Neuromuscular Interfacing for Advancing Kinesthetic and Teleoperated Programming by Demonstration of Collaborative Robots

Roberto Meattini, *Member, IEEE*, Armando Amerì, Alessandra Bernardini, *Graduate Student Member, IEEE*, Javier Gonzalez-Huarte, Aitor Ibarguren, Claudio Melchiorri, *Fellow, IEEE*, and Gianluca Palli, *Senior Member, IEEE*

***Abstract*—This study addresses the challenges of Programming by Demonstration (PbD) in the context of collaborative robots, focusing on the need to provide additional degrees of programming without hindering the user's ability to demonstrate trajectories. The study proposes the use of a wearable human-robot interface based on surface Electromyography (sEMG) to measure the forearm's muscle co-contraction level, enabling additional programming inputs through hand stiffening level modulations without interfering with voluntary movements. Vibrotactile feedback enhances the operator's understanding of the additional programming inputs during PbD tasks. The proposed approach is demonstrated through experiments involving a collaborative robot performing an industrial wiring task. The results showcase the effectiveness and intuitiveness of the interface, allowing simultaneous programming of robot compliance and gripper grasping. The framework, applicable to both teleoperation and kinesthetic teaching, demonstrated effectively in an industrial wiring task with a 100% success rate over the group of subjects. Furthermore, the presence of vibortactile feedback showed an average decrease of programming errors of 33%, and statistical analyses confirmed the subjects' ability to correctly modulate co-contraction levels. This innovative framework augments programming by demonstration by integrating neuromuscular interfacing and introducing structured programming logics, providing an intuitive human-robot interaction for programming both gripper and compliance in teleoperation and kinesthetic teaching.**

***Index Terms*—Programming by demonstration, robotic wiring, advanced human-robot interfaces.**

Received 29 July 2023; revised 22 December 2023 and 23 April 2024; accepted 14 October 2024. Date of publication 21 October 2024; date of current version 21 March 2025. This work was supported in part by European Commission's Horizon 2020 Programme with the Project REMODEL under Grant 870133 and in part by MUR with the Project "Sustainable Mobility Center" under Code CN00000023-CUP J33C22001120001. This article was recommended for publication by Associate Editor G. Berselli and Editor-in-Chief D. Prattichizzo upon evaluation of the reviewers' comments. *(Corresponding author: Roberto Meattini.)*

This work involved human subjects or animals in its research. The author(s) confirm(s) that all human/animal subject research procedures and protocols are exempt from review board approval.

Roberto Meattini, Armando Amerì, Alessandra Bernardini, Claudio Melchiorri, and Gianluca Palli are with the Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi," University of Bologna, 40136 Bologna, Italy (e-mail: roberto.meattini@unibo.it).

Javier Gonzalez-Huarte and Aitor Ibarguren are with the TECNALIA, Basque Research and Technology Alliance (BRTA) 20009 Donostia-San Sebastián, Spain (e-mail: aitor.ibarguren@tecnalia.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/TOH.2024.3484373, provided by the authors.

Digital Object Identifier 10.1109/TOH.2024.3484373

I. Introduction

Robotics is currently going through a crucial and exciting process of development due to the advent of collaborative robots [1]. Accordingly, one of the most important challenges regards the programming paradigm [2]. PbD approaches can be categorized [3] in *observational PbD* and *kinesthetic PbD*. Note that the first kind of approach requires a mapping of the human inputs to the robot trajectory [4]. On the other hand, the kinesthetic PbD approach consists in the user literally grabbing the robot in order to physically guide the end-effector through the desired trajectory, and therefore in this way providing the direct demonstration of the robot behaviour [5]. Existing literature on PbD mainly focuses on single modalities for robot trajectory demonstrations: observational [6] and kinesthetic PbD [7]. Our challenge is to provide additional programming degrees of freedom for collaborative functionalities while demonstrating robot trajectories using a composition of both teleoperated and kinesthetic modalities without compromising any PbD capability.

A. Enhancing PbD With Neuromuscular Interfacing

Our approach leverages Electromyography (sEMG) measurements of forearm's co-contraction level (CC-level), modulated by the user as an additional programming input via hand stiffening level changes. To assist users in this modulation, a vibrotactile feedback through a wearable coin motor is provided, regulating vibration intensity in real-time. Fig. 1 illustrates the proposed programming concept. Various studies have explored the use of neuromuscular interfacing to transfer motion skills from humans to robots. For instance, [8] allows users to control robot stiffness while teaching position trajectory through a H-R interface. In [9], the concept of teleimpedance involves sending a reference command with both desired motion and impedance profile from the human operator to the robot, using surface sEMG measurements. In [10], a robot PbD framework integrates teleimpedance for teaching robot stiffness with trajectory generalization based on a single human demonstration. [11] and [12] leveraged sEMG signals and kinesthetic teaching for PbD tasks, recording positions and impedance gains from muscle activity for offline application to the robot.Unlike these previous works, our framework uses sEMG to estimate the user's overall hand stiffness from

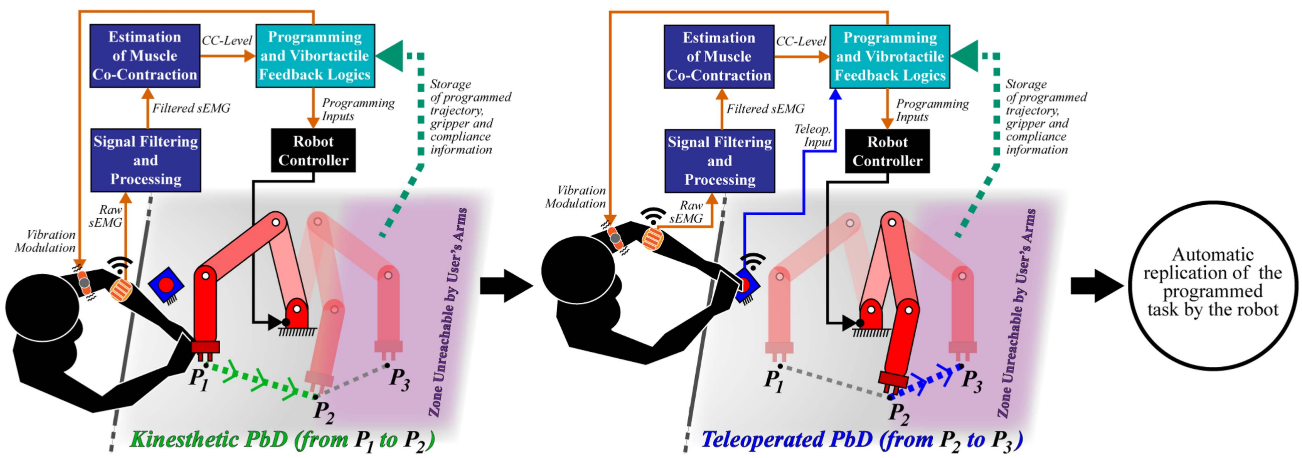


Fig. 1. Exemplifying scheme of the proposed combined kinesthetic-teleoperated PbD enhanced by neuromuscualr interfacing concept. From left to right: kinesthetic PbD + gripper/compliance programming; teleoperated PbD + gripper/compliance programming; automatic replication of the programmed task.

forearm muscles, instead of an estimation of the arm end-point impedance. This allows to provide an additional programming input for the user that can be freely modulated. Our previous works also dealt with exploiting sEMG signals to enhance intuitive programming of robotic tasks, i.e. [13], [14], [15]. In these works, we laid the foundation for incorporating sEMG signals into kinesthetic teaching. However, it is crucial to highlight the evolution of our research, leading to the current paper which represents a significant advancement beyond our prior studies. While our earlier research focused primarily on kinesthetic teaching, the current study introduces a more generalized PbD framework. This framework is applicable to both teleoperation and kinesthetic teaching modalities, offering several key novel contributions: *i)* introduction of a unified framework for both teleoperation and kinesthetic teaching within PbD; *ii)* capability to program both gripper and robot compliance levels using sEMG within a single PbD solution; *iii)* implementation of an effective Finite State Machine (FSM) coupled with vibrotactile feedback to enhance user awareness, facilitating the proper programming of gripper and robot compliance simultaneously to trajectory programming. Our current work not only broadens the scope of application for sEMG but also introduces structured programming logics and enhanced vibrotactile feedback for a more intuitive human-robot interaction. This innovation allows for the programming of both gripper and compliance for both teleoperation and kinesthetic teaching, a capability not explored in our earlier works.

II. Methods

A. General Framework for Combining Kinesthetic and Teleoperation PbD, Enhanced by Neuromuscular H-R Interfacing

Fig. 1 reports an exemplifying scheme of the general framework we propose in this study, and that will be thoroughly described in the following subsections of this section. In particular, referring to Fig. 1, we are considering a PbD scenario in which a user have to exploit both kinesthetic and teleoperated demonstration modalities for different parts of the robot trajectory to be programmed. In the left-hand side of Fig. 1 it is depicted a first phase of an exemplifying PbD task, in which the user starts programming the robot trajectory by means of the kineshetic teaching modality. Then, the right-hand side of the figure represents a second possible phase of the PbD task, in which the user, in order to continue performing the trajectory programming, switches to the teleoperation PbD modality, for the reason that the trajectory to be programmed belongs to a region of the workspace that is not reachable by the user's limbs. During these two phases, sEMG signals are properly acquired and processed, vibrotactile feedback is conveyed to the user, and the programming logics acts for allowing programming of gripper and robot compliance during the trajectory teaching – as detailed in the following. The robot control approach to realize these different types of PbD methods, as well as the automatic replication by the robot of the programmed task (see Fig. 1), is described in Section II-B.

