Impedance Learning for Robots Interacting with Unknown Environments

Yanan Li and Shuzhi Sam Ge, *Fellow, IEEE*

***Abstract*—In this paper, impedance learning is investigated for robots interacting with unknown environments. A two-loop control framework is employed and adaptive control is developed for the inner-loop position control. The environments are described as time-varying systems with unknown parameters in the state-space form. The gradient-following scheme and betterment scheme are employed to obtain a desired impedance model, subject to unknown environments. The desired interaction performance is achieved in the sense that a defined cost function is minimized. Simulation and experiment studies are carried out to verify the validity of the proposed method.**

***Index Terms*—Impedance learning, interaction control, robotic control, adaptive control, unknown environment.**

I. Introduction

Nowadays, robots are anticipated to participate in and learn from long-term interactions with different environments, and to be safely deployed in myriad social applications such as elderly care, health care, and human-robot cooperation. In these applications, the environments are usually unknown and dynamically changing, which brings along many problems to control engineers.

Impedance control is usually employed in interaction control and its robustness and feasibility have been acknowledged by many research studies [1], [2], [3], [4], [5], [6], [7], [8], [9]. In particular, a desired passive impedance model is usually prescribed and the safety of the robot and environment is guaranteed. However, to prescribe an impedance model is prone to be conservative in many situations, and a better interaction performance can be expected with other choices [10], [11]. Besides, a fixed impedance model may not suffice in many applications, and variable impedance is necessary [12], [13]. Therefore, to obtain the critical values of desired impedance parameters is essential and it is still an open problem due to the extreme difficulty of environment modeling [14], [15], [16]. Instead of modeling environments, human beings adjust their limb impedance through repetitive learning subject to unknown environments. For example, when a person opens a door, he/she may fail in the first time because he/she does not have the knowledge of this door, e.g., mass, inertia, friction at the hinge etc. After he/she "tries" to open the door for several times, he/she is able to open the door to a desired position with the least effort. During the process of opening a door, this person learns a "best" set of impedance parameters of his/her limb in the sense that the target position is achieved and the control effort is minimized.

It is possible to apply human beings' learning skill discussed above to robot control [17], [18], [19]. Specifically, the robot dynamics can be governed by a target impedance model with impedance control. Then, in a similar way as human beings adjust their limb impedance, parameters of the target impedance model are adjusted through learning based on a certain criteria. This kind of learning schemes has been developed in many research studies. In [20], associative search network learning is applied to a wall-following task. In [21], impedance parameters are regulated through learning of neural networks. However, as discussed in [22], artificial neural networks techniques need an expensive data preprocessing for training examples in order to learn. In [23], internal-model-based learning is developed for a high-speed insertion task. This method has a simple formulation but it is limited to a simple application. Instead, reinforcement learning is based on the trial-and-error method, which is more similar to the way of human learning [24]. In [25], an equilibrium point control model is employed and the stiffness matrix is updated according to different application requirements using natural actor-critic algorithm. In [26], a supervised learning paradigm is employed to acquire the plant dynamics and its stochastic properties, such that optimal control in the sense of task accuracy and energy cost is achieved. In [27], PI2 (policy improvement with path integrals) algorithm is adopted to make reinforcement learning in high degrees-of-the-freedom robotic systems become possible. Different from the above works of reinforcement learning, the environment dynamics are described explicitly in this paper so that it is possible to obtain the gradient of the defined cost function. In this way, the proposed impedance learning has a straightforward yet much simpler framework, and it provides feasibility in practical implementations. It is able to adjust time-varying stiffness and damping matrices simultaneously, and is applicable for a large range of applications.

When the desired impedance parameters are obtained through learning, there are several methods to impose the desired impedance model on the robot arm. An approach widely used is to design an inner position control loop in conjunction with an outer force control loop. This strategy is appealing for robot systems that are designed for position regulation, including almost all industrial robots [28]. In this strategy, the force control loop is used to determine the virtual desired trajectory. Then the position control loop serves to track the virtual desired trajectory, and the control objective becomes trajectory tracking which has been widely studied in

Yanan Li is with the Institute for Infocomm Research, A*STAR, Singapore 138632. hit.li.yn@gmail.com

Shuzhi Sam Ge is with the Social Robotics Laboratory, Interactive Digital Media Institute and the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576, and the Robotics Institute, and School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China. samge@nus.edu.sg

the past decades. It has been shown that a simple proportional-derivative (PD) control is able to guarantee a closed-loop system with a bounded tracking error, and this error may be reduced by increasing the PD gains [29]. To completely eliminate the tracking error, PD control is usually companied with computed-torque control [30]. However, the performance of computed-torque control is limited by the computation complexity and the existence of uncertainties. In the control literature, adaptive control has been extensively studied to cope with the problem of parametric and non-parametric uncertainties [31], [32]. In [33], adaptive control is proposed by introducing the regressor and using linear-in-the-parameters property. This method has been employed and developed in many other research studies. In [34], [35], adaptive control is developed with the regressor computed off line, and the simplicity of the control design is slightly improved. Although the model-based adaptive control mentioned above guarantees trajectory tracking, the knowledge of robot structure is required and the difficulty is caused in practice. To find a method without using the robot structure information, much effort is made with machine learning approaches, e.g., neural networks [36], [37], [38]. Instead of that, in this work, we will propose an adaptive control without using either the regressor or the machine learning approaches.

Based on the above discussion, we highlight the contributions of this paper as follows:

(i) Physical robot-environment interaction is investigated, where the environment is described as a time-varying system in the state-space form.
(ii) Impedance learning is proposed to iteratively adjust the impedance parameters of the robot arm, such that the desired impedance model is obtained despite unknown environments.
(iii) Adaptive control is developed for the position control in the two-loop control framework, and trajectory tracking is achieved without using the regressor.

The rest of this paper is organized as follows. In Section II, the system overview of a robot interacting with unknown environments is given, and the control objective is discussed. In Sections III and IV, the details of the proposed impedance learning and position control are presented, respectively. In Sections V-A and V-B, simulation and experiment with different control objectives are conducted to verify the feasibility of the proposed method. Concluding remarks are given in Section VI.

II. System Overview

A. Dynamic Model

In this paper, we consider a system where a rigid robot arm physically interacts with an unknown environment. Suppose that the kinematics of the robot arm are given by

$$x(t) = \phi(q(t)) \tag{1}$$

where $x(t)$, $q(t) \in \mathbb{R}^n$, and n are positions/oritations in the Cartesian space (operational space), joint coordinates in the joint space, and degree-of-the-freedom (DOF), respectively. Differentiating (1) with respect to time results in

$$\dot{x}(t) = J(q(t))\dot{q}(t) \tag{2}$$

where $J(q(t)) \in \mathbb{R}^{n\times n}$ is the Jacobian matrix.

The dynamics of the robot arm in the joint space are given by

$$\begin{aligned} &M(q(t))\ddot{q}(t) + C(q(t),\dot{q}(t))\dot{q}(t) + G(q(t)) \\ &= \tau(t) - f(t) \end{aligned} \tag{3}$$

where $M(q(t)) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $C(q(t),\dot{q}(t))\dot{q}(t) \in \mathbb{R}^n$ denotes the Coriolis and Centrifugal force, $G(q(t)) \in \mathbb{R}^n$ is the gravitational force, $\tau(t) \in \mathbb{R}^n$ is the control input, and $f(t) \in \mathbb{R}^n$ denotes the interaction force exerted by the environment and it can be measured by a force/torque sensor.

Property 1: [39] $\|M(q(t))\| \leq k_M$, $\|C(q(t),\dot{q}(t))\| \leq k_C\|\dot{q}(t)\|$ and $\|G(q(t))\| \leq k_G$, where k_M, k_C, k_G are unknown positive scalars.

Suppose that the environment dynamics are described by the following mass-damping-spring model

$$M_E(t)\ddot{q}(t) + C_E(t)\dot{q}(t) + G_E(t)q(t) = f(t) \tag{4}$$

where $M_E(t)$, $C_E(t)$, and $G_E(t)$ are inertia, damping, and stiffness matrices of the environment dynamics, respectively. Note that $M_E(t)$, $C_E(t)$, and $G_E(t)$ are unknown and they are only used for the analysis. This model represents a large range of environments [40]. For example, it may describe the dynamics of a human limb in physical human-robot interaction [16], or the dynamics of a viscoelastic object in robotic manipulation.

