Optimized Assistive Human–Robot Interaction Using Reinforcement Learning

Hamidreza Modares, Isura Ranatunga, *Student Member, IEEE*, Frank L. Lewis, *Fellow, IEEE*, and Dan O. Popa, *Member, IEEE*

***Abstract*—An intelligent human–robot interaction (HRI) system with adjustable robot behavior is presented. The proposed HRI system assists the human operator to perform a given task with minimum workload demands and optimizes the overall human–robot system performance. Motivated by human factor studies, the presented control structure consists of two control loops. First, a robot-specific neuro-adaptive controller is designed in the inner loop to make the unknown nonlinear robot behave like a prescribed robot impedance model as perceived by a human operator. In contrast to existing neural network and adaptive impedance-based control methods, no information of the task performance or the prescribed robot impedance model parameters is required in the inner loop. Then, a task-specific outer-loop controller is designed to find the optimal parameters of the prescribed robot impedance model to adjust the robot's dynamics to the operator skills and minimize the tracking error. The outer loop includes the human operator, the robot, and the task performance details. The problem of finding the optimal parameters of the prescribed robot impedance model is transformed into a linear quadratic regulator (LQR) problem which minimizes the human effort and optimizes the closed-loop behavior of the HRI system for a given task. To obviate the requirement of the knowledge of the human model, integral reinforcement learning is used to solve the given LQR problem. Simulation results on an *x*-*y* table and a robot arm, and experimental implementation results on a PR2 robot confirm the suitability of the proposed method.**

***Index Terms*—Adaptive impedance control, human–robot interaction (HRI), neuro-adaptive control, reinforcement learning (RL).**

I. Introduction

INDUSTRIAL robots have been successfully used to perform repetitive tasks with a high precision. However, there are tasks that are less structured and too complex to fully automate and thus cannot be totally performed by robots. Moreover, evaluation of the performance and fine tuning by the human are sometimes necessary, which makes it impossible to fully replace humans with robots. Therefore, human–robot interaction (HRI) systems are developed in industry to take advantages of the abilities of both humans and robots. In fact, humans and robots have complementary advantages and skills. Humans' strength lies in their abilities in reasoning, thinking, and acting when faced with unforeseen events, while robots are able to work in extremely risky environments with guaranteed performance. Other than industry, the HRI systems have also been employed in many other areas such as cooperative manipulation tasks [1], surgery [2], robot-assisted rehabilitation therapies [3], and elsewhere to assist the humans in their daily life.

Unlike ordinary industrial robotics where the environment is structured and known to them, in HRI systems, the robots interact with humans who may potentially have very different skills and capabilities. Therefore, it is desired to develop human–robot systems that are capable of adapting themselves to the level of the skill of the human operator to assist the human operator to accomplish a given task with minimum workload demands and to achieve a perfect closed-loop behavior of the human–robot system. This requires the design of an intelligent adaptive controller for the HRI system.

Adaptive robot controllers have been widely used to provide highly effective controllers in yielding guaranteed position control for industrial robot manipulators [4]–[9]. However, when the robot manipulator is in contact with an object or a human, it must be able to control not only positions, but also forces. Impedance control [10] provides an effective method for the control of both position and force simultaneously. Various impedance control methodologies have been developed in the literature to make a robot follow a desired trajectory while operating in physical contact with objects. Adaptive impedance control techniques using neural networks (NNs) have also been developed to tune the impedance model based on various considerations [11]–[16].

All these mentioned adaptive control methods and adaptive impedance control methods are based on tracking error dynamics, and/or making the error dynamics have a prescribed impedance characteristic. The objective of trajectory following with an error dynamics having prescribed impedance properties often restricts the applications of these approaches in HRI systems. For modern interactive HRI systems to be capable of performing a wide range of tasks successfully, it is required to include the effects of both robot and human dynamics. Human performance neuropsychological and human factors studies have shown that in coordinated motion with a robot, human

Manuscript received December 11, 2014; accepted March 6, 2015. Date of publication March 24, 2015; date of current version February 12, 2016. This work was supported in part by the National Science Foundation under Grant IIS-1208623, in part by the Office of naval Research under Grant N00014-13-1-0562, in part by the Air Force Office of Scientific Research European Office of Aerospace Research and Development under Grant 13-3055, and in part by Army Research Office under Grant W911NF-11-D-0001. This paper was recommended by Associate Editor T.-H. S. Li.

The authors are with the University of Texas at Arlington Research Institute, Fort Worth, TX 76118-7115 USA (e-mail: modares@uta.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2015.2412554

learning has two components [17]–[19]. The operator learns a robot-specific inverse dynamics model to compensate for the nonlinearities of the robot [20], [21], and simultaneously he learns a feedback control component that is specific to the successful performance of the task. These foundations can be incorporated in the design of the human–robot control system to include the effects of both robot and human dynamics, and their interactions in a task-specific outer control loop.

Recently, various adaptive impedance methods have been developed to improve the performance and safety of the HRI system. Adjustment of the impedance parameters based on various considerations such as the stability [22], the input torque [23], minimizing a cost function [24], and incorporating the human intention [22], [25] were considered. These methods, however, have not taken into account the skill differences of different operators. To overcome this problem, an approach, namely human adaptive mechatronics, is presented in [18] and [26]–[29]. This method takes into account the skill differences of the operators by adjusting the impedance of the robot according to the identified operator's model dynamics. Identifying the operator's model dynamics has been studied by several researchers [17], [30], [31]. The impedance parameters in human adaptive mechatronics approach are tuned based on a Lyapunov function to assure stability. But, stability is a bare minimum requirement for a controlled system, and it is desired to tune the impedance parameters to optimize the long-term performance of the system. References [32]–[34] performed the adjustment of the impedance based on estimation of the operator's impedance parameters. These techniques did not concern to optimize the overall long-term performance of the human–robot systems. Li *et al.* [35] and Wang *et al.* [36] developed an adaptive impedance method for HRI systems to find the optimal parameters of the robot impedance model.

In this paper, a novel approach is presented to develop an intelligent HRI system with adjustable robot behavior that assists the human operator to perform a given task using the minimum effort and achieves an optimal performance. In accordance with human factors studies, the proposed method has two control loops. A robot-specific inner loop is designed to make the robot with unknown dynamics behave like a simple prescribed robot impedance model as perceived by a human operator. Next, a task-specific outer loop is developed to find the optimal parameters of the prescribed robot impedance model. In the outer loop, the problem of finding the optimal parameters of the prescribed robot impedance model is formulated as a linear quadratic regulator (LQR) control problem [37] such that both tracking errors and human operator effort are minimized. Reinforcement learning (RL) [38]–[42] is used to solve the given LQR problem to obviate the requirement of the knowledge of the human model.

The contributions of this paper are as follows.

1) An inner-loop controller is designed to make the nonlinear unknown robot dynamics behave like a prescribed robot impedance model. This is more general than standard trajectory following. The proposed inner-loop controller does not require either task information or the specific prescribed robot impedance model parameters. This enables us to decouple the design of the robot-specific inner loop from the design of the task-specific outer-loop controller.
2) The problem of designing the optimal parameters of the prescribed robot impedance model is transformed into a LQR problem in a task-specific outer-loop control design. These parameters are determined by minimizing a performance function in terms of the human control effort and the tracking error.
3) A RL technique is employed to solve the task-specific LQR problem online in real time.
4) The proposed approach does not restrict the robot to a trajectory following task, because it leaves the task-specific details to the design of the outer loop which incorporates the human operator.

The rest of this paper is organized as follows. The next section presents the overall structure of the proposed control design method for the HRI systems. Both inner- and outer-loop control designs are briefly discussed. Sections III and IV discuss the inner- and outer-loop designs, respectively, in detail. Section V presents the simulation results, Section VI shows the experimental implementation results, and finally Section VII concludes this paper.

II. HRI Control Structure Overview

In this section, the importance of designing an assistive HRI system is first discussed and then, the structure of the proposed assistive HRI control system developed in this paper is overviewed.

A. Assistive HRI

Robots interacting with humans should be able to assist the human by adjusting themselves to the level of the skills of the human and compensating for possible human mistakes due to fatigue, stress, etc. This increases the performance of the overall HRI system in terms of safety, human force, and precision. In simple mechanical systems, for instance, a skillful operator is able to handle faster movements and thus achieve better performance using a small damping coefficient. Therefore, the damping coefficient should be tuned so that the response of the system is adapted to the level of the skill of the human operator.

In order to design such an intelligent HRI system, one major issue is that of how to best design a learning-based controller for the HRI system so that the assistance and safety are provided to the human using minimum knowledge about the human and robot dynamics, while compromise on performance is avoided. To design such an optimized assistive HRI system, the control structure proposed here decouples the robot-specific design from the task-specific design. The robot-specific inner-loop control makes the robot behave like a prescribed impedance model as perceived by the human. No task information is required in this step. In a task-specific outer loop, that includes the human operator and task information, the optimal prescribed impedance model is found. The overview of proposed method is detailed in the next section.

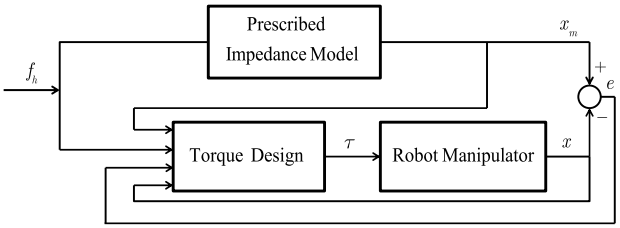


Fig. 1. Robot-specific inner-loop model following control design.

