Haptic-based Shared-Control Methods for a Dual-Arm System

Mario Selvaggio[1], Firas Abi-Farraj[2], Claudio Pacchierotti[2], Paolo Robuffo Giordano[2] and Bruno Siciliano[1]

Abstract—We propose novel haptic guidance methods for a dual-arm telerobotic manipulation system, which are able to deal with several different constraints, such as collisions, joint limits, and singularities. We combine the haptic guidance with shared-control algorithms for autonomous orientation control and collision avoidance meant to further simplify the execution of grasping tasks. The stability of the overall system in various control modalities is presented and analyzed via passivity arguments. In addition, a human subject study is carried out to assess the effectiveness and applicability of the proposed control approaches both in simulated and real scenarios. Results show that the proposed haptic-enabled shared-control methods significantly improve the performance of grasping tasks with respect to the use of classic teleoperation with neither haptic guidance nor shared control.

Index Terms—Haptics and Haptic Interfaces; Telerobotics and Teleoperation; Industrial Robots.

I. INTRODUCTION

OVER the last few years remote telemanipulation has shown significant advancements and promising results have been achieved in several fields such as minimally-invasive robotic surgery [1], sort and segregation of nuclear waste [2], [3], telemaintenance [4], and micromanipulation [5], [6]. In this respect, the nuclear industry is one of the most exciting fields of application for robotic telemanipulation, where several tasks, such as sort and segregation of nuclear waste, can clearly benefit from advanced telemanipulation techniques.

Current telerobotic systems designed for such tasks provide teleoperation capabilities through extremely primitive master consoles (e.g., passive joystick or teach pendants), making these operations prohibitively slow to process large amounts of material in a reasonable time. Besides being time demanding, these tasks usually require highly-skilled human operators. Indeed, steering a remote manipulator toward a desired grasping pose is a quite complex task for an operator directly controlling the 6-degrees-of-freedom (DoF) pose of a manipulator end-effector. This is due to (i) the complexity of regulating both the position and the orientation at the same time and (ii) the presence of several constraints (e.g., collisions, joint limits,

[1]Department of Information Technology and Electrical Engineering, University of Naples Federico II, Naples, Italy. {mario.selvaggio,bruno.siciliano}@unina.it
[2]CNRS, Univ Rennes, Inria, IRISA, Rennes, France. {firas.abi-farraj,claudio.pacchierotti,prg}@irisa.fr

singularities) that further limit the operator's maneuvering dexterity (but of which the operator has no direct or intuitive awareness). To overcome these limitations, the user needs to constantly pay close attention to the status of the robotic system, which can be sometimes difficult and cognitively demanding (e.g., for singularity or joints limit avoidance). A possible way to reduce the operator cognitive and physical workload is to exploit the sensory information collected at the slave side to design novel haptic-guided and shared control-enabled teleoperation systems.

Kinematic limitations avoidance has been recently applied to collaborative robots to improve the performance and the intuitiveness of physical human-robot interaction [7]. In robotic teleoperation this approach can be implemented through haptic guidance methods: instead of providing a high-fidelity haptic feedback, reflecting the actual physical contacts between a slave manipulator and the environment, the human operator is provided with an interface which informs her/him about proximity to constraints. Proposed solutions to this problem are based on potential fields [8], virtual spring-damper systems [9], and/or virtual fixtures [10], [11]. When the slave side is a complex robotic system (e.g., dual-arm, highly redundant) or environmental obstacles are present, (self-)collision avoidance methods need to be considered. In the past, several collision avoidance algorithms have been developed to implement reactive control strategies or to plan collision-free paths for redundant robots [12]–[15]. In this regard, self-collisions and joint limits have been used as criteria to find optimal inverse kinematic solutions for robotic manipulators [16], [17]. The idea of combining kinematic constraints and self-collision avoidance to create safe and intuitive haptic-guided telerobotic system has been preliminarily introduced in [18].

On the other hand, several shared-control techniques for telerobotic systems have been proposed in the past [19]–[21]. For instance, in [2], a sensory-based shared-control architecture for remote manipulation is presented: a visual-based autonomous algorithm regulates a subset of the gripper's DoF to ease the approach toward an object to grasp. At the same time, the operator has control over the remaining null-space motions w.r.t. the primary (autonomous) task and is informed about possible constraints of the single-slave robotic system through haptic cues. However, all these previous works consider one slave arm operating in a static obstacle-free environment.

In this paper, we consider the case of a dual-arm robotic system: one robot is (partially) teleoperated by a human, while the other autonomously executes a task in the same workspace. Similar scenarios are considered in [22], [23], where an operator teleoperates a robotic arm while a camera, mounted on a second
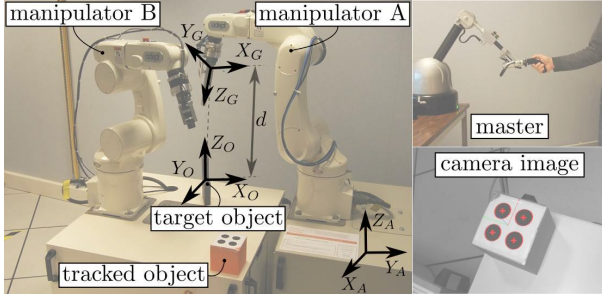
Fig. 1. System architecture and main reference frames attached to the gripper and the object to be grasped.

manipulator, keeps the scene visible. In [22], an occlusion-avoidance algorithm controls the manipulator with the camera and keeps a target object in the field of view. No haptic guidance is considered. In [23], users are provided with haptic feedback about proximity to kinematic constraints of the manipulator they are controlling. However, the two manipulators are far enough to prevent any interference between them.

In this work, two manipulators (one of which is controlled by the operator) have independent and sometimes conflicting tasks. Moreover, they share a limited workspace, risking dangerous collisions. The primary goal of our paper is to define and assess a suitable haptic shared-control interface for this system which facilitates the operator's task and decrease her/his mental workload in avoiding system constraints. To this end, we adapt and improve the approaches presented in [2] and in [18], combining their features, extending their capabilities, and providing a novel, extensive human subject evaluation and results analysis. Specifically,

- we improved the shared-control approach [2] by (i) dividing the approach-to-grasp phase into multiple parts, according to the gripper distance from the target object, and (ii) devising different, optimized strategies for each of these parts;
- we improved the self-collision-aware approach presented in [18] by (i) also considering collisions with the surrounding environment and (ii) replacing the convex meshes collision model of the robot with discretized sphere volumes;
- we combined the proposed, new above mentioned techniques to create a novel shared-control framework able to manage a dual-arm system, where one robot is (partially) controlled by the operator and the other one autonomously performs a task in the same environment;
- we presented a port-Hamiltonian model of the overall system subject to various control modalities, proving the passivity of the system w.r.t. the operator actions;
- we carried out an extensive human subject evaluation in simulated and real environments, enrolling a total of 20 participants.

