Programming Robot Interaction Behavior During Kinesthetic Teaching Exploiting sEMG-Based Interfacing and Vibrotactile Feedback

Roberto Meattini, Andrea Govoni, Kevin Galassi, *Graduate Student Member, IEEE*, Davide Chiaravalli, Gianluca Palli, and Claudio Melchiorri, *Life Fellow, IEEE*

***Abstract*—We present a programming-by-demonstration framework that allows users to simultaneously teach both motion and compliance behaviors to collaborative robots. The system integrates a wearable surface electromyography interface and real-time vibrotactile feedback into the kinesthetic teaching process, enabling intuitive point-by-point modulation of robot's mechanical compliance. Unlike prior methods that focus solely on trajectory or symbolic control, our approach leverages unsupervised co-contraction estimation for continuous impedance shaping, without the need for explicit offline procedures or labeled data. To validate the method, we redesigned and conducted two contact-rich tasks—industrial wiring and a welding-like operation—demonstrated by ten subjects using a 7-degrees of freedom (DOF) collaborative manipulator. Results show that users could effectively program both trajectory and compliant interaction behaviors, highlighting the framework's potential for intuitive and robust human–robot collaboration.**

***Index Terms*—Human-centered automation, learning from demonstration, physical human–robot interaction (HRI).**

I. Introduction

COLLABORATIVE robots are increasingly used across industries, driving demand for intuitive interfaces that enable users to teach both motion and interaction behaviors, including compliance and force control [1]. Recent research has focused on developing controllers compatible with human–robot interaction (HRI), including solutions for constrained task spaces [2], impedance learning [3], and neural network-based approaches [4]. As a result, programming-by-demonstration paradigms have emerged, allowing nonexpert operators to intuitively teach robots tasks [5]. Kinesthetic teaching [6] is a key approach in programming by demonstration, where the robot is physically guided through the desired motions. This method is particularly effective in industrial scenarios because it avoids the issue of kinematic dissimilarity between the operator and robot, which is a challenge in observational demonstration techniques [7]. Observational methods also require complex setups like vision systems [8], limiting their practical use. In contrast, kinesthetic teaching is widely applicable in real-world industrial settings that demand quick, intuitive robot reprogramming. However, kinesthetic teaching is still often limited to the sole demonstration of geometric trajectories, overlooking the potential of modern collaborative robots to also encode interaction dynamics through features such as impedance modulation. This restriction poses challenges in application scenarios where physical interaction is essential, not only for safety, but also for task execution quality. Fields, such as industrial robotics, human–robot coworking, and rehabilitation robotics frequently require the robot not only to follow a path, but also to adapt its mechanical behavior during contact with humans, objects, or environments. Programming such behaviors manually or through traditional interfaces can be time-consuming and unintuitive for nonexpert users. To address this, we propose an intuitive programming-by-demonstration framework that combines kinesthetic teaching with biosignal-based control, enabling users to simultaneously program both the robot trajectory and its compliance profile. Our approach leverages unsupervised learning techniques to estimate hand muscle cocontraction from surface electromyography (sEMG) and provides real-time vibrotactile feedback to help users modulate robot impedance during demonstration. This allows the operator to intuitively specify when and how the robot should behave more compliantly—especially in phases involving physical contact—resulting in more natural and flexible interaction programming. In summary, the main contributions of this work are as follows.

1) We propose a novel programming interface that enables users to simultaneously teach both motion and interaction behavior during kinesthetic guidance.

Received 24 April 2025; revised 30 June 2025; accepted 24 August 2025. Date of publication 9 September 2025; date of current version 17 October 2025. Recommended by Technical Editor A. Visioli Senior Editor D. Chen. This work was supported in part by European Commission's Horizon Europe Framework Programme with the project IntelliMan under Grant 101070136, in part by the MUR with the project "Sustainable Mobility Center" under Grant CN00000023-CUP J33C22001120001, and in part by the MICS (Made in Italy – Circular and Sustainable) Extended Partnership and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1551.11-10-2022) under Grant PE00000004. *(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.

The authors are with the Department of Electrical, Electronic and Information Engineering (DEI), University of Bologna, 40136 Bologna, Italy (e-mail: roberto.meattini@unibo.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2025.3603402.

Digital Object Identifier 10.1109/TMECH.2025.3603402

2) This is achieved by exploiting muscular cocontraction, which can be voluntarily modulated and reliably estimated in real time from surface EMG using unsupervised methods.
3) This results in an interactive and intuitive framework for teaching compliant behaviors, enhancing the exploitation of collaborative robots' advanced capabilities in human-interactive tasks.

While recent literature has explored related directions, such as EMG-based impedance control [9], vibrotactile feedback for assistive guidance [10], or variable-stiffness learning in specialized applications like surgical cutting [11], these works do not address the simultaneous and intuitive programming of both trajectory and compliance during kinesthetic teaching. In contrast, our framework introduces a unified solution where muscle cocontraction—estimated in real time from sEMG—directly modulates impedance during the demonstration phase, and where vibrotactile feedback actively supports user awareness and control. This bridges a critical gap in collaborative robotics by enabling low-effort, real-time interaction programming through a wearable biosignal-based interface.

II. Related Work and Proposed Approach

Information exchange in HRI has been explored through modalities, such as virtual reality for maintenance tasks [12] and haptic handles with force sensors [13]. Machine vision approaches have been used for tracking human motions, such as gaze [14], hand gestures [8], or body postures [15]. These methods often require bulky hardware or setup changes, while wearable interfaces offer a more intuitive, unobtrusive alternative [16]. Recent studies show that sEMG is advantageous for detecting human intentions, as it measures muscle activation levels [17]. sEMG's accessibility and portability make it particularly useful for HRI with collaborative robots. Prior research has used sEMG for advanced HRI applications, including estimating human limb impedance [18], gesture recognition [19], and applications in assistive robotics [20]. However, these studies have not focused on using sEMG for programming robot interaction behavior during kinesthetic teaching. While early vibrotactile investigations, such as [21], reveal how tactor placement and prompt type affect response speed, they remain confined to discrete input devices rather than continuous robot guidance. Teleimpedance approaches [22] and subsequent learning from demonstration extensions [23], [24], [25], [26] have combined sEMG-based stiffness estimation with trajectory learning. However, these methods either operate offline, require complex calibration, or lack support for real-time, simultaneous modulation of interaction during kinesthetic teaching. Several more recent works have sought to address these limitations. Wu and Billard [11] proposed a variable impedance learning strategy based on human demonstration for tissue cutting tasks, but their approach requires precise force feedback and remains task-specific. Bednarczyk et al. [9] introduced an EMG-driven variable impedance controller with passivity guarantees, where muscle activation is used to adapt stiffness in real time. However, this approach does not incorporate user feedback or support explicit, user-driven impedance programming during teaching. Similarly, Sirintuna et al. [10] employed vibrotactile feedback to guide users in obstacle-aware collaborative transport, yet their method does not address impedance learning or programming by demonstration (PbD), focusing instead on shared control and task execution safety. In contrast, the present work introduces a novel interface that allows users to program both motion *and* interaction behavior (i.e., compliance) in real time by exploiting voluntary hand muscle cocontraction. This cocontraction is estimated online from sEMG using an unsupervised method, and is reinforced through vibrotactile feedback that guides the user throughout kinesthetic demonstration. The cocontraction signal serves as an intuitive and continuous input for shaping robot impedance dynamically, on a point-by-point basis, during demonstration. This framework differs fundamentally from prior art, in that it enables seamless, low-effort compliance programming—without requiring any offline calibration, labeled data, or dedicated force sensors. Unlike Wu et al. [27], who relied on predefined coactivation thresholds, or Zhang et al. [28], who performed offline stiffness matrix learning, our approach allows direct, real-time compliance shaping. Moreover, Steinmetz et al. [29] focused on semantic-level trajectory learning but did not address impedance programming. Our method bridges this gap by integrating sEMG-based cocontraction estimation and vibrotactile feedback into the kinesthetic teaching loop, resulting in a unified and interactive solution for motion and interaction demonstration in collaborative robotics. Importantly, the present work significantly advances beyond our previous publications [30], [31]. In our 2023 PbD study [30], sEMG-based hand stiffness estimation was estimated via a much less simplistic approach and then used only offline to shape B-spline trajectories' precision, and in the 2023 augmented teaching article [31] hand cocontraction drove only discrete commands (e.g., gripper open/close) via vibrotactile feedback. In contrast, the present framework supports truly real-time, continuous, and simultaneous programming of both motion and impedance during a single kinesthetic demonstration. This enables point-by-point compliance shaping—unachievable in our earlier methods—and fully leverages collaborative robots' variable impedance capabilities for seamless, intuitive human–robot interaction.