B. Control Objective

The objective of this work is to control the interaction between the robot arm and the environment. To achieve it, we develop a control architecture as shown in Fig. 1. In particular, we consider the following target impedance model

$$\begin{aligned} &M_d(t)(\ddot{q}_d(t) - \ddot{q}_r(t)) + C_d(t)(\dot{q}_d(t) - \dot{q}_r(t)) \\ &+G_d(t)(q_d(t) - q_r(t)) = f(t) \end{aligned} \tag{5}$$

where $M_d(t)$, $C_d(t)$, and $G_d(t)$ are the desired inertia, damping, and stiffness matrices, respectively, $q_d(t)$ is the desired trajectory, and $q_r(t)$ is the virtual desired trajectory of the position control loop.

Two steps are included in this control architecture: impedance learning and position control. In the first step, we need to find $M_d(t)$, $C_d(t)$, and $G_d(t)$ to achieve a certain desired interaction performance, and the environment dynamics have to be taken into consideration. However, previous studies have shown that it is extremely difficult to obtain an exact model of the environment in many situations, e.g., human limb. In this regard, we aim to develop an iterative learning law, which is able to find $M_d(t)$, $C_d(t)$, and $G_d(t)$ through repetitive interactions and does not require the environment model. In the second step, position control is developed to

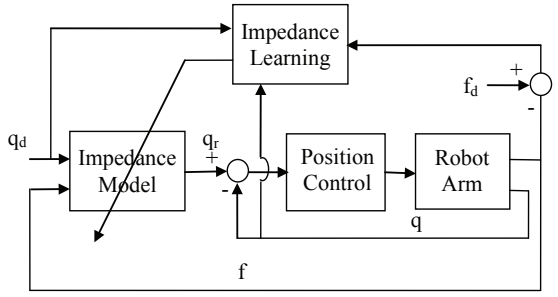

Fig. 1. Control architecture

make $q(t) \to q_r(t)$ as $t \to \infty$, such that the impedance model becomes

$$\begin{aligned} &M_d(t)(\ddot{q}_d(t) - \ddot{q}(t)) + C_d(t)(\dot{q}_d(t) - \dot{q}(t)) \\ &+G_d(t)(q_d(t) - q(t)) = f(t) \end{aligned} \quad (6)$$

Note that only the virtual desired trajectory $q_r(t)$ will be refined according to (5), while the control performance of the inner position control loop is not affected by the outer-loop impedance learning. In the following two sections, the proposed impedance learning and position control will be discussed in detail.

III. Impedance Learning

A. Preliminary: Generic Betterment Scheme

For the development of impedance learning in the following subsection, we introduce the following lemma about the betterment scheme.

Lemma 1: [41] Consider the following linear time-varying systems described by

$$\begin{aligned} \dot{\xi}(t) &= A(t)\xi(t) + B(t)u(t), \\ v(t) &= C(t)\xi(t),\ \xi(t) \in R^m,\ u(t),\ v(t) \in R^r \end{aligned} \quad (7)$$

The control input $u(t)$ is iteratively updated as

$$u^k(t) = u^{k-1}(t) + \alpha'[\dot{v}_d(t) - \dot{v}^k(t)] \quad (8)$$

where k is the iteration number, $v_d(t)$ is the desired output, and α' satisfies the following inequality

$$\|I - \alpha' B(t)C(t)\|_\infty < 1 \quad (9)$$

with I as the unit matrix of a proper dimension.

If $C(t)B(t)$ is nonsingular and $v^k(0) = v_d(0)$, then the betterment process for system (7) is convergent in the sense that $v^k(t) \to v_d(t)$ uniformly in $t \in [0, t_f]$ as $k \to \infty$, where t_f is the iteration period.

B. The Betterment Scheme Applied to Impedance Learning

To develop impedance learning, a cost function to measure the interaction performance is usually defined. We denote this cost function as $\Gamma(t)$ and explain it later. Because arbitrary selection of $M_d(t)$ may cause instability [42], it is fixed to equal to the apparent inertia and only $C_d(t)$ and $G_d(t)$ are updated during the learning process.

To gradually decrease $\Gamma(t)$ by updating $C_d(t)$ and $G_d(t)$, the gradient-following scheme is employed and the following learning law is proposed

$$\begin{aligned} C_d^k(t) &= C_d^{k-1}(t) - \beta_C(\frac{\partial \Gamma^k(t)}{\partial C_d^k(t)})^T \\ &= C_d^{k-1}(t) - \beta_C(\frac{\partial f^k(t)}{\partial C_d^k(t)})^T(\frac{\partial \Gamma^k(t)}{\partial f^k(t)})^T \\ G_d^k(t) &= G_d^{k-1}(t) - \beta_G(\frac{\partial \Gamma^k(t)}{\partial G_d^k(t)})^T \\ &= G_d^{k-1}(t) - \beta_G(\frac{\partial f^k(t)}{\partial G_d^k(t)})^T(\frac{\partial \Gamma^k(t)}{\partial f^k(t)})^T \end{aligned} \quad (10)$$

where positive scalars β_C and β_G represent the learning rates.

According to (6), we have

$$\frac{\partial f^k(t)}{\partial C_d^k(t)} = \dot{e}_q^{k^T}(t),\ \frac{\partial f^k(t)}{\partial G_d^k(t)} = e_q^{k^T}(t) \quad (11)$$

where $e_q(t) = q_d(t) - q(t)$. Because the environment dynamics are unknown, the gradient of reinforcement $\frac{\partial \Gamma^k(t)}{\partial f^k(t)}$ is not available. The betterment scheme in Lemma is adopted to cope with this problem in the following.

In order to employ the betterment scheme in Lemma 1, the environment model (4) has to be represented in the state-space form. Particularly, by choosing states $x_1(t) = q(t)$, $x_2(t) = \dot{q}(t)$ and $x_3(t) = \int_0^t f(s)ds$, we rewrite (4) in the following state-space form

$$\begin{aligned} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} &= \begin{bmatrix} 0 & I_n & 0 \\ -M_E^{-1}(t)G_E(t) & -M_E^{-1}(t)C_E(t) & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &\times \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ -M_E^{-1}(t) \\ I_n \end{bmatrix} f(t) \end{aligned} \quad (12)$$

In the above formulation, we assume that $f(0) = 0$ such that $\dot{x}_3(t) = f(t) - f(0) = f(t)$. By denoting

$$\begin{aligned} \xi(t) &= \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} \\ A(t) &= \begin{bmatrix} 0 & I_n & 0 \\ -M_E^{-1}(t)G_E(t) & -M_E^{-1}(t)C_E(t) & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ B(t) &= \begin{bmatrix} 0 \\ -M_E^{-1}(t) \\ I_n \end{bmatrix} \end{aligned} \quad (13)$$

we further write the above equation into a more compact form

$$\dot{\xi}(t) = A(t)\xi(t) + B(t)f(t) \quad (14)$$

To be coherent with the denotation in Lemma 1, we have

$$v(t) = C(t)\xi(t) \quad (15)$$

where $C(t)$ defines the relationship between the states (i.e., position, velocity, integral of interaction force) and the output $v(t)$.

As indicated by Lemma 1, if we take the interaction force $f(t)$ as the "control input" to the environment dynamics (14), it will be updated as

$$\begin{aligned} f^k(t) &= f^{k-1}(t) + \alpha'(\dot{v}_d(t) - \dot{v}^k(t)) \\ &= f^{k-1}(t) - \alpha'(\dot{v}^k(t) - \dot{v}_d(t)) \end{aligned} \quad (16)$$

where α' satisfies the inequality (9), such that $v^k(t) \to v_d(t)$ as $k \to \infty$. In other words, the "control input" f is iteratively updated to decrease the error between $v^k(t)$ and $v_d(t)$. Approximately, we measure this error by the cost function $\Gamma(t) = \|v(t) - v_d(t)\|_2$ where $\|\cdot\|_2$ denotes 2-norm, and obtain

$$f^k(t) = f^{k-1}(t) - \beta(\frac{\partial \Gamma^k(t)}{\partial f^k(t)})^T \quad (17)$$

Remark 1: The definition of the cost function $\Gamma(t)$ indicates that the aim of impedance learning in this paper can be trajectory tracking, integral force tracking or the combination/compromise of these two, by choosing different $C(t)$. For example, if the control objective is integral force tracking, we may choose $C(t) = [0, 0, c]^T$, where c is a constant. Note that the defined cost function includes position, velocity and integral interaction force which are different quantities with different units of measurements. Therefore, in practical implementations, partial knowledge of both the robot and environment and trial-and-error may be needed to define a proper cost function. This is the same as in the well-known linear quadratic regulator [43], where a typical cost function includes position, velocity, and control torque, and it is nontrivial to determine their weights.

