Design of an Assistive Controller for Physical Human–Robot Interaction Based on Cooperative Game Theory and Human Intention Estimation

Paolo Franceschi, Davide Cassinelli, Nicola Pedrocchi, Manuel Beschi, *Member, IEEE*, and Paolo Rocco, *Senior Member, IEEE*

***Abstract*— This article aims to design an assistive controller for physical Human-Robot Interaction (pHRI) based on Dynamic Cooperative Game Theory (DCGT). In particular, a distributed Model Predictive Control (dMPC) is formulated based on the DCGT principles (GT-dMPC). For proper implementation, one crucial piece of information regards human intention, which is defined as the desired trajectory that a human wants to follow over a finite rolling prediction horizon. To predict the desired human trajectory, a learning model is composed of cascaded Long-Short Term Memory (LSTM) and Fully Connected (FC) layers (RNN+FC). Iterative training and Transfer Learning (TL) techniques are proposed to adapt the model to different users. The behavior of the proposed GT-dMPC framework is thoroughly analyzed with simulations to understand its applicability and the tuning of its parameters for a pHRI assistive controller. Moreover, real-world experiments were carried out on a UR5 robotic arm equipped with a force sensor was installed. First, a brief validation of the RNN+FC model integrated with the GT-dMPC is proposed for the iterative procedure and the TL. Finally, an application scenario is proposed for co-manipulating two objects and comparing the obtained results with other controllers typically used in the pHRI. Results show that the proposed controller reduces the required force of the human in completing tasks, even in the presence of unknown and different loads and inertia. Moreover, the proposed controller allows for precise reaching of the target point and does not introduce any undesirable oscillations. Finally, a subjective questionnaire shows that the proposed controller is, in general, preferred by different users.**

***Note to Practitioners*—This work presents a method to design an assistive controller to help a human perform physically coupled shared tasks with a robot. The target applications of this work are co-handling tasks of large or heavy objects. Such tasks require two agents to be performed easily, and the proposed work aims to make the robot a companion for the human partner. The proposed approach also quickly adapts to new users or tasks, making it feasible for real production systems or daily scenarios. Another possible target application is the co-manipulating large flexible components such as carbon fiber plies. This application would require small modifications, particularly in how the force is exchanged. Some additional/different sensors should be used, such as vision to map object deformations with virtual forces. The present work does not directly consider these kinds of applications. Indeed, this work strictly relies on force measurements that are not reliable when dealing with flexible materials, at least in a compression state. Such an issue will be investigated in future works by using vision systems to measure a virtual force that allows this method to be applicable even in the case of flexible components.**

***Index Terms*— Physical human–robot interaction, learning human intention, human intention identification, dynamic cooperative game theory, model predictive control.**

I. Introduction

IN TODAY'S industrial scenarios, robotic assistants are continuously being studied for several applications such as handling components, welding, and assembly. Human-Robot Collaboration (HRC) [1], [2], [3], is the study of collaborative processes in which human and robot agents work together to achieve shared goals. Alongside the HRC, Human-Robot Interaction (HRI) focuses on studying interactions between humans and robots. To develop collaborative applications between humans and robots, different aspects of their interaction must be addressed, including modeling human behavior [4], possibly involving machine learning techniques [5], perception of the human [6], and natural communications channels such as forces or natural languages [7].

In the industrial scenario, a valuable role of robotic assistants is to help humans by cooperating and reducing fatigue and long-term musculoskeletal disorders. The physical Human-Robot Interaction (pHRI) [8], [9] is the study of the interactions between humans and robots that happen through

Manuscript received 19 December 2023; revised 11 June 2024; accepted 14 July 2024. This article was recommended for publication by Associate Editor H. Su and Editor H. Moon upon evaluation of the reviewers' comments. This work was supported by the European Union's Horizon 2020 Research and Innovation Program under Grant 101006732 (Drapebot). *(Corresponding author: Paolo Franceschi.)*

Paolo Franceschi is with the Department of Innovative Technologies, University of Applied Science and Arts of Southern Switzerland (SUPSI), 6962 Lugano, Switzerland, and also with the Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council of Italy (STIIMA-CNR), 20133 Milan, Italy (e-mail: paolo.franceschi@supsi.ch).

Davide Cassinelli and Paolo Rocco are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy (e-mail: paolo.rocco@polimi.it).

Nicola Pedrocchi is with the Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council of Italy (STIIMA-CNR), 20133 Milan, Italy (e-mail: nicola.pedrocchi@stiima.cnr.it).

Manuel Beschi is with the Dipartimento di Ingegneria Meccanica ed Industriale, University of Brescia, 25123 Brescia, Italy (e-mail: manuel.beschi@unibs.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2024.3429643.

Digital Object Identifier 10.1109/TASE.2024.3429643

a physical communication channel, namely an exchange of forces. Leveraging the methods of pHRI, the robot can serve as an assistant for the human, helping in manipulating heavy loads [10], [11] or positioning precisely around a target pose [12], [13], always caring for the human's safety [14].

For these purposes, three prominent aspects can be identified to deserve particular attention. First, the robot controller should be intrinsically safe and compliant. Second, the interaction between the human and the robot should be modeled for use in the robot control. Finally, anticipating human intentions allows the robot to assist. Therefore, a method for predicting human intention should be introduced.

A. Motivations and Contributions

Humans and robots will work side by side in the future, and robots should adapt their behavior according to the current task and person they are working with. For this purpose, it is fundamental that the robot understands human intentions through low-level communication channels, such as force exchanges. Therefore, in this work, we address the human-robot objects co-manipulation application, providing the robot with predictive capabilities enhanced by continuous prediction of the desired human motion intention.

By extending the previous work of some of the authors [15], this work presents the implementation and inclusion of the RNN+FC predictive model into a Game-Theoretic distributed Model Predictive Control, capable of fully exploiting the long-term prediction. Moreover, the GT-dMPC control framework is tested with different co-manipulated objects, trajectories, and target poses that are known or unknown to the robot. Other works address the human-robot object transportation. For example [16], [17] use a motion capture system to understand human intentions, or [18] divides the task into subtasks with different priorities. These works do not make any prediction of future directions, which can be very helpful to anticipate human intentions, providing better assistance. Reference [19] proposes LSTM to predict human forces instead of the desired motion direction, not allowing for including the direction anticipation in the control problem. Moreover, these works do not directly provide a formulation of the interactive task as a system with various agents acting on it. With the proposed GT-dMPC framework we allow for optimal control computed on a receding horizon, hence accounting for uncertainties, and for control correction, which typically happens in pHRI. Indeed, in those cases, human models present uncertainties and require online adaptation of the control strategy. Game theory is the study of mathematical models of strategic interactions among rational agents. Therefore, Game Theory allows for the description of multiple agents interacting with an environment, and it perfectly suits the pHRI problem, where humans and robots must interact to solve a task. Compared to the previous work, the RNN+FC model is trained more generally, avoiding pre-computed trajectories. Indeed, in this case, the trajectories are computed for each new trial with the BiTRRT motion planner within the Moveit! framework. This work also addresses reaching target points for which the model is not trained, showing that it can still provide the human with proper assistance. Finally, the results evaluating the application also include a questionnaire to test subjective appreciation of the proposed controller compared to other state-of-the-art controllers. Indeed, subjects' preferences may vary according to different tasks and control modes of the robot, and must always be considered towards smooth HRI [20].

The main contributions of this work are summarized as follows:

- application of a cooperative Game-Theoretical distributed Model Predictive Control (GT-dMPC) to the pHRI scenario, evaluation of the proposed GT-dMPC performances in simulations along with experimental evaluations;
- training of the RNN+FC in the Cartesian space in the presence of unknown loads and inertia, and its integration into the GT-dMPC framework.

II. Related Works

This section addresses the three main topics discussed in this work to design the assistive controller. First, control techniques for the pHRI are presented, focusing on using the Impedance/Admittance control. Then, Game-Theoretical modeling of the Human-Machine interaction is analyzed since it represents a powerful method for describing interactions between rational players, as in the pHRI case. Finally, we present a review of the human intention prediction, which is fundamental information to implement the proposed control.

A. Physical Human-Robot Interaction

The most widespread control technique for smooth and compliant pHRI relies on Impedance/Admittance Control [21]. Indeed, the Impedance Control allows the robot's motion to be governed by the mass-spring-damper system's dynamics, subject to an external force, allowing passive and smooth motion and reaction to external forces. With the aim of better assisting the human operator, the impedance control parameters and set-point can vary during a task. One approach builds on online adjusting the impedance reference set-point [22], [23], smoothly letting the mass-spring-damper dynamics govern the motion from the current pose to the target pose. An example of the adaptation of the impedance set-point according to interaction with a human can be found in [24], where the reference trajectory is shaped to ensure it is within the constrained task space. This approach is more often used in standard industrial applications such as trajectory-following applications where contacts might be foreseen [25] or assembly processes where the target position is known with a certain tolerance and minor adjustments should be allowed [26], [27]. A different approach aims at modifying online the mass, spring, and damping parameters [28], [28], [29], [30] to change the robot behavior. Indeed, typically in pHRI applications, underdamped motions are desired for large and fast motions, while overdamped behavior is more desirable for small and precise motions [31]. Such control methods are typically used in Manual Guidance, Programming-By-Demonstration, and trajectory learning frameworks [32], [33], [34], where the human guides the robot tip by directly grasping it to teach a prescribed trajectory.

This method can also handle external forces such as contacts, as proposed in [35]. Finally, there exist control methods that combine both approaches [36], [37], [38]. In these scenarios, the robot should assist during the motion by also allowing a passive motion, even in the case of partially known tasks [39]. To have the robot work as an assistant and ensure system passivity, we keep the impedance control parameters constant rather than make them variable. In this way, the robot's motion at the end effector can be seen as a system that follows the dynamics of the mass-spring-damper system, subject to two external control inputs: the human interaction force and an additional assistance term that the robot can provide in terms of a virtual force. With this problem transformation (*i.e.,* robot and human acting with a fixed system), the objective is to define the assistance term. The human and the robot can be seen as two agents acting on the same system. By adequately defining their objectives, Game-Theoretical formulation can describe this interaction. Game theory, in fact, is the study of interactions among rational decision-makers, and it becomes Dynamic Game Theory (DGT) when it deals with dynamic systems. The next subsection will review applications of GT to pHRI.

B. GT Modeling of Human-Machines Interaction

In the literature, Game-theoretical formulations of the pHRI problem exist. GT formulation can be found in [40], and similarly, [41], and in [42] and [43], to implement Role Arbitration laws exploiting the Nash Equilibrium concept. In these works, the robot allows the human to lead the action but does not directly provide additional assistance. References [44], [45], and [46] propose a universal game-theoretical framework that addresses various game-theoretical behavior under specific control parameter tuning care. These works mainly investigate the Non-Cooperative scenario. In [47] a thorough analysis of the Cooperative and Non-Cooperative modeling of pHRI is presented, highlighting the strengths and weaknesses of the two models, as well as their actual applicability to real-world scenarios. Some authors also investigated the GT formulation in [48], by adopting the Cooperative formulation of the GT problem to pHRI. Indeed, the Cooperative formulation allows players to achieve better outcomes from the game [49]. These works rely on the Differential GT formulation, which represents an extension of the optimal control to the multiple agents' scenario. Reference [50] presents the use of both differential models (cooperative and non-cooperative) with application to the Role Arbitration by dynamically switching between the two during a task.

Being humans unpredictable and not completely reliable, to take into account online modifications, we implement a Model Predictive Control (MPC) formulation of the GT problem. Typical formulations rely on the distributed Cooperative Model Predictive Control (dMPC), presented in [51]. Theoretical GT-dMPC modeling of different human behaviors (Non-Cooperative [52], Cooperative [53], and others [54], [55], [56]) have been presented, involving Linear-Quadratic formulation of the game, both from an Optimal Control and a Model Predictive Control perspectives. Experimental evaluations of such models are presented in [57] and [58], showing that GT-based formulation better describes the driver's behavior than the driver's classic steering control strategy.

