

# Adaptive Control of Uncertain Nonlinear Teleoperation Systems

Xia Liu<sup>1</sup>, Ran Tao<sup>2</sup>, Mahdi Tavakoli<sup>2</sup>

<sup>1</sup> School of Electrical and Information Engineering, Xihua University, Chengdu, Sichuan, 610039 China

<sup>2</sup> Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, T6G 2V4 Canada

e-mail: xliu\_uestc@yahoo.com, ran1@ualberta.ca, mahdi.tavakoli@ualberta.ca

**Abstract:** Kinematic parameters of a robotic manipulator are hard to measure precisely and the varying size and shape of tools held by the robot end-effector introduce further kinematic uncertainties. Moreover, the exact knowledge of the robot nonlinear dynamics may be unavailable due to model uncertainties. While adaptive master-slave teleoperation control strategies in the literature consider the *dynamic* uncertainties in the master and the slave robots, they stop short of accounting for the robots' *kinematic* uncertainties, which can undermine the transparency of the teleoperation system. In this paper, for a teleoperation system that is both dynamically and kinematically uncertain, we propose novel nonlinear adaptive controllers that require neither the exact knowledge of the kinematics of the master and the slave nor the dynamics of the master, the slave, the human operator, and the environment. Therefore, the proposed controllers can provide the master and slave robots with a high degree of flexibility in dealing with unforeseen changes and uncertainties in their kinematics and dynamics. A Lyapunov function analysis is conducted to mathematically prove the stability and master-slave asymptotic position tracking. The validity of the theoretical results is verified through simulations as well as experiments on a bilateral teleoperation test-bed of rehabilitation robots.

**Keywords:** Nonlinear adaptive control, kinematic uncertainty, dynamic uncertainty, teleoperation systems

## 1. Introduction

Transparency of a bilateral teleoperation system requires that, through appropriate control, the slave exactly reproduces the master's position trajectory in its environment while the master accurately displays the slave-environment contact force to the operator. In order to ensure the transparency of teleoperation systems while

preserving stability, various control approaches have been proposed [1]. Most of these control approaches assume perfect knowledge of the master and the slave dynamics. However, perfect knowledge of the master and the slave may be unavailable in practice due to model uncertainties. Therefore, adaptive control methods have been sought in the past to mitigate various parametric dynamical uncertainties [2] in a teleoperation system. In the following, we will summarize these prior schemes (sorted in the order of increased complexity).

Considering linear models for the master and the slave, an adaptive control scheme was proposed in [3] for teleoperation systems with dynamically uncertain slave and environment. In [4], several adaptive controllers were designed for teleoperation systems with a dynamically uncertain slave. A predictive adaptive controller was employed in [5] for teleoperation systems with time delay and a dynamically uncertain environment. In all of the above, the dynamics of the master and the slave were assumed to be linear.

Considering nonlinear multi-DOF models for the master and the slave, an adaptive control scheme was proposed in [6] for teleoperation systems with dynamically uncertain master and slave. In [7], adaptive controllers based on a virtual master model were designed for teleoperation systems. In [8], an adaptive teleoperation control scheme was proposed to ensure synchronization of positions and velocities. Later, it was shown in [9, 10] that the scheme in [8] could only be applied to teleoperation systems without gravity and then an improved adaptive controller was proposed. An adaptive controller was proposed by the authors for the master and slave robots having both linearly parameterized and nonlinearly parameterized dynamic uncertainties in [11]. While all of the above consider dynamically uncertain master and slave, they do not consider possible dynamic uncertainties in the human and the environment models.

Considering nonlinear master and slave models and linear human and environment models, adaptive teleoperation controllers were proposed in [12, 13]. An adaptive control method based on the inverse dynamics approach was developed by the authors in [14]. Here, the master, the slave, the operator and the environment were all

considered to be dynamically uncertain. These controllers work well for uncertain dynamics. However, in all of the above adaptive teleoperation control schemes, the kinematics of the master and slave robots are assumed to be known exactly.

Kinematic uncertainty of a robot is a practical fact and is a separate problem from dynamic uncertainty [15, 16]. The kinematics of a robot can be characterized by a set of parameters such as the lengths of links, link offsets, lengths and grasping angles of objects that the robot holds, and camera parameters if cameras are used to monitor the position of the end-effector. Kinematic parameters of a robot are hard to measure precisely. For instance, when a robot picks up objects of different lengths, unknown orientations and varying gripping points, the overall kinematics is unknown. Even if a known tool is used, the robot may not grasp the tool at the same point and with the same orientation every time. As a result of such kinematic uncertainty, the robot may not be able to manipulate the tool to a desired position. Thus, kinematic uncertainty has the potential to jeopardize the transparency of bilateral teleoperation systems. More examples of kinematic uncertainties are illustrated in Section 4.

Although interesting adaptive controllers coping with kinematic uncertainties were proposed in [17, 18], the results dealt with a single robot and not a teleoperation system; they did not directly address to the case of a teleoperation system in which the master/slave make contact with the operator/environment and bilateral control is involved. The contributions of this paper is in proposing a nonlinear adaptive control method for dynamically and kinematically uncertain teleoperation systems that works without the exact knowledge of the kinematics of the master or the slave, and without the exact knowledge of the dynamics of the master, the slave, the operator, or the environment. Considering the combined effects of not only the dynamic and kinematic uncertainties but also time delay in the communication channel of a teleoperation system is interesting yet beyond the scope of this paper. For a survey of teleoperation control schemes under time delay, please see [19].

The organization of this paper is as follows. The combined models of the master/operator and the slave/environment are discussed in Section 2. In Section 3, nonlinear adaptive controllers are designed for the master and the slave robots,

respectively, and the stability and asymptotic position tracking are mathematically proven. Examples of kinematic uncertainties are illustrated in Section 4. In Section 5, simulations as well as experiments are conducted to illustrate the performance of the proposed controller. The concluding remarks are presented in Section 6.

## 2. Dynamic and kinematic models of a teleoperation system

In this section, the models of the operator and the environment are incorporated into the models of the master and the slave, respectively, to obtain a combined model of the entire system.

### 2.1 Nonlinear dynamic models of the master and slave in joint space

The nonlinear dynamic models of an  $n$ -DOF master robot and an  $n$ -DOF slave robot in joint space are

$$\mathbf{M}_{qm}(\mathbf{q}_m)\ddot{\mathbf{q}}_m + \mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \mathbf{G}_{qm}(\mathbf{q}_m) = \boldsymbol{\tau}_m + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{f}_h \quad (1)$$

$$\mathbf{M}_{qs}(\mathbf{q}_s)\ddot{\mathbf{q}}_s + \mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s + \mathbf{G}_{qs}(\mathbf{q}_s) = \boldsymbol{\tau}_s - \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{f}_e \quad (2)$$

where  $\mathbf{q}_m, \mathbf{q}_s \in \mathfrak{R}^{n \times 1}$  are joint angle positions,  $\mathbf{M}_{qm}(\mathbf{q}_m), \mathbf{M}_{qs}(\mathbf{q}_s) \in \mathfrak{R}^{n \times n}$  are positive-definite and symmetric inertia matrices,  $\mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m), \mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s) \in \mathfrak{R}^{n \times n}$  are Coriolis/centrifugal matrices,  $\mathbf{G}_{qm}(\mathbf{q}_m), \mathbf{G}_{qs}(\mathbf{q}_s) \in \mathfrak{R}^{n \times 1}$  are gravity terms,  $\boldsymbol{\tau}_m, \boldsymbol{\tau}_s \in \mathfrak{R}^{n \times 1}$  are input control torques, and  $\mathbf{J}_m(\mathbf{q}_m), \mathbf{J}_s(\mathbf{q}_s) \in \mathfrak{R}^{n \times n}$  are the Jacobian matrices for the master and the slave, respectively. Besides,  $\mathbf{f}_h \in \mathfrak{R}^{n \times 1}$  and  $\mathbf{f}_e \in \mathfrak{R}^{n \times 1}$  denote the human/master and the slave/environment contact forces, respectively. Note that the subscripts  $m$  and  $s$  for the master and the slave, respectively, are omitted in the following properties:

**Property 1**[20]. The left sides of (1) and (2) are linear in a set of dynamic parameters

$$\boldsymbol{\theta}_d = [\theta_{d1}, \dots, \theta_{dp}]^T \text{ as}$$

$$\mathbf{M}_q(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_q(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_q(\mathbf{q}) = \mathbf{Y}_d(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\theta}_d$$

where  $\mathbf{Y}_d(\mathbf{q}, \dot{\mathbf{q}}) \in \mathfrak{R}^{n \times p}$  is called the dynamic regressor matrix.

**Property 2**[20]. The matrix  $\dot{\mathbf{M}}_q(\mathbf{q}) - 2\mathbf{C}_q(\mathbf{q}, \dot{\mathbf{q}})$  is skew-symmetric, i.e.,

$$\zeta^T (\dot{\mathbf{M}}_q(\mathbf{q}) - 2\mathbf{C}_q(\mathbf{q}, \dot{\mathbf{q}}))\xi = 0, \forall \xi \in \mathfrak{R}^{n \times 1}.$$

## 2.2 Nonlinear kinematic models of the master and slave

The kinematics of a robot specifies the relationship between the positions in the task space and the joint space. The robots' end-effector positions  $\mathbf{x}_m, \mathbf{x}_s \in \mathfrak{R}^{6 \times 1}$  of the master and the slave can be expressed as:

$$\mathbf{x}_m = \mathbf{h}_m(\mathbf{q}_m), \mathbf{x}_s = \mathbf{h}_s(\mathbf{q}_s) \quad (3)$$

where  $\mathbf{h}(\cdot) \in \mathfrak{R}^n \rightarrow \mathfrak{R}^6$  is nonlinear in general. The relationships between the task-space and the joint-space velocities are

$$\dot{\mathbf{x}}_m = \mathbf{J}_m(\mathbf{q}_m)\dot{\mathbf{q}}_m, \dot{\mathbf{x}}_s = \mathbf{J}_s(\mathbf{q}_s)\dot{\mathbf{q}}_s \quad (4)$$

where  $\mathbf{J}_m(\mathbf{q}_m)$  and  $\mathbf{J}_s(\mathbf{q}_s)$  are the Jacobian matrices of the master and the slave, respectively.