Remark 2: The cost function based method has been employed in most works of impedance learning in the literature, such as wall following [20], ball inserting [23], door opening and ball catching [25], and explosive movement tasks [13].

Comparing (16) and (17), we obtain

$$\frac{\partial \Gamma^k(t)}{\partial f^k(t)} = \alpha(\dot{v}^k(t) - \dot{v}_d(t))^T \quad (18)$$

where β has been absorbed by α as $\alpha = \frac{\alpha'}{\beta}$.

Substituting (11) and (18) to (10), we obtain the learning law

$$\begin{aligned} C_d^k(t) &= C_d^{k-1}(t) - \alpha\beta_C \dot{e}_q^k(t)(\dot{v}^k(t) - \dot{v}_d(t))^T \\ G_d^k(t) &= G_d^{k-1}(t) - \alpha\beta_G e_q^k(t)(\dot{v}^k(t) - \dot{v}_d(t))^T \end{aligned} \quad (19)$$

Remark 3: It is found that the above learning law has a simple formulation, which is developed based on the sensory feedback from the environment instead of modeling the environment. The sensory feedback includes the position and velocity errors $e_q(t)$ and $\dot{e}_q(t)$, which are respectively used to update $G_d(t)$ and $C_d(t)$. It may also include the force error $e_f(t) = f_d(t) - f(t)$, which is introduced by the defined cost function $\Gamma(t)$. However, learning rates β_C and β_G are open parameters which are difficult to tune in sophisticated tasks [44]. Further investigations on this issue are needed in practical implementations.

Remark 4: As the gradient of reinforcement $\frac{\partial \Gamma^k(t)}{\partial f^k(t)}$ is difficult to obtain in the presence of unknown environments, various estimation methods have been proposed in the literature. In [23], $\frac{\partial \Gamma^k(t)}{\partial f^k(t)}$ is estimated based on an internal model which is identified from the data collection of input $f^k(t)$ and output $\Gamma^k(t)$. Suppose that the internal model is obtained as $\hat{\Gamma}^k(t) = \varphi^T f^k(t)$, then the gradient of reinforcement is estimated as $\frac{\partial \hat{\Gamma}^k(t)}{\partial f^k(t)} = \varphi^T$. In this paper, however, we employ the betterment scheme as stated in Lemma 1. Under the betterment scheme, a time series of input signal to a plant is iteratively updated using an error signal between the output signal and the target signal such that the output signal at the next iteration approaches the target signal [21]. As a result, the estimation process is avoided.

IV. Position Control

As M_d, C_d, and G_d have been obtained through the outer-loop impedance learning, the virtual desired trajectory q_r is obtained according to the impedance model (5). Position control will be developed in this section to make the actual position q track q_r.

A. Preliminary: PD-Like Control

The following PD-like control was developed in [34], [35] for position control

$$\tau_{pd}(t) = K_p \text{Sin}_{vec}(e(t)) + K_d \dot{e}(t) + f(t) \quad (20)$$

where $e(t) = q_r(t) - q(t)$ is the tracking error, K_p and K_d are diagonal matrices with positive entries, $\text{Sin}_{vec}(e(t)) = [\text{Sin}(e_1(t)), \text{Sin}(e_2(t)), \ldots, \text{Sin}(e_n(t))]^T$ with $e_i(t)$, $i = 1, 2, \ldots, n$ as the components of $e(t)$, and $\text{Sin}(e_i(t))$ is defined as

$$\text{Sin}(e_i(t)) = \begin{cases} 1, & e_i(t) \geq \frac{\pi}{2}; \\ \sin(e_i(t)), & |e_i(t)| < \frac{\pi}{2}; \\ -1, & e_i(t) \leq -\frac{\pi}{2}. \end{cases} \quad (21)$$

Lemma 2: [34], [35] Denoting an auxiliary variable

$$y(t) = -\dot{e}(t) - \mu \text{Sin}_{vec}(e(t)) \quad (22)$$

with $\mu > 0$, we have the following inequality

$$\begin{aligned} & \{[M(q(t))\ddot{e}(t) + C(q(t), \dot{q}(t))\dot{e}(t) - K_p \text{Sin}_{vec}(e(t)) \\ & -K_d\dot{e}(t)] + [M(q(t)) - M(q_r(t))]\ddot{q}_r(t) \\ & +[C(q(t), \dot{q}(t)) - C(q_r(t), \dot{q}_r(t))]\dot{q}_r(t) \\ & +[G(q(t)) - G(q_r(t))]\}^T y(t) \\ \geq\ & V(e(t), \dot{e}(t)) + \frac{d}{dt}(W(e(t), \dot{e}(t))) \end{aligned} \quad (23)$$

where

$$\begin{aligned} & W(e(t), \dot{e}(t)) = \frac{1}{2}\dot{e}^T(t)M(q(t))\dot{e}(t) + \\ & \sum_{i=1}^{n} p_i(1 - \text{Cos}(\dot{q}_i(t))) + \mu \text{Sin}_{vec}^T(e(t))M(q(t))\dot{e}(t) \\ & V(e(t), \dot{e}(t)) = \dot{e}^T(t)\{K_d - c_1 I - \alpha[M(q(t)) + \\ & (c_2 + c_3)I]\}\dot{e}(t) + \text{Sin}_{vec}^T(e(t))(\mu K_p - c_4 I)\text{Sin}_{vec}(e(t)) \end{aligned} \quad (24)$$

with $p_i, i = 1, 2, \ldots, n$ as the diagonal entries of K_p, $q_i(t), i = 1, 2, \ldots, n$ as the components of $q(t)$, $c_j, j = 1, 2, 3, 4$ as some constants, and $\text{Cos}(\dot{q}_i(t))$ defined as

$$\text{Cos}(\dot{q}_i(t)) = \begin{cases} -\dot{q}_i(t) + \frac{\pi}{2}, & \dot{q}_i(t) \geq \frac{\pi}{2}; \\ \cos(\dot{q}_i(t)), & |\dot{q}_i(t)| < \frac{\pi}{2}; \\ \dot{q}_i(t) + \frac{\pi}{2}, & \dot{q}_i(t) \leq -\frac{\pi}{2}. \end{cases} \quad (25)$$

$V(e(t), \dot{e}(t))$ and $W(e(t), \dot{e}(t))$ are positive definite in $e(t)$ and $\dot{e}(t)$, if $\mu > 0$, and K_p and K_d are chosen properly.

Lemma 3: According to Property 1, the following inequality holds

$$[-M(q_r(t))\ddot{q}_r(t) - C(q_r(t), \dot{q}_r(t))\dot{q}_r(t) - G(q_r(t)) - K\text{sgn}_{vec}(y(t))]^T y(t) \leq 0 \quad (26)$$

where $K = \text{diag}(k_1, k_2, \ldots, k_n)$ and $\|K\| = k_M l_1 + k_C l_2^2 + k_G$, with l_1 and l_2 as the upper bounds of $\|\ddot{q}_r(t)\|$ and $\|\dot{q}_r(t)\|$, respectively, $\text{sgn}_{vec}(y(t)) = [\text{sgn}(y_1(t)), \text{sgn}(y_2(t)), \ldots, \text{sgn}(y_n(t))]^T$ with $y_1(t), y_2(t), \ldots, y_n(t)$ as the components of $y(t)$, and $\text{sgn}(\cdot)$ denotes the sign function.

Proof 1: See Appendix VII-A.

B. Adaptive Control

In [34], [35], it has been shown that the position control of the robot arm can be guaranteed by PD-like control (20) plus a learning term. This learning term only employs an auxiliary variable composed of the tracking error and velocity error. The above results motivate the control design in this paper, which includes PD-like control (20) plus an adaptive term. The extension from the learning to adaptive case is not straightforward. The underlying reason is that while the whole robot dynamics with the desired trajectory are invariant in each iteration, only physical parameters (e.g., inertia and link length) are invariant in each adaptive period, without considering the uncertainty. In this situation, we need to find something inherently invariant in the adaptive case, which are found to be the upper-bounds of the robot dynamics, as shown in Property 1. Note that these upper-bounds are not necessarily to be known, which can be estimated with the developed updating law.