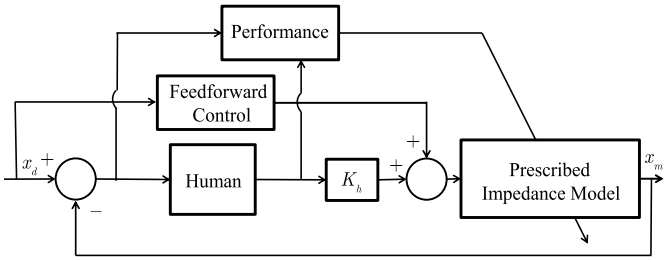


Fig. 2. Task-specific outer-loop control design.

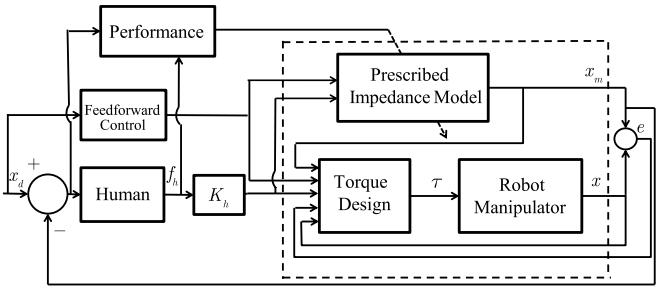


Fig. 3. Overall two-loop control design method for the adaptive HRI system.

Note that, there are many types of tasks in HRI systems. In this paper, we focus on trajectory following tasks. The desired trajectory in which the HRI system must follow is sometimes known *a priori*. For example, in robotic therapy, the patient is guided along a predetermined desired trajectory path. Point-to-point tasks with the goal of reaching a desired point in the workspace are also popular in human-in-the-loop systems. In these types of tasks, the desired point is determined *a priori*. In some applications of HRI systems, the desired trajectory is not given *a priori* and it is adjusted or planned online using various techniques such as estimation of human motion intention [43]–[45]. Although we focus on the trajectory following tasks in this paper, the proposed methodology can be extended to more general tasks. This is because no trajectory information is required in the inner loop and this leaves the freedom to incorporate task information in an outer-loop task-specific design.

B. Overall Structure of the Proposed Assistive HRI System

The HRI design here is motivated by the human factors studies [17]–[19] which states that the human learns a robot-specific inverse dynamics model to compensate for the nonlinearities of the robot, and simultaneously a feedback control component that is specific to the successful performance of the task. First, a robot torque controller is provided to avoid the need for the operator to learn a robot-specific model. Second, assistive inputs are provided to augment the operator's control effort so that the operator performs a given task with minimum workload demands and maximum performance.

To achieve these goals, the proposed method has two control loops. The first loop is a robot-specific inner loop which does not require any information of the task (see Fig. 1). The second loop is a task-specific outer loop which includes the human operator dynamics, the robot, and the task performance details (see Fig. 2).

The robot-specific inner-loop controller is shown in Fig. 1. The objective is to make the unknown robot manipulator dynamics behave like a prescribed robot impedance model as perceived by a human operator. Therefore, the human only needs to interact with the simplified impedance model. To compensate for the unknown robot nonlinearities, an adaptive NN controller is employed. This is not the same as the bulk of the work in robot impedance control and NN control, which is directed toward making a robot follow a prescribed trajectory, and/or causing the trajectory error dynamics to follow a prescribed impedance model. No trajectory information is needed for the inner-loop design. This leaves the freedom to incorporate task information in an outer-loop design. The robot-specific inner-loop controller is shown in Fig. 1 called model reference neuro-adaptive control. This is because an adaptive NN controller is developed to make the robot dynamics, from the human force to the robot motion, behave like the prescribed reference impedance model.

The task-specific outer-loop controller is shown in Fig. 2. Given the robot-specific inner-loop design, the optimal parameters of the prescribed robot impedance model are determined in this task-specific outer loop to guarantee motion tracking and also assist the human to perform the task with minimum effort. This design must take into account the unknown human dynamics as well as the desired overall performance of the human–robot system, which depends on the task.

Fig. 3 shows the overall schematic of the proposed two-loop control design method for the HRI system. The details of the inner- and outer-loop designs are given in Sections III and IV, respectively.

III. Robot-Specific Inner-Loop Control Design: Model Reference Neuro-Adaptive Controller

In this section, the design of the inner-loop controller is given in Fig. 1. The aim of the inner-loop controller is to make the robot manipulator behave like a prescribed robot impedance model. The proposed inner-loop control method has two main differences from the existing adaptive impedance control methods. First, in contrast to trajectory following based methods, the proposed method is a robot-specific method that does not require a reference motion trajectory. That is, the proposed method minimizes the model-following error between the output of the prescribed robot impedance model and the motion of the robot without needing task information. Second, in the proposed method, the designed control torque does not require any knowledge of the prescribed robot impedance model. This enables us to decouple the design of

the robot-specific inner-loop controller from the design of the task-specific outer-loop control design.

Consider the dynamical model of robot manipulator in Cartesian space [46]

$$M(q)\ddot{x} + C(q,\dot{q})\dot{x} + F_c(\dot{q}) + G(q) + \tau_d = \tau + K_h f_h \tag{1}$$

with $M = J^{-T}M^*J^{-1}$, $C = J^{-T}(C^* - M^*J^{-1}\dot{J})J^{-1}$, $M = J^{-T}M^*J^{-1}$, $F_c = J^{-T}F^*$, $G = J^{-T}G^*$, and $\tau = J^{-T}\tau^*$, where $q \in \mathbb{R}^n$ is the vector of generalized joint coordinates, n is the number of joints, $x \in \mathbb{R}^n$ is the end-effector Cartesian position, the control input force is $\tau = J^{-T}\tau^*$ with τ^* is the vector of generalized torques acting at the joints, $M^* \in R^{n\times n}$ is the symmetric positive definite mass (inertia) matrix, $C^*(q,\dot{q})\,\dot{q} \in \mathbb{R}^{n\times 1}$ is the vector of Coriolis and centripetal forces, $F_c^*(\dot{q}) \in \mathbb{R}^{n\times 1}$ is the Coulomb friction term, $G^*(q) \in \mathbb{R}^{n\times 1}$ is the vector of gravitational torques, τ_d is a general nonlinear disturbance, f_h is the human control effort, K_h is a gain, and J is the Jacobian matrix.

It is assumed that the robot manipulator dynamics in (1) are unknown.

Remark 1: Note that in the dynamics (1), it is assumed that the human force is sensed by a force sensor and is amplified by a gain K_h before applying to the robot. For example, for the x-y table example in [18], the force generated by the hand to the grip is measured by a force sensor and is magnified before it is applied to the stage. As shown in Section V, this gain is employed to help the human to minimize the tracking error and maximize the performance. It is shown in Remark 5 that if the amplification of the human force is not possible for a specific application, the proposed method can still be used.

Consider the prescribed robot impedance model

$$\bar{M}\,\ddot{x}_m + \bar{B}\,\dot{x}_m + \bar{K}\,x_m = K_h f_h + \bar{l}(x_d) \equiv l(f_h, x_d) \tag{2}$$

in Cartesian space, where x_m is the output of the prescribed robot impedance model, $\bar{M}$, $\bar{B}$, and $\bar{K}$ are the desired inertia, damping, and stiffness parameter matrices, respectively. These parameters are specified in the task-specific outer control loop design in Section IV. The auxiliary input $\bar{l}(x_d)$ is a trajectory dependent input and is also designed in Section IV.

Design Objective: The aim is to design the force τ in (1) to make the unknown robot dynamics (1) from the human force f_h to the Cartesian coordinates x behave like the prescribed robot impedance model (2). That is, it is desired to make the following model-following error go to zero:

$$e = x_m - x. \tag{3}$$

Note that, this is not a trajectory-following error. Therefore, this is a model-following design, and not a trajectory-following design, in contrast to most work on robot torque control [11]–[16]. No task information is required in this section. All task-specific details are taken into account in the next section.

It is now required to design a control torque τ to make the robot behave like the prescribed robot impedance model (2). Consider the control torque

$$\tau = \hat{W}^T\phi\left(\hat{V}^T z\right) + K_v r - v(t) - K_h f \tag{4}$$

where $v(t)$ is a robustifying signal to be specified, K_v is the control gain, and

$$r = \dot{e} + \Lambda_1 e + \Lambda_2 \varepsilon \tag{5}$$

is the sliding mode error with

$$\varepsilon = \int_0^t e(\tau)\,d\tau. \tag{6}$$

Finally

$$\hat{h}(z) = \hat{W}^T\phi\left(\hat{V}^T z\right) \tag{7}$$

is a NN with $z = [q, \dot{q}, \dot{x}_m, \ddot{x}_m, e, \dot{e}, \varepsilon]^T$ the input to the NN, $\hat{W}$ and $\hat{V}$ the NN weights, and $\phi(z)$ the vector of activation functions. As is shown in the proof of Theorem 1, the NN controller in (4) is used to compensate for the unknown robot function h defined as

$$h(z) = M(q)(\ddot{x}_m + \Lambda_1\dot{e} + \Lambda_2 e) + C(q,\dot{q})(\dot{x}_m + \Lambda_1 e + \Lambda_2\varepsilon) + F_c(\dot{q}) + G(q). \tag{8}$$

The NN universal approximation property specifies that any unknown continuous function can be approximated on a compact set using a two-layer NN to any arbitrary precision [46]. That is, for the continuous function $h(z)$ on a compact set $z \in \Omega$, one has

$$h(z) = W^T\phi\left(V^T z\right) + \varepsilon(z) \tag{9}$$

where V is a matrix of first-layer weights, W is a matrix of second-layer weights, and ε is the NN functional approximation error. The ideal weight vectors W and V are unknown and is approximated online. Therefore, $h(z)$ is approximated as (7) with $\hat{W}$ and $\hat{V}$ the estimations of W and V, respectively. Define

$$Z = \begin{bmatrix} W & 0 \\ 0 & V \end{bmatrix} \tag{10}$$

and $\hat{Z}$ equivalently.

