Estimation of human impedance and motion intention for constrained human-robot interaction

Article  (Accepted Version)


This version is available from Sussex Research Online: http://sro.sussex.ac.uk/id/eprint/85651/

This document is made available in accordance with publisher policies and may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher’s version. Please see the URL above for details on accessing the published version.

Copyright and reuse:
Sussex Research Online is a digital repository of the research output of the University.

Copyright and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable, the material made available in SRO has been checked for eligibility before being made available.

Copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.
Estimation of Human Impedance and Motion Intention for Constrained Human-Robot Interaction

Xinbo Yu\textsuperscript{1,2,3}, Yanan Li\textsuperscript{4}, Shuang Zhang\textsuperscript{1,2,3*}, Chengqian Xue\textsuperscript{1,2,3}, Yu Wang\textsuperscript{5}

1. School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China.
2. Institute of Artificial Intelligence, University of Science and Technology Beijing, Beijing 100083, China.
3. Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, University of Science and Technology Beijing, Beijing 100083, China.
4. Department of Engineering and Design, University of Sussex, Brighton BN1 9RH, UK.
5. Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Abstract—In this paper, a complete framework for safe and efficient physical human-robot interaction (pHRI) is developed for robot by considering both issues of adaptation to the human partner and ensuring the motion constraints during the interaction. We consider the robot’s learning of not only human motion intention, but also the human impedance. We employ radial basis function neural networks (RBFNNs) to estimate human motion intention in real time, and least square method is utilized in robot learning of human impedance. When robot has learned the impedance information about human, it can adjust its desired impedance parameters by a simple tuning law for operative compliance. An adaptive impedance control integrated with RBFNNs and full-state constraints is also proposed in our work. We employ RBFNNs to compensate for uncertainties in the dynamics model of robot and barrier Lyapunov functions are chosen to ensure that full-state constraints are not violated in pHRI. Results in simulations and experiments show the better performance of our proposed framework compared with traditional methods.

Index Terms — Human motion intention estimation, impedance learning, adaptive neural network control, full-state constraints, barrier Lyapunov functions.

I. INTRODUCTION

Robots are coming to our daily lives driven by social needs and development of robotics, intelligent control and machine learning [1]. Relying on complementary advantages of humans’ perception and

*Corresponding author is Shuang Zhang. E-mail: zhangshuang.ac@gmail.com
robots' execution capabilities, various tasks can be accomplished by cooperative efforts of human and robot. Physical human-robot interaction (pHRI) occurs in human robot collaborative tasks such as neuro-rehabilitation, object transportation [2] and so on.

In human robot collaborative tasks, it is essential that robots acquire force information by force or torque sensor and respond to humans in a proper way [3]. Force control is a well-developed interaction control strategy but limited by poor robustness against disturbances [4]. Hybrid position/force control is widely utilized in pHRI, but interaction forces are deemed as disturbances [5]. In [6], a compliance selection vector is employed in proposed controller to determine whether the system is under position or force control. However, the controller makes the interaction unstable when robot interacts with a stiff environment, and brings sluggish response in interaction with a soft environment. Due to robustness and feasibility, impedance control is widely used to relate interactive force with deviations from desired positions [7]. In [8], a position-based impedance control is developed where there are two control loops. In this controller, the output of outer loop is a virtual desired trajectory, and the control objective of inner loop is to track the virtual desired trajectory, so if tracking errors converge to zero, the robot will perform a desired impedance.

How a robot detects what the human is trying to do poses many challenges in HRI. If robots have no knowledge of human motion intention, they may become additional loads for human. Conversely, if robots know human motion intention, robots can be initiative to move actively, and human will cost less effort to guide robots. Therefore, how to estimate motion intention of human partners attracts substantial attention of researchers. In [9], human motion intention is estimated and it enables a robot to follow human for fast point-to-point tasks. Without force sensors, changes in control effort are utilized to obtain the estimation of human motion intention in [10]. In [11], intentional reaching direction is defined for describing the human's upper limb motion intention in real time in exoskeleton. In [12], a walk intention estimation method is proposed for an omnidirectional cane robot. Human motion intention has different definitions in literatures. In [13], human motion intention is defined as human's current position which is provided with the feedback by a multi-modal interface. In [14], human motion intention is defined as target position or time-varying desired trajectory which is estimated by online neural networks (NNs) based on available sensory information.

How to make robot adjust its desired impedance according to the environments' impedance [15] or human's impedance [16] is also a key issue in pHRI. In [17], authors propose an impedance learning method for a robot to interact with an unknown environment to avoid large interaction forces. Some
learning methods have been utilized in literatures [18]. Reinforcement learning, which adopts an actor-critic algorithm, is utilized in [19] to acquire optimal impedance parameters of robot for contact tasks. NNs are also trained in [20] by using an iterative method to regulate stiffness and viscosity parameters. Probabilistic methods are also employed in impedance learning. In [21], a human arm impedance estimation method is proposed for a 2-DOF assembly robot subject to nonlinear frictions. In [22], virtual stiffness can be estimated by the weight least-squares estimation, and it is included in the complete set of task-parameterized Gaussian mixture model. To our best knowledge, there are no works that combine human motion intention estimation and impedance learning.

Various control strategies have been developed to address uncertainties in dynamics, such as adaptive control strategies, optimal control [23], [24], fault-tolerant control [25], boundary control [26] and finite-time tracking control [27]. Radial basis function neural networks (RBFNNs) are used to compensate for unknown dynamic uncertainties in [28], and also utilized in bimanual dual arm robots [29], underactuated wheeled robots [30], flapping wing aerial vehicles [31], underwater robots [32], biped robots [33], flexible robots [34], gantry cranes [35], marine surface vessels [36], soft robots [37] and autonomous underwater vehicles [38]. Disturbance observer is also designed combined with NNs to estimate uncertain disturbances in literatures [39]. In [40], adaptive NN controllers integrating nonlinear disturbance observer are proposed in designing a human upper arm exoskeleton. In [39], a disturbance observer is proposed in robust tracking control for self-balancing mobile robots. State observer is also designed to obtain unknown system states for uncertain systems [41] and unmodeled nonlinear systems [42]. In our work, we utilize RBFNN to compensate for uncertainties in dynamic model of robot.