In particular, the proposed adaptive position control is given by

$$\begin{aligned} \tau(t) &= K_p\text{Sin}_{vec}(e(t)) + K_d\dot{e}(t) \\ &\quad -\hat{K}\text{sgn}_{vec}(y(t)) + f(t) \end{aligned} \quad (27)$$

where $\hat{K}(t)$ is a diagonal matrix with elements $\hat{k}_i(t)$, $i = 1, 2, \ldots, n$. For the analysis convenience, we denote

$$\begin{aligned} k(t) &= [k_1(t), k_2(t), \ldots, k_n(t)]^T \\ \hat{k}(t) &= [\hat{k}_1(t), \hat{k}_2(t), \ldots, \hat{k}_n(t)]^T \end{aligned} \quad (28)$$

and thus $K = \text{diag}(k)$ and $\hat{K}(t) = \text{diag}(\hat{k}(t))$.

The updating law for $\hat{k}(t)$ is developed as

$$\dot{\hat{k}}(t) = S^{-1}\bar{y}(t) \quad (29)$$

where S is a positive definite matrix and $\bar{y}(t) = [y_1(t)\text{sgn}(y_1(t)), \ldots, y_n(t)\text{sgn}(y_n(t))]^T$. Note that after $\hat{k}(t)$ is obtained from (29), $\hat{K}(t)$ in (27) is obtained as $\hat{K}(t) = \text{diag}(\hat{k}(t))$.

Considering (3) and the control input (27), the closed-loop system dynamics are obtained as

$$\begin{aligned} &M(q(t))\ddot{q}(t) + C(q(t), \dot{q}(t))\dot{q}(t) + G(q(t)) \\ &-K_p\text{Sin}_{vec}(e(t)) - K_d\dot{e}(t) + \hat{K}(t)\text{sgn}_{vec}(y(t)) = 0 \end{aligned} \quad (30)$$

Theorem 1: Given the dynamics (3), with the developed control (27) and updating law (29), position control of the robot arm is achieved, i.e., $\lim_{t\to\infty} e(t) = 0$. Besides, all the signals in the closed-loop system (30) are bounded.

Proof 2: See Appendix VII-B.

Remark 5: In [35], $\tau(t) = K_p\text{Sin}_{vec}(e(t)) + K_d\dot{e}(t) + Y(\ddot{q}_r(t), \dot{q}_r(t), q_r(t))\hat{\theta}(t) + f(t)$ with the updating law $\dot{\hat{\theta}}(t) = -S^{-1}Y^T(\ddot{q}_r(t), \dot{q}_r(t), q_r(t))y(t)$. $\hat{\theta}(t)$ is the estimate of $\theta \in \mathbb{R}^{n_\theta}$, θ is a vector of the physical parameters of the robot arm, n_θ is a positive integer denoting the number of these parameters, and $Y(\ddot{q}(t), \dot{q}(t), q(t)) \in \mathbb{R}^{n\times n_\theta}$ is the regression matrix. It employs the linear-in-parameters property, which causes a difficulty especially when the robot has a high DOF, because the robot structure is required as a priori knowledge. Similarly, most adaptive control designs for position control in the literature employ the linear-in-parameters property, e.g., [33], [29]. Instead of that, the proposed adaptive control in this paper does not require any robot dynamics information, and is thus model-free and provides feasibility for practical implementations.

Remark 6: Compared to the methods in [33], [29], [35], the second-order derivative of $q_r(t)$, i.e., $\ddot{q}_r(t)$, is not used in the proposed position control. This may help in some applications where $\ddot{q}_r(t)$ is not available.

V. Simulation and Experiment

A. Simulation Study

In this section, we verify the validity of the proposed impedance learning and adaptive control through simulation studies. A 6-DOF PUMA560 robot is considered and this simulation is implemented with the robotics toolbox introduced in [45].

In the first part of the simulation study, the effectiveness of the proposed adaptive control is verified, and the control performances with different control parameters are discussed. The initial position of the robot arm in the joint space is $q(0) = [0, 0, 0, 0, 0, 0]^T$. The desired trajectory of the robot arm for each joint is given by $12(\frac{t}{t_f})^5 - 30(\frac{t}{t_f})^4 + 20(\frac{t}{t_f})^3$, where $t_f = 5$. Control parameters in (27) and (29) are summarized in Table I. Note that in the first three cases, the proposed adaptive control is not adopted so the parameter S is not available.

Inner-loop tracking errors of each joint with different control parameters are shown in Fig. 2. It is found that tracking errors can be reduced by increasing K_p and K_d but they cannot be eliminated, and the stability of the system may be destroyed when K_p and K_d are too large. Tracking errors asymptotically converge to zero when the proposed adaptive control is adopted, of which the validity has been verified. The convergence rate can be increased by reducing S, but too small S may also cause system instability. Besides, for the

TABLE I
CONTROL PARAMETERS

Case	K_p	K_d	S
1	$\mathrm{diag}\,[500, 50, 50, 5, 5, 5]$	$\mathrm{diag}\,[25, 5, 5, 0.5, 0.5, 0.5]$	–
2	$\mathrm{diag}\,[1000, 100, 100, 10, 10, 10]$	$\mathrm{diag}\,[25, 5, 5, 0.5, 0.5, 0.5]$	–
3	$\mathrm{diag}\,[1000, 100, 100, 10, 10, 10]$	$\mathrm{diag}\,[50, 10, 10, 1, 1, 1]$	–
4	$\mathrm{diag}\,[1000, 100, 100, 10, 10, 10]$	$\mathrm{diag}\,[50, 10, 10, 1, 1, 1]$	$\mathrm{diag}\,[1, 0.1, 0.2, 1, 1, 1]$
5	$\mathrm{diag}\,[1000, 100, 100, 10, 10, 10]$	$\mathrm{diag}\,[50, 10, 10, 1, 1, 1]$	$\mathrm{diag}\,[0.2, 0.02, 0.04, 0.2, 0.2, 0.2]$

same control parameters, the tracking error is also affected by the desired trajectory. Particularly, further simulation can be carried out to show that it takes a longer time to track a "faster" desired trajectory. It is essential to emphasize that only the parameters in Table I have been tuned during the simulation, and the control design is much easier than adaptive control in [29], [33], [34], [35].

In the second part of the simulation study, the validity of the proposed impedance learning is verified. The environment dynamics are described by (4) with $M_E(t) = 0.01(\sin \pi t)^2 I$, $C_E(t) = 0.1(\sin \pi t)^2 I$, and $G_E(t) = 20(\sin \pi t)^2 I$, which are time-varying and unknown to the designer. To show the robustness of the proposed impedance learning, the discontinuity of the environment dynamics has also been considered during the simulation. In particular, the interaction force is considered to suddenly drop to 0 at $t = 1$s, i.e., $M_E(t) = 0$, $C_E(t) = 0$, and $G_E(t) = 0$ for $t > 1$s. The initial position and the desired trajectory are the same as in the first part of the simulation study. The control parameters in (27) and (29) are $K_p = \mathrm{diag}\,[1000, 100, 100, 10, 10, 10]$, $K_d = \mathrm{diag}\,[50, 10, 10, 1, 1, 1]$, and $S = \mathrm{diag}\,[0.2, 0.02, 0.04, 0.2, 0.2, 0.2]$. $M_d(t)$ is fixed to equal to the apparent inertia and the initial values of $C_d(t)$ and $G_d(t)$ in the first iteration ($k = 0$) are $C_d^0(t) = 10I$ and $G_d^0(t) = 10I$. The parameters in (19) are $\alpha\beta_C = 20$ and $\alpha\beta_G = 20$, which can be adjusted to modulate the convergence rate of the learning process. As discussed in Section III, the control objective can be trajectory tracking, integral force tracking and the combination/comprimise of these two with the proposed impedance learning, by choosing different cost functions.

In the first case, we choose $C(t) = [50, 0, 1]^T$, $f_d = 0$, and thus the cost function is $\Gamma(t) = \|50e_q(t) - \int_0^t e_f(s)ds\|_2$. 10 iterations are performed and the simulation results are shown in Figs. 3, 4, and 5. On the right-hand side of each figure, there is a color-bar which illustrates that the colors of simulation results become darker when the iteration number increases. Note that position errors and interaction forces in Fig. 4 are the simulation results of Joint 1. The results of other joints are similar and thus not shown in this simulation. From Fig. 3, it is found that the cost function becomes smaller when the iteration number increases. This is followed by the result that the position error becomes smaller while the interaction force from 0s to 1s becomes larger when the iteration number increases, as shown in Fig. 4. Correspondingly, impedance parameters from 0s to 1s become larger when the iteration number increases, as shown in Fig. 5. Observing the impedance parameters of the environment dynamics with respect to time, the impedance parameters of the robot arm in Fig. 5 are updated correspondingly. For example, the peak of the interaction force appears at around 0.6s when the impedance parameters of the environment dynamics become the largest, and the impedance parameters of the robot arm also become the largest. Besides, the interaction force is set to suddenly drop to 0 at $t = 1$s, and there is discontinuity at this point. As a result, an overshoot appears after $t = 1$s in Fig. 5. Nonetheless, the robot arm moves smoothly which can be observed by the position error in Fig. 4. The above results have indicated that the robot arm increases its impedance parameters iteratively to resist the interference from the environment, while it keeps its impedance parameters when there is no interference. These results are similar to that in [27] and in accord with the human motor control performance.