These works represent an exciting and solid background, showing the modeling of human-machine interactions based on GT-dMPC frameworks. Despite this, no implementation of the GT-dMPC framework is presented for the pHRI. Moreover, implementing the GT-dMPC requires knowledge of human intention. The previous works are limited to modeling the interaction and taking this information for granted or imposing the target path on humans. They do not directly address how human intention can be identified online without prior knowledge.

C. Human Intention Prediction

The two main research branches that aim at understanding the desired human intention of motion rely on (i) model-based and (ii) data-driven approaches.

The model-based approach builds on modeling the human arm with the standard impedance model [59], [60]. This approach requires estimating the human arm impedance parameters. The main drawback of this approach is that such parameters are subjective, time-variant, and depend on the specific task considered. For example, lifting an arm on the vertical plane has different parameters with respect to opening the same arm on the horizontal plane, and so on. Therefore, such a physical human arm model presents low flexibility and generalization. A different approach is to implement a control-oriented modeling of the human, considering it as a feedback controller [61], [62], with control gains as unknown parameters to be recovered. This approach requires a control model that might introduce modeling errors and parameter estimation techniques. Finally, in [63], the human intent is obtained by double integrating the estimated acceleration imposed on the robot admittance control.

The data-driven approach aims at training Machine Learning algorithms based on collected data (either real-world or synthetic) and builds a model that transforms inputs into outputs. Among the various, Neural Networks (NN) achieve excellent results in approximating complex non-linear systems with high uncertainties. In particular, when dealing with sequences (either logical or temporal), Recurrent Neural Networks (RNN) implements efficiently NN. Indeed, RNNs account for previous events, allowing information to persist over a specific horizon. There are various types of RNN [64], vanilla RNN, Gated recurrent unit (GRU), and Long-Short Term Memory (LSTM) are the most common types. In particular, the LSTM [65] architecture outperforms the classical RNN and is now widely adopted to solve multiple problems where the sequential pattern of the data may store important information, such as speech recognition. The same reasoning applies to human intent estimation since the previous motion state may be a manifestation of the intent.

Indeed, various works adopt RNN architectures for the prediction of human motion. Vision-based data are usually exploited [66], [67], [68], as cameras provide a clear understanding of the scene and can detect the human skeleton

without requiring any contact to feed both LSTMs and GRUs models. Reference [69] presents the AutoRegressive Integrated Moving Average (ARIMA) model for the visual identification of the elbow motion of a human. The main drawback of using vision-based information is that specific hardware is required, and image data are pretty complex to process, increasing the time required for training.

To overcome camera-related issues, other works exploit different data information. In [70], the input data to the LSTM model are read via Force MyoGraphy (FMG). Multiple subjects are used to train the model, but the approach is not general. A sensor based on electromyography (EMG) signals acquired from human arm muscles is used in [71]. In this work, the authors propose using a NN to classify the intended direction of human movement. This work limits the classification of the desired direction of motion to allow the robot to assist. Despite this, no prediction of future motion intentions regarding the desired trajectory is addressed. Also, Gaussian processes can be used, such as in [72] and [73]. The first proposes identifying human motion intention interacting with an exoskeleton via a sparse Gaussian process. The second models the human arm as 7 degrees of freedom (DoF) impedance and estimates the intention as the human force by the Gaussian Process. In [74], the motion intention prediction of the human is based on an autoregressive (AR) model for teleoperation. Reference [75] proposes a controller capable of learning human behavior and providing assistive or resistive force. The human behavior is estimated via a Recursive Least Squares method through measured force. In [76], an Adaptive Neural Network estimates the joint coordinates of the human lower limb interacting with an exoskeleton for rehabilitation. In contrast, [77] predicts human motion intention by online learning without training. In [38], the model is based on Radial Basis Function Neural Networks (RBFNN). In this work, an updating law adjusts the NN weights online to guarantee estimation accuracy even when human motion intention changes. Still, the prediction horizon is one step. Interestingly, [78] and [79] propose an LSTM to predict the reference set-point at the next step but do not address any adaptation to new users or objects. In addition, [79] proposes training and using LSTM to predict the reference one time instant but does not adapt the model to new users, so each new human has to record the complete set. Finally, some of the authors' previous work addresses a model composed of LSTM cascaded with Fully Connected (FC) standard NN layers [80]. Such a model is iteratively trained to adapt the model to understand the interaction better. Transfer Learning (TL) is proposed to adapt the same model to new users, trajectories, or co-manipulated objects, showing the capacity for generalization and long-term prediction.

In this work, we extend the previous work to the 3D Cartesian space and exploit its long-term prediction, including it in the GT-dMPC framework.

III. Game-Theoretic Distributed MPC

This section presents the human-robot system modeled as a Cartesian impedance with two separate external forces provided by the human and the robot. The problem is then reformulated as a Dynamic Cooperative Game and included in a distributed Model Predictive Control framework. Finally, the model for human intention identification is presented.

A. System Modeling

The robot motion at the end effector is modeled as a Cartesian impedance:

$$M_i\, a(t) + D_i\, v(t) + K_i \Delta x(t) = u_h(t) + u_r(t) \tag{1}$$

where M_i, D_i and $K_i \in \mathbb{R}^{6\times 6}$ are the desired inertia, damping, and stiffness matrices, respectively; $a(t)$, $v(t)$ and $\Delta x(t) \in \mathbb{R}^6$ are the Cartesian accelerations, velocities and delta positions at the end-effector, with $\Delta x(t) = x(t) - x_0(t)$ with $x_0(t)$ the equilibrium position of the virtual spring, and $u_h(t) \in \mathbb{R}^6$ and $u_r(t) \in \mathbb{R}^6$ represent the measured human and virtual robot effort applied to the system. The robot contribution u_r can be seen as an additional assistance the robot provides to the human. The Cartesian coordinates in x are defined according to [81], with the vector $x = [p^T\ \theta^T]^T$ where p^T are the position coordinates and θ^T the set of Euler angles[1] that defines the rotation matrix describing the end-effector orientation. We can write the vector containing the linear and angular velocities as $v = [\dot{p}^T\ \omega^T]^T$ with $\dot{p}$ and ω the linear and velocity.

Linearizing equation 1 around a working point, it is possible to write the system in linear state space as:

$$\begin{aligned} \dot{z} &= Az + B_h u_h + B_r u_r \\ y &= Cz \end{aligned} \tag{2}$$

where $z = [\Delta x^T\ v^T] \in \mathbb{R}^{12}$ is the state space vector, A is the state matrix $A^{12\times 12} = \begin{bmatrix} 0^{6\times 6} & I^{6\times 6} \\ -M_i^{-1}K_i & -M_i^{-1}D_i \end{bmatrix}$, and B is the input matrix $B_h^{12\times 6} = B_r^{12\times 6} = \begin{bmatrix} 0^{6\times 6} \\ M_i^{-1} \end{bmatrix}$ with $0^{6\times 6} \in \mathbb{R}^6$ denoting the zero matrix, $I^{6\times 6} \in \mathbb{R}^6$ the Identity matrix and C the output matrix of the system that converts z to y.

Finally, since the robot controllers accept commands in discrete time and data are collected in discrete time, we rewrite the system described in 2 in discrete time.

$$\begin{aligned} z(k+1) &= A_d\, z(k) + B_{d,h} u_h(k) + B_{d,r} u_r(k) \\ y(k) &= C_d\, z(k) \end{aligned} \tag{3}$$

with A_d, $B_{d,h}$ and $B_{d,r}$ indicating the discrete versions of the matrices A, B_h and B_r, and k indicating the current time instant, $z(k+1)$ the evolution of the system at the next step, and C_d the output matrix that converts $z(k)$ to $y(k)$. Notably, feeding the robot with the reference position in the joint space rather than Cartesian space is common. It is possible to obtain the reference velocity in the joint space through:

$$\dot{q}_{ref}(k) = J(q)^{+}\dot{x}(k) \tag{4}$$

where the $\dot{q}_{ref}(k) \in \mathbb{R}^n$ are the reference velocities in the joint space, n represents the number of joints, and

[1]This choice assumes that the angular rotation maintains limited values in the target applications, mainly along one rotation axis, as they work when taken far from the critical points.

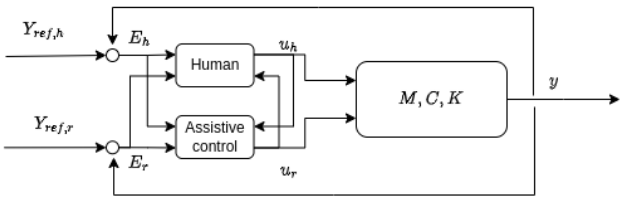


Fig. 1. The schema for the cooperative GT distributed MPC.

$J(q)^+$ is the pseudoinverse of the analytical Jacobian matrix. Joint positions are then computed via a simple integration. Assume $\dot{q} \simeq \dot{q}_{ref}(k)$ considering that today's robots have excellent tracking performance in the frequency range excitable by the operator.

B. Distributed Model Predictive Control

According to the dMPC presented in [51] the system in (3) can be augmented as

$$\begin{aligned} z_a(k+1) &= A_a z_a(k) + B_{h,a} u_h + B_{r,a} u_r \\ y_a(k) &= C_a z_a(k) \end{aligned} \tag{5}$$

with $z_a = [z^T \; z^T]^T$, $A_a = blkdiag(A_d, A_d)$, $B_{h,a} = [B_h \; B_h]^T$ and $B_{r,a} = [B_r \; B_r]^T$, and $C_a \in \mathbb{R}^{m \times 24}$ is defined based on the desired output. Conforming to the MPC formulation, defining the predicted horizon as N_p and the control horizon as N_c, the system in (5) can be written as

$$Y(k) = F z(k) + \Phi_h U_h(k) + \Phi_r U_r(k) \tag{6}$$

where $Y \in \mathbb{R}^{mN_p}$ is the predicted output and is equal to

$$Y(k) = \begin{bmatrix} y(k+1) \\ y(k+2) \\ \vdots \\ y(k+N_p) \end{bmatrix} = \begin{bmatrix} C_d z(k+1) \\ C_d z(k+2) \\ \vdots \\ C_d z(k+N_p) \end{bmatrix} \tag{7}$$

$F \in \mathbb{R}^{mN_p \times 24}$ is the free response matrix, equal to:

$$F = \begin{bmatrix} C_a A_a \\ C_a A_a^2 \\ \vdots \\ C_a A_a^{N_p} \end{bmatrix} \tag{8}$$

and $\Phi_i \in \mathbb{R}^{mNp \times 6N_c}$, with subscript $i = h, r$ denoting the human and robot, are matrices representing the forced response

$$\Phi_i = \begin{bmatrix} C_a B_{i,a} & 0 & \dots & 0 \\ C_a A_a B_{i,a} & C_a B_{i,a} & \dots & 0 \\ C_a A_a^2 B_{i,a} & C_a A_a B_{i,a} & \dots & 0 \\ \vdots & \vdots & \ddots & \\ C_a A_a^{N_p-1} B_{i,a} & C_a A_a^{N_p-2} B_{i,a} & & C_a A_a^{N_p-N_c} B_{i,a} \end{bmatrix} \tag{9}$$

The two vectors $U_h(k)$ and $U_r(k) \in \mathbb{R}^{6N_c}$ are the input vectors along the horizon that must be defined.

Consider also that the human and the robot have their reference trajectories. Define with $Y_{ref,h}(k) = [y_{ref,h}(k+1),\; y_{ref,h}(k+2),\; \dots,\; y_{ref,h}(k+N_p)]$ the trajectory a human would like to follow over the next N_p timesteps if there is no interaction with the robot. Similarly, $Y_{ref,r}(k) = [y_{ref,r}(k+1),\; y_{ref,r}(k+2),\; \dots,\; y_{ref,r}(k+N_p)]$ denotes the trajectory planned by a motion planner that the robot would like to follow over the next N_p timesteps if there is no interaction with the human.