A. Overview of the Proposed Programming Framework

Referring to the conceptual scheme illustrated in Fig. 1, the framework combines kinesthetic teaching with neuro-motor programming via sEMG, using a wearable armband to estimate hand cocontraction for real-time impedance control. Vibrotactile feedback enables continuous adjustment. Both trajectory and compliance are recorded for execution, supporting safe, adaptive robot behavior.

B. Kinesthetic Teaching and Interfacing Setup

The setup used in the present work can be observed in Fig. 2. In the following a description of the different components/subsystems is given.

1) Manipulator Control During Kinesthetic Teaching: The experiments were conducted using a 7-DoF Franka Emika Panda

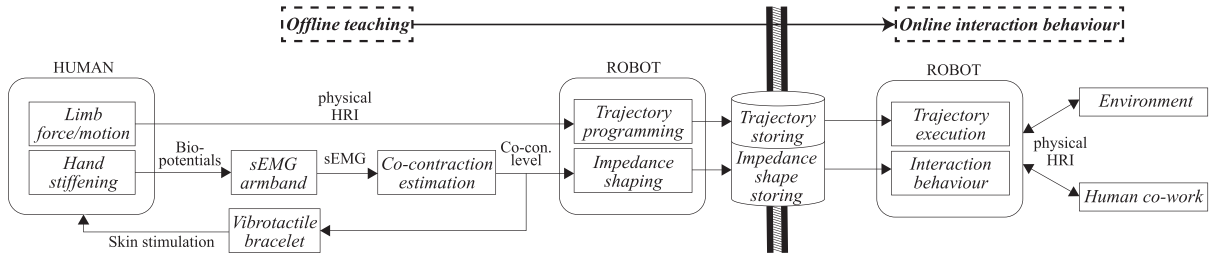


Fig. 1. Conceptual scheme of the proposed programming-by-demonstration framework based on kinesthetic teaching, sEMG and vibrotactile feedback.

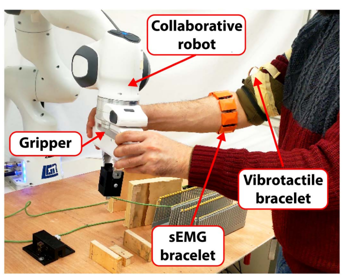


Fig. 2. Robotic and interface setup used to implement and test the programming-by-demonstration framework of the present work.

robot with its default parallel gripper. No external force sensor was used; interaction guidance relied on the robot's built-in features, including gravity and friction compensation and safety limits on force, velocity, and joint range. (see Fig. 2). During kinesthetic teaching, the operator guides the robot's end-effector using an impedance-based interface implemented via a force control loop. Let us consider the dynamic model of the manipulator, with $n = 7$ DoF, in the form [32]

$$M(x)\ddot{x} + C(x,\dot{x})\dot{x} + g(x) = F_{\text{input}} + F_h \tag{1}$$

where $x \in \mathbb{R}^6$ is the pose describing the end-effector position and orientation, and $M(x) \in \mathbb{R}^{n\times n}$, $C(x,\dot{x}) \in \mathbb{R}^{n\times n}$, and $g(x) \in \mathbb{R}^n$ are the inertia matrix, Coriolis matrix and gravitational term defined in the workspace, respectively. F_h denotes the external wrench imposed by the operator, and the control input F_{input} is enforced as

$$F_{\text{input}} = \hat{g}(x) + F_c \tag{2}$$

where the term $\hat{g}(x)$ denotes the estimate of the gravity term, and F_c is defined as a proportional-derivative (PD) controller between the actual pose x and a desired pose x_d, that is

$$F_c = K_p(x_d - x) - K_d\dot{x} \tag{3}$$

where $K_p, K_d \in \mathbb{R}^{6\times 6}$ are positive definite diagonal matrices. Note that setting appropriate diagonal gains in K_p to zero effectively defines the directions and rotations, in which kinesthetic teaching can be performed by the operator. In general, the control framework we implement also allows for the possibility of guiding the operator along desired orientations or directions, potentially assisting the teaching process through constraints or virtual fixtures. For instance, one may want to enforce a constant end-effector orientation during specific phases of the task or throughout the entire teaching procedure, or to facilitate programming in scenarios where the trajectory is already known and only timing, grasping, processing and/or interaction behavior needs to be specified. In such cases, virtual constraints could reduce the cognitive and physical load on the user. However, in our work, we intentionally leave all translational and rotational directions fully unconstrained, providing the operator with maximum freedom during teaching. As a result, in our experiments, both position and orientation trajectories are programmed by the user, as reported in the experimental results section.

2) Manipulator Control for Trajectory Replication: During execution, the control strategy tracks the taught trajectory while adapting compliance for interaction. To this end, (2) is modified by setting $F_r = 0$, since friction compensation is only relevant during kinesthetic teaching. Then, to implement a variable impedance dynamics behaviour and ensure the system's stability, a passivity-based approach using an energy tank [33] is adopted. The energy tank stores dissipated energy, allowing impedance changes only when sufficient energy is available, thus ensuring passivity. (3) is redefined as

$$F_c = \bar{K}_p(x_d - x) - \omega x_t(x_d - x) - K_d\dot{x} \tag{4}$$

where $\bar{K}_p = \text{diag}(k_{i_{\min}})$ with $k_{i_{\min}}$ being the minimum compliance along Cartesian directions. The control input ω modulates compliance by exchanging energy with the tank

$$\omega = \begin{cases} -\frac{\widetilde{K}_p}{x_t}, & \text{if } T(x_t) > \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where x_t is the tank state and

$$T(x_t) = \frac{1}{2}x_t^2. \tag{6}$$

In (5), $\widetilde{K}_p$ is defined as

$$\widetilde{K}_p = \text{diag}(k_{i_{\max}} - k_{i_{\min}})\,\xi(t), \quad i = 1,\ldots,6 \tag{7}$$

with $k_{i_{\max}}$ the maximum compliance and $\xi(t) \in [0,1]$ a control input that shapes impedance between $k_{i_{\min}}$ and $k_{i_{\max}}$. The evolution of $\xi(t)$ is set during kinesthetic teaching using an sEMG and vibrotactile interface (see next subsection and Section II-C). The tank is initialized as $x_t(0) = x_{\text{initial}}$ and always maintained above $x_{t_{\min}}$ to avoid singularities. From (1), the tank dynamics is defined as

$$\begin{cases} \dot{x}_t = \frac{\sigma}{x_t}\left(\dot{x}^\top K_d \dot{x}\right) + \omega^\top \dot{x} \\ y = -\left(\dot{x}^\top x_t\right)^\top \end{cases} \tag{8}$$

with $\dot{x}_d = 0$ (without loss of generality), and where the activation variable σ is

$$\sigma = \begin{cases} 1, & \text{if } T(x_t) \leq \bar{T} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

which limits the energy in the tank to prevent nonpassive conditions. The total stored energy is

$$W(x, \dot{x}, x_t) = V(x, \dot{x}) + T(x_t) \tag{10}$$

with $T(x_t)$ from (6), and $V(x, \dot{x})$ a Lyapunov function for the mechanical impedance

$$V(x, \dot{x}) = \frac{1}{2}\dot{x}^\top M(x)\dot{x} + \frac{1}{2}(x_d - x)^\top \bar{K}_p(x_d - x). \tag{11}$$

From (10) and (11), the derivative of the storage function is

$$\dot{W} = \dot{x}^\top F_h - (1-\sigma)\dot{x}^\top K_d \dot{x} \tag{12}$$

which satisfies the passivity condition

$$W(t) - W(0) \leq \int_0^t \dot{x}^\top(\tau) F_h(\tau)\, d\tau. \tag{13}$$

Thus, ensuring that the system defined by (1) and (8) is passive with respect to the input pair (F_h, y), and therefore stable [33]. Finally, during the reproduction of the programmed trajectory and associated point-wise compliance, physical interaction by the human operator, as in the collaborative tasks considered in this study and discussed in the following sections, may lead to trajectory–compliance desynchronization. This issue arises because the compliance profile is time-aligned with the reference trajectory, and any significant deviation due to interaction can result in a mismatch between the expected impedance behavior and the actual robot configuration. To address this, a synchronization monitoring mechanism is implemented: if, at any time t_e, the Euclidean distance between the current robot position $x(t)$ and the corresponding reference trajectory point $x_d(t)$ exceeds a threshold $\delta_{\max} = 0.4$ m, the point-wise update of the reference trajectory is paused. The update resumes only once the robot returns within the allowed threshold. Formally, this condition is defined as

$$x_d(t_e + \Delta t) = \begin{cases} x_d(t_e), & \text{if } \|x(t) - x_d(t)\| > \delta_{\max} \\ \text{desired evolution}, & \text{otherwise.} \end{cases} \tag{14}$$

This mechanism ensures temporal coherence between trajectory progression and compliance adaptation, thus preserving interaction safety and control consistency even under cooperative execution scenarios.