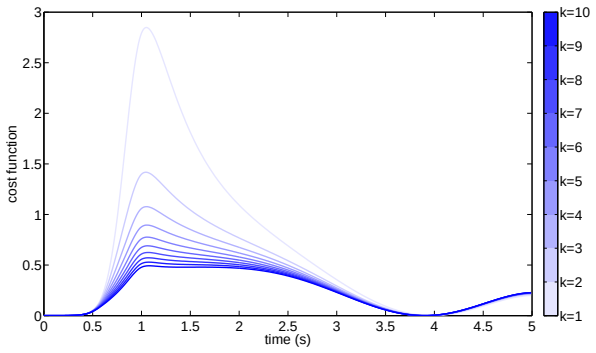

Fig. 3. Cost functions in the first case

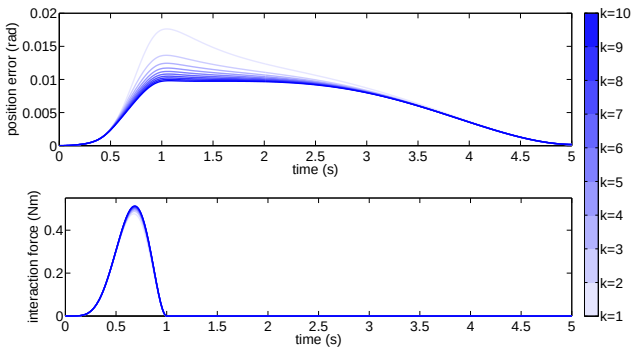

Fig. 4. Position errors and interaction forces in the first case

In the first case, the weight of the position error in the cost function is 50 while the weight of the integral interaction force is 1, which indicates that trajectory tracking is more important than zero force regulation. Therefore, impedance

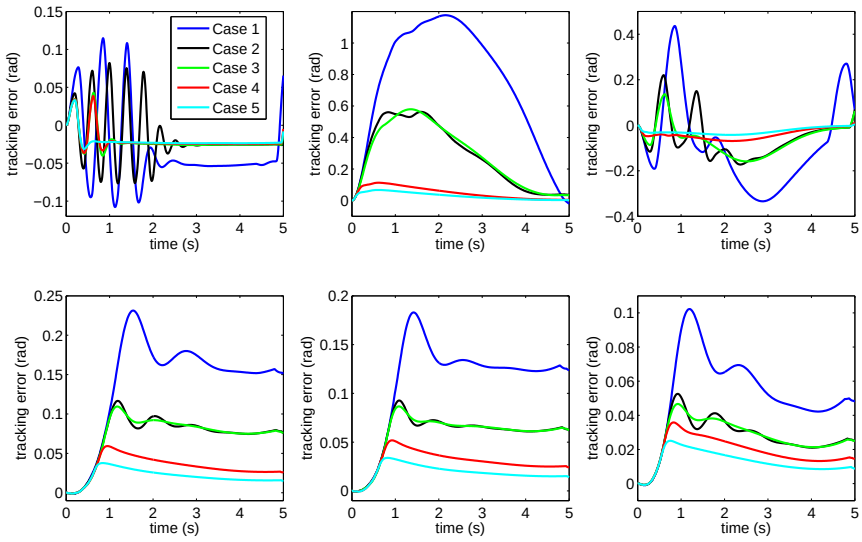

Fig. 2. Inner-loop tracking errors with different control parameters (Upper, from left to right: Joints 1, 2 and 3; Below, from left to right: Joints 4, 5 and 6)

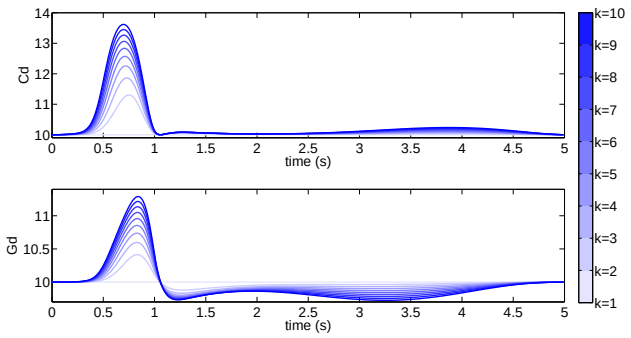

Fig. 5. Damping and stiffness parameters in the first case

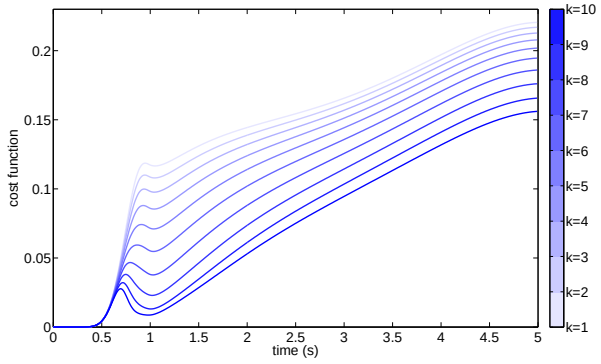

Fig. 6. Cost functions in the second case

parameters become larger such that the robot arm stiffen up to resist the interference from the environment. Subsequently, the position error becomes smaller and the interaction force becomes larger. In the second case, we change $C(t)$ to $C(t) = [3, 0, 1]^T$, and thus the cost function becomes $\Gamma(t) = \|3e_q(t) - \int_0^t e_f(s)ds\|_2$. As the weight of the position error in this case is smaller than that in the first case, it is expected that the robot arm becomes more compliant. 10 iterations are performed and the simulation results are shown in Figs. 6, 7, and 8. In particular, the impedance parameters shown in Fig. 8 indeed become smaller from 0s to 1s. Accordingly, in Fig. 7 it is shown that the interaction force becomes smaller while the position error becomes larger. In Fig. 6, the cost function still becomes smaller as the iteration number increases, although the performance is very different from that in Fig. 3. Similarly as in Fig. 5, there is also an overshoot in Fig. 8 due to the existence of discontinuity in the environment dynamics, but it does not have an obvious effect on the control performance of the robot arm.