II. SYSTEM ARCHITECTURE

Our telemanipulation system consists of two serial manipulators sharing a workspace in an industrial setting. One of the manipulators is performing an independent fully-autonomous task (in our case, a visual inspection with an onboard camera), while the second manipulator is (partially) teleoperated by a human operator whose aim is to grasp a target object (see

Fig. 1). The operator faces a number of constraints while controlling such a system: kinematic/workspace constraints for the manipulator she/he is directly controlling, as well as possible interferences with the task performed by the fully-autonomous arm. As explained, the purpose of this work is to detail and assess a suitable shared-control interface to facilitate the operator's task and decrease her/his mental workload.

A. System Modeling

With reference to Fig. 1, the slave system consists of two 6-DoF velocity-controlled manipulator arms, A and B, with joint configurations $\boldsymbol{q}_A \in \mathbb{R}^6$ and $\boldsymbol{q}_B \in \mathbb{R}^6$, respectively. Manipulator A is (partially) controlled by a human operator, while manipulator B autonomously performs a visual task with a potential overlap with the workspace of manipulator A. We also let

$$\boldsymbol{u}_A = \left[\begin{array}{c} \boldsymbol{v}_A \\ \boldsymbol{\omega}_A \end{array} \right] = \boldsymbol{J}_A \dot{\boldsymbol{q}}_A, \qquad \boldsymbol{u}_B = \left[\begin{array}{c} \boldsymbol{v}_B \\ \boldsymbol{\omega}_B \end{array} \right] = \boldsymbol{J}_B \dot{\boldsymbol{q}}_B \quad (1)$$

be the Cartesian linear/angular velocities of the gripper (for manipulator A) and of the camera (for manipulator B) in their respective frames, and $(\boldsymbol{J}_A, \boldsymbol{J}_B)$ the associated geometric Jacobians. In the following, we will equivalently consider joint velocities $(\dot{\boldsymbol{q}}_A, \dot{\boldsymbol{q}}_B)$ or Cartesian velocities $(\boldsymbol{u}_A, \boldsymbol{u}_B)$ as available control inputs, depending on the context.

The master device is instead modeled as a generic, gravity pre-compensated, mechanical system,

$$\boldsymbol{M}_m(\boldsymbol{x}_m)\dot{\boldsymbol{u}}_m + \boldsymbol{C}_m(\boldsymbol{x}_m, \boldsymbol{u}_m)\boldsymbol{u}_m + \boldsymbol{B}_m\boldsymbol{u}_m = \boldsymbol{f}_m + \boldsymbol{f}_h, \quad (2)$$

where $\boldsymbol{x}_m \in \mathbb{R}^6$ represents the device pose in Cartesian space and $\boldsymbol{u}_m \in \mathbb{R}^6$ the associated linear/angular velocity, $\boldsymbol{M}_m(\boldsymbol{x}_m) \in \mathbb{R}^{6\times6}$ is the positive-definite and symmetric inertia matrix, $\boldsymbol{C}_m(\boldsymbol{x}_m, \boldsymbol{u}_m) \in \mathbb{R}^{6\times6}$ consists of the Coriolis/centrifugal terms, $\boldsymbol{B}_m$ accounts for the friction term, and $\boldsymbol{f}_m, \boldsymbol{f}_h \in \mathbb{R}^6$ are the control and human forces, respectively. The control forces $\boldsymbol{f}_m = [\dots f_{m,i} \dots]^T \in \mathbb{R}^6$ will be exploited to provide force cues to the user meant to inform of the feasibility of her/his commands against the constraints of the slave system (see Sec. II-D).

B. Constraints on the Slave Side

Constraints at the slave side are encoded in a suitable cost function $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$, whose gradient w.r.t. the joint configuration vectors will be exploited to generate force cues $\boldsymbol{f}_m$ provided to the operator and to implement the reactive behavior of manipulator B (see Sec. II-C).

1) Joint Limits: As customary, we consider that both manipulator arms are subject to joint limits. Among many possibilities, we adopt the following cost function to encode proximity to joint limits

$$h_J(\boldsymbol{q}) = \rho_J \sum_{i=1}^{6} \left(e^{-\alpha_J(q_{i,\max} - q_i)} (q_{i,\max} - q_i)^{-\beta_J} + \right.$$
$$\left. e^{-\alpha_J(q_i - q_{i,\min})} (q_i - q_{i,\min})^{-\beta_J} \right), \quad (3)$$

where $\boldsymbol{q}$ is the configuration vector of one of the slave arms, $(q_{i,\min}, q_{i,\max})$ the i-th min/max joint limits, and $(\rho_J, \alpha_J, \beta_J)$ are scalar positive constants. The function $h_J(\boldsymbol{q})$ has a minimum at the center of the joint range and grows to infinity as one of the joints approaches its limits.
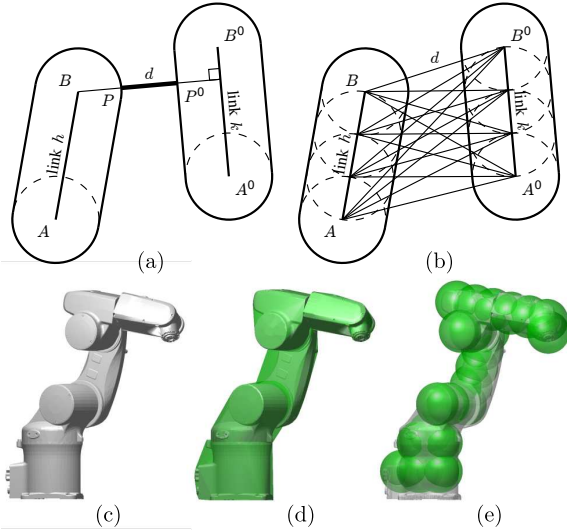
Fig. 2. Minimum distance between two line sphere-swept bounding volumes (a) and distances among discrete sphere-swept bounding volumes (b). Solid (c), convex (d), and discrete-sphere (e) collision models of the robot.

2) Singularities: As second constraint, we consider singularities. This constraint is encoded by the cost function [24],

$$h_S(\boldsymbol{q}) = \rho_S e^{-\alpha_S \det(\boldsymbol{\Gamma}\boldsymbol{J}(\boldsymbol{q}))} (\det(\boldsymbol{\Gamma}\boldsymbol{J}(\boldsymbol{q})))^{-\beta_S}, \quad (4)$$

where, again, $(\rho_S, \alpha_S, \beta_S)$ are scalar positive constants, and $\boldsymbol{q}$ and $\boldsymbol{J}$ stand for the joint vector and Jacobian of the manipulator A or B, respectively, and $\boldsymbol{\Gamma}$ is a diagonal and positive-definite weight matrix that accounts for non-homogeneous physical dimensions of the Jacobian entries, namely, rotations and translations. The function $h_S(\boldsymbol{q})$ grows to infinity as $\det(\boldsymbol{J}(\boldsymbol{q})) \to 0$ and it vanishes exponentially as $\det(\boldsymbol{J}(\boldsymbol{q})) \to \infty$.

3) Collisions: As final constraint, we consider collisions between the two slave manipulators.

