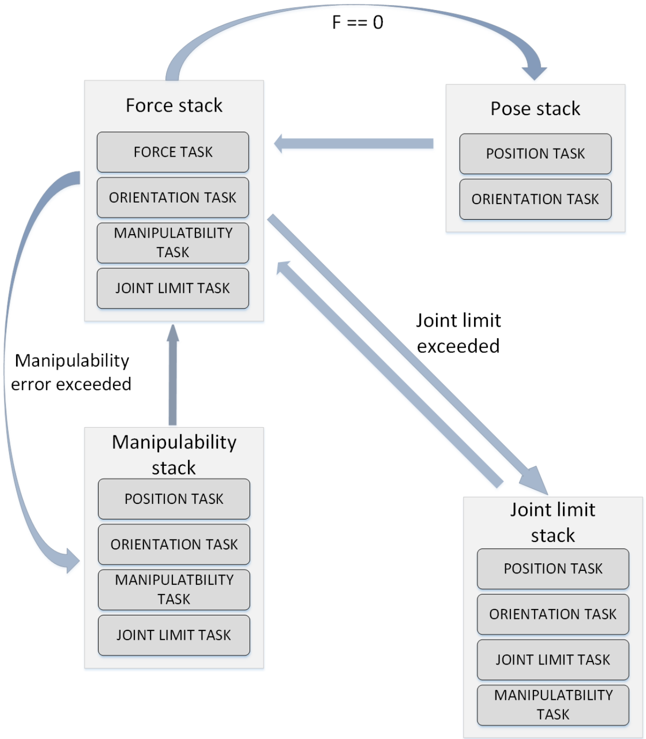


Fig. 3: Policy employed for the addressed co-manipulation system. Four stacks are represented, each of one contains a prioritized order of tasks. The corresponding condition regulates the switch between the stacks. When the switching condition does not hold anymore, the force stack is restored.

The desired task value is $\boldsymbol{\sigma}_m^*$, which is experimentally tuned on the basis of the available setup. One way to retrieve $\boldsymbol{\sigma}_m^*$ is to compute (14) on a discretization of the workspace and take the mean value on this set. Otherwise, it is possible to retrieve $\boldsymbol{\sigma}_m^*$ from a preferred dexterous manipulator configuration. Finally, the corresponding Jacobian matrix $\boldsymbol{J}_m(\boldsymbol{q}) \in \mathbb{R}^{1\times n}$ for the task is calculated as

$$\boldsymbol{J}_m(\boldsymbol{q}) = \left[\cos(q_1)\sin(q_1)\ \ldots\ \cos(q_n)\sin(q_n)\right]. \tag{15}$$