3) Wearable sEMG Sensing and Vibrotactile Stimulation: The wearable interface consisted of two armbands. An 8-channel gForcePro sEMG armband (OYMotion) on the forearm acquired signals from the FDS (*flexor digitorum superficialis*) and EDC (*extensor digitorum communis*) muscles to estimate hand cocontraction (see Fig. 2). It required only clean, dry skin and no precise placement. Raw data streamed at 1 kHz was filtered with a 50 Hz notch filter, a 20 Hz high-pass filter, and root mean square (rms) over a 200 ms window. Vibrotactile feedback was delivered via a custom bracelet worn on the participant's upper arm, incorporating a grove vibration motor module from Seeed Technology Company, Ltd. This module includes a coin-type eccentric rotating mass motor, capable to generate clearly perceptible vibratory stimulus. The motor was controlled by a microcontroller that received high-level commands and modulated the vibration intensity through pulsewidth modulation), varying the duty cycle to adjust the average voltage applied to the motor within its operational range (0–5 V). Since the vibrator was on the upper arm, far from the electrodes, any sEMG artifact is avoided. Although the actuator was worn over a layer of clothing, the vibratory signals were reported by all participants to be clearly perceivable. This was consistent with the actuator's specifications, which indicate a vibration frequency of approximately 150 Hz at full drive, providing sufficient vibration for reliable perception through fabric. In our implementation, cocontraction levels were mapped linearly to the vibration intensity range (as detailed later), from low perceptible levels to more intense vibrations, allowing participants to intuitively interpret the feedback and modulate their muscular activation accordingly during the teaching phase (see Fig. 2).

C. Estimation of Cocontraction and Vibrotactile Feedback

This section describes an algorithm to estimate hand cocontraction from forearm sEMG, leveraging neuromuscular principles. Muscle synergies coordinate motor tasks via shared neural drives [34], [35], and MUs coactivate during antagonist contractions [36]. These concepts guide cocontraction estimation and vibrotactile feedback modulation. The proposed estimation of cocontraction relies on an unsupervised decomposition of sEMG signals via nonnegative matrix factorization. Unlike supervised methods, which require ground truth labels that are impractical or undefined for cocontraction, this approach enables intuitive and robust estimation without compromising system usability or requiring intrusive calibration.

1) Neurophysiology-Based Hand Cocontraction Interface: Given a pair of neural drives, their effect on joint motion can be modeled using a *kinematic synergy matrix* P

$$\theta(t) = P \begin{bmatrix} \alpha_E(t) \\ \alpha_F(t) \end{bmatrix} \tag{15}$$

where $P \in \mathbb{R}^{1\times 2}$ in the simplified setup of Fig. 3(a). To produce $\theta(t)$, muscle activations $a_E(t)$, and $a_F(t)$ are needed and modeled as a linear combination via the *muscular synergy matrix* $M \in \mathbb{R}^{2\times 1}$

$$\begin{bmatrix} a_E(t) \\ a_F(t) \end{bmatrix} = M\theta(t). \tag{16}$$

Assuming negligible crosstalk, the sEMG signal for each muscle is modeled as a sum of MU activity convolved with

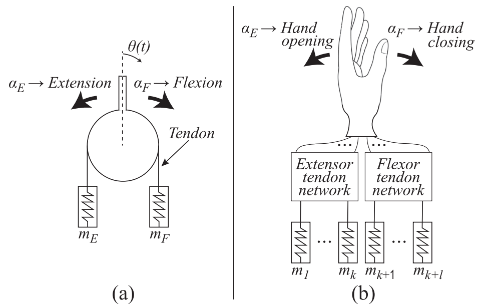


Fig. 3. Scheme of the musculoskeletal systems considered for the estimation of the hand co-contraction level. (a) Simplified single-DoF system; (b) human hand musculoskeletal system. The notation is in accordance with Section II-C.

volume conduction filters [37]

$$e_i(t) = \sum_{j=1}^{N_i} g_{ij}(t) * f_{ij}(a_i(t)) \tag{17}$$

where $i = \{E, F\}$, N_i is the number of MUs in m_i, $f_{ij}(a_i(t))$ is the MU firing response, and $g_{ij}(t)$ the associated filter. The rms value $e_i^{\mathrm{RMS}}(t)$ over a window T is proportional to $a_i(t)$ [38], so

$$\begin{bmatrix} e_E^{\mathrm{RMS}}(t) \\ e_F^{\mathrm{RMS}}(t) \end{bmatrix} = H \begin{bmatrix} a_E(t) \\ a_F(t) \end{bmatrix} = \mathrm{diag}(h_E, h_F) \begin{bmatrix} a_E(t) \\ a_F(t) \end{bmatrix} \tag{18}$$

where H is the *activation-to-EMG mixture matrix*. Substituting (15) and (16) into (18) yields

$$\begin{bmatrix} e_E^{\mathrm{RMS}}(t) \\ e_F^{\mathrm{RMS}}(t) \end{bmatrix} = HMP \begin{bmatrix} \alpha_E(t) \\ \alpha_F(t) \end{bmatrix} = W \begin{bmatrix} \alpha_E(t) \\ \alpha_F(t) \end{bmatrix} \tag{19}$$

with $W = HMP$ as the *drive-to-EMG mixture matrix*. Thus, $\alpha_E(t)$ and $\alpha_F(t)$ can be estimated from $e_E^{\mathrm{RMS}}(t)$ and $e_F^{\mathrm{RMS}}(t)$ using unsupervised linear decomposition, enabling computation of $c(t)$. We now extend (15)–(19) to the real-case scenario in Fig. 3(b) for hand cocontraction estimation.

We now extend the estimation model to the human hand system of Fig. 3(b). To estimate hand muscular cocontraction from forearm sEMG during voluntary stiffening, we define dominant neural drives for finger extension and flexion as $\widehat{\alpha_E}(t)$ and $\widehat{\alpha_F}(t)$, and define hand cocontraction as

$$\widehat{c}(t) = \min(\widehat{\alpha_E}(t), \widehat{\alpha_F}(t)). \tag{20}$$

Given the hand's n DoF, actuated by k extensor and l flexor muscles, finger motion can be modeled as

$$[\theta_1(t) \cdots \theta_n(t)]^T = \widehat{P} \begin{bmatrix} \widehat{\alpha_E}(t) \\ \widehat{\alpha_F}(t) \end{bmatrix} \tag{21}$$

with $\widehat{P} \in \mathbb{R}^{n \times 2}$ the kinematic synergy matrix. The muscle activations $a_1(t), \ldots, a_{k+l}(t)$ from the hand muscles $m_1(t), \ldots, m_{k+l}(t)$ are then

$$[a_1(t) \cdots a_{k+l}(t)]^T = \widehat{M} \begin{bmatrix} \theta_1(t) \\ \vdots \\ \theta_n(t) \end{bmatrix} \tag{22}$$

with $\widehat{M} \in \mathbb{R}^{(k+l) \times n}$ the muscular synergy matrix.

Differently from the simplified case of (17), forearm sEMG signals are affected by crosstalk, so the raw signal at electrode $s = 1, \ldots, 8$ becomes:

$$e_s(t) = \sum_{i=1}^{k+l} \sum_{j=1}^{N_i} g_{ij}^{(s)}(t) * f_{ij}(a_i(t)). \tag{23}$$

The corresponding rms sEMG signal, accounting for cross-terms [39], is

$$e_s^{\mathrm{RMS}}(t) = \sum_{i=1}^{k+l} e_{si}^{\mathrm{RMS}}(t) + \sum_{i=1}^{k+l} \sum_{\substack{h=1 \\ h \neq i}}^{k+l} \overline{e_{si}(t) e_{sh}(t)} \tag{24}$$

where $e_{si}^{\mathrm{RMS}}(t)$ is due to muscle i, and $\overline{e_{si}(t) e_{sh}(t)}$ is the cross-product root mean value. As in [38], assuming low correlation among $e_{si}(t)$ and $e_{sh}(t)$, we neglect cross-terms and write the rms sEMG vector $E(t) = [e_1^{\mathrm{RMS}}(t) \cdots e_8^{\mathrm{RMS}}(t)]^T$ as

$$E(t) = \widehat{H}\,[a_1(t) \cdots a_{k+l}(t)]^T \tag{25}$$

with $\widehat{H} \in \mathbb{R}^{8 \times (k+l)}$ the activation-to-EMG mixture matrix. Finally, substituting (21) and (22) into (25) yields

$$E(t) = \widehat{H}\widehat{M}\widehat{P} \begin{bmatrix} \widehat{\alpha_E}(t) \\ \widehat{\alpha_F}(t) \end{bmatrix} = \widehat{W} \begin{bmatrix} \widehat{\alpha_E}(t) \\ \widehat{\alpha_F}(t) \end{bmatrix}, \tag{26}$$

with $\widehat{W} = \widehat{H}\widehat{M}\widehat{P} \in \mathbb{R}^{8 \times 2}$ the drive-to-EMG mixture matrix. As in the simplified case, $\widehat{\alpha_E}(t)$ and $\widehat{\alpha_F}(t)$ can be extracted using an unsupervised linear decomposition algorithm, such as nonnegative matrix factorization (NMF), as described next.