Differentiating (4) with respect to time yields

$$\ddot{\mathbf{x}}_m = \dot{\mathbf{J}}_m(\mathbf{q}_m)\dot{\mathbf{q}}_m + \mathbf{J}_m(\mathbf{q}_m)\ddot{\mathbf{q}}_m \quad (5)$$

$$\ddot{\mathbf{x}}_s = \dot{\mathbf{J}}_s(\mathbf{q}_s)\dot{\mathbf{q}}_s + \mathbf{J}_s(\mathbf{q}_s)\ddot{\mathbf{q}}_s \quad (6)$$

**Property 3**[17, 18]. Equation (4) is linear in a set of kinematic parameters

$\boldsymbol{\theta}_k = (\theta_{k1}, \dots, \theta_{kw})^T$  and can be expressed as

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{Y}_k(\mathbf{q}, \dot{\mathbf{q}})\boldsymbol{\theta}_k$$

where  $\mathbf{Y}_k(\mathbf{q}, \dot{\mathbf{q}}) \in \mathfrak{R}^{6 \times w}$  is called the kinematic regressor matrix.

## 2.3 Linear dynamic models of the operator and environment in task space

The dynamics of the human operator and the environment are naturally specified in the task space where they make contact with the master and the slave robots. For the operator and environment, the following second-order LTI models have been successfully used in the past [12, 13]:

$$\mathbf{f}_h = \mathbf{f}_h^* - (\mathbf{M}_h \ddot{\mathbf{x}}_m + \mathbf{B}_h \dot{\mathbf{x}}_m + \mathbf{K}_h \mathbf{x}_m) \quad (7)$$

$$\mathbf{f}_e = \mathbf{f}_e^* + \mathbf{M}_e \ddot{\mathbf{x}}_s + \mathbf{B}_e \dot{\mathbf{x}}_s + \mathbf{K}_e \mathbf{x}_s \quad (8)$$

where  $\mathbf{M}_h, \mathbf{M}_e, \mathbf{B}_h, \mathbf{B}_e, \mathbf{K}_h, \mathbf{K}_e \in \mathfrak{R}^{6 \times 6}$  are the mass, damping, and stiffness matrices of the operator's hand and the environment, respectively. These are constant, symmetric and positive matrices. Also,  $\mathbf{f}_h^*$  and  $\mathbf{f}_e^*$  are the exogenous forces of the human operator and the environment, respectively.

#### 2.4 End-to-end teleoperation system model in joint space

To this end, substituting (3)-(6) into (7)-(8), the models of the human and the environment in joint space become

$$\mathbf{f}_h = \mathbf{f}_h^* - \mathbf{M}_h \mathbf{J}_m(\mathbf{q}_m) \ddot{\mathbf{q}}_m - (\mathbf{B}_h \mathbf{J}_m(\mathbf{q}_m) + \mathbf{M}_h \dot{\mathbf{J}}_m(\mathbf{q}_m)) \dot{\mathbf{q}}_m - \mathbf{K}_h \mathbf{h}_m(\mathbf{q}_m) \quad (9)$$

$$\mathbf{f}_e = \mathbf{f}_e^* + \mathbf{M}_e \mathbf{J}_s(\mathbf{q}_s) \ddot{\mathbf{q}}_s + (\mathbf{B}_e \mathbf{J}_s(\mathbf{q}_s) + \mathbf{M}_e \dot{\mathbf{J}}_s(\mathbf{q}_s)) \dot{\mathbf{q}}_s + \mathbf{K}_e \mathbf{h}_s(\mathbf{q}_s) \quad (10)$$

Multiplying (9)-(10) by  $\mathbf{J}_m^T(\mathbf{q}_m)$  and  $\mathbf{J}_s^T(\mathbf{q}_s)$ , respectively, and substituting them in (1)-(2) gives a combined model for the master/operator system and another combined model for the slave/environment system:

$$\mathbf{M}_m(\mathbf{q}_m) \ddot{\mathbf{q}}_m + \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m) \dot{\mathbf{q}}_m + \mathbf{G}_m(\mathbf{q}_m) = \boldsymbol{\tau}_m \quad (11)$$

$$\mathbf{M}_s(\mathbf{q}_s) \ddot{\mathbf{q}}_s + \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s) \dot{\mathbf{q}}_s + \mathbf{G}_s(\mathbf{q}_s) = \boldsymbol{\tau}_s \quad (12)$$

where

$$\mathbf{M}_m(\mathbf{q}_m) = \mathbf{M}_{qm}(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{M}_h \mathbf{J}_m(\mathbf{q}_m),$$

$$\mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m) = \mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m) + \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{B}_h \mathbf{J}_m(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{M}_h \dot{\mathbf{J}}_m(\mathbf{q}_m),$$

$$\mathbf{G}_m(\mathbf{q}_m) = \mathbf{G}_{qm}(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{K}_h \mathbf{h}_m(\mathbf{q}_m) - \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{f}_h^*,$$

$$\mathbf{M}_s(\mathbf{q}_s) = \mathbf{M}_{qs}(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{M}_e \mathbf{J}_s(\mathbf{q}_s),$$

$$\mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s) = \mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s) + \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{B}_e \mathbf{J}_s(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{M}_e \dot{\mathbf{J}}_s(\mathbf{q}_s),$$

$$\mathbf{G}_s(\mathbf{q}_s) = \mathbf{G}_{qs}(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{K}_e \mathbf{h}_s(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{f}_e^*.$$

Note that for the combined models (11)-(12), Property 1 still holds but Property 2 does not hold. Instead, a new property holds:

**Property 4.** For  $\forall \boldsymbol{\xi} \in \mathfrak{R}^{6 \times 1}$ , we have

$$\xi^T (\dot{\mathbf{M}}_m(\mathbf{q}_m) - 2\mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m))\xi = -2\xi^T (\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m))\xi,$$

and

$$\xi^T (\dot{\mathbf{M}}_s(\mathbf{q}_s) - 2\mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s))\xi = -2\xi^T (\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s))\xi.$$

The detailed proof of Property 4 can be found in Appendix 1.

### 3. Control of a dynamically and kinematically uncertain teleoperation system

In this section, nonlinear adaptive controllers are designed for the master and the slave robots with dynamic and kinematic uncertainties. We also study the stability of the system and the position tracking performance between the master and slave via a Lyapunov function analysis.

#### 3.1 Dynamic and kinematic uncertainties in teleoperation systems

In the presence of parametric uncertainties in the dynamics, the left sides of (11) and (12) become

$$\hat{\mathbf{M}}_m(\mathbf{q}_m)\ddot{\mathbf{q}}_m + \hat{\mathbf{C}}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_m + \hat{\mathbf{G}}_m(\mathbf{q}_m) = \mathbf{Y}_{md}(\mathbf{q}_m, \dot{\mathbf{q}}_m, \ddot{\mathbf{q}}_m)\hat{\boldsymbol{\theta}}_{md} \quad (13)$$

$$\hat{\mathbf{M}}_s(\mathbf{q}_s)\ddot{\mathbf{q}}_s + \hat{\mathbf{C}}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_s + \hat{\mathbf{G}}_s(\mathbf{q}_s) = \mathbf{Y}_{sd}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s)\hat{\boldsymbol{\theta}}_{sd} \quad (14)$$

where  $\hat{\boldsymbol{\theta}}_{md}, \hat{\boldsymbol{\theta}}_{sd}$  are estimates of the dynamic parameter vectors  $\boldsymbol{\theta}_{md}, \boldsymbol{\theta}_{sd}$ , respectively.

On the other hand, when the kinematic parameters of the master and the slave are uncertain, the Jacobian matrices experience parametric uncertainties, which means that (4) becomes

$$\hat{\dot{\mathbf{x}}}_m = \hat{\mathbf{J}}_m(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})\dot{\mathbf{q}}_m = \mathbf{Y}_{mk}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\hat{\boldsymbol{\theta}}_{mk} \quad (15)$$

$$\hat{\dot{\mathbf{x}}}_s = \hat{\mathbf{J}}_s(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})\dot{\mathbf{q}}_s = \mathbf{Y}_{sk}(\mathbf{q}_s, \dot{\mathbf{q}}_s)\hat{\boldsymbol{\theta}}_{sk} \quad (16)$$

where  $\hat{\dot{\mathbf{x}}}_m$  and  $\hat{\dot{\mathbf{x}}}_s$  are the estimates of  $\dot{\mathbf{x}}_m$  and  $\dot{\mathbf{x}}_s$ ,  $\hat{\mathbf{J}}_m(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})$  and  $\hat{\mathbf{J}}_s(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})$  are the estimates of  $\mathbf{J}_m(\mathbf{q}_m)$  and  $\mathbf{J}_s(\mathbf{q}_s)$ , and  $\hat{\boldsymbol{\theta}}_{mk}$  and  $\hat{\boldsymbol{\theta}}_{sk}$  are the estimates of the kinematic parameter vectors of the master  $\boldsymbol{\theta}_{mk}$  and the slave  $\boldsymbol{\theta}_{sk}$ , respectively.

### 3.2 Proposed adaptive teleoperation control architecture

The proposed adaptive control is expected to achieve master/slave position tracking irrespective of the dynamic and kinematic uncertainties described in Section 3.1. The principle of PEB (position-error-based) teleoperation control [21] is to minimize the difference between the master and the slave positions while reflecting a force proportional to this difference to the operator once the slave makes contact with an environment. In such a way, teleoperation transparency can be achieved via PEB architecture. Our proposed *adaptive* PEB teleoperation system is shown in Fig. 1. Adaptive controllers are designed for the combined master/operator system and the combined slave/environment system assuming the master, the slave, the human operator, and the environment are dynamically uncertain and the master and the slave are kinematically uncertain.