Assumption 1: The ideal NN weights are bounded by a constant scalar so that

$$\|Z\| \le Z_B. \tag{11}$$

The following theorem shows that the proposed control input τ given by (4) guarantees the boundedness of the model-following error e and the NN weights.

Theorem 1: Consider the robot manipulator dynamics (1) and the prescribed robot impedance model (2). Let the control input be chosen as (4). Let Assumption 1 hold. Let the update rule for the NN weights be given by

$$\dot{\hat{W}} = F\,\hat{\phi}\,r^T - F\,\hat{\phi}'\,\hat{V}^T z\,r^T - k\,F\,\|\,r\,\|\,\hat{W} \tag{12}$$

$$\dot{\hat{V}} = G\,z\left(\hat{\phi}'\,\hat{W}r\right)^T r^T - k\,G\,\|\,r\,\|\,\hat{V} \tag{13}$$

where $\hat{\phi} = \phi(\hat{V}^T z)$, $\hat{\phi}' = d\phi(y)/dy|_{y=\hat{V}^T z}$, $F = F^T > 0$, $G = G^T > 0$, and $k > 0$ is a small design parameter. Let the robustifying term be

$$v(t) = -K_z\left(\left\|\hat{Z}\right\| + Z_B\right) \tag{14}$$

where $K_z > 0$. Then, $e(t)$ in (3) and the NN estimated weights are uniformly ultimately bounded.

Proof: By differentiating (3) with respect to time one has $\dot{e} = \dot{x}_m - \dot{x}$, or equivalently $\dot{x} = \dot{x}_m - \dot{e}$. Differentiating $\dot{x}$ gives $\ddot{x} = \ddot{x}_m - \ddot{e}$. Considering the sliding mode tracking error r defined in (5), one has $\dot{e} = r - \Lambda_1 e - \Lambda_2 \varepsilon$. Differentiating $\dot{e}$ gives $\ddot{e} = \dot{r} - \Lambda_1 \dot{e} - \Lambda_2 e$. Using these expressions in (1) yields

$$M(q)(\ddot{x}_m - (\dot{r} - \Lambda_1 \dot{e} - \Lambda_2 e)) + C(q,\dot{q})(\dot{x}_m - (r - \Lambda_1 e - \Lambda_2 \varepsilon)) + F_c(\dot{q}) + G(q) + \tau_d = \tau + K_h f_h. \tag{15}$$

This gives the sliding mode error dynamics

$$M(q)\dot{r} = -C(q,\dot{q})r + h(q,\dot{q},\dot{x}_m,\ddot{x}_m,e,\dot{e},\varepsilon) + \tau_d - \tau - K_h f_h \tag{16}$$

with h defined in (8). The robot manipulator dynamics (1) is assumed to be unknown and therefore h in (16) is unknown and approximated online by (7). Then, the closed-loop filtered error dynamics (16) becomes

$$M(q)\dot{r} = -C(q,\dot{q})r + \hat{W}^T\phi\left(\hat{V}^T z\right) + \tau_d - \tau - K_h f_h + \tilde{h} \tag{17}$$

where $\tilde{h} = h - \hat{h}$ is the estimation error. Substituting τ from (4) in (17) gives

$$M(q)\dot{r} = -C(q,\dot{q})r - K_v r + \tau_d + \tilde{h} + v(t). \tag{18}$$

The remainder of the proof is the same as [46] and thus is only outlined here. A Lyapunov function is defined as

$$L = \frac{1}{2}r^T M(q) r + tr\left(\tilde{W}^T F^{-1}\tilde{W}\right) + tr\left(\tilde{V}^T F^{-1}\tilde{V}\right) \tag{19}$$

where the weight estimation errors are $\tilde{W} = W - \hat{W}$ and $\tilde{V} = V - \hat{V}$, and it is shown using (12)–(14) and (17) that the Lyapunov function derivative is negative outside a compact set. This guarantees the boundedness of the filtered tracking error r as well as the NN weights. Specific bounds on r and the NN weights are given in [46]. ■

Note that the proposed controller (4) is composed of four parts. The first part is a nonlinear compensator consisting of a NN controller to compensate for unknown function h defined in (8). The second part of the controller is a stabilizing Proportional Integral Derivative controller that stabilizes the model-following error e. The third part is a robust term that is designed to achieve robustness against uncertainties. Finally, the last part is used to compensate for the human input $K_h f_h$.

Fig. 4 shows the detailed schematic of the proposed inner-loop controller. We call this model reference neuro-adaptive control because the NN adaptive controller causes the robot dynamics to behave like the prescribed impedance model (2). This is in contrast to NN torque control work, which seek to make the robot motion $x(t)$ follow a prescribed trajectory.

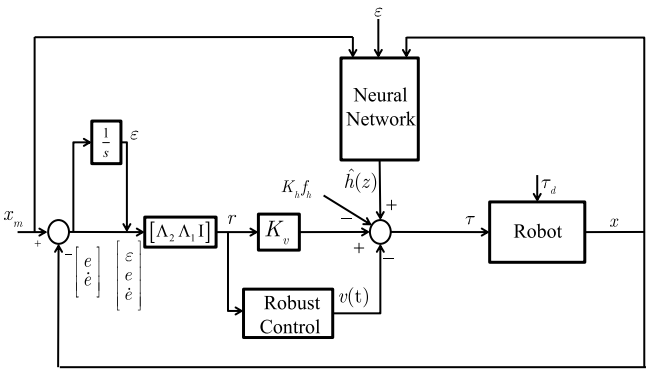


Fig. 4. Model reference neuro-adaptive inner-loop controller.

Remark 2: Function $h(z)$ in (8) does not contain the parameters $\bar{M}$, $\bar{D}$, and $\bar{K}$ of the prescribed robot impedance model (2). This means that the NN does not need to estimate the impedance model. This is in contrast to methods that design the torque τ in (1) to guarantee following a desired trajectory, and demand that the trajectory tracking error dynamics follow a prescribed impedance model [11]–[16]. Our design of the robot-specific inner-loop controller is independent from any task objectives. All task-specific information is considered in the outer-loop design in Section IV.

IV. Task-Specific Outer-Loop Adaptive Impedance Control: Finding Optimal Parameters of the Prescribed Robot Impedance Model

In this section, the design of the outer task loop controller is shown in Fig. 2. It was shown in Section III that the robot-specific controller of Theorem 1 makes the nonlinear unknown robot (1) behave like the simple prescribed robot impedance model (2) as perceived by the human operator. In this section, the parameters of the prescribed robot impedance model given in (2) are optimized to assist the human to perform a given task with minimum effort and to minimize a tracking error. To this end, the problem of optimizing the parameters of the prescribed robot impedance model is transformed into a LQR problem and then RL is used to solve the given problem without requiring the human dynamics model.

Design Objective: The aim of the task-specific outer-loop controller is to find the optimal values of the prescribed impedance parameters $\bar{B}$, $\bar{K}$, the human gain K_h (or $\bar{M}$ if $K_h = 1$), and the auxiliary input $\bar{l}(x_d)$ in (2) to minimize the human control effort f_h and optimize the tracking performance depending on the task.

The dynamics of the human model and the interaction of the robot and human are considered in this outer-loop control design. It was shown in [26] that the human dynamics change during the task learning process. After learning, an expert human operator is characterized by a simple linear transfer characteristic. Therefore, the human impedance model

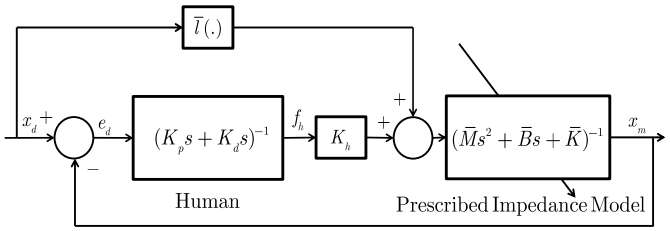


Fig. 5. Human–robot interface in the task-specific outer loop.

is assumed to be

$$(K_d s + K_p) f_h = k_e\, e_d \tag{20}$$

where K_d, K_p, and k_e are unknown gains. These parameters vary from one individual to another and depend on the specific task.

A. Task-Specific Outer-Loop Control Method: LQR Approach

The block diagram of the outer-loop task controller is sketched in Fig. 2 and shown in detail in Fig. 5. As shown in Fig. 5, in addition to the adaptive impedance loop that specifies the optimal impedance parameters, an assistive feedforward input and a human force gain are employed to help the human to minimize the tracking error. The feedforward term $\bar{l}(x_d)$ in (2) is designed to make the steady-state tracking error go to zero. The human gain K_h and the optimal values of the prescribed impedance parameters $\bar{K}$ and $\bar{B}$ in (2) are determined to minimize the human effort and the tracking error for a given task.