2) Computation of the Drive-to-EMG Mixture Matrix $\widehat{W}$: The drive-to-EMG matrix $\widehat{W}$ in (26) is computed offline by collecting rms sEMG data $E_{\mathrm{offline}} \in \mathbb{R}^{8 \times d}$ from an eight-channel armband (see Section II-B3) during four hand open-close cycles: two with minimal, two with maximal hand stiffening. This captures various stiffness levels. Thus

$$E_{\mathrm{offline}} = \widehat{W} A_{\mathrm{offline}} \tag{27}$$

with $A_{\mathrm{offline}} = [\widehat{\alpha_E}^T\ \widehat{\alpha_F}^T]^T \in \mathbb{R}^{2 \times d}$. Applying NMF [40] to (27) yields both $\widehat{W}$ and A_{offline}, from which scaling factors are defined

$$\gamma_E = \max(\widehat{\alpha_E}_i), \quad \gamma_F = \max(\widehat{\alpha_F}_i). \tag{28}$$

3) Online Cocontraction Level Estimation and Feedback: Using $\widehat{W}$ and (28), the online neural drives $A(t) = [\widehat{\alpha_E}(t)\ \widehat{\alpha_F}(t)]^T$ are estimated as

$$A(t) = \Gamma\, \widehat{W}^{+} E(t) \tag{29}$$

with $\widehat{W}^{+}$ the pseudoinverse of $\widehat{W}$, and $\Gamma = \mathrm{diag}(1/\gamma_E, 1/\gamma_F)$. The online cocontraction level $\widehat{c}(t)$ is obtained from (20), and the vibrotactile feedback command is

$$\eta(t) = (\rho_{\max} - \rho_{\min})(1 - \widehat{c}(t)) + \rho_{\min} \tag{30}$$

where $\rho_{\max}$ and $\rho_{\min}$ are the vibrotactile intensity bounds. Thus, the operator increases hand stiffness to reduce vibration, guiding cocontraction toward desired levels during kinesthetic teaching.

The cocontraction profile $\hat{c}(t)$ is recorded offline and reused as impedance modulation input $\xi(t)$ during task execution (see Section II-B1), i.e., $\xi(t) = \hat{c}(t)$. As introduced in (7), this scalar value $\xi(t) \in [0, 1]$ modulates the robot's Cartesian stiffness via a linear interpolation between minimum and maximum directional stiffness values ($k_{i_{\min}}$ and $k_{i_{\max}}$). This modulation defines the mechanical impedance profile along each Cartesian axis, shaping the robot's compliant behavior during execution. There is therefore a mapping between the user's muscular cocontraction (as estimated from the EMG signal) and the commanded impedance levels. These impedance levels, in turn, influence the interaction force outcomes indirectly through the robot-environment dynamics during physical contact.

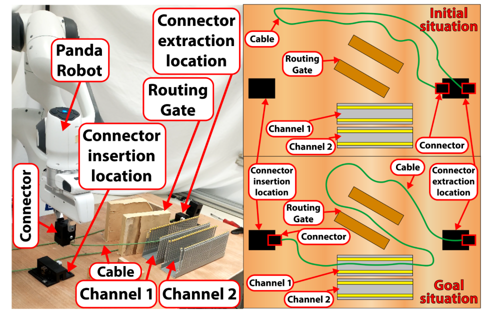
4) State Machine for Reliable Compliance Programming: Although hand cocontraction is largely decoupled from upper limb motion—unlike full-arm impedance modulation—it may still be affected by unintentional muscle activations. To ensure that only deliberate input modulates the robot's behavior, a simple state machine governs the interaction logic. By default, the robot remains in a nonprogramming state with fixed minimum compliance, allowing free cocontraction without effect. The user enters the programming mode by performing a specific activation pattern: two short spikes and a sustained cocontraction (above 0.6) for at least three seconds. During this transition, vibrotactile pulses indicate successful threshold crossings, while in programming mode, continuous vibration—proportional to cocontraction—enables real-time compliance shaping. Exiting the mode requires repeating the same pattern. The final compliance value is set not at exit, but corresponds to the level maintained three seconds prior, allowing intentional selection. This mechanism promotes robust, user-controlled transitions, mitigating unintended effects and enabling reliable point-by-point compliance programming.

III. Experiment

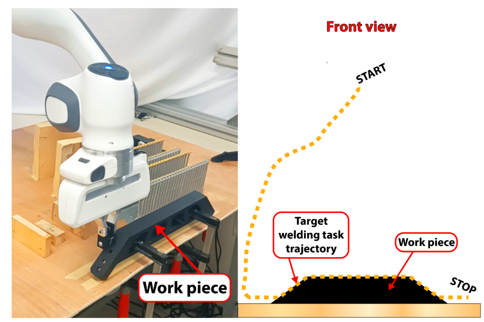
A. Description of the Experimental Protocol

1) Group of Subjects: A total of ten healthy adult volunteers (one female, nine males; mean age: 27.2 ± 1.8 years; range: 25–30.5) participated in the experimental evaluation. None had prior experience with the robotic setup or similar kinesthetic teaching interfaces. Eight participants were right-handed and two left-handed. All subjects provided written informed consent prior to participation, in compliance with the declaration of Helsinki. The study preserved anonymity, referring to participants as S1 through S10.

2) Experimental Tasks: Two experimental tasks were designed to evaluate the effectiveness of the proposed programming-by-demonstration framework: a robotic wiring task and a welding-like task. Both required kinesthetic demonstrations of full trajectories involving position, orientation, and environmental contact forces. In the *robotic wiring task*, participants performed a cable-handling sequence on a custom switchgear mock-up (see Fig. 4). The sequence included extracting a connector from its housing, passing the cable through a routing gate, guiding it along a constrained channel (channel 1), and inserting it into a second connector slot. Both insertion and extraction points featured spring-loaded ratchet mechanisms with passive compliance, facilitating connector handling and tolerance to alignment errors—particularly beneficial when the robot was programmed with higher compliance. The routing geometry required careful modulation of both position and orientation. The *welding-like task* simulated a simplified welding operation on a rigid trapezoidal-profile workpiece (also in Fig. 4). The operator demonstrated a trajectory over its entry incline, central flat surface, and exit slope, maintaining continuous contact while minimizing interaction forces. Excessive contact force could induce undesirable effects, such as jerky motion or blockage due to friction. During autonomous execution, impedance control gains were modulated according to the programmed compliance levels. Specifically, for maximum stiffness (i.e., minimum compliance), the gains were set to $K_p = 800$ and $K_d = 45$, while for minimum stiffness (i.e., maximum compliance), they were reduced to $K_p = 200$ and $K_d = 15$. Intermediate levels were linearly interpolated. These values were selected through preliminary empirical tuning sessions, in which interaction forces and motion smoothness have been observed based on the practical experience of the experimenters, ensuring stable and responsive behavior across the full range of compliance level. The control architecture was implemented on a PC equipped with an Intel Core i7 11th generation processor. Communication with the Franka Emika Panda robot was established using the `libfranka` C++ library, which provides a low-level interface for real-time control at a frequency of 1 kHz and access to the robot's state and dynamics. Post-processing and data analysis were performed using Python and MATLAB.



(a)



(b)

Fig. 4. Setup for the robotic (a) wiring and (b) welding-like task experiments (see Section III), with schematic exemplifications for the required robot programming.