The tracking errors can be written in compact form as

$$E_h = Y(k) - Y_{ref,h}(k) \tag{10}$$

$$E_r = Y(k) - Y_{ref,r}(k) \tag{11}$$

and finally the augmented system tracking errorE_a is defined as,

$$E_a(k) = \begin{bmatrix} y(k) - y_{ref,h}(k+1) \\ y(k) - y_{ref,r}(k+1) \\ y(k+1) - y_{ref,h}(k+2) \\ y(k+1) - y_{ref,r}(k+2) \\ \vdots \\ y(k+N_p-1) - y_{ref,h}(k+N_p) \\ y(k+N_p-1) - y_{ref,r}(k+N_p) \end{bmatrix} \tag{12}$$

According to [53] and [56], the human and the robot have the objective of minimizing a cost function that depends on the tracking error and the control effort required, defined as

$$\begin{aligned} J_i(k) &= \sum_{l=1}^{N} e_i(k+l)^T Q_{i,i} e_i(k+l) + e_j(k+l)^T Q_{i,j} e_j(k+l) \\ &\quad + u_i(k+l)^T R_i u_i(k+l) \\ &= \sum_{l=1}^{N} \begin{bmatrix} e_i(k+l)^T & e_j(k+l)^T \end{bmatrix} \begin{bmatrix} Q_{i,i} & 0 \\ 0 & Q_{i,j} \end{bmatrix} \begin{bmatrix} e_i(k+l) \\ e_j(k+l) \end{bmatrix} \\ &\quad + u_i(k+l)^T R_i u_i(k+l) \\ &= \sum_{l=1}^{N} e_a(k+l)^T Q_i e_a(k+l) + u_i(k+l)^T R_i u_i(k+l) \end{aligned} \tag{13}$$

with $i, j = \{h, r\}$ subscripts denoting the human and robot matrices. In equations (13), $Q_{i,j}$ defines the weight that the human and the robot assign to their own and the opponent's reference tracking error, $e_i(k+l) = y(k+l) - y_{ref,i}(k+l)$ refers to the tracking errors foreseen for the human and the robot at time step $k+l$, with $e_a(k+l) = \begin{bmatrix} e_h(k+l) \\ e_r(k+l) \end{bmatrix}$, and $u_h(k+l)$ and $u_r(k+l)$ are the control inputs of the human and the robot at time step $k+l$. Defining then

$$\tilde{Q}_i(k) = \begin{bmatrix} Q_i & & \\ & \ddots & \\ & & Q_i \end{bmatrix} \tag{14}$$

$$\tilde{R}_i(k) = \begin{bmatrix} R_i & & \\ & \ddots & \\ & & R_i \end{bmatrix} \tag{15}$$

equations (13) can be rewritten in compact form as

$$J_i(k) = E_a(k)^T \tilde{Q}_i E_a(k) + U_i(k)^T \tilde{R}_i U_i(k) \tag{16}$$

C. Cooperative Game-Theoretic Problem Formulation

In a cooperative game, the players agree to cooperate by sharing a common objective. The common objective is a weighted sum of the singular objectives, and the weights are the bargaining outcome. The bargaining is defined by a set of parameters $\{\alpha_i,\ i = 1 : N_{players},\ \sum_{i=1}^{N} \alpha_i = 1,\ 0 < \alpha_i < 1\}$, which weight the contribution of each player.

The shared objective is defined as

$$J_{gt}(k) = \alpha J_h(k) + (1-\alpha) J_r(k) \tag{17}$$

It is then possible to define the cooperative matrix of the weights on the system state as $Q_{gt} = \alpha \tilde{Q}_h + (1-\alpha)\tilde{Q}_r$, and the cooperative matrices of the weights on the control input of the human and the robot as $R_{gt,h} = \alpha \tilde{R}_h$ and $R_{gt,r} = (1-\alpha)\tilde{R}_r$, respectively. Therefore, from (17), the Cooperative Game-Theoretic single objectives of the two players are defined as

$$J_{i,gt}(k) = E_a(k)^T Q_{gt} E_a(k) + U_i(k)^T R_{gt,i} U_i(k) \tag{18}$$

The GT-dMPC problem for the Cooperative Game Theoretic pHRI can then be summarized as

$$\begin{aligned} &\min_{u_i} J_{i,gt} = E_a(k)^T Q_{gt} E_a(k) + U_i(k)^T R_{gt,i} U_i(k),\ i = \{h,r\} \\ &s.t.\ Y(k) = Fz(k) + \Phi_h U_h(k) + \Phi_r U_r(k). \end{aligned} \tag{19}$$

Following [51] and [56], the solution of problems (19) can be computed as

$$U^* = \begin{bmatrix} U_h^* \\ U_r^* \end{bmatrix} = \begin{bmatrix} I & K_h \\ K_r & I \end{bmatrix}^{-1} \begin{bmatrix} L_h & 0 \\ 0 & L_r \end{bmatrix} \begin{bmatrix} Z_h \\ Z_r \end{bmatrix} \tag{20}$$

in which, defining

$$S_i = (\Phi_i^T Q_{gt} \Phi_i + R_{gt,i})^{-1} \Phi_i^T Q_{gt},\ i = \{h,r\} \tag{21}$$

the gains are computed as

$$K_i = S_i \Phi_i \tag{22}$$

$$L_i = [-S_i F_i \quad S_i] \tag{23}$$

and

$$Z_h = Z_r = \begin{bmatrix} z_{gt}(k) \\ y_{ref,h}(k+1) \\ y_{ref,r}(k+1) \\ \vdots \\ y_{ref,h}(k+N) \\ y_{ref,r}(k+N) \end{bmatrix} \tag{24}$$

Finally, to implement the receding horizon logic, only the components of U_h^* and U_r^* relative to the next step are used, hence $u_h(k) = U^*(1)$ and $u_r(k) = U^*(1+N)$. Note that while the $Y_{ref,r}$ comes from a motion planner and is known, the same does not apply to $Y_{ref,h}$. Therefore, a method to identify the human intention is necessary, and it is introduced in the next section. In particular, we define with *human intention* the desired trajectory that the human wants to follow over a finite rolling prediction horizon, *i.e.,* $Y_{ref,h}$.

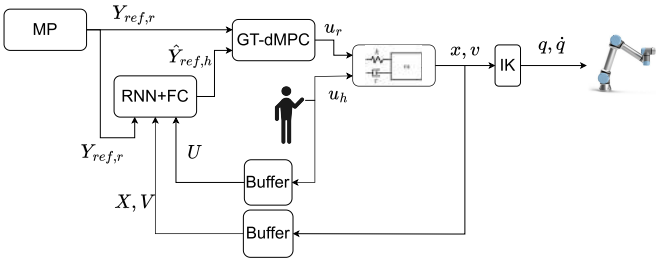


Fig. 2. The control schema with the RNN+FC model.

IV. Human Intention Prediction

This section introduces the method to predict the desired human trajectory.

As also discussed in [38], the human arm dynamics in interaction with a robot at its tip can be described as

$$-C_h v - K_h(x_{ref,h} - x) = u_h, \tag{25}$$

with C_h and K_h damping and stiffness matrices of the human arm. Assuming that $C_h = C_h(x,\dot{x})$ and $K_h = K_h(x)$, the desired human motion can be defined as

$$x_{ref,h} = F(x, \dot{x}, u_h). \tag{26}$$

The function F is nonlinear and time-variant. The problem becomes even more complicated if the human and the robot interact while transporting a large object, with additional inertia and different contact points. Therefore, this work proposes using a Recurrent Neural Network (RNN), cascaded with a Fully Connected (FC) layer (RNN+FC), to learn the complex human dynamics and provide the robot with the desired trajectory over the next horizon. In particular, among the various types, this work proposes the adoption of Long-Short Term Memory (LSTM), which has proven to have better performances for long-time series than the basic RNN.

The proposed method aims to identify and predict, over a finite rolling horizon, the desired human trajectory, given the history over a finite horizon. The RNN+FC model takes the last p time instant as input and predicts the human desired trajectory over the next N_p steps. The model accepts as inputs the actual robot positions and velocities x and v as defined in section III-A, the measured wrench the human exerts u_h, and the nominal robot trajectory $Y_{ref,r}$. At the current time instant k, the input data are $X = \{x(k-p), \ldots, x(k)\}$ and $V = \{v(k-p), \ldots, v(k)\}$ containing the positions and velocities of the past p time instant, respectively, $U_h = \{u_h(k-p), \ldots, u_h(k)\}$ containing the human interaction wrench over the past p time instant and $Y_{ref,r} = \{Y_{ref,r}(k-p), \ldots, Y_{ref,r}(k)\}$ containing the reference trajectory of the robot of the past k time instant. The output of the model is a finite sequence of reference positions in the horizon ranging from time $k+1$ to $k+N_p$, defined as $\widehat{Y}_{ref,h} = \{\widehat{y}_{ref,h}(k+1), \ldots, \widehat{y}_{ref,h}(k+N_p)\}$, where $\widehat{(\cdot)}$ denotes an estimate. A schematic view of the proposed model integrated into the GT-dMPC framework, with inputs and output highlighted, is visible in Figure 2.

A. Iterative Training

The RNN+FC model requires training its parameters to learn an approximation of the function (26). The training needs

data, and data collection is a tedious process. It is sometimes possible to generate such data with simulations [82]. Unfortunately, in the case of pHRI, it is very challenging to simulate realistic force interaction. Moreover, in the proposed control schema, predicting the desired trajectory directly influences the control inputs, as visible from (20), making it even more complex to set up a realistic simulation scenario. Therefore, data are collected on a real setup adopting the following procedure. An iterative training procedure is proposed to overcome the problem that a particular model influences the robot's behavior. Initially, because no trained model is available, the first dataset $\mathscr{D}_0$, is collected letting $\widehat{Y}_{ref,h} = Y_{ref,r}$. With $\mathscr{D}_0$ it is possible to train a model $\mathscr{M}_0$ which depends on the first dataset only $\mathscr{M}_0 = \mathscr{M}_0(\mathscr{D}_0)$. At this point, it is possible to use $\mathscr{M}_0$ to predict the future desired human trajectory. A second dataset $\mathscr{D}_1$ is collected, with the difference that $\widehat{Y}_{ref,h} = Y_{ref,r}$ does not hold anymore, and $\widehat{Y}_{ref,h} = \mathscr{M}_0(X,V,U_h,Y_{ref,r})$ is used instead. Note that the robot's behavior changes accordingly. A second model $\mathscr{M}_1 = \mathscr{M}_1(\mathscr{M}_0, \mathscr{D}_0)$ is trained. This procedure is iterated for K times until a stop criterion is reached, and the model $\mathscr{M}_K = \mathscr{M}_K(\mathscr{M}_{k-1}, \mathscr{D}_{k-1})$ is finally ready for usage. A stopping criterion can be based on the prediction error, defined as the average of the Root Mean Square Error (RMS),

$$e_{RMS} = \frac{1}{L}\sum_{T=1}^{L}\sqrt{\frac{1}{N_p}\sum_{k=T}^{T+N_p}(\|\widehat{Y}_{ref,h}(k) - X(k)\|)^2}, \tag{27}$$

where $\widehat{Y}_{ref,h}(k)$ is the predicted human intention at time instant k, $X(k)$ the measured poses, L is the length of the trajectory, and Np is the prediction horizon. The iterative procedure stops when $||e^{k+1}_{RMS} - e^{k}_{RMS}|| < tol$.

B. Transfer Learning

The procedure described in section IV-A is time-consuming, as it requires time for data collecting and model training. Moreover, despite being good at prediction, the final model obtained $\mathscr{M}_K$ is tailored and trained on a specific subject and task. Making the model as general as possible is theoretically feasible by collecting multiple data involving different subjects and tasks, creating a vast dataset, and finally training the model with it. Such an approach requires knowledge in advance of all the possible tasks and objects that can be co-manipulated and a massive experimental campaign with different people to acquire the dataset. To overcome this issue, we propose a much more efficient solution that relies upon making the model adaptive instead of general by letting the previous model's knowledge be transferred into a new model that describes similar features, the so-called Transfer Learning (TL). TL is a widely adopted method to speed up training starting from a pre-trained model. A widely used TL strategy in fields such as computer vision or NLP domains consists of "freezing" some model layers and re-train only a few layers on new data, which means having fewer parameters to be tuned compared with the complete model. Similarly, we propose to freeze the RNN part of the model that has more parameters to be tuned, and fine-tune only the final FC layers [83], [84]. In doing this, we follow the insight that the RNN learns the features of the pHRI (*e.g.,* a force in a direction means that the human wants to steer the system in the same direction, an increasing force means the human is accelerating), while the FC layer learns how a specific user does interact (*e.g.,* how much force a particular user uses to steer the system, how fast a specific user accelerate the system, etc.). With the TL approach, it is possible to quickly adapt the model $\mathscr{M}_K$ to new users and additional objects co-manipulation, with little data collection and fast training, leading to $\mathscr{M}_{TL} = \mathscr{M}_{TL}(\mathscr{M}_K, \mathscr{D}_K)$.