In the following, it is shown how the problem of finding optimal values of $\bar{B}$, $\bar{K}$, and K_h is transformed into a LQR problem, and how these parameters are obtained by solving an algebraic Riccati equation (ARE).

Define the tracking error

$$e_d = x_d - x_m \in \mathbb{R}^n \tag{21}$$

and

$$\bar{e}_d = \left[e_d^T \;\; \dot{e}_d^T\right]^T = \bar{x}_d - \bar{x} \in \mathbb{R}^{2n} \tag{22}$$

with

$$\bar{x} = \left[x_m^T \;\; \dot{x}_m^T\right]^T \in \mathbb{R}^{2n} \tag{23}$$

and

$$\bar{x}_d = \left[x_d^T \;\; \dot{x}_d^T\right]^T \in \mathbb{R}^{2n}. \tag{24}$$

Based on this tracking error, define the performance index

$$J = \int_t^\infty \left(\bar{e}_d^T Q_d\, \bar{e}_d + f_h^T Q_h f_h + u_e^T R\, u_e\right) d\tau \tag{25}$$

where $Q_d = Q_d^T > 0$, $Q_h = Q_h^T > 0$, $R = R^T > 0$, and u_e is the feedback control input which depends linearly on the tracking error $\bar{e}_d$ and the human effort f_h. Then

$$u_e = K_1\, \bar{e}_d + K_2 f_h. \tag{26}$$

It is shown in Theorem 2 that the control input (26) has two components. The first component, i.e., K_1 tunes the prescribed impedance parameters $\bar{B}$ and $\bar{K}$ and the second component, i.e., K_2 tunes the human control gain K_h (or $\bar{M}$ if $K_h = 1$).

Remark 3: Note that by minimizing the performance index (25), both tracking error $\bar{e}_d$ and human effort f_h are minimized.

By defining the augmented state

$$X = \begin{bmatrix} \bar{e}_d \\ f_h \end{bmatrix} \in \mathbb{R}^{3n} \tag{27}$$

the performance index (25) can be written as

$$J = \int_t^\infty \left(X^T Q X + u_e^T R\, u_e\right) d\tau \tag{28}$$

where $Q = \mathrm{diag}(Q_d, Q_h)$ and $u_e = K X$ with $K = [\,K_1 \quad K_2\,]$.

The dynamics of the system with augmented state (27) are now given. Using (2), one has

$$\dot{\bar{x}} = \begin{bmatrix} 0 & I_{n\times n} \\ 0 & 0 \end{bmatrix} \bar{x} + \begin{bmatrix} 0 \\ I_{n\times n} \end{bmatrix} u \equiv A_q \bar{x} + B_q u \tag{29}$$

where $\bar{x}$ is defined in (23), and

$$u = \bar{M}^{-1}\left(-K_q \bar{x} + K_h f_h\right) + \bar{M}^{-1}\, \bar{l}(x_d) \tag{30}$$

with

$$K_q = \left[\bar{K} \quad \bar{B}\right] \tag{31}$$

where $K_q \in \mathbb{R}^{n\times 2n}$, $\bar{B}$, $\bar{K}$, and $\bar{M}$ are the prescribed impedance model in (2). On the other hand, based on the human model (20), we have

$$\left(K_d s + K_p\right) f = k_e\, e_d \tag{32}$$

which can be written in time domain as

$$K_d \dot{f}_h + K_p f_h = k_e\, e_d \tag{33}$$

or equivalently

$$\dot{f}_h = -K_d^{-1} K_p f_h + k_e\, K_{d,0} \bar{e}_d \equiv A_h f_h + E_h\, \bar{e}_d \tag{34}$$

where $K_{d,0} = [K_d^{-1}\; 0] \in \mathbb{R}^{n\times 2n}$ and $\bar{e}_d$ is defined in (22).

The following theorem shows how the problem of finding the optimal parameters of the prescribed impedance model and the human gain are obtained by solving a LQR problem.

Remark 4: Note that in [18], the human transfer function from e_d to f_h was considered as

$$G(s) = \frac{K_d s + K_p}{Ts + 1}. \tag{35}$$

For this case, A_h and B_h in (34) become $A_h = -T^{-1}$ and $E_h = T^{-1}[K_p \;\; K_d]$.

Theorem 2: Consider the prescribed robot impedance model (2). Based on dynamics in (29) and (34), define augmented matrices A and B by

$$A = \begin{bmatrix} A_q & 0 \\ E_h & A_h \end{bmatrix}, B = \begin{bmatrix} B_q \\ 0 \end{bmatrix}. \tag{36}$$

Define

$$K = \left[K_q \;\; K_h\right] \in \mathbb{R}^{n\times 3n} \tag{37}$$

as the matrix of the impedance parameters and the human gain. Then, the optimal value of K which minimizes the performance index (25) is given by

$$K = -\bar{M}\,R^{-1}B^T P \tag{38}$$

where P is the solution to the ARE

$$0 = A^T P + PA + P\,BR^{-1}B^T P + Q. \tag{39}$$

Then, the optimal feedback control is given by

$$u_e = \bar{M}^{-1}\bar{K}\,e_d + \bar{M}^{-1}\bar{B}\,\dot{e}_d + \bar{M}^{-1}K_h f_h. \tag{40}$$

Proof: Manipulating (30) gives

$$\begin{aligned} u &= \bar{M}^{-1}\left(K_q \bar{e}_d + K_h f_h\right) + M^{-1}\left(\bar{l}(x_d) - K_q \bar{x}_d\right) \\ &\equiv u_e + u_d \end{aligned} \tag{41}$$

where $\bar{e}_d$ and $\bar{x}_d$ are defined in (22) and (24), and

$$u_e = \bar{M}^{-1}\left(K_q \bar{e}_d + K_h f_h\right) \tag{42}$$

is a feedback control input, and

$$u_d = M^{-1}\left(\bar{l}(x_d) - K_q \bar{x}_d\right) \tag{43}$$

is a feedforward control input. The steady state or the feedforward term is used to guarantee perfect tracking. That is, in the steady state one has

$$\dot{\bar{x}}_d = A_q \bar{x}_d + B_q u_d \tag{44}$$

where $\bar{x}_d$ is defined in (24). Therefore

$$\bar{l}(x_d) = \bar{M}\,u_d + K_q \bar{x}_d = \bar{M}\,B_q^{-1}\left(\dot{\bar{x}}_d - A_q \bar{x}_d\right) + K_q \bar{x}_d. \tag{45}$$

Taking derivative of $\bar{e}_d$ and using (29) and (44), and some manipulations gives

$$\dot{\bar{e}}_d = A_q \bar{e}_d + B_q u_e. \tag{46}$$

Using the augmented state (27), and using (34) and (46) one has

$$\begin{aligned} \dot{X} &= \begin{bmatrix} \dot{\bar{e}}_d \\ \dot{f}_h \end{bmatrix} = \begin{bmatrix} A_q & 0 \\ E_h & A_h \end{bmatrix}\begin{bmatrix} \bar{e}_d \\ f_h \end{bmatrix} + \begin{bmatrix} B_q \\ 0 \end{bmatrix} u_e \\ &\equiv AX + Bu_e. \end{aligned} \tag{47}$$

The control input u_e in terms of the augmented state can be written as

$$u_e = \bar{M}^{-1}\left(K_q \bar{e}_d + K_h f_h\right) = \bar{M}^{-1} K\,X. \tag{48}$$

Finding the optimal feedback control (48) to minimize the performance index (25) subject to the augmented system (47) is a LQR problem and its solution is given by [37]

$$u_e^* = -R^{-1}B^T P X \tag{49}$$

where P is the solution to the Riccati equation (39). Equating the right-hand sides of (48) and (49) yields

$$K = \begin{bmatrix} K_q & K_h \end{bmatrix} = -\bar{M}R^{-1}B^T P. \tag{50}$$

This completes the proof. ■

Remark 5: The K vector defined in (37) includes both parameters (31) of the robot impedance model and the gain K_h of the human force. Therefore, the solution to the formulated LQR problem gives the optimal values of the prescribed impedance model parameters and the gain of the human operator force. If the human gain cannot be magnified for a specific HRI application, i.e., if $K_h = 1$, then based on (48) one can set the coefficient of f_h in the control input as $\bar{M}^{-1}$ and then find $\bar{M}$ instead of K_h. That is, if $K_h = 1$ and $\bar{M}$ is unknown, then (50) becomes $K = [\bar{M}^{-1}K_q \;\; \bar{M}^{-1}] = -R^{-1}B^T P$, which gives unknown parameters of the impedance model (2).

Remark 6: The outer-loop control design consists of two components: 1) an adaptive impedance component which finds the optimal values of the parameters (31) of the prescribed impedance model and 2) an assistive component including the human force gain K_h and the feedforward term $\bar{l}(x_d)$ to help the human to minimize the tracking error.