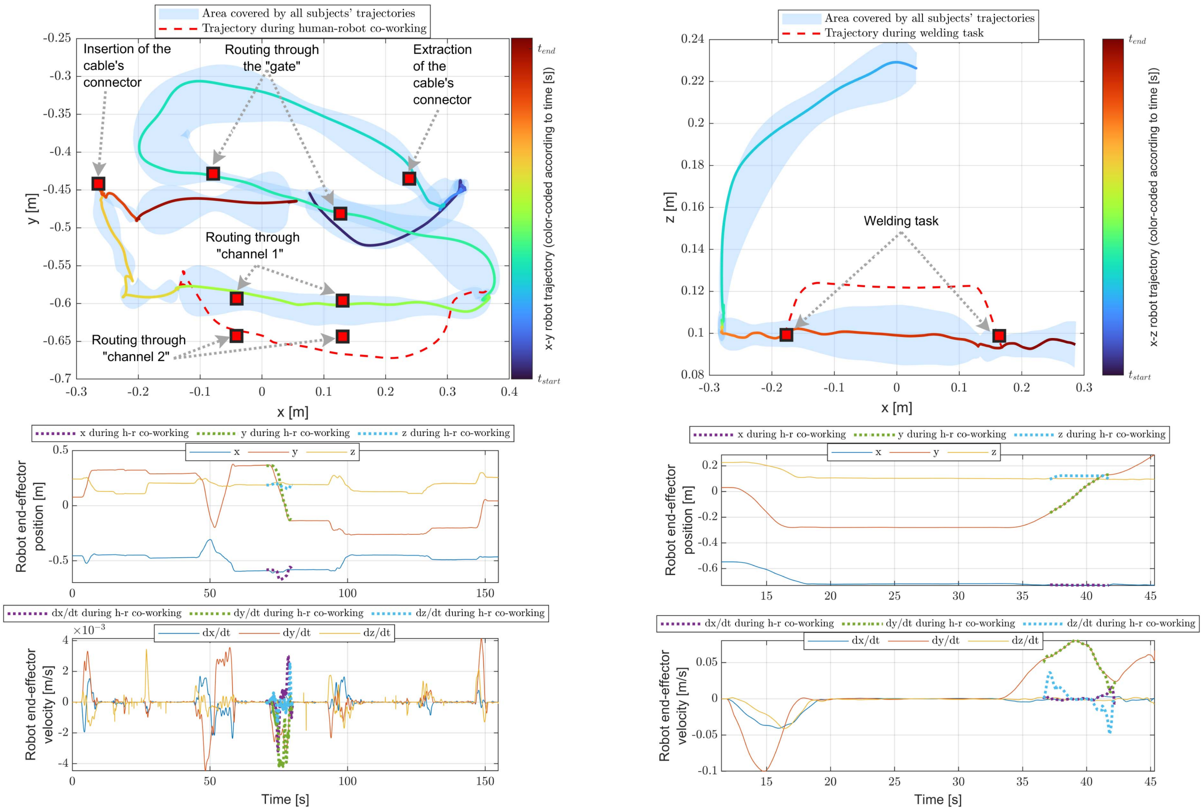
3) Cocontraction Modulation Protocol: Subjects were instructed to modulate hand muscle cocontraction to dynamically program robot compliance throughout the tasks. Kinesthetic teaching was performed by grasping the robot end-effector with both hands at all times. Prior to the experimental session, a 15-min familiarization phase was conducted. Participants wore a vibrotactile armband alternately on both forearms and practiced associating three vibration intensities with distinct cocontraction levels: high (high compliance), medium (intermediate compliance), and low (high stiffness), to ensure intuitive control during demonstration. During the wiring task (see Fig. 4), participants adjusted compliance according to subtask demands. High compliance was used for connector insertion and extraction, to accommodate positioning errors and minimize contact forces with spring-loaded elements. Medium compliance was required when routing inside channel 1, enabling safe human-robot physical interaction during execution in channel 2, while maintaining guidance along the channel geometry. All other segments were demonstrated with high stiffness. In the welding-like task, high stiffness was used throughout, except for the contact segment against the trapezoidal surface, where medium compliance was required to ensure stable force application and avoid stick-slip or excessive normal forces. Gripper actuation was controlled hands-free via speech using the `SpeechRecognition` Python library, with the Google Web Speech API selected for accuracy. An environmental microphone (Jabra Speak 510) placed near the workspace continuously listened for the keywords *"open"* and *"close,"* mapped to control the gripper.

4) Interaction Behaviors Evaluation: Following the kinesthetic teaching phase, we evaluated the robot's interaction behavior during trajectory reproduction under two conditions: 1) with compliance adaptation and 2) without compliance adaptation (i.e., fixed high stiffness). Both conditions were tested across the two experimental tasks: the robotic wiring task and the welding-like task. In the wiring task (see Fig. 4), interaction evaluation focused on two aspects. First, we considered the nonpredetermined physical interaction that occurred during the routing phase: in a dedicated coworking scenario, participants deliberately modified the playback trajectory by physically guiding the robot to reroute the cable from channel 1 to channel 2. Second, we analyzed the contact forces generated during interaction with specific elements of the workbench, particularly the connector extraction and insertion locations. These phases involved mechanical coupling with the ratchet-based fixtures, where compliance was expected to reduce peak forces. In the welding-like task, evaluation focused on the interaction between the robot and the work piece during the mock-up welding phase. Specifically, we examined the forces exchanged while the robot applied pressure on the inclined and horizontal surfaces of the trapezoidal workpiece, comparing behavior across the compliant and noncompliant conditions. Quantitative analysis of the interaction forces and behavior recorded during these scenarios is provided in Section III-B2.

B. Results

This section presents the experimental results obtained from both the wiring and welding-like tasks. We analyze the effectiveness of the proposed framework in terms of trajectory programming, compliance modulation, and interaction behavior during task execution. To quantify differences in modulated cocontraction levels and robot interaction forces across experimental conditions, statistical analysis was conducted, using one-way ANOVA where assumptions of normality and sphericity were verified and satisfied. The underlying paradigm evaluated in this study is that of one-shot PbD, wherein each user provides a single, direct demonstration of the task, with the objective of eliciting an immediate and accurate autonomous replication by the robot. This "one-shot" approach is inherently challenging and of particular interest, as it mirrors realistic deployment scenarios where repeated training or iterative refinement may be undesired. Consequently, in our experimental design, each subject was instructed to provide only one demonstration per task condition, followed by a single corresponding autonomous repetition executed by the robot. The study does not rely on multiple demonstrations per user or multiple autonomous executions per demonstration, as our aim is to assess system performance in the "one-shot" setting. Due to this design, the statistical evaluation does not derive from intrasubject variance over multiple trials, but rather from intersubject variability in one-shot demonstrations and replications. Specifically, for each experimental condition (e.g., with and without compliance modulation), a set of $N = 10$ independent samples was obtained and used in the ANOVA, corresponding to the ten distinct subjects participating in the study. This methodological choice ensures that the statistical findings reported herein are directly representative of system behavior under the constraints and variability naturally arising from one-shot human–robot programming scenarios. Further details on task conditions and individual outcome metrics are discussed in the following subsections.

1) Trajectory and Compliance Programming: Fig. 5 displays the end-effector trajectories programmed by participants for the wiring task (projected on the x–y plane) and the welding-like task (on the z–y plane). The colored line shows subject S1's trajectory, with cyan-shaded regions indicating the total workspace coverage across all subjects. For S1, corresponding position and velocity profiles are also shown, aligned with task landmarks from Fig. 4. Fig. 6 illustrates S1's cocontraction behavior during wiring. Entry into compliance programming mode is marked by a characteristic pattern—two rapid threshold crossings followed by sustained contraction—activating a modulation window (yellow), and exited using the same pattern (red). Compliance is then fixed based on the level at the beginning of the preceding cyan segment. The middle panel shows vibrotactile feedback: discrete cues indicate threshold crossings outside programming, while continuous vibration reflects real-time co-contraction during modulation, supporting user awareness. The bottom panel confirms corresponding compliance shaping on the robot: stiffness starts high, is modulated during interaction, and reset to high afterward as per protocol. Across subjects, the intended compliance levels were consistently achieved: high during insertion/extraction, medium during routing or welding contact, and low elsewhere. Aggregated cocontraction levels during these phases were: high—0.88 ± 0.07, medium—0.58 ± 0.06, and low—0.21 ± 0.05. A one-way repeated-measures ANOVA confirmed significant differences between the three co-contraction



(a) (b)

Fig. 5. Taught and resulting trajectories for the (a) wiring and (b) welding-like tasks. In the top graphs of both columns (a) and (b), solid color-coded lines represent the trajectories manually programmed by subject S1, while dashed lines indicate variations resulting from interactions with the operator or test-bench components. The cyan-shaded areas highlight the workspace regions covered by the trajectories taught by all subjects. Bottom graphs show the position and velocity profiles corresponding to the trajectory taught by subject S1. Indicated testbench's locations refer to those indicated in Fig. 4.