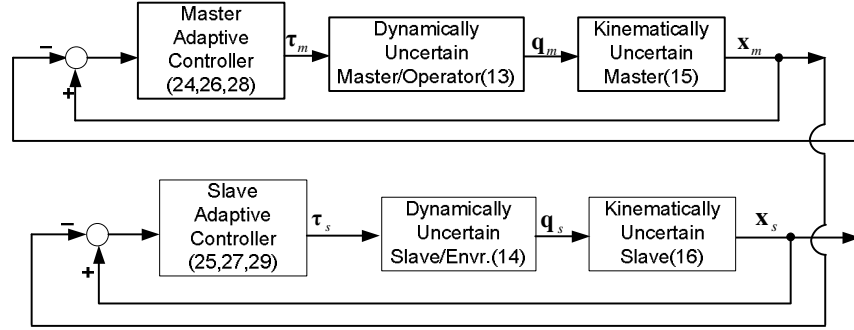


Fig. 1. Principle of adaptive PEB bilateral teleoperation control

### 3.3 Design of nonlinear adaptive controller

First, define two new vectors in joint space for the master side and the slave side:

$$\dot{\mathbf{q}}_{mr} = \hat{\mathbf{J}}_m^{-1}(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk}) \dot{\mathbf{x}}_{mr}, \dot{\mathbf{q}}_{sr} = \hat{\mathbf{J}}_s^{-1}(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk}) \dot{\mathbf{x}}_{sr} \quad (17)$$

where

$$\dot{\mathbf{x}}_{mr} = \dot{\mathbf{x}}_s - \alpha \Delta \mathbf{x}_m, \dot{\mathbf{x}}_{sr} = \dot{\mathbf{x}}_m - \alpha \Delta \mathbf{x}_s \quad (18)$$

and  $\Delta \mathbf{x}_m = \mathbf{x}_m - \mathbf{x}_s$  and  $\Delta \mathbf{x}_s = \mathbf{x}_s - \mathbf{x}_m$  are position errors. Also,  $\alpha$  is a positive constant. Next, define two adaptive sliding vectors in joint space for the master and



the slave as

$$\mathbf{s}_m = \dot{\mathbf{q}}_m - \dot{\mathbf{q}}_{mr}, \mathbf{s}_s = \dot{\mathbf{q}}_s - \dot{\mathbf{q}}_{sr} \quad (19)$$

Thus, we have

$$\dot{\mathbf{q}}_m = \mathbf{s}_m + \dot{\mathbf{q}}_{mr}, \dot{\mathbf{q}}_s = \mathbf{s}_s + \dot{\mathbf{q}}_{sr} \quad (20)$$

and

$$\ddot{\mathbf{q}}_m = \dot{\mathbf{s}}_m + \ddot{\mathbf{q}}_{mr}, \ddot{\mathbf{q}}_s = \dot{\mathbf{s}}_s + \ddot{\mathbf{q}}_{sr} \quad (21)$$

Substituting (20)-(21) into (11)-(12) and using Property 1, the equations governing the *open-loop* system can be written as

$$\mathbf{M}_m(\mathbf{q}_m)\dot{\mathbf{s}}_m + \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\mathbf{s}_m + \mathbf{Y}_{md}(\mathbf{q}_m, \dot{\mathbf{q}}_m, \dot{\mathbf{q}}_{mr}, \ddot{\mathbf{q}}_{mr})\boldsymbol{\theta}_{md} = \boldsymbol{\tau}_m \quad (22)$$

$$\mathbf{M}_s(\mathbf{q}_s)\dot{\mathbf{s}}_s + \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\mathbf{s}_s + \mathbf{Y}_{sd}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \dot{\mathbf{q}}_{sr}, \ddot{\mathbf{q}}_{sr})\boldsymbol{\theta}_{sd} = \boldsymbol{\tau}_s \quad (23)$$

where

$$\mathbf{Y}_{md}\boldsymbol{\theta}_{md} = \mathbf{M}_m(\mathbf{q}_m)\ddot{\mathbf{q}}_{mr} + \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\dot{\mathbf{q}}_{mr} + \mathbf{G}_m(\mathbf{q}_m),$$

$$\mathbf{Y}_{sd}\boldsymbol{\theta}_{sd} = \mathbf{M}_s(\mathbf{q}_s)\ddot{\mathbf{q}}_{sr} + \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\dot{\mathbf{q}}_{sr} + \mathbf{G}_s(\mathbf{q}_s).$$

The proposed control algorithm is composed of three parts: control laws, dynamic update laws, and kinematic update laws. In the following, we list each of these.

- Control laws for the master and the slave:

$$\boldsymbol{\tau}_m = \mathbf{Y}_{md}\hat{\boldsymbol{\theta}}_{md} - \hat{\mathbf{J}}_m^T(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})\mathbf{K}_m(\Delta\hat{\mathbf{x}}_m + \alpha\Delta\mathbf{x}_m) \quad (24)$$

$$\boldsymbol{\tau}_s = \mathbf{Y}_{sd}\hat{\boldsymbol{\theta}}_{sd} - \hat{\mathbf{J}}_s^T(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})\mathbf{K}_s(\Delta\hat{\mathbf{x}}_s + \alpha\Delta\mathbf{x}_s) \quad (25)$$

where  $\mathbf{K}_m$  and  $\mathbf{K}_s$  are symmetric positive definite matrices,  $\Delta\hat{\mathbf{x}}_m = \hat{\mathbf{x}}_m - \dot{\mathbf{x}}_s$ , and  $\Delta\hat{\mathbf{x}}_s = \hat{\mathbf{x}}_s - \dot{\mathbf{x}}_m$ .

- Dynamic update laws:

$$\dot{\hat{\boldsymbol{\theta}}}_{md} = -\mathbf{L}_{md}\mathbf{Y}_{md}^T\mathbf{s}_m \quad (26)$$

$$\dot{\hat{\boldsymbol{\theta}}}_{sd} = -\mathbf{L}_{sd}\mathbf{Y}_{sd}^T\mathbf{s}_s \quad (27)$$

- Kinematic update laws:

$$\dot{\hat{\boldsymbol{\theta}}}_{mk} = 2\mathbf{L}_{mk}\mathbf{Y}_{mk}^T(\mathbf{q}_m, \dot{\mathbf{q}}_m)\mathbf{K}_m(\Delta\dot{\mathbf{x}}_m + \alpha\Delta\mathbf{x}_m) \quad (28)$$

$$\dot{\hat{\boldsymbol{\theta}}}_{sk} = 2\mathbf{L}_{sk}\mathbf{Y}_{sk}^T(\mathbf{q}_s, \dot{\mathbf{q}}_s)\mathbf{K}_s(\Delta\dot{\mathbf{x}}_s + \alpha\Delta\mathbf{x}_s) \quad (29)$$

where  $\mathbf{L}_{md}, \mathbf{L}_{mk}, \mathbf{L}_{sd}$  and  $\mathbf{L}_{sk}$  are symmetric and positive-definite matrices.

Each of the control laws (24)-(25) consists of two parts. The first part  $\mathbf{Y}_{md}\hat{\boldsymbol{\theta}}_{md}$  for the master ( $\mathbf{Y}_{sd}\hat{\boldsymbol{\theta}}_{sd}$  for the slave) is feedforward model-based compensation for the robot dynamics while the second part  $\hat{\mathbf{J}}_m^T(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})\mathbf{K}_m(\Delta\hat{\mathbf{x}}_m + \alpha\Delta\mathbf{x}_m)$  for the master ( $-\hat{\mathbf{J}}_s^T(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})\mathbf{K}_s(\Delta\hat{\mathbf{x}}_s + \alpha\Delta\mathbf{x}_s)$  for the slave) involves feedback compensation for velocity and position tracking.

Substitute (24)-(25) into (22)-(23) to obtain the dynamics for the *closed-loop* teleoperation system as

$$\mathbf{M}_m(\mathbf{q}_m)\dot{\mathbf{s}}_m + \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)\mathbf{s}_m + \mathbf{Y}_{md}\Delta\boldsymbol{\theta}_{md} + \hat{\mathbf{J}}_m^T(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})\mathbf{K}_m(\Delta\hat{\mathbf{x}}_m + \alpha\Delta\mathbf{x}_m) = 0 \quad (30)$$

$$\mathbf{M}_s(\mathbf{q}_s)\dot{\mathbf{s}}_s + \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s)\mathbf{s}_s + \mathbf{Y}_{sd}\Delta\boldsymbol{\theta}_{sd} + \hat{\mathbf{J}}_s^T(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})\mathbf{K}_s(\Delta\hat{\mathbf{x}}_s + \alpha\Delta\mathbf{x}_s) = 0 \quad (31)$$

where  $\Delta\boldsymbol{\theta}_{md} = \boldsymbol{\theta}_{md} - \hat{\boldsymbol{\theta}}_{md}$  and  $\Delta\boldsymbol{\theta}_{sd} = \boldsymbol{\theta}_{sd} - \hat{\boldsymbol{\theta}}_{sd}$ .

**Theorem:** Consider the nonlinear teleoperation system described by (11)-(12) under the dynamic uncertainties (13)-(14) and the kinematic uncertainties (15)-(16). Then, using the control laws (24)-(25) with the dynamic update laws (26)-(27) and the kinematic update laws (28)-(29) makes the position tracking error in the closed-loop system (30)-(31) asymptotically converge to zero, i.e.,  $\lim_{t \rightarrow \infty}(\mathbf{x}_s - \mathbf{x}_m) \rightarrow 0$ , and the force tracking error  $\mathbf{f}_h - \mathbf{f}_e$  is bounded.

**Proof:** Consider a Lyapunov candidate function as

$$V = V_1 + V_2 \quad (32)$$

where each of  $V_1$  and  $V_2$  are the Lyapunov functions for a single robot [18]:

$$V_1 = \frac{1}{2}\mathbf{s}_m^T\mathbf{M}_m(\mathbf{q}_m)\mathbf{s}_m + \alpha\Delta\mathbf{x}_m^T\mathbf{K}_m\Delta\mathbf{x}_m + \frac{1}{2}\Delta\boldsymbol{\theta}_{md}^T\mathbf{L}_{md}^{-1}\Delta\boldsymbol{\theta}_{md} + \frac{1}{2}\Delta\boldsymbol{\theta}_{mk}^T\mathbf{L}_{mk}^{-1}\Delta\boldsymbol{\theta}_{mk} \quad (33)$$

$$V_2 = \frac{1}{2}\mathbf{s}_s^T\mathbf{M}_s(\mathbf{q}_s)\mathbf{s}_s + \alpha\Delta\mathbf{x}_s^T\mathbf{K}_s\Delta\mathbf{x}_s + \frac{1}{2}\Delta\boldsymbol{\theta}_{sd}^T\mathbf{L}_{sd}^{-1}\Delta\boldsymbol{\theta}_{sd} + \frac{1}{2}\Delta\boldsymbol{\theta}_{sk}^T\mathbf{L}_{sk}^{-1}\Delta\boldsymbol{\theta}_{sk} \quad (34)$$

Since  $\mathbf{M}_m(\mathbf{q}_m), \mathbf{M}_s(\mathbf{q}_s), \mathbf{K}_m, \mathbf{K}_s, \mathbf{L}_{md}, \mathbf{L}_{sd}, \mathbf{L}_{mk}$  and  $\mathbf{L}_{sk}$  are all positive definite,  $v$  is

positive definite. Using Property 4, the derivative of  $v$  along the trajectories of the closed-loop system (30)-(31) is

$$\begin{aligned}\dot{v} = & -\mathbf{K}_m(\Delta\dot{\mathbf{x}}_m^T\Delta\dot{\mathbf{x}}_m + \alpha^2\Delta\mathbf{x}_m^T\Delta\mathbf{x}_m + \Delta\boldsymbol{\theta}_{mk}^T\mathbf{Y}_{mk}^T(\mathbf{q}_m, \dot{\mathbf{q}}_m)\mathbf{Y}_{mk}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\Delta\boldsymbol{\theta}_{mk}) \\ & -\mathbf{K}_s(\Delta\dot{\mathbf{x}}_s^T\Delta\dot{\mathbf{x}}_s + \alpha^2\Delta\mathbf{x}_s^T\Delta\mathbf{x}_s + \Delta\boldsymbol{\theta}_{sk}^T\mathbf{Y}_{sk}^T(\mathbf{q}_s, \dot{\mathbf{q}}_s)\mathbf{Y}_{sk}(\mathbf{q}_s, \dot{\mathbf{q}}_s)\Delta\boldsymbol{\theta}_{sk}) \\ & -\mathbf{s}_m^T(\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m))\mathbf{s}_m - \mathbf{s}_s^T(\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s))\mathbf{s}_s\end{aligned}\quad (35)$$

Note that  $\mathbf{K}_m, \mathbf{K}_s, \mathbf{B}_h$  and  $\mathbf{B}_e$  are all positive definite matrices, and that for

$\mathbf{s}_m = \mathbf{s}_s = \Delta\mathbf{x}_m = \Delta\mathbf{x}_s = \Delta\boldsymbol{\theta}_{mk} = \Delta\boldsymbol{\theta}_{sk} = 0$ , but  $\Delta\boldsymbol{\theta}_{md} \neq 0$  or  $\Delta\boldsymbol{\theta}_{sd} \neq 0$ , we have  $\dot{v} = 0$ .