Any cost function encoding the collision avoidance constraint needs to exploit some measure of the minimum distance between the two arms. In order to facilitate the computation of the minimum distance between two bodies, we take inspiration from the line sphere-swept model [25] (see Fig. 2a). However, as well known, the minimum distance between two (also convex) bounding volumes can have a discontinuous gradient when the pair of closest points on the two shapes suddenly jumps due to small perturbations of the shape locations. We therefore opted for discretizing the bounding volumes with a finite number of spheres, and exploiting all the possible inter-sphere distances to build the cost function associated to the collision avoidance (see Fig. 2b). An illustrative example of the resulting collision model is given in Fig. 2e, where one can appreciate how the adopted discretized-sphere model represents a fairly good approximation of the manipulator original and convex mesh models (Fig. 2c-d).

Let then $d_{ij}(\boldsymbol{q}_A, \boldsymbol{q}_B)$ represent the minimum distance between the i-th sphere on manipulator A and the j-th sphere on manipulator B. The collision avoidance cost function is defined as

$$h_C(\boldsymbol{q}_A, \boldsymbol{q}_B) = \rho_C \sum_{i=1}^{S_A} \sum_{j=1}^{S_B} e^{-\alpha_C d_{ij}} d_{ij}^{-\beta_C}, \quad (5)$$

where S_A and S_B are the number of spheres used for discretizing the bodies of manipulators A and B, and $(\rho_C, \alpha_C, \beta_C)$ are scalar positive constants. One can easily verify that $h_C(\boldsymbol{q}_A, \boldsymbol{q}_B) \to \infty$ as any of the inter-sphere distances vanishes while $h_C(\boldsymbol{q}_A, \boldsymbol{q}_B) \to 0$ as all the inter-sphere distances become large enough.

4) Total Cost Function: Finally, the total cost function accounting for all the above mentioned constraints is

$$\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B) = h_J(\boldsymbol{q}_A) + h_J(\boldsymbol{q}_B) + h_S(\boldsymbol{q}_A) + h_S(\boldsymbol{q}_B) + h_C(\boldsymbol{q}_A, \boldsymbol{q}_B). \quad (6)$$

As explained, the gradient of $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$ w.r.t. the joint vector $\boldsymbol{q}_A$ can be used for cueing the operator about the feasibility of her/his commands against the constraints of the slave side, while the gradient of $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$ w.r.t. the joint vector $\boldsymbol{q}_B$ can be used for implementing a reactive behavior in manipulator B for avoiding possible collisions with manipulator A (see Sec. II-C). Note that these two actions (force cues and reactive behavior) are potentially coupled because of the mixed term $h_C(\boldsymbol{q}_A, \boldsymbol{q}_B)$.

C. Slave Control

We start detailing the control architecture of manipulator B, which performs an autonomous visual task to keep an object of interest in visibility during the operation. Let $\boldsymbol{s} \in \mathbb{R}^2$ represent the image plane location of a representative point on the target object (see Fig. 1), $\boldsymbol{s}_d$ a desired value for $\boldsymbol{s}$, and $\boldsymbol{L}_s \in \mathbb{R}^{2 \times 6}$ the associated point feature interaction matrix [26], such that $\dot{\boldsymbol{s}} = \boldsymbol{L}_s \boldsymbol{J}_B \dot{\boldsymbol{q}}_B$. Manipulator B is then controlled by employing the usual projected gradient control [27] (hereafter we use $\mathcal{H}$ to indicate $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$)

$$\dot{\boldsymbol{q}}_B = k_{B1} \underbrace{(\boldsymbol{L}_s \boldsymbol{J}_B)^\dagger (\boldsymbol{s}_d - \boldsymbol{s})}_{\substack{\text{primary task} \\ \text{(visual servoing)}}} - k_{B2} \underbrace{(\boldsymbol{I} - (\boldsymbol{L}_s \boldsymbol{J}_B)^\dagger (\boldsymbol{L}_s \boldsymbol{J}_B)) \nabla_{\boldsymbol{q}_B} \mathcal{H}}_{\substack{\text{secondary task} \\ \text{(minimize } \mathcal{H})}},$$
$$(7)$$

with $k_{B1} > 0$ and $k_{B2} \geq 0$. The primary task consists in the regulation of $\boldsymbol{s}$ toward $\boldsymbol{s}_d$. Its null-space is then exploited for the secondary task of minimizing the constraint cost function $\mathcal{H}$ in (6). This null-space action will keep manipulator B away from its singularities and joint limits, as well as maintain a safe distance from manipulator A. This action can be deactivated by setting $k_{B2} = 0$, asking manipulator B to carry out only the primary visual servoing task.

For manipulator A, we instead considered two possible control modalities: *Full Teleoperation* and *Shared Control*.

1) Full Teleoperation: In this modality, the user is given full control over the 6-DoF pose of the gripper on manipulator A. This is achieved by simply setting

$$\boldsymbol{u}_A = k_A \boldsymbol{u}_m, \quad k_A > 0 \quad (8)$$

thereby implementing a classic velocity-velocity coupling between master and slave gripper linear/angular velocities.

2) Shared Control: The shared-control modality is meant to facilitate the user's operation in approaching the object to be grasped by letting an autonomous algorithm control a suitable subset of the gripper pose. This modality is split into two phases depending on the distance between the gripper and the object.
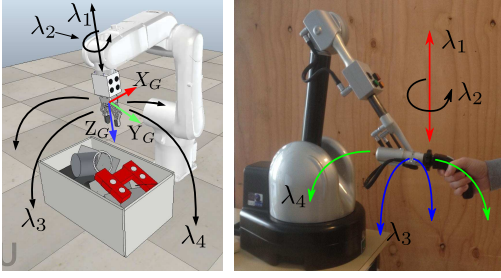
Fig. 3. A visualization of the four null-space directions defined in (12).

With reference to Fig. 1, consider the frames $\mathcal{F}_O$: $\{O_O; X_O, Y_O, Z_O\}$ attached to the object to be grasped and $\mathcal{F}_G$: $\{O_G; X_G, Y_G, Z_G\}$ attached to the gripper of manipulator A, let $^G P_O$ represent the object position in the gripper frame and $d = \|^G P_O\|$ the gripper/object distance. When $d < d_{th}$ (threshold value), we adopt the shared-control strategy described in [2], whose aim is to autonomously keep the pointing direction toward the object

$$p = \frac{^G P_O}{\|^G P_O\|} \in \mathbb{S}^2 \qquad (9)$$

always aligned with the gripper approaching direction $p_d = Z_G$, while the operator is in control of the remaining DoF in the null-space of this primary task. By doing so, the gripper is always oriented toward the object, thus helping the user in completing the task. This behavior is obtained as follows: defining $P_p = I - pp^T$, $[\cdot]_\times$ the usual skew-symmetric operator, and

$$L_p = \left[\begin{array}{cc} -\frac{1}{d} P_p & [p]_\times \end{array} \right] \in \mathbb{R}^{3 \times 6} \qquad (10)$$