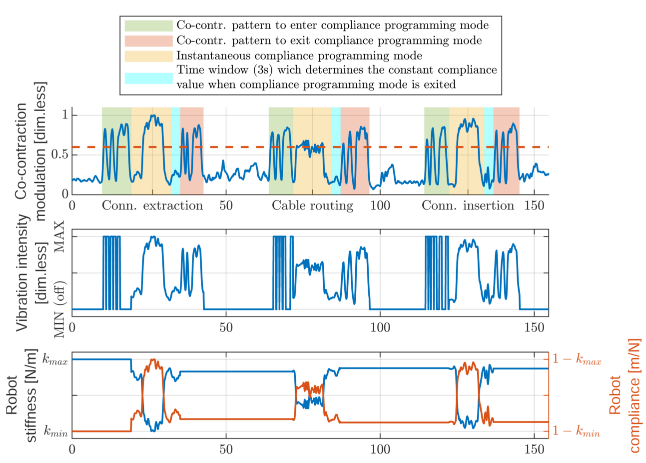


Fig. 6. Cocontraction modulation and corresponding vibration feedback intensity during wiring task kinesthetic teaching for subject S1.

levels, $F(2, 18) = 152.4$, $p < 0.001$, supporting the effectiveness and repeatability of the cocontraction-based compliance programming strategy. Assumption checks confirmed normality (Shapiro–Wilk test, $p > 0.05$ for all conditions) and sphericity (Mauchly's test, $p = 0.42$) for the use of repeated-measures ANOVA.

2) Online Task and Interaction Behavior: In the following, we report the outcome of the online executions of both the robotic wiring and welding-like tasks, performed based on the offline kinesthetic programming sessions previously carried out by each participant, as described in Section III-A. Each task was executed autonomously by the robot, reproducing the subject-specific programmed trajectories and compliance profiles. During these executions, human–robot physical coworking was intermittently introduced by the experimenter, who randomly requested the operator to intervene and alter the cable routing path. In these cases, the user leveraged the preprogrammed interaction behavior of the robot—specifically its compliance modulation capabilities—to perform an online reconfiguration of the path, in accordance with the experimental protocol defined in Section III-A3. Similarly, in the welding-like task, the robot executed the trajectory involving surface contact over the trapezoidal-profile work piece. The objective was to apply pressure along the surface. In the case without impedance adaptation, excessive contact forces occasionally resulted in jerky motions or execution halt due to friction-induced blocking. To assess the effect of compliance programming, both tasks were repeated under two conditions: with and without the implementation of the variable impedance dynamics. This comparison allowed us to evaluate the role of cocontraction-based compliance

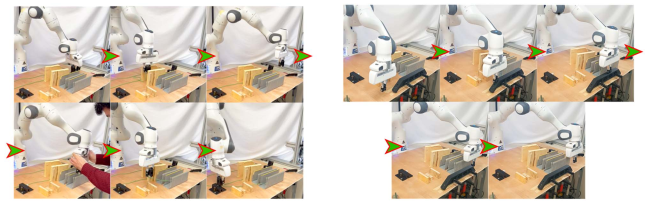


Fig. 7. Frame sequence of the online execution of the robotic wiring and welding-like tasks. Wiring task (left), welding-like task (right), and contact interaction with the work piece is observable.

modulation in enabling safer and adaptable physical interaction during task execution.

3) Online Task With Variable Impedance: This experimental session aimed to evaluate the role of variable impedance dynamics during automatic task execution in the presence of both predetermined and nonpredetermined physical interactions. The results align with the protocol detailed in Section III-A4 and provide insight into the robot's behavior in contact-rich situations for both the wiring and welding-like tasks.

Interaction during contact with environment: For both tasks, we analyzed the forces exchanged between the robot and the environment during contact phases. In the wiring task, this included the extraction and insertion of the cable connector, which involved mechanical coupling with the ratchet-based fixtures (see Fig. 4). In the welding-like task, interaction was evaluated during pressure application on the workpiece surface. As shown in Fig. 8, the use of compliance programming significantly reduced peak forces and ensured stable and smoother contact behavior. Notably, in the welding-like task without impedance adaptation, the robot occasionally halted execution due to excessive friction-induced forces.

On-the-Fly modification of cable routing: During a dedicated online execution of the wiring task, the experimenter instructed subjects to modify the cable routing by guiding the robot from the originally programmed path to the alternative channel 2, as illustrated in Fig. 4. All subjects successfully completed this on-the-fly reconfiguration on their first attempt by leveraging the preprogrammed compliant behavior. The effectiveness of this human–robot coworking phase is illustrated in Figs. 5, 7, and 8, showing smooth physical interaction between operator and robot.

4) Online Task Without Variable Impedance: In this experimental session, the robot executed the programmed trajectories without compliance modulation (i.e., constant, high stiffness).

Interaction during contact with environment: For the wiring task, interaction was evaluated during the phases of connector extraction and insertion, which require the robot to engage with spring-loaded ratchet mechanisms. In the welding-like task, interaction occurred while the robot applied pressure along the surface of the rigid trapezoidal-profile work piece. Fig. 8(a) and (b) shows representative force profiles for subject S1. In both cases, the absence of impedance adaptation led to substantially increased contact forces. In the welding task in particular, this

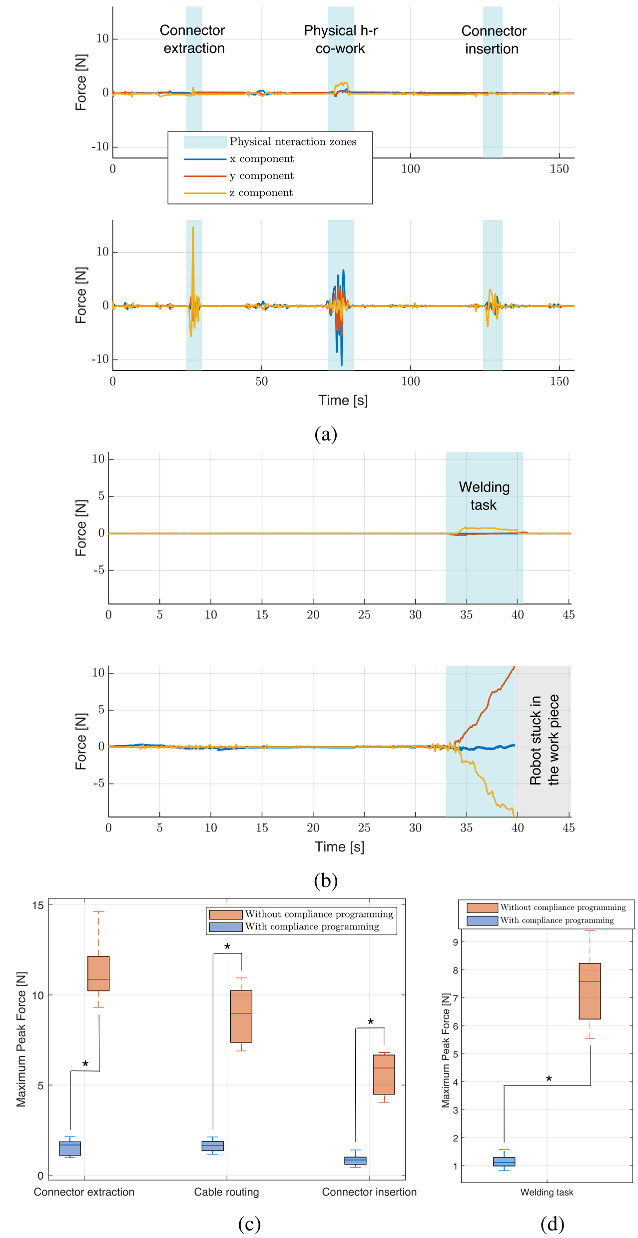


Fig. 8. Interaction forces at the robot end-effector during automatic execution. (a) Wiring task—force components for subject S1 with (top) and without (bottom) compliance programming, showing effects during connector handling and human–robot coworking. (b) Welding-like task—force components for subject S1 with (top) and without (bottom) compliance programming; in the noncompliant case, force escalation leads to execution halt due to excessive friction. (c) and (d) Boxplots of absolute maximum peak force components for the wiring and welding tasks across all subjects/repetitions (* means $p < 0.001$).

resulted in escalating force values due to excessive friction, halting the execution.