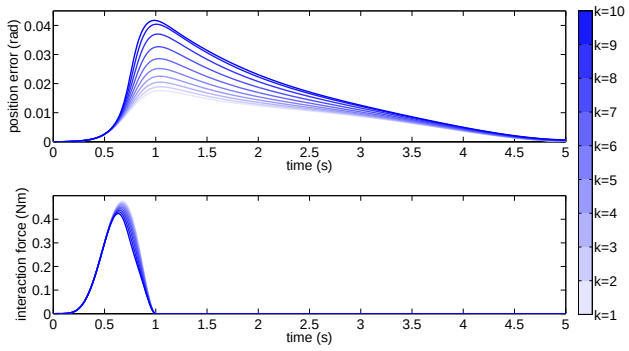

Fig. 7. Position errors and interaction forces in the second case

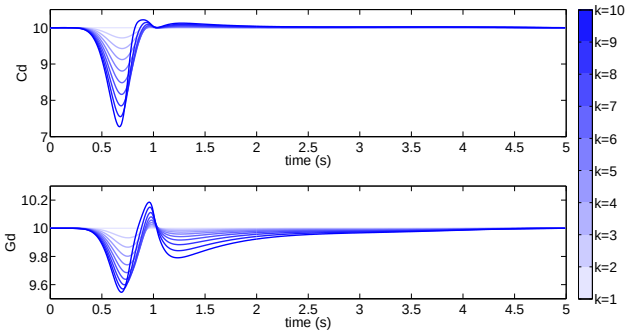

Fig. 8. Damping and stiffness parameters in the second case

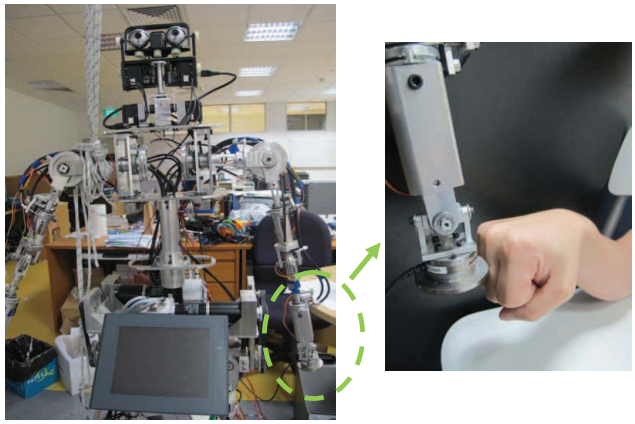

Fig. 9. Nancy and experiment scenario

B. Experiment Study

In this section, the proposed method is further examined on a real robot, Nancy, which is developed in Social Robotics Laboratory, National University of Singapore [46]. The motor which drives the joint is controlled by Maxon's EPOS2 70/10 dual loop controller. It works in the CANopen network and provides multiple operational modes including position, velocity and current modes. An ATI mini-40 force/torque sensor is installed at the left wrist of Nancy to measure the force/torque exerted by environments.

In this experiment, the left wrist of Nancy follows a desired trajectory $q_r(t) = 0.02t$ rad while it interacts with a human hand which plays the role of unknown environment, as shown in Fig. 9. The left wrist of Nancy moves back to the original position at the beginning of each iteration, and the position of the human hand is fixed during the interaction. In each iteration with a period of 18s, the interaction starts at $t = 5$s and ends at $t = 16$s. The interaction phases, i.e., interaction and non-interaction, switch according to the time displayed on the computer screen. Similarly as in simulation studies, the interaction force drops to zero at $t = 16$s immediately, and thus there is discontinuity. Two cases with different control objectives are considered. In these two cases, $M_d(t)$ is fixed to equal to the apparent inertia and the initial values of $C_d(t)$ and $G_d(t)$ are $C_d^0(t) = 3.6$ and $G_d^0(t) = 3.6$. The parameters in (19) are $\alpha\beta_C = 40$ and $\alpha\beta_G = 40$. Other values of these parameters can be chosen to adjust the convergence rate of the learning process.

In the first case, we choose the cost function as $\Gamma(t) = \|10e_q(t) - \int_0^t e_f(s)ds\|_2$ and then trajectory tracking is more important than zero force regulation. In Figs. 10 and 11, the results at different iterations $k = 0, 5, 10$ are respectively shown. From Fig. 11, it is found that the position error becomes smaller when the iteration number increases. Correspondingly, the stiffness parameter $G_d(t)$ becomes larger and the defined cost function becomes smaller when the iteration number increases, as shown in Fig. 10. The result of the damping parameter $C_d(t)$ is similar to that of $G_d(t)$, and is thus omitted. Similarly as in simulation studies, the above results have revealed the expected interaction performance: impedance parameters become larger to make the robot arm stiffen up such that the interference from the environment is resisted, and they keep unchanged if there is no interference from the environment.

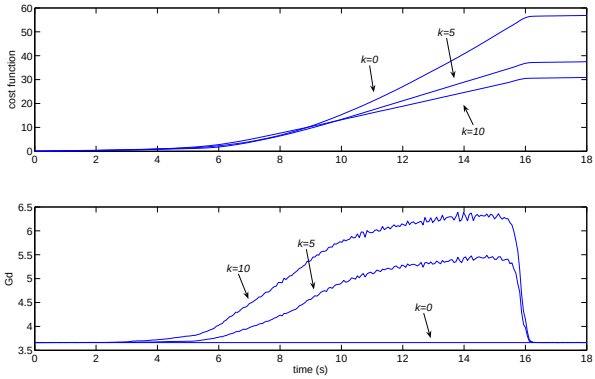

Fig. 10. Cost functions and stiffness parameters in the first case

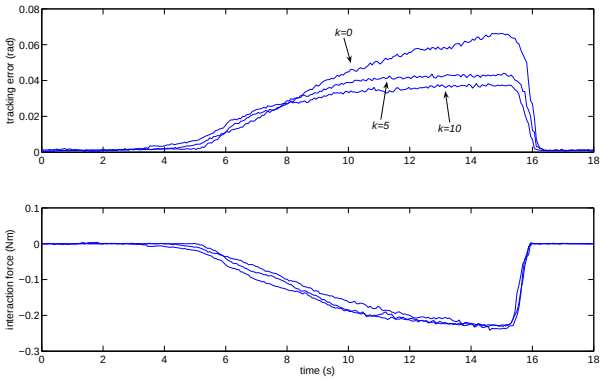

Fig. 11. Position errors and interaction forces in the first case

In the second case, we change the cost function to $\Gamma(t) = \|e_q(t) - \int_0^t e_f(s)ds\|_2$, in which the weight of trajectory tracking is smaller and it is expected that the robot arm becomes more compliant. The results at $k = 0, 5, 10$ in the second case are shown in Figs. 12 and 13. Different from that in Figs. 10 and 11, the position error becomes larger

and correspondingly the stiffness parameter $G_d(t)$ becomes smaller when the iteration number increases. It is obvious that the robot arm becomes more compliant in this case. As the interaction force drops to zero immediately at $t = 6$s, there is an overshoot in the result of stiffness parameter in Fig. 12, but it does not have obvious effect on the motion of the robot arm. Besides, the force signal is typically noisy and the learning process is not as smooth as that in simulation studies. While the above results are acceptable, these practical issues need to be further considered for better interaction control.

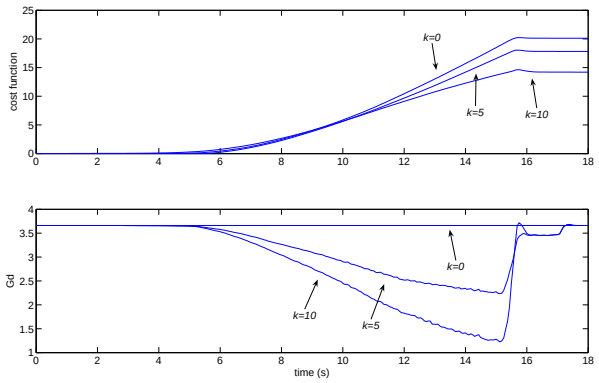

Fig. 12. Cost functions and stiffness parameters in the second case

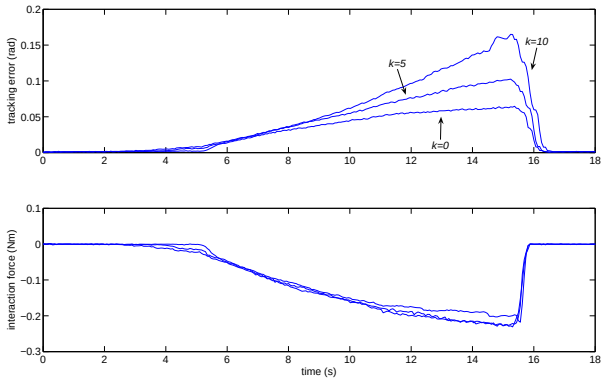

Fig. 13. Position errors and interaction forces in the second case

To summarize, by choosing different cost functions, it is determined that the control objective can be trajectory tracking, integral force tracking or the combination/comprimise of these two. The proposed impedance learning guarantees that the defined cost function becomes smaller and subsequently the control objective is achieved, subject to unknown dynamic environments. The advantage of the proposed impedance learning over impedance control with fixed impedance parameters lies in: a modest performance can be obtained if a good set of fixed impedance parameters is predefined (when $k = 1$), and a better performance can be obtained only with variant impedance parameters because the environments are dynamically changing.

VI. Conclusion

In this paper, the problem of robots interacting with unknown environments has been investigated. Impedance learning has been developed to obtain desired impedance parameters subject to unknown dynamic environments. The proposed impedance learning has employed gradient-following and betterment schemes. Adaptive control has been developed for the trajectory tracking in the inner position control loop, and subsequently the control objective has been achieved. The feasibility and validity of the proposed method have been verified by simulation and experiment.