B. Robot Control for Combining Kinesthetic and Teleoperated PbD

Let us consider the Euler-Lagrange dynamic model of a collaborative robot manipulator with f degrees of freedom (DoF) [16]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = F_c + J^T(q)F_e, \qquad (1)$$

where $q \in \mathbb{R}^f$ is the vector of joint coordinates, $F_c \in \mathbb{R}^f$ is the input control torque vector, $F_e \in \mathbb{R}^6$ is the external wrench applied to the end-effector by either the environment or a human operator, $J(q)$ is the robot Jacobian, and $M(q) \in \mathbb{R}^{f\times f}$, $C(q,\dot{q}) \in \mathbb{R}^{f\times f}$ and $g(q) \in \mathbb{R}^f$ are the inertia matrix, Coriolis and centrifugal effect matrix, and gravitational term, respectively. The control input F_c is enforced as

$$F_c = \widehat{g}(q) + J_A^T(q)K_C^{-1}\widetilde{x} - J_A^T K_D J_A(q)\dot{q}, \qquad (2)$$

where $\widehat{g}(q)$ is the estimate of the gravity term, $J_A(q) = T_A(q)^{-1}J(q)$ is the analytic robot Jacobian [16] with T_A the tranformation matrix from analytic to geometric Jacobian [16], $\widetilde{x} = x_d - x$ with $x, x_d \in \mathbb{R}^6$ the actual and desired end-effector pose defined in workspace coordinates (i.e. position and Euler angles), and $K_C, K_D \in \mathbb{R}^{f\times f}$ are diagonal positive definite matrices. In particular, K_C is the *compliance matrix*, which allows to define a robot compliant behaviour in desired operational space directions by enforcing higher gain values on the corresponding elements on the diagonal. Being the control law of (2) a well-known proportional-derivative control with online gravity compensation (PD + gravity compensation) exploited to realize an operational space compliance control, in absence of external forces the controlled system is asymptotically stable with respect to a desired reference x_d. On the other hand, in presence of external forces due to the robot interacting with the environment, by substituting (2) in (1) it is immediate to verify that in steady-state conditions we have that

$$\widetilde{x} = -K_C T_A^T F_e = -K_C F_A, \tag{3}$$

where $F_A := T_A^T F_e$. The parameters are specialized as follow:

$$K_C = \begin{cases} \mathbf{0_{f\times f}}, & \text{for kinesthetic PbD} \\ \text{diag}(k_{C1,\text{min}}, \dots, k_{Cf,\text{min}}), & \text{for teleoperated PbD} \\ \text{diag}(k_{C1}, \dots, k_{Cf}), & \text{for trajectory replication} \end{cases},$$
$$K_D = \begin{cases} \mathbf{0_{f\times f}}, & \text{for kinesthetic PbD} \\ \text{diag}(\bar{k}_{D1}, \dots, \bar{k}_{Df}), & \text{for teleoperated PbD} \\ \text{diag}(\bar{k}_{D1}, \dots, \bar{k}_{Df}), & \text{for trajectory replication} \end{cases}, \tag{4}$$

where $k_{Ci} = \{k_{Ci,\text{min}}, k_{Ci,\text{max}}\}$ $(i = 1, \dots, f)$ is based on the compliance level programming that will be detailed in the following Section II-C1, with $k_{Ci,\text{min}}$ and $k_{Ci,\text{max}}$ are proper minimum and maximum compliance values, respectively, and $\bar{k}_{Di}$ $(i = 1, \dots, f)$ is a constant gain properly selected to obtain a desired convergence rate. Eventually, note that the selection of K_C and K_D for the *kinesthetic PbD* corresponds to a gravity compensation control, for the *teleoperated PbD* corresponds to compliance control with high values of stiffness to allow a more precise control of the robot by the operator, and for the *trajectory replication* is a compliance control with the possibility of a high compliance level, based on desired programming, in order to allow a smoother interaction of the robot with the environment (according to (2) and (1).) The actual implementation of the different robot compliance control modalities of (4) in the real experimental robotic setup will be reported in Section III-A1, realized thanks to a dedicated ROS-based software architecture.

C. sEMG-Based Co-Contraction Estimation for Additional Programming Capability During PbD

1) CC-Level Estimation Approach: We consider the human hand actuated by two predominant muscular extension and flexion actions driven by two high-level neural drives $h_E(t)$ and $h_F(t)$, respectively. It is therefore possible to define the hand CC-level $C_c(t)$ as

$$C_c(t) = \min(h_E(t), h_F(t)), \tag{5}$$

Let us consider that in general the complexity of the human hand musculoskeletal system can be modelled as d DoF actuated by a group of m extrinsic muscles:

$$D(t) = SH(t) = S \begin{bmatrix} h_E(t) \\ h_F(t) \end{bmatrix}, \tag{6}$$

where $H(t) \in \mathbb{R}^2$ is the neural drive vector, $S \in \mathbb{R}^{d\times 2}$ is the human hand kinematic synergy matrix, and $D \in \mathbb{R}^d$ is the vector of hand DoF. The motion described by (6) requires proper muscle activations, i.e.:

$$D(t) = MA(t) \;\Rightarrow\; A(t) = M^+ D(t), \tag{7}$$

where $M \in \mathbb{R}^{m\times d}$ is the muscular synergy matrix, and $A(t) = [a_1(t) \cdots a_m(t)]^T \in \mathbb{R}^m$ is the vector of muscle activations. According to [17], the root mean square (RMS) value of the raw L-dimensional sEMG signal $E(t) \in \mathbb{R}^L$ can be considered proportional to the muscle activation levels, that is

$$E(t) = VA(t). \tag{8}$$

where $V \in \mathbb{R}^{L\times m}$ is the muscle activation mixture matrix. Therefore, substituting (6) in (7), and then in (8), we can write

$$E(t) = VM^+ SH(t) = NH(t), \tag{9}$$

where $N = VM^+S \in \mathbb{R}^{L\times 2}$ is the neural drive mixture matrix. It follows that

$$H(t) = N^+ E(t), \tag{10}$$

where N is obtained as $N = KU$, in which $K \in \mathbb{R}^{2\times 2}$ is a diagonal positive definite scaling matrix such that $h_E(t), h_F(t) \in [0, 1]$ and $U \in \mathbb{R}^{L\times 2}$ is computed as detailed in the following. Therefore, finally, the online CC-level $C_c(t)$ can be estimated using the result of (10) in (5). Note that, according to (10), $C_c(t) \in [0, 1]$. Therefore, the influence of the processed sEMG signals in the overall control of the system regards the specification, for each sample of the programmed robot trajectory, of *i)* the level of robot compliance, in accordance with (4), and *ii)* the closing or opening of the gripper, which will be enforced online as provided during the offline programming phase, see also Section II-D.

2) Simulation Tests of the CC-Level Estimation: A simulator of synthetic sEMG data is used in this section to model the generation of forearm muscle sEMG signals during hand motions. The physiological process involves the recruitment of motor units based on an activation threshold [18]. The motor unit action potential (MUAP) is computed for a single rectangular electrode and further extended to a differential configuration for simulating sEMG signals [19]. The pulse train generated by each recruited motor unit is characterized by an Integral Pulse Frequency Modulation (IPFM) mechanism [19]. The overall sEMG signal for a group of muscles and differential sensors is computed by summing individual muscle contributions. The simulator accounts for the forearm's anatomy and muscle distribution, utilizing an optimization procedure to determine muscle activations corresponding to specific hand motions and co-contraction levels. The exploited simulator offers a comprehensive framework for generating synthetic sEMG data, exploiting modelling of motor unit recruitment, MUAP characteristics, and muscle