the interaction matrix associated to the task (9) (such that $\dot{p} = L_p u_A$), the following controller is implemented on manipulator A,

$$u_A = k_{A1} L_p^\dagger (p_d - p) + k_{A2} N \Lambda u_m \qquad (11)$$

with $k_{A1} \geq 0$ and $k_{A2} > 0$. Here, $N = [n_1 \ldots n_4] \in \mathbb{R}^{6 \times 4}$ collects the four null-space directions

$$n_1 = \begin{bmatrix} p \\ 0 \end{bmatrix}, n_2 = \begin{bmatrix} 0 \\ p \end{bmatrix}, n_3 = \begin{bmatrix} -[p]_\times e_y \\ -P_p e_y / d \end{bmatrix}, n_4 = \begin{bmatrix} [p]_\times e_x \\ P_p e_x / d \end{bmatrix}, \qquad (12)$$

with $e_x = [1\ 0\ 0]^T$ and $e_y = [0\ 1\ 0]^T$, which represent a basis of the four-dimensional null-space of the primary task, and $\Lambda \in \mathbb{R}^{4 \times 6}$ is a selection matrix for extracting four components[1] of u_m used by the operator for commanding a 'velocity' along the four motion directions spanned by N. Figure 3 provides a visual illustration of the four null-space motion directions associated to (12), while further details can be found in [2].

On the other hand, when $d \geq d_{th}$, we do not constrain the gripper to be oriented toward the object, since, when far from the target object, it would unnecessarily limit the slave maneuverability. Therefore, when $d \geq d_{th}$, the shared-control algorithm switches to a modality in which the operator has control over the *translational* motion of the slave gripper, while the gripper orientation is autonomously controlled so as to minimize the cost function $\mathcal{H}$ and, thus, stay away as much

---

[1]In this case, we assume that the motion of the master device is blocked (via software) along the two non-selected components.

as possible from the system constraints. This is obtained by replacing (11) with

$$u_A = k_{A1} S u_m - k_{A2} Z J_A^{-T} \nabla_{q_A} \mathcal{H}, \qquad (13)$$

where

$$Z = \left[\begin{array}{cc} 0 & 0 \\ 0 & I_3 \end{array} \right], \quad S = \left[\begin{array}{cc} I_3 & 0 \\ 0 & 0 \end{array} \right] \in \mathbb{R}^{6 \times 6} \qquad (14)$$

are selection matrices. As shown in the Appendix, controller (13) is indeed able to minimize $\mathcal{H}$ when $S u_m = 0$ (i.e., in free translational motion) by acting on the angular velocity of manipulator A.

D. Haptic guidance

As mentioned before, the control forces f_m on the master side (2) are exploited to inform the operator about the feasibility of her/his commands against the system constraints, encoded in the cost function $\mathcal{H}$. This is obtained as follows, depending on the control modality considered for manipulator A:

1) Full Teleoperation: In this case, manipulator A is controlled by (8), and the force cues are generated as

$$f_m = -k_M J_A^{-T} \nabla_{q_A} \mathcal{H}, \qquad k_M > 0, \qquad (15)$$

the rationale being that (15) provides the force that, when applied to all the master device DoF, would make the master move so as to minimize $\mathcal{H}$ at the slave side. Therefore, cues (15) provide the human operator with information about where to move in order to stay away from the constraints.

2) Shared Control: When manipulator A is close enough to the object ($d < d_{th}$), its behavior is controlled by (11). In this case, the force cues are generated as

$$f_m = -k_M \Lambda^T N^T J_A^{-T} \nabla_{q_A} \mathcal{H}, \qquad k_M > 0. \qquad (16)$$

Similarly to the previous case, cues (16) represent the forces that should be applied to the free directions of the master device for letting the manipulator A minimize $\mathcal{H}$ along the null-space directions spanned by N (where the operator can act).

When manipulator A is instead far from the object ($d \geq d_{th}$), its behavior is regulated by (13), and the force cues are

$$f_m = -k_M S J_A^{-T} \nabla_{q_A} \mathcal{H}, \quad k_M > 0, \qquad (17)$$

with again the idea of providing a force feedback that, when applied to the free master DoF (i.e., the translational ones in this case), would make manipulator A move so as to minimize $\mathcal{H}$.

III. PASSIVITY ANALYSIS

Enforcing passivity is very important when dealing with teleoperation systems in order to guarantee a stable closed-loop behavior. Passivity has been extensively studied and applied to haptic feedback teleoperation systems, especially when time delays are present [28]. In this section, our aim is to analyze the passivity of the considered teleoperation system subject to the control modalities introduced in Secs. II-C and II-D.

By considering the two velocity-controlled robots as simple integrators, the total energy of the system can be written as

$$\mathcal{V}(p_m, q_A, q_B) = \frac{1}{2} p_m^T M_m^{-1} p_m + \mathcal{H}(q_A, q_B) \qquad (18)$$

where $\boldsymbol{p}_m = \boldsymbol{M}_m \boldsymbol{u}_m$ is the haptic device momentum, $\boldsymbol{q}_A$, $\boldsymbol{q}_B \in \mathbb{R}^6$ are the generalized coordinates of the two slave manipulators, respectively, and $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$, introduced in (6), is the potential energy associated with the system constraints. We can thus analyze passivity by showing that the closed-loop system equations can be put in port-Hamiltonian form. Formally, a port-Hamiltonian system (PHS) can be represented by the following set of equations

$$\begin{cases} \dot{\boldsymbol{x}} = [J(\boldsymbol{x}) - R(\boldsymbol{x})]\dfrac{\partial \mathcal{V}}{\partial \boldsymbol{x}} + g(\boldsymbol{x})\,\boldsymbol{u} \\ \boldsymbol{y} = g^{\mathrm{T}}(\boldsymbol{x})\dfrac{\partial \mathcal{V}}{\partial x} \end{cases} \tag{19}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the system state, $\mathcal{V}(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ is the Hamiltonian function, namely the sum of system energies, $J(\boldsymbol{x}) = -J(\boldsymbol{x})^{\mathrm{T}}$ represents the internal interconnection, $R(\boldsymbol{x}) \geq 0$ the internal dissipation, $g(\boldsymbol{x})$ the input matrix, $\boldsymbol{u}$ the system input, and $\boldsymbol{y}$ the system output. It is easy to show that for PHS the following inequality holds

$$\boldsymbol{u}^{\mathrm{T}}\boldsymbol{y} = \dot{\mathcal{V}}(\boldsymbol{x}) - \frac{\partial^{\mathrm{T}}\mathcal{V}}{\partial \boldsymbol{x}}R(\boldsymbol{x})\frac{\partial \mathcal{V}}{\partial \boldsymbol{x}} \leq \dot{\mathcal{V}}(\boldsymbol{x}). \tag{20}$$

Equation (20) establishes the inherent passivity condition of a PHS with respect to the input-output pair $(\boldsymbol{u}, \boldsymbol{y})$ with storage function $\mathcal{V}(\boldsymbol{x})$. In the following, we then show that the three control modalities described in Sec. II lead to a closed-loop PHS formulation. For the analysis, we consider the primary task in (7) and (11) to have reached a steady state ($\boldsymbol{s} \to \boldsymbol{s}_d$ and $\boldsymbol{p} \to \boldsymbol{p}_d$).