Furthermore, safety is extremely important in situations where human directly interacts with robots [43]. In this sense, constrained robots have drawn much attention of researchers [44], such as input constraint [45], [46] and output constraint [47]. Using barrier Lyapunov functions (BLFs) is an effective method to make robots take into account motion constraints, such as position constraints and velocity constraints in joint space or Cartesian space. In [48], output constraints based on BLFs are introduced into controller for nonlinear systems and constraints are guaranteed not to be transgressed. In [49], output constraints are considered to be time-varying, and constraint satisfaction is ensured by constructing proper BLFs. In [50], log-type BLFs are designed to avoid the violation of output constraints with full-state feedback control. In [51], force/motion control is designed for a mobile robotic manipulator with uncertain holonomic constraints. In our paper, full-state constraints based on log-type BLFs are considered for constraining position and velocity in human robot collaborative tasks.
Based on above discussions, to our best knowledge no works propose integrated methods considering human motion intention estimation and impedance learning. Therefore, in this paper we consider robot’s learning of not only human motion intention, but also human impedance. An adaptive impedance control integrated with NNs and motion constraints is proposed for robots collaborating with human to perform human-robot collaborative tasks. The main contributions of our work include:

1) Different from existing works in the field of pHRI, we estimate both impedance parameters and motion intention of human partner, so that the robot can collaborate with human partner to perform tasks actively.

2) We propose NNs to compensate for uncertainties in dynamic model of robot which improves tracking accuracy. Impedance controller design and stability analysis are conducted rigorously.

3) Full-state constraints are considered in the controller design to avoid collision and violation of speed limit. In this sense, a complete framework for safe and efficient human-robot interaction is developed for robot by considering both issues of adaptation to the human partner and ensuring the motion constraints during the interaction.

Our work is structured as follows: in Section II, the dynamic models of robot and human are analysed, and the control objective is introduced; in Section III, we analyse the human motion intention estimation method based on RBFNNs, while impedance learning is developed afterwards, and then an adaptive position-based impedance control strategy is considered subject to full-state constraints, with RBFNNs also utilized to compensate for uncertainties in robot’s dynamic model to improve tracking accuracy when robot tracks the virtual desired trajectory in the inner control loop; in Section IV, the simulation results show the effectiveness of our proposed framework; in Section V, an experiment on a Baxter robot is designed to evaluate the performance of our controller; in Section VI, conclusion of the paper is given.

II. System Description

We consider a human robot collaborative task as shown in Fig. 1, where a human partner collaborates with robot to carry out an object transporting task. In this task, compliant cooperation should be ensured for reducing human effort. Position and velocity of robot should be restricted for safe cooperation. In this section, we firstly describe the dynamic model of both robot and human.

A. Dynamic Model

I. Dynamic Model of Robot
The dynamic model of robot in the task space is described as follows:

$$M_t(x)\ddot{x} + C_t(x, \dot{x})\dot{x} + G_t(x) = u + f,$$  \hspace{1cm} (1)

where $x \in \mathbb{R}^n$ represents position vector, $\dot{x} \in \mathbb{R}^n$ represents velocity vector, $\ddot{x} \in \mathbb{R}^n$ represents acceleration vector. $M_t(x) \in \mathbb{R}^{n \times n}$, $C_t(x, \dot{x}) \in \mathbb{R}^n$ and $G_t(x) \in \mathbb{R}^n$ denote the inertia matrix, Coriolis and centrifugal force vector and gravitational force vector, respectively. $u \in \mathbb{R}^n$ and $f \in \mathbb{R}^n$ denote control force vector of robot and interaction force vector between human and robot, respectively. The dynamic model of robot in joint space is given in Appendix A.

II. Dynamic Model of Human

In object transporting tasks, interaction force is exerted on the robot by human arm. Therefore, it is essential to study the dynamic model of the human arm. In this paper, we suppose that the dynamics of human arm are described by the following damping-stiffness model:

$$f = -D_h\ddot{x} + K_h(x_d - x),$$  \hspace{1cm} (2)
where \( D_h \) and \( K_h \) denote damper and stiffness matrices of human, respectively. \( x_d \) denotes the human motion intention. In our paper, human impedance \( D_h \) and \( K_h \) can be time-varying or constant, and \( x_d \) is time-varying.

**Remark 1:** The human’s dynamic model (2) is a simplified model, as discussed and verified in [52]: the damper and stiffness matrices usually dominate the model of human arm, so the mass matrix is ignored.

**B. Control Objective**

Position-based impedance control structure is employed in our work. Firstly, we consider a target impedance model for robot as follows:

\[
f = D_d(\ddot{x}_d - \ddot{x}) + K_d(\dot{x}_d - x),
\]

where \( D_d \) and \( K_d \) denote the desired damper and stiffness matrices respectively, \( x \) is the virtual desired trajectory, and \( \ddot{x}_d \) denotes the estimated human motion intention. The control structure is proposed in Fig. 2. Seen from Fig. 2, when tracking control design is effective to ensure \( x \) tracking \( x_r \), we rewrite the impedance model (3) as follows:

\[
f = D_d(\ddot{x}_d - \ddot{x}) + K_d(\dot{x}_d - x) .
\]

As discussed before, human is leading the task and he/she has the knowledge of the target position or desired trajectory, and the robot will follow the human motion. If the robot does not know the human motion intention, i.e., \( x_d \) may be far away from \( x \) in (2), human will cost more effort to operate the robot due to a significant interaction force \( f \), but if the robot knows the human motion intention \( x_d \) and changes \( \ddot{x}_d \) accordingly, the human partner will move the robot easily because the interaction force in (2) is less. Therefore, firstly we will propose an estimation method to obtain the human motion intention.

When human plays an active role to move robot in pHRI, it would be desirable that the robot is able to reduce its desired impedance parameters in (4) to make it compliant. Conversely, if human is not willing

---

**Fig. 2: Control architecture.**
to lead tasks, robot should increase the impedance parameters to ensure the positioning accuracy subject to external disturbances. Inspired by this idea, we suppose that impedance matrices $D_d, K_d, D_h, K_h$ are diagonal, and design an impedance tuning rule for robot:

$$\hat{D}_h + D_d = \bar{D}$$

$$\hat{K}_h + K_d = \bar{K},$$

(5)

where $\bar{D}, \bar{K} \in \mathbb{R}^{n \times n}$ are given positive diagonal matrices, $\hat{D}_h$ and $\hat{K}_h$ are the estimated matrices of $D_h$ and $K_h$, respectively. According to the proposed tuning rule (5), robot can adjust its desired impedance parameters to adapt to different humans' impedance. Therefore, we will propose an identification method to obtain the human impedance parameters.

For ensuring the operational safety, robot should follow human in a limited area and with a constrained speed. To address this issue, we will propose full-state constraints (i.e., position constraints and velocity constraints) in trajectory tracking in the Cartesian space, i.e., for $\forall t > 0$, $|x_i(t)| \leq k_{i1}, |\dot{x}_i(t)| \leq k_{i2}$. $k_1$ and $k_2$ denote predefined constrained constant vectors and $i = 1, 2, ..., n$.