B. Learning Optimal Parameters of the Prescribed Impedance Model Using Integral Reinforcement Learning

Solving (39) requires the knowledge of the matrix A in (36) and consequently the knowledge of the human model. Several model-free RL algorithms have been developed to solve the optimal control of linear systems without requiring any knowledge of the system dynamics [47]–[52]. In this paper, the off-policy integral RL (IRL) algorithm [49]–[52] is used to solve the given LQR problem. The IRL is an iterative policy iteration algorithm for solving (39) that consists of two iteration steps: 1) policy evaluation and 2) policy improvement. In the policy evaluation step, the value function related to a fixed policy is evaluated using an IRL Bellman equation [see (52)] which does not involve the system dynamics. In the policy improvement step, an improved policy is found using the value obtained in the policy evaluation step.

To ensure sufficient exploration of the state space, which is crucial for a proper convergence to the optimal value function, a small exploratory probing noise consisting of sinusoids of varying frequencies is added to the control input to satisfy persistently exciting (PE) qualitatively [53], [54]. Consider the system (47) explored by a known time-varying probing signal e_τ

$$\dot{X} = AX + B\left[u_e + e_\tau\right]. \tag{51}$$

The IRL Bellman equation [49], [50] uses only the information given by measuring the system state and an integral of the utility function in finite reinforcement intervals to evaluate a control policy. The IRL Bellman equation for the given LQR problem for the system (51) including probing noise is given, for time interval $\Delta t > 0$, by [52]

$$\begin{aligned} X(t)^T P X(t) + \int_t^{t+\Delta t} \left[2X(t)^T PB\,e_\tau\right] d\tau \\ = \int_t^{t+\Delta t} \left[X(t)^T Q\,X(t) + u_e^T R\,u_e\right] d\tau \\ + X(t+\Delta t)^T P\,X(t+\Delta t). \end{aligned} \tag{52}$$

This equation explicitly contains the probing noise and is called an off-policy Bellman equation. Using (52) for the policy evaluation step and an update law in form of (49) to

Algorithm 1 Online IRL Algorithm for Outer-Loop Control Design

Initialization: Start with an admissible control input $u^0 = K_1^0 X$
Policy evaluation: Given a control policy u^i, find P^i using the off-policy Bellman equation

$$X(t)^T P^i X(t) + \int_t^{t+\Delta t} [2X(t)^T P^i B\, e_\tau]\, d\tau = \int_t^{t+\Delta t} \left[X(t)^T Q\, X(t) + u_e^T R\, u_e\right] d\tau + X(t+\Delta t)^T P^i X(t+\Delta t). \quad (53)$$

Policy improvement: update the control input using

$$u_e^{i+1} = -R^{-1} B_1^T P^i X. \quad (54)$$

find an improved policy, the following exploratory IRL-based algorithm is obtained for solving (39).

Note that the probing signal e_τ in (51) must be applied during learning to assure convergence of Algorithm 1. After convergence, however, the probing noise is no longer required and can be removed.

Remark 7: Note that for the LQR problem, since the system is linear and the performance function is quadratic, the optimal solution is unique and is found by solving the ARE (39). It is shown in [51] and [52] that the off-policy IRL Algorithm 1 converges to the global optimal solution found by solving the ARE (39), provided that the probing noise is PE. Explicitly including the probing noise in the IRL Bellman equation (52) means that the algorithm converges with no bias, as shown in [52].

Remark 8: The solution for P^i in the policy evaluation step (53) is generally carried out in a least squares (LSs) sense. In fact, (53) is a scalar equation and P^i is a symmetric $n \times n$ matrix with $n(n+1)/2$ independent elements and therefore at least $n(n+1)/2$ data sets are required before (53) can be solved using LS. Consequently, the computational complexity of computing P^i depends on the size of the system.

Remark 9: Note that Algorithm 1 solves the ARE (39) and does not require knowledge of the A matrix which contains knowledge of the human dynamics. In fact, the information of A is embedded in the online measurement of system data.

V. Simulation Results

In this section, the proposed RL-based optimized assistive control and the method of [18] are applied to an x-y table and their simulation results are compared. Then, the proposed method is simulated on a two-link planar robot arm, as a more complicated example.

Example 1 (x-y Table): In order to compare the proposed method to the method presented in [18], a simulation is conducted for a haptic interface system consists of a two degree-of-freedom planer xy-stage. It is assumed that the human operator moves a grip attached to the xy-stage and a pointer on a monitor shows the movement of the position of the grip. The x- and y-axis of the stage are mechanically independent

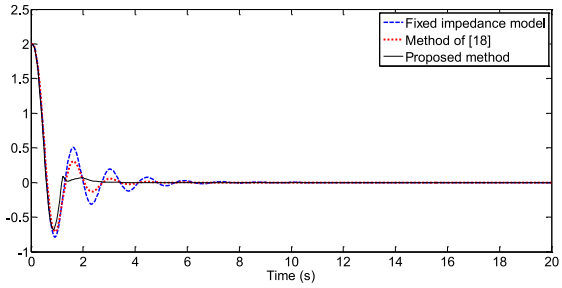


Fig. 6. Tracking error for the proposed method and the method of [18].

and each is driven by a drive motor. A point-to-point task is considered by setting a target point in the work space [18]. The dynamics of the stage is

$$m_p \ddot{x}_p + d_p \dot{x}_p = \tau + k_h f_h. \quad (55)$$

The subscripts of x and y are omitted in (55) because controllers of the x- and y-axis can be designed separately [18]. To perform the simulation, similar to [18], the human transfer function is considered as (35) with $K_p = 779$, $K_d = 288$, and $T = 0.18$. The goal is to move x_p to a desired point x_d which is assumed to be origin here. That is, $x_d = 0$.

The method presented in [18] first identifies the human dynamics and then finds a set of parameters for the stage based on a stability criterion. In the following, to simplify the simulation, we assumed that the human dynamics are known when applying the method of [18]. It is shown that although the method proposed herein does not require to know or to identify the human dynamics, it gives a better performance.

Based on (2), consider the prescribed impedance model as

$$m\ddot{x}_m + d\dot{x}_m + k x_m = f_h. \quad (56)$$

To do a fair comparison, the initial values of the impedance model are chosen as $m = 50$, $d = 50$, and $k = 0$ for both methods. The methods of [18] and the IRL Algorithm 1 are then performed to tune the impedance parameters and improve the performance. The sampling time is chosen as $\Delta t = 0.02$. At the end of the simulation, the improved impedance parameters are found as $m = 25.41$, $d = 68.83$, and $k = 0$ for the method of [18]. For Algorithm 1, a number of ten samples are collected to perform a LS in each iteration. After six iterations (i.e., 1.2 s), the optimal impedance parameters are found as $m = 3.21$, $d = 113.91$, and $k = 44.32$. Figs. 6 and 7 show the performance of the HRI system in terms of the tracking error and the required human force for three different cases: 1) the HRI system with fixed initial impedance model; 2) the HRI system tuned by the method of [18]; and 3) the HRI system optimized by Algorithm 1. It is seen after the learning that the proposed method is faster, has a better performance and requires less effort to perform the task. This is because the proposed method found the optimal set of impedance parameters faster. By contrast, although the method of [18] improves the performance compared to the HRI system with fixed impedance model, it does not converge to an optimal impedance model. These results confirm that the proposed method is faster that the method of [18] and leads to a better performance.

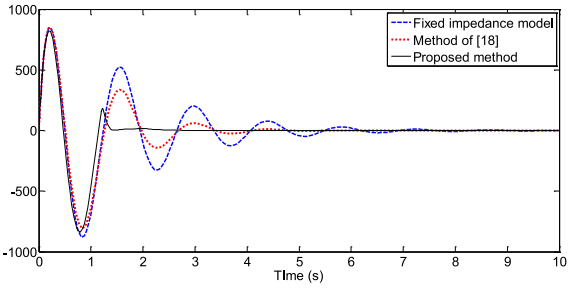


Fig. 7. Required human effort for the proposed method and the method of [18].

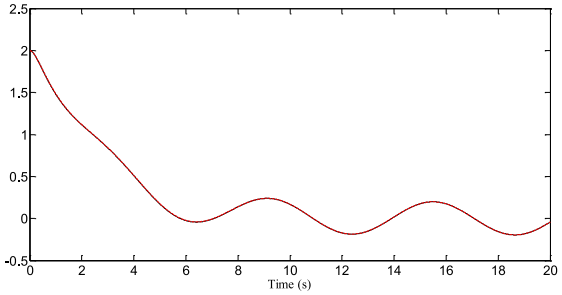


Fig. 8. Trajectory of the robot arm and the prescribed impedance model in x-direction for case 1.

Example 2 (Two-Link Planar Robot Arm): The proposed method is now applied to a two-link robot arm. The length of the links are $L_1 = 1$ m and $L_2 = 1$ m. The masses of the rigid links are $m_1 = 0.8$ kg and $m_2 = 2.3$ kg. The gravitational acceleration is $g = 9.8$ m/s^2. In the following, it is first shown how the proposed inner-loop control design method makes the robot behave like a prescribed impedance model regardless of its impedance parameters. Then, it is shown how to find optimal parameters of the prescribed impedance parameters to make the tracking error in an optimal manner.

A. Inner Loop

It is shown here how a robot two-link planar robot arm behaves like a given impedance model using the inner-loop control design method presented in Section III. The human force is assumed sinusoidal in both directions in this section. The simulations are performed for two different sets of impedance gains to show that the robot behaves like the impedance model regardless of the impedance model parameters.