1) Full teleoperation: setting $k_M = k_A = k$, the closed-loop system can be written as follows

$$\begin{bmatrix} \dot{\boldsymbol{p}}_m \\ \dot{\boldsymbol{q}}_A \\ \dot{\boldsymbol{q}}_B \end{bmatrix} = \left(\begin{bmatrix} 0 & -k\boldsymbol{J}_A^{-T} & 0 \\ k\boldsymbol{J}_A^{-1} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \right.$$
$$\left. - \begin{bmatrix} \boldsymbol{B}_m & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{P} \end{bmatrix} \right) \begin{bmatrix} \nabla_{\boldsymbol{p}_m}\mathcal{V} \\ \nabla_{\boldsymbol{q}_A}\mathcal{V} \\ \nabla_{\boldsymbol{q}_B}\mathcal{V} \end{bmatrix} + \begin{bmatrix} \mathbf{1} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{f}_h. \tag{21}$$

2) Shared control ($d < d_{th}$): setting $k_M = k_{A2} = k$, the closed-loop system can be written as follows

$$\begin{bmatrix} \dot{\boldsymbol{p}}_m \\ \dot{\boldsymbol{q}}_A \\ \dot{\boldsymbol{q}}_B \end{bmatrix} = \left(\begin{bmatrix} 0 & -k\boldsymbol{\Lambda}^T\boldsymbol{N}^T\boldsymbol{J}_A^{-T} & 0 \\ k\boldsymbol{J}_A^{-1}\boldsymbol{N}\boldsymbol{\Lambda} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \right.$$
$$\left. - \begin{bmatrix} \boldsymbol{B}_m & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{P} \end{bmatrix} \right) \begin{bmatrix} \nabla_{\boldsymbol{p}_m}\mathcal{V} \\ \nabla_{\boldsymbol{q}_A}\mathcal{V} \\ \nabla_{\boldsymbol{q}_B}\mathcal{V} \end{bmatrix} + \begin{bmatrix} \mathbf{1} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{f}_h. \tag{22}$$

3) Shared control ($d \geq d_{th}$): setting $k_M = k_{A1} = k$, the closed-loop system can be written as follows

$$\begin{bmatrix} \dot{\boldsymbol{p}}_m \\ \dot{\boldsymbol{q}}_A \\ \dot{\boldsymbol{q}}_B \end{bmatrix} = \left(\begin{bmatrix} 0 & -k\boldsymbol{S}\boldsymbol{J}_A^{-T} & 0 \\ k\boldsymbol{J}_A^{-1}\boldsymbol{S}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \right.$$
$$\left. - \begin{bmatrix} \boldsymbol{B}_m & 0 & 0 \\ 0 & k_{A2}\boldsymbol{J}_A^{-1}\boldsymbol{Z}\boldsymbol{J}_A^{-T} & 0 \\ 0 & 0 & \boldsymbol{P} \end{bmatrix} \right) \begin{bmatrix} \nabla_{\boldsymbol{p}_m}\mathcal{V} \\ \nabla_{\boldsymbol{q}_A}\mathcal{V} \\ \nabla_{\boldsymbol{q}_B}\mathcal{V} \end{bmatrix} + \begin{bmatrix} \mathbf{1} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{f}_h, \tag{23}$$

where $\boldsymbol{P} = \left(\boldsymbol{I} - (\boldsymbol{L}_s\boldsymbol{J}_B)^\dagger(\boldsymbol{L}_s\boldsymbol{J}_B)\right)$ is a null-space projector such that $\boldsymbol{P} = \boldsymbol{P}^T \geq 0$. Therefore, since in all cases the closed-loop systems can be put in a PHS form (being
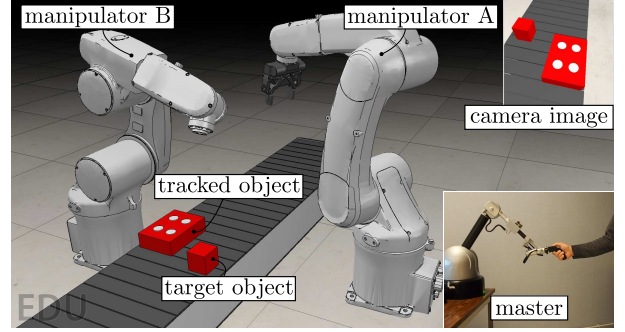

Fig. 4. Experimental setup used for user study: slave side composed of simulated robots and vision; master side comprising the haptic interface.

the resulting interconnection and dissipation matrixes always skew-symmetric and positive semi-definite, respectively), one can conclude passivity of the three modalities w.r.t. the pair $(\boldsymbol{u}_m, \boldsymbol{f}_h)$ with energy function $\mathcal{V}(\boldsymbol{p}_m, \boldsymbol{q}_A, \boldsymbol{q}_B)$ as sought[2].

IV. EXPERIMENTS IN SIMULATION

A. Experimental setup, task, and participants

The experimental setup is shown in Fig. 4. The slave side is simulated using V-REP, and it is composed of our two velocity-controlled manipulators: an Adept Viper 850, controlled by the human operator (manipulator A), and an Adept Viper 650, controlled by an autonomous algorithm (manipulator B). The master side is composed of an Haption Virtuose 6D Desktop interface, which is used to control the Viper 850 robot and provides kinesthetic haptic feedback. Manipulator A is endowed with a ROBOTIQ 2-finger gripper, while manipulator B is equipped with a vision sensor. The remote environment is composed of two objects, a cube and a rectangular parallelepiped, placed on a conveyor belt. As detailed in Sec. II, the autonomous manipulator B robot is in charge of tracking the parallelepiped using standard visual servoing techniques [30]. On the other hand, participants are required to control the motion of manipulator A to grasp the cube and lift it from the ground, avoiding collisions with the fully-autonomous robot. The task starts when the manipulator moves for the very first time, and it is considered successfully completed when the object is lifted from the ground. All the control policies described are implemented in MATLAB/SIMULINK and interfaced with V-REP using ROS through the matlab_ros_bridge (https://github.com/lagadic/matlab_ros_bridge). The control loop runs at 100 Hz. A video of the experiment is available as supplemental material and at https://youtu.be/uWzVEGW2i5c.

Fifteen right-handed subjects (average age 25.9, 11 males, 4 females) participated in the study. Four of them had previous experience with haptic interfaces. Each subject spent about ten minutes practicing the control of the telemanipulation

---

[2]We note that the presented analysis does not account for the controller switching in the shared-control modality when close/far from the target object. If this switch needs to be taken into account because of non-negligible effects on the total energy, one could employ the energy tank machinery for passifying potential instabilities due to the switching mechanism [29]. However, we empirically found the switching to have a negligible effect on the system stability during our experiments.

system before starting the experiment. Participants were briefed about all the tasks and afterwards signed an informed consent, including the declaration of having no conflict of interest. All of them were able to give the consent autonomously. The study was done in accordance with the 1964 Helsinki declaration and its later amendments.