Indicated by Fig. 2, four parts are considered in the controller design. First, in order to decrease the interaction force $f$, we propose a method to obtain the estimated value $\hat{x}_d$ of human motion intention. Second, in the inner trajectory tracking control loop, if $x$ tracks the virtual desired trajectory $x_r$ which is acquired by the output of the outer loop, the robot would track the target impedance model, and NNs are utilized to address uncertainties in dynamic model of robot. Third, we propose an impedance learning method to acquire the human arm impedance and design a tuning rule to adjust desired impedance parameters of robot for efficient collaboration. Eventually, full-state constraints are considered in the trajectory tracking control to ensure safety of human–robot interaction.

III. CONTROL DESIGN

A. Human Motion Intention Estimation

Suppose that $D_h$ and $K_h$ are unknown functions of system state variables, we cannot obtain $\hat{x}_d$ by (2) directly. We employ the method in [14] to estimate $x_d$ according to $x, \dot{x}$ and $f$ as follows:

$$\hat{x}_d = Y(f, x, \dot{x}).$$

(6)

$Y(\cdot)$ is an unknown function and may be nonlinear, so an effective estimation method should be used to approximate $Y(\cdot)$. As an important online learning NN method, RBFNN can be used to estimate $x_d$ and
an adaptive method is proposed to estimate the actual weights of RBFNN. A typical RBFNN structure is described as follows

\[ \hat{x}_{di} = \hat{\Theta}_i^T S_i(\eta_i) + \varepsilon_i, \]  

(7)

where the input of RBFNN is \( \eta_i=[x_i^T, x_i^T, x_i^T] \), \( S_i(\cdot) \) is the radial basis function, \( \hat{\Theta}_i \) and \( \varepsilon_i \) denote the estimated weight and the estimation error, respectively. According to (2), interaction force \( f \) will become small if \( x \) gets close to \( x_d \). Therefore, \( \hat{\Theta}_i \) is adjusted based on the steepest descent method with respect to the cost function

\[ E_M = \frac{1}{2} f_i^2, \]  

(8)

where \( f_i \) is the interaction force in one direction. Then we propose the adaptation law as follows

\[
\dot{\hat{\Theta}}_i = -\beta_i \frac{\partial E_M}{\partial \hat{\Theta}_i} \\
= -\beta_i \frac{\partial E_M}{\partial f_i} \frac{\partial f_i}{\partial \hat{x}_{di}} \frac{\partial \hat{x}_{di}}{\partial \hat{\Theta}_i} \\
= -\beta_i f_i K_{hi} S_i(\eta_i) \\
= -\gamma f_i S_i(\eta_i),
\]

(9)

where \( \beta_i \) is a positive scalar, and \( \beta_i \) and \( K_{hi} \) are absorbed by \( \gamma \) (the detailed derivation seen in Appendix B). So \( \hat{\Theta}_i \) can be designed as

\[ \hat{\Theta}_i = \hat{\Theta}_i(0) - \gamma \int_0^t [f_i(\omega) S(\eta_i(\omega))] d\omega. \]  

(10)

Then human motion intention estimation \( \hat{x}_d \) can be acquired based on (7) and (10).

**B. Human Impedance Learning**

If we obtain the estimates \( \hat{D}_h \) and \( \hat{K}_h \) in dynamic model of human (2), we can adjust impedance parameter matrices of robot in (4) based on the tuning rule (5) for robot adapting to different interactive situations. In this section, we propose a human impedance learning method for obtaining \( \hat{D}_h \) and \( \hat{K}_h \). If we have the estimates of \( D_h, K_h \) and \( x_d \), the estimated interaction force \( \hat{f} \) can be written as

\[ \hat{f} = -\hat{D}_h \hat{x} + \hat{K}_h \]  

(11)
where $z = x - \hat{x}_d$. As $\hat{x}_d$ can be obtained in Section 3.1, $z$ and $\hat{x}$ are known in (11), and actual interaction force $f$ is measurable, we apply parameter estimation method to obtain the unknown matrices. Least square (LS) method can be employed in system parameter estimation, based on which we consider a cost function

$$E_{li} = \sum_{j=1}^{N} (f_{ij} - \hat{f}_{ij})^2,$$

where $j$ denotes the sampling number. In particular, we want to make partial derivatives of $E_{li}$ with respect to $D_h$ and $K_h$ zero:

$$\frac{\partial E_{li}}{\partial D_h} = 0, \frac{\partial E_{li}}{\partial K_h} = 0.$$  

In practical human-robot interactive processes, $D_h$ and $K_h$ can be time-varying. Based on the moving average method, we can get the latest sampling region for improving the estimation accuracy of the time-varying parameters. We obtain the estimated parameters $\hat{D}_h(t)$ and $\hat{K}_h(t)$ as below

$$\begin{bmatrix} \hat{D}_h(t) \\ \hat{K}_h(t) \end{bmatrix} = \begin{bmatrix} -\sum_{j=1}^{s} \hat{x}_{ij}^2 & \sum_{j=1}^{s} \hat{x}_{ij}\hat{z}_{ij} \\ -\sum_{j=1}^{s} \hat{z}_{ij}\hat{x}_{ij} & \sum_{j=1}^{s} \hat{z}_{ij}^2 \end{bmatrix}^{-1} \times \begin{bmatrix} \sum_{j=1}^{s} \hat{x}_{ij}\hat{f}_{ij} \\ \sum_{j=1}^{s} \hat{z}_{ij}\hat{f}_{ij} \end{bmatrix} \quad (s \leq S),$$

$$= \begin{bmatrix} -\sum_{j=s+1}^{s+T} \hat{x}_{ij}^2 & \sum_{j=s+1}^{s+T} \hat{x}_{ij}\hat{z}_{ij} \\ -\sum_{j=s+1}^{s+T} \hat{z}_{ij}\hat{x}_{ij} & \sum_{j=s+1}^{s+T} \hat{z}_{ij}^2 \end{bmatrix}^{-1} \times \begin{bmatrix} \sum_{j=s+1}^{s+T} \hat{x}_{ij}\hat{f}_{ij} \\ \sum_{j=s+1}^{s+T} \hat{z}_{ij}\hat{f}_{ij} \end{bmatrix} \quad (s > S),$$

where $s$ denotes the sampling number at time $t$ and $S$ denotes the sampling interval in a sampling time period $T$. According to the proposed method, the estimation of time-varying $\hat{D}_h(t)$ and $\hat{K}_h(t)$ can be obtained (the detailed derivation seen in Appendix B). To ensure symmetric positive definite (SPD) for $\hat{D}_h$ and $\hat{K}_h$, we adopt the method in [53] to compute $\hat{D}_h^T$ and $\hat{K}_h^T$ as the SPD matrices nearest to $\hat{D}_h$ and $\hat{K}_h$ according to the Frobenius norm as follows:

$$\hat{D}_h^T = \frac{A + P}{2}, A = \frac{\hat{D}_h + \hat{D}_h^T}{2},$$

$$\hat{K}_h^T = \frac{B + Y}{2}, B = \frac{\hat{K}_h + \hat{K}_h^T}{2}.$$  

where $P$ and $Y$ denote the symmetric polar factor which can be found from the singular value decomposition of $A$ and $B$, respectively, with $A = LP$, $B = RY$, $L^T L = I$, $R^T R = I$. Therefore, we can obtain desired impedance parameter matrices of robot $D_d$ and $K_d$ according to the tuning rule (5).
C. Tracking Control with Full-state Constraints