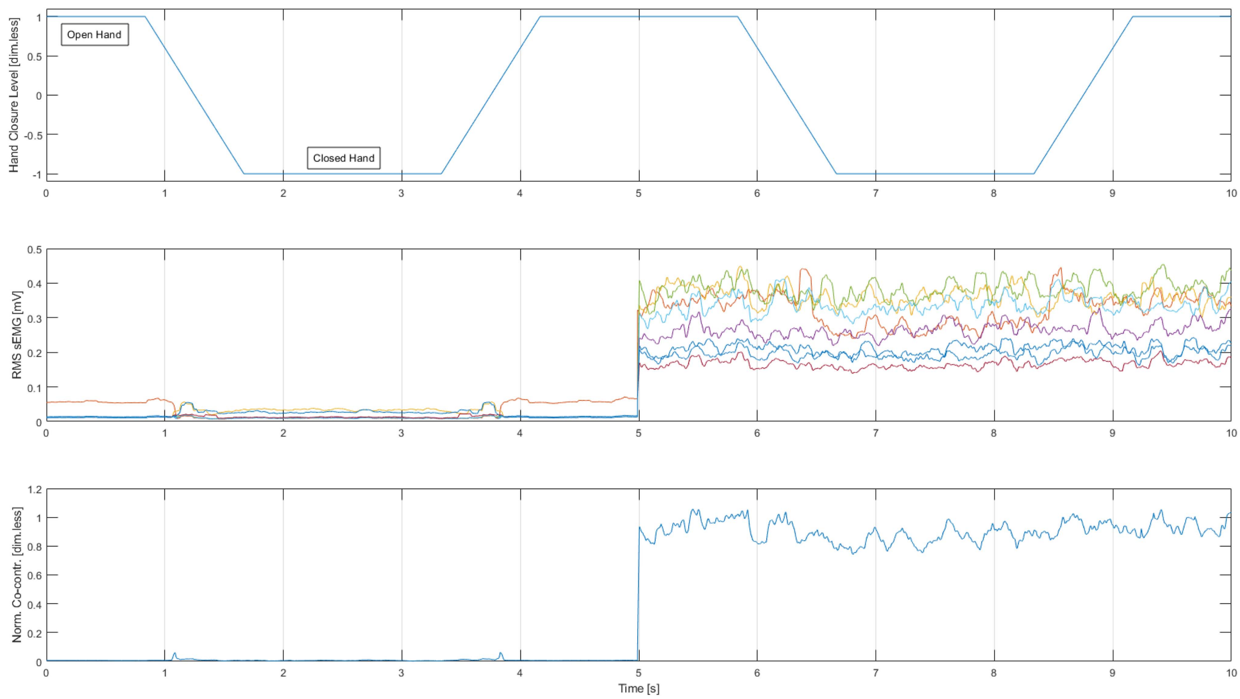


Fig. 2. Simulation of synthetic sEMG data. Graphs from top to bottom: hand closure level, RMS sEMG, estimated CC-level (refer to Section II-C2).

activation profiles during hand movements. The simulation of the forearm sEMG signals has been therefore carried out considering two consecutive open-close-open motions of the hand. In particular, the sEMG signals of the first and second motions have been simulated by imposing a minimum ($c_d = 0$) and maximum level of CC-level (c_d equal to the summation of the maximum activation levels of all considered muscles), respectively. The result of this simulation is reported in Fig. 2. As it can be clearly observed, in Fig. 2 the second motion starting at $t = 5s$ is characterized by higher values of the RMS sEMG, due to the imposition of a maximum CC-level, showing the typical behaviour of the forearm sEMG signal during maximum voluntary hand stiffening [20]. Accordingly, it can be also observed how in the bottom graph the estimated CC-level correctly resembled the CC-level imposed to the sEMG simulator, showing an almost zero estimated CC-level for the first hand motion, whereas a maximum estimated CC-level for the second hand motion.

D. H-R Communication During PbD: Programming and Vibrotactile Feedback Logics for Gripper and Compliance Level

1) Finite State Machine Logics: The logics for the programming of the gripper and compliance based on $C_c(t)$ can be described with a finite state machine (FSM), which is illustrated in Fig. 3, and it is based on a thresholding technique applied to the the CC-level. Specifically, given that $C_c(t) \in [0, 1]$ (as already introduced in (5)), and a threshold $\tau_{CC} = 0.5$, at each instant of time the implemented thresholding technique provides two different information:

i) if the CC-level have surpassed ($C_c(t) \geq \tau_{CC}$) or have moved below ($C_c(t) < \tau_{CC}$) the threshold τ_{CC};

ii) the time T_{thr} related to how along the condition $C_c(t) \geq \tau_{CC}$ or $C_c(t) < \tau_{CC}$ is continuously matched without interruption.

By looking at Fig. 3, the user starts from a *PbD state* (the one with the Gripper Open (CO) and the Compliance Low (CL).) In this state the user can freely perform kinesthetic or teleoperated PbD. Then, if the user wants to program gripper or compliance, it is necessary to enter in the *Programming Mode state*. To do this, the user has first to enter in the *Programming Request state* by modulating the CC-level in order to continuously surpass the threshold τ_{CC} for a time period longer than $T_{ACK} = 1$ s, and then to transition to the *Programming Mode state* by moving the CC-level below the threshold for a time period $T_{\text{thr}} \geq T_{ACK}$. At this point, if the user wants to program the closing of the gripper, it is necessary to surpass the threshold τ_{CC} for a time period lower than T_{ACK}, otherwise, if the the user wants to program a high compliance level, it is necessary to surpass the threshold τ_{CC} for a time period $T_{\text{thr}} \geq T_{ACK}$.

2) Vibrotactile Feedback Patterns: Let us consider a coin vibration motor placed on the upper arm skin of the user by means of a proper bracelet, which can be controlled in order to get a vibration at a constant frequency and intensity for a desired time duration. Three types of vibrotactile feedback patterns are considered: *i)* a continuous vibration of 1s (pattern FB#1); two consecutive single vibrations of 0.25s (FB#2); a single vibration of 0.25s (FB#3); a single vibration of 0.25s immediately followed by a continuous vibration of 1s (FB#4.) In particular, the vibrotactile feedback pattern FB#1 is provided to the user concurrently with the occurrence of the transition from the *PbD state* to the *Programming Request state*. The vibrotactile feedback pattern FB#2 is instead provided when the transition to the *Programming Mode state* occurs. Finally, the vibrotactile

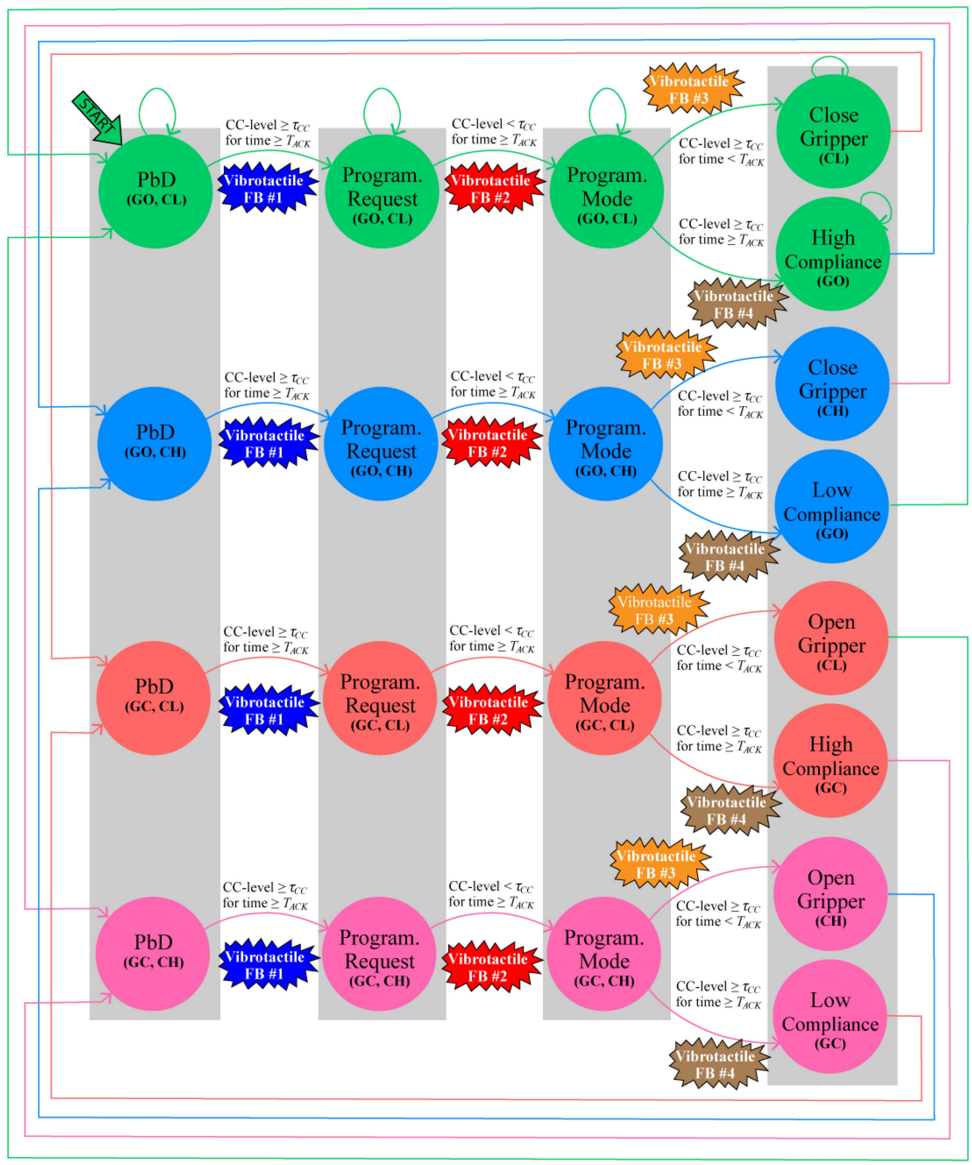


Fig. 3. FSM for the programming on gripper and robot compliance level during PbD via CC-level modulation and vibrotactile feedback provided to the operator. Notation is conform with Section II-D.