Case 1: The first matrix gains for the impedance model in (2) are chosen as

$$\bar{M} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \bar{B} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \bar{K} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

Case 2: The second matrix gains for the impedance model in (2) are chosen as

$$\bar{M} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \bar{B} = \begin{bmatrix} 15 & 0 \\ 0 & 15 \end{bmatrix}, \bar{K} = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}.$$

Figs. 8–11 show that the trajectories of the prescribed impedance model are very close to the trajectories of the robot arm for both cases. These results confirm that the robot behaves like the impedance model regardless of the impedance parameters. This enables us to tune the impedance parameters in the outer loop to minimize the tracking error and the human workload. Note that since the initial conditions are known, we start both robot and prescribed impedance model from the same initial conditions.

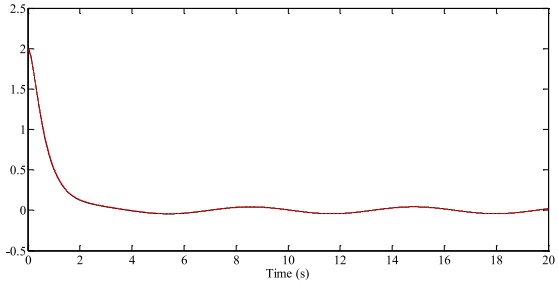


Fig. 9. Trajectory of the robot arm and the prescribed impedance model in y-direction for case 1.

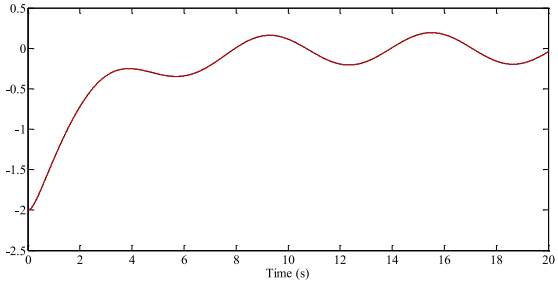


Fig. 10. Trajectory of the robot arm and the prescribed impedance model in x-direction for case 2.

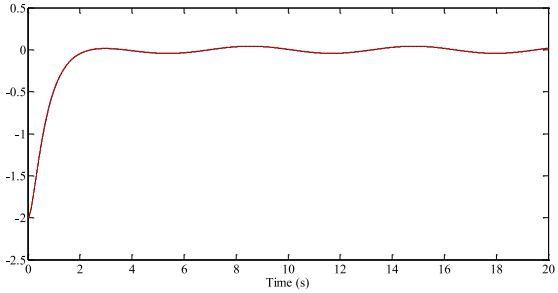


Fig. 11. Trajectory of the robot arm and the prescribed impedance model in y-direction for case 2.

B. Outer Loop

The results of the proposed outer-loop controller method are now presented. It is shown that the proposed online Algorithm 1 gives the same set of parameters as the one found by solving the ARE (39). The performance of the proposed method is verified on a trajectory tracking task. The mass matrix for the prescribed impedance model is set to the identity of appropriate dimension and it is assumed that the desired trajectory to be followed by the robot is $x_d = [\sin(t), \cos(t)]$. It is also assumed that the human admittance parameters are $K_d = 10$, $K_p = 20$, and $h = 1$. Note that, these parameters are generally not constant and may vary from human to human.

The matrices A and B in (36) then become

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0 & 0 & 0 & -2 & 0 \\ 0 & 00.1 & 0 & 0 & 0 & -2 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (57)$$

Then, the solution of the ARE (39) and consequently the control gain (38) are given as

$$P^* = \begin{bmatrix} 507.364 & 0.000 & 7.076 & -0.000 & 3.378 & -0.000 \\ 0.000 & 507.364 & 0.000 & 7.076 & -0.000 & 3.378 \\ 7.075 & 0.000 & 7.170 & -0.000 & 0.046 & 0.000 \\ -0.000 & 7.076 & -0.000 & 7.170 & -0.000 & 0.046 \\ 3.378 & -0.000 & 0.046 & -0.000 & 99.995 & -0.000 \\ -0.000 & 3.378 & 0.000 & 0.046 & -0.000 & 99.995 \end{bmatrix}$$
$$K^* = \begin{bmatrix} -70.758 & 0.000 & -71.704 & 0.000 & 0.458 & 0.000 \\ 0.000 & -70.758 & 0.000 & -71.704 & -0.000 & 0.458 \end{bmatrix} \quad (58)$$

and consequently, the optimal gain matrices for the prescribed impedance model and the human force gain becomes

$$\bar{B} = \begin{bmatrix} 71.704 & -0.000 \\ -0.000 & 71.704 \end{bmatrix}, \bar{K} = \begin{bmatrix} 70.75 & 80.000 \\ -0.000 & 70.758 \end{bmatrix}$$
$$K_h = \begin{bmatrix} 0.458 & 0.000 \\ 0.000 & 0.458 \end{bmatrix}.$$

It is now shown that the proposed online IRL Algorithm 1 gives the same set of the optimal parameters and consequently the same performance as the offline method, but without requiring the knowledge of the human dynamics. To initialize the value function in Algorithm 1, we assume that $K_d = \bar{K}_d + \Delta K_d$ and $K_p = \bar{K}_p + \Delta K_p$, where $\bar{K}_d$ and $\bar{K}_p$ are nominal parameters for an expert human and ΔK_d and ΔK_p changes from one human to another. Note that matrix A_q in A [see (36)] is known and so we can solve the ARE (39) with matrix A containing only nominal values of the human dynamics. This gives us a very appropriate initial value for the value function kernel matrix P.

Fig. 12 shows the convergence of the control gain parameters to their optimal values given in (58) using online Algorithm 1. Note that, Algorithm 1 converges fast after only two iterations because we initialized the value function kernel matrix in an appropriate way. The final gain and kernel matrix found by Algorithm 1 are

$$P_{16} = \begin{bmatrix} 507.364 & 0.000 & 7.076 & -0.000 & 3.378 & -0.003 \\ 0.000 & 507.364 & 0.000 & 7.076 & -0.003 & 3.378 \\ 7.075 & 0.000 & 7.170 & -0.000 & 0.046 & 0.000 \\ -0.000 & 7.076 & -0.000 & 7.170 & -0.000 & 0.046 \\ 3.378 & -0.000 & 0.046 & -0.000 & 99.984 & -0.000 \\ -0.000 & 3.378 & 0.000 & 0.046 & -0.000 & 99.984 \end{bmatrix}$$
$$K_{16} = \begin{bmatrix} -70.758 & 0.000 & -71.704 & 0.000 & 0.461 & 0.000 \\ 0.000 & -70.758 & 0.000 & -71.704 & -0.000 & 0.461 \end{bmatrix}. \quad (59)$$

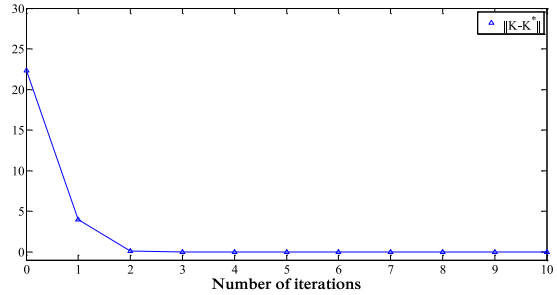


Fig. 12. Convergence of the prescribed impedance gains to their optimal values using online Algorithm 1.

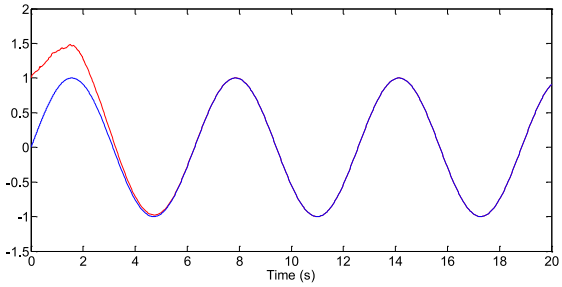


Fig. 13. Trajectory of the robot arm and the desired trajectory in x-direction.

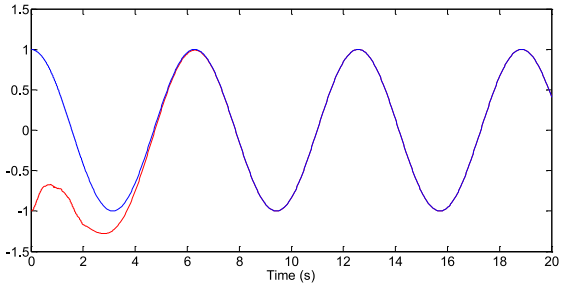


Fig. 14. Trajectory of the robot arm and the desired trajectory in y-direction.

By comparing (59) to (58), it is obvious that the solution found by Algorithm 1 is the same as one found by solving the ARE (39). Figs. 13 and 14 show the results of the proposed method during and after learning. It can be seen that the system states follow the desired trajectories very fast after the learning is finished, because of the proper gains chosen for the prescribed impedance model.

VI. Experimental Implementation Results

In this section, a practical experiment is conducted on a PR2 robot at the University of Texas at Arlington Research Institute. Fig. 15 shows the PR2 robot and the experimental setup. In this experiment, the human operator holds the gripper of the PR2 to perform point-to-point motion between red and blue points along the y-axis, as can be seen in Fig. 15. Human force is measured using an ATI Mini40 FT sensor attached between the gripper and forearm of the PR2. The controller is implemented using the real-time controller manager framework of the PR2 in ROS Groovy. The real-time loop on the PR2 runs at 1000 Hz and communicates with the sensors and actuators on an EtherCAT network.