Therefore,  $\dot{v}$  is negative semi-definite, meaning that  $v$  is bounded and the signals

$\mathbf{s}_m, \mathbf{s}_s, \Delta\mathbf{x}_m, \Delta\mathbf{x}_s, \Delta\boldsymbol{\theta}_{md}, \Delta\boldsymbol{\theta}_{sd}, \Delta\boldsymbol{\theta}_{mk}$  and  $\Delta\boldsymbol{\theta}_{sk}$  are all bounded as well. Knowing that  $\Delta\mathbf{x}_s$

is bounded, let us integrate  $\dot{v}$  with respect to time to get

$$\begin{aligned}V(t) = & \int_0^t \dot{v} dt = -\int_0^t \mathbf{K}_m(\|\Delta\dot{\mathbf{x}}_m\|^2 + \alpha^2\|\Delta\mathbf{x}_m\|^2 + \|\mathbf{Y}_{mk}(\mathbf{q}_m, \dot{\mathbf{q}}_m)\Delta\boldsymbol{\theta}_{mk}\|^2) dt \\ & -\int_0^t \mathbf{K}_s(\|\Delta\dot{\mathbf{x}}_s\|^2 + \alpha^2\|\Delta\mathbf{x}_s\|^2 + \|\mathbf{Y}_{sk}(\mathbf{q}_s, \dot{\mathbf{q}}_s)\Delta\boldsymbol{\theta}_{sk}\|^2) dt \\ & -\int_0^t (\mathbf{B}_h\mathbf{s}_m^T(\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{J}_m(\mathbf{q}_m))\mathbf{s}_m + \mathbf{B}_e\mathbf{s}_s^T(\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{J}_s(\mathbf{q}_s))\mathbf{s}_s) dt\end{aligned}\quad (36)$$

Since  $V(t)$  is proven to be bounded, we get that  $\int_0^\infty \|\Delta\mathbf{x}_s\|^2 dt$  and  $\int_0^\infty \|\Delta\dot{\mathbf{x}}_s\|^2 dt$  are

bounded (i.e.,  $\Delta\mathbf{x}_s, \Delta\dot{\mathbf{x}}_s \in L^2$ ). As for a robotic manipulator, it is not unreasonable to

deduce that  $\Delta\dot{\mathbf{x}}_s$  is also bounded as the kinetic energy is limited anyway. Since  $\Delta\mathbf{x}_s$

is bounded,  $\Delta\dot{\mathbf{x}}_s$  is bounded,  $\Delta\mathbf{x}_s \in L^2$ , and using Barbalat's lemma [22], we can

finally get that  $\lim_{t \rightarrow \infty} \Delta\mathbf{x}_s = \lim_{t \rightarrow \infty} (\mathbf{x}_s - \mathbf{x}_m) \rightarrow 0$ .

Now let's analyze force tracking performance. From (18) we could get

$\dot{\mathbf{x}}_{mr} - \dot{\mathbf{x}}_{sr} = \dot{\mathbf{x}}_s - \dot{\mathbf{x}}_m - \alpha\Delta\mathbf{x}_m + \alpha\Delta\mathbf{x}_s = \Delta\dot{\mathbf{x}}_s - \alpha\Delta\mathbf{x}_m + \alpha\Delta\mathbf{x}_s$ . Since  $\Delta\mathbf{x}_m$ ,  $\Delta\mathbf{x}_s$  and  $\Delta\dot{\mathbf{x}}_s$  are

bounded,  $\dot{\mathbf{x}}_{mr} - \dot{\mathbf{x}}_{sr}$  is bounded. Therefore,  $\dot{\mathbf{q}}_{mr} - \dot{\mathbf{q}}_{sr}$  is bounded according to (17).

Furthermore, from (19) we know that  $\mathbf{s}_m - \mathbf{s}_s = \dot{\mathbf{q}}_m - \dot{\mathbf{q}}_s + \dot{\mathbf{q}}_{sr} - \dot{\mathbf{q}}_{mr}$ . Thus, the

boundedness of  $\dot{\mathbf{q}}_{mr} - \dot{\mathbf{q}}_{sr}$  means that  $\dot{\mathbf{q}}_m - \dot{\mathbf{q}}_s$  is bounded as  $\mathbf{s}_m$  and  $\mathbf{s}_s$  have proved

to be bounded. In addition, since  $\mathbf{s}_m, \mathbf{s}_s, \Delta\mathbf{x}_m, \Delta\mathbf{x}_s, \Delta\boldsymbol{\theta}_{md}, \Delta\boldsymbol{\theta}_{sd}, \Delta\boldsymbol{\theta}_{mk}$  and  $\Delta\boldsymbol{\theta}_{sk}$  are all

bounded, it is not difficult to obtain that  $\dot{\mathbf{s}}_m$  and  $\dot{\mathbf{s}}_s$  are bounded and so is  $\dot{\mathbf{s}}_m - \dot{\mathbf{s}}_s$ . Therefore, from (22)-(23) we get that  $\boldsymbol{\tau}_m - \boldsymbol{\tau}_s$  is bounded. As a result, from (11)-(12) we can obtain that  $\ddot{\mathbf{q}}_m - \ddot{\mathbf{q}}_s$  is bounded. So far, we have got  $\dot{\mathbf{q}}_m - \dot{\mathbf{q}}_s$  and  $\ddot{\mathbf{q}}_m - \ddot{\mathbf{q}}_s$  are bounded. Now, let us go back to (9)-(10). Generally, in (9)-(10) the environment is assumed to be passive, i.e.,  $\mathbf{f}_e^* = 0$  [21], and the exogenous force of the human operator  $\mathbf{f}_h^*$  is subject to  $\|\mathbf{f}_h^*\|_\infty \leq \alpha_h < +\infty$ , where  $\alpha_h > 0$  is a constant [12, 13]. Also, since  $\Delta \mathbf{x}_m$  is bounded, from (3) we can get that  $\mathbf{h}_m(\mathbf{q}_m) - \mathbf{h}_s(\mathbf{q}_s)$  is bounded as  $\mathbf{h}_m(\mathbf{q}_m) - \mathbf{h}_s(\mathbf{q}_s) = \Delta \mathbf{x}_m = \mathbf{x}_m - \mathbf{x}_s$ . The results got so far finally ensure the boundedness of the force tracking error  $\mathbf{f}_h - \mathbf{f}_e$  between the human/master contact force and the slave/environment contact force.

**Remark 1:** In (17), we use the estimated Jacobian matrices  $\hat{\mathbf{J}}_m^{-1}(\mathbf{q}_m, \hat{\boldsymbol{\theta}}_{mk})$  and  $\hat{\mathbf{J}}_s^{-1}(\mathbf{q}_s, \hat{\boldsymbol{\theta}}_{sk})$  assuming that the robots are operating in a finite task space such that the estimated Jacobian matrices are of full rank. In addition, standard projection algorithms [23, 24] can be used to ensure that the estimated kinematic parameter vectors  $\hat{\boldsymbol{\theta}}_{mk}$  and  $\hat{\boldsymbol{\theta}}_{sk}$  remain in an appropriate region such that (17) is defined for all  $\hat{\boldsymbol{\theta}}_{mk}$  and  $\hat{\boldsymbol{\theta}}_{sk}$  during adaptation. Also, we note that singularities often depend only on  $\mathbf{q}_m$  and  $\mathbf{q}_s$ , not  $\hat{\boldsymbol{\theta}}_{mk}$  and  $\hat{\boldsymbol{\theta}}_{sk}$ . Alternatively, we may use a singularity-robust inverse of the estimated Jacobian matrix [25].

#### 4. Examples of kinematic uncertainties

In this section, we consider a two-link, revolute-joint robot to illustrate three typical examples that involve kinematic uncertainties [17-18].

First, we know that the dynamics of a two-link, revolute-joint robot is [20]

$$\mathbf{M}_q(\mathbf{q}) = \begin{bmatrix} l_2^2 m_2 + 2l_1 l_2 m_2 c_2 + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 c_2 \\ l_2^2 m_2 + l_1 l_2 m_2 c_2 & l_2^2 m_2 \end{bmatrix},$$

$$\mathbf{C}_q(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} -2l_1 l_2 m_2 s_2 \dot{q}_2 & -l_1 l_2 m_2 s_2 \dot{q}_2 \\ l_1 l_2 m_2 s_2 \dot{q}_2 & 0 \end{bmatrix},$$

$$\mathbf{G}_q(\mathbf{q}) = \begin{bmatrix} m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ m_2 l_2 g c_{12} \end{bmatrix}.$$

where  $q_1$  and  $q_2$  are the first and the second joint angles,  $l_1$  and  $l_2$  are the lengths of the links,  $m_1$  and  $m_2$  are the point masses of the links, respectively, and  $s_1 = \sin(q_1)$ ,  $s_2 = \sin(q_2)$ ,  $s_{12} = \sin(q_1 + q_2)$ ,  $c_1 = \cos(q_1)$ ,  $c_2 = \cos(q_2)$ ,  $c_{12} = \cos(q_1 + q_2)$ . Also,  $g$  is the gravitational constant. Therefore, dynamic uncertainty exists when  $l_1$ ,  $l_2$ ,  $m_1$  and/or  $m_2$  are uncertain.

As for kinematics, here are three examples that involve kinematic uncertainties.