Quantitative comparison of force levels: A quantitative comparison of the interaction forces across all subjects and repetitions is shown in Fig. 8(c)–(d), which reports boxplots of the absolute values of peak force components grouped by condition (with versus without compliance programming). For the wiring task [see Fig. 8(c)], the peak force levels were

TABLE I
COMPACT COMPARISON ACROSS TASK TYPES AND EXPERIMENTAL CONDITIONS

Task Specificity	Compliance Programming	Interaction Force (N, avg ± std)	Success Rate (%)
Connector insertion	With	1.2 ± 0.3	100
	Without	5.5 ± 1.5	30
Connector extraction	With	1.8 ± 0.4	100
	Without	11 ± 1.8	0
Cable rerouting	With	2.0 ± 0.4	100
	Without	9.5 ± 2	0
Welding contact with work piece	With	1.2 ± 0.3	100
	Without	7.5 ± 1.5	60
Entering the compliance programming mode	With vibrotactile feedback on right arm		100
Exiting the compliance programming mode	With vibrotactile feedback on the left arm		100
Either entering or exiting compliance programming mode	Without vibrotactile feedback		25
Execution time difference with compliance programming versus without compliance programming: < 0.1 s (not significant)			

TABLE II
SUBJECTIVE EVALUATION OF THE PROGRAMMING FRAMEWORK ACROSS THREE INTERFACE MODALITIES

Outcome Type	Questionnaire Item (by condition)	Average Score ($\pm$ std dev.)
Perceived ease of use (PE)	*(EMG + Vibrotactile) It was easy to interpret the vibrotactile feedback to the aim of hand co-contraction modulation during kinesthetic teaching.*	6.5 $\pm$ 0.53
	(EMG Only) It was easy to understand the EMG-based control while programming the robot.	4.0 $\pm$ 1.22
	(Tablet Bar Only) It was easy to modulate the bar on the tablet to program the robot.	3.8 $\pm$ 1.72
Perceived usefulness (PU)	*(EMG + Vibrotactile) I think the provided interfacing with the robot is useful to program the impedance dynamics during kinesthetic teaching.*	6.6 $\pm$ 0.50
	(EMG Only) I think the EMG interface is useful to program the robot during kinesthetic teaching.	3.0 $\pm$ 0.89
	(Tablet Bar Only) I think the tablet-based bar is useful to program the robot during kinesthetic teaching.	2.6 $\pm$ 1.14
Cognitive comfort (CC)	*(EMG + Vibrotactile) It was comfortable in terms of perceived complexity and mental effort to program the robot with the proposed framework.*	6.2 $\pm$ 0.63
	(EMG Only) I felt cognitively comfortable using the EMG interface.	3.7 $\pm$ 1.21
	(Tablet Bar Only) I felt cognitively comfortable using the tablet bar to program the robot.	2.7 $\pm$ 1.97
Perceived physical effort (PPE)	*(EMG + Vibrotactile) I did not experience significant physical fatigue or discomfort while modulating co-contraction during the task.*	6.1 $\pm$ 0.60
	(EMG Only) I did not feel physical strain using the EMG interface during the task.	4.0 $\pm$ 0.89
	(Tablet Bar Only) I did not feel physical strain using the tablet bar to control the robot.	3.6 $\pm$ 2.50
Responsiveness (R)	*(EMG + Vibrotactile) I was able to promptly modulate my co-contraction as desired during the programming phase.*	6.5 $\pm$ 0.51
	(EMG Only) I was able to promptly vary the EMG-based modulation.	2.4 $\pm$ 0.89
	(Tablet Bar Only) I was able to promptly vary the tablet bar value.	2.8 $\pm$ 1.30
Confidence in control (CC2)	*(EMG + Vibrotactile) I felt confident that my modulation of co-contraction was accurately according to my intention.*	6.3 $\pm$ 0.56
	(EMG Only) I felt confident in programming the robot using EMG signals.	2.2 $\pm$ 1.17
	(Tablet Bar Only) I felt confident using the tablet bar to program the robot.	3.8 $\pm$ 2.17
Perceived integration (PI)	*(EMG + Vibrotactile) The overall interface felt well integrated into the robot programming experience.*	6.6 $\pm$ 0.48
	(EMG Only) The EMG interface felt well integrated into the robot programming process.	3.2 $\pm$ 1.17
	(Tablet Bar Only) The tablet bar interface felt well integrated into the robot programming process.	2.8 $\pm$ 1.79

Scores: likert scale from 1 (entirely disagree) to 7 (entirely agree).

statistically significantly lower in the presence of compliance programming. Specifically, a one-way ANOVA was conducted, according to which statistically significant differences were found: $F(1,18) = 318.77$, $p < .001$ (connector extraction), $F(1,18) = 213.52$, $p < .001$ (cable routing), and $F(1,18) = 167.64$, $p < .001$ (connector insertion), indicating that compliance programming effectively reduces interaction forces during contact. For the welding-like task [see Fig. 8(d)], similar trends were observed. A one-way ANOVA revealed a statistically significant difference in peak force values between the two conditions, with $F(1,18) = 276.41$, $p < .001$, confirming the benefits of programmed compliance even in a surface contact scenarios. In all cases, the assumptions of normality (Shapiro–Wilk test, $p > 0.05$) and sphericity (Mauchly's test, $p > 0.05$) were satisfied. These results confirm that the absence of compliance modulation leads to higher interaction forces and less safe physical interaction.

5) Subjective Usability and Experimental Conditions: To better evaluate usability and user experience across all experimental conditions, participants were asked to repeat selected programming tasks under varying interface configurations, including with and without vibrotactile feedback. These additional tasks were designed to reflect the main task specificities reported in Table I, particularly those involving cocontraction control and physical interaction. In the same table, we also report the success rates observed for each task under both compliance programming conditions. For connector insertion, extraction, and cable rerouting, a task execution was marked as a failure if the peak interaction force exceeded five times the value measured when compliance programming was active. In the welding task, failure was defined instead by mechanical blocking of the robot's movement due to excessive resistance at contact. This criterion reflects typical failure modes encountered in unadapted interaction behavior. As shown in the table, compliance programming leads to a significant improvement in task success. For instance, in the extraction task, where the robot must "pull" a connector out of its housing, the absence of compliance results in failure in 100% of the trials, due to excessive peak forces. Insertion is more variable, as different users programmed slightly different trajectories; in some cases, these trajectories allowed successful insertions in a few cases even without active compliance. Similarly, cable rerouting without compliance consistently exceeded the acceptable force thresholds. In the welding contact task, the success rate still drops evidently without compliance. At the end of the study, participants completed a questionnaire assessing seven key dimensions related to subjective usability and cognitive effort. Each item was rated on a seven-point Likert scale (1: strongly disagree; 7: strongly agree). To enable comparison and provide a baseline reference, participants performed the same tasks under three distinct interface configurations: the full framework (EMG + vibrotactile feedback), EMG-based control without vibrotactile feedback, and a baseline condition using a graphical bar on a tablet without any physiological interface. The statements covered aspects of perceived ease of use, usefulness, cognitive and physical effort, system responsiveness, confidence in control, and integration of the interface components. Results are presented in Table II, which reports average scores and standard deviations for each dimension and condition. A clear trend emerges in favor of the full EMG + vibrotactile interface, which consistently receives the highest ratings across all metrics. In contrast, both the EMG-only and tablet-based baseline interfaces yield substantially lower scores, particularly in dimensions related to responsiveness, confidence in control, and perceived integration. This comparison highlights the added value of the vibrotactile feedback in supporting users' interpretation and modulation of cocontraction levels, resulting in a more intuitive and coherent programming experience. Notably, the vibrotactile cue proved also crucial in helping users correctly recognize and execute transitions in and out of the impedance programming mode—an aspect further supported by the success rates reported in Table I.

IV. Conclusion

In this work, we introduced a novel programming-by-demonstration framework for collaborative robots that enables simultaneous kinesthetic trajectory teaching and intuitive programming of interaction behavior through a wearable sEMG interface and vibrotactile feedback. Using unsupervised cocontraction estimation, users modulated robot impedance by adjusting hand stiffness. The system was validated with 10 users on two representative tasks—cable wiring and welding-like surface contact—showing that compliance programming markedly improved success rates. Wiring tasks were completed without failure even under physical interaction, while cable routing in coworking conditions succeeded only with active compliance. These results demonstrate the framework's effectiveness for robust and flexible robot teaching. Future work will extend its applicability to more complex tasks and richer interfaces.

References

[1] D. Lee and C. Ott, "Incremental kinesthetic teaching of motion primitives using the motion refinement tube," *Auton. Robots*, vol. 31, no. 2, pp. 115–131, 2011.

[2] W. He, C. Xue, X. Yu, Z. Li, and C. Yang, "Admittance-based controller design for physical human–robot interaction in the constrained task space," *IEEE Trans. Automat. Sci. Eng.*, vol. 17, no. 4, pp. 1937–1949, Oct. 2020.