TABLE I
TRACKING PERFORMANCES

	$R_r = 0.001$			$R_r = 0.0005$			$R_r = 0.0001$		
α	0.2	0.5	0.9	0.2	0.5	0.9	0.2	0.5	0.9
N_5	6.15	6.15	6.14	6.15	6.15	6.14	6.14	6.14	6.09
N_{20}	5.99	6.07	5.93	5.96	6.03	5.60	5.66	5.56	2.86
N_{50}	4.45	5.16	3.77	4.25	4.73	2.48	3.28	2.71	0.61

V. GT-dMPC AND RNN+FC MODEL EVALUATION

This section presents simulations of the GT-dMPC frameworks presented in section III to understand its performances for different values of the main parameters, with the objective of understanding the set that better fits with the pHRI application. Then the training procedure of the RNN+FC model is explained and evaluated.

A. GT-dMPC Performances Analysis

The GT-dMPC framework presented in section III is simulated with different parameters tuning. For the simulation, only one dof is considered. The system is discretized at 0.008 seconds. The parameters analyzed are the prediction horizon $N_p = \{0.04,\ 0.16,\ 0.4\}$ which correspond to 5, 20, and 50 timesteps horizons, the bargaining solution parameter $\alpha = \{0.2,\ 0.5,\ 0.9\}$, and the weight of the robot's cost function for the control input $R_r = \{0.001,\ 0.0005,\ 0.0001\}$. The parameters of the human cost function cannot be made variable arbitrarily because they are descriptive of the intrinsic human behavior and must be recovered with inverse techniques such as in [85] and [86]. In this work, an average sufficiently descriptive is used, resulting in $Q_{h,h} = diag([1, 0.0001])$, $Q_{h,r} = diag([0,0])$ and $R_h = 0.0005$. The impedance parameters of (1) are set to $m_i = 10$, $c_i = 100$ and $k_i = 0$. The two references are given as $Y_{ref,h}(k) = sin(k : k+N_p)$ and $Y_{ref,r}(k) = 0.5\,sin(k : k+N_p)$.

To compare the tracking performances of the proposed controller with different tuning, we define a tracking error index as

$$e_{trac} = \int_0^t \|Y_{ref,h} - x\|dt \tag{28}$$

Note that we are interested in designing an assistive controller, and the objective is to reduce the tracking error of the desired human reference trajectory. Table I presents results for the simulated scenarios of the e_{trac} index. As visible, with high values of $\alpha(= 0,9)$, the system tends to follow the human reference closer than the robot's one. Therefore, since this work aims to define an assistive controller for the human, it is suggested to use high values of α, thus allowing for better assistance from the robot. On the contrary, according to GT

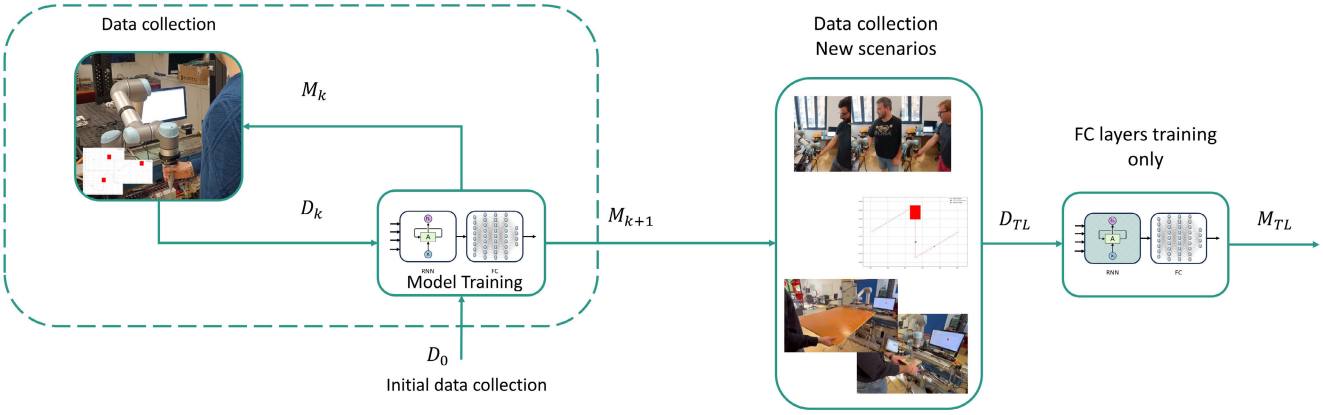
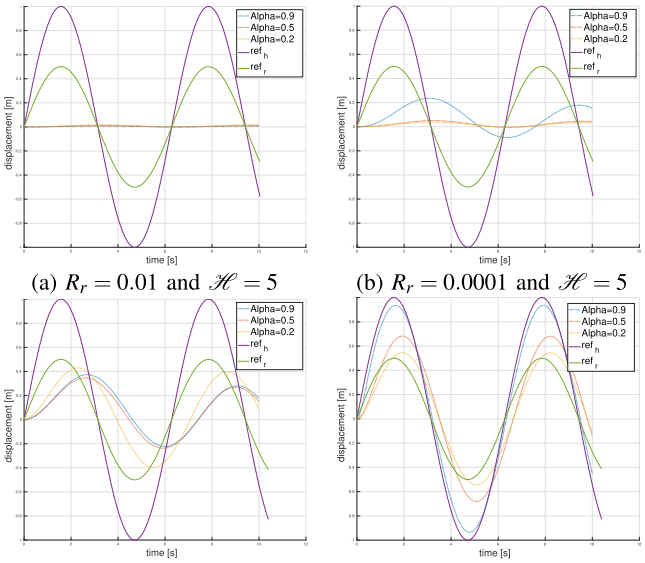


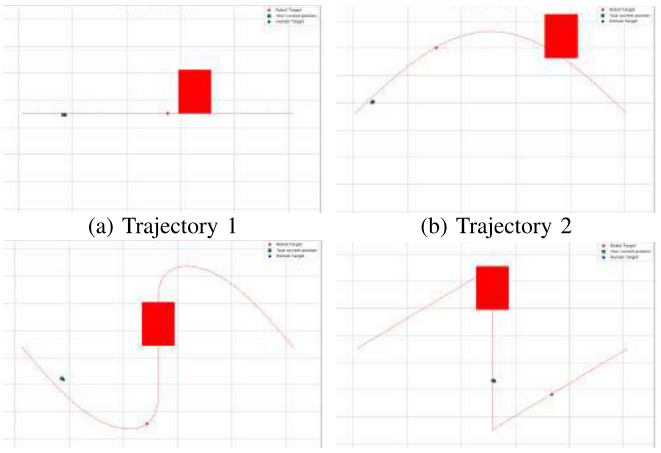
Fig. 3. The iterative training and transfer learning schema. On the left, the initial dataset D_0 is used for training an initial model. The iterative training is depicted in the dashed box. After the convergence of the main model, on the right, a new short dataset collection is done with new scenarios. Finally, the model is updated by the transfer learning procedure.



(a) $R_r = 0.01$ and $\mathscr{H} = 5$ (b) $R_r = 0.0001$ and $\mathscr{H} = 5$

(c) $R_r = 0.01$ and $\mathscr{H} = 50$ (d) $R_r = 0.0001$ and $\mathscr{H} = 50$

Fig. 4. Qualitative evaluation of tracking performances according to different tuning parameters of the GT-dMPC controller. Low values of R_r and long prediction horizons generally allow human reference tracking performances.



(a) Trajectory 1 (b) Trajectory 2

(c) Trajectory 3 (d) Trajectory 4

Fig. 5. The trajectories visible on the monitor. The red box is the obstacle, the green cross is the current position, and the red dot is the robot reference. The training uses trajectories 5a, 5b, and 5c. Trajectory 5d is for the evaluation and TL.

formulation, with $\alpha = 0,2$, it should be the human that helps the robot follow its reference. This situation is not realistic. In fact, humans do not know the robot's reference. Moreover, it is unnatural for the human to assist the robot, and it prevents the human from being assisted. So low values of α should be discarded.

Figure 5 shows some of the system tracking simulations.

B. RNN+FC Prediction Performances

The prediction performances of the RNN+FC are evaluated in this subsection. The robotic platform used for the tests is a UR5 robot, with a Robotiq FT300 mounted at the tip for measuring the human interaction force. Right after the FT sensor a lightweight 3D-printed handle is mounted allowing the human to grasp the robot at the end effector. The weight and inertial components of the handle are negligible so that it does not affect the force measurements. To evaluate the performances, we consider a collaborative motion along the x–y plane. The experiments want to simulate a situation in which a human and a robot are moving together along a trajectory, and the human, at a certain point, needs to deform the trajectory (e.g., because there is an obstacle that the robot doesn't know in between).

1) Dataset Collection: For the model's training, we collect the robot's actual poses and velocities from the robot's controller. The interaction force is measured at the robot tip via the FT sensor. An external computer provides the robot with nominal trajectories and streams real-time commands. The data are sampled at 8 milliseconds. Three nominal robot trajectories are defined: linear, curved, and sinusoidal. The three trajectories are visible in Figure 5a, 5b, and 5c, respectively. The human must follow three trajectories visible on a monitor and deform them to avoid an obstacle that appears randomly at some point in the trajectory. The robot does not know the position of the obstacle.

A single operator performed 20 repetitions for each trajectory for a total of 60 trials for each iteration. Despite the stop criterion (27) being matched after 1 iteration, we decided to perform 4 iterations to evaluate if iterating more can produce some improvement. The RNN+FC model has input data collected in the 125 time instant precedent, and the prediction

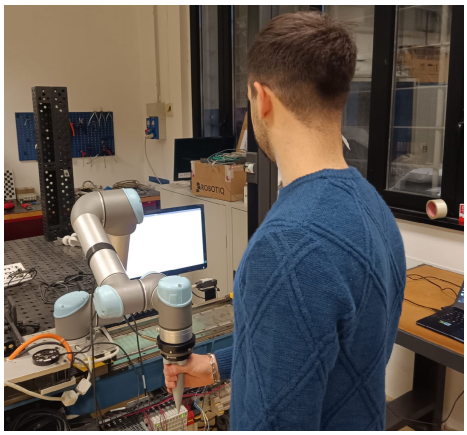


Fig. 6. Experimental setup: a Robotiq FT300 sensor is at the robot tip; a monitor shows the reference trajectory.

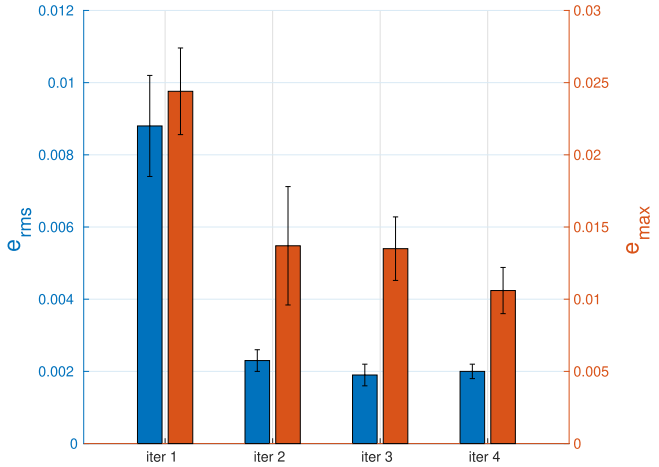


Fig. 7. e_{RMS} from (27) and e_{MAX} from (29) at the various iterations.