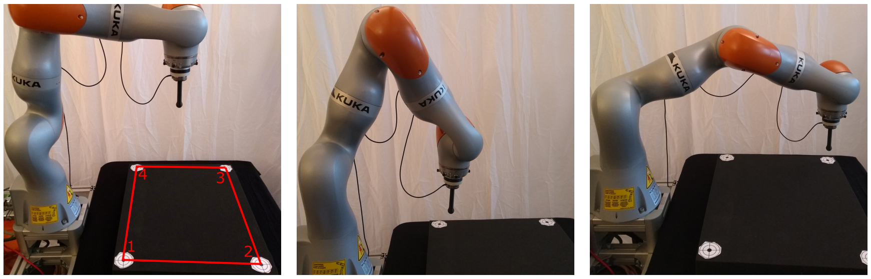
5 Dynamic Task Composition

As previously declared, the stack of the active tasks during the co-manipulation action can be modified on-line by activating or deactivating a certain number of tasks or by changing their priority. The policy employed in the carried out experiments is represented in Fig. 3. Notice that, in principle, any combination of tasks and priorities can be adopted based on the desired cobot behavior. When the human operator does not exert any force on the robot end-effector, the only two active tasks are the position and the orientation ones, keeping the reached pose of the robot. When the operator interacts with the robot, the force stack is active. The force task is the main one, followed by the orientation task, the manipulability task, and the joint limit task. If during the interaction the manipulability index $\sigma_m(\boldsymbol{q})$ goes under a certain threshold $\bar{\sigma}_m$, the manipulability stack is activated. This means that the force task is removed from the stack, and the position task becomes the primary objective. The desired end-effector position is set as the one at the switching moment. Once $\sigma_m(\boldsymbol{q})$ becomes again greater than $\bar{\sigma}_m$, the force task is restored. Instead, if during the interaction the following inequality $|q_i - q_i^*|> \bar{\sigma}_l$ holds for a joint i , with $\bar{\sigma}_l$ a given threshold, the joint limit stack is activated. This means that the force task is removed from the stack, and the position task becomes the primary objective. The desired end-effector position becomes the one at the switching moment. The manipulation and joint limit tasks switch their priorities. Once $|q_i - q_i^*|$ becomes again lower than $\bar{\sigma}_l$, the force task is restored.

Through the described policy, the human operator is prevented from moving the robot when the manipulability index is too low, or one of the joint is diverging too much from its desired value.

6 Case Study

The effectiveness of the proposed system has been assessed by defining a collaborative manipulation case study where a human worker cooperates with a lightweight redundant robotic arm to bring its end-effector in predefined locations. In order to show the advantage of the presented approach, a user, already trained on the system, performed a test with the proposed method and one with a classical admittance control scheme. A video of one representative experiment can be found online[1].



(a) (b) (c)

Fig. 4: Experimental setup: 4(a) the operator must reach the visual markers with the end-effector of the robot, bringing the robot configuration close to the mechanical joint limits (4(b)) and to a kinematic singularity (4(c)).

[1] https://youtu.be/-ZHwauDONQY

The experimental setup is depicted in Fig. 4(a). The purpose of the test is to overcome the depicted markers with the end-effector, drawing a square. Marker placement was chosen to stress the robot configurations to be close to joint limits (see Fig. 4(b)) and kinematic singularity (see Fig. 4(c)). The tests were performed using the Kuka LWR iiwa robot, controlled via ROS middleware, running on GNU/Linux OS as from Hennersperger et al (2016).

Regarding the practical implementation of dynamic task composition, each task is represented by a static matrix, while the related Jacobian is implemented as a dynamic matrix using C++ standard vector class[2] to prevent any issue related to memory reallocation when the dimensions of the Jacobian matrix changes with respect to the activated tasks.

The following gains regarding the proposed methodology were experimentally tuned: the gain for the force task is equal to $\boldsymbol{\Lambda}_f = 0.02\boldsymbol{I}_3$; the gain for the position task is equal to $\boldsymbol{\Lambda}_p = 5.5\boldsymbol{I}_3$; the gain for the orientation task is equal to $\boldsymbol{\Lambda}_o = 5.5\boldsymbol{I}_3$; the gain for the manipulability task is equal to $\boldsymbol{\Lambda}_m = 0.1\boldsymbol{I}_3$; the gain for the joint limits task is equal to $\boldsymbol{\Lambda}_l = 0.001\boldsymbol{I}_3$; the threshold to activate the manipulability stack is set to $\bar{\sigma}_m = 1.5$; the threshold to activate the joint limit stack is set to $\bar{\sigma}_l = 2$ rad. The general admittance control scheme is instead taken from Siciliano et al (2009), with the following values for the apparent dynamics of the end-effector position: desired mass matrix $\boldsymbol{I}_3$; desired damping matrix $100\boldsymbol{I}_3$; desired stiffness matrix $0.001\boldsymbol{I}_3$.

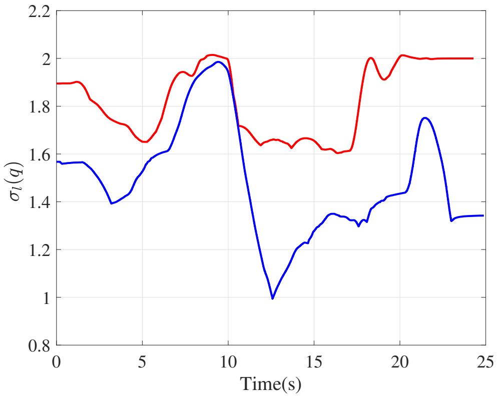


Fig. 5: Time history of the $\sigma_l(\boldsymbol{q})$ index. In blue, during the proposed approach test. In red, during the admittance control test.