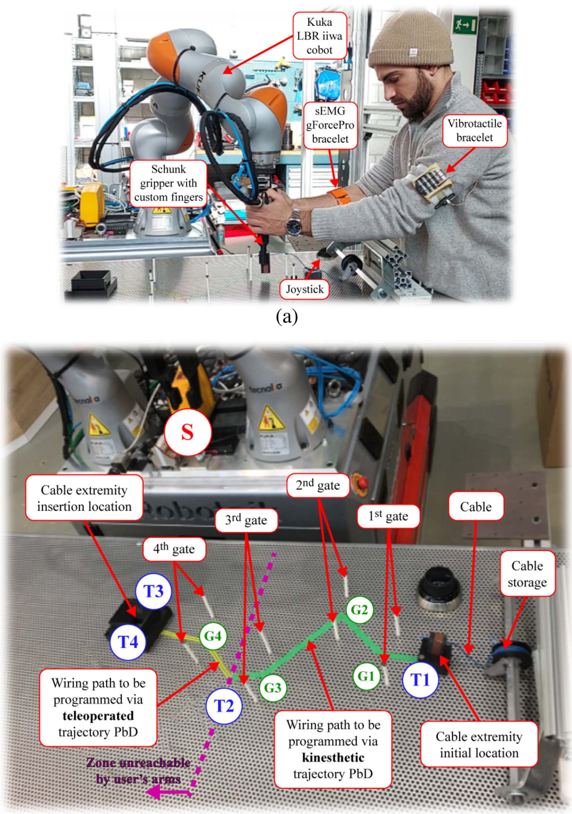
feedback patterns FB#3 and FB#4 are conveyed to the user if the programming of the gripper or of the compliance level is applied, respectively.

III. Experiments

A. Experimental Setup

1) Collaborative Robot Manipulator and Gripper: The collaborative robotic manipulator used in the experiment was a 7-DoF LBR iiwa by Kuka, as can be observed in Fig. 4(a). As it can be seen from Fig. 4(a), (b), the experimental scenario exactly reproduce the situation schematically depicted in Fig. 1, in which the operator cannot perform kinesthetic PbD for the entire task. The manipulator was equipped with a Schunk WSG50-110 gripper and a 3Dconnexion SpaceMouse Wireless for the teleoperation. The core of the software elements were implemented exploiting the Robot Operating System (ROS), expect for the KUKA Sunrise Bridge directly implemented in the Kuka LBR iiwa's inner controller. In the following, specific details are provided about the modules shown in Fig. 5:

- *Robot Programming And Execution Application:* This is the module in charge of recording and executing trajectories, in accordance with (4).
- *Custom Robot Driver:* The driver offers three different functionalities: *i) Gravity Compensation:* This capability activates and deactivates the gravity compensation for the Kuka LBR iiwa robot; *ii) Servo Control:* This capability implements a high-frequency (250Hz) control in joint space, including a controller able to convert Cartesian to joint velocities; *iii) Trajectory Execution:* This capability allows the reception and management of ROS trajectory messages [21] including trajectories with high or low compliance levels (see (4)).
- *WSG Driver:* The Schunk WSG50 ROS driver [22].



(b)

Fig. 4. (a) Experimental setup. (b) Experimental task and protocol.

- *Kuka Sunrise Bridge:* This module act as a bridge for the execution of the different capabilities available on the ROS side.

2) sEMG and Vibrotactile Bracelets: The sensing system used in the experiments for the acquisition of the sEMG signals was the gForcePro bracelet by OYMotion, an 8-channel sEMG armband that was placed on the users' forearm in proximity of the bellies of the *Flexor Digitorium Superficialis* and *Extensor Digitorium Communis muscles* (see Fig. 4(a)). The signal provided by the bracelet, and Bluetooth-streamed at 1kHz to a nearby computer, was a raw signal, processed by a standard filtering chain as reported in [13]. On the other hand, for the experiments, the vibrotactile feedback was provided by means of a custom-made bracelet placed on the upper arm's skin of the user in order to convey cutaneous vibration stimuli, see Fig. 4(a). The vibrotactile feedback, conveyed d by a coin vibration motor, was very important in order to the make the user aware of the programming state of the gripper and robot compliance level by means of the CC-level modulation,as explained in Section II-D.

3) Switchgear Test-Bed: The experiment was conducted using a switchgear test-bed that we developed in our laboratories in order to test the proposed programming framework, as can be observed in Fig. 4(b). Specifically, the test-bed was composed by: *i)* a cable storage reel from which it is possible to extract a cable by grasping and pulling its extremity; *ii)* metal pins used to define specific gates along the path on which the cable needs to be wired; and *iii)* a goal cable extremity insertion location. In particular, this test-bed was directly aligned with the challenges identified in existing manual manufacturing techniques for wiring harnesses, and specifically referring to the field of aerospace wiring harness manufacturing. Indeed in the typical aerospace industry scenarios, the wiring is executed on a horizontal tables, as for example occurs for helicopters and airplanes. In such scenarios, pins are very important as they constitute the direction for the wiring path. The main wiring operations performed are: cable grasping, routing and insertion in final locations for welding of extremities with contactors. Accordingly with this reference scenario of the test-bed, the cable used was a multi-conductor cable, which are among the most extensively used in aerospace wiring harness manufacturing, since they provide a compact and organized solution for transmitting multiple signals or power lines within a limited space, which is crucial in the confined environments of aircraft [23].

B. Experimental Task Protocol Description

1) Subjects: We conducted an experimentation involving a group of 10 healthy participants (age: 29.5 ± 2.1). Prior to the experiment, all participants were given detailed information about the study and provided their informed consent.

2) Programming Task - Robotic Wiring: The switchgear test-bed, as illustrated in Fig. 4(b), served as the platform for a cable routing and insertion task. First of all, in order to ensure the participants were well-acquainted with the system and the tasks at hand, a familiarization session of 15 minutes was conducted, providing them with the opportunity to practice with the experimental setup.

3) Instructions Provided to the Subjects: During the experiment, each participant was given a set of specific programming goals to realize, constituting on the teaching of a robot end-effector trajectory, using for the first part of the path the kinesthetic PbD modality, whereas for the second part – not reachable by users' arms (see Fig. 4(a), (b)) – the teleoperated PbD modality. At the same, also a proper programming of the gripper grasping and robot compliance level modulating the CC-level via hand stiffening and exploiting the vibrotactile feedback was required, as detailed in the following. By referring to the task locations reported in Fig. 4(b), the actions instructed to the subjects encompassed the following steps:

i) programming of the motion of the robot end-effector from the starting location S to the cable extremity initial location T1 via kinesthetic PbD;
ii) in T1, programming of the grasping of the cable extremity by using kinesthetic PbD for a fine positioning of the gripper in conjunction with the modulation of the CC-level (in accordance with Section II-D) in order to program the closure of the gripper;
iii) programming of the motion of the robot end-effector from T1 to T2 performing a cable routing through the gates G1, G2 and G3 via kinesthetic PbD;
iv) programming the motion of the robot end-effector from T2 to T3 performing a cable routing through the gate G4 via teleoperated PbD (workspace zone not reachable by subjects' arms);

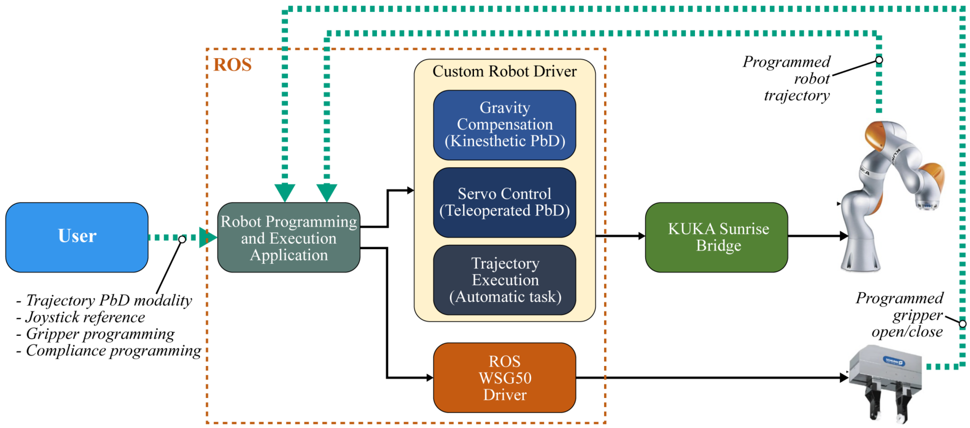


Fig. 5. Robot control framework implemented for the experiment. The dashed lines indicate information flow which is present only during the programming phase of the robot.

v) in T3, programming of the robot compliance level to "high compliance" by modulation of the CC-level, in order to let the robot be compliant during the exchange of forces that will take place for the cable insertion in the next step;

vi) programming of the insertion of the cable extremity within the insertion location at T4 via teleoperated PbD;

vii) programming of the gripper opening by modulation of the CC-level, in order to release the cable extremity.

4) Online Execution of the Task: Finally, as conclusive phase of the experimentation, the programmed task was automatically performed by the robot two times: the first one, in the same setup conditions of the programming phase, whereas the second one by introducing an artificial displacement of 5 mm to the insertion location T4, by manually displacing the related component shown in Fig. 4, in order to test the adaptation of the robot to the new situation thanks to programmed high compliance level. Note that the subjects were instructed to employ either the hand related to the arm equipped with the sEMG/vibrotactile bracelets or both hands for physically guiding the robot during kinesthetic PbD and using the joystick during the teleoperated PbD. Furthermore, throughout the cable and routing insertion task, the subjects were instructed to maintain the lowest possible CC-level during the task phases in which no programming of gripper or compliance was required.