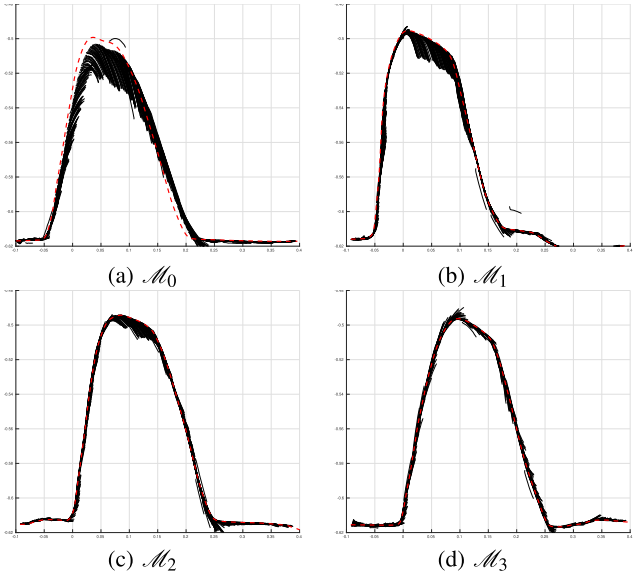


Fig. 8. Model Evaluation: the prediction at the various model iterations $\mathscr{M}_k$. The maximum prediction horizon is considered (0.4s). In solid black, the prediction at each time instant. In dashed red, the executed trajectory. X and Y axes represent the Cartesian position in meters.

horizon is set to 50 time steps. The LSTM model comprises 3 layers with 250 nodes, and the FC comprises two fully-connected layers, with 100 output nodes predicting the x–y trajectory for a horizon of 50 time instants. The learning rate, decay function, and the optimizer of the RNN+FC model are obtained with Optuna [87]. The model is trained for 25 epochs.

The impedance parameters are $M_i = diag(10,10)$, $C_i = diag(100,100)$, and the stiffness is set to null $K_i = diag(0,0)$, as typically in pHRI. The mass and damping coefficients have been hand-tuned to allow smooth motions. The cost of the two players are set as $Q_{h,h} = Q_{r,r} = diag([1,1,0.0001,0.0001])$, $Q_{h,r} = Q_{r,h} = 0^{2\times 2}$ and $R_h = R_r = diag([0.0005,0.0005])$. The human cost function parameters $Q_{h,h}$, $Q_{h,r}$ and R_h are recovered via Inverse Optimal Control (IOC) as in [85], and an average value is used. The robot parameters $Q_{r,r}$, $Q_{r,h}$, and R_r are set equal to the human's to mimic a person. Different tuning may result in more assistive behavior, which might be desirable. Finally, the parameter $\alpha = 0.8$ is chosen to allow sufficient assistance. The value of α allows high assistance and the robot to recover the position of the robot set-point autonomously. Figure 6 shows the setup.

2) Iterative Model Adaptation: First, the iterative procedure is analyzed. During this phase, the trainings are based on data collected by a single subject. The e_{RMS} in (27), and the maximum error defined as

$$e_{MAX} = \max\left(\max\left(\sqrt{\|\widehat{Y}_{ref,h}(k) - X(k)\|^2}\right)\right) \tag{29}$$

are used to evaluate the prediction performance of the RNN+FC model at each iteration. Note that the maximum error happens when the human suddenly imposes a force on the system to deviate from the nominal trajectory. In this case, in the time instants before this happens, the model has no information that can suggest that the trajectory is going to be modified. At each iteration, a model is trained based on the data collected. Figure 7 presents results of the e_{RMS} and e_{MAX} indices for four iterations for a prediction horizon of 50 timesteps. Each bar shows the average and standard deviation computed over five complete trajectories.

It is clearly visible that the two errors are dramatically reduced after the first iteration. Therefore, with just a couple of training the model is ready to be used. The average prediction error is in the range of a few millimeters, and the maximum error does not exceed 10 millimeters. Note that these results are presented for a prediction horizon of 50 timesteps. Despite this, varying the prediction horizon may also lead to different convergence rates, requiring more iterations. Moreover, we are considering only two Degrees of Freedom (DoFs) in this case, while considering more DoFs may also require more than two iterations to show no sensible improvements.

Figure 8 shows the executed and predicted trajectory at each iteration using the adapted models.

3) Transfer Learning Adaptation: To evaluate the Transfer Learning capabilities to adapt the model to new users, we consider the same e_{RMS} and e_{MAX} used for the iterations. First, we evaluate the model's capability to predict human intent with a new trajectory not used during the training. To measure this, the same subject that trained the full model performs the new set of experiments. Then, TL learns the new trajectory and compares the errors. The TL allows for reducing both errors, making it comparable to that of the known trajectories. Then, five new subjects are asked to perform the three trajectories

TABLE II

TIME REQUIRED FOR EACH STEP. THE ITERATION REQUIRES A LOT OF TIME, WHILE THE TL PROCEDURE CAN ADAPT THE MODEL QUICKLY WITH A FEW DATA

	$\mathscr{M}_3$		$\mathscr{M}_{TL}$	
	e_{RMS}	e_{MAX}	e_{RMS}	e_{MAX}
traj	2.71 ± 0.052	19.4 ± 5.43	1.68 ± 0.092	12.6 ± 2.54
subj	3.52 ± 0.585	6.67 ± 0.74	2.12 ± 1.05	4.35 ± 2.28
obj	3.69 ± 0.29	6.50 ± 0.56	1.23 ± 0.23	2.09 ± 0.22

TABLE III

TIME REQUIRED FOR EACH STEP. THE ITERATION REQUIRES A LOT OF TIME, WHILE THE TL PROCEDURE CAN ADAPT THE MODEL QUICKLY WITH A FEW DATA

	Iterations	TL subjects	TL object
dataset collection	60 ± 10 min	5 ± 2 min	5 ± 2 min
training	45 ± 5 min	4 ± 1 min	4 ± 1 min

used during the training and deform them by directly grasping the robot at the tip. Finally, the subject that trained the base model grasps a panel of 106×82cm assisted by the robot. By adding the object, different forces are exchanged. Table II shows the average of the e_{RMS} and e_{MAX} for the three different cases. All the values are given as mean and standard deviation over five repetitions of the tasks. The results regarding the subjects is the average of the results of each subject. The TL improves the predicting performances for the case of different subjects, making the average error after the TL comparable to the error of the iterated model, about 2mm. The TL decreases the errors also for the case of the co-manipulated object.

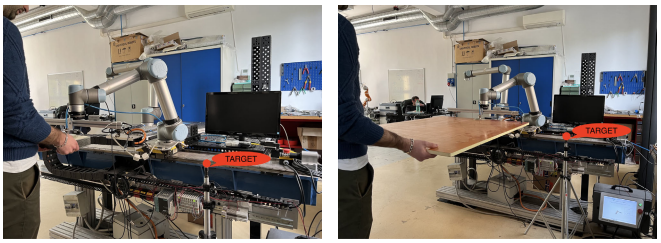
4) Time Required: Table III shows the time required for each iteration's dataset collection and model training, comparing it with the TL time. The iterative procedure takes about two hours for each iteration. The TL approach allows performing this procedure only once. After that, the model can be adapted to new situations in about 10 minutes, thanks to the TL approach. The computation runs on a laptop with Intel i7 and NVIDIA GeForce 1050.

VI. APPLICATION TO COLLABORATIVE TRANSPORTATION

The proposed approach exploiting the RNN+FC predictive model into the GT-dMPC framework is evaluated with experiments involving two different co-manipulated objects for three sets of experiments each. Five new subjects, not known to the RNN+FC model, with different levels of experience with robots, aged between 28 and 36 years old, with different heights [170-193cm] and weights [72-98kg], performed the experiments.

The experiments involve the full motion along the Cartesian positions (x–y–z). Three sets of experiments are designed, involving two different components. The first component is a granite brick weighing about 3.6 kg, which is usually used for calibration. The second component is a composite board (size 900×700 mm), an aeronautical component assembled in the cargo area of airplanes. The proposed GT-dMPC controller is compared with two other standard controllers used in pHRI, the Manual Guidance (MG), and the Impedance Control (IC), namely. The two objects and the collaborative setup are visible in figure 9.

The experiments are composed of the following steps.



(a) Mass setup (b) Panel setup

Fig. 9. The two setups used to perform Transfer Learning and comparison with MG and IMP controllers.

1) *Itertive Training:* The training is carried out in this stage without any tool or co-manipulated object. The robot mounts a lightweight 3d-printed handle to allow the human to grasp it right after the FT sensor. The trajectories are computed with Moveit!, and the human is allowed to deform the nominal robot trajectory in the x–y–z Cartesian space along the translational degrees of freedom. The RNN+FC model has 375 hidden layers in the RNN, and it outputs the prediction of the human desired positions in the x–y–z coordinates. A total of five iterations are performed to train and adapt the model to learn the actual interaction. The iterative training procedure output is the model $\mathscr{M}_4$, which is trained to collaboratively reach a precise target T_0 without any additional inertia.

2) *Transfer Learning:* After that, the model $\mathscr{M}_4$ can be adapted via TL to co-manipulate different objects with different users. Each new subject is asked to reach the target point co-manipulating an object, using the GT-dMPC controller, and the model $\mathscr{M}_4$ predicts the desired motion. The target point is the same one used for the iterative training procedure T_0. Each subject, for each of the two objects, performs 15 reaching tasks to the target T_0. After the dataset collection, TL training is quickly performed, and the comparison of the GT-dMPC controller with the IMP and MG can be performed.

3) *Reaching Tasks:* Three different reaching tasks are defined. The first requires the co-manipulation of the objects and the reaching of the same target used during the training T_0. Each subject performs such a reaching task five times with the three controllers. After that, a brief questionnaire is filled out. Then, a new target T_1 is defined. The subjects are asked to reach it in two different conditions. In the first case, the target T_1 is also known to the robot, defined in this case as $T_{1,k}$, and Moveit! computes a nominal trajectory from the home position to $T_{1,k}$. The second case still involves reaching the target T_1, but now it is unknown to the robot ($T_{1,u}$), and the trajectory computed connects the home position to the target pose T_0. After each set, the same questionnaire is filled out, and the subjects are asked to score different performances for each controller.

The following performance indexes are defined to evaluate the performances of the proposed controller compared with the MG and the IMP controllers. Since the objective of the proposed work the design an assistive controller, we consider as performance indexes the required force to move the object, the precision of positioning such an object to a target position, and the smoothness of motion to measure the naturalness of the interaction. The force Root Mean Squared error is considered

over the entire trajectory, defined as

$$f_{RMS} = \sqrt{\frac{1}{L}\sum_{k=0}^{L} \|f_k\|^2}, \tag{30}$$

with L denoting the length of the trajectory and $\|f_k\|$ the module of the measured interaction force at time instant k.

To measure the precision in reaching the target pose, we defined a tolerance δ. The distance from the target pose is defined as the Euclidean distance as $d(c,t) = \|x_c - x_t\|$, with subscripts c and t denoting the current and target pose, respectively. As a performance index, we compute the average $d(c,t)$ for two seconds after the first occurrence of $d(c,t) < \delta$. The precision performance index is defined as

$$d_{avg} = \frac{1}{n}\sum_{k=0}^{n} d(c,t)_k. \tag{31}$$

It is important to note that the average over a time horizon gives a measure of the precision of reaching a target and the stability with which such a target is reached over time.

To measure the smoothness of the interaction, the Spectral-Arc Length (SAL), as defined in [88].

Finally, we proposed a questionnaire to the subjects to evaluate subjective performance indexes which give an intuition of the naturalness of the interaction. In the proposed questionnaire the subjects are asked to score five points:

- *assistance*: measures the overall assistance felt by the users provided by the different controllers;
- *naturalness*: measures how natural the interaction was perceived by the subjects in completing the tasks;
- *smoothness*: measures how smooth the interaction was;
- *effort*: measures the required force perceived by the subjects during the interaction;
- *detection of intention*: measures how well the robot correctly intended the subjects' motion intention during motion.

A. Results

As an illustrative example, figure 10 shows the actually executed trajectory, the predicted portion of the trajectory each time instant, and the nominal robot trajectory, during the execution of the first reaching task, involving the lumped mass and the initial target pose T_0.