B. Experimental conditions

We considered two different levels of human involvement in the control of manipulator A (teleoperation vs. shared control, T vs. S), two haptic feedback modalities (haptic feedback about the distance from the second robot and workspace/joints constraints vs. no haptic feedback, H vs. $\cancel{H}$), and two control policies for manipulator B (reactive vs. non-reactive, R vs. $\cancel{R}$).

1) Human involvement (T vs. S): In condition T, subjects are able to control all the 6 DoF of manipulator A through the grounded master interface (see Sec. II-C1). Conversely, in condition S, the orientation of the gripper is controlled by an autonomous algorithm. When the gripper is close to the object to grasp, it is automatically oriented towards it; otherwise, the gripper is oriented to stay as much as possible away from the system constraints (see Sec. II-C2).

2) Haptic feedback (H vs. $\cancel{H}$): When haptic feedback is activated (H), subjects receive haptic stimuli about the feasibility of their commands against system constraints, such as joint limits, singular configurations, and collisions with manipulator B (see Sec. II-D). Conversely, in condition $\cancel{H}$, subjects do not receive any haptic feedback.

3) Control of the fully-autonomous manipulator (R vs. $\cancel{R}$): Manipulator B always performs an autonomous visual task to keep an object of interest visible. In condition R, it also uses the null-space of the above-mentioned primary visual task to minimize the constraint cost function $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$ (see Sec. II-C). Notably, this secondary action will enable manipulator B to keep a safe distance from manipulator A, re-actively moving away when the latter comes too close. Conversely, in $\cancel{R}$, manipulator B only focuses on the primary visual task, indifferent to what the other manipulator does.

Considering all the possible combinations, we ended up with eight different experimental conditions: THR, TH$\cancel{R}$, T$\cancel{H}$R, T$\cancel{H}\cancel{R}$, SHR, SH$\cancel{R}$, S$\cancel{H}$R, S$\cancel{H}\cancel{R}$. For brevity, from now on we will omit the $\cancel{H}$ and $\cancel{R}$ variables (e.g., S$\cancel{H}\cancel{R}$ is called S).

The cost functions introduced in Sec. II include several parameters, which let us control the curvature, rate of increase, and proximity to the limits of these functions. The choice of these parameters is challenging, system-dependent, and very delicate, as it must ensure a smooth force feedback. To choose the right parameters for our system and target application, we asked 2 expert operators to repeatedly carry out the task, changing the parameters at runtime (i.e., $\rho_X, \alpha_X, \beta_X, k_{A1}, k_{A2}, k_{B1}, k_{B2}, k_M$) to make the teleoperation as intuitive, safe, and comfortable as possible. Finally, we asked them to find a consensus on the parameters' values and we used those in our implementation.

C. Results

Average task success rate across conditions was 92.4 $\pm$ 6.3%. A Friedman test showed no statistically significant
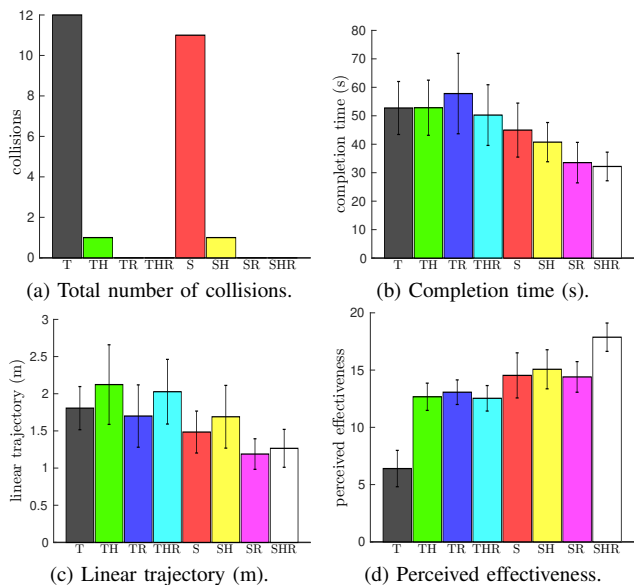

Fig. 5. Experimental evaluation. Mean and 95% confidence interval of (a) total number of collisions, (b) completion time, (c) linear motion, and (d) perceived effectiveness of the eight feedback conditions are plotted.

difference between the means of the eight feedback conditions. Figure 5a shows the total number of collisions occurred during the experiment between the two manipulators.

To compare other metrics, we ran three-way repeated-measures ANOVA tests on the data. Human involvement in the control (shared control vs. teleoperation, S vs. T), presence of haptic feedback (haptic feedback vs. no haptic feedback, H vs. $\cancel{H}$), and behavior of the fully-autonomous robot (reactive vs. non-reactive, R vs. $\cancel{R}$) were treated as within-subject factors. All data passed the Shapiro-Wilk normality test. Sphericity was assumed for all variables, since they all have two levels of repeated measures. Interaction effects between the factors were not statistically significant. Figure 5b shows the completion time. The three-way ANOVA test revealed a statistically significant change in the metric for the human involvement in the control (F(1, 14) = 52.165, $p < 0.001$, shared control was better) and the control behavior for manipulator B (F(1, 14) = 6.400, $p = 0.024$, reactive mode was better) variables. Figure 5c shows the linear motion covered by the robotic gripper during the task. The three-way ANOVA test revealed a statistically significant change in the metric for the human involvement in the control (F(1, 14) = 13.599, $p = 0.002$, shared control was better) and the control behavior for the second robot (F(1, 14) = 6.567, $p = 0.023$, reactive mode was better) variables. Immediately after the experiment, subjects were also asked to report the effectiveness of each feedback condition in completing the given task using bipolar Likert-type twenty-two-point scales. Figure 5d shows the perceived effectiveness for the eight experimental conditions. The three-way ANOVA test revealed a statistically significant change in the metric for the human involvement in the control (F(1, 14) = 34.700, $p < 0.001$), the presence of haptic feedback (F(1, 14) = 33.217, $p < 0.001$, shared control was better), and the control behavior for the second robot (F(1, 14) = 25.305, $p < 0.001$, reactive mode was better) variables. Finally, all fifteen subjects found conditions using the shared-control approach to be the
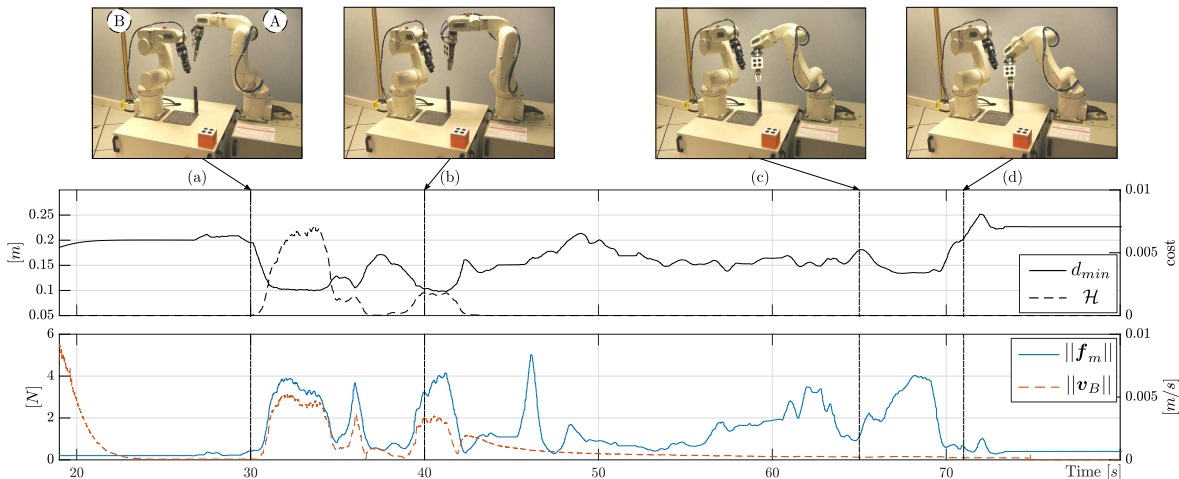
Fig. 6. Data time history recorded during the real experiments. Upper graph: minimum distance $d_{\min}$ and collision cost value $\mathcal{H}$; bottom graph: haptic guidance $\boldsymbol{f}_m$ and escaping velocity $\boldsymbol{v}_B$ norms.