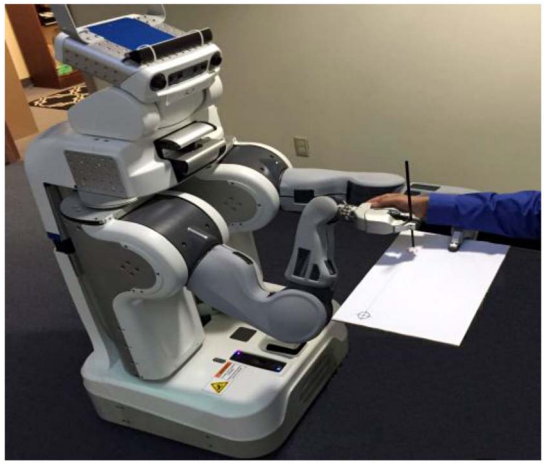


Fig. 15. PR2 robot and the experimental setup.

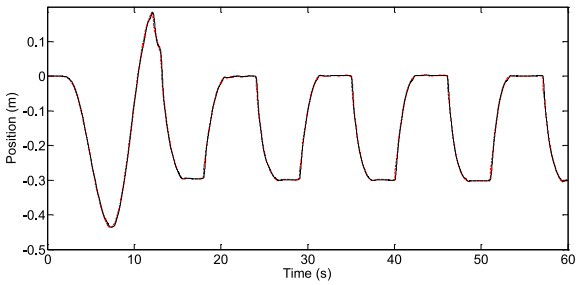


Fig. 16. Inner-loop results: the trajectory of the prescribed impedance control (red) versus the robot trajectory (black).

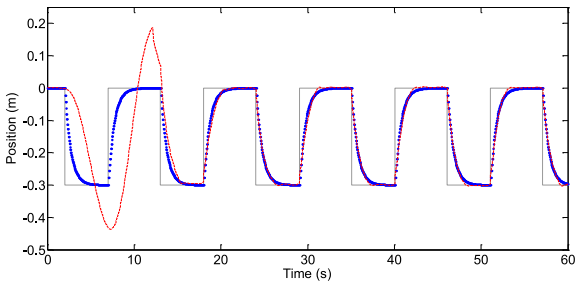


Fig. 17. Outer-loop results: the trajectory of the robot (red) versus the desired trajectory (blue).

The proposed controller is now implemented to this HRI system. Figs. 16–18 show the results of this experiment. Fig. 16 shows that the trajectory of the robot tracks the trajectory of the prescribed impedance model in the inner loop. Fig. 17 shows the outer-loop controller performance. At the beginning, the prescribed impedance model is initialized with a set of nonoptimal parameters and thus the performance of the overall systems is not satisfactory. However, after a short time of interaction between the human and robot, the outer-loop controller learns the optimal parameters for the prescribed impedance model and therefore the HRI system tracks the desired trajectory successfully. Fig. 18 shows how the human force is reduced after the learning is performed and the optimal set of the prescribed impedance model is found by the outer-loop controller.

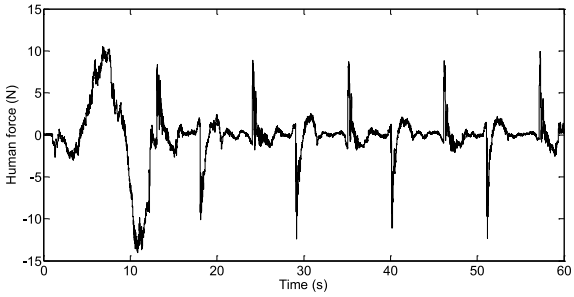


Fig. 18. Human force.

VII. Conclusion

A novel HRI control design method is presented inspired by the human factors studies. The proposed control structure has two control loops. The first loop is an inner-control loop which makes the unknown nonlinear robot look like a prescribed robot impedance model. In contrast to the previous trajectory tracking-based methods, the proposed inner loop does not require the knowledge of the task or the prescribed impedance parameters. This decomposes the robot-specific control design from the task specific design. The second loop is a task-specific loop which includes the human, the robot, and their interaction and finds the optimal parameters of the prescribed impedance parameters to assist the human to perform the task with less effort and optimal performance.

References

[1] A. Mörtl *et al.*, "The role of roles: Physical cooperation between humans and robots," *Int. J. Robot. Res.*, vol. 31, no. 13, pp. 1656–1674, 2012.

[2] J. E. Anderson, D. C. Chang, J. K. Parsons, and M. A. Talamini, "The first national examination of outcomes and trends in robotic surgery in the United States," *J. Amer. Coll. Surg.,* vol. 215, no. 1, pp. 107–114, 2012.

[3] S. Mohammed, Y. Amirat, and H. Rifai, "Lower-limb movement assistance through wearable robots: State of the art and challenges," *Adv. Robot.,* vol. 26, nos. 1–2, pp. 1–22, 2012.

[4] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems.* London, U.K.: Taylor and Francis, 1999.

[5] R.-J. Wai and P.-C. Chen, "Robust neural-fuzzy-network control for robot manipulator including actuator dynamics," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1328–1349, Jun. 2006.

[6] M. K. Ciliz, "Adaptive control of robot manipulators with neural network based compensation of frictional uncertainties," *Robotica*, vol. 23, no. 2, pp. 159–167, 2005.

[7] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators.* Singapore: World Scientific, 1998.

[8] T. Sun, H. Pei, Y. Pan, H. Zhou, and C. Zhang, "Neural network-based sliding mode adaptive control for robot manipulators," *Neurocomputing*, vol. 74, nos. 14–15, pp. 2377–2384, 2011.

[9] N. Kumar, V. Panwar, J. H. Borm, J. Chai, and J. Yoon, "Adaptive neural controller for space robot system with an attitude controlled base," *Neural Comput. Appl.,* vol. 23, nos. 7–8, pp. 2333–2340, 2013.

[10] N. Hogan, "Impedance control: An approach to manipulation. I—Theory. II—Implementation. III—Applications," *ASME Trans. J. Dyn. Syst. Meas. Control*, vol. 107, pp. 1–24, Mar. 1985.

[11] S. S. Ge, C. C. Hang, L. C. Woon, and X. Q. Chen, "Impedance control of robot manipulators using adaptive neural networks," *Int. J. Intell. Control Syst.*, vol. 2, no. 3, pp. 433–452, 1998.

[12] G. Xu and A. Song, "Adaptive impedance control based on dynamic recurrent fuzzy neural network for upper-limb rehabilitation robot," in *Proc. IEEE Int. Conf. Control Autom.*, Christchurch, New Zealand, Dec. 2009, pp. 1376–1381.

[13] L. Huang, S. S. Ge, and T. H. Lee, "Neural network based adaptive impedance control of constrained robots," in *Proc. 2002 IEEE Int. Symp. Intell. Control,* Vancouver, BC, Canada, pp. 615–619.

[14] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 2011, pp. 4326–4332.

[15] S. Hussain, S. Q. Xie, and P. K. Jamwal, "Adaptive impedance control of a robotic orthosis for gait rehabilitation," *IEEE Trans. Cybern.,* vol. 43, no. 3, pp. 1025–1034, Jun. 2013.

[16] S. Jung and T. C. Hsia, "Neural network impedance force control of robot manipulator," *IEEE Trans. Ind. Electron.*, vol. 45, no. 3, pp. 451–461, Jun. 1998.

[17] S. Baron, D. L. Kleinman, and W. H. Levison, "An optimal control model of human response. Part II: Prediction of human performance in a complex task," *Automatica*, vol. 6, no. 3, pp. 371–383, 1970.

[18] S. Suzuki and K. Furuta, "Adaptive impedance control to enhance human skill on a haptic interface system," *J. Control Sci. Eng.*, vol. 2012, pp. 1–10, Jan. 2012.

[19] S. Franklin, D. M. Wolpert, and D. W. Franklin, "Visuomotor feedback gains upregulate during the learning of novel dynamics," *J. Neurophysiol.,* vol. 108, no. 2, pp. 467–478, 2012.

[20] F. Stulp *et al.*, "Model-free reinforcement learning of impedance control in stochastic environments," *IEEE Trans. Auton. Mental Develop.*, vol. 4, no. 4, pp. 330–341, Dec. 2012.

[21] T. Tsuji and Y. Tanaka, "Tracking control properties of human-robotic systems based on impedance control," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 4, pp. 523–535, Jul. 2005.

[22] V. Duchaine and C. Gosselin, "Safe, stable and intuitive control for physical human-robot interaction," in *Proc. IEEE Int. Conf. Robot. Autom.*, Kobe, Japan, 2009, pp. 3383–3388.

[23] R. Ikeura, T. Moriguchi, and K. Mizutani, "Optimal variable impedance control for a robot and its application to lifting an object with a human," in *Proc. 11th IEEE Int. Workshop Robot Hum. Interact. Commun.*, 2002, pp. 500–505.

[24] S. Oh, H. Woo, and K. Kong, "Frequency-shaped impedance control for safe human–robot interaction in reference tracking application," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 6, pp. 1907–1916, Dec. 2014.

[25] 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.

[26] K. Furuta, Y. Kado, and S. Shiratori, "Assisting control in human adaptive mechatronics-single ball juggling," in *Proc. IEEE Int. Conf. Control Appl.*, Munich, Germany, 2006, pp. 545–550.