The results of the f_{RMS} index are presented in figure 11. In general, it is visible that the IMP controller is the one that requires the higher force to co-manipulate the objects in both cases. This is because the IMP controller has a virtual spring that always tries to restore the current pose to the nominal robot trajectory and applies a force opposite to the trajectory deformation imposed by the subject. The GT-dMPC and MG controllers show similar performances. In the case of lumped mass, the GT-dMPC slightly performs better, showing that the RNN+FC model has learned the interaction model even if a payload is applied.

The results of the d_{avg} are shown in figure 12. In this case, the IMP controller performs well when the target pose is known to the controller, as the virtual spring applies a force to move the robot towards it. Despite this, the additional

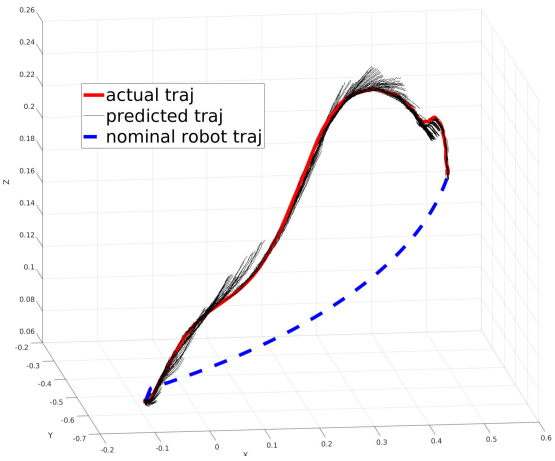
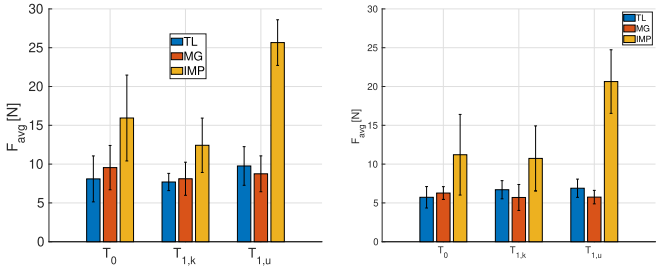
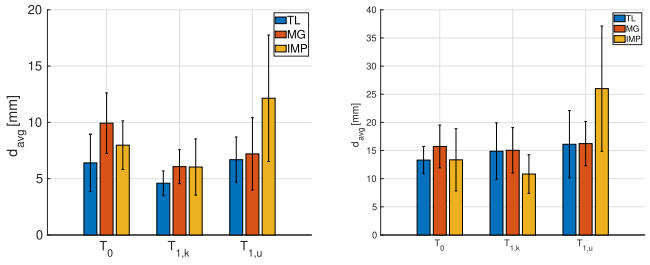


Fig. 10. Predicted, executed, and nominal trajectories for the co-manipulation of lumped mass object.



(a) average force mass (b) average force panel

Fig. 11. Force RMS, measured as in (30), for the three tasks and the three controllers.

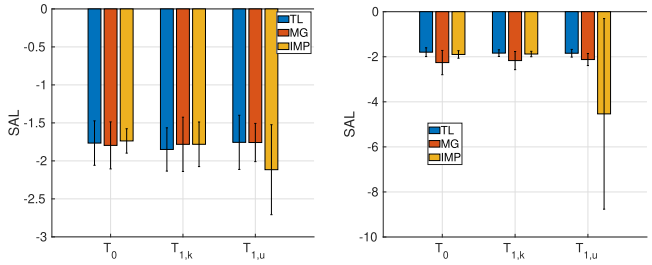


(a) average distance mass (b) average distance panel

Fig. 12. Average distances, measured as (31) for the three tasks and the three controllers.

payload is not considered by the IMP controller. Therefore, humans must still apply vertical force to reach the target pose properly. Moreover, when the human wants to reach a target different from the nominal one, the IMP cannot provide any assistance. On the contrary, the IMP controller tends to reach its known target, deriving from the human one. The MG controller cannot provide any assistance to reach the target pose, and its performances strictly depend on each subject's capabilities. The GT-dMPC controller performs better than the two others in almost all three cases regarding the mass task.

The SAL index shows that the proposed GT-dMPC controller performs comparably to the other two controllers in most cases. In general, the MG controller allows smooth motions because it has a passive behavior, and the IMP controller allows smooth motion because it tends to follow the nominal trajectory, which is typically computed to avoid too high acceleration in pHRI tasks. It is interesting to see that the GT-dMPC with the active assistive contribution shows



(a) SAL mass (b) SAL panel

Fig. 13. Average SAL for the three tasks and the three controllers.

TABLE IV
SUBJECTIVE QUESTIONNAIRE SHOWING THE PERCENTAGE OF RESPONSES COUNT

Score %		0	1	2	3	4
Assistance	GT-dMPC	0	6.67	20.0	40.0	33.3
	MG	13.3	13.3	13.3	60.0	0
	IMP	13.3	26.7	13.3	20.0	26.7
Naturalness	GT-dMPC	0	0	33.3	20	46.7
	MG	0	13.3	13.3	66.7	6.67
	IMP	13.3	26.7	53.3	6.67	0
Smoothness	GT-dMPC	0	0	6.67	66.7	26.7
	MG	6.67	13.3	26.7	53.3	0
	IMP	0	40.0	20.0	40.0	0
Effort	GT-dMPC	0	46.7	46.7	6.67	0
	MG	0	13.3	60.0	26.7	0
	IMP	6.67	33.3	0	26.7	33.3
DoI	GT-dMPC	0	0	26.7	26.7	46.7
	MG	0	6.67	33.3	46.7	13.3
	IMP	20.0	13.3	53.3	6.67	6.67

comparable performances and that the introduced assistance does not worsen the SAL index. Moreover, in the unknown reaching task, the IMP controller is incapable of any assistance, making it hard to deform the trajectory and precisely reach the target. In this case, the SAL is more negative (indicating more jerky trajectory execution), and the robot's behavior introduces oscillations. Results of the SAL computation are visible in figure 13.

Finally, the questionnaire results are presented in table IV. Note that the presented results are an aggregate of the singular scores given after each task to each controller. Therefore, the results presented are, in some cases, contradictory. This happens because, for example in the IMP case, the IMP controller is very assistive for the tasks involving the reaching of a known target but is not assistive at all if the target is unknown to the controller. Therefore the same subject can perceive high assistance in some cases and low in others. We present an aggregate of the results to give an overall performance of the controllers, and to evaluate also their flexibility.

The GT-dMPC controller is the one that shows higher assistance, with most of the scores ranging between 3 and 4. The MG controller also shows good results, with about one-third of the scores equal to 3. This can be interpreted as the humans like a slightly damped interaction as this can sometimes help in being precise and damps the load descent, letting the human perceive a little assistance in terms of "gravity compensation". The IMP controller shows various and different scores. This happens because it can be quite assistive in cases where the target pose is known. In such cases, the IMP controller helps with precise positioning by applying an attractive virtual force. For the same reason, it applies an attractive virtual force in the vertical direction to the nominal vertical target, partially relieving the human operator and giving the impression of sustaining the weight.

Naturalness evaluates how natural is perceived the interaction by the user. The GT-dMPC controller shows great performance, with the majority of the scores in 4, indicating that the proposed controller allows natural pHRI. Also the MG controller shows good performance, as its passivity guarantees a natural behavior of the robot. Despite this, it is perceived as a little less natural since it does not provide any support and the humans felt to be the only ones in charge of the task. Compared with the other two, the IMP controller does not show significant results. This is mainly because it has a reference trajectory that tends to follow, introducing forces that sometimes are not following the desired motion of the human.

Regarding smoothness, the GT-dMPC performs slightly better than the MG controller, with great results for both. In this case, the IMP controller is also evaluated with good scores, only slightly lower than the other two controllers. This can still be because the IMP controller, reacting to external forces that the additional inertia can produce, might oscillate a little, even in proximity to the target position.

The effort is measured as the lower, the better since it was asked to score how much effort each subject should put into completing each task. In this case, the GT-dMPC scores are in the lower half, showing that the proposed controller is capable of relieving the users from carrying the weight and also allows trajectory deformation according to the desired user intention without requiring excessive interaction force. The MG controller performances are in the upper half of the range. This is because it does not require any particular force to deviate from the nominal trajectory. Still, it cannot assist in reducing the perceived weight of the co-transported object. The IMP controller has different scores, either they are high or low. This happens because in some tasks when the target is known, the IMP controller assists in quickly reaching the target and lifting the weight, and applying a reactive force in the vertical direction. On the contrary, when the user wants to deviate from the nominal trajectory, as in the case of an unknown target, the user should apply a higher force to counteract the attractive virtual force to the nominal trajectory.

Finally, the perceived Detection of Intention is evaluated. The GT-dMPC is the controller that better gives the impression of detecting where the human wants to go and provides proper assistance. The MG controller also shows good results because it smoothly allows motions according to human desire. The IMP scores are slightly different, as in the case the target is known, the human can think that the robot properly understands the intention of reaching a desired point, but when the target it is not known it does not.

VII. DISCUSSION

The method proposed in this manuscript addresses an assistive controller framework for pHRI specifically intended for large/heavy rigid object co-manipulation. Despite this, with minor modifications, this framework could possibly involve the co-manipulation of flexible objects such as large sheets/composite material plies. These new scenarios require

some adaptation of the framework. First of all, the F/T sensor used in this case will no longer be reliable as flexible components can transmit forces only in traction. Therefore, techniques to estimate the exchanged "force" must be implemented, as in [89], to infer human intentions. Moreover, the model is likely to be trained again from scratch. TL might still work and adapt the model, but performances will be increased by re-training the model. This topic will be deeply addressed in future works. At the current status, the RNN+FC model is trained to predict changing human intention, such as in the case of obstacle avoidance or different target reaching. In general, the model given some historical inputs is capable of predicting a future desired trajectory. Since this prediction happens every 25 Hz, approximately, a new prediction is available every 40 ms, making sudden changes in human intention detectable with reasonable time response. This is visible from figure 8 in the time instants where the human deviates from the horizontal trajectory. The model suddenly starts to predict a bend trajectory for the next steps. The model is trained for specific tasks that can still generalize several human behaviors. The model itself knows how a human wants to stay in a stop or how the human wants to move and even deviate from a trajectory. These human behaviors, can generalize most of the situations that can happen during collaborative transportation: standing stop, moving towards a goal, and deviating from it for any reason (intermediate obstacle, different target, etc). Suppose a new task is faced (e.g., the collaborative transportation of flexible plies). In that case, if it still presents similarities to the two proposed in the experiments, it is possible that with minor adjustments of the model it can still work. Suppose a new task, such as a shared driving task, must be addressed. In that case, it is reasonable to adopt a different model, or at least the proposed model has to be trained on totally new data collected with the new system from scratch (the force will be applied as a torque on a steering wheel, and Cartesian wrenches cannot be used anymore as input for the learning model). Consider a sawing task, where the human grasps the saw from one side and the robot from the other. In this case, other performance evaluations would be very interesting to better evaluate the prediction performances. This would require experiments designed ad-hoc for an index that might substantially deviate from the purpose of the proposed work, in its current form, and will be investigated in future works. In this case, there would be sudden changes in the trajectory, and possibly, the model would require new training with a different dataset. Despite this, the proposed work does not consider such a scenario, and in large/heavy objects, co-manipulation, such behavior would probably never happen, especially because the inertia of the co-manipulated object does not easily allow for such behavior. Moreover, such kind of model can be used in social robotics to describe the desired path a human can follow while walking. This allows a mobile robot to approach or avoid humans in cases such as airports, hospitals, and hotels. Having a preview of the human direction allows for predicting where the human wants to go and gives additional input to the path planner. In this case, the same architecture can be used, trained on a different dataset and fine-tuned online for different humans, or categories of humans. Another interesting application involves medical robots, or teleoperated robots in general. In those cases, having the human intention information allows the robot to move towards the human goal, reducing possible uncertainties given by the actual human behavior. This is of utmost importance in surgical robotics, where predicting the desired human trajectory might allow for smoothing a trajectory online. Considering the dataset collection, one could ask what happens if the dataset contains data from different users. In general, having a more general model trained on different users might help a little in the generalization of human behaviors, at the cost of requiring longer dataset collection and training time. Moreover, the authors noted that the prediction improvement is given mainly by the TL step, which represents the actual personalization step. The "baseline model" provides the training with a stable base, but the TL procedure is the step that actually fits a model (either the single-user or the multi-user) to a specific user. In an Industrial scenario, this would require days since, for each training iteration, a new dataset must be collected with several subjects, possibly the ones that collected the base dataset. Still, a TL step would be necessary to personalize the prediction. The aim of this work is indeed to make the prediction personalized rather than generalized, and we preferred to show a method that requires only one long training and very quick adaptation. Other performance evaluations, despite being for sure interesting, would require experiments designed ad-hoc for an index that might substantially deviate from the purpose of the proposed work. Consider for instance a sawing task. In this case, the authors agree that there are sudden changes in the trajectory. Despite this, the proposed work does not consider such a scenario, and in large/heavy objects co-manipulation such a behavior would probably never happen, especially because the inertia of the co-manipulated object does not easily allow for such a behavior. Considering also the case of collaborative applications, the maximum allowed speed is 250mm/s. With such a velocity, rapid changes are not easy to happen.