### Example 1: A robot with uncertain kinematics

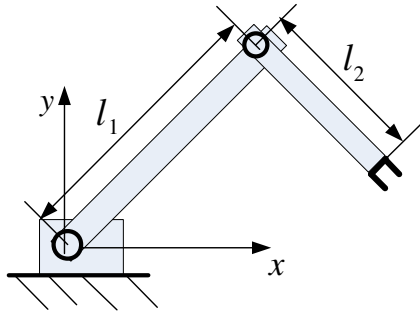


Fig.2 A two-link robot with uncertain kinematics

If a position sensor is used for the end-effector, the task space is defined as the Cartesian space. For the two-link robot shown in Fig. 2, the Jacobian matrix  $J(q)$  mapping from the robot joint space to Cartesian space is

$$J(q) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \end{bmatrix}$$

Therefore, kinematic uncertainty exists when  $l_1$  and/or  $l_2$  are uncertain.

**Example 2: A robot holding an object**

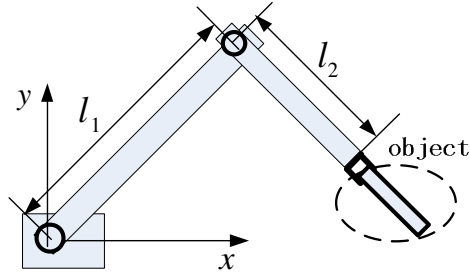


Fig.3 A two-link robot holding an object

For a two-link robot holding an object as shown in Fig.3, the Jacobian matrix  $J(q)$  from joint space to Cartesian space can be derived as

$$J(q) = \begin{bmatrix} -(l_1 s_1 + l_2 s_{12} + l_0 s_{120}) & -(l_2 s_{12} + l_0 s_{120}) \\ l_1 c_1 + l_2 c_{12} + l_0 c_{120} & l_2 c_{12} + l_0 c_{120} \end{bmatrix}$$

where  $l_0$  and  $q_0$  are the length and grasping angle of the object, respectively, and

$$s_{120} = \sin(q_1 + q_2 + q_0), c_{120} = \cos(q_1 + q_2 + q_0).$$

Therefore, kinematic uncertainty exists for uncertain  $l_1$ ,  $l_2$ ,  $l_0$  and/or  $q_0$ .

**Example 3: A robot with vision system**

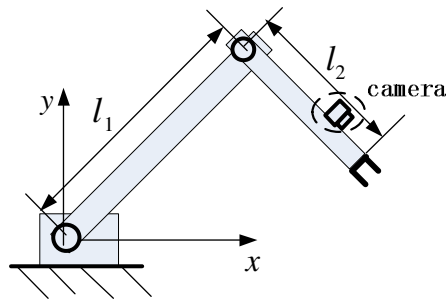


Fig.4 A two-link robot with a camera in an eye-in-hand configuration

If cameras are used to monitor the position of the end-effector, the task space is defined as the image space (in pixels). For a two-link planar robot with a camera in an

eye-in-hand configuration[26] as shown in Fig.4, the image Jacobian matrix[27]is given by

$$J_I(q) = \frac{f}{z-f} \begin{bmatrix} \beta_1 & 0 \\ 0 & \beta_2 \end{bmatrix}$$

where  $f$  is the focal length of the camera,  $z$  is the perpendicular distance between the robot and the camera, and  $\beta_1$  and  $\beta_2$  denote the scaling factors in pixels/m. The manipulator Jacobian matrix  $J_e(q)$  was given in Example 1 as

$$J_e(q) = \begin{bmatrix} -l_1s_1 - l_2s_{12} & -l_2s_{12} \\ l_1c_1 + l_2c_{12} & l_2c_{12} \end{bmatrix}$$

Thus, the Jacobian matrix from the joint space to image space is obtained as

$$J(q) = J_I(q)J_m(q) = \frac{f}{z-f} \begin{bmatrix} \beta_1 & 0 \\ 0 & \beta_2 \end{bmatrix} \begin{bmatrix} -l_1s_1 - l_2s_{12} & -l_2s_{12} \\ l_1c_1 + l_2c_{12} & l_2c_{12} \end{bmatrix}$$

Therefore, kinematic uncertainty exists when one or more of the constants  $l_1$ ,  $l_2$ ,  $f$ ,  $z$ ,  $\beta_1$  and/or  $\beta_2$  are uncertain.

For simplicity, we will focus on Example 1 as a kinematically uncertain robot in the simulations and experiments studies in Section5.

## 5. Simulations and Experiments

In this section, simulations as well as experiments are conducted to compare the position and force tracking performance of our proposed scheme with that of the adaptive teleoperation control scheme in [6] which is meant to deal with dynamic uncertainties but not kinematic uncertainties.

### 5.1 Simulations

In the simulations both the master and the slave are considered to be two-DOF, two-link, revolute-joint planar robots as shown in Example1. For simplicity, it is assumed that the robots are in a horizontal plane such that gravity can be ignored. The parameters of the robots, operator, environment and the controllers are given in

Table1 and Table 2.

Table 1. Parameters of the master, slave, operator and environment

$l_1$	$l_2$	$m_1$	$m_2$	$m_h$
0.5(m)	0.5(m)	4.6(kg)	2.3(kg)	0.2(kg)
$b_h$	$k_h$	$m_e$	$b_e$	$k_e$
50(Nsm <sup>-1</sup> )	1000(Nm <sup>-1</sup> )	0.1(kg)	20(Nsm <sup>-1</sup> )	1000(Nm <sup>-1</sup> )

Table 2. Parameters of the controllers

$\alpha$	$\mathbf{K}_m$	$\mathbf{L}_{md}$	$\mathbf{L}_{mk}$	$\mathbf{K}_s$	$\mathbf{L}_{sd}$	$\mathbf{L}_{sk}$
0.25	200 $\mathbf{I}$	40 $\mathbf{I}$	10 $\mathbf{I}$	200 $\mathbf{I}$	40 $\mathbf{I}$	10 $\mathbf{I}$

### 5.1.1 Simulations in contact motion

In the case of contact motion, for the operator and the environment models, let us take

$$\mathbf{M}_h = m_h \mathbf{I}, \mathbf{B}_h = b_h \mathbf{I}, \mathbf{K}_h = k_h \mathbf{I}, \mathbf{M}_e = m_e \mathbf{I}, \mathbf{B}_e = b_e \mathbf{I}, \mathbf{K}_e = k_e \mathbf{I}$$

where  $m_h, b_h, k_h, m_e, b_e$  and  $k_e$  are the mass, damping, and stiffness coefficients of the operator's hand and the environment models, respectively, and  $\mathbf{I}$  is the identity matrix. Besides, for a realistic simulation, let  $\mathbf{f}_h^*$  rise from a zero value,

$$\mathbf{f}_h^* = [f_{hl}^*, 0]^T = [25(1 - \cos(0.1t)), 0]^T. \text{ Also, take } \mathbf{f}_e^* = [0, 0]^T. \text{ As a result of the above,}$$

the unknown dynamic and kinematic parameter vectors can be expressed as

$$\begin{aligned} \boldsymbol{\theta}_{md} &= [l_2^2(m_2 + m_h), l_1 l_2 m_2, l_1 l_2 m_h, l_1 l_2 b_h, l_2^2 b_h, \\ &\quad l_1 l_2 k_h, l_1^2(m_1 + m_2 + m_h), l_1^2 b_h, f_{h1}^* l_1, f_{h1}^* l_2]^T, \\ \boldsymbol{\theta}_{sd} &= [l_2^2(m_2 + m_e), l_1 l_2 m_2, l_1 l_2 m_e, l_1 l_2 b_e, l_2^2 b_e, l_1 l_2 k_e, l_1^2(m_1 + m_2 + m_e), l_1^2 b_e]^T, \\ \boldsymbol{\theta}_{mk} &= \boldsymbol{\theta}_{sk} = [l_1, l_2]^T. \end{aligned}$$

Then the dynamic and kinematic regressor matrices  $\mathbf{Y}_{md}, \mathbf{Y}_{sd}, \mathbf{Y}_{mk}$  and  $\mathbf{Y}_{sk}$  can be obtained based on Property 1 and Property 3, respectively.

According to Table 1, the actual dynamic and kinematic parameter vectors are



calculated as

$$\boldsymbol{\theta}_{md} = [0.625, 0.575, 0.05, 12.5, 12.5, 250, 1.775, 12.5, 17.5(1 - \cos 0.1t), 17.5(1 - \cos 0.1t)]^T,$$

$$\boldsymbol{\theta}_{sd} = [0.6, 0.575, 0.025, 5, 5, 250, 1.75, 5]^T,$$

$$\boldsymbol{\theta}_{mk} = \boldsymbol{\theta}_{sk} = [0.5, 0.5]^T.$$

The initial values for positions and unknown vectors are randomly set (i.e., some initial estimates are lower than the actual values and some are higher than the actual values):

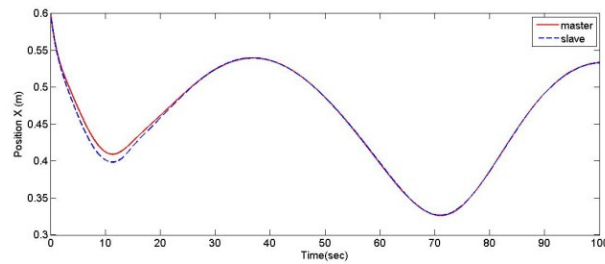
$$\mathbf{x}_m(0) = \mathbf{x}_s(0) = [0.6, 0.2]^T,$$

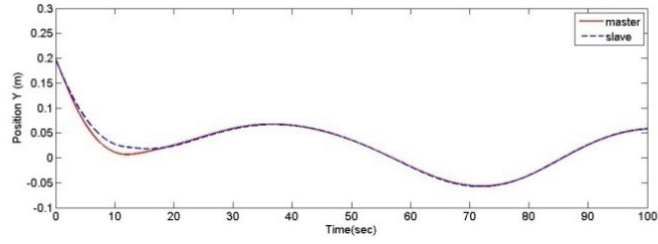
$$\hat{\boldsymbol{\theta}}_{md}(0) = [0.5, 0.6, 0.1, 11, 13, 240, 1, 12, 13, 10]^T,$$

$$\hat{\boldsymbol{\theta}}_{sd}(0) = [0.3, 0.5, 0.02, 6, 6, 240, 2, 4]^T,$$

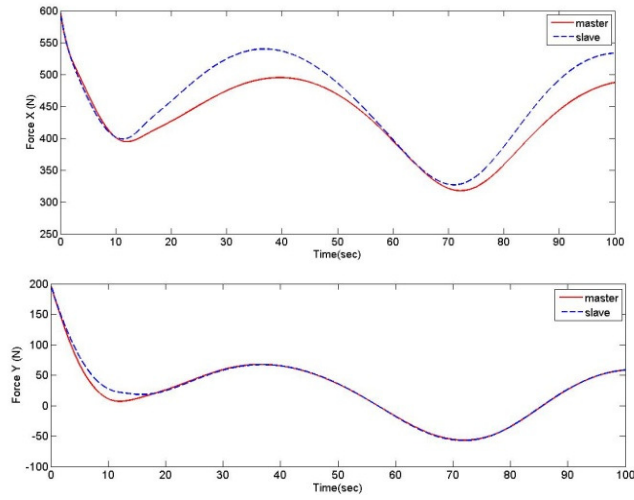
$$\hat{\boldsymbol{\theta}}_{mk}(0) = \hat{\boldsymbol{\theta}}_{sk}(0) = [1, 1]^T.$$

The simulation results in contact motion are shown in Figs. 5-6. As can be seen from Figs. 5(a) and 6(a), using the proposed control scheme the slave tracks the position of the master well in both x-direction and y-direction, while with the adaptive control scheme in [6] the position errors in both x-direction and y-direction are clearly larger. As for the force tracking, from Figs. 5 (b) and 6(b) we can see that in y-direction good force tracking can be achieved with our proposed scheme while that is not the case with the scheme in [6]. Regarding the force error in x-direction, although there are errors in both of the schemes, the force error in the proposed control scheme is smaller overall.