A human operator performed the first test with the proposed methodology and, immediately after, one with an admittance control scheme. At any level of precision, it was impossible for the human operator to replicate the same trajectories exactly. Moreover, since the two experiments were performed in cascade,

[2] `https://en.cppreference.com/w/cpp/container/vector`

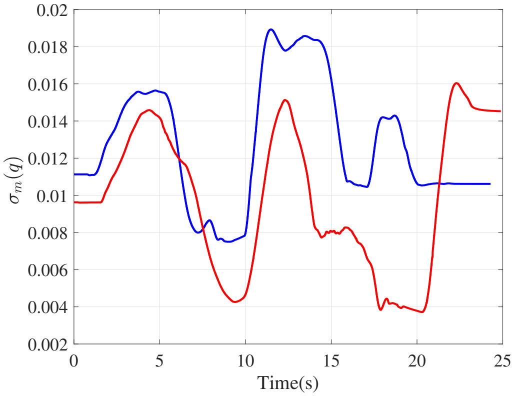


Fig. 6: Time history of the $\sigma_m(\boldsymbol{q})$ index. In blue, during the proposed approach test. In red, during the admittance control test.

the initial conditions of the two tests were slightly different as well. The time history of the $\sigma_l(\boldsymbol{q})$ index is depicted in Fig. 5. In blue, the one obtained through the proposed approach, in red the one obtained through the admittance control scheme. It is clear that the proposed methodology outperforms the classical admittance control scheme since the joint limit avoidance task is explicitly taken into account. The history of the stack switching in Fig. 7 highlights that the procedure made suitable changes in the task priority to deal with those situations in which some joint was very far from its desired value. The time history of the $\sigma_m(\boldsymbol{q})$ index is depicted in Fig. 6. In blue, the one obtained through the proposed approach, in red the one obtained through the admittance control scheme. It is again clear that the proposed methodology outperforms the classical admittance control scheme for most of the experiment. At the end of the test, the admittance control obtained a better configuration regarding the robot manipulability. This can be anyway handled by the proposed methodology by adding, for instance, the manipulability as a lower priority task within the pose stack.

7 Conclusion and Future Work

A hierarchical prioritized task control with dynamic task composition was presented in this paper to allow an effective co-manipulation for redundant manipulators. In the proposed approach, the kinematic redundancy of the robot was exploited to increase the performance of the manipulator during human-robot physical interaction. In particular, a set of primary and secondary tasks were composed to avoid mechanical joint limits and kinematic singularity during the execution of a shared task. The human operator was allowed to freely move the position of the robot end-effector through a proper force sensor mounted on the tip of the manipulator. To test the benefits of the proposed cobot system, a standard admittance control was compared to the proposed approach. The collected results demonstrated the effectiveness of the system.

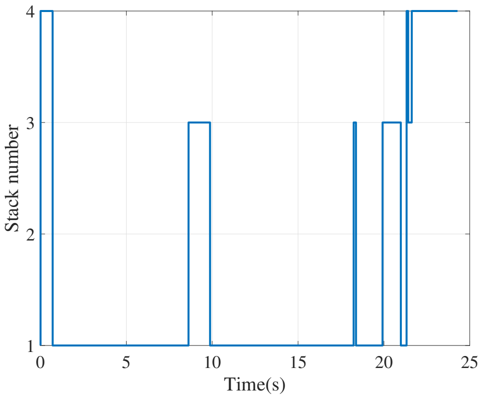


Fig. 7: Time history of the switching between the stack during the proposed approach test. Stack 1 is the force stack. Stack 2 is the manipulability stack. Stack 3 is the joint limit stack. Stack 4 is the pose stack.

Future work is dedicated to adding a phase in which the movements of the human operator can be employed to predict which of the secondary tasks is going to fail and re-arrange the prioritized stack accordingly.

Ackwnoledgments

The research leading to these results has been partially supported by the H2020-ICT-731590 REFILLS project; the WELDON project, in the frame of Programme STAR, financially supported by UniNA and Compagnia di San Paolo; the PRINBOT project, in the frame of the PRIN 2017 research program with grant number 20172HHNK5_002. The authors are solely responsible for its content.