5) Metrics: In the following section, the results of the experimental session will be reported. The following metrics will be exploited for the presentation of the outcomes: *i)* single subject and grouped analysis of the modulation of the CC-level in accordance with the required programming goals, exploiting boxplots and ANOVA statistical analyses; *ii)* success rate of the programmed robotic wiring task; *iii)* success rate of the programmed programming task in presence of artificial displacement of the final cable insertion case location.

IV. Results

A. Offline Programming of the Robotic Wiring Task

First of all, the trajectories demonstrated by the 10 subjects using kinesthetic and teleoperation modalities can be observed in Fig. 6, plotted as their projection onto the x-y plane for a better visualization. In the figure, also the task locations are reported, denoted as T1,...,T4 (in accordance with Section III-B2.)

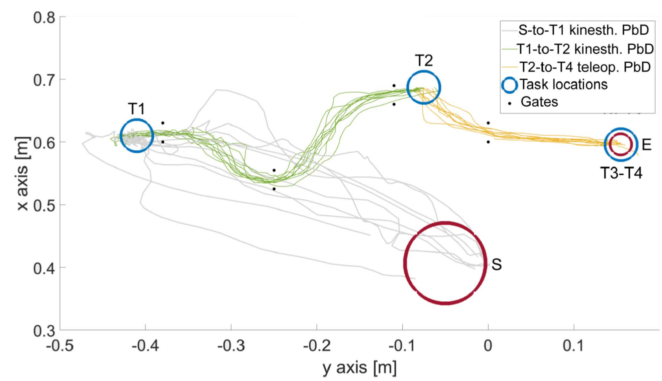


Fig. 6. Robot end-effector trajectories demonstrated by the 10 subjects involved in the experiment.

Furthermore, as illustrated in Fig. 7(a), we focus on the behaviouor shown by a single subject, namely the subject S1 (see Section III-B1). On the other hand, in Fig. 8(a), (b), we extend our analysis by considering the aggregated results over all 10 subjects. In the following, a detailed presentation of the results and figures that we preliminarily introduced is provided.

1) Single Subject Results: In the bottom graph of Fig. 7(a), it is possible to observe the behaviour exhibited by the subject S1 during the programming of the robotic wiring task, with specific regard to the modulation of the CC-level. From the figure, it can be seen how the subject S1 successfully modulated the CC-level in order to program gripper actions and compliance level in accordance with the requirements specified by the experimental protocol. Specifically, the subject correctly programmed the closure of the gripper in proximity of the location T1, in order to grasp the cable extremity, while positioning the robot via kinesthetic PbD. To do this, in accordance with the FSM logics described in Section II-D, the subject firstly stiffened his hand in order to bring the CC-level over the threshold until the virbotactile feedback pattern FB#1 (denoted in the bottom plot of Fig. 7(a) by a black-filled square symbol) was perceived, then modulated the CC-level below the threshold in order to enter in the *Programming Mode state* (acknowledged by the

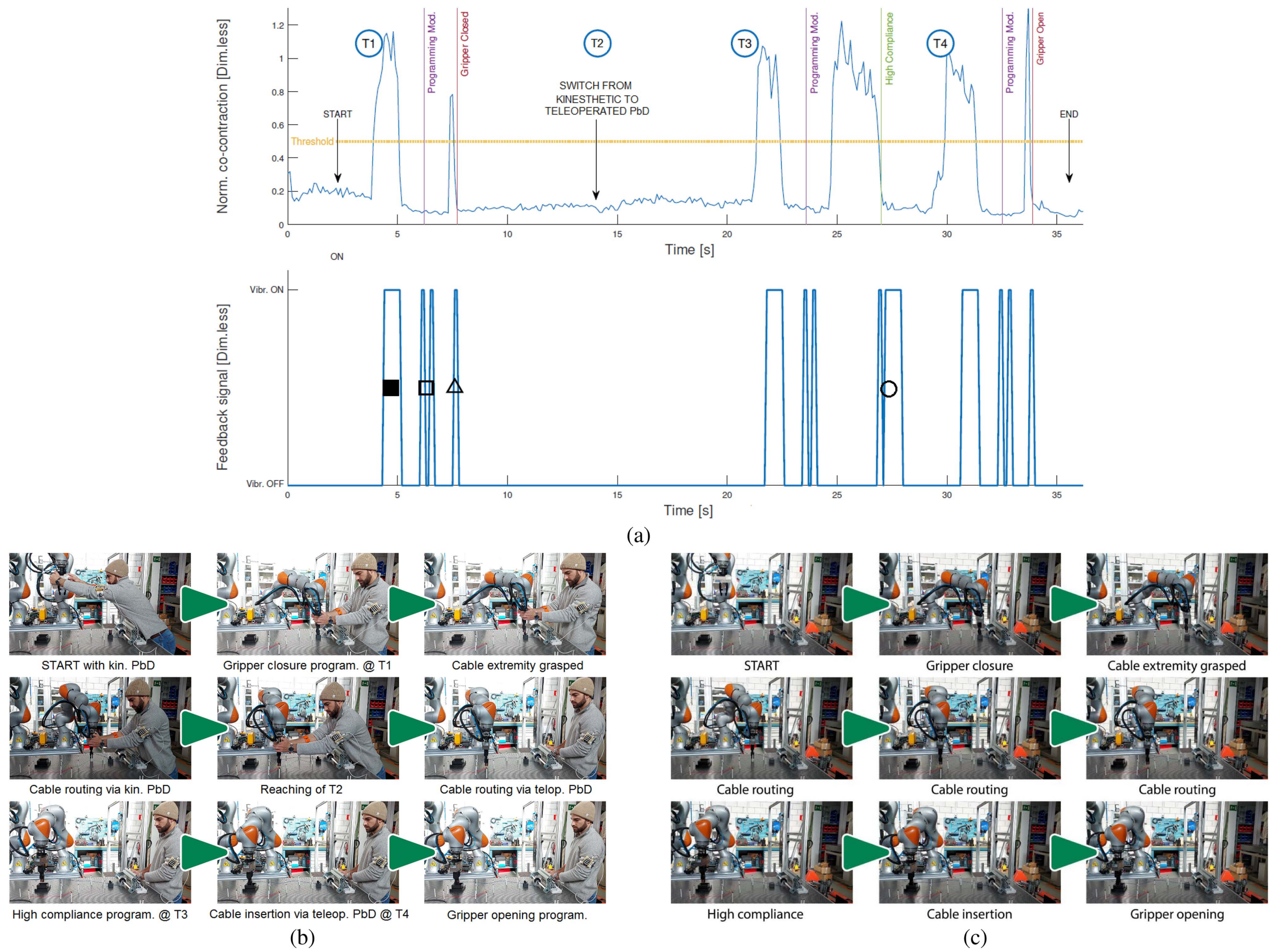


Fig. 7. Single subject results. (a) Modulation of the CC-level exhibited by the subject S1 during the programming of the robotic wiring task (upper graph), and vibrotactile feedback patterns conveyed to subject S1 (bottom graph: black-filled square, square, triangle and circle symbols denote FB#1, FB#2, FB#3 and FB#4, see Section II-D.) (b) Video frame sequence of the programming task carried out by the subject S1. (c) Video frame sequence of the task programmed by the subject S1.

reception of the vibrotactile feedback pattern FB#2, as denoted by the unfilled square symbol in the bottom graph of Fig. 7(a)), and finally programmed the gripper closure by performing a short surpassing of the threshold by the CC-level and receiving a vibrotactile feedback pattern FB#3 as acknowledgement (triangle symbol in the bottom plot of Fig. 7(a).) Thereafter, a programming of the robot compliance level was performed by modulating the CC-level in proximity of the task location T3, this time during the teleoperated PbD. In this relation, again the subject S1 showed to be able to enter in the Programming Mode state, and then to program a high level of compliance by stiffening his hand in order to regulate the CC-level over the threshold until the vibrotactile feedback pattern FB#4 was perceived, denoted by the circle symbol in the bottom graph of Fig. 7(a). Finally, the programming of the gripper was repeated after that the cable insertion was successfully demonstrated at task location T4 by means of the teleoperated PbD modality, in order to release the cable extremity and therefore accomplish the robotic wiring programming task. Moreover, from Fig. 7(a) it is clear how, during movements between task locations and in every instant in which no gripper or compliance programming was required, the subject successfully limited the CC-level under the threshold. In addition, a sequence of frames extracted from a video of the programming task executed by the subject S1 is provided, reporting in Fig. 7(b) the kinesthetic and teleoperation teaching and in Fig. 7(c) the automatic replication of the wiring task by the robot, showing and highlighting the most relevant phases of the experiment.