VIII. Conclusion

This work proposes the use of a distributed Game-Theoretical Model Predictive Control (GT-dMPC) combined with a Recurrent Neural Network cascaded with Fully Connected layers (RNN+FC) capable of predicting the desired trajectory of a human to design an assistive controller for the physical Human-Robot Interaction. The set of parameters describing the GT-dMPC performances is analyzed in simulations to understand which set of parameters better suits the requirement of assistance to the human. The RNN+FC model involving iterative training and transfer learning is deeply evaluated. It has been shown that iterative training allows the model to fit better with human intention, and such a procedure can significantly reduce prediction errors. As iterative training is time-consuming in data collection and model training, transfer learning is proposed to quickly adapt the model to new users, trajectories, and co-manipulated objects. These two main components are finally combined to design an industrial-like scenario,

to perform experiments on the human-robot co-manipulation of large/massive objects. The proposed controller is compared with Manual Guidance and Impedance Control by evaluating different performance indexes and a subjective questionnaire. Results show that the proposed controller can provide better assistance in various tasks by reducing the required interaction force and allowing better precision in stable positioning around a target pose. The assistive component does not introduce any additional oscillation, allowing natural interaction with the human. The final subjective questionnaire shows that the users generally appreciate the proposed controller better than the other two.

Future works will address a model capable of learning human intention according to variable α. This will allow online Human-Robot Role Arbitration, with the possibility of changing the robot's behavior according to the task's state. Moreover, a model to allow online learning and adaptation will be investigated.

REFERENCES

[1] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, Nov. 2018.

[2] E. Prati, V. Villani, F. Grandi, M. Peruzzini, and L. Sabattini, "Use of interaction design methodologies for human–robot collaboration in industrial scenarios," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3126–3138, Oct. 2022.

[3] S. Proia, R. Carli, G. Cavone, and M. Dotoli, "Control techniques for safe, ergonomic, and efficient human-robot collaboration in the digital industry: A survey," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 1798–1819, Jul. 2022.

[4] R. Jahanmahin, S. Masoud, J. Rickli, and A. Djuric, "Human-robot interactions in manufacturing: A survey of human behavior modeling," *Robot. Comput.-Integr. Manuf.*, vol. 78, Dec. 2022, Art. no. 102404.

[5] F. Semeraro, A. Griffiths, and A. Cangelosi, "Human–robot collaboration and machine learning: A systematic review of recent research," *Robot. Comput.-Integr. Manuf.*, vol. 79, Feb. 2023, Art. no. 102432. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584522001156

[6] S. E. Navarro et al., "Proximity perception in human-centered robotics: A survey on sensing systems and applications," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1599–1620, Jun. 2022.

[7] A. Bucker, L. Figueredo, S. Haddadinl, A. Kapoor, S. Ma, and R. Bonatti, "Reshaping robot trajectories using natural language commands: A study of multi-modal data alignment using transformers," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 978–984.

[8] S. Haddadin and E. Croft, *Physical Human–Robot Interaction*. Cham, Switzerland: Springer, 2016, pp. 1835–1874.

[9] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction," *Mech. Mach. Theory*, vol. 43, no. 3, pp. 253–270, 2008.

[10] L. Roveda, S. Haghshenas, M. Caimmi, N. Pedrocchi, and L. M. Tosatti, "Assisting operators in heavy industrial tasks: On the design of an optimized cooperative impedance fuzzy-controller with embedded safety rules," *Frontiers Robot. AI*, vol. 6, p. 75, Aug. 2019.

[11] L. Roveda et al., "Human-robot cooperative interaction control for the installation of heavy and bulky components," in *Proc. IEEE Int. Conf. Syst. Man, Cybern. (SMC)*, Oct. 2018, pp. 339–344.

[12] T. Wojtara et al., "Human–robot collaboration in precise positioning of a three-dimensional object," *Automatica*, vol. 45, no. 2, pp. 333–342, Feb. 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109808004615

[13] D. Bazzi, F. Roveda, A. M. Zanchettin, and P. Rocco, "A unified approach for virtual fixtures and goal-driven variable admittance control in manual guidance applications," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 6378–6385, Oct. 2021.

[14] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 882–893, Apr. 2016.

[15] P. Franceschi, F. Bertini, F. Braghin, L. Roveda, N. Pedrocchi, and M. Beschi, "Learning human motion intention for pHRI assistive control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 7870–7877.

[16] D. Sirintuna, A. Giammarino, and A. Ajoudani, "An object deformation-agnostic framework for human–robot collaborative transportation," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 2, pp. 1986–1999, Jul. 2004.

[17] D. Sirintuna, A. Giammarino, and A. Ajoudani, "Human-robot collaborative carrying of objects with unknown deformation characteristics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 10681–10687.

[18] J. E. Solanes, L. Gracia, P. Mu noz-Benavent, J. V. Miro, M. G. Carmichael, and J. Tornero, "Human–robot collaboration for safe object transportation using force feedback," *Robot. Auton. Syst.*, vol. 107, pp. 196–208, Sep. 2018.

[19] J. E. Domínguez-Vidal, N. Rodríguez, and A. Sanfeliu, "Perception–intention–action cycle in human–robot collaborative tasks: The collaborative lightweight object transportation use-case," *Int. J. Social Robot.*, 2024, doi: 10.1007/s12369-024-01103-7.

[20] P. Franceschi, M. Maccarini, D. Piga, M. Beschi, and L. Roveda, "Human preferences' optimization in pHRI collaborative tasks," in *Proc. 20th Int. Conf. Ubiquitous Robots (UR)*, Jun. 2023, pp. 693–699.

[21] N. Hogan, "Impedance control: An approach to manipulation: Part I—Theory," *J. Dyn. Syst., Meas., Control*, vol. 107, no. 1, pp. 1–7, Mar. 1985.

[22] D. P. Losey and M. K. O'Malley, "Trajectory deformations from physical human–robot interaction," *IEEE Trans. Robot.*, vol. 34, no. 1, pp. 126–138, Feb. 2018.

[23] S. S. Ge, Y. Li, and H. He, "Neural-network-based human intention estimation for physical human-robot interaction," in *Proc. 8th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI)*, Nov. 2011, pp. 390–395.

[24] 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. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1937–1949, Oct. 2020.

[25] F. Han, S. Tam, Z. Cao, X. Zhao, B. Tao, and H. Ding, "Nonlinear impedance control with trajectory adaptation for collaborative robotic grinding," *Sci. China Technol. Sci.*, vol. 66, no. 7, pp. 1928–1936, Jun. 2023, doi: 10.1007/s11431-022-2418-4.

[26] P. Franceschi and N. Castaman, "Combining visual and force feedback for the precise robotic manipulation of bulky components," in *Multimodal Sensing and Artificial Intelligence: Technologies and Applications II*, vol. 11785, E. Stella, Ed., Bellingham, WA, USA: SPIE, 2021, Art. no. 1178510, doi: 10.1117/12.2595613.

[27] P. Franceschi, N. Castaman, S. Ghidoni, and N. Pedrocchi, "Precise robotic manipulation of bulky components," *IEEE Access*, vol. 8, pp. 222476–222485, 2020.

[28] F. Müller, J. Janetzky, U. Behrnd, J. Jäkel, and U. Thomas, "User force-dependent variable impedance control in human-robot interaction," in *Proc. IEEE 14th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2018, pp. 1328–1335.

[29] D. Bazzi, G. Priora, A. M. Zanchettin, and P. Rocco, "RRT* and goal-driven variable admittance control for obstacle avoidance in manual guidance applications," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1920–1927, Apr. 2022.

[30] M. J. Pollayil et al., "Choosing stiffness and damping for optimal impedance planning," *IEEE Trans. Robot.*, vol. 39, no. 2, pp. 1281–1300, Apr. 2023.

[31] F. Ficuciello, L. Villani, and B. Siciliano, "Variable impedance control of redundant manipulators for intuitive human–robot physical interaction," *IEEE Trans. Robot.*, vol. 31, no. 4, pp. 850–863, Aug. 2015.

[32] F. Benzi, F. Ferraguti, G. Riggio, and C. Secchi, "An energy-based control architecture for shared autonomy," *IEEE Trans. Robot.*, vol. 38, no. 6, pp. 3917–3935, Dec. 2022.

[33] 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, doi: 10.1109/TRO.2023.3258669.

[34] A.-N. Sharkawy, P. N. Koustoumpardis, and N. Aspragathos, "A recurrent neural network for variable admittance control in human–robot cooperation: Simultaneously and online adjustment of the virtual damping and inertia parameters," *Int. J. Intell. Robot. Appl.*, vol. 4, no. 4, pp. 441–464, Dec. 2020, doi: 10.1007/s41315-020-00154-z.

[35] H.-Y. Li, I. Paranawithana, L. Yang, and U.-X. Tan, "Variable admittance control with robust adaptive velocity control for dynamic physical interaction between robot, human and environment," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 2299–2306.

[36] L. Roveda et al., "Model-based reinforcement learning variable impedance control for human-robot collaboration," *J. Intell. Robotic Syst.*, vol. 100, no. 2, pp. 417–433, Nov. 2020.

[37] L. Roveda, A. Testa, A. A. Shahid, F. Braghin, and D. Piga, "Q-Learning-based model predictive variable impedance control for physical human-robot collaboration," *Artif. Intell.*, vol. 312, Nov. 2022, Art. no. 103771.

[38] Y. Li and S. S. Ge, "Human–robot collaboration based on motion intention estimation," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 3, pp. 1007–1014, Jun. 2014.

[39] D. Papageorgiou, T. Kastritsi, Z. Doulgeri, and G. A. Rovithakis, "A passive pHRI controller for assisting the user in partially known tasks," *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 802–815, Jun. 2020.

[40] Y. Li, K. P. Tee, W. L. Chan, R. Yan, Y. Chua, and D. K. Limbu, "Continuous role adaptation for human–robot shared control," *IEEE Trans. Robot.*, vol. 31, no. 3, pp. 672–681, Jun. 2015.

[41] W. Bi, X. Wu, Y. Liu, and Z. Li, "Role adaptation and force, impedance learning for physical human-robot interaction," in *Proc. IEEE 4th Int. Conf. Adv. Robot. Mechatronics (ICARM)*, Jul. 2019, pp. 111–117.

[42] Y. Li, K. P. Tee, R. Yan, W. L. Chan, Y. Wu, and D. K. Limbu, "Adaptive optimal control for coordination in physical human-robot interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 20–25.

[43] Y. Li, K. P. Tee, R. Yan, W. L. Chan, and Y. Wu, "A framework of human–robot coordination based on game theory and policy iteration," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1408–1418, Dec. 2016.

[44] Y. Li, G. Carboni, F. Gonzalez, D. Campolo, and E. Burdet, "Differential game theory for versatile physical human–robot interaction," *Nature Mach. Intell.*, vol. 1, no. 1, pp. 36–43, Jan. 2019.

[45] R. Zou, Y. Liu, J. Zhao, and H. Cai, "A framework for human-robot-human physical interaction based on N-Player game theory," *Sensors*, vol. 20, no. 17, p. 5005, Sep. 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/17/5005

[46] S. Music and S. Hirche, "Haptic shared control for human-robot collaboration: A game-theoretical approach," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 10216–10222, 2020.