In previous sections, \( D_d \), \( K_d \) and \( \dot{x}_d \) are obtained, so we can calculate virtual desired reference trajectory \( x_r \) according to (3). If \( x \) tracks \( x_r \), the robot would track the target impedance model. For ensuring the operational safety, the robot should follow the human arm in a constrained area and with a constrained speed. BLF-based method is used in tracking control for avoiding constraint violation. In the first part of this section, a model-based control is developed considering position and velocity constraints.

1. Model-based (MB) Control

Suppose that the dynamic model of robot is known for the designer, i.e., \( M_i(x) \), \( G_i(x, \dot{x}) \) and \( G_i(x) \) are known. For convenience of analysis, denoting \( x_1 = x \), \( x_2 = \dot{x} \), the dynamic model of robot in state-space form is described as follows:

\[
\dot{x}_1 = x_2 \\
\dot{x}_2 = M_i(x_1)^{-1}(u - f - C_i(x_1, x_2)x_2 - G_i(x_1)) \\
y = x_1.
\]  

(16)

Then we define the tracking error as

\[
e_1 = x_1 - x_r, e_2 = x_2 - \alpha_1,
\]  

(17)

where \( \alpha_1 \) denotes a virtual stabilization variable to be defined later. Choose a log-type BLF candidate \( V_1 \) as follows:

\[
V_1 = \frac{1}{2} \sum_{i=1}^{n} \ln \frac{k_{ai}^2}{k_{ai}^2 - e_{1i}^2},
\]  

(18)

where \( k_a \) and \( k_b \) denote tracking error constraints, where \( k_a = k_1 - \tilde{k}_1, k_b = k_2 - \tilde{k}_2, k_a = [k_{a1}, k_{a2}, ..., k_{an}], k_b = [k_{b1}, k_{b2}, ..., k_{bn}] \), \( \tilde{k}_1 \) and \( \tilde{k}_2 \) denote vectors composed of the maximum absolute values of \( x_{ri} \) and \( \alpha_{1i} \), respectively. They can be described as \( x_{ri} \leq \tilde{k}_1 \), \( \alpha_{1i} \leq \tilde{k}_2 \). Differentiating \( V_1 \) with respect to time, we obtain

\[
\dot{V}_1 = \sum_{i=1}^{n} \frac{e_{1i} \dot{e}_{1i}}{k_{ai}^2 - e_{1i}^2}.
\]  

(19)

Then differentiating \( e_1 \) with respect to time, we obtain

\[
\dot{e}_1 = e_2 + \alpha_1 - \dot{x}_r.
\]  

(20)
We define $\alpha_1$ in (17) as follows:

$$\alpha_1 = \dot{x}_r - A,$$  \hfill (21)

where

$$A = \begin{bmatrix}
(k_{11}^2 - e_{11}^2)g_{11}e_{11} & (k_{12}^2 - e_{12}^2)g_{12}e_{12} \\
(k_{21}^2 - e_{12}^2)g_{21}e_{21} & \cdots \\
(k_{n1}^2 - e_{1n}^2)g_{1n}e_{1n} & 
\end{bmatrix},$$  \hfill (22)

where $g_{1i}$ is gain parameter. Substituting (20), (21) and (22) into (19), we have

$$V_1 = -\sum_{i=1}^n g_{1i}e_{1i}^2 + \sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai} - e_{1i}^2}.$$  \hfill (23)

Then we construct a BLF candidate $V_2$ as follows:

$$V_2 = V_1 + \frac{1}{2}e_2^2M_r(x_1)e_2 + \frac{1}{2}\sum_{i=1}^n \ln \frac{k_{bi}^2}{k_{bi}^2 - e_{2i}^2}.$$  \hfill (24)

Then differentiating $V_2$ with respect to time, we obtain

$$V_2 = -\sum_{i=1}^n g_{1i}e_{1i}^2 + \sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai} - e_{1i}^2} + \sum_{i=1}^n \frac{e_{2i}e_{2i}}{k_{bi}^2 - e_{2i}^2} + e_2^T[u_m - f - G_r(x_1, x_2)\alpha_1 - G_r(x_1) - M_r(x_1)\dot{\alpha}_1].$$  \hfill (25)

Differentiating $e_2$ with respect to time, we can obtain

$$\dot{e}_2 = x_2 - \alpha_1$$
$$= M_r(x_1)^{-1}(u_m - f - G_r(x_1, x_2)x_2 - G_r(x_1)) - \alpha_1.$$  \hfill (26)

Substituting (26) to (25), we can design the MB control input as follows:

$$u_m = -G_r e_2 + G_r(x_1, x_2)\alpha_1 + G_r(x_1) + M_r(x_1)\alpha_1 + f$$
$$-\left(e_2^T\right)^{+}\sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai}^2 - e_{1i}^2} - \left(e_2^T\right)^{+}\sum_{i=1}^n \frac{e_{2i}(a_i - \alpha_{1i})}{k_{bi}^2 - e_{2i}^2}$$  \hfill (27)
where \( (e_2^T)^\dagger \) denotes the Moore-Penrose inverse of \( e_2^T \). \( G_2 \) denotes gain matrix. We can obtain the following relationship according to the property of Moore-Penrose inverse:

\[
(e_2^T)^\dagger e_2^T = \begin{cases} 
0, & e_2 = [0,0,\ldots,0]^T \\
1, & \text{otherwise}.
\end{cases}
\] (28)

Under the control input \( u_m \) in (27), \( V_2 \) satisfies the following condition:

\[
V_2 = -\sum_{i=1}^{n} k_{ii} e_{1i}^2 - e_2^T G_2 e_2 < 0.
\] (29)

So we can conclude that tracking errors \( e_1 \) and \( e_2 \) remain in the interval \( \forall t > 0, -k_{ai} \leq e_{1i} \leq k_{ai}, -k_{bi} \leq e_{2i} \leq k_{bi}, \) and the states \( x_1 \) and \( x_2 \) remain in the interval \( \forall t > 0, |x_{1i}(t)| \leq k_{1i}, |x_{2i}(t)| \leq k_{2i}. \)

Remark 2: We consider a special situation when \( e_2 = 0 \), which leads to \( V_2 = -\sum_{i=1}^{n} k_{ii} e_{1i}^2 \leq 0 \) in (29). The asymptotic stability of the system can be drawn by the Barbalat lemma. In this paper, we consider \( e_2 \neq 0 \) and design controllers (27) and (32).