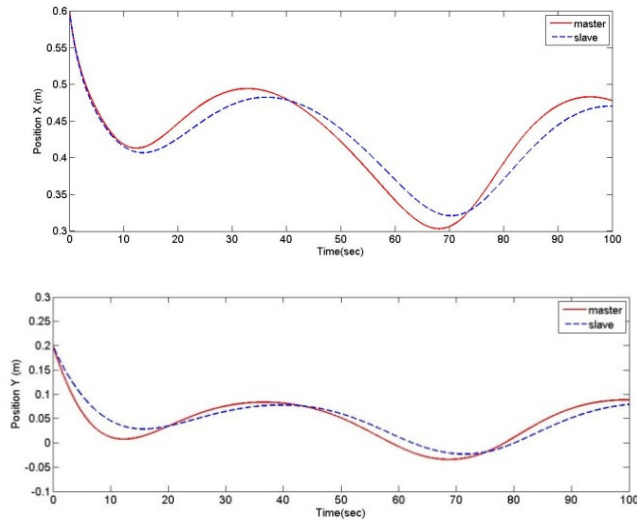


(a) Position tracking performance

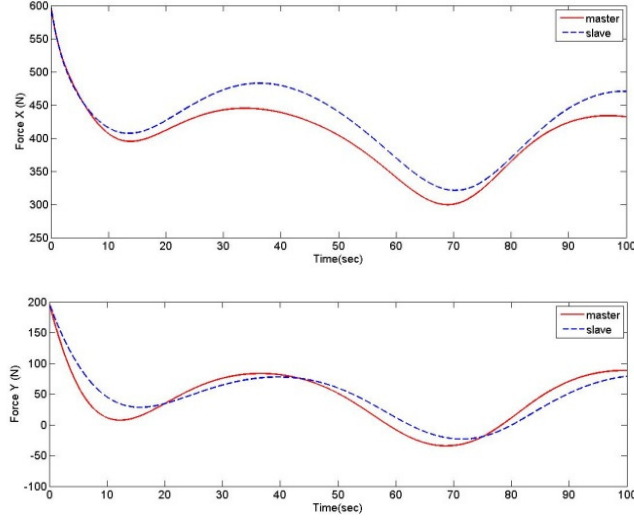


(b) Force tracking performance

Fig.5 Results of our proposed adaptive control scheme (contact motion)



(a) Position tracking performance



(b) Force tracking performance

Fig. 6 Results of the adaptive control scheme in [6] (contact motion)

### 5.1.2 Simulations in free motion

In free motion there is no contact force, i.e.,  $\mathbf{f}_h = 0$  and  $\mathbf{f}_e = 0$ , thus the unknown dynamic parameter vectors can be expressed as  $\boldsymbol{\theta}_{md} = \boldsymbol{\theta}_{sd} = [l_2^2 m_2, l_1 l_2 m_2, l_1^2 (m_1 + m_2)]^T$  while the kinematic parameter vectors are the same as those in contact motion. According to Table 1, the actual dynamic parameter vectors are calculated as  $\boldsymbol{\theta}_{md} = \boldsymbol{\theta}_{sd} = [0.575, 0.575, 1.725]^T$ . The initial positions and the initial estimates of dynamic parameter vectors are randomly set

$$\begin{aligned} \mathbf{x}_m(0) &= [0.6, 0.2]^T, \mathbf{x}_s(0) = [0.5, 0.3]^T, \\ \hat{\boldsymbol{\theta}}_{md}(0) &= \hat{\boldsymbol{\theta}}_{sd}(0) = [0.4, 0.8, 1]^T. \end{aligned}$$

Besides, the actual kinematic parameter vectors and their initial estimate values are kept the same as those in contact motion.

The simulation results in free motion are shown in Figs. 7 and 8. As can be seen from Figs. 7 and 8, using the proposed control scheme the master and the slave can track the position of each other much faster (it takes about 20s for the master and the slave positions to converge) than using the scheme in [6] (it takes about 30s for the

master and the slave positions to converge) in both x-direction and y-direction. This clearly demonstrates that the proposed controller has better position tracking performance.

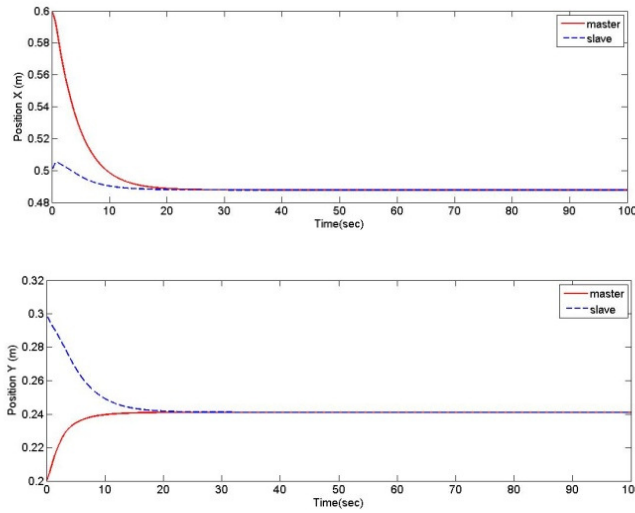


Fig.7 Results of our proposed adaptive control scheme (free motion)

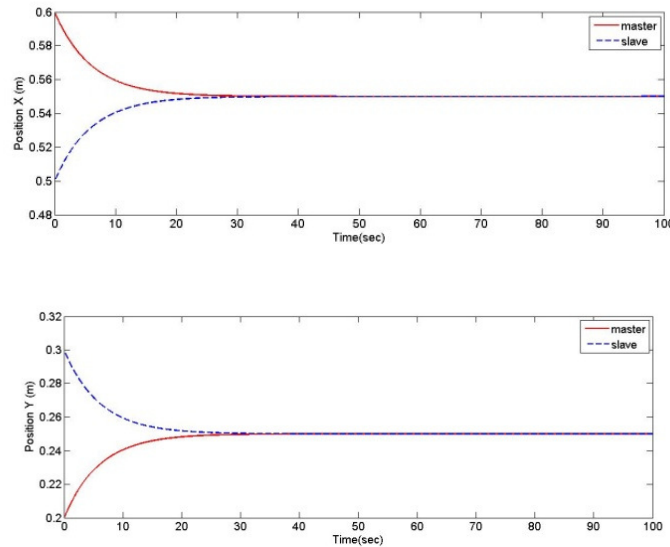


Fig. 8 Results of the adaptive control scheme in [6] (free motion)

## 5.2 Experiments

Experiments are also conducted to compare our proposed adaptive control scheme

with the one in [6]. The experiments are performed with two identical 2-DOF planar rehabilitation robots manufactured by Quanser, Inc., Canada and the experimental setup is shown in Fig.9, where the rehabilitation robot on the left is used as the master and the one on the right is used as the slave.



Fig. 9. Experimental teleoperation setup

More details about rehabilitation robot can be found in [28] and here we focus on its dynamics and kinematics

$$\mathbf{M}_q(\mathbf{q}) = \begin{bmatrix} \alpha_1 & -\frac{1}{2}\alpha_2 \sin(q_1 - q_2) \\ -\frac{1}{2}\alpha_2 \sin(q_1 - q_2) & \alpha_3 \end{bmatrix},$$

$$\mathbf{C}_q(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 0 & \frac{1}{2}\alpha_2 \sin(q_1 - q_2)\dot{q}_2 \\ \frac{1}{2}\alpha_2 \sin(q_1 - q_2)\dot{q}_1 & 0 \end{bmatrix},$$

$$x = l_1 \cos(q_1) + l_2 \sin(q_2)$$

$$y = l_1 \sin(q_1) - l_2 \cos(q_2),$$

$$J = \begin{bmatrix} -l_1 \sin(q_1) & l_2 \cos(q_2) \\ l_1 \cos(q_1) & l_2 \sin(q_2) \end{bmatrix},$$

where  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  are constants. Due to the planar configurations, gravity terms are ignored. Then according to Prop.3, the kinematic parameter vectors for the master and the slave can be found as  $\boldsymbol{\theta}_{mk} = \boldsymbol{\theta}_{sk} = [l_1, l_2]^T$ . Similarly, the dynamic parameter

vectors  $\boldsymbol{\theta}_{md}$  and  $\boldsymbol{\theta}_{sd}$  can be found according to Prop.1. The values of  $l_1$ ,  $l_2$ ,  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  are shown in Table 3, where  $l_1$ ,  $l_2$  are actually measured (values provided by Quanser) and  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  are indeed identified by system identification techniques in [28].