Bibliography

Aertbeliën E, De Schutter J (2014) eTasl/eTC: A constraint-based task specification language and robot controller using expression graphs. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 1540–1546

Agravante D, Cherubini A, Bussy A, Gergondet P, Kheddar A (2014) Collaborative human-humanoid carrying using vision and haptic sensing. In: 2014 IEEE International Conference on Robotics and Automation, pp 607–612

Antonelli G (2009) Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems. IEEE Transactions on Robotics 25:985–994

Baerlocher P, Boulic R (1998) Task-priority formulations for the kinematic control of highly redundant articulated structures. In: 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 323–329

Buonocore L, Cacace J, Lippiello V (2015) Hybrid visual servoing for aerial grasping with hierarchical task-priority control. In: 2015 23rd Mediterranean Conference on Control and Automation, pp 617–623

Caccavale F, Lippiello V, Muscio G, Pierri F, Ruggiero F, Villani L (2013) Grasp planning and parallel control of a redundant dual-arm/hand manipulation system. Robotica 31(7):1169–1194

Caccavale R, Cacace J, Fiore M, Alami R, Finzi F (2016) Attentional supervision of human-robot collaborative plans. In: 2016 25th IEEE International Symposium on Robot and Human Interactive Communication, pp 867–873

Casalino G, Zereik E, Simetti E, Torelli S, Sperind A, Turetta A (2012) A task subsystem priority based control strategy for underwater floating manipulators. IFAC Proceedings Volumes 45(5):170–177

Chiaverini S (1997) Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. IEEE Transactions on Robotics and Automation 13(3):398–410

Corrales JA, Garcia Gomez GJ, Torres F, Perdereau V (2012) Cooperative tasks between humans and robots in industrial environments. International Journal of Advanced Robotic Systems 9

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

Dubey R, Luh JYS (1988) Redundant robot control using task based performance measures. Journal of Field Robotics 5:409–432

Ficuciello F, Romano A, Villani L, Siciliano B (2014) Cartesian impedance control of redundant manipulators for human-robot co-manipulation. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 2120–2125

Gullapalli V, Grupen R, Barto A (1992) Learning reactive admittance control. In: 1992 IEEE International Conference on Robotics and Automation, pp 1475–1480

Hennersperger C, Fuerst B, Virga S, Zettinig O, Frisch B, Neff T, Navab N (2016) Towards mri-based autonomous robotic US acquisitions: A first feasibility study. IEEE transactions on medical imaging 36(2):538–548

Labrecque P, Gosselin C (2018) Variable admittance for phri: From intuitive unilateral interaction to optimal bilateral force amplification. Robotics and Computer-Integrated Manufacturing 52:1–8

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

Lippiello V, Cacace J, Santamaria-Navarro A, Andrade-Cetto J, Trujillo M, Esteves Y, Viguria A (2016) Hybrid visual servoing with hierarchical task composition for aerial manipulation. IEEE Robotics and Automation Letters 1(1):259–266

Mansard N, Stasse O, Evrard P, Kheddar A (2009) A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks. In: 2009 International Conference on Advanced Robotics

Nakamura Y, H H, Yoshikawa T (1987) Task-priority based redundancy control of robot manipulators. The International Journal of Robotics Research 6(2):3–15

Newman WS (1992) Stability and Performance Limits of Interaction Controllers. Journal of Dynamic Systems, Measurement, and Control 114(4):563–570

Rocchi A, Hoffman E, Caldwell D, Tsagarakis N (2015) OpenSoT: A whole-body control library for the compliant humanoid robot COMAN. In: 2015 IEEE International Conference on Robotics and Automation, pp 6248–6253

Siciliano B, Sciavicco L, Villani L, Oriolo G (2009) Robotics: Modelling, Planning and Control. Springer-Verlag London Limited

Yoshikawa T (1985) Manipulability of robotic mechanisms. The International Journal of Robotics Research 4(2):3–9

Zanchettin A, Rocco P (2017) Probabilistic inference of human arm reaching target for effective human-robot collaboration. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 6595–6600