II. Adaptive RBFNN Control with Full-state Feedback

To address uncertainties in the dynamic model of robot, i.e., \( M(x), C_i(x,\dot{x}) \) and \( G_i(x) \) are unknown, an adaptive NN control design is proposed. We design NN adaptive law as follows:

\[
\dot{\hat{W}}_i = -\Gamma_i [S_i(Z_i) e_{2i} + \sigma_i \hat{W}_i], i = 1,2,\ldots,n
\] (30)

where \( \Gamma_i = \Gamma_i^T \) denotes positive gain matrix, \( \hat{W}_i \) denotes the weight estimate of NN and \( \sigma_i \) is a small positive constant for improving system robustness. The input of NN is \( Z_i = [x_1^T, x_2^T, \alpha^T, \alpha_i^T], \) and \( \hat{W}_i S(Z) \) is utilized to estimate \( W_i^\ast T S(Z) \):

\[
W_i^\ast T S(Z) = G_i(x_1, x_2) \alpha_1 + G_i(x_1) + \alpha_1 M_i(x_1) - \varepsilon(Z),
\] (31)

where \( W_i^\ast \) denotes actual NN weight, \( \varepsilon(Z) \) denotes estimation error which is in bounds over the compact set \( \Omega, \forall Z \in \Omega, ||\varepsilon(Z)|| < \bar{\varepsilon}(\bar{\varepsilon} > 0). \) We design NN control input \( u \) as follows:

\[
u = -G_3 e_2 + \hat{W}_i S(Z) + f - (e_2^T)^\dagger \sum_{i=1}^{n} \frac{e_{1i} e_{2i}}{k_{ai}^2 - e_{2i}^2} e_1^2 - (e_2^T)^\dagger \sum_{i=1}^{n} \frac{g_1 e_{1i}^2}{k_{bi}^2 - e_{2i}^2} + (e_2^T)^\dagger \sum_{i=1}^{n} \frac{g_2 e_{2i}^2}{k_{bi}^2 - e_{2i}^2},
\] (32)
where \( g_{2i} \) and \( G \) denote the gain parameter and positive definite gain matrix, respectively. Then we construct another BLF functions \( V_3 \) as

\[
V_3 = V_2 + \frac{1}{2} \sum_{i=1}^{n} \hat{W}_i^T \Gamma_i^{-1} \hat{W}_i,
\]

(33)

where the weight error \( \hat{W}_i = W_i^* - \hat{W}_i \). Then we differentiate \( V_3 \) as

\[
\dot{V}_3 = -\sum_{i=1}^{n} g_{1i} \dot{e}_{1i}^2 + \sum_{i=1}^{n} \frac{e_{1i} e_{2i}}{k_{ai}^2 - e_{1i}^2} + \sum_{i=1}^{n} \frac{e_{2i} e_{2i}}{k_{bi}^2 - e_{2i}^2}
+ e_2^T [u - f(t) - C_i(x_1, x_2) \alpha_1 - G_i(x_1) - M_i(x_1) \alpha_1].
\]

(34)

We have

\[
\dot{V}_3 \leq -\sum_{i=1}^{n} g_{1i} \dot{e}_{1i}^2 - e_2^T G_2 e_2 - \sum_{i=1}^{n} \frac{g_{1i} e_{1i}^2}{k_{ai}^2 - e_{1i}^2} - e_2^T \mathcal{E}(Z)
- \sum_{i=1}^{n} \frac{g_{2i} e_{2i}^2}{k_{bi}^2 - e_{2i}^2} + e_2^T \hat{W}_T S(Z) - e_2^T W^* T S(Z)
+ \sum_{i=1}^{n} \hat{W}_i^T \Gamma_i^{-1} \{-\Gamma_i [S_i(Z) e_{2i} + \sigma_i \hat{W}_i]\}
\leq -e_2^T (G_2 - I) e_2 - \sum_{i=1}^{n} \frac{g_{1i} e_{1i}^2}{k_{ai}^2 - e_{1i}^2} - \sum_{i=1}^{n} \frac{g_{2i} e_{2i}^2}{k_{bi}^2 - e_{2i}^2}
+ \frac{1}{2} || \dot{\mathcal{E}}(Z) ||^2 + \frac{\sigma_i}{2} (|| W_i^* ||^2 - || \hat{W}_i ||^2)
\leq -\rho V_3 + C,
\]

(35)

where

\[
\rho = \min(\min(2g_{1i}), \min(2g_{2i}), \frac{2\lambda_{\min}(G_2 - I)}{\lambda_{\max}(M_i(x))}, \min(\frac{\sigma_i}{\Gamma_i^{-1}}))
C = \frac{1}{2} || \mathcal{E} ||^2 + \frac{\sigma_i}{2} || W_i^* ||^2.
\]

(36)

where \( \lambda_{\min} \) and \( \lambda_{\max} \) denote the minimum and maximum eigenvalues of a matrix.

**Theorem 1:** For initial conditions \( |x_{1i}(0)| \leq k_{1i}, |x_{2i}(0)| \leq k_{2i} \), control law (32) ensures that all error signals are semi-globally uniformly bounded (SGUB) and position and velocity constraints are not violated, i.e., \( \forall t > 0, |x_{1i}(t)| \leq k_{1i}, |x_{2i}(t)| \leq k_{2i} \). The closed-loop error signals \( e_1, e_2 \) and \( \hat{W} \) remain in compact sets.
\( \Omega_{e_1}, \Omega_{e_2}, \Omega_{W}, \) respectively:

\[
\Omega_{e_1} = \{ e_1 \in \mathbb{R}^n \mid \|e_1\| \leq \sqrt{k_w^2(1 - e^{-Q}), i = 1, 2, 3, \ldots, n} \}
\]

\[
\Omega_{e_2} = \{ e_2 \in \mathbb{R}^n \mid \|e_2\| \leq \sqrt{\frac{Q}{\lambda_{\min}(M_f(x))}} \} \cap \{ e_2 \in \mathbb{R}^n \mid \|e_2\| \leq \sqrt{k_w^2(1 - e^{-Q}), i = 1, 2, 3, \ldots, n} \}
\]

\[
\Omega_W = \{ W \in \mathbb{R}^{q \times n} \mid \|W\| \leq \sqrt{\frac{Q}{\lambda_{\min}(\Gamma^{-1})}} \},
\]

where \( Q = V_3(0) + C/\rho \) with positive constants \( C \) and \( \rho \) given in (36).