Table 3.Parameters of the rehabilitation robot

$l_1$	$l_2$	$\alpha_1$	$\alpha_2$	$\alpha_3$
0.254(m)	0.2667(m)	0.06256	0.00289	0.04194

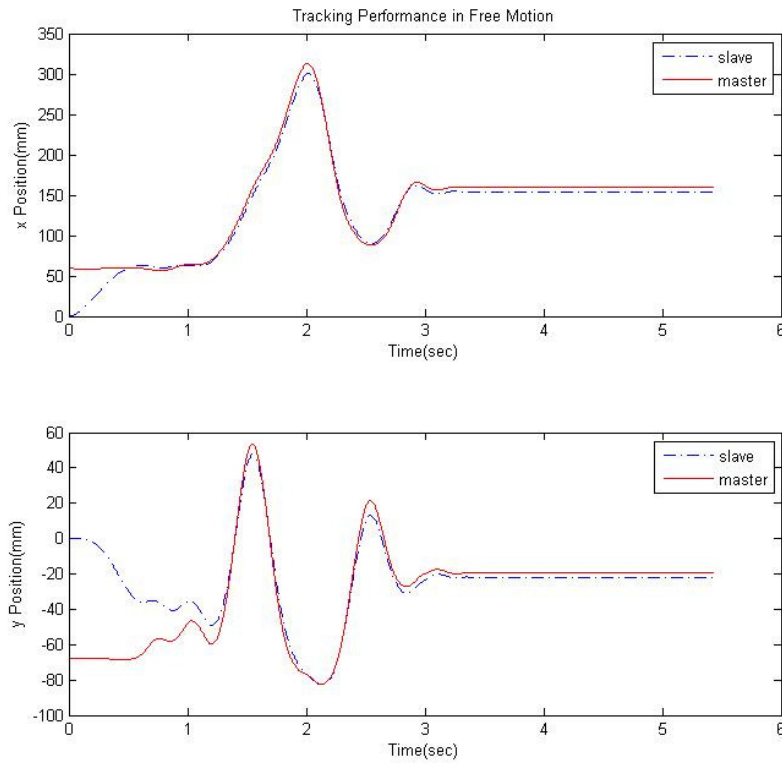
In our experiments all the dynamic and kinematic parameters  $l_1$ ,  $l_2$ ,  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  have inaccurate starting values, compared to the measured or identified values in Table 3. Specifically during implementation, kinematic and dynamic parameter vectors are assigned inaccurate initial values as:  $\hat{\boldsymbol{\theta}}_{md}(0)=1.1*\boldsymbol{\theta}_{md}$ ,  $\hat{\boldsymbol{\theta}}_{sd}(0)=0.9*\boldsymbol{\theta}_{sd}$ ,  $\hat{\boldsymbol{\theta}}_{mk}(0)=0.9*\boldsymbol{\theta}_{mk}$  and  $\hat{\boldsymbol{\theta}}_{sk}(0)=1.1*\boldsymbol{\theta}_{sk}$ .

Moreover, in order to facilitate the experiments, the terms  $\mathbf{K}_m(\Delta\dot{\mathbf{x}}_m + \alpha\Delta\mathbf{x}_m)$  and  $\mathbf{K}_s(\Delta\dot{\mathbf{x}}_s + \alpha\Delta\mathbf{x}_s)$  in the control laws (24)-(25) and in the kinematic update laws (28)-(29) are transformed into another equivalent forms<sup>[17]</sup> as:  $(\mathbf{K}_{mv}\Delta\dot{\mathbf{x}}_m + \mathbf{K}_{mp}\Delta\mathbf{x}_m)$  and  $(\mathbf{K}_{sv}\Delta\dot{\mathbf{x}}_s + \mathbf{K}_{sp}\Delta\mathbf{x}_s)$ . Then the control gains are selected as Table 4.

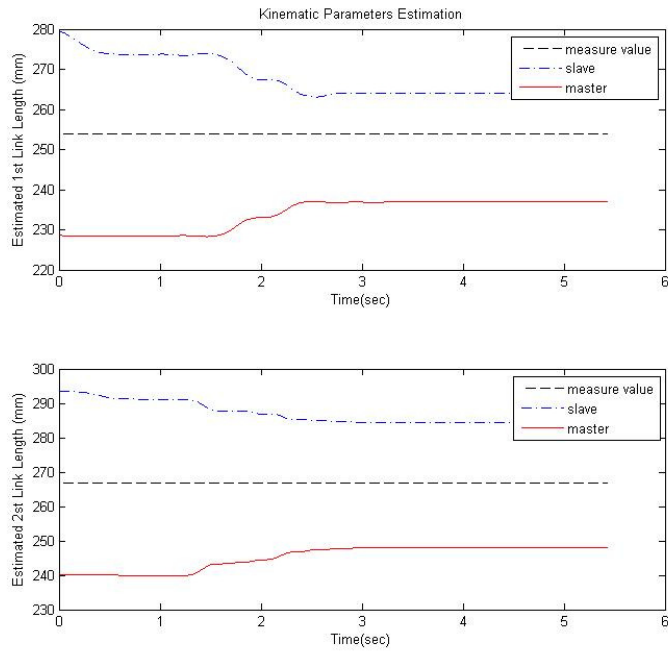
Table 4. Selection of control gains

$\mathbf{K}_{mv}$	$\mathbf{K}_{sv}$	$\mathbf{K}_{mp}$	$\mathbf{K}_{sp}$
3 $\mathbf{I}$	3 $\mathbf{I}$	20 $\mathbf{I}$	35 $\mathbf{I}$
$\mathbf{L}_{md}$	$\mathbf{L}_{sd}$	$\mathbf{L}_{mk}$	$\mathbf{L}_{sk}$
0.01 $\mathbf{I}$	0.01 $\mathbf{I}$	0.022 $\mathbf{I}$	0.022 $\mathbf{I}$

While doing the experiments, the software package QUARC which is developed by Quanser Inc., Canada, is used for real-time control implementation. The sampling time is set to be 0.001 s. The experiment is done by first moving the end-effector of the master robot to a position 60mm away from that of the slave robot in x-direction and about 70mm away from the slave robot in y-direction. The end-effector position of the slave is taken to be 0. The experimental results are shown in Figs.10 and 11.



(a)Position tracking performance



(b) Estimates of the kinematic parameters

Fig. 10 Results of our proposed adaptive control scheme

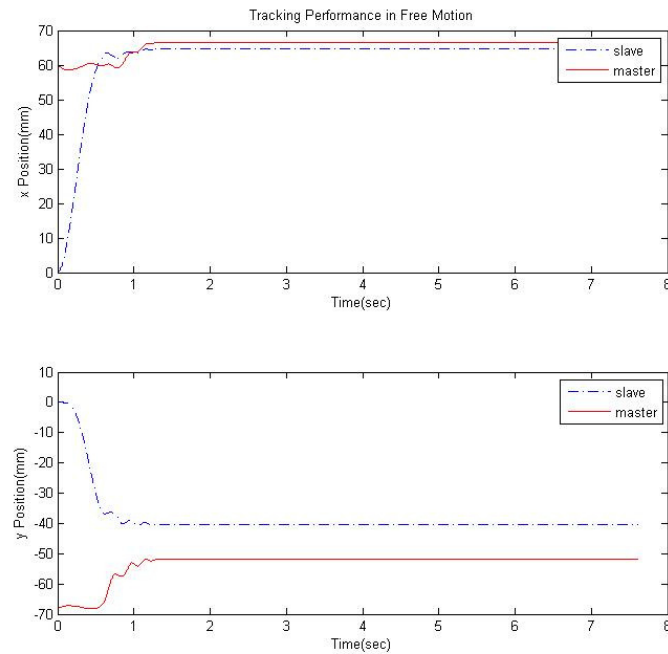


Fig. 11 Results of the adaptive control scheme in [6]



It can be seen from Fig. 10(a) that with our proposed method the position tracking performance in both x-direction and y-direction are good. In x-direction, within 1 second the position error converges to zero and in y-direction it takes slightly over 1 second to converge. However both of them take a few seconds to become stationary. Comparatively, using the adaptive control scheme in [6], the position error in x-direction between the master and the slave is obviously much bigger as shown in Fig.11. On the other hand, as for y-direction, we can also clearly see from Fig.11 that the slave cannot track the position of the master and the position error in this direction can never converge to zero. Generally speaking, the main reason for such a result is that the adaptive control scheme in [6] only aims at dynamic uncertainties but not kinematic uncertainties (i.e.,  $\mathbf{L}_{mk}$  and  $\mathbf{L}_{sk}$  are equal to zero in [6]), while our proposed adaptive control can effectively deal with kinematic uncertainties with the help of the kinematic update as shown in Fig.10(b).

Besides, as we focus on free motion in the experiments, there is no contact force, i.e.,  $\mathbf{f}_h = 0$  and  $\mathbf{f}_e = 0$ . Also, the parameters of the human operator and the environment, i.e.,  $m_h, b_h, k_h, m_e, b_e, k_e, \mathbf{f}_h^*$  and  $\mathbf{f}_e^*$ , are all set to be zeros. Consequently, the force tracking plots are not reported.

**Remark 2:** The kinematic parameters  $l_1$  and  $l_2$  estimations have the tendency to converge to the actual value, however, they do not. What is found during experiments is that in order for the kinematic parameters to converge to the true values,  $\mathbf{L}_{mk}$  and  $\mathbf{L}_{sk}$  have to take a large value, such that  $\boldsymbol{\theta}_{mk}$  and  $\boldsymbol{\theta}_{sk}$  can have larger evolutions during such a short period of time. However larger  $\mathbf{L}_{mk}$  and  $\mathbf{L}_{sk}$  create instability in the system that is harder to handle. So the results shown here are a compromise after a number of trials. This finding is in accordance with the fact that a key point in adaptive control is that the output tracking error is expected to converge to zero

regardless of whether the input is persistently exciting or not. In other words, one should not necessarily need parameter convergence for the convergence of the output tracking error to zero – this is a point evident in the experiments as shown in Fig.10(b).

## 6. Concluding Remarks

In this paper, we have proposed a novel adaptive nonlinear teleoperation control scheme that works without exact knowledge of either the dynamics of the master, the slave, the operator, and the environment, or the kinematics of the master and the slave, allowing for a high degree of flexibility in dealing with unforeseen changes and uncertainties in the master and slave robots' kinematics and dynamics. The stability and asymptotic zero convergence of the closed-loop system is mathematically proven and confirmed through simulations as well as experiments on a bilateral teleoperation test-bed of rehabilitation robots.

Interestingly, we further find that the adaptive control laws proposed in this paper can encompass previous adaptive teleoperation control laws as its special cases. Indeed, when the teleoperator is in free motion, i.e.,  $\mathbf{f}_h = \mathbf{f}_e = 0$ , and no kinematic uncertainty is considered, the control laws (24)-(25) can reduce to

$$\begin{aligned}\boldsymbol{\tau}_m &= \mathbf{Y}_{md} \hat{\boldsymbol{\theta}}_{md} - \mathbf{J}_m^T(\mathbf{q}_m) \mathbf{K}_m (\Delta \dot{\mathbf{x}}_m + \alpha \Delta \mathbf{x}_m) \\ \boldsymbol{\tau}_s &= \mathbf{Y}_{sd} \hat{\boldsymbol{\theta}}_{sd} - \mathbf{J}_s^T(\mathbf{q}_s) \mathbf{K}_s (\Delta \dot{\mathbf{x}}_s + \alpha \Delta \mathbf{x}_s)\end{aligned}$$

which are the same as those in [6] for the free motion case.