2) Aggregated Results: The performance of all other subjects closely resembled that of subject S1, as evidenced by the aggregated results shown in Fig. 8, in which the comprehensive behaviour obtained from the regulation of the CC-level during the combined kiensthetic-teleoperated PbD task is presented, considering experimental data from the group of subjects. The data has been grouped using boxplots, referencing the same relevant phases of the programming task as the one shown in a subject-specific manner in Fig. 7. The boxplots of Fig. 8(a) show how each of the subjects was able to successfully regulate the hand stiffening in order to modulate the CC-level over the activation threshold, in conjunction with the virbtoactile feedback already described for the single subject results, in order to properly perform the programming of gripper and compliance

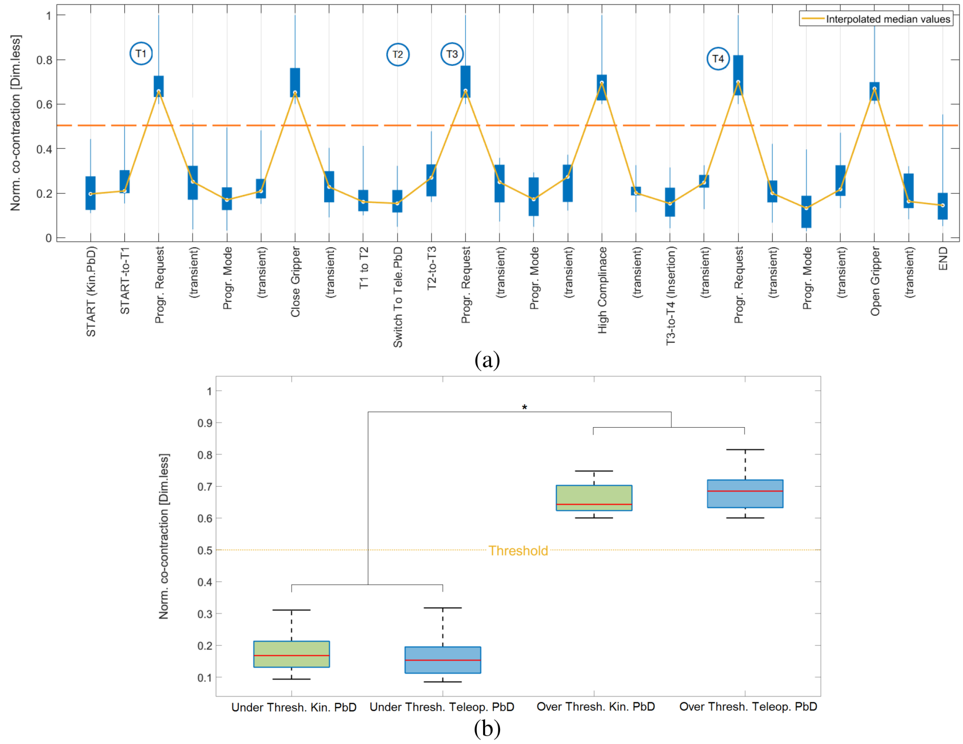


Fig. 8. Aggregated results over the group of 10 subjects. (a) Regulation of the CC-level during the combined kiensthetic-teleoperated PbD task, grouped over the subjects based on relevant task phases in accordance with Fig. 7. (b) Subjects' CC-level modulations grouped on the basis of the factors *requested CC-level modulation* (under threshold, over threshold) and *PbD modality* (kinesthetic, teleoperated.) The symbol "*" denotes statistical significant difference, $p < 0.01$.

level. Furthermore, the outcome of a statistical analysis on the aggregated results is presented in Fig. 8(b). Specifically, we further grouped the CC-level modulations based on the different requested levels (under and over the threshold), and on the PbD modality used (kinesthetic or teleoperated PbD.) We therefore performed a two-way repeated measures Analysis of Variance (ANOVA), with factors the *requested CC-level modulation* and the *PbD modality*. The statistical significance threshold was set to $p < .01$. Before performing the ANOVA, we ensured the validity of its assumptions through the Shapiro-Wilk test for normality and Mauchly' test for sphericity. The results indicated that these assumptions were not violated. The outcome of the ANOVA revealed a statistically significant effect of the only factor *requested CC-level modulation*, with a post-hoc Tukey test reporting for p-values lower than 0.01 for any comparison of the first two boxplots with the second two boxplots in Fig. 8(b). Therefore, the group of subjects was globally able to modulate the CC-level based on the necessity of programming gripper and robot compliance level during kinesthetic/teleoperated PbD with statical evidence. Consequently, the study demonstrates the subjects' robust ability to modulate the CC-level provided the proposed wearable H-R interface, highlighting the potential of this approach for advancing PbD more a powerful, yet still intuitive, programming of robots.

B. Online Automatic Execution of the Wiring Task

The online robotic wiring phase involved a cyclic repetition of the programmed task by the robot, performed for a total of 5 consecutive times. As previously detailed in Section III-B2, the experimental protocol required the programming of the gripper for the cable extremity grasping and release, and the programming of the level of robot compliance to ensure smooth interactions with the test-bed cable insertion component. Remarkably, the results obtained from the experimental session with the online automatic execution of the robotic wiring reported for a success rate of 100%. This means that all cable picking, routing and insertion operations were effectively performed by the robot in all cases, across all subjects and cyclic task repetitions, thereby highlighting the effectiveness of the proposed programming framework. Furthermore, as additional evaluation situation, an intentional physical interaction was introduced between the robot end-effector and cable insertion component. This interaction was enforced on-the-fly during two of the five repetitions of the automatic execution of the wiring task, by applying a 5 mm displacement to the cable insertion component position. During this test, the robot demonstrated the ability to successfully compensate for the introduced component displacement, thanks to the programming of a high robot compliance level via CC-level modulation. In order to provide a clearer picture of this behaviour, we report in Fig. 9 a graph of the cable insertion component in the new position displaced by 5 mm with respect to the nominal one used during the robot programming phase. In particular, the graph also reports in dashed orange line the three-dimensional trajectory demonstrated by the subject S1 limited to the portion from the task location T2 to the task location T4, which was provided by means of the teleoperation modality (see Section III-B2), whereas in solid blue line the

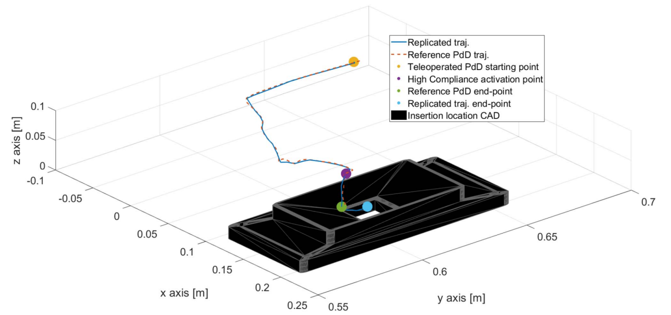


Fig. 9. Cable insertion component displaced by 5 mm with respect to the nominal location. The dashed orange line denotes the trajectory demonstrated by the subject S1, whereas the solid blue line the actual end-effector trajectory resulting from the online automatic execution of the programmed cable insertion.

actual end-effector trajectory resulting from the online automatic replication of the programmed wiring task. As it is possible to observe, the artificial displacement of the insertion component caused the robot end-effector to be not properly center with respect to the insertion hole, resulting in an undesired contact between the robot and the test-bed component which could potentially bring to the damage of hardware parts, or to an emergency stop of the automatic wiring task due to levels of forces surpassing the safety limits. However, the robot is capable of compliantly compensate the unexpected interaction forces and complete the insertion task without failures, as shown by the online end-effector trajectory reported in solid blue line in Fig. 9. This experimental outcome highlights the effectiveness of the proposed method for programming the robot compliance at desired trajectory locations, ensuring the safety and success of the tasks, such as the cable insertion operation, with the possibility of enhancement of the productivity and safety in diverse robotic applications.

C. Systematic Evaluation of Vibrotactile Feedback

In order to more systematically analyse the presence of the vibrotactile feedback in the proposed PbD framework, we report in this section the result of a test performed in a purposely controlled and structured experimental situation. The subjects were required to exploit the modulation of the CC-level to carry out simulated programming of both gripper and robot compliance level –in accordance with the methods illustrated in Section II-D and Fig. 3– in two different conditions: with and without the presence of the vibrotactile feedback. Specifically, we want to emphasize the following benefits of the presence of the vibrotactile feedback in the H-R communication of our PbD approach:

- the users are enabled to be aware of the actual *programming state* of the system, since the vibrotactile feedback is provided in correspondence of transitions in the FSM for the programming of gripper and robot compliance (refer also to Fig. 3): based on the different encoded vibration patterns, the users can recognize when they entered the programming mode, gripper programming or compliance programming states, drastically reducing –and even eliminating– programming failures due to misinterpretations of the current programming state of the system;
- the users can exploit the vibrotactile feedback to improve/optimize the modulation of the CC-level in order to activate different programming states of the FSM for the programming of gripper and robot compliance (see Fig. 3): indeed, since transitions may require to modulate the CC-level over or under a threshold for a minimum or maximum time (see detailed explanation in Section II-D), the presence of the vibrotactile feedback, by providing the information about *when* the transitions have been enforced, allows to decrease/minimize the time necessary for providing correct CC-level-based programming inputs, attenuating, in this way, also the unnecessary additional muscular and/or mental effort for the realization of the PbD task.

Therefore, to this aim, specific temporal, programming failure and muscular metrics were defined in relation to the activation of the programming mode, gripper programming and compliance programming states of the FSM of Fig. 3. In particular, about the temporal and programming failure metrics, with reference to the exemplification graph reported in Fig. 10(b), we considered *i)* the time T_{P1} in which the CC-level is kept over the threshold once the *programming request* state has been activated, *ii)* the time T_{P2} in which the CC-level is kept under the threshold once the *programming mode* state has been activated, and *iii)* the number of failures in correctly activating the programming mode when required by the task. Accordingly, we then considered also *i)* the time T_G and T_C in which the CC-level is kept over the threshold once the gripper and compliance programming have been activated, respectively, and *ii)* the number of failures in correctly enforcing gripper and compliance programming. On the other hand, regarding the muscular metrics, we considered the percentage of cumulative CC-level kept over the threshold during the times T_{P1}, T_{P2}, T_G and T_C.