### IV. Simulation

In simulations, two-link revolute joint robot shown in Fig. 1(a) is in interaction with human, and interaction force generated by human is applied on the handle near the end-effector of robot.

#### A. Simulation Results about Human Impedance Learning

In this part, we consider unknown fixed and time-varying human impedance in two cases, and parameter matrices \( \mathcal{D} \) and \( \mathcal{R} \) in impedance tuning rule (5) are designed as \( \mathcal{D} = \text{diag}[3, 3] \) and \( \mathcal{R} = \text{diag}[3, 3] \). We consider fixed impedance \( D_h = \text{diag}[1, 1], K_h = \text{diag}[2, 2] \) in the first case, and time-varying impedance \( D_h = \text{diag}[1 + 0.2\sin(t), 1 + 0.2\sin(t)] \) in the second case. We utilize least square method (LS) combined with moving average algorithm in (14) to obtain estimated impedance matrices \( \hat{D}_h \) and \( \hat{K}_h \), and SPD can be ensured based on (15). (5) is employed to tune the robot's desired impedance online based on estimated impedance. In Fig. 3(a) and Fig. 3(b), we can find that fixed or time-varying damper parameters of human can be estimated and we can obtain desired damper parameters of robot. In a similar way, stiffness parameters by our proposed method can be obtained.

#### B. Simulation Results about Human Intention Estimation

We employ our proposed RBFNN method to estimate the human motion intention \( x_d \) in (7). Adaptation law (9) is designed to adjust \( \hat{\Theta}_i \) in (10). Human motion intention estimation is calculated based on (7). We choose RBFNN centers in the region of \([-1, 1]\], number of nodes in the RBFNN as \( 2^6 \), and we define the initial value of the RBFNN weights \( \Theta_i \) as 0. Indicated from Fig. 4(a), we set different initial human motion intention \( x_{d1}(0) \) as 1.10m, 1.65m and 1.80m, respectively. Under our human motion intention
estimation method, interaction forces are all below 2N in pHRI. Fig. 4(b) shows that errors between \( x_{d1} \) and \( \hat{x}_{d1} \) in three different conditions all converge to zero, where \( x_{d1} \) and \( \hat{x}_{d1} \) denote human motion intention and its estimate in the x-axis in the task space, respectively.

\[
\begin{align*}
\text{interaction force} & = x_{d1} - \hat{x}_{d1} \\
\text{human motion intention} & = x_{d1}
\end{align*}
\]

Fig. 4: IV. B-human interaction force and motion intention estimation.

C. Simulation Results about Tracking Control and Full-state Constraints

In this part, we evaluate our proposed method compared with the situation without constraints in controller design. For design of our proposed method with constraints in (32), the tracking error constraints are set as \( k_{a1} = 0.4 \text{m}, k_{a2} = 0.4 \text{m}, k_{b1} = 0.4 \text{m/s}, k_{b2} = 0.4 \text{m/s} \) and full-state constraint vectors \( k_1 = [1.5 \text{m}; 1.5 \text{m}] \).
$k_2 = [1.5\text{m/s}; 1.5\text{m/s}]$; gain parameters in (32) $g_{11} = 2, g_{12} = 2, g_{21} = 10, g_{22} = 10$. $G_3 = \text{diag}[10, 10]$. Fig. 5(a) and Fig. 5(b) illustrate the comparative position tracking results. Fig. 5(c) and Fig. 5(d) illustrate the comparative velocity tracking results. Indicated from Fig. 5, under our proposed controller all error signals do not transgress constraints, and tracking errors converge to a small region around zero when initial states are in bounds. Compared with control design with no constraints, tracking errors with constraints are smaller. Indicated from Fig. 5(c), the velocity tracking error in x-axis is over bound obviously without constraints.

![Graphs](image)

Fig. 5: IV. C-tracking performance with and without constraints.

**D. Synthetic Simulation**

In this part, both human impedance learning and motion intention estimation in pHRI subject to constraints are considered, which are set the same as above. The robot’s initial position $x_1(0) = [0.85\text{m}, 1.05\text{m}]$, initial
velocity \( x_2(0) = [0 \text{m/s}, 0 \text{m/s}] \), and we set the robot’s virtual reference trajectory \( x_r \) as

\[
x_r = \begin{bmatrix}
0.1 \sin(t) + \cos(t) \text{m} \\
0.1 \sin(t) + \cos(t) \text{m}
\end{bmatrix}
\]

(38)

For our proposed RBFNN control, we choose gain parameters as \( g_{11} = 2 \), \( g_{12} = 2 \), \( g_{21} = 5 \), \( g_{22} = 5 \), \( K_3 = \text{diag}[10, 10] \). RBFNN centers in the region of \([-1, 1]\), the initial value of the RBFNN weight as 0, the number of nodes in the RBFNN as \( 2^8 \). \( \Gamma_1 \) and \( \Gamma_2 \) are chosen as 100 and \( \sigma_1 \) is chosen as 0.002. Fig. 6(a)

and Fig. 6(b) show that control (32) can guarantee that tracking errors converge to a small region around zero, and all error signals do not violate error constraints. In Fig. 6(c), we can see that corresponding control inputs are bounded. Indicated by Fig. 6(d), the interaction force under the traditional control design not considering motion intention estimation is larger than the force under the proposed control design considering the human motion intention estimation.

Fig. 6: IV. D-tracking performance, control input and interaction force.
V. EXPERIMENT

A. Experiment Setup

As shown in Fig. 7, Baxter robot is used in our experiment. Each arm of the Baxter bimanual robot has 7 flexible joints, i.e., joints $S_0$, $S_1$, $E_0$, $E_1$, $W_0$, $W_1$ and $W_2$, and each joint has angle and torque sensors. The resolution of angle sensors is 0.022 degree per tick, and the maximum torques are 50N/m in joints ($S_0$, $S_1$, $E_0$, $E_1$) and 15N/m in joints ($W_0$, $W_1$, $W_2$). Due to the limited computing speed of computers, two computers (Computer 1 and Computer 2) are employed for controlling the Baxter robot and calculation in this experiment. Computer 2 is employed to calculate the dynamics compensation by RBFNN in MATLAB SIMULINK and transfer the results to Computer 1 by Ethernet. Computer 1 is utilized to receive the angle, angular velocity and torque information from the Baxter robot and generate control signals to control the robot by Robot Operating System SDK (RSDK) in Ubuntu 14.04 LTS. Robot operating system (ROS) is a robot development framework that is used for integrating software libraries and tools for building robot applications.