[3] L. Kong, W. He, C. Yang, Z. Li, and C. Sun, "Adaptive fuzzy control for coordinated multiple robots with constraint using impedance learning," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 3052–3063, Aug. 2019.

[4] C. Sun, W. He, and J. Hong, "Neural network control of a flexible robotic manipulator using the lumped spring-mass model," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 47, no. 8, pp. 1863–1874, Aug. 2017.

[5] C. Wong, C. Mineo, E. Yang, X.-T. Yan, and D. Gu, "A novel clustering-based algorithm for solving spatially constrained robotic task sequencing problems," *IEEE/ASME Trans. Mechatron.*, vol. 26, no. 5, pp. 2294–2305, Oct. 2021.

[6] S. Calinon, *Robot Programming by Demonstration*. Lausanne, Switzerland: EPFL Press, 2009.

[7] K. Yamane and J. Hodgins, "Simultaneous tracking and balancing of humanoid robots for imitating human motion capture data," in *Proc. 2009 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 2510–2517.

[8] M. Geibel, S. Mostert, A. Mertens, C. Brandl, and V. Nitsch, "Human-robot cooperation in manual assembly–interaction concepts for the future workplace," in *Proc. AHFE 2019 Int. Conf. Hum. Factors Robots Unmanned Syst.*, Washington, DC, USA, 2019, pp. 60–71.

[9] M. Bednarczyk, H. Omran, and B. Bayle, "EMG-Based variable impedance control with passivity guarantees for collaborative robotics," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 4307–4312, Apr. 2022.

[10] D. Sirintuna, T. Kastritsi, I. Ozdamar, J. M. Gandarias, and A. Ajoudani, "Enhancing human-robot collaborative transportation through obstacle-aware vibrotactile warning and virtual fixtures," *Robot. Auton. Syst.*, vol. 178, 2024, Art. no. 104725.

[11] R. Wu and A. Billard, "Learning from demonstration and interactive control of variable-impedance to cut soft tissues," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 5, pp. 2740–2751, Oct. 2022.

[12] Z. Lin et al., "ARei: Augmented-reality-assisted touchless teleoperated robot for endoluminal intervention," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 5, pp. 3144–3154, Oct. 2022.

[13] D. S. Walker, R. P. Wilson, and G. Niemeyer, "User-controlled variable impedance teleoperation," in *Proc. 2010 IEEE Int. Conf. Robot. Automat.*, 2010, pp. 5352–5357.

[14] M. Häring, J. Eichberg, and E. André, "Studies on grounding with gaze and pointing gestures in human-robot-interaction," in *Proc. Int. Conf. Social Robot.*, 2012, pp. 378–387.

[15] M. Cakmak, S. S. Srinivasa, M. K. Lee, S. Kiesler, and J. Forlizzi, "Using spatial and temporal contrast for fluent robot-human hand-overs," in *Proc. 6th Int. Conf. Hum.-Robot Interaction*, 2011, pp. 489–496.

[16] E. Ruffaldi, A. Di Fava, C. Loconsole, A. Frisoli, and C. A. Avizzano, "Vibrotactile feedback for aiding robot kinesthetic teaching of manipulation tasks," in *Proc. 26th IEEE Int. Symp. Robot Hum. Interactive Commun. (RO-MAN)*, 2017, pp. 818–823.

[17] D. Farina, R. Merletti, and R. M. Enoka, "The extraction of neural strategies from the surface EMG: An update," *J. Appl. Physiol.*, vol. 117, no. 11, pp. 1215–1230, 2014.

[18] J. Su, Y. Meng, L. Wang, and X. Yang, "Learning to assemble noncylindrical parts using trajectory learning and force tracking," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 5, pp. 3132–3143, Oct. 2022.

[19] G. D. Morais, L. C. Neves, A. A. Masiero, and M. C. F. de Castro, "Application of MYO armband system to control a robot interface," in *Proc. Int. Joint Conf. Biomed. Eng. Syst. Technol.*, 2016, pp. 227–231.

[20] M. Connan et al., "Learning to teleoperate an upper-limb assistive humanoid robot for bimanual daily-living tasks," *Biomed. Phys. Eng. Exp.*, vol. 8, 2021, Art. no. 015022.

[21] L. Scalera, S. Seriani, P. Gallina, M. Di Luca, and A. Gasparetto, "An experimental setup to test dual-joystick directional responses to vibrotactile stimuli," *IEEE Trans. Haptics*, vol. 11, no. 3, pp. 378–387, Third Quarter 2018.

[22] 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. (ICRA)*, 2015, pp. 1497–1502.

[23] A. Ajoudani, N. Tsagarakis, and A. Bicchi, "Tele-impedance: Teleoperation with impedance regulation using a body–machine interface," *Int. J. Robot. Res.*, vol. 31, no. 13, pp. 1642–1656, 2012.

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

[25] 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.*, vol. 2018, no. 1, 2018, Art. no. 8658791.

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

[27] R. Wu, H. Zhang, T. Peng, L. Fu, and J. Zhao, "Variable impedance interaction and demonstration interface design based on measurement of arm muscle co-activation for demonstration learning," *Biomed. Signal Process. Control*, vol. 51, pp. 8–18, 2019.

[28] Y. Zhang, L. Cheng, X. Xia, and H. Zhang, "Learning variable impedance skills from demonstrations with passivity guarantee," 2023, *arXiv:2306.11308*.

[29] F. Steinmetz, V. Nitsch, and F. Stulp, "Intuitive task-level programming by demonstration through semantic skill recognition," *IEEE Robot. Automat. Lett.*, vol. 4, no. 4, pp. 3742–3749, Oct. 2019.

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

[31] 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.

[32] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Berlin, Germany: Springer, 2010.

[33] F. Ferraguti, C. Secchi, and C. Fantuzzi, “A tank-based approach to impedance control with variable stiffness,” in *Proc. 2013 IEEE Int. Conf. Robot. Automat.*, 2013, pp. 4948–4953.

[34] A. d’Avella, A. Portone, L. Fernandez, and F. Lacquaniti, “Control of fast-reaching movements by muscle synergy combinations,” *J. Neurosci.*, vol. 26, no. 30, pp. 7791–7810, 2006.

[35] C. Castellini and P. van der Smagt, “Evidence of muscle synergies during human grasping,” *Biol. Cybern.*, vol. 107, no. 2, pp. 233–245, 2013.

[36] C. J. De Luca and B. Mambrito, “Voluntary control of motor units in human antagonist muscles: Coactivation and reciprocal activation,” *J. Neuriophysiol.*, vol. 58, no. 3, pp. 525–542, 1987.

[37] N. Jiang, K. B. Englehart, and P. A. Parker, “Extracting simultaneous and proportional neural control information for multiple-DoF prostheses from the surface electromyographic signal,” *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1070–1080, Apr. 2009.

[38] 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.

[39] A. Hyvärinen and P. Pajunen, “Nonlinear independent component analysis: Existence and uniqueness results,” *Neural Netw.*, vol. 12, no. 3, pp. 429–439, 1999.

[40] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” *Comput. Statist. Data Anal.*, vol. 52, no. 1, pp. 155–173, 2007.

Roberto Meattini received the M.Sc. degree in automation engineering and the Ph.D. degree in robotics and automatic control from the University of Bologna, Bologna, Italy, in 2014 and 2018, respectively.

He is currently an Assistant Professor with the University of Bologna. His research interests include manipulation, telemanipulation and advanced human-robot interaction.

Andrea Govoni received the M.Sc. degree in automation engineering in 2020 from the University of Bologna, Bologna, Italy, where he is currently working toward the Ph.D. degree in robotic manipulation.

His research interests include model-based and data-driven control of industrial robotic manipulators.

Kevin Galassi (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in automation engineering from the University of Bologna, Italy in 2018 and 2020, respectively. He is currently working toward the National Ph.D. degree in artificial intelligence with the Laboratory of Robotics, University of Bologna.

His research interests include industrial robotic manipulation of deformable objects, collaborative robotics, and mobile robotics.

Davide Chiaravalli received the M.Sc. and Ph.D. degrees in automation engineering from the University of Bologna, Bologna, Italy, in 2015 and 2018, respectively.

He is currently a Research Fellow with the University of Bologna. His research interest include robotic manipulation, haptic control, human–robot interaction, and mobile robotics.

Gianluca Palli 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 an Associate Professor with the University of Bologna. He has authored or coauthored 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.

Claudio Melchiorri (Life 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 University of Bologna. He has authored more than 350 scientific publications and is the author or editor of nine books, including two on trajectory planning and several in robotics. His research interests include dexterous manipulation, haptics, telemanipulation, and nonlinear and motion control.