[47] P. Franceschi, M. Beschi, N. Pedrocchi, and A. Valente, "Modeling and analysis of pHRI with differential game theory," in *Proc. 21st Int. Conf. Adv. Robot. (ICAR)*, Abu Dhabi, United Arab Emirates, 2023, pp. 277–284, doi: 10.1109/ICAR58858.2023.10406758.

[48] P. Franceschi, N. Pedrocchi, and M. Beschi, "Adaptive impedance controller for human-robot arbitration based on cooperative differential game theory," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, France, May 2022, pp. 7881–7887.

[49] A. Seierstad, "Pareto improvements of Nash equilibria in differential games," *Dyn. Games Appl.*, vol. 4, no. 3, pp. 363–375, Sep. 2014.

[50] P. Franceschi, N. Pedrocchi, and M. Beschi, "Human–robot role arbitration via differential game theory," *IEEE Trans. Autom. Sci. Eng.*, early access, Oct. 10, 2023, doi: 10.1109/TASE.2023.3320708.

[51] J. Rawlings, D. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. San Francisco, CA, USA: Nob Hill Publishing, 2017. [Online]. Available: https://books.google.it/books?id=MrJctAEACAAJ

[52] X. Na and D. J. Cole, "Linear quadratic game and non-cooperative predictive methods for potential application to modelling driver–AFS interactive steering control," *Vehicle Syst. Dyn.*, vol. 51, no. 2, pp. 165–198, Feb. 2013.

[53] X. Na and D. J. Cole, "Modelling of a human driver' s interaction with vehicle automated steering using cooperative game theory," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 5, pp. 1095–1107, Sep. 2019.

[54] X. Na and D. J. Cole, "Game-theoretic modeling of the steering interaction between a human driver and a vehicle collision avoidance controller," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 1, pp. 25–38, Feb. 2015.

[55] A. Ji and D. Levinson, "A review of game theory models of lane changing," *Transportmetrica A, Transp. Sci.*, vol. 16, no. 3, pp. 1628–1647, Jan. 2020.

[56] S. Ko and R. Langari, "Shared control between human driver and machine based on game theoretical model predictive control framework," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2020, pp. 649–654.

[57] X. Na and D. J. Cole, "Experimental evaluation of a game-theoretic human driver steering control model," *IEEE Trans. Cybern.*, vol. 53, no. 8, pp. 4791–4804, Aug. 2023, doi: 10.1109/TCYB.2022.3140362.

[58] X. Na and D. Cole, "Theoretical and experimental investigation of driver noncooperative-game steering control behavior," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 1, pp. 189–205, Jan. 2021.

[59] M. Sharifi, A. Zakerimanesh, J. K. Mehr, A. Torabi, V. K. Mushahwar, and M. Tavakoli, "Impedance variation and learning strategies in human–robot interaction," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6462–6475, Jul. 2022.

[60] C. Fang, G. Rigano, N. Kashiri, A. Ajoudani, J. Lee, and N. Tsagarakis, "Online joint stiffness transfer from human arm to anthropomorphic arm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 1457–1464.

[61] A. Popovici, P. Zaal, and D. M. Pool, "Dual extended Kalman filter for the identification of time-varying human manual control behavior," in *Proc. AIAA Model. Simul. Technol. Conf.*, Jun. 2017, p. 3666.

[62] P. Franceschi, N. Pedrocchi, and M. Beschi, "Identification of human control law during physical Human–Robot interaction," *Mechatronics*, vol. 92, Jun. 2023, Art. no. 102986. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957415823000429

[63] I. Ranatunga, S. Cremer, D. O. Popa, and F. L. Lewis, "Intent aware adaptive admittance control for physical human-robot interaction," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 5635–5640.

[64] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Phys. D: Nonlinear Phenomena*, vol. 404, Mar. 2020, Art. no. 132306. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167278919305974

[65] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[66] J. Zhang, H. Liu, Q. Chang, L. Wang, and R. X. Gao, "Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly," *CIRP Ann.*, vol. 69, no. 1, pp. 9–12, 2020.

[67] H.-S. Moon and J. Seo, "Prediction of human trajectory following a haptic robotic guide using recurrent neural networks," in *Proc. IEEE World Haptics Conf. (WHC)*, Jul. 2019, pp. 157–162.

[68] X. Zhao, S. Chumkamon, S. Duan, J. Rojas, and J. Pan, "Collaborative human-robot motion generation using LSTM-RNN," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, Sep. 2018, pp. 1–9.

[69] Y. Wang, Y. Sheng, J. Wang, and W. Zhang, "Optimal collision-free robot trajectory generation based on time series prediction of human motion," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 226–233, Jan. 2018.

[70] M. Anvaripour, M. Khoshnam, C. Menon, and M. Saif, "FMG- and RNN-based estimation of motor intention of upper-limb motion in human-robot collaboration," *Frontiers Robot. AI*, vol. 7, 2020, Art. no. 573096, doi: 10.3389/frobt.2020.573096.

[71] D. Sirintuna, I. Ozdamar, Y. Aydin, and C. Basdogan, "Detecting human motion intention during pHRI using artificial neural networks trained by EMG signals," in *Proc. 29th IEEE Int. Conf. Robot Hum. Interact. Commun. (RO-MAN)*, Aug. 2020, pp. 1280–1287.

[72] Y. Long, Z.-J. Du, W.-D. Wang, and W. Dong, "Human motion intent learning based motion assistance control for a wearable exoskeleton," *Robot. Comput.-Integr. Manuf.*, vol. 49, pp. 317–327, Feb. 2018.

[73] J. R. Medina, S. Endo, and S. Hirche, "Impedance-based Gaussian processes for predicting human behavior during physical interaction," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 3055–3061.

[74] J. Luo, D. Huang, Y. Li, and C. Yang, "Trajectory online adaption based on human motion prediction for teleoperation," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3184–3191, Oct. 2022.

[75] X. Xing, K. Maqsood, D. Huang, C. Yang, and Y. Li, "Iterative learning-based robotic controller with prescribed human–robot interaction force," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3395–3408, Oct. 2022.

[76] Y. Huo, X. Li, X. Zhang, and D. Sun, "Intention-driven variable impedance control for physical human-robot interaction," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2021, pp. 1220–1225.

[77] S. Cremer, S. K. Das, I. B. Wijayasinghe, D. O. Popa, and F. L. Lewis, "Model-free online neuroadaptive controller with intent estimation for physical human–robot interaction," *IEEE Trans. Robot.*, vol. 36, no. 1, pp. 240–253, Feb. 2020.

[78] H. Maithani, J. A. C. Ramon, and Y. Mezouar, "Predicting human intent for cooperative physical human-robot interaction tasks," in *Proc. IEEE 15th Int. Conf. Control Autom. (ICCA)*, Jul. 2019, pp. 1523–1528.

[79] W. Lu, Z. Hu, and J. Pan, "Human-robot collaboration using variable admittance control and human intention prediction," in *Proc. IEEE 16th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2020, pp. 1116–1121.

[80] P. Franceschi, F. Bertini, F. Braghin, L. Roveda, N. Pedrocchi, and M. Beschi, "Predicting human motion intention for pHRI assistive control," 2023, *arXiv:2307.10743*.

[81] B. Siciliano and L. Villani, *Robot Force Control*. Springer, 1999.

[82] D. Nicolis, A. M. Zanchettin, and P. Rocco, "Human intention estimation based on neural networks for enhanced collaboration with robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1326–1333.

[83] M. Iman, H. R. Arabnia, and K. Rasheed, "A review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, p. 40, Mar. 2023.

[84] H. Ravishankar et al., "Understanding the mechanisms of deep transfer learning for medical images," in *Deep Learning and Data Labeling for Medical Applications*, G. Carneiro et al., Eds., Cham, Switzerland: Springer, 2016, pp. 188–196.

[85] P. Franceschi, N. Pedrocchi, and M. Beschi, "Inverse optimal control for the identification of human objective: A preparatory study for physical human-robot interaction," in *Proc. IEEE 27th Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2022, pp. 1–6.

[86] F. Köpf, J. Inga, S. Rothfuß, M. Flad, and S. Hohmann, "Inverse reinforcement learning for identification in linear-quadratic dynamic games," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14902–14908, Jul. 2017.

[87] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2019, pp. 2623–2631.

[88] S. Balasubramanian, A. Melendez-Calderon, and E. Burdet, "A robust and sensitive metric for quantifying movement smoothness," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 8, pp. 2126–2136, Aug. 2012.

[89] G. Nicola, E. Villagrossi, and N. Pedrocchi, "Co-manipulation of soft-materials estimating deformation from depth images," *Robot. Comput.-Integr. Manuf.*, vol. 85, Feb. 2024, Art. no. 102630. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584523001059

Paolo Franceschi received the B.S. degree in aerospace engineering and the M.S. degree in mechanical engineering from the Politecnico di Milano in 2015 and 2017, respectively, and the Ph.D. degree from the University of Brescia in 2024. From 2016 to 2017, he was a Visiting Scholar with the IBM Almaden Research Center, San José, CA, USA, to complete the master's thesis on modeling and controlling a bipedal robot. In 2018, he joined STIIMA-CNR, Milan, Italy, as a Research Fellow. From 2022 to 2023, he was a Visiting Ph.D. Student with the Istituto Dalle Molle sull' Intelligenza Artificiale (IDSIA), Lugano, Switzerland. Since October 2023, he has been a Researcher with the Department of Innovative Technologies, University of Applied Science and Arts of Southern Switzerland (SUPSI), Lugano. His research interests include human–robot interaction, interaction control for robotic manipulators, and optimization of complex robotics manipulation working area and tasks. For more details please visit https://paolofrance.github.io/.

Davide Cassinelli received the bachelor's and master's degrees in automation and control engineering from the Politecnico di Milano, Milan, Italy, in 2019 and 2023, respectively. His interests are industrial robotics and industrial applications, with a focus on physical human–robot interaction applications for objects co-manipulation.

Nicola Pedrocchi received the M.S. degree in mechanical engineering from the University of Brescia, Brescia, Italy, in 2004, and the Ph.D. degree in applied mechanics and robotics in 2008. From 2008 to 2011, he was a Research Assistant with the Institute of Industrial Technologies and Automation, National Research Council of Italy (STIIMA-CNR), where he has been a Full Researcher, since 2011. Since 2015, he has been coordinating the activity of the Robot Motion Control and Robotized Processes Laboratory, STIIMA-CNR. His research interests include control techniques for industrial manipulators in advanced applications requiring the interaction of robots and environment, such as technological tasks or robots and human operators, such as workspace sharing and teach-by-demonstration. He is involved in research for accurate elastic modeling and dynamic calibration of industrial robots.

Manuel Beschi (Member, IEEE) received the bachelor's and master's degrees in industrial automation engineering from the University of Brescia, Brescia, Italy, in 2008 and 2010, respectively, and the Ph.D. degree in computer science, engineering, and control systems technologies from the Department of Mechanical and Industrial Engineering. He is currently an Assistant Professor with the University of Brescia.

Paolo Rocco (Senior Member, IEEE) is currently a Full Professor in Automatic Control and Robotics and a member of the Board of Governors with the Politecnico di Milano, Milan, Italy, where he was the Chair of the B.Sc. and M.Sc. Programs on Automation and Control Engineering. He is also a Co-Founder with Smart Robots, a spin-off company of the Politecnico di Milano. He has authored about 200 papers in the areas of robotics, motion control, and mechatronics. His research interests concern industrial robotics, with a particular focus on safe and productive human–robot interaction. He is currently with the Board of Directors of euRobotics, the association of all stakeholders in robotics in Europe, private part in the Public Private Partnership (PPP) SPARC. He has been in charge of several research projects with industrial partners and public bodies. He serves as an expert for the European Commission and the Italian Ministry of University and Research for the evaluation of proposals and ongoing projects. He has served in various positions in the editorial boards of journals and conferences, including being a Senior Editor of IEEE ROBOTICS AND AUTOMATION LETTERS. He serves as a Senior Editor for IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING and an Associate Editor for *Mechatronics* (IFAC).