We design an experiment where a human subject (age: 24; height: 172cm; weight: 61kg) collaborates with the right arm of Baxter robot to perform an object co-transporting task. Indicated from Fig. 1(b), the task objective is to move “orange” on the table to “cup” by human’s and robot’s collective efforts. Gripper of the robot is controlled to grasp or release the “orange” by a button near the gripper operated by human subject manually. Human subject knows the position of target “cup” through vision, and he guides the robot arm by physical interaction to grasp “orange” from the table, move towards “cup” and release it to “cup”. The snapshot of experiment is shown in Fig. 1(b). For our proposed RBFNN control in (32), the number of nodes in the RBFNN is set as $7^3$ for each $\phi_i(Z)$, and variance of centers is set as $\eta = 0.75$. $\Gamma_i$ in adaptation law is set as 500I and $\sigma_i = 0.02$. The control gain matrices $G_3=\text{diag}[17.70, 15.00, 15.70, 10.02, 10.30, 14.60, 22.00]$, and $g_{11} = 7.10, g_{12} = 22.00, g_{13} = 2.00, g_{14} = 2.75, g_{15} = 4.10, g_{16} = 2.00, g_{17} = 5.6, g_{21} = 5.1, g_{22} = 12.00, g_{23} = 1.20, g_{24} = 2.50, g_{25} = 2.10, g_{26} = 2.10, g_{27} = 4.50$.

B. Case 1: Comparative Experiments about Motion Intention Estimation

In this case, we compare our method with human motion intention estimation with the method without estimation. We employ our proposed RBFNN method to estimate human motion intention $x_d$ in (7).
Adaptation law (9) is designed to adjust $\hat{\Theta}_i$ in (10). Human motion intention estimation is calculated based on (7). We choose RBFNN centers in the region of $[-1, 1]$, nodes number in RBFNN as $2^6$, and we define the initial value of the RBFNN weights $\Theta_i$ as 0. Methods with and without human motion intention estimation are compared in the same task process shown in Fig. 1(b), which shows that they track an almost same trajectory from the same initial position to the same target position in about 20s. Interaction torques can be measured by torque sensors in robot internal platform. Seen from Fig. 8(a) and Fig. 8(b), we can find that interaction torques with human motion intention estimation are much smaller than those without estimation in some joints, such as joints $S_0$ (in 5s-14s), $S_1$ (in 15s-20s), $E_0$ (10s-13s), $E_1$ (15s-22s), $W_0$ (10s-12s), $W_1$ (15s-20s), and interaction torques without human motion intention estimation are larger than those with human motion intention estimation. Therefore, we can conclude that human makes more effort in the task when human intention estimation is not involved. We notice that torques in $W_2$ are almost zero in Fig. 8(a) and Fig. 8(b) because human subject may not use the joint in the task. The comparative results show that the robot can collaborate with human subject to perform tasks more actively using our proposed method.

C. Case 2: Comparative Experiments about Constraints

In this case, the same task is performed, with gain and NN parameters set the same as those in Case 1. The lower and upper bounds of joint errors are set as $-0.10\text{rad}$ and $0.10\text{rad}$ in all joints in this case. We compare our proposed method with methods without constraints and the tracking errors are shown in Fig. 9. Indicated from Fig. 9, we find that all tracking errors under our proposed method are within
Fig. 8: Interaction torque in each joint of Baxter robot.

Fig. 9: Tracking performance with (top) and without (bottom) constraints.
the bounds \([-0.10\text{rad}, 0.10\text{rad}]\), but errors in some joints are over bounds under the method without constraints. With these results, we can conclude that our proposed method can ensure safe interaction in the task.

VI. CONCLUSION AND FUTURE WORKS

In this paper, we proposed an adaptive NN impedance control involving full-state constraints in human robot collaborative tasks. Estimation of human impedance and motion intention were considered to improve pHRI. Safe and compliant PHRI was ensured under our proposed method. RBFNNs were employed to estimate motion intention, and least square method was utilized in human impedance learning. RBFNNs were also employed in compensating for uncertainties in robot dynamics, while BLF was chosen to ensure position and velocity constraints not violated. Simulation and experiment results were presented to verify the effectiveness of our proposed method.

The current motion intention estimation method only allows a robot to follow the human partner, which is inapplicable to situations where the robot’s autonomy is essential. This will be investigated in our future works. Impedance learning was tested in the experiments, where uncertainties due to human’s involvement may lead to problems that need to be addressed. Finally, while a simplistic scenario was studied in this paper for proof of concept, in our future works, we will focus on specific collaborative tasks such as sawing and assembly.

VII. ACKNOWLEDGEMENT

The authors would like to thank the Editor-in-Chief, the Associate Editor, and the anonymous reviewers for their constructive comments which helped to improve the quality and presentation of this paper. This work was supported by the National Natural Science Foundation of China under Grant 61873297, the China Postdoctoral Science Foundation under Grant 2018M630074 and the Fundamental Research Funds for the Central Universities under grant No. FRF-GF-18-027B.

REFERENCES


[38] Rongxin Cui, Chenguang Yang, Yang Li, and Sanjay Sharma. Adaptive neural network control of auvs with control input


VIII. APPENDIX A

We consider an $m$ degree-of-freedom (DOF) dynamic model of robot in the joint space as follows [54]

$$M(q)\ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = \tau + J^T(q)f,$$

(39)

where $q, \dot{q}, \ddot{q} \in \mathbb{R}^m$ are the joint angular displacement, velocity and acceleration vectors, respectively. $M(q) \in \mathbb{R}^{m \times m}$ is the symmetric and positive definite inertia matrix, $C(q, \dot{q}) \dot{q} \in \mathbb{R}^m$ is the vector of Coriolis and centripetal force, $G(q) \in \mathbb{R}^m$ denotes the vector of the gravitational force. $\tau \in \mathbb{R}^m$ denotes the vector of the control input torque, $f \in \mathbb{R}^n$ is the vector of the interaction force generated by the human partner, $J(q) \in \mathbb{R}^{n \times m}$ is the Jacobian matrix, where $n$ denotes the dimension in the Cartesian space. The forward kinematics of the robot is given by $x = \Phi(q)$, differentiating which with respect to time results in $\dot{x} = J(q)\dot{q}$. Based on the inverse kinematics, the velocity vector $\dot{q}$ and the acceleration vector $\ddot{q}$ in the joint space can be described as

$$\dot{q} = J^{-1}(q)\dot{x}$$

$$\ddot{q} = J^{-1}(q)\ddot{x} + J^{-1}(q)\dot{J}(q)\dot{x},$$

(40)

where $J^{-1}(q)$ denotes the pseudoinverse of $J(q)$. $M_r(x) \in \mathbb{R}^{n \times n}$, $C_r(x, \dot{x}) \dot{x} \in \mathbb{R}^n$ and $G_r(x) \in \mathbb{R}^n$ in the Cartesian space in (1) can be calculated as

$$M_r(x) = J^{-T}(q)M(q)J^{-1}(q)$$

$$C_r(x, \dot{x}) = J^{-T}(q)(C(q, \dot{q}) - M(q)J^{-1}(q)\dot{J}(q))J^{-1}(q)$$

$$G_r(x) = J^{-T}(q)G(q)$$

$$u = J^{-T}(q)\tau.$$ 

(41)