most effective at completing the grasping task. Ten subjects chose SHR as the most effective, three SH, and two SR.

V. EXPERIMENT IN A REAL ENVIRONMENT

We also carried out an experiment in the real environment. The setup and task are similar to the simulated scenario of Sec. IV. The setup is shown in Fig. 1. Five right-handed subjects participated in the study. Three of them had previous experience with haptic interfaces. In this real-world case, we implemented control modalities TR, THR, SR, and SHR, which led to no collisions in Sec. IV. Of course, in this real-world case, we could not consider conditions which may lead to a collision between the two manipulators. All subjects successfully completed the task in all conditions and no collisions occurred. All subjects chose SHR to be the most intuitive and effective condition. A video of the experiment in the SHR condition (shared control, haptic feedback, and reactive control of manipulator B) is available as supplemental material and at https://youtu.be/uWzVEGW2i5c. A sequence of frames taken from the video is shown in Fig. 6, where it is possible to recognize all the relevant phases of the grasping task. In snapshots (a)-(b), it is possible to see the manipulator B reactively avoiding collisions while keeping track of the desired object. In snapshots (c)-(d), the shared-control algorithm is active and automatically orients the gripper of manipulator A toward the object to grasp. From the bottom graphs, we can see that the first half of the experiment is characterized by persistent haptic forces and by a significant reactive velocity of manipulator B. In the second half, the risk of collisions is lower and the operator can safely approach the object, aided by the action of the shared-control algorithm.

VI. DISCUSSION AND CONCLUSION

We presented a shared-control framework with haptic feedback, which enables the intuitive and effective teleoperation of a dual-arm robotic system. We tested the proposed architecture in simulated and real environments composed of two 6-DoF robotic manipulators. The human operator controls one arm (manipulator A), equipped with a gripper, through a 6-DoF

grounded haptic interface. Haptic feedback provides the operator with information about joint and workspace limits as well as about the presence of singularity configurations and imminent collisions. The shared-control algorithm autonomously controls 2-DoF of the robotic manipulator, orienting the gripper toward the object to grasp. The other robotic arm (manipulator B) is equipped with a camera, and it moves autonomously to track a second object, placed near the one to grasp.

To demonstrate the feasibility and effectiveness of our methods, we conducted a human subject experiment in the simulated scenario. Fifteen subjects controlled manipulator A to grasp an object placed on a conveyor belt. We tested eight experimental conditions, considering two levels of human involvement in the control (shared control vs. classic teleoperation, S vs. T), two feedback modalities (haptic feedback about imminent collisions and workspace/joints constraints vs. no haptic feedback, H vs. $\cancel{H}$), and two control policies for manipulator B (reactively moving away from the other manipulator vs. non-reactive/still, R vs. $\cancel{R}$). Results proved the effectiveness and viability of our haptic-enabled shared-control approaches. Using shared control (S) on manipulator A and the reactive mode (R) on manipulator B significantly improved the performance in most metrics (completion time, linear motion, perceived effectiveness). Conditions employing shared control were also the most preferred, confirming the all-round viability of such approach. Moreover, as expected, in conditions R, manipulator B was always able to prevent collisions with the other robot by moving away when the latter was approaching. Nonetheless, even in conditions $\cancel{R}$, repulsive haptic feedback (H) provided when the robots were too close showed good results (only two collisions happened in conditions TH, SH). This result is very promising, as haptic feedback acts only at the master side, leaving the action of manipulator B unaffected. In this respect, it is interesting to notice that, although the applied force $\boldsymbol{f}_m$ should go to infinite as the distance between the two robots goes to zero (see eq. (5)), we still experienced two collisions in conditions H. This is due to the limited actuation capabilities of our haptic interface, which is obviously not able to provide arbitrarily high forces.

Finally, the experiment in the real scenario confirmed the results obtained with the simulated setup. Haptic guidance effectively steered the user toward the safe zones of the workspace, the reactive behavior enabled a safe interaction between manipulators, and the shared control made the task fast, easy, and intuitive to complete.

The proposed shared-control architecture is independent from the distribution of DoF between the autonomous controller and the human operator. While having the autonomous controller keep the gripper oriented toward the object seemed a natural choice for this work, the underlying shared-control architecture is independent from the task, the primary task variables, and the level of human involvement. For the future, we are planning to study how the number of DoF controlled autonomously affects the task performance, e.g., a system could use a highly-autonomous shared-control approach (i.e., many DoF managed by the autonomous controller) when it is operated by novices, while it could implement a lowly-autonomous shared-control approach (i.e., few DoF managed by the autonomous controller) when it is operated by experts. This flexible approach could be also useful when teaching new operators, employing different levels of autonomy according to the operator's experience. As future work, we will also investigate the effects of employing different cost functions and/or control gains on task performance. Finally, we plan to carry out more human subject experiments in real scenarios.