In order to systematically evaluate these temporal metrics in presence and absence of vibrotactile feedback, we carried out an experiment involving a group of 10 subjects. Each of the subjects was sat in front of a screen showing a robotic gripper, simulated in Matlab's Simscape environment, and a text instructing for a specific programming input to be produced via CC-level modulation (see Fig. 10(a)). The subject was therefore wearing the sEMG bracelet (in order to be properly able to modulate the CC-level in the same manner as described in Section II-C1) and the vibrotactile bracelet. A series of programming instructions were therefore provided to the subject, reproducing the same sequence of the experiments with the robot manipulator described in Section II-D, that is *i)* "program gripper closing", *ii) program high compliance level*, and *iii) program gripper opening*. In this way, we reproduced a simulated programming situation in which each subject received –in the same manner as with the case with the real robot– the visual feedback of the programming of the gripper opening/closing, whereas, for the vibrotactile feedback, we tested to different condition: *i)* presence (as described in Section II-D) and *ii)* absence of the vibrotactile feedback for the programming of both gripper and compliance level. This experiment was therefore repeated, for each subject, 10 times

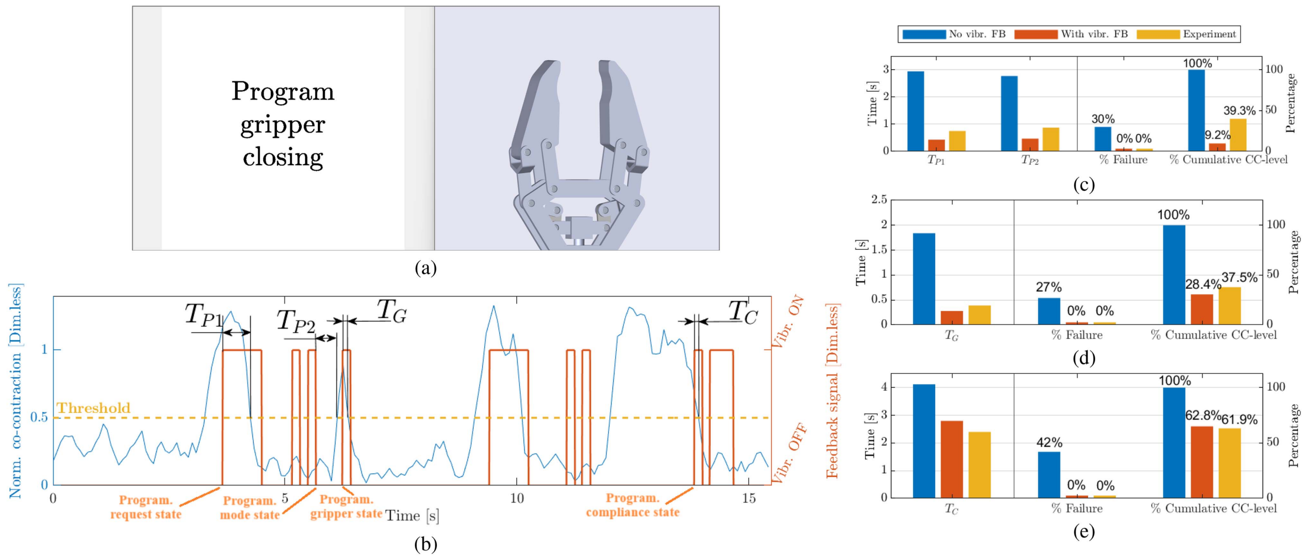


Fig. 10. (a) Programming instruction text and simulated robotic gripper used in the systematic evaluation of the vibrotactile feedback. (b) Exemplification plot of the CC-level modulation for the programming of gripper and compliance level, with temporal metrics for the evaluation of the presence/absence of the vibrotactile feedback. (c)–(e) Bar graphs comparing the results of the temporal metrics and failure percentages among the following conditions: absence of vibrotactile f.b., presence of vibrotactile f.b., and experiment with real robot manipulator programming.

with vibrotactile feedback, and 10 times without vibrotactile feedback, and the metrics T_{P1}, T_{P2}, T_G, T_C, programming failure percentage and cumulative CC-level percentage were computed in order compare results. In particular, the cumulative CC-level percentages were scaled such that the percentage in the absence of vibrotactile feedback condition was normalized to 100%. Figs. 10(c)–(e) report the results of the mean values of the metrics obtained by *i)* the test without vibrotactile feedback ("No vibr. FB" in Fig. 10(c)–(e)), *ii)* the test with vibrotactile feedback ("With vibr. FB" in Fig. 10(c)–(e)), and *iii)* the experiment with the real robot described in Section III. Specifically, the mean temporal metrics, failure percentage and cumulative CC-level percentage are reported in Fig. 10(c), Fig. 10(d) and Fig. 10(e) in relation to the activation of the *programming mode* state, *gripper programming* state, and *compliance programming* state, respectively. As it is possible to observe in the bar graphs reported in these figures, it is clear how the absence of vibrotactile feedback substantially increased all temporal metrics and failure percentages for all the considered cases, showing how the information provided by the vibrotactile feedback in relation to the transitions in the FSM of Fig. 3 allowed the users optimize/minimize the time for enforcing programming inputs by means of CC-level modulations. At the same time, it is also clear how the presence of vibrotactile feedback in both the simulated and real experiments allowed a decrease of the percentage of cumulative CC-level with respect to the condition without vibrotactile feedback, highlighting a relevantly lower muscular effort in order to accomplish the programming tasks.

D. Qualitative Comparison With Representative Literature Works

In this section, we consider literature works that exploits neuromuscular H-R interfacing in order to enhance intuitive robot trajectory methoda. In particular, a qualitative comparison is reported, due to difficulties in considering a common ground truth for metrics comparison, which is an intrinsic issue due to the diverse approached and robotic task considered in the different studies. We therefore report in Table I the features of four selected sEMG-enhanced approaches for robot intuitive transfer of motion skills from human to robots, along with our proposed method. As can be seen in the table, our proposed PbD approach is the only framework that *i)* has been developed to be exploited with both kinesthetic teaching and teleoperation trajectory teaching modalities, *ii)* allows to exploit the neuromuscular interfacing for teaching multiple features (both compliance levels and gripper opening/closing), *iii)* realize a bidirectional H-R communication bye means of feedback provided to the user (vibrotactile feedback), and *iv)* has been tested in an industrial-like scenario experiment (simplified robotic wiring for switchgears).

The idea of using neuromuscular interfacing for transferring motion skills from humans robots has been experimented in other literature works: in the following we discuss the representative approaches that are summarized in Table I. The concept of teleimpedance was introduced in [9], where in a leader–follower teleoperation scheme, a reference command composed of both the desired motion trajectory and the impedance profile is sent from the human operator to the remotely operated robot. Note that in this appr impedance profiles are deduced from sEMG measurements on the operator' arm. In [10], a framework for robot PbD based on a single demonstration provided by the human operator is proposed, in which the teleimpedance approach, for teaching the robot stiffness, is integrated with a generalization of the trajectories demonstrated by the user in order to let the robot adapt to different object and environment configurations. In [11], sEMG signals and kinesthetic teaching are exploited in a PbD task, where the positions imposed by the human operator and impedance gains deduced from the

TABLE I
FEATURE COMPARISON BETWEEN PROPOSED METHOD AND REPRESENTATIVE ROBOT TRAJECTORY TEACHING APPROACHES ENHANCED BY H-R NEUROMUSCULAR INTERFACING

Ref.	Approach considered for the trajectory teaching	Neuromuscular H-R interfacing	Physiological quantity	Additional provided teaching inputs	Feedback provided to the user	Experimental evaluation
[9]	Teleoperation (teleimpedance approach)	sEMG	Human arm impedance (sensors required on both forearm and upper arm)	Robot end-point impedance	None	Peg-in-the-hole and ball-catching tasks
[10]	One-shot teleoperation and Dynamic Movement Primitives (DMP) (teleimpedance approach)	sEMG	Human arm impedance (sensorson both forearm and upper arm)	Robot end-point impedance	None	Peg-in-the-hole and object-sorting tasks
[11]	Kinesthetic teaching	sEMG	Human wrist stiffness (sensors on the forearm)	Robot end-point stiffness	None	Analysis of robot trajectory profiles
[12]	Kinesthetic teaching	sEMG	Human elbow stiffness (sensors on the upper arm)	Robot end-point stiffness	None	Cutting and pick-and-place tasks
Proposed PbD framework	Both kiensthetic teaching and teleoperation	sEMG	Human hand muscle co-contraction (sensors on the forearm)	Gripper opening/cloising and robot end-point compiance	Vibrotactile feedback	Robotic wiring task: picking, routing and inserting of a cable

muscles activity are recorded and then applied offline to the robot. Similarly, in [12], the trajectory demonstrated by a human operator together with sEMG signals is used to command both the position and the stiffness of a robot manipulator interacting with the environment. In these cases, the signals are encoded by using DMPs models, so that they can be easily adapted to new tasks.