In Section IV, we define $m_r$ as the mass of link $r$, define $l_r$ as the length of link $r$, and define $l_{cr}$ as the distance from the mass center of link $r$ to joint $r-1$, and define $I_r$ as the moment of Inertia of link $r$. The simulation parameter values are chosen as: $m_1=2.0$kg, $m_2=0.85$kg, $l_1=1.40$m, $l_2=1.24$m, $l_{c1}=0.70$m, $l_{c2}=0.62$m, $I_1=0.980$kgm$^2$, $I_2=0.953$kgm$^2$.

For simulations, dynamic model parameter matrices of robot $M(q), C(q, \dot{q}), G(q)$ in the joint space in
(39) can be calculated as

\[
M(q) = \begin{bmatrix}
m_{11} & m_{12} \\
m_{13} & r(2)
\end{bmatrix}
\]

(42)

\[
C(q, \dot{q}) = \begin{bmatrix}
c_{r1} & c_{r2} \\
c_{r3} & 0
\end{bmatrix}
\]

(43)

\[
G(q) = \begin{bmatrix}
g_{r1} \\
g_{r2}
\end{bmatrix}
\]

(44)

where \( m_{11} = r(1) + r(2) + 2r(3)\cos(q(3)), \ m_{12} = r(2) + r(3)\cos(q(3)), \ m_{13} = r(2) + r(3)\cos(q(3)), \ c_{r1} = -r(3)q(4)\sin(q(3)), \ c_{r2} = -r(3)(q(2) + q(4))\sin(q(3)), \ c_{r3} = r(3)q(2)\sin(q(3)), \ g_{r1} = r(4)\cos(q(1)) + r(5)\cos(q(1) + q(3)), \ g_{r2} = r(5)\cos(q(1) + q(3)), \) where the system state variables \( q = [q(1); q(3)], \) \( q = [q(2); q(4)], \) \( q(1) \) and \( q(3) \) denote first and second joint angle, respectively, \( q(2) \) and \( q(4) \) denote first and second joint angular velocity, respectively, the variables \( r(1) = m_{11}l_{12}^2 + m_{12}l_{13}^2 + I_1, \ r(2) = m_{21}l_{12}^2 + I_2, \ r(3) = m_{21}l_{13} + m_{22}l_{14}, \ r(4) = m_{11}l_{12} + m_{22}l_{14} \) and \( r(5) = m_{22}l_{14}. \) The Jacobian matrix in (39) can be obtained according to \( l_1, l_2 \) and \( q \) as follows

\[
J = \begin{bmatrix}
J_{11} & J_{12} \\
J_{21} & J_{22}
\end{bmatrix}
\]

(45)

where \( J_{11} = -l_1\sin(q(1)) - l_2\sin(q(1) + q(3)), \ J_{12} = -l_2\sin(q(1) + q(3)), \ J_{21} = l_1\cos(q(1)) + l_2\cos(q(1) + q(3)), \ J_{22} = l_2\cos(q(1) + q(3)). \)

If we obtain \( M(q), \ C(q, \dot{q}), \ G(q) \) and \( J, \) we can calculate dynamic parameter matrices of robot in the Cartesian space \( M(x), \ C(x, \dot{x}) \) and \( G(x) \) based on (41).

**IX. APPENDIX B**

Some trivial derivations are described in this appendix. According to Eqs. (2), (7) and (8), we can obtain the partial derivative \( \frac{\partial E_M}{\partial f_i}, \frac{\partial f_i}{\partial x_{di}} \) and \( \frac{\partial x_{di}}{\partial \Theta_i} \) as follows

\[
\frac{\partial E_M}{\partial f_i} = f_i, \quad \frac{\partial f_i}{\partial x_{di}} = K_{hi}, \quad \frac{\partial x_{di}}{\partial \Theta_i} = S(h_i).
\]

(46)

Then, Eq. (9) can be obtained.
From Eq. (13) we can obtain

\[
\sum_{j=1}^{N} 2(-\dot{D}_h \dot{x}_{ij} + \dot{K}_h z_{ij} - f_{ij})(-\dot{x}_{ij}) = 0
\]

\[
\sum_{j=1}^{N} 2(-\dot{D}_h \dot{x}_{ij} + \dot{K}_h z_{ij} - f_{ij})z_{ij} = 0.
\]  

(47)

which leads to

\[-\dot{D}_h \sum_{j=1}^{N} \dot{x}_{ij}^2 + \dot{K}_h \sum_{j=1}^{N} \dot{x}_{ij}z_{ij} - \sum_{j=1}^{N} \dot{x}_{ij}f_{ij} = 0\]

\[-\dot{D}_h \sum_{j=1}^{N} \dot{x}_{ij}z_{ij} + \dot{K}_h \sum_{j=1}^{N} \dot{x}_{ij}^2 - \sum_{j=1}^{N} z_{ij}f_{ij} = 0.\]  

(48)

Then, we can obtain estimates \(\dot{D}_h\) and \(\dot{K}_h\) by solving the above two equations:

\[
\begin{bmatrix}
\dot{D}_h \\
\dot{K}_h
\end{bmatrix} = \begin{bmatrix}
-\sum_{j=1}^{N} \dot{x}_{ij}^2 & \sum_{j=1}^{N} \dot{x}_{ij}z_{ij} \\
-\sum_{j=1}^{N} z_{ij}\dot{x}_{ij} & \sum_{j=1}^{N} \dot{x}_{ij}^2
\end{bmatrix}^{-1} \begin{bmatrix}
\sum_{j=1}^{N} \dot{x}_{ij}f_{ij} \\
\sum_{j=1}^{N} z_{ij}f_{ij}
\end{bmatrix}.
\]  

(49)

According to the moving average algorithm, we can obtain (14).