APPENDIX

We wish to prove that, in the shared-control modality when far from the object to be grasped, the slave angular velocity $\boldsymbol{\omega}_A$ in (13) is able to minimize the constraint cost function $\mathcal{H}(\boldsymbol{q}_A, \boldsymbol{q}_B)$ in free-motion ($\boldsymbol{v}_m = \boldsymbol{0}$). This can be easily shown as follows: by considering variations w.r.t. $\boldsymbol{q}_A$ ($\boldsymbol{q}_B$ is taken care of by (7)), one has $\dot{\mathcal{H}} = \frac{\partial^T \mathcal{H}}{\partial \boldsymbol{q}_A} \dot{\boldsymbol{q}}_A = \frac{\partial^T \mathcal{H}}{\partial \boldsymbol{q}_A} \boldsymbol{J}_A^\dagger \boldsymbol{u}_A$. By applying (13) when $\boldsymbol{v}_m = \boldsymbol{0}$, one has

$$\dot{\mathcal{H}} = -k_{A2} \frac{\partial^T \mathcal{H}}{\partial \boldsymbol{q}_A} \boldsymbol{J}_A^\dagger \boldsymbol{Z} (\boldsymbol{J}_A^\dagger)^T \frac{\partial \mathcal{H}}{\partial \boldsymbol{q}_A},$$

which is a negative semi-definite quadratic form, therefore resulting in $\dot{\mathcal{H}} \le 0$ as claimed.

REFERENCES

[1] C. Freschi, V. Ferrari, F. Melfi, M. Ferrari, F. Mosca, and A. Cuschieri, "Technical review of the da vinci surgical telemanipulator," *Int J Med Robot*, vol. 9, no. 4, pp. 396–406, 2013.

[2] F. Abi-Farraj, N. Pedemonte, and P. Robuffo Giordano, "A visual-based shared control architecture for remote telemanipulation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 4266–4273.

[3] A. M. G. Esfahani, F. Abi-Farraj, P. R. Giordano, and R. Stolkin, "Human-in-the-loop optimisation: mixed initiative grasping for optimally facilitating post-grasp manipulative actions," *arXiv:1707.08147*, 2017.

[4] D. Aschenbrenner, M. Fritscher, F. Sittner, M. Krauß, and K. Schilling, *Teleoperation of an Industrial Robot in an Active Production Line*, 2015.

[5] C. Pacchierotti, F. Ongaro, F. van den Brink, C. Yoon, D. Prattichizzo, D. H. Gracias, and S. Misra, "Steering and control of miniaturized untethered soft magnetic grippers with haptic assistance," *IEEE Trans. Automation Science and Engineering*, vol. 15, no. 1, pp. 290–306, 2018.

[6] C. Pacchierotti, S. Scheggi, D. Prattichizzo, and S. Misra, "Haptic feedback for microrobotics applications: a review," *Frontiers in Robotics and AI*, vol. 3, no. 53, 2016.

[7] P. LeBel, C. Gosselin, and A. Campeau-Lecours, "An anticipative kinematic limitation avoidance algorithm for collaborative robots: Three-dimensional case," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2017, pp. 3075–3080.

[8] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *IEEE Int. Conf. on Robotics and Automation*, vol. 2, 1985, pp. 500–505.

[9] A. D. Santis, A. Albu-Schaffer, C. Ott, B. Siciliano, and G. Hirzinger, "The skeleton algorithm for self-collision avoidance of a humanoid manipulator," in *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, 2007, pp. 1–6.

[10] M. Selvaggio, G. Notomista, F. Chen, B. Gao, F. Trapani, and D. Caldwell, "Enhancing bilateral teleoperation using camera-based online virtual fixtures generation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 1483–1488.

[11] M. Selvaggio, F. Chen, B. Gao, G. Notomista, F. Trapani, and D. Caldwell, "Vision based virtual fixture generation for teleoperated robotic manipulation," in *IEEE Int. Conf. on Advanced Robotics and Mechatronics*, 2016, pp. 190–195.

[12] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, "Integration of reactive, torque-based self-collision avoidance into a task hierarchy," *IEEE Trans. Robot.*, vol. 28, no. 6, pp. 1278–1293, 2012.

[13] C. Dube, M. Tsoeu, and J. Tapson, "A model of the humanoid body for self collision detection based on elliptical capsules," in *IEEE Int. Conf. on Robotics and Biomimetics*, 2011, pp. 2397–2402.

[14] K. Steinbach, J. Kuffner, T. Asfour, and R. Dillmann, "Efficient collision and self-collision detection for humanoids based on sphere trees hierarchies," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2006, pp. 560–566.

[15] F. Seto, K. Kosuge, and Y. Hirata, "Self-collision avoidance motion control for human robot cooperation system using robe," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2005, pp. 3143–3148.

[16] B. Dariush, G. B. Hammam, and D. Orin, "Constrained resolved acceleration control for humanoids," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 710–717.

[17] T. F. Chan and R. V. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant manipulators," in *IEEE Int. Conf. on Robotics and Automation*, 1993, pp. 395–402.

[18] M. Selvaggio, S. Grazioso, G. Notomista, and F. Chen, "Towards a self-collision aware teleoperation framework for compound robots," in *IEEE World Haptics Conference*, 2017, pp. 460–465.

[19] H. Saeidi, J. R. Wagner, and Y. Wang, "A mixed-initiative haptic teleoperation strategy for mobile robotic systems based on bidirectional computational trust analysis," *IEEE Trans. Robot.*, vol. 33, no. 6, pp. 1500–1507, 2017.

[20] H. Boessenkool, D. A. Abbink, C. J. M. Heemskerk, F. C. T. van der Helm, and J. G. W. Wildenbeest, "A task-specific analysis of the benefit of haptic shared control during telemanipulation," *IEEE Trans. Haptics*, vol. 6, no. 1, pp. 2–12, 2013.

[21] B. Khademian and K. Hashtrudi-Zaad, "Shared control architectures for haptic training: Performance and coupled stability analysis," *Int J Rob Res*, vol. 30, no. 13, pp. 1627–1642, 2011.

[22] D. Nicolis, M. Palumbo, A. M. Zanchettin, and P. Rocco, "Occlusion-free visual servoing for the shared autonomy teleoperation of dual-arm robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 796–803, 2018.

[23] N. Pedemonte, F. Abi-Farraj, and P. R. Giordano, "Visual-based shared control for remote telemanipulation with integral haptic feedback," in *IEEE Int. Conf. Robotics and Automation*, 2017, pp. 5342–5349.

[24] B. Dariush, G. B. Hammam, and D. Orin, "Constrained resolved acceleration control for humanoids," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 710–717.

[25] J. Cascio, M. Karpenko, Q. Gong, P. Sekhavat, and I. M. Ross, "Smooth proximity computation for collision-free optimal control of multiple robotic manipulators," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 2452–2457.

[26] F. Chaumette and S. Hutchinson, "Visual servo control, part i: Basic approaches," *IEEE Robot.Autom.Mag.*, vol. 13, no. 4, pp. 82–90, 2006.

[27] B. Siciliano and J. J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Fifth Int. Conf. on Advanced Robotics*, 1991, pp. 1211–1216 vol.2.

[28] D. Lee and M. W. Spong, "Passive bilateral teleoperation with constant time delay," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 269–281, 2006.

[29] C. Secchi, A. Franchi, H. H. Bülthoff, and P. Robuffo Giordano, "Bilateral teleoperation of a group of uavs with communication delays and switching topology," in *IEEE Int. Conf. on Robotics and Automation*, 2012, pp. 4307–4314.

[30] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robot. Autom. Mag.*, vol. 12, no. 4, pp. 40–52, 2005.