The proposed controller is based on the PEB teleoperation architecture. Extending the proposed control to other teleoperation control architectures (e.g., the 4-channel method) and accommodating the combined effects of not only dynamic and kinematic uncertainties but also time delays in the communication channel remain as future work. Besides, in modeling the dynamics of the human operator and the environment, only LTI models are considered for simplicity. The extension of this model to a more general case requires further research.

## Appendix 1: Proof of Property 4

For the master, according to (11), we know that the new inertia and Coriolis/centrifugal matrices are

$$\begin{aligned}\mathbf{M}_m(\mathbf{q}_m) &= \mathbf{M}_{qm}(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\mathbf{J}_m(\mathbf{q}_m), \\ \mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m) &= \mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m).\end{aligned}$$

Thus, we get

$$\dot{\mathbf{M}}_m(\mathbf{q}_m) = \dot{\mathbf{M}}_{qm}(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\dot{\mathbf{M}}_h\mathbf{J}_m(\mathbf{q}_m) + 2\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m).$$

As  $\mathbf{M}_h$  is a constant matrix which has been defined in (7), we further obtain

$$\begin{aligned}\dot{\mathbf{M}}_m(\mathbf{q}_m) &= \dot{\mathbf{M}}_{qm}(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\dot{\mathbf{M}}_h\mathbf{J}_m(\mathbf{q}_m) + 2\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m) \\ &= \dot{\mathbf{M}}_{qm}(\mathbf{q}_m) + 2\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m).\end{aligned}$$

Then, for any  $\forall \xi \in \mathfrak{R}^{n \times 1}$ , we have

$$\begin{aligned}& \xi^T (\dot{\mathbf{M}}_m(\mathbf{q}_m) - 2\mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)) \xi \\ &= \xi^T (\dot{\mathbf{M}}_{qm}(\mathbf{q}_m) + 2\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m) - 2(\mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m) \\ & \quad + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m) + \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{M}_h\dot{\mathbf{J}}_m(\mathbf{q}_m))) \xi \\ &= \xi^T (\dot{\mathbf{M}}_{qm}(\mathbf{q}_m) - 2\mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m) - 2\mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m)) \xi\end{aligned}$$

Using Property 2, we already have

$$\xi^T (\dot{\mathbf{M}}_{qm}(\mathbf{q}_m) - 2\mathbf{C}_{qm}(\mathbf{q}_m, \dot{\mathbf{q}}_m)) \xi = 0.$$

Thus,

$$\xi^T (\dot{\mathbf{M}}_m(\mathbf{q}_m) - 2\mathbf{C}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)) \xi = -2\xi^T \mathbf{J}_m^T(\mathbf{q}_m)\mathbf{B}_h\mathbf{J}_m(\mathbf{q}_m)\xi.$$

Similarly, for the slave, according to (12), we know that

$$\begin{aligned}\mathbf{M}_s(\mathbf{q}_s) &= \mathbf{M}_{qs}(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\mathbf{J}_s(\mathbf{q}_s), \\ \mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s) &= \mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s).\end{aligned}$$

Thus, we get

$$\dot{\mathbf{M}}_s(\mathbf{q}_s) = \dot{\mathbf{M}}_{qs}(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\dot{\mathbf{M}}_e\mathbf{J}_s(\mathbf{q}_s) + 2\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s).$$

As  $\mathbf{M}_e$  is a constant matrix which has been defined in (8), we further obtain

$$\begin{aligned}\dot{\mathbf{M}}_s(\mathbf{q}_s) &= \dot{\mathbf{M}}_{qs}(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\dot{\mathbf{M}}_e\mathbf{J}_s(\mathbf{q}_s) + 2\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s) \\ &= \dot{\mathbf{M}}_{qs}(\mathbf{q}_s) + 2\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s).\end{aligned}$$

Then, for any  $\forall \xi \in \mathfrak{R}^{n \times 1}$ , we have

$$\begin{aligned}& \xi^T (\dot{\mathbf{M}}_s(\mathbf{q}_s) - 2\mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s))\xi \\ &= \xi^T (\dot{\mathbf{M}}_{qs}(\mathbf{q}_s) + 2\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s) - 2(\mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s) \\ & \quad + \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s) + \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{M}_e\dot{\mathbf{J}}_s(\mathbf{q}_s)))\xi \\ &= \xi^T (\dot{\mathbf{M}}_{qs}(\mathbf{q}_s) - 2\mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s) - 2\mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s))\xi\end{aligned}$$

Using Property 2, we already have

$$\xi^T (\dot{\mathbf{M}}_{qs}(\mathbf{q}_s) - 2\mathbf{C}_{qs}(\mathbf{q}_s, \dot{\mathbf{q}}_s))\xi = 0.$$

Thus, we can obtain

$$\xi^T (\dot{\mathbf{M}}_s(\mathbf{q}_s) - 2\mathbf{C}_s(\mathbf{q}_s, \dot{\mathbf{q}}_s))\xi = -2\xi^T \mathbf{J}_s^T(\mathbf{q}_s)\mathbf{B}_e\mathbf{J}_s(\mathbf{q}_s)\xi.$$

## Acknowledgements

This research was supported by the National Natural Science Foundation of China under grant 61305104, the Scientific and Technical Supporting Programs of Sichuan Province under grant 2013GZX0152, the Key Scientific Research Fund Project of Xihua University under grant Z1220934 and the Natural Sciences and Engineering Research Council of Canada.

## References

- [1] P. Hokayem, M. Spong, Bilateral teleoperation: an historical survey, *Automatica*, 42(12)(2006) 2035-2057.
- [2] C. Passenberg, A. Peer, M. Buss, A survey of environment-operator-and task-adapted controllers for teleoperation systems, *Mechatronics*, 20 (7) (2010)787 - 801.
- [3] H. Lee, M. J. Chung, Adaptive controller of a master-slave system for transparent teleoperation, *Journal of Robotic systems*, 15 (8)(1998) 465-475.
- [4] M. Shi, G. Tao, H. Liu, Adaptive control of teleoperation systems, *Journal of X-Ray Science and Technology*, 10 (1-2)(2002)37-57.
- [5] K. B. Fite, M. Goldfarb, A. Rubio, Loop shaping for transparency and stability robustness in time-delayed bilateral telemanipulation, *Journal of Dynamic Systems, Measurement and Control*, 126 (3)(2004)650-656.

- [6] J.H.Ryu, and D. S.Kwon,A novel adaptive bilateral control scheme using similar closed-loop dynamic characteristics of master/slave manipulators, *Journal of Robotic Systems*, 2001, 18 (9) 533-543.
- [7] N. V.Q.Hung,T.Narikiyo,H. D.Tuan,Nonlinear adaptive control of master-slave system in teleoperation, *Control Engineering Practice*, 11 (1)(2003)1-10.
- [8] N.Chopra, M.Spong,R.Lozano,Synchronization of bilateral teleoperators with time delay,*Automatica*,44 (8)(2008) 2142-2148.
- [9] E.Nuño,R.Ortega,L.Basañez,An adaptive controller for nonlinear teleoperators, *Automatica*, 46 (1)(2010) 155-159.
- [10]E.Nuñoa, L.Basañez, R.Ortega, Passivity based control for bilateral teleoperation: tutorial,*Automatica*,47(3)(2011) 485-495.
- [11]X. Liu, M. Tavakoli, Adaptive control of teleoperation systems with linearly and nonlinearly parameterized dynamic uncertainties, *Journal of Dynamic System, Measurement, and Control*, 134 (2) (2012)1015-1024.
- [12]W.Zhu,S. E.Salcudean,Stability guaranteed teleoperation: an adaptive motion/force control approach,*IEEE transactions on automatic control*, 45 (11)(2000) 1951-1969.
- [13]P.Malysz,S.Sirouspour,Nonlinear and filtered force/position mapping in bilateral teleoperation with application to enhanced stiffness discrimination,*IEEE Transaction on Robotics*, 25 (5)(2009)1134-1149.
- [14]X. Liu, M. Tavakoli, Adaptive inverse dynamics four-channel control of uncertain nonlinear teleoperation systems, *Advanced Robotics*, 25 (14) (2011) 1729-1750.
- [15]D. H.Kim, K. H.Cook,J. H.Oh,Identification and compensation of robot kinematic parameter for positioning accuracy improvement,*Robotica*, 9 (1)(1991)99-105.
- [16]W. E. Dixon,Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics. *IEEE Transactions on Automatic Control*, 52(3)(2007):488-493.
- [17]C. C.Cheah,C. Liu, J. J. E.Slotine,Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models, *IEEE Transactions on Automatic Control*, 51(6)(2006): 1024-1029.
- [18]C. C.Cheah,C. Liu, J. J. E.Slotine, Adaptive tracking control for robots with unknown kinematic and dynamic properties, *The International Journal of Robotics Research*, 25(3)2006:283-296.

- [19] P. Arcara, C. Melchiorri, Control schemes for teleoperation with time delay: A comparative study. *Robotics and Autonomous Systems*, 38 (1) (2002) 49-64.
- [20] R. Kelly, V. Santibanez, A. Loria, Control of Robot Manipulators in Joint Space, Springer, Berlin, Germany, 2005.
- [21] M. Tavakoli, A. Aziminejad, R. V. Patel, M. Moallem, High-fidelity bilateral teleoperation systems and the effect of multimodal haptics, *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, 37 (6) (2007) 1512-1528.
- [22] J. J. E. Slotine, W. Li, Applied nonlinear control, Prentice-Hall, Englewood Cliffs, NJ, 1991.
- [23] P. A. Ioannou and J. Sun, Robust Adaptive Control. Upper Saddle River, NJ: Prentice Hall, 1997.
- [24] D. Braganza, W. E. Dixon, D. M. Dawson, and B. Xian. Tracking control for robot manipulators with kinematic and dynamic uncertainty. *International Journal of Robotics and Automation*, 23 (2): 117-126, 2008.
- [25] S. Chiaverini. Singularity-robust task-priority redundancy resolution for realtime kinematic control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 13 (3): 398-410, 1997.
- [26] A. Shademan, Uncalibrated vision-based control and motion planning of robotic arms in unstructured environments, Ph. D dissertation, University of Alberta, 2012.
- [27] G. H. S. Hutchinson and P. Cork, A tutorial on visual servo control, *IEEE Transactions on Robot and Automation*, 12(5) (1996) 651-670.
- [28] Matthew Dyck, Mahdi Tavakoli, Measuring the dynamic impedance of the human arm without a force sensor, *IEEE International Conference on Rehabilitation Robotics*, Seattle, Washington USA, 2013, pp. 978-985.