VII. Appendix

A. Proof of Lemma 3

$$\begin{aligned}
& [-M(q_r(t))\ddot{q}_r(t) - C(q_r(t),\dot{q}_r(t))\dot{q}_r(t) - G(q_r(t)) \\
& -K\text{sgn}_{vec}(y(t))]^T y(t) \\
\leq\ & [\|M(q_r(t))\ddot{q}_r(t)\| + \|C(q_r(t),\dot{q}_r(t))\dot{q}_r(t)\| \\
& +\|G(q_r(t))\|]\|y(t)\| - \|Ky(t)\| \\
\leq\ & [\|M(q_r(t))\|\|\ddot{q}_r(t)\| + \|C(q_r(t),\dot{q}_r(t))\|\|\dot{q}_r(t)\| \\
& +\|G(q_r(t))\|]\|y(t)\| - \|Ky(t)\| \\
\leq\ & (k_M\|\ddot{q}_r(t)\| + k_C\|\dot{q}_r(t)\|\|\dot{q}_r(t)\| \\
& +k_G)\|y(t)\| - \|K\|\|y(t)\| \leq 0
\end{aligned} \tag{31}$$

B. Proof of Theorem 1

Rewrite the closed-loop dynamics (30) as

$$\begin{aligned}
& M(q(t))\ddot{q}(t) + C(q(t),\dot{q}(t))\dot{q}(t) + G(q(t)) \\
& -K_p\text{Sin}_{vec}(e(t)) - K_d\dot{e}(t) + \hat{K}\text{sgn}_{vec}(y(t)) \\
=\ & [M(q(t))\ddot{e}(t) + C(q(t),\dot{q}(t))\dot{e}(t) - K_p\text{Sin}_{vec}(e(t)) \\
& -K_d\dot{e}(t)] + [M(q(t))\ddot{q}_r(t) + C(q(t),\dot{q}(t))\dot{q}_r(t) \\
& +G(q(t))] + \hat{K}\text{sgn}_{vec}(y(t)) \\
=\ & [M(q(t))\ddot{e}(t) + C(q(t),\dot{q}(t))\dot{e}(t) - K_p\text{Sin}_{vec}(e(t)) \\
& -K_d\dot{e}(t)] + \{[M(q(t)) - M(q_r(t))]\ddot{q}_r(t) \\
& +[C(q(t),\dot{q}(t)) - C(q_r(t),\dot{q}_r(t))]\dot{q}_r(t) + [G(q(t)) \\
& -G(q_r(t))]\} + [M(q_r(t))\ddot{q}_r(t) + C(q_r(t),\dot{q}_r(t))\dot{q}_r(t) \\
& +G(q_r(t))] + \hat{K}\text{sgn}_{vec}(y(t)) \\
=\ & [M(q(t))\ddot{e}(t) + C(q(t),\dot{q}(t))\dot{e}(t) - K_p\text{Sin}_{vec}(e(t)) \\
& -K_d\dot{e}(t)] + \{[M(q(t)) - M(q_r(t))]\ddot{q}_r(t) \\
& +[C(q(t),\dot{q}(t)) - C(q_r(t),\dot{q}_r(t))]\dot{q}_r(t) \\
& +[G(q(t)) - G(q_r(t))]\} + [M(q_r(t))\ddot{q}_r(t) \\
& +C(q_r(t),\dot{q}_r(t))\dot{q}_r(t) + G(q_r(t))] + [K\text{sgn}_{vec}(y(t)) \\
& +\tilde{K}\text{sgn}_{vec}(y(t))] = 0
\end{aligned} \tag{32}$$

where $\tilde{K}(t) = \hat{K}(t) - K$. From the above equation, we obtain

$$\begin{aligned}
& M(q(t))\ddot{e}(t) + C(q(t),\dot{q}(t))\dot{e}(t) - K_p\text{Sin}_{vec}(e(t)) \\
& -K_d\dot{e}(t) + \{[M(q(t)) - M(q_r(t))]\ddot{q}_r(t) \\
& +[C(q(t),\dot{q}(t)) - C(q_r(t),\dot{q}_r(t))]\dot{q}_r(t) + [G(q(t)) \\
& -G(q_r(t))]\} + \tilde{K}\text{sgn}_{vec}(y(t)) \\
=\ & -M(q_r(t))\ddot{q}_r(t) - C(q_r(t),\dot{q}_r(t))\dot{q}_r(t) - G(q_r(t)) \\
& -K\text{sgn}_{vec}(y(t))
\end{aligned} \tag{33}$$

According to Lemma 2, we have

$$\begin{aligned}
&\{[M(q(t))\ddot{e}(t)+C(q(t),\dot{q}(t))\dot{e}(t)-K_p\text{Sin}_{vec}(e(t))\\
&-K_d\dot{e}(t)]+[M(q(t))-M(q_r(t))]\ddot{q}_r(t)\\
&+[C(q(t),\dot{q}(t))-C(q_r(t),\dot{q}_r(t))]\dot{q}_r(t)+[G(q(t))\\
&-G(q_r(t))]\}^T y(t)\\
\geq\ & V(e(t),\dot{e}(t))+\frac{d}{dt}[W(e(t),\dot{e}(t))] \qquad (34)
\end{aligned}$$

Besides, we have the following result according to the updating law (29)

$$\begin{aligned}
\frac{d}{dt}(\frac{1}{2}\tilde{k}^T S\tilde{k}(t)) &= \tilde{k}^T(t)S\dot{\tilde{k}}(t)=\tilde{k}^T(t)S\dot{\hat{k}}(t)=\tilde{k}^T(t)\bar{y}(t)\\
&= \tilde{K}(t)\text{sgn}_{vec}(y(t))y^T(t) \qquad (35)
\end{aligned}$$

Substituting the above equation into (34), we have

$$\begin{aligned}
&\{[M(q(t))\ddot{e}(t)+C(q(t),\dot{q}(t))\dot{e}(t)-K_p\text{Sin}(e(t))\\
&-K_d\dot{e}(t)]+[M(q(t))-M(q_r(t))]\ddot{q}_r(t)\\
&+[C(q(t),\dot{q}(t))-C(q_r(t),\dot{q}_r(t)]\dot{q}_r(t)\\
&+[G(q(t))-G(q_r(t))]\}^T y(t)+\tilde{K}(t)y^T(t)\text{sgn}_{vec}(y(t))\\
\geq\ & V(e(t),\dot{e}(t))+\frac{d}{dt}[W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)] \qquad (36)
\end{aligned}$$

which leads to

$$\begin{aligned}
&[-M(q_r(t))\ddot{q}_r(t)-C(q_r(t),\dot{q}_r(t))\dot{q}_r(t)-G(q_r(t))\\
&-K\text{sgn}_{vec}(y(t))]^T y(t)\geq\\
&V(e(t),\dot{e}(t))+\frac{d}{dt}[W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)] \qquad (37)
\end{aligned}$$

According to Lemma 3, we obtain

$$V(e(t),\dot{e}(t))+\frac{d}{dt}[W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)]\leq 0 \qquad (38)$$

and thus

$$\frac{d}{dt}[W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)]\leq -V(e(t),\dot{e}(t)) \qquad (39)$$

Taking the integral of both sides of the above equation, we have

$$\begin{aligned}
&W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)-W(e(0),\dot{e}(0))\\
&-\frac{1}{2}\tilde{k}(0)S\tilde{k}(0)\leq -\int_0^t V(e(s),\dot{e}(s))ds\leq 0 \qquad (40)
\end{aligned}$$

The above inequality indicates that $W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)\leq W(e(0),\dot{e}(0))+\frac{1}{2}\tilde{k}(0)S\tilde{k}(0)$ and thus $W(e(t),\dot{e}(t))+\frac{1}{2}\tilde{k}(t)S\tilde{k}(t)$ is bounded suppose $W(e(0),\dot{e}(0))$ and $\frac{1}{2}\tilde{k}(0)S\tilde{k}(0)$ are bounded. According to Lemma 2, $W(e(t),\dot{e}(t))$ is positive definite in $e(t)$ and $\dot{e}(t)$, thus $W(e(t),\dot{e}(t))$ and $\tilde{k}(t)S\tilde{k}(t)$ are bounded, and immediately we obtain $e(t)\in L_\infty$, $\dot{e}(t)\in L_\infty$, and $\tilde{k}(t)\in L_\infty$. Besides, it is found from the above inequality that $\int_0^t V(e(s),\dot{e}(s))ds$ is bounded, and thus we have $e(t)\in L_2$ as $V(e(t),\dot{e}(t))$ is positive definite in $e(t)$ and $\dot{e}(t)$. $\dot{e}(t)\in L_\infty$ and $e(t)\in L_2$ lead to $\lim_{t\to\infty}e(t)=0$, which completes the proof.