[27] K. Kosuge, K. Furuta, and T. Yokoyama, "Virtual internal model following control of robot arms," in *Proc. IEEE Int. Conf. Robot. Autom.,* vol. 4. Raleigh, NC, USA, 1987, pp. 1549–1554.

[28] K. Kurihara, S. Suzuki, F. Harashima, and K. Furuta, "Human adaptive mechatronics (HAM) for haptic system," in *Proc. 30th IEEE Annu. Conf. Ind. Electron.,* vol. 1. Busan, Korea, 2004, pp. 647–652.

[29] S. Suzuki, K. Kurihara, K. Furuta, F. Harashima, and Y. Pan, "Variable dynamic assist control on haptic system for human adaptive mechatronics," in *Proc. 44th IEEE Conf. Decis. Control Eur. Control Conf.*, Seville, Spain, Dec. 2005, pp. 4596–4601.

[30] A. Tustin, "The nature of the operator's response in manual control and its implications for controller design," *J. Inst. Elect. Eng. IIA*, vol. 94, no. 2, pp. 190–202, 1947.

[31] J. R. Ragazzini, "Engineering aspects of the human being as a servo-mechanism," presented at the Meeting of the American Psychological Association, Boston, MA, USA, 1948.

[32] Y. Kim, T. Oyabu, G. Obinata, and K. Hase, "Operability of joystick-type steering device considering human arm impedance characteristics," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 2, pp. 295–306, Mar. 2012.

[33] T. Tsumugiwa, R. Yokogawa, and K. Hara, "Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1. Washington, DC, USA, 2002, pp. 644–650.

[34] C. Mitsantisuk, K. Ohishi, and S. Katsura, "Variable mechanical stiffness control based on human stiffness estimation," in *Proc. IEEE Int. Conf. Mechatronics*, Istanbul, Turkey, 2011, pp. 731–736.

[35] Y. Li, S. S. Ge, and C. Yang, "Impedance control for multi-point human-robot interaction," in *Proc. 8th Asian Control Conf. (ASCC)*, Kaohsiung, Taiwan, May 2011, pp. 1187–1192.

[36] C. Wang, Y. Li, S. S. Ge, K. P. Tee, and T. H. Lee, "Continuous critic learning for robot control in physical human-robot interaction," in *Proc. 13th Int. Conf. Control Autom. Syst. (ICCAS),* Gwangju, Korea, Oct. 2013, pp. 833–838.

[37] F. L. Lewis, D. Vrabie, and V. Syrmos, *Optimal Control*, 3rd ed. New York, NY, USA: Wiley, 2012.

[38] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[39] D. P. Bertsekas, *Dynamic Programming and Optimal Control: Approximate Dynamic Programming*, 4th ed. Belmont, MA, USA: Athena Scientific, 2012.

[40] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. New York, NY, USA: Wiley, 2007.

[41] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control," *IEEE Control Syst. Mag.*, vol. 32, no. 6, pp. 76–105, Dec. 2012.

[42] D. Vrabie, K. G. Vamvoudakis, and F. L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles* (Control Engineering Series). Stevenage, U.K.: IET, 2012.

[43] Z. Wang, A. Peer, and M. Buss, "An HMM approach to realistic haptic human–robot interaction," in *Proc. 3rd Joint Eurohaptics Conf. Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst.*, Salt Lake City, UT, USA, 2009, pp. 374–379.

[44] M. S. Erden and T. Tomiyama, "Human-intent detection and physically interactive control of a robot without force sensors," *IEEE Trans. Robot.*, vol. 26, no. 2, pp. 370–382, Apr. 2010.

[45] S. S. Ge, Y. Li, and H. He, "Neural-network-based human intention estimation for physical human-robot interaction," in *Proc. Int. Conf. Ubiquitous Robot. Ambient Intell.*, Incheon, Korea, 2011, pp. 390–395.

[46] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2003.

[47] B. Kiumarsi, F. L. Lewis, M. B. Naghibi-Sistani, and A. Karimpour, "Optimal tracking control of unknown discrete-time linear systems using input–output measured data," *IEEE Trans. Cybern.*, to be published.

[48] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Trans. Autom. Control*, vol. 59, no. 11, pp. 3051–3056, Nov. 2014.

[49] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, Feb. 2009.

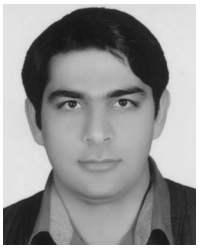
[50] D. Vrabie and F. L. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Netw.*, vol. 22, no. 3, pp. 237–246, Apr. 2009.

[51] Y. Jiang and Z. P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[52] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral reinforcement learning for continuous-time input-affine nonlinear systems with simultaneous invariant explorations," *IEEE Trans. Cybern.*, to be published.

[53] H. Modares, F. L. Lewis, and M. B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive control of partially-unknown continuous-time systems," *Automatica*, vol. 50, no. 1, pp. 193–202, 2014.

[54] H. Modares, F. L. Lewis, and M. B. Naghibi-Sistani, "Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 10, pp. 1513–1525, Oct. 2013.



Hamidreza Modares received the B.S. degree from the University of Tehran, Tehran, Iran, and the M.S. degree from the Shahrood University of Technology, Shahrud, Iran, in 2004 and 2006, respectively. He is currently pursuing the Ph.D. degree with the University of Texas at Arlington, Arlington, TX, USA.

He joined the Shahrood University of Technology as a University Lecturer, from 2006 to 2009. Since 2012, he has been a Research Assistant with the University of Texas at Arlington Research Institute, Fort Worth, TX, USA. His current research interests include optimal control, reinforcement learning, distributed control, robotics, and pattern recognition.

Isura Ranatunga (S'09) received the B.Sc. degree from the University of Texas at Arlington, Arlington, TX, USA, in 2010, where he is currently pursuing the Ph.D. degree with a focus on robotics and automation, both in electrical engineering.

He is a Graduate Research Assistant with the University of Texas at Arlington Research Institute, Fort Worth, TX, USA, and the Next Generation Systems Research Group. His current research interests include force control, physical human–robot interaction, bipedal walking, adaptive robot control, and autonomous navigation.

Frank L. Lewis (S'70–M'81–SM'86–F'94) received the bachelor's degree in physics and electrical engineering and the M.S.E.E. degree, both from Rice University, Houston, TX, USA, the M.S. degree in aeronautical engineering from the University of West Florida, Pensacola, FL, USA, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA.

He is a University of Texas at Arlington Distinguished Scholar Professor, a Teaching Professor, and the Moncrief-O'Donnell Chair with the University of Texas at Arlington Research Institute, Fort Worth, TX, USA. He is the Qian Ren Thousand Talents Consulting Professor with Northeastern University, Shenyang, China. He is a Distinguished Visiting Professor with the Nanjing University of Science and Technology, Nanjing, China, and the Project 111 Professor with Northeastern University. His current research interests include feedback control, intelligent systems, cooperative control systems, and nonlinear systems. He has authored numerous journal special issues, journal papers, 20 books, including *Optimal Control*, *Aircraft Control*, *Optimal Estimation*, and *Robot Manipulator Control*, which are used as university textbooks worldwide and he holds six U.S. patents.

Dr. Lewis was a recipient of the Fulbright Research Award, the National Science Foundation Research Initiation Grant, the American Society for Engineering Education Terman Award, the International Neural Network Society Gabor Award, the U.K. Institute of Measurement and Control Honeywell Field Engineering Medal, the IEEE Computational Intelligence Society Neural Networks Pioneer Award, the Outstanding Service Award from Dallas IEEE Section, and selected as an Engineer of the Year by the Fort Worth IEEE Section. He was listed in Fort Worth Business Press Top 200 Leaders in Manufacturing and Texas Regents Outstanding Teaching Award in 2013. He is a PE of Texas and a U.K. Chartered Engineer. He is a member of the National Academy of Inventors and a fellow of International Federation of Automatic Control and the U.K. Institute of Measurement and Control. He is a Founding Member of the Board of Governors of the Mediterranean Control Association.

Dan O. Popa (M'93) received the B.A. degree in engineering, mathematics, and computer science and the M.S. degree in engineering, both from Dartmouth College, Hanover, NH, USA, and the Ph.D. degree in electrical, computer and systems engineering from Rensselaer Polytechnic Institute (RPI), Troy, NY, USA, in 1998, focusing on control and motion planning for nonholonomic systems and robots.

He is an Associate Professor with the Department of Electrical Engineering, University of Texas at Arlington, and the Head of the Next Generation Systems Research Group. He joined the Center for Automation Technologies at RPI, where he was a Research Scientist until 2004, for over 20 industry-sponsored projects. He was an Affiliated Faculty Member of the University of Texas at Arlington Research Institute, Fort Worth, TX, USA, and a Founding Member of the Texas Microfactory Initiative, in 2004. His current research interests include the simulation, control, packaging of microsystems, the design of precision robotic assembly systems, and control and adaptation aspects of human–robot interaction. He has authored over 100 refereed publications.

Dr. Popa was a recipient of several prestigious awards, including the University of Texas Regents Outstanding Teaching Award. He serves as an Associate Editor of the IEEE TRANSACTION ON AUTOMATION SCIENCE AND ENGINEERING and the *Journal of Micro and Bio Robotics (Springer)*. He is an active member of the IEEE Robotics and Automation Society Conference Activities Board, the IEEE Committee on Micro-Nano Robotics, and the ASME Committee on Micro-Nano Systems and a member of ASME.