Differently from the abovementioned works (refer also to Table I), in the framework proposed in this article, we are not going to estimate the user's arm stiffness and/or compliance with the objective of making the robot replicate the human stiffness profile. Furthermore, note that, the estimation of human arm's stiffness/compliance may be highly dependent by arm configuration, and in general is not independent from the user's upper limb motions/postures. In an alternative way, in our approach, we use sEMG to estimate the user' hand (fingers and wrist joints) muscle co-contraction level – from the activation of the predominant extrinsic hand' antagonistic muscles located in the forearm – in order to provide the user with additional programming capabilities. Furthermore, we demonstrated that our approach can be used to specify at runtime, during *both* the execution of kinesthetic teaching based or teleoperation-based robot trajectory programming. Importantly, since the user should be able to voluntarily and intuitively enforce the additional programming capabilities, a vibrotactile feedback is provided to inform her/him about the actual programming state of the robotic system. This approach allows us to better capture the human intentions for PbD tasks, and enhance their effectiveness for more complex tasks, by still maintaining an high level of intuitiveness.

V. CONCLUSION

In this study, by exploiting wearable human-robot interfaces based on surface Electromyography (sEMG), users can modulate the level of forearm muscle co-contraction without generating additional motions or forces, effectively providing additional degrees of programming during PbD. The integration of vibrotactile feedback enhances the communication between the operator and the robotic system, ensuring the effectiveness and intuitiveness of the interface. The experiments conducted on an industrial wiring task demonstrate the successful programming of robot compliance and gripper grasping functionalities, both through kinesthetic and teleoperated PbD. Online automatic execution of the robotic wiring task achieved a 100% success rate, showcasing the approach practical viability. Future research could explore the extension of this approach to program additional collaborative robot functionalities, further enhancing their capabilities and usability in real-world applications.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2008, pp. 1371–1394.

[2] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Philos. Trans. Roy. Soc. London. Ser. B: Biol. Sci.*, vol. 358, no. 1431, pp. 537–547, 2003.

[3] D. Lee and C. Ott, "Incremental kinesthetic teaching of motion primitives using the motion refinement tube," *Auton. Robots*, vol. 31, pp. 115–131, 2011.

[4] B. Dariush, M. Gienger, B. Jian, C. Goerick, and K. Fujimura, "Whole body humanoid control from human motion descriptors," in *Proc. 2008 IEEE Int. Conf. Robot. Automat.*, 2008, pp. 2677–2684.

[5] S. Calinon, "Learning from demonstration (programming by demonstration)," in *Encyclopedia of Robotics*. Berlin, Germany: Springer, 2018, pp. 1–8.

[6] M. J. Zeestraten, I. Havoutis, and S. Calinon, "Programming by demonstration for shared control with an application in teleoperation," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1848–1855, Jul. 2018.

[7] M. Tykal, A. Montebelli, and V. Kyrki, "Incrementally assisted kinesthetic teaching for programming by demonstration," in *Proc. 11th ACM/IEEE Int. Conf. Hum.-Robot Interact.*, 2016, pp. 205–212.

[8] L. Peternel, T. Petrič, and J. Babič, "Human-in-the-loop approach for teaching robot assembly tasks using impedance control interface," in *Proc. 2015 IEEE Int. Conf. Robot. Automat.*, 2015, pp. 1497–1502.

[9] A. Ajoudani, S. B. Godfrey, M. Catalano, G. Grioli, N. G. Tsagarakis, and A. Bicchi, "Teleimpedance control of a synergy-driven anthropomorphic hand," in *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1985–1991.

[10] G. Lentini, G. Grioli, M. G. Catalano, and A. Bicchi, "Robot programming without coding," in *Proc. 2020 IEEE Int. Conf. Robot. Automat.*, 2020, pp. 7576–7582.

[11] R. Wu, H. Zhang, T. Peng, L. Fu, and J. Zhao, "Human-robot interaction and demonstration learning mode based on electromyogram signal and variable impedance control," *Math. Problems Eng.*, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:125428197

[12] C. Yang, C. Zeng, C. Fang, W. He, and Z. Li, "A DMPs-based framework for robot learning and generalization of humanlike variable impedance skills," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 3, pp. 1193–1203, Jun. 2018.

[13] R. Meattini, D. Chiaravalli, L. Biagiotti, G. Palli, and C. Melchiorri, "Combining unsupervised muscle co-contraction estimation with biofeedback allows augmented kinesthetic teaching," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 6180–6187, Oct. 2021.

[14] R. Meattini, D. Chiaravalli, K. Galassi, G. Palli, and C. Melchiorri, "Experimental evaluation of intuitive programming of robot interaction behaviour during kinesthetic teaching using semg and cutaneous feedback," *IFAC-PapersOnLine*, vol. 55, no. 38, pp. 1–6, 2022.

[15] L. Biagiotti, R. Meattini, D. Chiaravalli, G. Palli, and C. Melchiorri, “Robot programming by demonstration: Trajectory learning enhanced by sEMG-based user hand stiffness estimation,” *IEEE Trans. Robot.*, vol. 39, no. 4, pp. 3259–3278, Aug. 2023.
[16] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Berlin, Germany: Springer, 2010.
[17] N. Jiang, P. A. Parker, and K. B. Englehart, “Spectrum of the nonstationary electromyographic signal modelled with integral pulse frequency modulation and its application to estimating neural drive information,” *J. Electromyogr. Kinesiol.*, vol. 19, no. 4, pp. e267–e279, 2009.
[18] E. Henneman, “Relation between size of neurons and their susceptibility to discharge,” *Science*, vol. 126, no. 3287, pp. 1345–1347, 1957.
[19] E. J. Bayly, “Spectral analysis of pulse frequency modulation in the nervous systems,” *IEEE Trans. Biomed. Eng.*, vol. BME-15, no. 4, pp. 257–265, Oct. 1968.
[20] R. Meattini, S. Benatti, U. Scarcia, D. De Gregorio, L. Benini, and C. Melchiorri, “An sEMG-based human–robot interface for robotic hands using machine learning and synergies,” *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 8, no. 7, pp. 1149–1158, Jul. 2018.
[21] “ROS trajectory msg,” (n.d.), 2023. [Online]. Available: http://wiki.ros.org/trajectory_msgs
[22] “ROS WSG50 driver,” (n.d.), 2023. [Online]. Available: https://github.com/nalt/wsg50-ros-pkg
[23] “Remodel project,” Dec. 2023. [Online]. Available: https://remodel-project.eu/

Roberto Meattini (Member, IEEE) received the M.Sc. degree in automation engineering and Ph.D. degree in biomedical, electrical, and systems engineering from the University of Bologna, Bologna, Italy, in 2014 and 2018, respectively. He is currently a Junior Assistant Professor with the Department of Electrical, Electronic, and Information Engineering, University of Bologna. His research interests mainly include dexterous manipulation, telemanipulation, human-in-the-loop control of robots, and human-robot interaction.

Armando Amerì received the M.Sc. degree in automation engineering from the University of Bologna, Bologna, Italy, in 2023. During 2023–2024, he worked in collaboration with small companies, developing web applications in the medical and robotics field. He is currently a Member with Cybersecurity Team, Cognizant, as Android Reverse Engineer.

Alessandra Bernardini (Graduate Student Member, IEEE) received the B.S. and M.S. degree (both with *laude*) in 2019 and 2021, respectively, from the University of Bologna, Bologna, Italy, where she is currently working toward the Ph.D. degree in biomedical, electrical and system engineering. Her research interests include human–robot interaction, robot hand control, and robotic manipulation.

Javier Gonzalez-Huarte received the B.Sc. degree in biomedical engineering and the M.Sc. degree in biomedical technologies from Mondragon University, Mondragón, Spain, in 2019 and 2021, respectively. In 2018, he began his Research Career with the Institut des Sciences du Mouvement (Aix-Marseille University, Marseille, France), focusing on movement sciences. Since 2021, he has been with TECNALIA, progressing from the Medical Robotics division to the Advanced Manufacturing Area, focusing on robotics solutions across various fields.

Aitor Ibarguren received the M.Sc. and Ph.D. degrees in computer science from the University of the Basque Country, Leioa, Spain, in 2004 and 2008, respectively. From 2004 to 2016, he was with Tekniker, as a Member with Robotics Group, applying Machine Learning and optimization technologies to the robotics field. In 2017, he joined Tecnalia, as a Senior Researcher, focusing his research on mobile manipulation applications.

Claudio Melchiorri (Fellow, IEEE) received the Laurea and Ph.D. degrees in electrical engineering from the University of Bologna, Bologna, Italy, in 1985 and 1990, respectively. He is currently a Full Professor of robotics with the Department of Electrical, Electronic and Information Engineering “Guglielmo Marconi,” University of Bologna. He was with the organization of several international conferences. He has authored or coauthored about 350 scientific papers and is author of two books on trajectory planning for automatic machines and robots, and author or (co-)editor of seven books in robotics. His research interests include dexterous manipulation, haptics, telemanipulation, nonlinear control, and motion control.

Gianluca Palli (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in automation engineering from the University of Bologna, Bologna, Italy, in 2003 and 2007, respectively. He is currently a Full Professor with the University of Bologna. He has authored or co-authored more than 100 scientific articles presented at conferences or published in journals. His research interests include design and control of manipulation devices, mobile manipulation, manipulation of deformable objects and soft robotics.