REFERENCES

[1] N. Hogan, "Impedance control: an approach to manipulation-Part I: Theory; Part II: Implementation; Part III: Applications," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 1–24, 1985.

[2] J. J. Gonzalez and G. R. Widmann, "A force commanded impedance control scheme for robots with hard nonlinearities," *IEEE Transactions on Control Systems Technology*, vol. 3, no. 4, pp. 398–408, 1995.

[3] C. H. Wu, , K. S. Hwang, and S. L. Chang, "Analysis and implementation of a neuromuscular-like control for robotic compliance," *IEEE Transactions on Control Systems Technology*, vol. 5, no. 6, pp. 586–597, 1997.

[4] R. B. Gillespie, J. E. Colgate, and M. A. Peshkin, "A general framework for cobot control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 391–401, August 2001.

[5] K. M. Lynch, C. Liu, A. Sorensen, S. Kim, M. Peshkin, J. E. Colgate, T. Tickel, D. Hannon, and K. Shiels, "Motion guides for assisted manipulation," *International Journal of Robotics Research*, vol. 21, no. 1, pp. 27–43, January 2002.

[6] S. Jung, T. C. Hsia, and R. G. Bonitz, "Force tracking impedance control of robot manipulators under unknown environment," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 3, pp. 474–483, 2004.

[7] Y. Hirata, Z. Wang, K. Fukaya, and K. Kosuge, "Transporting an object by a passive mobile robot with servo brakes in cooperation with a human," *Advanced Robotics*, vol. 23, pp. 387–404, 2009.

[8] L. P. J. Selen, D. W. Franklin, and D. M. Wolpert, "Impedance control reduces instability that arises from motor noise," *The Journal of Neuroscience*, vol. 29, no. 40, pp. 12606–12616, 2009.

[9] Y. Li, S. S. Ge, and C. Yang, "Learning impedance control for physical robot-environment interaction," *International Journal of Control*, vol. 85, no. 2, pp. 182–193, 2012.

[10] S. P. Buerger and N. Hogan, "Complementary stability and loop shaping for improved human-robot interaction," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 232–244, 2007.

[11] V. Duchaine and C. Gosselin, "Investigation of human-robot interaction stability using Lyapunov theory," in *IEEE International Conference on Robotics and Automation, Piscataway, NJ, United States*, pp. 2189–2194, 2008.

[12] T. Tsumugiwa, R. Yokogawa, and K. Hara, "Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task," *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pp. 644–650, 2002.

[13] D. J. Braun, M. Howard, and S. Vijayakumar, "Optimal variable stiffness control: formulation and application to explosive movement tasks," *Autonomous Robots*, vol. 33, pp. 237–253, 2012.

[14] R. Ikeura and H. Inooka, "Variable impedance control of a robot for cooperation with a human," *Proceedings of the 1995 IEEE International Conference on Robotics and Automation*, pp. 3097–3102, 1995.

[15] T. Tsumugiwa, R. Yokogawa, and K. Hara, "Variable impedance control with regard to working process for man-machine cooperation-work system," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1564–1569, 2001.

[16] M. Rahman, R. Ikeura, and K. Mizutani, "Investigation of the impedance characteristic of human arm for development of robots to cooperate with humans," *JSME International Journal Series C*, vol. 45, no. 2, pp. 510–518, 2002.

[17] R. M. Sanner and M. Kosha, "A mathematical model of the adaptive control of human arm motions," *Biological Cybernetics*, vol. 80, pp. 369–382, 1999.

[18] T. Liu and C. Wang, "Learning from neural control in motor systems," *Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics*, pp. 1995–2000, 2007.

[19] G. Ganesh, A. Albu-Schäffer, M. Haruno, M. Kawato, and E. Burdet, "Biomimetic motor behavior for simultaneous adaptation of force, impedance and trajectory in interaction tasks," *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2705–2711, 2010.

[20] M. Cohen and T. Flash, "Learning impedance parameters for robot control using an associative search network," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 382–390, June 1991.

[21] T. Tsuji and P. G. Morasso, "Neural Network learning of robot arm impedance in operational space," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 26, no. 2, pp. 290–298, 1996.

[22] D. S. M. F. A. Albers, S. Schillo and P. Meckl, "A new two-layer reinforcement learning approach the control of a 2DOF manipulator,"

2010 8th IEEE International Conference on Control and Automation (ICCA), pp. 546–551, 2010.
[23] B. H. Yang and H. Asada, "Progressive learning and its application to robot impedance learning," *IEEE Transactions on Neural Networks*, pp. 941–952, 1996.
[24] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.
[25] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 40, no. 2, pp. 433–443, 2010.
[26] D. Mitrovic, S. Klanke, and S. Vijayakumar, "Learning impedance control of antagonistic systems based on stochastic optimization principles," *International Journal of Robotics Research*, vol. 30, no. 5, pp. 556–573, 2011.
[27] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *International Journal of Robotics Research*, vol. 30, pp. 820–833, 2011.
[28] S. P. Buerger, *Stable, high-force, low-impedance robotic actuators for human-interactive machines*. PhD thesis, MIT, Department of Mechanical Engineering, 2005.
[29] L. L. Whitcomb, A. A. Rizzi, and D. E. Koditschek, "Comparative experiments with a new adaptive controller for robot arms," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 1, pp. 59–70, 1993.
[30] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line Computational Scheme for Mechanical Manipulators," *Journal of Dynamics, Systems, Measurement, and Control*, vol. 102, pp. 69–76, 1980.
[31] W. He and S. S. Ge, "Robust adaptive boundary control of a vibrating string under unknown time-varying disturbance," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 1, pp. 48–58, 2012.
[32] W. He, S. S. Ge, B. V. E. How, Y. S. Choo, and K.-S. Hong, "Robust adaptive boundary control of a flexible marine riser with vessel dynamics," *Automatica*, vol. 47, no. 4, pp. 722–732, 2011.
[33] J.-J. E. Slotine and W. Li, "On the adaptive control of robotic manipulators," *International Journal of Robotics Research*, vol. 6, no. 3, 1987.
[34] S. Arimoto, "Fundamental problems of robot control: part I, innovations in the realm of robot servo-loops," *Robotica*, vol. 13, pp. 19–27, 1995.
[35] S. Arimoto, "Fundamental problems of robot control: part II, nonlinear circuit theory towards an understanding of dexterous motions," *Robotica*, vol. 13, pp. 111–122, 1995.
[36] S. S. Ge, C. C. Hang, and L. C. Woon, "Adaptive neural network control of robot manipulators in task space," *IEEE Transactions on Industrial Electronics*, vol. 44, no. 6, pp. 746–752, 1997.
[37] C. Kwan, F. L. Lewis, and D. M. Dawson, "Robust neural-network control of rigid-link electrically driven robots," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 581–588, 1998.
[38] S. Lin and A. A. Goldenberg, "Neural-network control of mobile manipulators," *IEEE Transactions on Neural Networks*, vol. 12, no. 5, pp. 1121–1133, 2001.
[39] A. Tayebi, "Adaptive iterative learning control for robot manipulators," *Automatica*, vol. 40, no. 7, pp. 1195–1203, 2004.
[40] K. Dupree, C. H. Liang, G. Hu, and W. E. Dixon, "Adaptive Lyapunov-based control of a robot and mass-spring system undergoing an impact collision," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 38, no. 4, pp. 1050–1061, 2008.
[41] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.
[42] J. E. Colgate and N. Hogan, "Robust control of dynamically interacting systems," *International Journal of Control*, vol. 48, no. 1, pp. 65–88, 1988.
[43] R. Johansson and M. W. Spong, "Quadratic optimization of impedance control," *Proceedings of IEEE International Conference of Robotics and Automation*, vol. 1, pp. 616–621, 1994.
[44] J. Kober and J. Peters, "Policy search for motor primitives in robotics," *Machine Learning*, vol. 84, pp. 171–203, 2011.
[45] P. I. Corke, "A robotics toolbox for MATLAB," *IEEE Robotics and Automation Magazine*, vol. 3, no. 1, pp. 24–32, 1996.
[46] S. S. Ge, J. J. Cabibihan, Z. Zhang, Y. Li, C. Meng, H. He, M. R. Safizadeh, Y. B. Li, and J. Yang, "Design and development of nancy, a social robot," *Proceedings of the 8th International Conference on Ubiquitous Robots and Ambient Intelligence*, pp. 568–573, November 2011.